Feeding LLM Annotations to BERT Classifiers at Your Own Risk

Anonymous ACL submission

Abstract

Using LLM-generated labels to fine-tune smaller encoder-only models for text classification has gained popularity in various settings. While this approach may be justified in simple and low-stakes applications, we conduct empirical analysis to demonstrate how the perennial curse of training on synthetic data manifests itself in this specific setup. Compared to models trained on gold labels, we observe not only the expected performance degradation in accuracy and F1 score, but also increased instability across training runs and premature performance plateaus. These findings cast doubts on the reliability of such approaches in real-world applications. We contextualize the observed phenomena through the lens of error propagation and offer several practical mitigation strategies, including entropy-based filtering and ensemble techniques. Although these heuristics offer partial relief, they do not fully resolve the inherent risks of propagating non-random errors from LLM annotations to smaller classifiers, underscoring the need for caution when applying this workflow in high-stakes text classification tasks.

1 Introduction

Text classification remains a crucial application of LLMs. In settings where unlabeled data is abundant but gold labels and computational resources are scarce, recent work (e.g., Golde et al. (2023); Pangakis and Wolken (2024a); Mohamed Serouis and Sèdes (2024)) suggested fine-tuning smaller encoder-only language models, such as BERT (Devlin et al., 2019) using LLM-generated labels as training samples. This strategy promises to strike a balance between performance and cost, and has become increasingly popular across commercial, academic, and policy applications, some of which carry potentially high societal impact. Examples range from healthcare (Kumichev et al., 2024; Smolyak et al., 2024) to legal analysis (Freitas, 2024; Colombo et al., 2024), and to policy decision making (Dell, 2024; Halterman and Keith, 2025).

However, the reliability of such approaches remains under-explored. Previous work often treats LLM-generated labels as adequate approximations of human annotations, focusing narrowly on performance parity (Wang et al. (2021); Csanády et al. (2024); Pangakis and Wolken (2024b)). This overlooks risks inherent to synthetic data training, such as error propagation and model collapse—issues well-documented in broader machine learning literature (Bauer et al., 2024; Liu et al., 2024; Shumailov et al., 2024a). These gaps are particularly consequential in applied settings like computational social science, where researchers increasingly leverage LLM annotations for large text corpora despite lacking validation mechanisms (Hopkins et al.). While prior work has studied synthetic text-label pairs (Kuo et al., 2024; Li et al., 2023), our focus on label generation alone addresses a more common real-world constraint: abundant unlabeled text data paired with expensive annotation processes.

We address this gap through experiments on four benchmark datasets of varying complexity, demonstrating that the trade-offs of training with LLM-generated labels extend beyond modest accuracy/F1 degradation. In summary, our main contributions are:

1. **Empirical Analysis of Synthetic Label Training:** We reveal how synthetic labels erode prediction robustness and leads to early performance plateau — dimensions often ignored in prior analyses. These phenomena persist across datasets, contradicting assumptions of "more data always helps."

2. **Evaluation of Mitigation Strategies:** We test entropy-based filtering (removing low-confidence LLM labels) and consistency ensembles (aggregating multiple LLM annota-

tions), showing they recover only 60–75% of the gold-label performance gap. More critically, neither strategy stabilizes training variance or mitigates early plateaus, underscoring fundamental limitations of post hoc corrections.

Our findings challenge the premise that synthetic labels are a "safe" substitute for human annotations, even in ostensibly simple classification tasks. The paper proceeds as follows: Section 2 details baseline experimental protocols, while Section 3 analyzes performance degradation, instability, and performance plateau alongside theoretical interpretation. Sections 4 and 5 evaluate mitigation strategies, concluding with implications for practitioners relying on LLM-generated training data.

2 Baseline Experiments

2.1 Methods

We compare classifiers fine-tuned on LLM-generated labels vs. gold labels across four datasets chosen for task diversity and difficulty:

- **IMDB**: balanced binary sentiment analysis
- **ECommerce**: slightly imbalanced multi-class product categorization
- **Manifestos**: nuanced political stance detection, imbalanced data, smaller training size
- **Toxic**: hate speech vs. offensive language detection on twitter texts, highly imbalanced

Details about these datasets are in Appendix A. Following prior work, we use `roberta-base` with standard classification heads as our encoder-only classifiers. For annotation, we use `Qwen2.5-Instruct` (3b, 7b), as representatives LLMs in their respective weight classes (Yang et al., 2024). Using three-shot prompts, we generate synthetic labels for training texts while withholding gold labels. Fine-tuning details are Appendix B. Few shot classification details are in Appendix C.

