Good Data, Large Data, or No Data? Comparing Three Approaches in Developing Research Aspect Classifiers for Biomedical Papers

Shreya Chandrasekhar, Chieh-Yang Huang, and Ting-Hao 'Kenneth' Huang
Pennsylvania State University, University Park, PA, USA
{sxc6175, chiehyang, txh710}@psu.edu

Abstract

The rapid growth of scientific publications, particularly during the COVID-19 pandemic, emphasizes the need for tools to help researchers efficiently comprehend the latest advancements. One essential part of understanding scientific literature is research aspect classification, which categorizes sentences in abstracts to Background, Purpose, Method, and Finding. In this study, we investigate the impact of different datasets on model performance for the crowd-annotated CODA-19 research aspect classification task. Specifically, we explore the potential benefits of using the large, automatically curated PubMed 200K RCT dataset and evaluate the effectiveness of large language models (LLMs), such as LLaMA, GPT-3, ChatGPT, and GPT-4. Our results indicate that using the PubMed 200K RCT dataset does not improve performance for the CODA-19 task. We also observe that while GPT-4 performs well, it does not outperform the SciBERT model fine-tuned on the CODA-19 dataset, emphasizing the importance of a dedicated and task-aligned datasets dataset for the target task. Our code is available at https://github.com/Crowd-AI-Lab/CODA-19-exp.

1 Introduction

The rapid growth of scientific publications, particularly during the COVID-19 pandemic, has made it increasingly challenging to keep up with the latest research advancements. To address this issue, researchers have developed various systems, such as search engines (Lahav et al., 2022; Zhang et al., 2020), visualization tools (Hope et al., 2020; Tu et al., 2020), claim verification systems (Wadden et al., 2020; Pradeep et al., 2021), question-answering systems (Frisoni et al., 2022), and summarization techniques (Meng et al., 2021). These tools help efficiently comprehend publications by organizing large amounts of information.

One critical aspect of understanding scientific literature is the classification of sentences within abstracts according to their research aspects, such as Background, Purpose, Method, and Finding. This is particularly crucial in the biomedical domain, where abstracts tend to be longer and more complex. With the annotated aspects, readers can quickly grasp the key aspects of a scientific paper. For example, FacetSum (Meng et al., 2021) summarizes papers in four aspects to quickly convey the information. Several datasets have been proposed for research aspect classification, including PubMed 200K RCT (Dernoncourt and Lee, 2017), PubMed-PICO-Detection (Jin and Szolovits, 2018), CODA-19 (Huang et al., 2020), and more.

In this paper, we focus on the **research aspect classification task using crowd-annotated CODA-19** (Huang et al., 2020) and explore the impact of different datasets on model performance. Specifically, we investigate whether the **automatically curated large dataset** (PubMed 200K RCT (Dernoncourt and Lee, 2017)) can help the target task despite its shifted data distribution from the target task. Additionally, we examine whether large language models (LLMs), trained on a massive general textual corpus, can solve the task with **limited or no task-specific data provided**. In particular, we evaluate six LLMs, including three open-sourced models (LLaMA-65B (Touvron et al., 2023), MPT (Team, 2023), and Dolly-12B (Databricks, 2023)) and three closed models (GPT-3 (Brown et al., 2020), ChatGPT (OpenAI, 2022), and GPT-4 (OpenAI, 2023)).

Our study suggests that using PubMed 200K **does not** improve the performance of the CODA-19 research aspect classification task, regardless of the approach used. We experimented with (*i*) training purely on PubMed data, (*ii*) simply-mixing PubMed and CODA-19 data, (*iii*) upsampling CODA-19 data and mixing it with PubMed data to address the data imbalance issue, and (*iv*) using a two-staged training approach where the model is trained on PubMed and CODA-19 sequentially.

However, none of these approaches could improve the performance of the target task. We hypothesize this is due to the use of the SciBERT (Beltagy et al., 2019), which has pre-trained on papers from scientific domains and thus reduces the advantage of incorporating PubMed 200K. Our results also showed that although GPT-4 performed well in both zero-shot and few-shot settings, **it was not able to outperform** the SciBERT model fine-tuned on the CODA-19 dataset. This finding suggests that having a dedicated dataset that aligns well with the target task is still important.

2 Related Work

Moradi et al. (2021) compared BioBERT and GPT-3 in a few-shot learning setting for sentence classification tasks and found that neither model could outperform fully fine-tuned models. For biomedical information extraction tasks, Jimenez Gutierrez et al. (2022) observed that GPT-3 could not outperform the fine-tuned RoBERTa-large model, whereas Agrawal et al. (2022) reported promising performance for GPT-3. Gururangan et al. (2020) investigated whether adapting pre-trained models to the domain of the target task could help, and concluded that domain-adaptive pretraining is always helpful, highlighting the need for task-aligned data and training strategies. In this paper, we focus on the aspect classification task on CODA-19 and examine whether slightly domain-shifted large datasets and LLMs can improve performance.

