# Knowledge Distillation in Automated Annotation: Supervised Text Classification with LLM-Generated Training Labels

**Nicholas Pangakis** and **Samuel Wolken**
University of Pennsylvania
{njpang@sas., sam.wolken@asc.}upenn.edu

## Abstract

Computational social science (CSS) practitioners often rely on human-labeled data to fine-tune supervised text classifiers. We assess the potential for researchers to augment or replace human-generated training data with surrogate training labels from generative large language models (LLMs). We introduce a recommended workflow and test this LLM application by replicating 14 classification tasks and measuring performance. We employ a novel corpus of English-language text classification data sets from recent CSS articles in high-impact journals. Because these data sets are stored in password-protected archives, our analyses are less prone to issues of contamination. For each task, we compare supervised classifiers fine-tuned using GPT-4 labels against classifiers fine-tuned with human annotations and against labels from GPT-4 and Mistral-7B with few-shot in-context learning. Our findings indicate that supervised classification models fine-tuned on LLM-generated labels perform comparably to models fine-tuned with labels from human annotators. Fine-tuning models using LLM-generated labels can be a fast, efficient and cost-effective method of building supervised text classifiers.

## 1 Introduction

Supervised text classification often relies on human-labeled text data for training and validation. Computational social science (CSS) researchers frequently use these types of supervised models to classify large quantities of text, ranging from news articles on the internet to government documents (Grimmer et al., 2022; Lazer et al., 2020). Collecting training and validation labels generated by humans for these tasks, however, is expensive, slow, and prone to a variety of errors (Grimmer and Stewart, 2013; Neuendorf, 2016).

To address these limitations, prior research suggests utilizing few-shot capabilities of generative large language models (LLMs) to annotate text data instead of human annotators (Gilardi et al., 2023). Generative LLMs are faster and cheaper than human annotators and do not suffer from common human challenges such as limited attention span or fatigue. While this approach has its limitations and generative LLMs do not excel at all text annotation tasks (Pangakis et al., 2023), prior research illustrates that there are numerous circumstances where generative LLMs can produce high quality text-annotation labels.[1]

Although past work suggests LLM few-shot annotation is highly effective, it may be cost prohibitive in many settings. Research with text data often involves classifying millions of documents or text samples. For example, a recent CSS article studies a data set of 6.2 million tweets labeled on four dimensions (Hopkins et al., 2024), a task that would have cost nearly $9,000 if using GPT-4 alone.[2] Using a knowledge distillation approach (Dasgupta et al., 2023; Gou et al., 2021; Hinton et al., 2015), it may be possible to approximate the performance of a larger "teacher" model (e.g., GPT-4 (OpenAI, 2023), estimated to have over 1.7T parameters (Schreiner, 2023)) with much smaller and cheaper task-specific "student" models (e.g., BERT Base (Devlin et al., 2019), approximately 110 million parameters).

In this paper, we evaluate using generative LLMs to create surrogate labels for fine-tuning downstream supervised classification models. Our approach involves first using a generative LLM to label a subset of text samples and then fine-tuning supervised text classifiers with the LLM-generated labels. Using our outlined approach, we replicate 14 classification tasks from recently published CSS articles. We compare several supervised classifiers (i.e., BERT (Devlin et al., 2019), RoBERTa (Liu

[1] See Appendix A.1 for a longer discussion of automated annotation research in CSS.

[2] Appendix A.2 elaborates on costs with LLM annotation.



*Proceedings of the Sixth Workshop on Natural Language Processing and Computational Social Science (NLP+CSS)*, pages 113–131

et al., 2019), DistilBERT (Sanh et al., 2019), XLNet (Yang et al., 2020), and Mistral-7B (Jiang et al., 2023)) fine-tuned on varying quantities of either human-labeled samples or GPT-4-labeled samples. We benchmark the supervised classifiers' performance against GPT-4 and Mistral-7B few-shot labels. In a series of ablation experiments, we also explore whether GPT-4 outputs change over time and how well the student models handle noise in the GPT-generated text labels.

A small number of studies have utilized similar approaches in related domains. Chen et al. (2023b) use ChatGPT annotations to train various Graph Neural Networks for a fraction of the cost of human annotations. Golde et al. (2023) also harness ChatGPT to create surrogate text data that aligns with a specific valence (i.e., positive and negative) and then subsequently fine-tune a supervised classifier using the synthetic text. Most analogous to our approach here, Wang et al. (2021) train RoBERTa (Liu et al., 2019) and PEGASUS (Zhang et al., 2020) models on labels generated by GPT-3. Despite strong performance across their analyses, Wang et al. (2021), as well as the previously mentioned studies, exclusively evaluate closed-source models (i.e., GPT-3 and ChatGPT) on popular, publicly available NLP benchmark tasks (e.g., AGNews, DBPedia, etc), which are plausibly included in the training data for the generative LLM. As a result, these analyses cannot offer a clear indication of performance because their results plausibly suffer from contamination (Balepur et al., 2024; Li and Flanigan, 2023; Magar and Schwartz, 2022; Srivastava et al., 2024). Put otherwise, strong performance may reflect memorization, which casts doubt on the generalizability of the findings.

To compare supervised classifiers fine-tuned using LLM-generated labels against those fine-tuned with labels from human annotators, researchers must assess performance on tasks less likely to be affected by contamination. To this end, all 14 of the classification tasks we replicate are conducted on labeled data sets stored in password-protected archives. Each of the classification tasks in our corpus are real CSS applications and contain human-labeled ground-truth annotations.[3]

Our main contributions are as follows:

1. Across 14 classifications tasks, supervised models fine-tuned with GPT-generated labels perform comparably to models fine-tuned with human-labeled data. The median F1 performance gap between models fine-tuned using GPT-labels and models fine-tuned on human-labeled data is only 0.039. While supervised classifiers fine-tuned with LLM-generated labels perform slightly worse than classifiers fine-tuned with human labels, LLM-generated labels can be a fast, efficient and cost-effective method to fine-tune supervised text classifiers.
2. Supervised models fine-tuned on GPT-4 generated labels perform remarkably close to GPT few-shot models, with a median F1 difference of only 0.006 across the classification tasks.
3. GPT-4 few-shot models and supervised classifiers fine-tuned on GPT-4 generated labels perform significantly better than all other models on *recall*, but noticeably worse on *precision*.

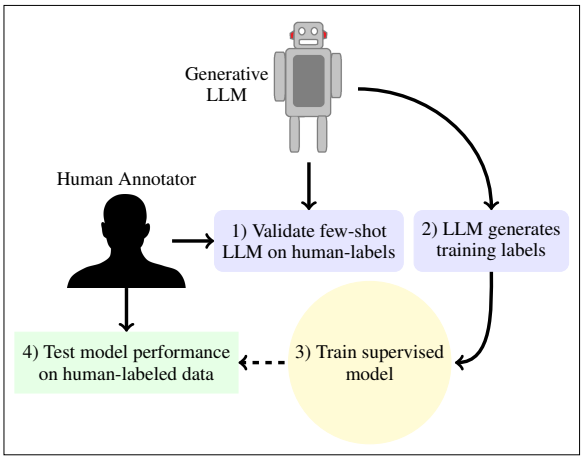


Figure 1: Supervised text classification with LLM-generated training labels.

## 2 Methodology

Figure 1 shows our four-step workflow. First, we validate LLM few-shot performance against a small subset (n=250) of human-labeled text samples for each task. We provide GPT-4[4] with detailed instructions to label the text samples into conceptual categories outlined in the original study.[5] Because LLM few-shot annotation performance varies across tasks and data sets, validation is always necessary (Pangakis et al., 2023). As such, we validate

[3]Table A2 and Table A3 include a full list of the data sets and classification tasks.

[4]We select GPT-4 as our main generative model due to its high performance on popular leaderboard websites. In Appendix E.1, we also explore few-shot performance of an open-source model (i.e., Mistral-7B).

[5]We include all prompt details in the supplementary material. We also include our code to query the GPT-4 API.