2.2 Evaluation Metrics

Accuracy and Macro-F1. We evaluate the overall performance by looking at accuracy and macro-F1. Since we are especially interested in the stability of our classification models, we perform each experiment five times and compute the variance of accuracy and macro-F1 as well.

Stability at the Individual Level. In addition to variation in overall performances, another important indicator to consider in high-stakes situations is prediction stability at the individual level. We measure this using Krippendorff's Alpha α_K which quantifies inter-rater agreement across training run and the proportion of unchanged predictions p_{uc} across five trials, providing an intuitive measure of model decisiveness.

3 Baseline Results

Non-Random Performance Degradation Unsurprising, models trained on synthetic labels consistently underperform those trained on gold labels across all datasets, with the performance gap widening as task complexity increases. On IMDB, a simple benchmark first introduced in 2011, the performance difference is negligible. However, for multi-class classification on Ecommerce , models trained on labels from the 3B parameter LLM suffer a dramatic 30-point accuracy drop (66.05% versus 96.26% with gold labels). Notably, scaling up to a 7B parameter model fails to bridge this gap, achieving only 92.74% accuracy. The discrepancy between accuracy and F1 scores on the Manifestos and Toxic datasets reveals a more nuanced issue: LLM-generated labels lead to systematic failures in modeling tail distributions. Through manual error analysis, we found that both the LLM annotator and subsequently trained RoBERTa classifier consistently underperform on minority classes. This phenomenon can be interpreted as a mild form of model collapse during synthetic data training, as described by (Shumailov et al., 2024b), where the model fails to adequately learn tail distributions.

Performance Plateau As Figures 1 and 2 shows, models trained on synthetic labels exhibit premature performance plateaus compared to those trained on gold labels, showing diminishing returns as training data increases. The observed plateaus can be attributed to the propagation of systematic errors present in LLM annotations, as documented by (Chen et al., 2022) in few-shot learning contexts, as well as by (Li et al., 2023)'s findings regarding LLMs' difficulties with subjective classification tasks.

Prediction Instability Does Not Decrease with LLM Size Perhaps the most concerning finding is that models trained on synthetic labels exhibit significant prediction instability, and this instability persists even when using larger LLMs for label generation. On the Manifestos dataset, we observe a dramatic drop in Krippendorff's alpha (α_K) from

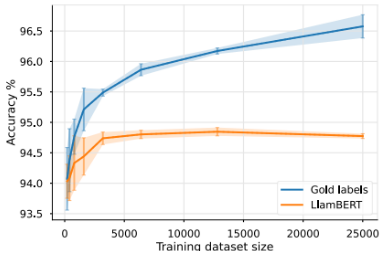

Figure 1: Performance of RoBERTa trained on "Gold labels" vs on synthetic labels ("LlamBERT"). Plot from Csanády et al. (2024), as we limit training to 5000 data points to reduce environmental impact

84.30 with gold labels to 52.72 with 3B labels, and this further deteriorates to 43.75 with 7B-generated labels. The proportion of unchanged predictions (p_{uc}) tells a similar story, dropping from 82.04% to 50% and 30.45% respectively. This pattern holds across other datasets, though to varying degrees. Even on the simpler IMDB task, where accuracy remains competitive, we still see a consistent decline in prediction stability metrics. The Toxic dataset particularly highlights this issue, where using 7B-generated labels leads to high variance in predictions ($p_{uc} = 77.49\%$) despite relatively strong accuracy scores. These results suggest that models trained on synthetic labels not only underperform but also make less consistent predictions across different training runs.

3.1 Theoretical Interpretation

Framework Denote the true data generating process of text and label pair as the joint distribution $P(Y, X)$, where Y is label/class, and X is input text. The supervised text classifier is trained to estimate the conditional distribution $P(Y|X)$ from i.i.d. sample $D_P = \{(y_i, x_i)_{i=1}^N\}$ by minimizing cross-entropy loss:

$$\mathcal{L}_{\text{CE}}(\theta, D_P) = -\frac{1}{N}\sum_{i=1}^{N} \log \hat{P}(y_i|\mathbf{x}_i; \theta)$$