3 Methodology

In this section, we describe models trained to explore the impact of different datasets and training strategies on the CODA-19 research aspect classification task. For detailed model training details such as hyperparameters, please refer to Appendix B.

3.1 Good Data: CODA-19

We use CODA-19 (Huang et al., 2020) as our Good Data. CODA-19 consists of clause-level aspect annotations for abstracts from medical papers, including *Background, Purpose, Method, Finding/Contribution, and Other*. It contains 137K/15K/15K samples for train/validation/test sets. We fine-tune SciBERT (Beltagy et al., 2019) on the original CODA-19 to create $\text{SciBERT}_{\text{CODA}}$ and on the position-encoded CODA-19 to create $\text{SciBERT}_{\text{CODA+Pos}}$ (see Section 3.4).

3.2 Large Data: PubMed

PubMed 200K (Dernoncourt and Lee, 2017) provides sentence-level research aspects based on structured abstracts. It contains 2.2M/29K/29K samples for the train/validation/test set. Despite including a larger amount of data, the distribution is slightly different from the CODA-19 task. We first create two models, $\text{SciBERT}_{\text{PubMed}}$ and $\text{SciBERT}_{\text{PubMed+Pos}}$, by fine-tuning SciBERT on PubMed 200K. We also explore combining CODA-19 and PubMed with three strategies: (*i*) simply-mixing the two datasets to create $\text{SciBERT}_{\text{Mix+Pos+S}}$; (*ii*) upsampling CODA-19 ten times to balance the data size and mixing it with PubMed to create $\text{SciBERT}_{\text{Mix+Pos+U}}$; and (*iii*) two-staged training, where the model is fine-tuned on PubMed and CODA-19 sequentially to create $\text{SciBERT}_{\text{Mix+Pos+T}}$.

3.3 No Data: LLMs

In the No-Data setting, the goal is to classify the research aspect of abstract sentences using LLMs with limited or no task-specific training data. LLMs are trained on a massive amount of web data, which has a different distribution compared to our target dataset. To explore the performance of LLMs in this scenario, we use zero-shot and few-shot classification with three open-sourced models: LLaMA-65B (Touvron et al., 2023), MPT-7B (Team, 2023), and Dolly-12B (Databricks, 2023); and three closed models: GPT-3 (Brown et al., 2020), ChatGPT (OpenAI, 2022), and GPT-4 (OpenAI, 2023).

3.4 Position Encoding

To predict a research aspect for a sentence, the position of the sentence within the whole abstract is important. Prior work such as Dernoncourt and Lee (2017) used CRF to model the relationship between sentences. In this paper, we incorporate position information by simply adding a position encoding to the beginning of each sentence in the form of "[POSITION=0.38]", where the number represents the normalized sentence position, i.e., sentence_id / #sentences_in_abstract. Examples of position-encoded data can be found in Appendix A.

4 Experiments and Results

We conducted three experiments to (*i*) verify whether our fine-tuned SciBERT model can outperform the model proposed in the original PubMed

Setting		Background			Methods			Objective			Results			Conclusions			All	
Support		2,663			9,751			2,377			10,276			4,426			29,493	
Model	FTR	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	ACC	F1
SciBERT	T	.712	.734	.723	.935	.955	.945	.787	.655	.715	.918	.932	.925	.858	.848	.853	.887	.832
SciBERT	T+P	**.742**	**.824**	**.781**	.945	**.970**	.958	**.796**	**.679**	**.733**	**.959**	.940	.950	.946	**.946**	**.946**	**.919**	**.873**
bi-ANN+CRF	T	.707	.811	.756	**.955**	.965	**.960**	.771	.653	.707	.956	**.948**	**.952**	**.946**	.937	.942	.916	.863

Table 1: Performance on PubMed 200k (Dernoncourt and Lee, 2017), where **best** and second-best results are highlighted. The FTR column shows the feature used for the model, where T and T+P stand for text-only and position-encoded text. According to the overall accuracy and F1 score, our SciBERT$_{\text{PubMed+Pos}}$ performs the best.

	Setting		Background			Purpose			Method			Finding			Other			ALL	
	Support		5,062			821			2,140			6,890			562			15,475	
Data	Model	FTR	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	ACC	F1
C	SciBERT	T	.700	.808	.750	.656	.588	.620	.716	.635	.673	.802	.743	.772	.797	**.867**	.830	.746	.729
C	SciBERT	T+P	.825	.794	.809	.638	**.655**	**.647**	**.741**	.665	**.701**	.823	.867	.845	**.819**	.843	.831	.803	**.767**
P	SciBERT	T	.710	.433	.538	.415	.167	.238	.371	**.757**	.498	.678	.756	.715	-	-	-	.592	.497
P	SciBERT	T+P	**.854**	.415	.559	.233	.251	.241	.362	.720	.482	.750	.857	.800	-	-	-	.630	.520
Mix$_s$	SciBERT	T+P	.762	.815	.788	.674	.446	.537	.677	.640	.658	.824	.843	.833	.808	.642	.716	.777	.706
Mix$_u$	SciBERT	T+P	.802	**.822**	.812	.669	.585	.624	.734	.636	.681	.826	.857	.841	.812	.820	.816	.799	.755
Mix$_t$	SciBERT	T+P	.816	.812	**.814**	**.687**	.574	.625	.736	.654	.693	**.827**	**.870**	**.848**	.807	.865	**.835**	**.805**	.763
C	BERT	T+P	.846	.761	.801	.626	.646	.636	.702	.637	.668	.803	**.879**	.839	.803	.847	.824	.793	.754
Mix$_t$	BERT	T+P	.828	.775	.801	.663	.639	**.651**	.715	.639	.675	.808	.872	.839	.809	.854	.831	.795	.759