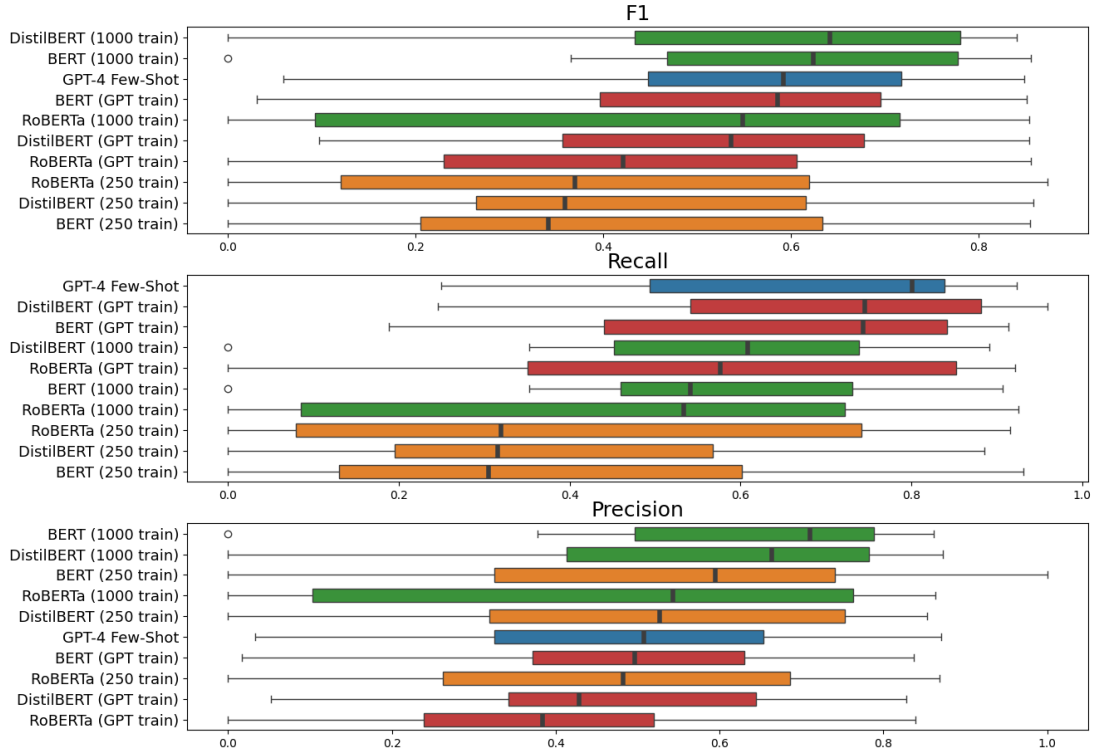


Figure 2: Box plots of performance on test data across 14 tasks. Thick vertical line denotes median. Color represents model type, with green corresponding to models fine-tuned on 1,000 human labels, orange to 250 human labels, red to 1,000 GPT labels, and blue to a few-shot model.

each generative LLM on a subsample and then adjust the prompt to optimize performance on this initial sample. This process is discussed in greater detail in Appendix C.1. Using the validated prompt, the second step in our workflow involves labeling an additional 1,000 text samples per task using the same generative LLM, which will later be used as data to fine-tune the supervised classifier.

In the third and fourth steps, we fine-tune a variety of supervised text classifiers and assess performance against a held-out set of 1000 human-labeled samples. Our supervised models include a variety of BERT-family models (i.e., BERT, RoBERTa, and DistilBERT).[6] In Appendix E.1, we conduct ablation experiments with XLNet and Mistral-7B. Appendix C.2 describes on our hyperparameter tuning process and additional evaluation details, including how multi-class tasks were split into separate binary tasks. Ultimately, we compare performance between text classifiers fine-tuned on 1000 LLM-generated samples, 250 human-labeled samples, and 1000 human-labeled samples.

In addition to analyzing performance across different model architectures and training sample sizes, we also implement a variety of ablation experiments to assess how robust the analyses are to several sources of variance. First, we examine how robust these models are to noisy GPT-generated labels. Specifically, in Appendix E, we implement a novel technique designed to measure noise in GPT-generated labels and then compare supervised models fine-tuned on GPT-generated labels *with noise* against models fine-tuned on GPT-generated labels *without noise*. In a second set of ablation experiments, we replicate the GPT-4 few-shot labels at different points in time. To account for the potential of changing model weights in GPT-4, we re-analyzed each task six months after our initial analyses and compared results across time. Extended discussion and the results for these ablation experiments are shown in Appendix E.

## 3 Results

Classification results for the BERT-family models and GPT-4 few-shot are shown in Table 1.[7] In Figure 2, each box plot displays the range of

[6]We select these models because of their low cost, speed, and their frequent application in CSS (Büyüköz et al., 2020; Terechshenko et al., 2020).

[7]We conduct few-shot classification by using the classification instructions from the original study as a prompt for the LLM.

| Model | Training data | Accuracy | F1 | Precision | Recall |
|---|---|---|---|---|---|
| GPT-4 | Few shot | 0.88 | 0.59 | 0.51 | 0.80 |
| BERT | Human annotation: 250 | 0.89 | 0.34 | 0.59 | 0.30 |
| | Human annotation: 1000 | 0.92 | 0.62 | 0.71 | 0.54 |
| | GPT-4 annotation: 1000 | 0.87 | 0.59 | 0.50 | 0.74 |
| DistilBERT | Human annotation: 250 | 0.89 | 0.36 | 0.53 | 0.32 |
| | Human annotation: 1000 | 0.89 | 0.64 | 0.66 | 0.61 |
| | GPT-4 annotation: 1000 | 0.85 | 0.54 | 0.43 | 0.75 |
| RoBERTa | Human annotation: 250 | 0.88 | 0.37 | 0.48 | 0.32 |
| | Human annotation: 1000 | 0.90 | 0.55 | 0.54 | 0.53 |
| | GPT-4 annotation: 1000 | 0.84 | 0.42 | 0.38 | 0.58 |

Table 1: Comparison of classification performance on held-out validation data. Median performance across 14 tasks shown.

evaluation metrics across all 14 tasks for a given model/training data combination. The thick vertical line denotes the median performance metric across all analyzed tasks. Across all 14 classification tasks, DistilBERT and BERT fine-tuned on 1000 human-samples are the highest performing models, with a median F1 score of 0.641 and 0.624, respectively.[8] Not far behind, however, is the GPT-4 few-shot model (0.592 median F1) and BERT fine-tuned on 1000 GPT-labeled samples (0.586 median F1). From this we draw two conclusions: First, models fine-tuned on few-shot surrogate labels from a generative LLM perform comparably to models fine-tuned on human labels. Despite a small performance gap, training supervised models on LLM-labeled data can be a quick, effective, and budget-friendly approach for constructing supervised text classifiers.

Second, models trained on surrogate labels from GPT-4 demonstrate very similar validation performance as labels from GPT-4 with few-shot in-context learning. As each additional GPT-4 query incurs more expense, researchers can save resources by avoiding classifying an entire data set using a generative LLM and instead use them to create training labels for a supervised model.

A secondary finding is that GPT few-shot models and supervised models trained on GPT-generated labels produce remarkably high performance on recall.[9] GPT-4 few-shot (0.8 median recall) as well as DistilBERT and BERT fine-tuned on GPT-labels (both with 0.746 median recall) achieve significantly better median recall than any model fine-tuned with human labels. The opposite is true for precision: BERT fine-tuned on human-labels achieved the highest precision of the models tested, which was 0.214 higher than median precision for BERT models fine-tuned on GPT-4 labels. Therefore, using surrogate training labels may be better suited for tasks where recall is prioritized over precision.

## 4 Discussion

Surrogate labels from generative LLMs offer a viable, low-resource strategy for fine-tuning task-specific supervised classifiers, but a few points of caution are worth emphasizing. As the variation in our few-shot results indicates, there are cases where GPT-4 performs poorly on classification tasks. While advancements in LLM technology and additional prompt engineering could mitigate these concerns, it is essential that researchers validate generative LLM performance against ground-truth human-labeled data. Downstream supervised classifiers will not mitigate bias or poor performance in LLM few-shot labels. Thus, while generative LLMs can improve the classification workflow, their application must remain human-centered.

[8] We use F1 as our primary evaluation criteria due to class imbalance. Full results are shown in Table A4.

[9] Appendix D displays PR curves for each of the BERT-family supervised models.

## 5 Limitations

Here, we identify three main limitations of our analysis. First, as discussed in Section 4 and shown in full detail in Table A4, there are various circumstances where supervised models fine-tuned on LLM-generated labels fail to produce satisfactory results. This may be due to inaccurate annotations from GPT-4, poor performance from the supervised classifier, or both. While it is possible that additional prompt engineering or hyperparameter tuning could improve performance, it is essential to stress that each of these optimization strategies rely on human labels for comparison. As a result, we argue that it is essential to center human judgement as ground truth when optimizing models and adjudicating between models.

A second, related limitation refers to understanding the errors in the model outputs. Specifically, it is possible that errors from a GPT-trained model produces correlated but unobservable errors. Building a supervised classifier on top of GPT-4 labels would magnify, rather than offset, any such biases. This, too, underscores the importance of human validation and error analysis. It is, of course, also essential to minimize bias by human annotators. For instance, recruiting human annotators from varying demographic backgrounds when conducting an annotation project may diminish the potential for correlated errors across annotators.

Finally, treating human labels as ground truth is an additional limitation. Although most data sets in our analysis employed multiple human coders, it is of course possible that these annotators made correlated errors. As a result, some disagreements between human ground truth labels and surrogate GPT-4 labels may stem from human error. Such errors could bias performance metrics downward for any of the models assessed. Because our primary interest is making comparisons across models, however, we are mainly interested in their relative performance. Because each model would suffer from the same errors in the human labeled data, we do not see this as a significant concern for this analysis.

For the analysis in this paper, our reliance on text classification tasks and data from peer-reviewed research in high-impact journals helps to mitigate concerns about data annotation quality. The annotation procedures in each of these tasks received IRB approval and was assessed by independent reviewers to be of quality enough for publication in a high-impact journal. Still, it is important to acknowledge that applied researchers should invest in high-quality human labels, even if only to validate generative LLM annotation performance.