However, since we are using labels generated from LLM, the data we see is actually drawn from [1]

$$\begin{aligned} D_S = \{(y_i, x_i)_{i=1}^N\} &\sim P(X)P_S(Y|X) \\ &\nsim P(X)P(Y|X) \end{aligned}$$

[1]Incidentally, from this formulation, one can see that when a large pool of unlabeled text is available, using synthetic labels is theoretically superior than using synthetic text and label pairs, as it avoids additional LLM approximation error on the marginal distribution of input text $P(X)$.

where subscript S stands for synthetic. Consequently, the expected [2] KL-divergence between true target conditional distribution $P(Y|X)$ and the the learned distribution $\hat{P}(Y|X)$ can be decomposed as (Heskes, 1998):

$$\begin{aligned} Error(\hat{P}) &= E_{D_S}\Big[KL\big(P \,\|\, \hat{P}\big)\Big] = \\ &= KL(P\|P_S) + E_{D_S}\Big[KL\big(P_S \,\|\, \hat{P}\big)\Big] \end{aligned}$$

where the first term represents the irreducible approximation error coming from P_S, and the second terms is the estimation error coming from training.

Interpretation Crucially, the first term irreducible approximation error implies that no amount of synthetic labels can remove the systematic biases LLM annotators introduces, leading to performance plateau. The decomposition also helps explain the amplification of instability when training on synthetic labels. In addition to the usual finite sample estimation errors, in regions where $P_S(Y|X)$ is particularly off from $P(Y|X)$, even small fluctuations in the synthetic data can lead to larger estimation errors. Essentially, the estimation error can be amplified by the underlying approximation error, leading to more variance in performance across different training runs.

4 Mitigation Experiments

As the theoretical framework suggests, the key driver of performance degradation is the divergence between the true conditional distribution $P(Y|X)$ and the LLM-generated distribution $P_S(Y|X)$. Intuitively, one way to mitigate this error is to filter out unreliable LLM-generated labels and increase the signal to noise ratio to control the error size. For any given input text x, we can try to control the size of error by mixing in true labels to obtain a better conditional distribution

$$\begin{aligned} P_F(y|X = x) = F(x)P_S(y|X = x) \\ +(1 - F(x))P(y|X = x) \end{aligned}$$

where $F(x)$ is the data-dependent filtering function. In principle, one could parameterize F and treat it as a learnable function. However, in our low resource setup, we resort to computationally cheap heuristics.

We evaluate mitigation strategies using the 7B LLM annotator with RoBERTa-base, selected for its balance of performance and practical relevance.

[2]expected since $\hat{P}$ depends on the realization of synthetic sample D_S

Model	Label	IMDB			Ecommerce			Manifestos			Toxic		
		μ_{acc}	μ_{f1}	α_K	μ_{acc}	μ_{f1}	α_K	μ_{acc}	μ_{f1}	α_K	μ_{acc}	μ_{f1}	α_K
		σ_{acc}	σ_{f1}	p_{uc}	σ_{acc}	σ_{f1}	p_{uc}	σ_{acc}	σ_{f1}	p_{uc}	σ_{acc}	σ_{f1}	p_{uc}
RoBERTa-Base	Gold	93.84	93.82	90.6	96.26	96.23	96.69	83.56	79.38	84.30	91.13	75.24	84.08
		0.28	0.29	90	0.25	0.22	95.24	0.69	0.54	82.04	0.23	1.01	88.50
	3B	93.33	93.31	89.99	66.05	66.62	75.04	66.02	41.91	52.72	86.86	65.18	57.29
		0.16	0.16	89.68	3.14	3.69	70.14	0.00	3.68	50	1.74	1.93	47.39
	7B	92.95	92.94	87.28	92.74	92.88	79.18	71.51	60.62	43.75	83.89	56.53	68.49
		0.20	0.20	86.56	0.81	0.76	69.35	1.30	7.96	30.45	5.31	5.04	77.49

Table 1: Experimental results across different models, label types, and datasets. For each dataset, we report average accuracy μ_{acc}, average macro F1-score μ_{f1}, standard deviation of accuracy σ_{acc}, and standard deviation of macro F1 σ_{f1}. In addition, we compute Krippendorff's alpha α_K and the proportion of predictions that remain unchanged across experimental runs p_{uc}. All numbers are scaled up by 100 for ease of presentation.