Table 2: Performance on CODA-19 (Huang et al., 2020), where **best** and second-best results are highlighted. The Data column specifies the training data, C: CODA-19, P: PubMed 200K, Mix$_s$: simply-mixing, Mix$_u$: upsampling, and Mix$_t$: two-staged training. The FTR column shows the feature used for the model, T: text-only, and T+P: position-encoded text. According to the overall performance, SciBERT$_{\text{CODA+Pos}}$ and SciBERT$_{\text{Mix+Pos+T}}$ achieve the best performance.

paper (Dernoncourt and Lee, 2017); (*ii*) compare models trained on good data and large data, aiming to examine whether a large automatically curated dataset can enhance performance; (*iii*) benchmark the performance of open-sourced and closed LLMs for the CODA-19 aspect classification task and compare them with the best-performing SciBERT model.

4.1 Verifying the PubMed Model

In this experiment, we aim to assess the effectiveness of our fine-tuned PubMed model by comparing it with the model reported in the original PubMed paper (Dernoncourt and Lee, 2017).

Experimental Setup. We evaluate two PubMed models in our study: SciBERT$_{\text{PubMed}}$ and SciBERT$_{\text{PubMed+Pos}}$. To compare with the reported model, we apply them to PubMed 200K test set, which contains 29,493 samples, to predict the PubMed label set (*Background, Methods, Objective, Results, and Conclusions*). We report precision, recall, and F1 scores for each label and calculate the accuracy and macro F1 as overall metrics. Note that the micro F1 score provided in the original PubMed paper is equivalent to accuracy since each instance is assigned with only one label. To obtain the macro F1 score, we average the F1 scores across all labels.

Results. The results shown in Table 1 demonstrate that our SciBERT$_{\text{PubMed+Pos}}$ model outperforms the bi-ANN+CRF model (Dernoncourt and Lee, 2017) in both of accuracy (0.919 vs. 0.916) and macro F1 score (0.873 vs. 0.863). These findings suggest that, despite not considering a whole abstract simultaneously, we can achieve competitive performance by incorporating position encoding. Based on these results, we use SciBERT$_{\text{PubMed+Pos}}$ for further comparisons.

4.2 Good Data and Large Data

In this experiment, we aim to compare whether using a dataset with a larger amount of samples but a slight domain shift could help improve the performance of the CODA-19 aspect classification

	Setting		Background			Purpose			Method			Finding			Other			ALL	
	Support		250			250			250			250			250			1,250	
Data	Model	FTR	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	ACC	F1
C	SciBERT	T+P	.492	.784	**.605**	.879	**.696**	**.777**	.710	.684	**.697**	.669	.632	**.650**	**.983**	.696	**.815**	**.698**	**.709**
P	SciBERT	T+P	.241	.080	.120	.633	.152	.245	.288	**.836**	.428	.441	.672	.532	-	-	-	.348	.332
-	LLaMA	Zero	.212	.160	.182	**1.000**	.012	.024	**1.000**	.004	.008	.700	.056	.104	.180	.748	.291	.196	.122
-		Few	.402	.556	.466	.800	.240	.369	.663	.228	.339	.484	.596	.534	.556	.968	.707	.518	.483
-	MPT	Zero	.229	**.960**	.370	.000	.000	.000	.923	.048	.091	**1.000**	.004	.008	.746	.564	.642	.315	.222
-		Few	.230	.304	.262	**1.000**	.008	.016	.667	.008	.016	.289	.824	.428	.748	.604	.668	.350	.278
-	Dolly	Zero	.208	.956	.342	.000	.000	.000	.000	.000	.000	.304	.096	.146	.522	.048	.088	.220	.115
-		Few	.462	.048	.087	.615	.032	.061	.000	.000	.000	.230	**.904**	.367	.652	.592	.621	.315	.227
-	GPT-3	Zero	.435	.628	.514	.838	.268	.406	.562	.580	.571	.770	.348	.479	.543	.952	.692	.555	.532
-		Few	**.604**	.408	.487	.691	.492	.575	.783	.404	.533	.623	.528	.571	.443	**.996**	.613	.566	.556
-	ChatGPT	Zero	.409	.416	.413	.833	.200	.323	.661	.436	.525	.645	.392	.488	.401	.992	.571	.487	.464
-		Few	.446	.516	.479	.833	.200	.323	.663	.464	.546	.621	.472	.536	.461	.988	.628	.528	.502
-	GPT-4	Zero	.579	.560	.569	.749	.548	.633	.562	.692	.620	.800	.416	.547	.615	.952	.747	.634	.623
-		Few	.570	.588	.579	.888	.444	.592	.630	.668	.649	.679	.600	.637	.646	.984	.780	.657	.647