## 6 Ethics Statement

Our research complies with the ACL Ethics Policy. Specifically, our research positively contributes to society and human well-being by providing tools that can aid computational social scientists studying the social world. Using the methods we introduce and test will help scientists better understand a wide range of complicated social problems. Because the techniques proposed and assessed in this article require dramatically less resource expenditure than alternatives, our results can help address inequities in resources across researchers.

Due to the inherent risks of deploying biased models, we stress the necessity of human validation throughout our paper. Given the ease and efficiency gains of using generative LLMs to train supervised classifiers, we believe it is essential to build rigorous testing and evaluation standards that are human-centered. This is why we took great efforts to center our analyses on data sets less prone to contamination risks.

Moreover, our research and data analysis does not cause any harm while also respecting privacy and confidentiality concerns. As we discuss in our data collection procedures in Appendix B, we conformed to each data repository's usage and replication policies. Each of the original studies received IRB approval and our analyses conformed to the same safety protocols. All collected data was anonymized by the original authors. Appendix C.3 provides additional details on human annotation protocols, which were all conducted by the original studies and received IRB approval.

## References

Nishant Balepur, Abhilasha Ravichander, and Rachel Rudinger. 2024. Artifacts or abduction: How do llms answer multiple-choice questions without the question?

Berfu Büyüköz, Ali Hürriyetoğlu, and Arzucan Özgür. 2020. Analyzing elmo and distilbert on socio-political news classification. In *Proceedings of the Workshop on Automated Extraction of Socio-political Events from News 2020*, pages 9–18.

Dallas Card, Serina Chang, Chris Becker, Julia Mendelsohn, Rob Voigt, Leah Boustan, Ran Abramitzky, and

Dan Jurafsky. 2022. Computational analysis of 140 years of us political speeches reveals more positive but increasingly polarized framing of immigration. *Proceedings of the National Academy of Sciences of the United States of America*, 31.

Lingjiao Chen, Matei Zaharia, and James Zou. 2023a. How is chatgpt's behavior changing over time?

Zhikai Chen, Haitao Mao, Hongzhi Wen, Haoyu Han, Wei Jin, Haiyang Zhang, and Hui Liuand Jiliang Tang. 2023b. Label-free node classification on graphs with large language models (llms).

Cheng-Han Chiang and Hung-yi Lee. 2023. Can large language models be an alternative to human evaluations? *arXiv preprint arXiv:2305.01937*.

Michael Chmielewski and Sarah C. Kucker. 2020. An mturk crisis? shifts in data quality and the impact on study results. *Social Psychological and Personality Science*, 11(4):464–473.

Sayantan Dasgupta, Trevor Cohn, and Timothy Baldwin. 2023. Cost-effective distillation of large language models. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 7346–7354, Toronto, Canada. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Le, 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)*, page 4171–4186.

Bosheng Ding, Chengwei Qin, Linlin Liu, Lidong Bing, Shafiq Joty, and Boyang Li. 2022. Is gpt-3 a good data annotator?

Benjamin D. Douglas, Patrick J. Ewell, and Markus Braue. 2023. Data quality in online human-subjects research: Comparisons between mturk, prolific, cloudresearch, qualtrics, and sona. *PLoS One*, 18.

Naoki Egami, Christian J. Fong, Justin Grimmer, Margaret E. Roberts, and Brandon M. Stewart. 2022. How to make causal inferences using texts. *Science Advances*, 8(42):eabg2652.

Fabrizio Gilardi, Meysam Alizadeh, and Maël Kubli. 2023. Chatgpt outperforms crowd-workers for text-annotation tasks.

Jonas Golde, Patrick Haller, Felix Hamborg, Julian Risch, and Alan Akbik. 2023. Fabricator: An open source toolkit for generating labeled training data with teacher llms.

Jianping Gou, Baosheng Yu, Stephen J. Maybank, and Dacheng Tao. 2021. Knowledge distillation: A survey. *International Journal of Computer Vision*, page 1789–1819.

Justin Grimmer, Margaret E. Roberts, and Brandon Stewart. 2022. *Text as Data: A New Framework for Machine Learning and the Social Sciences*. Princeton University Press.

Justin Grimmer and Brandon M. Stewart. 2013. Text as data: The promise and pitfalls of automatic content analysis methods for political texts. *Political Analysis*, 21(3):267–297.

Xingwei He, Zhenghao Lin, Yeyun Gong, A-Long Jin, Hang Zhang, Chen Lin, Jian Jiao, Siu Ming Yiu, Nan Duan, and Weizhu Chen. 2023. Annollm: Making large language models to be better crowdsourced annotators.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network.

Daniel J. Hopkins, Yphtach Lelkes, and Samuel Wolken. 2024. The rise of and demand for identity-oriented media coverage. *American Journal of Political Science*.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b.

Ross Deans Kristensen-McLachlan, Miceal Canavan, Márton Kardos, Mia Jacobsen, and Lene Aarøe. 2023. Chatbots are not reliable text annotators.

David MJ Lazer, Alex Pentland, Duncan J Watts, Sinan Aral, Susan Athey, Noshir Contractor, Deen Freelon, Sandra Gonzalez-Bailon, Gary King, and Helen Margetts. 2020. Computational social science: Obstacles and opportunities. *Science*, 369(6507):1060–1062.

Changmao Li and Jeffrey Flanigan. 2023. Task contamination: Language models may not be few-shot anymore.

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.

Inbal Magar and Roy Schwartz. 2022. Data contamination: From memorization to exploitation.

Wes McKinney. 2011. pandas: a foundational python library for data analysis and statistics. *Python for high performance and scientific computing*, 14(9):1–9.

Jonathan Mellon, Jack Bailey, Ralph Scott, James Breckwoldt, and Marta Miori. 2022. Does gpt-3 know what the most important issue is? using large language models to code open-text social survey responses at scale. Working paper.

Stefan Müller. 2022. The temporal focus of campaign communication. *The Journal of Politics*, 84(1):585–590.

Kimberly A. Neuendorf. 2016. *The Content Analysis Guidebook*. Sage Publications.

OpenAI. 2023. Gpt-4 technical report.

Alexander Pan, Jun Shern Chan, Andy Zou, Nathaniel Li, Steven Basart, Thomas Woodside, Hanlin Zhang, Scott Emmons, and Dan Hendrycks. 2023. Do the rewards justify the means? measuring trade-offs between rewards and ethical behavior in the machiavelli benchmark. *Proceedings of the 40th International Conference on Machine Learning*, pages 26837–26867.

Nicholas Pangakis, Samuel Wolken, and Neil Fasching. 2023. Automated annotation with generative ai requires validation.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, and Trevor Killeen et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*.

Hao Peng, Daniel M. Romero, and Emoke-Agnes Horvat. 2022. Dynamics of cross-platform attention to retracted papers. *Proceedings of the National Academy of Sciences*, 119(25):585–590.

Michael Reiss. 2023. Testing the reliability of chatgpt for text annotation and classification: A cautionary remark. Working paper.

Christopher Michael Rytting, Taylor Sorensen, Lisa Argyle, Ethan Busby, Nancy Fulda, Joshua Gubler, and David Wingate. 2023. Towards coding social science datasets with language models.

Punyajoy Saha, Narla, Komal Kalyan, and Animesh Mukherjee. 2023. On the rise of fear speech in online social media. *Proceedings of the National Academy of Sciences of the United States of America*.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter.

Maximilian Schreiner. 2023. Gpt-4 architecture, datasets, costs and more leaked. Blog post.

Saurabh Srivastava, Annarose M B, Anto P V au2, Shashank Menon, Ajay Sukumar, Adwaith Samod T, Alan Philipose, Stevin Prince, and Sooraj Thomas. 2024. Functional benchmarks for robust evaluation of reasoning performance, and the reasoning gap.

Zhanna Terechshenko, Fridolin Linder, Vishakh Padmakumar, Michael Liu, Jonathan Nagler, Joshua A Tucker, and Richard Bonneau. 2020. A comparison of methods in political science text classification: Transfer learning language models for politics. *Working Paper*.

Surendrabikram Thapa, Usman Naseem, and Mehwish Nasim. 2023. From humans to machines: can chatgpt-like llms effectively replace human annotators in nlp tasks. *Workshop Proceedings of the 17th International AAAI Conference on Web and Social Media*.

Petter Törnberg. 2023. Chatgpt-4 outperforms experts and crowd workers in annotating political twitter messages with zero-shot learning.

Veniamin Veselovsky, Manoel Horta Ribeiro, and Robert West. 2023. Artificial artificial artificial intelligence: Crowd workers widely use large language models for text production tasks.

Shuohang Wang, Yang Liu, Yichong Xu, Chenguang Zhu, and Michael Zeng. 2021. Want to reduce labeling cost? gpt-3 can help.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison and´ Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. *In Proceedings of EMNLP*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2020. Xlnet: Generalized autoregressive pretraining for language understanding.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. *International Conference on Machine Learning*, page 11328–11339.

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

Caleb Ziems, William Held, Omar Shaikh, Jiaao Chen, Zhehao Zhang, and Diyi Yang. 2023. Can large language models transform computational social science? Working paper.