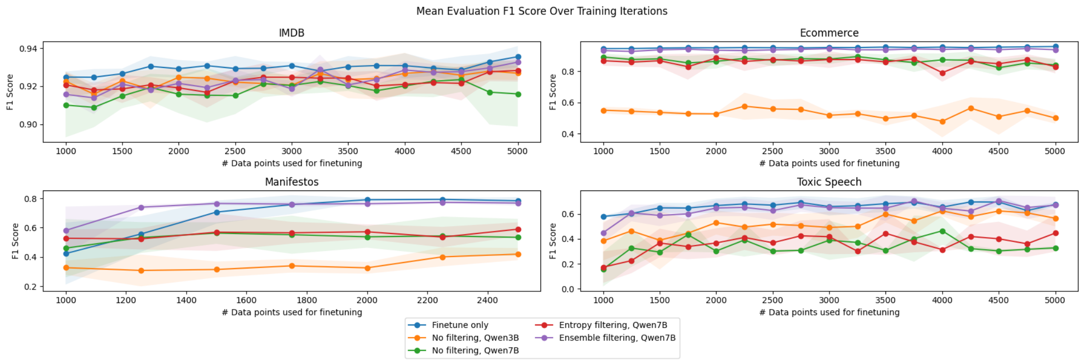

Figure 2: Performance as data point increases

Entropy-Ranking Filtering For each input x, we compute the conditional entropy of the LLM's class predictions:

$$H(Y|X = x) = -\sum_{y \in \mathcal{Y}} p(y|x) \log p(y|x)$$

where $p(y|x)$ is the LLM's predicted probability for class y. Entropy is commonly used as a baseline for assessing uncertainty (Huang et al., 2024). We rank predictions and replace LLM annotations with gold labels if they are in the top $\alpha \in \{5, 25\}$ percent. Importantly, we are not using a fixed threshold to sidestep temperature scaling, and to account for the fact that many out-of-shelf LLMs are poorly calibrated (Desai and Durrett, 2020; Zhu et al., 2023). Note that in binary classification, entropy ranking is equivalent to logits-ranking, which is the approach Wang et al. (2021) took.

Consistency Ensemble Another simple fix is prompt LLM to generate multiple predictions with different demonstrations. The idea is that a robust prediction should not depend too much on the specific examples we provide in the prompt. We replace cases where predictions flip with human annotations.

5 Mitigation Results

Entropy-based Filtering does not work well. While for IMDB, Ecommerce, Manifestos, entropy-based filtering stabilizes predictions to a certain. On Toxic, on the contrary, it leads to lowers the proportion of unchanged predictions u_{pc} from 77.49 to 56.20. Given that entropy-based filtering is theoretical more appealing simple alternative simple uncertainty estimation heuristics including logits or log-probabilities, this does not bode well for prospect of having a cheap fix for the instability problem we identify.

Consistency Ensemble seems to work, but at a cost. Experiments suggests that consistency ensemble seems to manage to pick up many of LLM annotations that are distorting decision boundary for the classifiers. However, we need to be careful about this approach, however, because it requires multiple inferences on the same data point. For example, with 5 percent of the total unlabeled pool, 5-time ensemble means we are effectively performing inference on 25% of the pool, which defeats the purpose of cost saving.

Model	Label Type	IMDB			Ecommerce			Manifestos			Toxic		
		μ_{acc}	μ_{f1}	α_K	μ_{acc}	μ_{f1}	α_K	μ_{acc}	μ_{f1}	α_K	μ_{acc}	μ_{f1}	α_K
		σ_{acc}	σ_{f1}	p_{uc}	σ_{acc}	σ_{f1}	p_{uc}	σ_{acc}	σ_{f1}	p_{uc}	σ_{acc}	σ_{f1}	p_{uc}
RoBERTa-Base	Gold	93.84	93.82	90.6	96.26	96.23	96.69	83.56	79.38	84.30	91.13	75.24	84.08
		0.28	0.29	90	0.25	0.22	95.24	0.69	0.54	82.04	0.23	1.01	88.50
	Entropy	93.28	93.27	89.33	91.79	91.85	83.09	71.06	62.48	59.80	81.56	61.42	68.44
		0.32	0.32	89	0.71	0.61	76.89	2.11	2.63	52.82	3.77	3.67	56.20
	Ensemble	93.46	83.45	89.43	95.14	95.15	93.87	81.58	77.97	82.55	89.17	74.05	61.48
		0.02	0.02	94.72	0.20	0.14	95.54	0.24	0.40	81.51	0.69	0.86	77.20

Table 2: Experimental results across different models, label types, and datasets. For each dataset, we report average accuracy μ_{acc}, average macro F1-score μ_{f1}, standard deviation of accuracy σ_{acc}, and standard deviation of macro F1 σ_{f1}. In addition, we compute Krippendorff's alpha α_K and the proportion of predictions that remain unchanged across experimental runs p_{uc}. All numbers are scaled up by 100 for ease of presentation.