Table 3: Performance on a randomly sampled subset of CODA-19 (Huang et al., 2020). We highlight the **best** and the second-best results. The Data column specifies the training data: C: CODA-19 and P: PubMed. The FTR column shows the feature used for the model, T: text-only, T+P: position-encoded text, zero: zero-shot learning, and few: few-shot learning. Even the best-performing LLM, GPT-4, does not outperform $\text{SciBERT}_{\text{CODA+Pos}}$, showing the need for task-aligned data. Open-sourced models (LLaMA, MPT, and Dolly) currently show lower performance compared to closed models.

task.

Experimental Setup. We evaluate seven different models on CODA-19 test set, which consists of 15,475 samples across five labels: *Background, Purpose, Method, Finding, and Other*. The models include two trained on the CODA-19 dataset ($\text{SciBERT}_{\text{CODA}}$ and $\text{SciBERT}_{\text{CODA+Pos}}$), two trained on the PubMed dataset ($\text{SciBERT}_{\text{PubMed}}$ and $\text{SciBERT}_{\text{PubMed+Pos}}$), and three trained on a mix of the CODA-19 and PubMed datasets ($\text{SciBERT}_{\text{Mix+Pos+S}}$, $\text{SciBERT}_{\text{Mix+Pos+U}}$, and $\text{SciBERT}_{\text{Mix+Pos+T}}$). For the PubMed models, we map the predicted labels to the corresponding CODA-19 label using a predefined mapping function: *Background* to *Background*, *Methods* to *Method*, *Objective* to *Purpose*, *Results* to *Finding*, and *Conclusions* to *Finding*. Note that *Other* (unclear or confusing sentences) in CODA-19 does not have a corresponding label in PubMed. We report precision, recall, and F1 scores for each label and calculate the accuracy and macro F1 as overall metrics. For the PubMed models, the macro F1 score is averaged over the four valid labels.

Results. The results of our experiment are presented in Table 2. We find that $\text{SciBERT}_{\text{CODA+Pos}}$ and $\text{SciBERT}_{\text{Mix+Pos+T}}$ achieve the highest accuracy (0.803 and 0.805) and macro F1 (0.767 and 0.763) scores, respectively, and outperform the other models. The best performing PubMed model ($\text{SciBERT}_{\text{PubMed+Pos}}$) does not show any improvement over the performance on the CODA-19 test set (accuracy: 0.630 and macro F1: 0.520); and is particularly weak in identifying the *Purpose* label (*Purpose* F1: 0.241). When comparing the different mixing strategies, the models trained with simply-mixing and upsampling perform even worse (accuracy: 0.777/0.799 and macro F1: 0.706/0.755). Although the two-staged mixing strategy does not yield lower scores, it only achieves the same results as $\text{SciBERT}_{\text{CODA+Pos}}$.

Since SciBERT is pre-trained on a huge amount of scientific papers, with 82% of the papers belonging to the biomedical domain (Beltagy et al., 2019), we also compare the performance of two approaches, the two-staged mixing strategy, and pure fine-tuning, using BERT (Devlin et al., 2019). This allows us to eliminate the impact of SciBERT's pre-training. The results, shown in the last two rows of Table 2, indicate that the two-staged mixing strategy does not yield the expected improvement (with overall F1 scores of 0.754 for $\text{BERT}_{\text{CODA+Pos}}$ and 0.759 for $\text{BERT}_{\text{Mix+Pos+T}}$). Despite this, the overall score for SciBERT remains higher. We hypothesize that the two-staged mixing strategy with the classification objective may not be the best way for adapting the model to a specific domain. Therefore,

when large-scale pretraining, such as training SciBERT, is not available, having a dedicated dataset that aligns well with the target task is still important and will give the best performance.

4.3 Comparison of No-Data

We investigate whether recent advances in LLMs can solve the CODA-19 aspect classification task.

Experimental Setup. Due to resource limitations, we experiment on a subset of CODA-19 test set, where 250 samples for each of the five labels are randomly selected, resulting in 1,250 samples.

