Benchmarking Debiasing Methods for LLM-based Parameter Estimates

Anonymous ACL submission

Abstract

Large language models (LLMs) offer an inexpensive yet powerful way to annotate text, but are often inconsistent when compared with experts. These errors can bias downstream estimates of population parameters such as regression coefficients and causal effects. To mitigate this bias, researchers have developed debiasing methods such as Design-based Supervised Learning (DSL) and Prediction-Powered Inference (PPI), which promise valid estimation by combining LLM annotations with a limited number of expensive expert annotations.

Although these methods produce consistent estimates under theoretical assumptions, it is unknown how they compare in finite samples of sizes encountered in applied research. We make two contributions: First, we study how each method's performance scales with the number of expert annotations, highlighting regimes where LLM bias or limited expert labels significantly affect results. Second, we compare DSL and PPI across a range of tasks, finding that although both achieve low bias with large datasets, DSL often outperforms PPI on bias reduction and empirical efficiency, but its performance is less consistent across datasets. Our findings indicate that there is a bias-variance tradeoff at the level of debiasing methods, calling for more research on developing metrics for quantifying their efficiency in finite samples.

1 Introduction

Large language models (LLMs) are transforming disciplines that use text as a form of evidence in testing theories, something particularly evident in computational social science (Ziems et al., 2024; Törnberg, 2024; Bail, 2024; Argyle et al., 2023). LLMs can be used to extract features that are important for substantive research questions, such as theoretical constructs (e.g., political ideology; Sim et al., 2013) or stylistic properties of the text (e.g., tone; El-Haj et al., 2016 and politeness; Priya et al., 2024), and epidemiology (Kino et al., 2021). The use of LLMs promises to speed up the process of annotating these variables, which would previously have required hand-annotation by experts. This shift has led to more agile research workflows in which researchers can use larger amounts of data and more variation in their analyses.

However, although powerful, these models do not annotate in a way that is fully consistent with expert annotators (Lin and Zhang, 2025); the distribution of errors can be heterogeneous or correlated with other variables of interest. This means that the estimates derived from the LLM annotations are likely to be biased, resulting in misleading substantive interpretations (McFarland and McFarland, 2015). To handle these biases, *debiasing* methods have been developed, most prominently Prediction-Powered Inference (PPI) (Angelopoulos et al., 2023) and Design-based Supervised Learning (DSL) (Egami et al., 2023, 2024). Both frameworks produce an unbiased downstream model by combining the LLM annotations with a smaller set of expert annotations. The biases in LLM-based estimates are then compensated for by a *rectifier* created by comparing the two sets of annotations for the samples that have both.

There is a lack of knowledge about *when* and *how much* debiasing methods provide added value in finite samples. There are no closed-form expressions that relate their efficacy to the allocation of expert versus model-generated annotations, leaving practitioners without analytic guidance on when one should prefer DSL or PPI over simply collecting more expert annotations. Accordingly, our research questions are the following.

RQ1: When is a debiased, large-scale LLM annotation dataset statistically preferable to a finite expert-only dataset for unbiased estimation of a population parameter?

RQ2: What are the performance differences be-

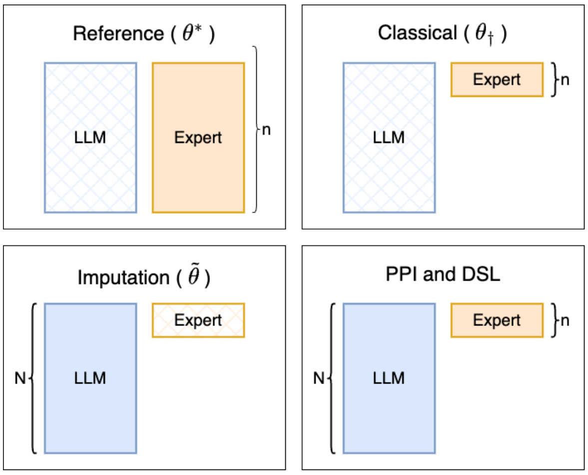

Figure 1: The reference model (top-left) is estimated from expert annotations for all samples in the dataset; the classical model (top-right) only from the expert samples available to the debiasing frameworks; the imputation model (bottom-left) only from the generated annotations for all samples. The *debiased* models (bottom-right) are estimated from *both* LLM annotations for all samples and expert annotations for a subset.

tween the debiasing methods and how do they vary across datasets and LLM-based annotators?

We tackle these questions by comparing PPI and DSL across four datasets and four annotation procedures. To our knowledge, ours is the first effort to compare debiasing methods directly.

2 Background: Debiasing Methods

Let $\mathcal{D} = \{(d_i, \mathbf{x}_i, \widehat{y}_i)\}_{i=1}^{N}$ be a corpus of documents d_i, with associated independent variables $\mathbf{x}_i \in \mathbf{X} = \{\mathbf{x}_i\}_{i=1}^{N}$ and LLM annotations $\widehat{y}_i \in \widehat{Y} = \{\widehat{y}_i\}_{i=1}^{N}$. A subset of $\mathcal{D}$ also has additional expert annotations $y_j \in Y_\dagger = \{y_j\}_{j=1}^{n}$, $n \leq N$. Expert annotations are taken to be the ground truth and are generally costly (Gilardi et al., 2023).

Next, we focus on a general parameter of interest θ, which represents the result of the downstream statistical analysis. For example, this could be a regression coefficient or a class prevalence rate. The goal of the debiasing methods is to create an estimator f which estimates θ based on $\mathbf{X}$, $\widehat{Y}$, and $Y_\dagger$. Ideally, the estimator should be *consistent*, meaning that $f(\mathbf{X}, \widehat{Y}, Y_\dagger) \to \theta$ as $N \to \infty$, and *precise*, meaning that we want to keep the variance and confidence intervals as small as possible.