6 Conclusion

In this short paper, we identify previously overlooked risks in using LLM-generated labels to train smaller text classifiers: performance drops, unstable predictions, and early plateaus in learning. These problems are worse for complex tasks and minority classes, which can amplify existing biases (Gallegos et al., 2024). While we tested some fixes like filtering and ensembles, they only partially address these problems. As Chen et al. (2024) warns about rushing to adopt LLMs without proper scrutiny, our results provide concrete evidence of risks in this specific use case. While using LLM-generated labels might work for simple tasks, we urge caution in critical applications.

7 Limitations and Ethical Considerations

Limitations. One clear limitation is that our work does not offer a comprehensive solution to the problem we identified. While we explored a few heuristic mitigation strategies, we did not investigate more sophisticated approaches. For instance, our theoretical discussion suggests that using the embeddings as inputs to a simple ridge regression on a small validation set could help predict where the LLM is likely to make mistakes, thereby guiding targeted improvements through higher-quality annotations. However, given the scope of this short paper, we leave more in depth exploration of best strategies to LLM-generated labels to text classification pipeline to future work.

A second limitation stems from the rapid evolution of foundation models. As state-of-the-art models become increasingly capable of approximating the conditional distribution $P(Y|X)$ arbitrarily well, our approach may become less relevant. Nonetheless, we welcome such advancements as they contribute positively to the field.

Finally, our theoretical analysis touches on the impact of approximation error, yet it lacks a rigorous exposition of how this error influences the variance and convergence rates of our estimates. Addressing this gap remains an important avenue for future research.

Ethical Considerations. We do not foresee significant ethical risks associated with our work. On the contrary, our paper cautions against the uncritical adoption of pipelines that utilize LLM-generated labels to fine-tune BERT-like models for classification.

Use of AI We acknowledge the use of artificial intelligence tools to assist with code debugging and prose refinement throughout this work.

References

André Bauer, Simon Trapp, Michael Stenger, Robert Leppich, Samuel Kounev, Mark Leznik, Kyle Chard, and Ian Foster. 2024. Comprehensive exploration of synthetic data generation: A survey. *Preprint*, arXiv:2401.02524.

Hanjie Chen, Guoqing Zheng, Ahmed Hassan Awadallah, and Yangfeng Ji. 2022. Pathologies of pre-trained language models in few-shot fine-tuning. *Preprint*, arXiv:2204.08039.

Zhiyu Zoey Chen, Jing Ma, Xinlu Zhang, Nan Hao, An Yan, Armineh Nourbakhsh, Xianjun Yang, Julian McAuley, Linda Petzold, and William Yang Wang. 2024. A survey on large language models for critical societal domains: Finance, healthcare, and law. *Preprint*, arXiv:2405.01769.

Pierre Colombo, Telmo Pessoa Pires, Malik Boudiaf, Dominic Culver, Rui Melo, Caio Corro, Andre F. T. Martins, Fabrizio Esposito, Vera Lúcia Raposo, Sofia Morgado, and Michael Desa. 2024. Saullm-7b: A pioneering large language model for law. *Preprint*, arXiv:2403.03883.

Bálint Csanády, Lajos Muzsai, Péter Vedres, Zoltán Nádasdy, and András Lukács. 2024. Llambert: Large-scale low-cost data annotation in nlp. *Preprint*, arXiv:2403.15938.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. *Preprint*, arXiv:1703.04009.

Melissa Dell. 2024. Deep learning for economists. *Preprint*, arXiv:2407.15339.

Shrey Desai and Greg Durrett. 2020. Calibration of pre-trained transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 295–302, Online. Association for Computational Linguistics.

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.

Lucas José Gonçalves Freitas. 2024. Text clustering applied to unbalanced data in legal contexts. In *Proceedings of the 16th International Conference on Computational Processing of Portuguese - Vol. 1*, pages 639–642, Santiago de Compostela, Galicia/Spain. Association for Computational Lingustics.