We first include both $\text{SciBERT}_{\text{CODA+Pos}}$ and $\text{SciBERT}_{\text{PubMed+Pos}}$ for comparison and report scores obtained by running on the evaluation set specifically for this experiment. Additionally, we include six LLMs for comparison, *i.e.*, LLaMA-65B, MPT-7B, Dolly-12B, GPT-3, ChatGPT, and GPT-4, each in both the zero-shot and few-shot settings. A total of 14 models are included. Note that out of the six LLMs, LLaMA-65 is the only one not trained with instruction-following tasks. We use the crowd workers' annotation guidelines from CODA-19 (Huang et al., 2020) as our zero-shot prompt (see Table 7 in Appendix C for the actual prompt). For the few-shot setting, we assume a scenario where users annotate a single abstract as an example. Thus, we randomly select one abstract from CODA-19 train set that contains four primary labels (*Background, Purpose, Method, and Finding*). To avoid LLMs incorrectly considering the order of samples as information, we shuffle the samples in the few-shot prompt (see Table 8 in Appendix C for the actual prompt). To query each model, we use the parameters described in Appendix B. Once we obtain the generated texts, we use regex to parse the predicted label. When the predicted label is not in the CODA-19 label sets or is missing, we treat it as *Other*. We report the per-label precision, recall, and F1 score, as well as the overall accuracy and macro F1 score.

Results. The results of our experiment are presented in Table 3. $\text{SciBERT}_{\text{CODA+Pos}}$ remains the best-performing model with an accuracy of 0.698 and a macro F1 score of 0.709. We observe that the zero-shot setting of LLaMA-65B performs poorly, possibly due to the model not being trained for any instruction-following tasks. The majority of its predictions are on *Background* and *Other* labels, leading to very low recall for *Purpose*, *Method*, and *Finding* labels (0.012, 0.004, and 0.056, respectively). Such biased prediction issues also happen for MPT and Dolly in both zero-shot and few-shot settings even though they are trained with instruction-following datasets, suggesting that there is still a huge performance gap between open-sourced models and closed models.

When comparing closed models, our results show that ChatGPT performs worse than GPT-3, possibly due to its optimization toward human-favored conversation. On the other hand, GPT-4 outperforms GPT-3 by a large margin but is still unable to outperform $\text{SciBERT}_{\text{CODA+Pos}}$. While we believe that LLMs have the potential to outperform $\text{SciBERT}_{\text{CODA+Pos}}$ in the future, our current results emphasize the importance of having a dedicated dataset that aligns well with the target task.

5 Conclusion

In this paper, we investigate the impact of different datasets and LLMs on the CODA-19 research aspect classification task. Our findings show that using a huge but slightly different dataset, PubMed 200K, does not help improve performance. LLMs, trained with massive web corpus, are also unable to outperform the SciBERT trained on the target dataset, emphasizing the importance of task-aligned datasets. In the future, we will explore methods for the model to consider the context and predict all sentences within a single abstract at once.

Acknowledgements

We thank all anonymous reviewers' constructed feedback to improve this work.

Limitations

One important aspect of achieving optimal performance when using LLMs is the design of a high-quality prompt. In this study, we consider both zero-shot and few-shot learning scenarios, which assume no or very limited task-specific data. However, iteratively refining the prompt over time to obtain the best-performing prompt may break the zero-shot or few-shot scenario. Moreover, the final prompt used in this study is specifically designed to guide the crowd workers in the annotation process of the CODA-19 dataset, with frequently asked questions (FAQs) refined over time to address workers' confusion. In a real-world scenario, users would not have access to such helpful FAQs when

working on a new task. Therefore, the performance of LLMs may be lower in practice.

Also, LLMs are susceptible to a data leakage problem due to their training with Internet data. For example, ChatGPT is known to have been trained on Internet data prior to September 2021. Considering that the CODA-19 dataset was released in July 2020, with its train, validation, and test sets made publicly available, there is a possibility that some closed models have seen the exact test instances, leading to an unfair comparison. Since the training data are not disclosed for the closed models, the impact of this exposure on the models' performance remains unknown.

Ethics Statement

Deploying the model in this paper would likely result in unknown false predictions. It requires further research to actually put it into practice.

References

Monica Agrawal, Stefan Hegselmann, Hunter Lang, Yoon Kim, and David Sontag. 2022. Large language models are few-shot clinical information extractors. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1998–2022, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. Scibert: A pretrained language model for scientific text. In *EMNLP*. 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, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Databricks. 2023. Databricks' dolly, a large language model trained on the databricks machine learning platform. `https://github.com/databrickslabs/dolly`.

Franck Dernoncourt and Ji Young Lee. 2017. Pubmed 200k rct: a dataset for sequential sentence classification in medical abstracts. *arXiv preprint arXiv:1710.06071*.

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

Lianyuan Feng, Shiyong Yao, Hejiang Sun, Nan Jiang, and Junjie Liu. 2015. Tr-piv measurement of exhaled flow using a breathing thermal manikin. *Building and environment*, 94:683–693.

Giacomo Frisoni, Miki Mizutani, Gianluca Moro, and Lorenzo Valgimigli. 2022. BioReader: a retrieval-enhanced text-to-text transformer for biomedical literature. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5770–5793, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

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. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Tom Hope, Jason Portenoy, Kishore Vasan, Jonathan Borchardt, Eric Horvitz, Daniel S Weld, Marti A Hearst, and Jevin West. 2020. Scisight: Combining faceted navigation and research group detection for covid-19 exploratory scientific search. *arXiv preprint arXiv:2005.12668*.