# A Appendix: Prior automated annotation research in computational social science

## A.1 Overview of automated annotation research

A growing body of research studying automated annotation claims that few-shot classifications from generative LLMs can match humans on annotation tasks (Chiang and Lee, 2023; Ding et al., 2022; Gilardi et al., 2023; He et al., 2023; Mellon et al.,

| GPT-4: Entire Corpus (n=6.2m) | GPT-4: n=1000 | Crowdworker: n=1000 | Trained Assistant: n=1000 |
|---|---|---|---|
| $8,990 | $15 | $124 | $187 |

Table A1: Comparing annotation costs applied to Hopkins et al. (2024).

2022; Pan et al., 2023; Rytting et al., 2023; Thapa et al., 2023; Törnberg, 2023; Zhu et al., 2023; Ziems et al., 2023). For example, Gilardi et al. (2023) find that LLMs outperform typical crowd-sourced human annotators: "[t]he evidence is consistent across different types of texts and time periods. It strongly suggests that ChatGPT may already be a superior approach compared to crowd annotations on platforms such as MTurk." Analyzing a range of social science applications, Rytting et al. (2023) similarly write, "GPT-3 can match the performance of human coders [and in] some cases, it even outperforms humans in increasing intercoder agreement scores." Törnberg (2023) argues that automated annotations by LLMs in his analyses are even as accurate as annotations by human experts. While there are clearly circumstances where automated annotation fails to accurately reflect human judgment (Kristensen-McLachlan et al., 2023; Reiss, 2023), researchers can safely use automated annotation procedures as long as they validate against human labels not prone to contamination (Pangakis et al., 2023).

### A.2 Costs associated with implementing automated annotation

While prior research demonstrates that automated annotation can align with human reasoning in many scenarios, directly using the strategies introduced in prior studies to label an entire text corpus would be cost prohibitive when applied to a typical CSS data set, which often contain millions of observations. Consider the cost for using GPT-4 to label a data set of 6.2 million tweets, which is what Hopkins et al. (2024) analyze. At the time of writing, GPT-4 costs $0.01 per 1k input tokens and $0.03 per 1k output tokens, with 1000 tokens corresponding to roughly 750 words.[10] The prompt instructions to replicate Hopkins et al. (2024) contained approximately 500 words and the average tweet length was around 25 words. Because the full corpus contained 6.2 million tweets and the code to query the OpenAI API was implemented in batches of 10 tweets, a full automated annotation to process the corpus in Hopkins et al. (2024) would require 620,000 batches fed into GPT-4. Each batch (i.e., 750 words per input) corresponds to roughly 1,000 input tokens, per OpenAI's suggested benchmark. Since the outputs were standardized, the outputs for these analyses tended to be around 150 tokens.

Thus, when broken down into tokens, the total number of processed input tokens for this analysis would be $1,000 \times 620,000$ and the total processed output tokens would be $150 \times 620,000$. When factoring the cost per token for input and output tokens, the total cost comes to $\$8,990 = (1,000 \times 620,000 \times 0.00001) + (150 \times 620,000 \times 0.00003)$. While this is a loose estimate, it illustrates the challenges posed by the marginal per-sample cost of automated LLM annotation for large-N CSS research. Using our approach, labeling 1,000 text samples and training a supervised classifier would cost under $15.

Implementing our proposed workflow also reduces annotation labor costs. For example, hiring crowd-source workers to label a subset of text samples to serve as training observations would still cost significantly more than using automated annotation. Hopkins et al. (2024), for example, hire MTurk workers and paid them $0.06 to $0.07 per task depending on the total number of annotations ($15.00 per hour for six tasks per minute), which extrapolates to 360 tasks per hour. Under the standard assumption of three MTurk workers per task and taking a majority vote, the entire annotation time to label 1,000 tweets would have taken slightly under three hours and cost $124. However, due to serious data quality concerns about crowd-workers (Chmielewski and Kucker, 2020; Douglas et al., 2023; Veselovsky et al., 2023), a better cost comparison is against trained research assistants instead. Assuming 45 seconds per task and a $15 hourly rate, manually annotating 1,000 text samples would take 12.5 hours and cost approximately $187.

Table A1 shows a comparison of these costs. Not only is automated annotation remarkably faster than human annotators, our procedures introduced here can cost researchers less than 10% the cost of typical alternatives. These efficiency gains are

[10] See https://openai.com/pricing

conservative in the sense that they disregard the time to find, hire, and train the annotator.[11]

## B Appendix: Data sets

In this section, we elaborate on the data sets used in our analysis. Our corpus includes 14 classification tasks across five data sets representing recent applications in computational social science. To avoid the potential for contamination, we rely exclusively on data sets stored in password-protected data archives (e.g., Dataverse). We draw from research published in outlets across a spectrum of disciplines ranging from interdisciplinary publications (e.g., *Proceedings of the National Academy of Sciences*) to high-impact field journals in social science (e.g., *American Journal of Political Science*). To find these articles, we searched journals for articles related to computational social science that implemented some type of manual annotation procedure. The human-labeled data from the original study is treated as the ground truth. We discuss the human annotation procedures in the original studies at greater length in Appendix C.3.

It is important to note that while the raw data (e.g., tweets and Facebook posts) may be included in the LLM pretraining data, the accompanying labels from the human annotators are certainly not included in the pretraining data. This is because the labels accompanying each text sample (e.g., whether a tweet referenced a specific racial identity frame) are not public-facing. If the text without the associated label is not included in the pretraining data, there is no cause for concern that the annotation task would suffer from contamination.

Table A2 and Table A3 contain the full details for every task and data set. Overall, our data encompass diverse degrees of class imbalance: Across tasks, the mean positive class frequency is 16.2%, the minimum is 0.04%, and the maximum is 61%. The sources of labels are representative of common approaches to annotation: 42.9% of tasks were annotated by crowdsourced workers, 28.6% by experts, and 28.6% by research assistants.

Our replications involve fine-tuning supervised classifiers using manually annotated data from the replication data sets. For every replication classification task, we conformed to each data repository's replication policies. Each of the original studies received IRB approval and our analyses conformed to the same safety protocols, including full anonymization and agreeing to not publicly post the raw data without permission. As such, our replication of each data set is compatible with its intended usage.

Although all of the data sets were anonymized before our replications, we manually reviewed each data set to confirm privacy protections. One of the data sets (Saha et al., 2023) contains hate speech, but this is because it is a central part of the research question from the original study. As a result, we include examples of hate speech in that particular replication. From manual review, no other data set contained offensive material.

## C Appendix: Additional methodological details

### C.1 Prompt tuning

As discussed in Section 2, for every task we adjusted each GPT-4 prompt with a human-in-the-loop update procedure to optimize for accurate annotations. This human-in-the-loop process involved three steps. First, we used the generative LLM to annotate a small subset of the text samples per task (n=250).[12] Second, we manually reviewed instances where humans and the generative LLM disagreed on the text's label. Because our accuracy at this stage hovered around 0.8, this usually entailed manually reviewing roughly 50 text labels. Third, we adjusted the prompt instructions to clarify instances where automated annotation failed to correctly align with human judgment.

The prompt tuning process should be minimal (e.g., one or two iterations), because any further efforts could lead to overfitting the prompt to a small subset of the data (Egami et al., 2022). If the prompt is overly tailored to a small subset of the data, then the instructions may not generalize to unseen data. Moreover, if the researcher makes major changes to the prompt, there may be a mismatch between the human annotator's codebook and the generative LLM's instructions. Like the previous concern, the differences in the instructions could lead to poor performance on a held-out set. As a result, if there are substantial changes made to the LLM's prompt, then the researcher

[11] It is worth stressing here that validation against human-created labels is still essential. Therefore, researchers may want to prioritize their budgets for hiring domain experts to code a small subset of data to serve as validation and test data, as we demonstrate in Figure 1. Our cost efficiency calculations are based on training data, not validation and test sets.

[12] This subset of text samples was not included in the held-out test set.

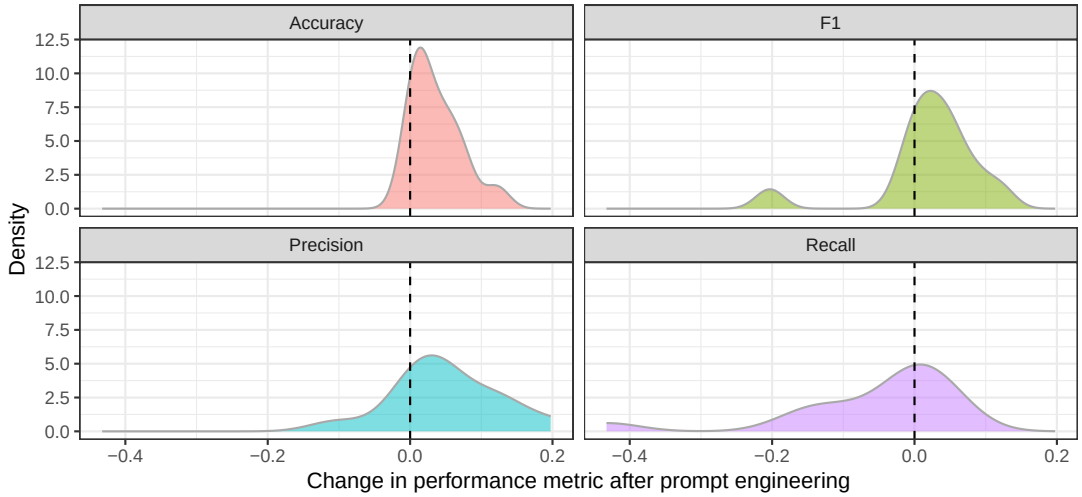


Figure A3: Change in LLM annotation performance on training data after one round of prompt optimization

should also change the human codebook as well and re-annotate new text samples. As such, these procedures should not be resource or time intensive. Instead, prompt tuning is intended to be a part of a validation process of few-shot in-context learning.