Isabel O. Gallegos, Ryan A. Rossi, Joe Barrow, Md Mehrab Tanjim, Sungchul Kim, Franck Dernoncourt, Tong Yu, Ruiyi Zhang, and Nesreen K. Ahmed. 2024. Bias and fairness in large language models: A survey. *Preprint*, arXiv:2309.00770.

A. Gautam. 2019. E commerce text dataset (version - 2).

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. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 1–11, Singapore. Association for Computational Linguistics.

Andrew Halterman and Katherine A. Keith. 2025. Codebook llms: Evaluating llms as measurement tools for political science concepts. *Preprint*, arXiv:2407.10747.

Tom Heskes. 1998. Bias/variance decompositions for likelihood-based estimators. *Neural Computation*, 10(6):1425–1433.

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

Hsiu-Yuan Huang, Yutong Yang, Zhaoxi Zhang, Sanwoo Lee, and Yunfang Wu. 2024. A survey of uncertainty estimation in llms: Theory meets practice. *Preprint*, arXiv:2410.15326.

Gleb Kumichev, Pavel Blinov, Yulia Kuzkina, Vasily Goncharov, Galina Zubkova, Nikolai Zenovkin, Aleksei Goncharov, and Andrey Savchenko. 2024. *MedSyn: LLM-Based Synthetic Medical Text Generation Framework*, page 215–230. Springer Nature Switzerland.

Hsun-Yu Kuo, Yin-Hsiang Liao, Yu-Chieh Chao, Wei-Yun Ma, and Pu-Jen Cheng. 2024. Not all llm-generated data are equal: Rethinking data weighting in text classification. *Preprint*, arXiv:2410.21526.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Zhuoyan Li, Hangxiao Zhu, Zhuoran Lu, and Ming Yin. 2023. Synthetic data generation with large language models for text classification: Potential and limitations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10443–10461, Singapore. Association for Computational Linguistics.

Ruibo Liu, Jerry Wei, Fangyu Liu, Chenglei Si, Yanzhe Zhang, Jinmeng Rao, Steven Zheng, Daiyi Peng, Diyi Yang, Denny Zhou, and Andrew M. Dai. 2024. Best practices and lessons learned on synthetic data. In *First Conference on Language Modeling*.

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. *Preprint*, arXiv:1907.11692.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Ibrahim Mohamed Serouis and Florence Sèdes. 2024. Leveraging llms for fair data labeling and validation in crowdsourcing environments [vision paper]. In *2024 IEEE International Conference on Big Data (BigData)*, pages 468–472.

Stefan Müller. 2020. Replication Data for: The Temporal Focus of Campaign Communication.

Nicholas Pangakis and Sam Wolken. 2024a. Knowledge distillation in automated annotation: Supervised text classification with LLM-generated training labels. In *Proceedings of the Sixth Workshop on Natural Language Processing and Computational Social Science (NLP+CSS 2024)*, pages 113–131, Mexico City, Mexico. Association for Computational Linguistics.

Nicholas Pangakis and Samuel Wolken. 2024b. Keeping humans in the loop: Human-centered automated annotation with generative ai. *Preprint*, arXiv:2409.09467.

Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Yarin Gal, Nicolas Papernot, and Ross Anderson. 2024a. The curse of recursion: Training on generated data makes models forget. *Preprint*, arXiv:2305.17493.

Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, et al. 2024b. Ai models collapse when trained on recursively generated data. *Nature*, 631:755–759.

Daniel Smolyak, Margrét V Bjarnadóttir, Kathy Crowley, and Ritu Agarwal. 2024. Large language models and synthetic health data: progress and prospects. *JAMIA Open*, 7(4):ooae114.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2020. How to fine-tune bert for text classification? *Preprint*, arXiv:1905.05583.

Shuohang Wang, Yang Liu, Yichong Xu, Chenguang Zhu, and Michael Zeng. 2021. Want to reduce labeling cost? GPT-3 can help. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4195–4205, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Brandon T Willard and Rémi Louf. 2023. Efficient guided generation for llms. *arXiv preprint arXiv:2307.09702*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Chiwei Zhu, Benfeng Xu, Quan Wang, Yongdong Zhang, and Zhendong Mao. 2023. On the calibration of large language models and alignment. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 9778–9795, Singapore. Association for Computational Linguistics.

A Dataset Descriptions

IMDB The Stanford Large Movie Review Dataset (Maas et al., 2011), IMDB for short, needs no introduction among NLP practitioners.