Ting-Hao'Kenneth' Huang, Chieh-Yang Huang, Chien-Kuang Cornelia Ding, Yen-Chia Hsu, and C Lee Giles. 2020. Coda-19: Using a non-expert crowd to annotate research aspects on 10,000+ abstracts in the covid-19 open research dataset. *arXiv preprint arXiv:2005.02367*.

Bernal Jimenez Gutierrez, Nikolas McNeal, Clayton Washington, You Chen, Lang Li, Huan Sun, and Yu Su. 2022. Thinking about GPT-3 in-context learning for biomedical IE? think again. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4497–4512, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Di Jin and Peter Szolovits. 2018. PICO element detection in medical text via long short-term memory neural networks. In *Proceedings of the BioNLP 2018 workshop*, pages 67–75, Melbourne, Australia. Association for Computational Linguistics.

Dan Lahav, Jon Saad Falcon, Bailey Kuehl, Sophie Johnson, Sravanthi Parasa, Noam Shomron, Duen Horng Chau, Diyi Yang, Eric Horvitz, Daniel S Weld, et al. 2022. A search engine for discovery of scientific challenges and directions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11982–11990.

Rui Meng, Khushboo Thaker, Lei Zhang, Yue Dong, Xingdi Yuan, Tong Wang, and Daqing He. 2021. Bringing structure into summaries: a faceted summarization dataset for long scientific documents. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language*

Processing (Volume 2: Short Papers), pages 1080–1089, Online. Association for Computational Linguistics.

Milad Moradi, Kathrin Blagec, Florian Haberl, and Matthias Samwald. 2021. Gpt-3 models are poor few-shot learners in the biomedical domain. *arXiv preprint arXiv:2109.02555*.

OpenAI. 2022. ChatGPT: A large language model. https://openai.com/blog/chatgpt/. Knowledge cutoff: September 2021.

OpenAI. 2023. Gpt-4 technical report.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Ronak Pradeep, Xueguang Ma, Rodrigo Nogueira, and Jimmy Lin. 2021. Scientific claim verification with VerT5erini. In *Proceedings of the 12th International Workshop on Health Text Mining and Information Analysis*, pages 94–103, online. Association for Computational Linguistics.

MosaicML NLP Team. 2023. Introducing mpt-7b: A new standard for open-sourcely usable llms.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Jingxuan Tu, Marc Verhagen, Brent Cochran, and James Pustejovsky. 2020. Exploration and discovery of the covid-19 literature through semantic visualization. *arXiv preprint arXiv:2007.01800*.

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. 2020. Fact or fiction: Verifying scientific claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7534–7550, Online. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Edwin Zhang, Nikhil Gupta, Rodrigo Nogueira, Kyunghyun Cho, and Jimmy Lin. 2020. Rapidly deploying a neural search engine for the COVID-19 Open Research Dataset. In *Proceedings of the 1st Workshop on NLP for COVID-19 at ACL 2020*, Online. Association for Computational Linguistics.

Label	Id	Sentence
Background	0	**[POSITION=0.00]** Breathing is a high-risk behavior for spreading infectious diseases in enclosed environments ,
Background	1	**[POSITION=0.07]** so it is important to investigate the characteristics of human exhalation flow and dispersal of exhaled air to reduce the risk .
Background	2	**[POSITION=0.14]** This paper used two-dimensional time-resolved particle image velocimetry (2D TR-PIV) to measure the exhaled flow from a breathing the rmal manikin .
Method	3	**[POSITION=0.21]** Since the exhaled flow is transient and periodic ,
Method	4	**[POSITION=0.29]** the phase-averaged method was used to analyze the flow characteristics .,

Table 4: A sample of position-encoded CODA-19 data extracted from the paper ((Feng et al., 2015)). Here, **Id** indicates the sentence index with respect to the abstract. Removing the position encoding (*e.g.*, **[POSITION=0.00]**), we could get the original CODA-19 data.

Label	Id	Sentence
Methods	3	**[POSITION=0.38]** The serum levels of follicle stimulating hormone (FSH), luteinizing hormone (LH), and estradiol (E(2)) were detected before and after the treatment.
Results	4	**[POSITION=0.50]** After 12 weeks of treatment, HAMD scores in both groups decreased significantly ($p<0.05$) with no significant difference between the groups ($p>0.05$).
Results	5	**[POSITION=0.62]** The levels of FSH decreased significantly and the level of E(2) increased significantly in both groups, and they changed more in the control group.
Results	6	**[POSITION=0.75]** No side-effect of treatment was reported in either group during treatment.
Conclusions	7	**[POSITION=0.88]** The Chinese medicinal formula GNL showed promise in relieving perimenopausal depression and merits further study.

Table 5: A sample of position-encoded PubMed data extracted from Paper ID: 19769482. Here, **Id** indicates the sentence index with respect to the abstract. Removing the position encoding (*e.g.*, **[POSITION=0.38]**), we could get the original PubMed data.