Some researchers argue that small changes to the LLM prompt instructions can dramatically alter automated annotation performance (Reiss, 2023), whereas others claim that alterations have a marginal effect (Rytting et al., 2023). To test how variations in the prompt instructions affect performance, we evaluated automated annotation performance before and after the prompt tuning process.

Figure A3 shows the distributions of change in performance metrics after updating the LLM prompt and re-annotating the same text samples. This analysis demonstrates whether and how prompt optimization affects LLM annotation, holding constant the data and conceptual categories. In most cases, prompt optimization led to minor improvement in accuracy and F1—although recall decreased in more cases than improved after updating the prompts. The small magnitude of change in classification performance suggests that generative LLMs are fairly robust to slight word changes in the prompt, which aligns with prior work that conducts similar experiments (Rytting et al., 2023). While the magnitude of improvement was generally small, researchers experiencing subpar LLM annotation performance can use human-in-the-loop prompt optimization to ensure that their instructions are not the cause of poor performance.

Qualitatively, the most common mistakes we observed by the generative LLM during the prompt optimization stage were false positives stemming from the text sample containing language broadly associated with the conceptual category of interest. For example, one task focused on identifying immigration content in American political speeches (Card et al., 2022). Initially, the generative model consistently categorized a text sample as containing an immigration reference if the speech mentioned a foreign country or foreign national, irrespective of whether the mention was connected to immigration in any way. For the prompt-update process for this task, changes in this case meant clarifying that any reference to a foreign country or foreign national did not warrant a positive class instance unless it was explicitly referenced in relation to American immigration or immigration policy. While this process was manual, we also believe that future work could conduct these procedures algorithmically—plausibly using generative AI as well.

### C.2 Hyperparameter tuning, evaluation, and compute details

Our experiments involved varying the training data used to fine-tune numerous supervised classifiers (i.e., 250 human samples, 1000 human samples, and 1000 GPT-labeled samples). To select each supervised classifier, we implemented a grid search over 18 possible hyperparameter combinations. In particular, we optimized learning rate (1e-5, 2e-5, and 5e-5), batch size (8 and 16), and epochs (2, 4, and 6). We conducted our search on a subsample of 250 text samples per task and retained the best hyperparameters (in terms of highest F1) across each task. We subsequently used the best-performing

combination of hyperparameters for all applications of a specific model (see best-performing hyperparameter configurations in Table A5). Despite not adopting a more exhaustive approach to hyperparameter tuning, we observe strong performance across our classification tasks, with a few exceptions. Table A6 displays additional model hyperparameters that remained constant across tasks, as well as basic information about each model's architecture.

Overall, for each task we had a total of 2,500 labeled text samples labeled by both human annotators and the LLM: (1) a training set of 1,000 text samples; (2) two separate validation sets (both with n=250); and (3) a test set (n=1000). Each of these sets of data were labeled by humans and the generative LLM. The training set (n=1000) was used to fine-tune the supervised classifiers. The first validation set (n=250) was used to optimize the generative LLM prompt and validate its few-shot performance. The second, separate validation set (n=250) was used to conduct our grid search. The test set (n=1000) was used to assess the final performance of the few-shot model and the supervised models.

For all 14 tasks, evaluation was conducted on a test set of 1000 held-out text samples that had previously been labeled by human annotators. To harmonize the diverse range of annotation tasks into a common framework for evaluation, we treat every task dimension as a separate binary annotation task. Thus, if an article included a classification task with three potential labels, we split the annotation process into three discrete binary classification tasks. As is standard in binary classification evaluation, we report accuracy, F1, precision, and recall for every task and model.[13] Table A4 displays the full classification results across all tasks and models.

All of our supervised training analyses were implemented in Python 3.10.12 with HuggingFace's Transformers (Wolf et al., 2020) and PyTorch libraries (Paszke et al., 2019). We conducted all data preprocessing in Python Pandas (McKinney, 2011). Our computing infrastructure was Google Colab, where we used 215 T4 GPU compute units (roughly 421.4 GPU hours). As with our model selection, we chose this computing environment due to its low cost and ease of application. Any computational social scientist could conduct the same analyses. In the supplementary material, we include all code to run our supervised training procedures.

[13] Because our tasks are all binary, there is no need to report any multi-label classification metrics, like Macro-F1.

### C.3 Additional details on human annotation procedures

We introduce a novel corpus of labeled text data for annotations. To create this data set, we compile labeled data from recent studies, as detailed in A2. As a result, we did not work with annotators to generate any original data. We adopted materials from these original studies instead. While we do not report the instructions given to each study's human annotators, we do provide the prompt instructions that were used to query GPT-4 in the supplementary material. These instructions were taken directly from the original study's human annotator instructions. All additional details on the annotation procedures (e.g., how they were recruited, payment, consent, and demographic characteristics) can be found in the original studies' supplementary material.

While we do not describe each study's procedures in detail, we manually selected our annotation studies due to their high-quality human labeling practices. All of the replicated studies were approved by an IRB. These studies all deployed either expert coders or numerous non-expert coders of varying backgrounds. Because all of the human annotation text is part of the peer-review process in high-impact journals and due to the strict annotation guidelines and principles these studies adhered to, we conclude that the human annotations are of high-quality.

## D Appendix: Extended results

Figure A4 shows precision-recall (PR) curves for each of the BERT-family models trained on either human labels or GPT labels, pooling all classification tasks. The decrease in performance for GPT-generated labels compared with human labels is small based on area under the curve (AUC). Thus, supervised classifiers trained with GPT-generated labels perform comparably to classifiers trained with human-generated labels on these tasks. Across models and tasks, precision appears to drop below 1.0 around 0.7 recall.

## E Appendix: Ablation experiments

We conducted a variety of ablation experiments to account for sources of variance. The next three

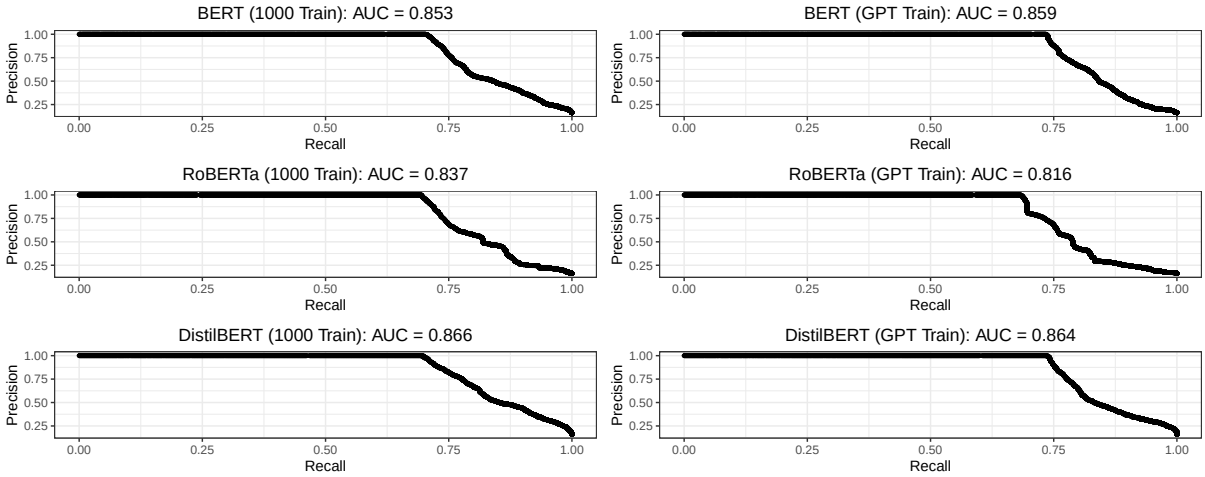


Figure A4: Precision-recall curves across each BERT-family model

.

sections detail these experiments and their main findings.

### E.1 Comparing classifiers with different model size and architecture

First, to account for variation in model architecture and model size, we compare performance across two additional language models for supervised classification (i.e., XLNet and Mistral-7B). These models are beyond the BERT-family models included in the main analyses (i.e., BERT, DistilBERT, and RoBERTa). In addition to a Mistral-7B supervised sequence classification model, we also generate few-shot labels using Mistral-7B using the same procedures we employed in the GPT-4 few-shot model.