E-commerce (Gautam, 2019) The Ecommerce dataset contains 50,425 product listings scraped from Indian ecommerce platforms, consisting of product titles and descriptions. Each item is categorized into one of four classes: Electronics, Household, Books, or Clothing and Accessories. The dataset is slightly imbalanced across these four classes, with each product represented by its textual description.

Manifestos (Müller, 2020) The Manifesto Project dataset comprises annotated political texts, including party election manifestos from 50+ countries, labeled with policy positions and topics. We focus on the English-language subset, which includes over 4,000 documents annotated at the sentence level. Each sentence is categorized into one of 56 policy areas (e.g., "Environment," "Education"). The dataset is widely used for political text analysis and multi-label classification tasks. We preprocess the text to remove metadata and retain only sentences with unambiguous policy labels.

Toxic speech (Davidson et al., 2017) This dataset contains 24,802 tweets annotated via crowdsourcing into three categories: *hate speech*, *offensive language*, or *neither*. Tweets were collected using a crowd-sourced lexicon of hate speech keywords, and annotations emphasize distinguishing hate speech (targeted attacks on protected groups) from general offensiveness. The dataset is imbalanced, with most tweets labeled as offensive. Racist and homophobic content is more reliably classified as hate speech, while sexist remarks are often misclassified as merely offensive. We use this dataset to evaluate nuanced hate speech detection, focusing on precision-recall trade-offs. To reduce environmental impacts, we limit the number of data points for train to up to 5000 for all datasets and shrink the size of test datasets with $<= 2000$ by randomly drawing from existing test sets.

B Fine-tuning Details

We employ Huggingface's pre-trained weights for both BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) as provided in the Transformers library (Wolf et al., 2020). We conduct full fine-tuning of the pre-trained language models following Sun et al. (2020), without freezing any pre-trained layers. The classification head consists of a dropout layer (set at the default value 0.1) followed by a linear layer that maps the [CLS] token representation to dimension of the target label space.

While extensive hyperparameter tuning could potentially yield better performance, we prioritize consistent experimental conditions across datasets to isolate the effects of synthetic labels on performance stability. As a result, our baseline performance on gold-label fine-tuning may be slightly below state-of-the-art, but provides a fair foundation for comparative analysis.

Training runs for 3 epochs with a batch size of 16 for training and 32 for evaluation. We use the AdamW optimizer with a learning rate of 2e-5 and weight decay of 0.01. A linear learning rate scheduler with a warmup ratio of 0.05 is applied. The best checkpoint is selected based on validation F1 score, with a maximum of 2 checkpoints saved during training to conserve storage. All experiments use mixed-precision training (FP16) and are conducted on a single NVIDIA RTX 8000 GPU.

C LLM Annotation Details

We utilize `vLLM` (Kwon et al., 2023) for improved memory efficiency and to better simulate a production environment. In addition, guided decoding (Willard and Louf, 2023) is imposed to ensure that the outputs follow a consistent format. In particular, the model is constrained to generate only two tokens: the first token is the predicted class token (with labels mapped to integers) and the second token is the end-of-sequence (`<EOS>`) marker. The annotation pipeline uses a structured prompt template that puts together a task description, label description, demonstrations (randomly drawn from training datasets), and input text as follows:

```
### Instruction ###
{task description}
Respond with only the label name, nothing else.
### Available Labels ###
{label description}
### Examples ###
{demonstrations}
### Input ###
Text to classify: {input_text}
### Output ###
Label:
```

Dataset	**Task Description**	**Label Mapping**
IMDB	You are an AI assistant specializing in sentiment analysis of movie reviews. You are going to help classify movie reviews as positive or negative.	{"0": “negative”, "1": “positive”}
Ecommerce	You are an AI assistant and you are very good at doing ecommerce products classification. You are going to help a customer to classify the products on the ecommerce website.	{"0": “books”, "1": “clothing & accessories”, "2": “electronics”, "3": “household”}
Manifestos	You are an AI assistant specializing in classifying the temporal alignment of political party manifestos.You are going to help classify political party manifestos as about the future, the present, or the past.	{"0": “present”, "1": “future”, "2": “past”}
Toxic	You are an AI assistant specializing in detecting hate speech and offensive language. You are going to help classify tweets as hate speech, offensive language, or neither.	{"0": “hate speech”, "1": “offensive language”, "2": “neither”}

Table 3: Task specifications for various datasets.