A Sample Data

In this section, we show some sample data for CODA-19 dataset (Table 4) and the PubMed dataset (see Table 5). As shown in the table, [POSITION=0.38] is the position encoding we added to inject the positional information.

B Training and Testing Details

Here, we describe all the training details for the models we build in this study. All the models are trained using PyTorch (Paszke et al., 2019) and HuggingFace (Wolf et al., 2019).

- **SciBERT$_{\text{CODA}}$.** We fine-tune SciBERT using the original CODA-19 training set using the hyperparameters listed in Table 6.
- **SciBERT$_{\text{CODA+Pos}}$.** The position encoding is first added to create the position-encoded CODA-19 dataset. We then fine-tune SciBERT on the created dataset using the hyperparameters listed in Table 6.
- **SciBERT$_{\text{PubMed}}$.** Similar to the above but on the original PubMed dataset.
- **SciBERT$_{\text{PubMed+Pos}}$.** Similar to the above but on the position-encoded PubMed dataset.
- **SciBERT$_{\text{Mix+Pos+S}}$.** We first turn the position-encoded PubMed's label into the CODA-19 label space using the pre-defined mapping: *Background* to *Background*, *Methods* to *Method*, *Objective* to *Purpose*, *Results* to *Finding*, and *Conclusions* to *Finding*. Second, we simply mix the position-encoded CODA-19 training set with the position-encoded PubMed training set together to create a simply-mixing dataset. We then fine-tune SciBERT on the created dataset using the hyperparameters listed in Table 6. Note that the CODA-19 validation set is used to checkpoint the best model.
- **SciBERT$_{\text{Mix+Pos+U}}$.** As the data sizes of CODA-19 and PubMed differ a lot (137K vs. 2.2M in the training set), we upsample position-encoded CODA-19's training set 10 times to create a more balanced upsampling dataset. Before mixing, PubMed's label has been transferred to the CODA-19 label space using the pre-defined mapping function. We then fine-tune SciBERT using this dataset with the pre-defined hyperparameters (Table 6). Again, the CODA-19 validation set is used to checkpoint the best model.
- **SciBERT$_{\text{Mix+Pos+T}}$.** For the two-staged training strategy using PubMed and CODA-19 dataset,

Hyperparameter	Value
Model	scibert_scivocab_uncased
Batch Size	64
Learning Rate	1e-5
Epochs	20
Metric for Best Model	Evaluation Accuracy
Max Sequence Length	128
Warmup Ratio	0.1
Early Stopping Patience	6

Table 6: General hyperparameters used for training the models. We used HuggingFace's Trainer for fine-tuning all the models. Parameters not specified here remain the default values.

we first fine-tune SciBERT on the position-encoded PubMed dataset with the pre-defined hyperparameters (Table 6). Here, the PubMed validation set is used to checkpoint the best model. In the second stage, we fine-tune the checkpointed model on the position-encoded CODA-19 dataset with the pre-defined hyperparameters (Table 6).

- **LLaMA.** We obtain LLaMA-65B from the official Github[1] and run the generation using HuggingFace's interface (Wolf et al., 2019). Temperature sampling is used for text generation with temperature = 0.1, num_beams = 1, top_p = 0.95, repetition_penalty = 1.0, min_new_tokens = 1, and max_new_tokens = 10.
- **MPT.** We use `mosaicml/mpt-7b-instruct` and run the generation using HuggingFace's interface (Wolf et al., 2019) with the same parameters described in LLaMA.
- **Dolly.** We use `databricks/dolly-v2-12b` and run the generation using HuggingFace's interface (Wolf et al., 2019) with the same parameters described in LLaMA.
- **GPT-3.** We use `text-davinci-003` with the parameters: temperature = 0.0, max_tokens = 10, top_p = 0.95, frequency_penalty = 0.0, and presence_penalty = 0.0.
- **ChatGPT.** We use `gpt-3.5-turbo` with the same parameters described in GPT-3. Note that when calling ChatGPT, we simply put all the prompts in a single user input.
- **GPT-4.** We use `gpt-4` with the same parameters described in GPT-3. Again, when calling GPT-4, we simply put all the prompts in a single user input.

[1] https://github.com/facebookresearch/llama

C Prompts

Table 7 and Table 8 show the zero-shot prompt and the few-shot prompt we used for querying LLMs.

Zero-shot Prompt
Classify the given text into one of the following labels.

[Background]: Text segments answer one or more of these questions: Why is this problem important?, What relevant works have been created before?, What is still missing in the previous works?, What are the high-level research questions?, How might this help other research or researchers?
[Purpose]: Text segments answer one or more of these questions: What specific things do the researchers want to do?, What specific knowledge do the researchers want to gain?, What specific hypothesis do the researchers want to test?
[Method]: Text segments answer one or more of these questions: How did the researchers do the work or find what they sought?, What are the procedures and steps of the research?
[Finding]: Text segments answer one or more of these questions: What did the researchers find out?, Did the proposed methods work?, Did the thing behave as the researchers expected?
[Other]: Text fragments that do NOT fit into any of the four categories above. Text fragments that are NOT part of the article. Text fragments that are NOT in English. Text fragments that contains ONLY reference marks (e.g., "[1,2,3,4,5") or ONLY dates (e.g., "April 20, 2008"). Captions for figures and tables (e.g. "Figure 1: Experimental Result of ...", or "Table 1: The Typical Symptoms of ...") Formatting errors. I really don't know or I'm not sure.