The primary difference between the BERT-family models and XLNet is the training objective. The BERT-based models are pretrained using a Masked Language Modeling (MLM) objective, whereas XLNet is an autoregressive model that uses Permutation Language Modeling (PLM), which involves learning context across input tokens in any permutation order. In addition to being significantly larger than the BERT models, Mistral-7B utilizes a distinct type of attention in the pretraining process (i.e., grouped-query attention (GQA) and sliding window attention (SWA)). We include the Mistral-7B few-shot model as a smaller, open-source alternative to GPT-4. Mistral-7B was selected because the model weights are available for download and it displays higher performance than Llama-13B (Jiang et al., 2023).

Figure A5 shows the classification performance from these additional models and compares them to the results from BERT and GPT-4 few-shot in the main analyses. The test set for these analyses is the same as the main analysis shown in the paper. Our results from examining these additional models do not change the substantive conclusions in the paper: Models trained on surrogate training labels perform comparably to models trained with human labeled data. XLNet even performs slightly better than the fully human labels. The gap between Mistral-7b fine-tuned using human labels and GPT-labels, however, is notably larger than the other models, with a median difference of 0.12. Overall, BERT and GPT-4 still appear to be the strongest performing models.

There is also a fairly sizeable gap between the open-source (Mistral-7B) and closed-source (GPT-4) few-shot models. Although it may be expected from a significantly smaller and free-to-use model, F1 scores for Mistral-7B are 0.16 worse, on average, than GPT-4. Mistral-7B also took significantly longer to run than GPT-4. These findings further reinforce the necessity of human validation.

### E.2 Comparing classifiers with and without noise

Our second set of ablation experiments involve comparing supervised models trained on GPT-generated labels *with noise* against GPT-generated labels *without noise*. To measure noise in the GPT-labels, we utilize the predicted token sampling process of generative LLMs to gauge an LLM's "confidence" in the annotation of each text sample. By introducing randomness in the LLM sampling

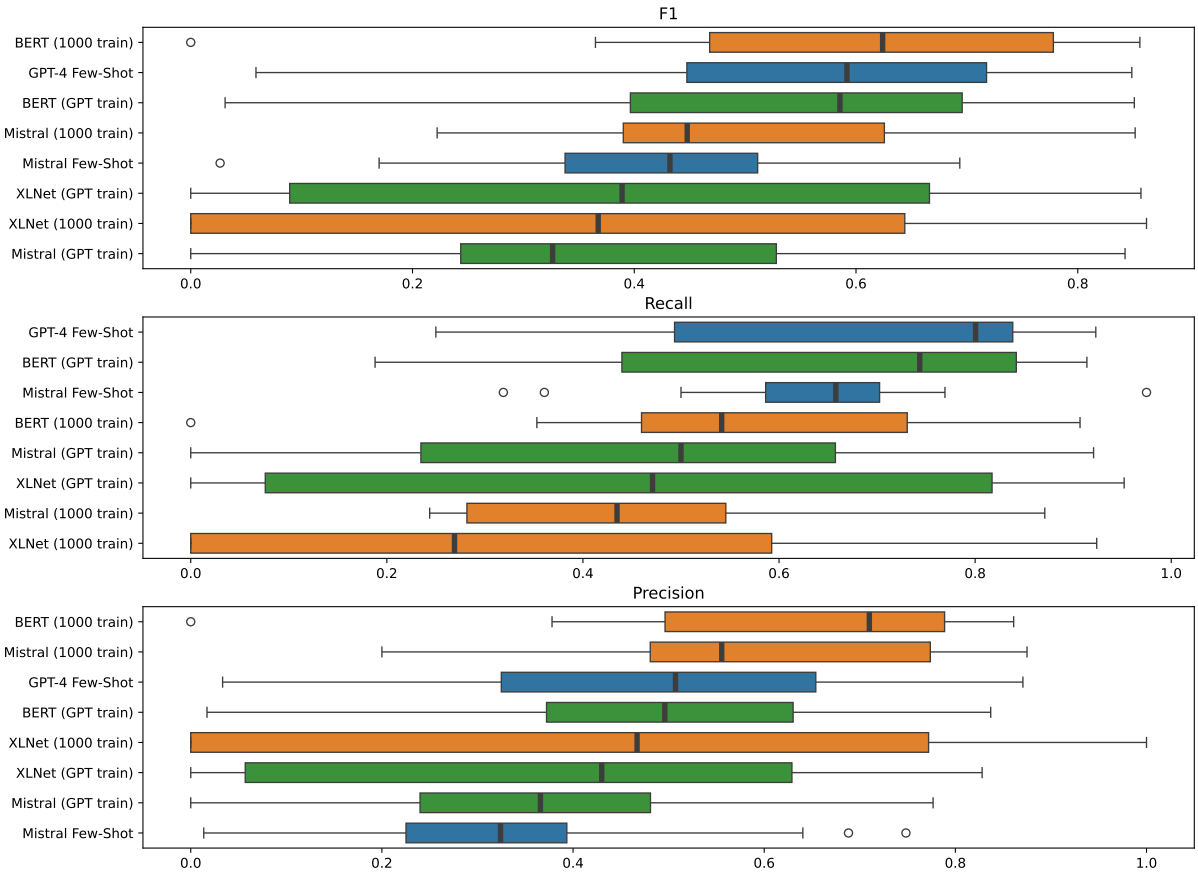


Figure A5: Box plots of ablation performance on test data across 14 tasks. Thick vertical line denotes median.

.

process through the temperature setting and by repeatedly classifying the same text sample multiple times, we identify text samples that cannot be clearly classified into one of the annotation categories specified by the prompt instructions.[14]

Classifications that vary across iterations may be "edge cases" and have a lower probability of correct classification.[15] This approach rests on the core assumption that the full distribution of token probabilities captures latent information about the annotation's classification. If, for example, the top tokens are similar in probability, then choosing one of these tokens may misrepresent the model's annotation decision. Instead, measuring the variability across iterations allows us to find these "edge cases." We call this measure of uncertainty in the annotation label a "consistency score." We define an indicator function $C(i)$ that is equal to 1 when label $i$ for a given task is equal to the LLM's modal classification, $m$, for task :

$$C(i) = \begin{cases} 1 & \text{if } i = m \\ 0 & \text{otherwise} \end{cases}$$

Given a vector of classifications, $\mathbf{a}$, with length $l$ for a given classification task, *consistency* is measured as the proportion of classifications that match the modal label:

$$Consistency = \frac{1}{l} \sum_{j=1}^{l} C(a_j)$$

For these ablation experiments, we classify every text sample three times at a temperature of 0.7 and measure each text sample's consistency score. Because there are only three iterations, each text sample can only have two values for consistency score: 0.67 and 1.0. Across all analyzed tasks, classifications with a consistency of 1.0 show significantly higher accuracy (19.4% increase), true positive rate (16.4% increase), and true negative rate (21.4% increase) compared to classifications with a consistency less than 1.0. Roughly 85% of classifications had a consistency of 1.0.

[14] Generative LLMs output a series of probabilities that correspond to each token in its vocabulary. To select a specific token from this probability distribution, generative LLMs sample a randomly selected token, weighted by its probability. The temperature hyperparameter governs this sampling process. A higher temperature setting flattens the probability distribution and causes the sampling draw to become more uniform across tokens. A lower temperature, however, isolates the sampling to select only the most likely tokens.

[15] Accessing token log probabilities directly, once available, will be an effective way to a similar analysis.

Table A7 shows supervised model performance for BERT models fine-tuned on 1,250 training observations labeled by GPT-4 (i.e., labels with noise) compared to BERT models fine-tuned on training observations with a consistency score of 1.0 (i.e., labels without noise), which reduced our training set to slightly more than 1000 samples per task. Put otherwise, the second series of models involved dropping about 250 text samples per task so that the training set only retained annotations where GPT-4 consistently labeled the same category across all iterations.

Our findings indicate that there are minimal differences between models trained on labels with noise and labels without noise. Models trained without noise display, on average, 0.004 lower F1 score than models trained with noise. These results suggest that the supervised models explored here are fairly robust to noise in the labels.

### E.3 Comparing GPT-4 few-shot performance over time

Our final set of ablation experiments involved replicating the GPT-4 few-shot model at different points in time. An unsettling scenario involves the potential drift in capabilities as generative LLMs undergo opaque changes and updates. Some research, such as Chen et al. (2023a), claim that GPT-4 performance is declining over time. To account for the potential of changing model weights in GPT-4, we re-analyzed each task six months after our initial analyses and compared results across time.

Figure A6 shows evaluation comparisons of few-shot tasks in both April 2023 and November 2023. Our results do not suggest significant changes in GPT-4 performance over time. If anything, Figure A6 reveals a small *increase* in performance since my initial experiments. Across the 14 tasks, accuracy improved by 0.007 and F1 increased by 0.022 when the same annotation procedures were carried out in November 2023.

## F Appendix: Miscellaneous additional information

Additional sources:

- Robot image (used in Figure 1): https://commons.wikimedia.org/wiki/File:Grey_cartoon_robot.png
- Human silhouette image (used in Figure 1): https://commons.wikimedia.org/wiki/File:SVG_Human_Silhouette.svg

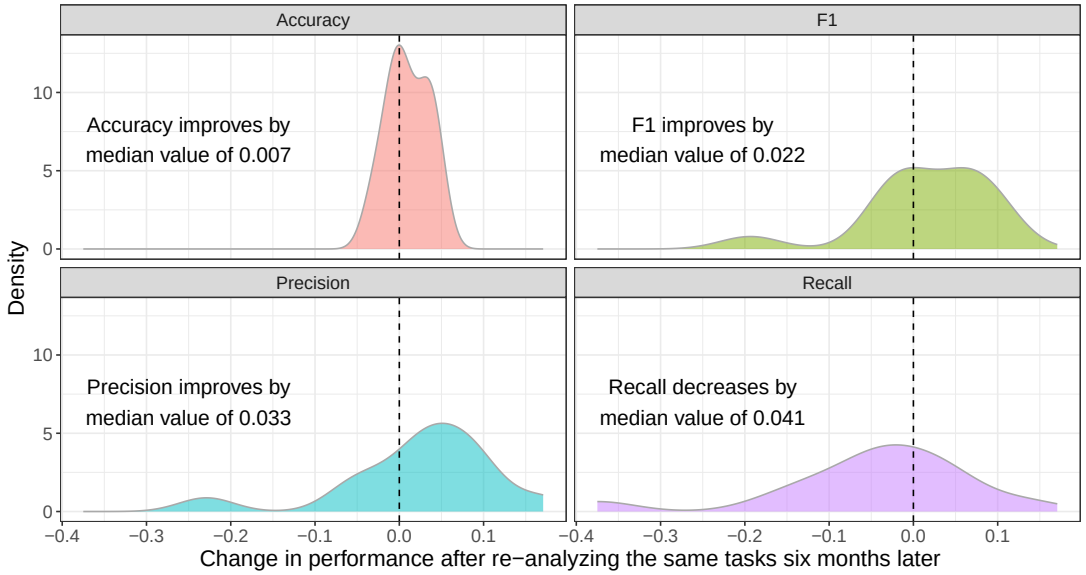


Figure A6: Examining GPT-4 performance over time

| Author(s) | Title | Journal | Year |
|---|---|---|---|
| Card et al. | Computational analysis of 140 years of US political speeches reveals more positive but increasingly polarized framing of immigration | PNAS | 2022 |
| Hopkins, Lelkes, and Wolken | The Rise of and Demand for Identity-Oriented Media Coverage | American Journal of Political Science | 2024 |
| Müller | The Temporal Focus of Campaign Communication | Journal of Politics | 2021 |
| Peng, Romero, and Horvat | Dynamics of cross-platform attention to retracted papers | PNAS | 2022 |
| Saha et al. | On the rise of fear speech in online social media | PNAS | 2022 |

Table A2: Replication data sources.

| Study | # of tasks | Annotation source | Classification tasks |
|---|---|---|---|
| Card et al. (2022) | 4 | Research assistants | Classify US congressional speeches to identify whether the speech discussed immigration or immigration policy, along with an accompanying tone: pro-immigration, anti-immigration, or neutral. |
| Hopkins, Lelkes, and Wolken (2024) | 4 | Crowd | Classify headlines, Tweets, and Facebook share blurbs to identify references to social groups defined by a) race/ethnicity; b) gender/sexuality; c) politics; d) religion. |
| Müller (2021) | 3 | Expert | Classify sentences from political party manifestos for temporal direction: past, present, or future. |
| Peng, Romero, and Horvat (2022) | 1 | Expert | Classify whether Tweets express criticism of findings from academic papers. |
| Saha et al. (2020) | 2 | Crowd | Classify social media posts into fear speech, hate speech, both, or neither. |

Table A3: Descriptions of replication classification tasks.