FAQs
1. This text fragment has terms that I don't understand. What should I do? Please use the context in the article to figure out the focus. You can look up terms you don't know if you feel like you need to understand them.
2. This text fragment is too short to mean anything. What should I do? If the text fragment is too short to have significant meanings, you could consider the entire sentence and answer based on the entire sentence.
3. This text fragment is NOT in English. What should I do? If the whole fragment (or the majority of words in the fragment) is in Non-English, please label it as "Other". If the majority of the words in this fragment are in English with a few non-English words, please judge the label normally.
4. I'm not sure if this should be a "background" or a "finding." How do I tell? When a sentence occurs in the earlier part of an article, and it is presented as a known fact or looks authoritative, it is often a "background" information.
5. Do "potential applications of the proposed work" count as "background" or "purpose"? It should be "background." The "purpose" refers to specific things the paper wants to achieve.
6. If the article says it's a "literature review" (e.g., "We reviewed the literature" / "In this article, we review.." etc), would we classify those as finding/contribution or purpose? Most parts of a literature review paper should still be "background" or "purpose", and only the "insight" drew from a set of prior works can be viewed as a "finding/contribution".
7. What should I do with the case study on a patient? Typically, it has a patient come in with a set of signs and symptoms in the ER, and then the patient gets assessed and diagnosed. The patient is admitted to the hospital ICU and tests are done and they may be diagnosed with something else. In such cases, please label the interventions done by the medical staff (e.g., CT scans, X-rays, and medications given) as "Method", and the patient's final result (e.g. the patient's pneumonia resolved and he was released from the hospital) as "Finding/Contribution".

Classify the following sentence into one of the label: Background, Purpose, Method, Finding, and Other.

Text: **"'{Target-Sentence}"'**
The answer label for Text is [

Table 7: Zero-shot prompt used when calling LLMs (LLaMA, MPT, Dolly, GPT-3, ChatGPT, and GPT-4). The **{Target-Sentence}** will be replaced by the sentence we would like to predict.

Few-shot Prompt
Classify the given text into one of the following labels.

[Background]: Text segments answer one or more of these questions: Why is this problem important?, What relevant works have
. . .
· · · (Same as the zero-shot prompt that describe the definition of the label and FAQs. Skip for space.)
. . .
Classify the following sentence into one of the label: Background, Purpose, Method, Finding, and Other.

Text: "'With the features of extremely high selectivity and efficiency in catalyzing almost all the chemical reactions in cells ,"'
Label: [Background]
Text: "'enzymes play vitally important roles for the life of an organism and hence have become frequent targets for drug design .'"
Label: [Background]
Text: "'by which users can easily obtain their desired results ."'
Label: [Method]
Text: "'An essential step in developing drugs by targeting enzymes is to identify drug-enzyme interactions in cells ."'
Label: [Background]
Text: "'a user-friendly web server was established ,"'
Label: [Method]
Text: "'called " iEzy-Drug , " in which each drug compound was formulated by a molecular fingerprint with 258 feature components ,"'
Label: [Method]
Text: "'and the prediction engine was operated by the fuzzy K-nearest neighbor algorithm ."'
Label: [Method]
Text: "'Although some computational methods were developed in this regard based on the knowledge of the three-dimensional structure of enzyme ,"'
Label: [Background]
Text: "'Here , we reported a sequence-based predictor ,"'
Label: [Purpose]
Text: "'Moreover , to maximize the convenience for the majority of experimental scientists ,"'
Label: [Method]
Text: "'It is both time-consuming and costly to do this purely by means of experimental techniques alone ."'
Label: [Background]
Text: "'The overall success rate achieved by iEzy-Drug via rigorous cross-validations was about 91 % ."'
Label: [Finding]
Text: "'unfortunately their usage is quite limited because threedimensional structures of many enzymes are still unknown ."'
Label: [Background]
Text: "'each enzyme by the Chou 's pseudo amino acid composition generated via incorporating sequential evolution information and physicochemical features derived from its sequence ,"'
Label: [Method]
Text: **"'{Target-Sentence}"'**
The answer label for Text is [

Table 8: Few-shot prompt used when calling LLMs (LLaMA, MPT, Dolly, GPT-3, ChatGPT, and GPT-4). The **{Target-Sentence}** will be replaced by the sentence we would like to predict. The skipped label description and FAQs are the same as the zero-shot prompt (see Table 7). The few-shot samples are from one single abstract to simulate the scenario where people annotate some data as a reference.