| Data set | Task | Model | Training data: Few shot Ac. | Few shot F1 | Few shot Pr. | Few shot Re. | Human: 250 Ac. | Human: 250 F1 | Human: 250 Pr. | Human: 250 Re. | Human: 1000 Ac. | Human: 1000 F1 | Human: 1000 Pr. | Human: 1000 Re. | GPT: 1000 Ac. | GPT: 1000 F1 | GPT: 1000 Pr. | GPT: 1000 Re. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Card et al. | Cat: Neg | GPT-4 | 0.85 | 0.65 | 0.54 | 0.83 | | | | | | | | | | | | |
| | | BERT | | | | | 0.88 | 0.58 | 0.74 | 0.48 | 0.87 | 0.56 | 0.65 | 0.49 | 0.81 | 0.56 | 0.47 | 0.72 |
| | | RoBERTa | | | | | 0.85 | 0.51 | 0.59 | 0.45 | 0.84 | 0.48 | 0.55 | 0.42 | 0.78 | 0.57 | 0.43 | 0.82 |
| | | DistilBERT | | | | | 0.86 | 0.56 | 0.61 | 0.51 | 0.86 | 0.58 | 0.61 | 0.55 | 0.81 | 0.58 | 0.47 | 0.74 |
| | Cat: Imm | GPT-4 | 0.81 | 0.81 | 0.74 | 0.90 | | | | | | | | | | | | |
| | | BERT | | | | | 0.85 | 0.84 | 0.79 | 0.89 | 0.86 | 0.86 | 0.81 | 0.91 | 0.84 | 0.83 | 0.76 | 0.91 |
| | | RoBERTa | | | | | 0.86 | 0.85 | 0.80 | 0.92 | 0.85 | 0.84 | 0.77 | 0.92 | 0.82 | 0.82 | 0.74 | 0.92 |
| | | DistilBERT | | | | | 0.85 | 0.84 | 0.80 | 0.88 | 0.84 | 0.84 | 0.79 | 0.89 | 0.82 | 0.82 | 0.73 | 0.92 |
| | Cat: Neut. | GPT-4 | 0.83 | 0.26 | 0.27 | 0.25 | | | | | | | | | | | | |
| | | BERT | | | | | 0.80 | 0.35 | 0.29 | 0.44 | 0.85 | 0.36 | 0.38 | 0.35 | 0.87 | 0.38 | 0.44 | 0.34 |
| | | RoBERTa | | | | | 0.88 | 0.30 | 0.46 | 0.23 | 0.88 | 0.00 | 0.00 | 0.00 | 0.84 | 0.33 | 0.33 | 0.34 |
| | | DistilBERT | | | | | 0.85 | 0.28 | 0.32 | 0.25 | 0.85 | 0.36 | 0.37 | 0.35 | 0.86 | 0.38 | 0.40 | 0.36 |
| | Cat: Pro | GPT-4 | 0.88 | 0.50 | 0.55 | 0.46 | | | | | | | | | | | | |
| | | BERT | | | | | 0.86 | 0.33 | 0.44 | 0.27 | 0.84 | 0.44 | 0.42 | 0.46 | 0.87 | 0.45 | 0.51 | 0.40 |
| | | RoBERTa | | | | | 0.87 | 0.37 | 0.51 | 0.30 | 0.84 | 0.37 | 0.41 | 0.34 | 0.85 | 0.41 | 0.43 | 0.39 |
| | | DistilBERT | | | | | 0.87 | 0.29 | 0.55 | 0.19 | 0.83 | 0.38 | 0.38 | 0.37 | 0.84 | 0.35 | 0.40 | 0.31 |
| Hopkins et al. | Political | GPT-4 | 0.88 | 0.43 | 0.30 | 0.79 | | | | | | | | | | | | |
| | | BERT | | | | | 0.95 | 0.32 | 0.60 | 0.22 | 0.96 | 0.62 | 0.71 | 0.54 | 0.82 | 0.34 | 0.21 | 0.82 |
| | | RoBERTa | | | | | 0.84 | 0.37 | 0.23 | 0.85 | 0.96 | 0.62 | 0.73 | 0.54 | 0.84 | 0.37 | 0.23 | 0.85 |
| | | DistilBERT | | | | | 0.94 | 0.29 | 0.50 | 0.20 | 0.96 | 0.63 | 0.72 | 0.56 | 0.83 | 0.34 | 0.22 | 0.80 |
| | Gender | GPT-4 | 0.95 | 0.74 | 0.68 | 0.82 | | | | | | | | | | | | |
| | | BERT | | | | | 0.91 | 0.20 | 0.46 | 0.13 | 0.96 | 0.80 | 0.86 | 0.74 | 0.94 | 0.72 | 0.62 | 0.85 |
| | | RoBERTa | | | | | 0.91 | 0.08 | 0.44 | 0.04 | 0.95 | 0.73 | 0.78 | 0.68 | 0.92 | 0.67 | 0.54 | 0.87 |
| | | DistilBERT | | | | | 0.94 | 0.52 | 0.83 | 0.38 | 0.97 | 0.81 | 0.87 | 0.75 | 0.93 | 0.71 | 0.59 | 0.88 |
| | Race | GPT-4 | 0.96 | 0.57 | 0.41 | 0.92 | | | | | | | | | | | | |
| | | BERT | | | | | 0.97 | 0.00 | 0.00 | 0.00 | 0.98 | 0.56 | 0.71 | 0.46 | 0.98 | 0.64 | 0.54 | 0.77 |
| | | RoBERTa | | | | | 0.97 | 0.00 | 0.00 | 0.00 | 0.97 | 0.00 | 0.00 | 0.00 | 0.97 | 0.59 | 0.45 | 0.85 |
| | | DistilBERT | | | | | 0.97 | 0.00 | 0.00 | 0.00 | 0.99 | 0.71 | 0.77 | 0.65 | 0.97 | 0.54 | 0.46 | 0.65 |
| | Religion | GPT-4 | 0.98 | 0.61 | 0.47 | 0.88 | | | | | | | | | | | | |
| | | BERT | | | | | 0.98 | 0.21 | 1.00 | 0.12 | 0.99 | 0.73 | 0.75 | 0.71 | 0.98 | 0.61 | 0.48 | 0.82 |
| | | RoBERTa | | | | | 0.98 | 0.00 | 0.00 | 0.00 | 0.98 | 0.00 | 0.00 | 0.00 | 0.98 | 0.00 | 0.00 | 0.00 |
| | | DistilBERT | | | | | 0.98 | 0.00 | 0.00 | 0.00 | 0.99 | 0.69 | 0.67 | 0.71 | 0.97 | 0.53 | 0.37 | 0.94 |
| Müller | Future | GPT-4 | 0.82 | 0.85 | 0.87 | 0.83 | | | | | | | | | | | | |
| | | BERT | | | | | 0.83 | 0.85 | 0.88 | 0.84 | 0.82 | 0.85 | 0.85 | 0.85 | 0.81 | 0.85 | 0.84 | 0.87 |
| | | RoBERTa | | | | | 0.84 | 0.87 | 0.87 | 0.88 | 0.82 | 0.85 | 0.86 | 0.85 | 0.82 | 0.86 | 0.84 | 0.87 |
| | | DistilBERT | | | | | 0.83 | 0.86 | 0.85 | 0.86 | 0.81 | 0.84 | 0.87 | 0.82 | 0.82 | 0.85 | 0.83 | 0.88 |
| | Past | GPT-4 | 0.91 | 0.74 | 0.66 | 0.84 | | | | | | | | | | | | |
| | | BERT | | | | | 0.94 | 0.83 | 0.74 | 0.93 | 0.95 | 0.83 | 0.80 | 0.85 | 0.93 | 0.79 | 0.71 | 0.89 |
| | | RoBERTa | | | | | 0.94 | 0.80 | 0.81 | 0.79 | 0.95 | 0.85 | 0.79 | 0.92 | 0.85 | 0.00 | 0.00 | 0.00 |
| | | DistilBERT | | | | | 0.94 | 0.79 | 0.77 | 0.80 | 0.94 | 0.80 | 0.79 | 0.82 | 0.93 | 0.79 | 0.68 | 0.96 |
| | Present | GPT-4 | 0.82 | 0.62 | 0.64 | 0.60 | | | | | | | | | | | | |
| | | BERT | | | | | 0.83 | 0.65 | 0.66 | 0.64 | 0.83 | 0.65 | 0.64 | 0.66 | 0.81 | 0.61 | 0.63 | 0.58 |
| | | RoBERTa | | | | | 0.84 | 0.66 | 0.71 | 0.61 | 0.84 | 0.68 | 0.68 | 0.67 | 0.83 | 0.61 | 0.68 | 0.56 |
| | | DistilBERT | | | | | 0.83 | 0.64 | 0.69 | 0.59 | 0.83 | 0.65 | 0.66 | 0.64 | 0.82 | 0.59 | 0.66 | 0.54 |
| Peng et al. | Critical | GPT-4 | 0.85 | 0.54 | 0.48 | 0.63 | | | | | | | | | | | | |
| | | BERT | | | | | 0.87 | 0.43 | 0.59 | 0.34 | 0.91 | 0.63 | 0.76 | 0.54 | 0.79 | 0.43 | 0.35 | 0.56 |
| | | RoBERTa | | | | | 0.88 | 0.44 | 0.61 | 0.34 | 0.87 | 0.62 | 0.54 | 0.73 | 0.78 | 0.43 | 0.34 | 0.59 |
| | | DistilBERT | | | | | 0.83 | 0.43 | 0.42 | 0.44 | 0.86 | 0.54 | 0.50 | 0.58 | 0.77 | 0.41 | 0.33 | 0.56 |
| Saha et al. | CV | GPT-4 | 0.97 | 0.06 | 0.03 | 0.25 | | | | | | | | | | | | |
| | | BERT | | | | | 1.00 | 0.00 | 0.00 | 0.00 | 1.00 | 0.00 | 0.00 | 0.00 | 0.94 | 0.03 | 0.02 | 0.25 |
| | | RoBERTa | | | | | 1.00 | 0.00 | 0.00 | 0.00 | 1.00 | 0.00 | 0.00 | 0.00 | 0.93 | 0.05 | 0.03 | 0.50 |
| | | DistilBERT | | | | | 1.00 | 0.00 | 0.00 | 0.00 | 0.99 | 0.00 | 0.00 | 0.00 | 0.94 | 0.10 | 0.05 | 0.75 |
| | HD | GPT-4 | 0.88 | 0.35 | 0.28 | 0.45 | | | | | | | | | | | | |
| | | BERT | | | | | 0.91 | 0.17 | 0.24 | 0.13 | 0.92 | 0.41 | 0.45 | 0.38 | 0.90 | 0.21 | 0.24 | 0.19 |
| | | RoBERTa | | | | | 0.92 | 0.24 | 0.35 | 0.19 | 0.92 | 0.47 | 0.43 | 0.52 | 0.91 | 0.20 | 0.26 | 0.16 |
| | | DistilBERT | | | | | 0.91 | 0.26 | 0.32 | 0.22 | 0.91 | 0.40 | 0.38 | 0.42 | 0.91 | 0.28 | 0.33 | 0.25 |

Table A4: Complete task-by-task classification performance results. Ac., Pr., and Re. refer to accuracy, precision, and recall, respectively.

| Study | Task | Hyperparameters |
|---|---|---|
| Card et al. | Classify immigration speeches | learning rate (5e-05), batch size (8), epochs (4) |
| | Classify pro-immigration speeches | learning rate (5e-05), batch size (16), epochs (6) |
| | Classify anti-immigration speeches | learning rate (5e-05), batch size (8), epochs (6) |
| | Classify neutral immigration speeches | learning rate (5e-05), batch size (8), epochs (4) |
| Hopkins et al. | Classify race/ethnicity | learning rate (5e-05), batch size (8), epochs (4) |
| | Classify gender | learning rate (5e-05), batch size (8), epochs (6) |
| | Classify political groups | learning rate (5e-05), batch size (16), epochs (6) |
| | Classify religious groups | learning rate (5e-05), batch size (8), epochs (6) |
| Müller | Classify past | learning rate (5e-05), batch size (8), epochs (4) |
| | Classify present | learning rate (5e-05), batch size (8), epochs (4) |
| | Classify future | learning rate (2e-05), batch size (8), epochs (6) |
| Peng et al. | Classify criticism | learning rate (5e-05), batch size (8), epochs (6) |
| Saha et al. | Classify fear speech | learning rate (5e-05), batch size (8), epochs (6) |
| | Classify hate speech | learning rate (5e-05), batch size (8), epochs (4) |

Table A5: Hyperparameter settings per task.

| | BERT-base | RoBERTa-base | DistilBERT | XLNet-base | Mistral-7B |
|---|---|---|---|---|---|
| # parameters | 110m | 125m | 66m | 110m | 7b |
| # attention heads | 12 | 12 | 12 | 12 | 32 |
| Hidden dim. | 768 | 768 | 768 | 768 | 4096 |
| Feedforward dim. | 3072 | 3072 | 3072 | 3072 | 14336 |

Table A6: Model architectures and additional hyperparameters.

| Data set and task | BERT F1 score (training obs w/o noise) | BERT F1 score (training obs w/ noise) | Difference |
|---|---|---|---|
| Hopkins (AJPS): Political | 0.340 | 0.344 | -0.004 |
| Hopkins (AJPS): religion | 0.609 | 0.609 | 0.000 |
| Hopkins (AJPS): gender | 0.716 | 0.684 | 0.032 |
| Hopkins (AJPS): race | 0.635 | 0.640 | -0.005 |
| Muller (JOP): future | 0.851 | 0.851 | 0.000 |
| Muller (JOP): past | 0.791 | 0.755 | 0.036 |
| Muller (JOP): present | 0.606 | 0.601 | 0.005 |
| Card (PNAS): cat_imm | 0.832 | 0.815 | 0.017 |
| Card (PNAS): cat_anti | 0.565 | 0.573 | -0.008 |
| Card (PNAS): cat_neutral | 0.385 | 0.428 | -0.043 |
| Card (PNAS): cat_pro | 0.448 | 0.436 | 0.012 |
| Peng (PNAS) | 0.431 | 0.444 | -0.013 |
| Saha (PNAS): CV | 0.031 | 0.059 | -0.028 |
| Saha (PNAS): HD | 0.210 | 0.276 | -0.066 |

Table A7: Comparing BERT F1 score for models fine-tuned with and without noise