One way to achieve this would be to ignore $\widehat{Y}$ entirely and only use the unbiased expert annotations $Y_\dagger$. We call this the *classical estimator* $\theta_\dagger = f(\mathbf{X}, Y_\dagger)$, which is usually generated by minimizing a loss. Although this estimator produces unbiased estimates, it can have a large variance if we have few expert annotations. We call the classical estimator trained with expert annotations for *all* samples the *reference estimator*, θ^*, which corresponds to the ideal but costly model that the debiasing methods are aiming towards.

Another approach would be to only use LLM annotations $\widehat{Y}$ and ignore the expert annotations $Y_\dagger$. We call this the *imputation estimator*, $\tilde{\theta} = f(\mathbf{X}, \widehat{Y})$. Here, we rely on the assumption that we can exchange the expert annotations for the LLM annotations. The hope is that, while LLM annotations might be noisier than expert annotations, we can counteract the noise by simply generating as many labels as needed, given a large enough corpus. However, the LLM may exhibit systematic biases different from those of the expert human annotators, meaning that $|\tilde{\theta} - \theta^*| > 0$ as $N \to \infty$, and therefore this assumption does not hold in general. In turn, this leads to a biased downstream estimate, and one runs the risk of being "precisely inaccurate" (McFarland and McFarland, 2015).

A third approach claims to be both unbiased and more precise than $\theta_\dagger$. Such methods typically work by estimating parameters on LLM annotations, with a *rectifier* constructed from the difference between the generated and expert annotations for the subset of the corpus for which we have both (see Figure 1). In this paper, we investigate PPI and DSL as two of the most prominent among these methods.

PPI offers a protocol for integrating LLM predictions into downstream statistical inference via first-order debiasing (Angelopoulos et al., 2023). It begins by treating the LLM predictions as if they were true labels and forming the "imputation estimate": $\tilde{\theta} = \operatorname{argmin}_\theta \frac{1}{N} \sum_{i=1}^{N} \ell_\theta(\mathbf{x}_i, \widehat{y}_i)$, where ℓ_θ is the loss defining our estimand, such as the binary cross-entropy for a logistic regression. In general $\tilde{\theta}$ is biased, so PPI introduces the *rectifier*, which, in the one parameter case equals

$$r_\theta = \mathbb{E}\big[\nabla_\theta \ell_\theta(\mathbf{x}_i, y_i) - \nabla_\theta \ell_\theta(\mathbf{x}_i, \widehat{y}_i)\big],$$

the gradient terms capturing the systematic distortion from substituting $\widehat{y}_i$ for the true y_i (the gradient difference reveals the bias direction in parameter space, which we then offset to debias). We estimate r_θ on the labeled sample and estimate the imputed gradient on the unlabeled set using plugin estimators. The final, first-order debiased estimate

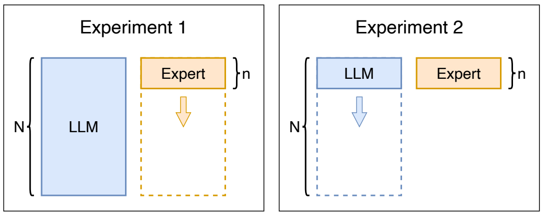

Figure 2: Setup for Experiments 1 and 2.

is then $\tilde{\theta} - \widehat{r}_\theta$. Because $\widehat{r}_\theta$ is estimated from sample averages, confidence sets can be readily obtained.

DSL (Egami et al., 2023, 2024) adopts a *design-based sampling* scheme, which assumes $\pi(\widehat{y}_i, \mathbf{x}_i) = \Pr(r_i = 1 \mid \widehat{y}_i, \mathbf{x}_i) > 0$, where $r_i \in \{0, 1\}$ denotes whether document i is labeled by experts and where $\pi(\cdot)$ is known. The data is partitioned into K folds and used to cross-fit $\widehat{g}_k$, a model to predict y_i as a function of $\widehat{y}_i$ and $\mathbf{x}_i$:

$$\widetilde{y}_i^k = \widehat{g}_k(\widehat{y}_i, \mathbf{x}_i) + \frac{r_i}{\pi(\widehat{y}_i, \mathbf{x}_i)}\big(y_i - \widehat{g}_k(\widehat{y}_i, \mathbf{x}_i)\big).$$

Then, $\mathbb{E}\big[\widetilde{y}_i \mid \widehat{y}_i, \mathbf{x}_i\big] = \mathbb{E}\big[y_i \mid \widehat{y}_i, \mathbf{x}_i\big]$ regardless of misspecification of $\widehat{g}_k$ via double robustness.

Many estimands admit a moment equation form: $\mathbb{E}\big[m(y_i, \mathbf{x}_i; \theta)\big] = 0$ (e.g., maximum likelihood). DSL solves the empirical analogue of the moment condition with the debiased $\widehat{y}_i$, using $\sum_{i=1}^{N} m\big(\widetilde{y}_i, \mathbf{x}_i; \theta\big) = 0$, where each $\widetilde{y}_i$ is constructed as above. Cross-fitting and M-estimation theory then yield consistent "sandwich" estimators of variance, giving valid confidence intervals.

3 Methodology

Our analysis focuses on two experiments, which we use to benchmark and contrast the $\theta_\dagger$, PPI, and DSL estimators (see Figure 2). In both experiments, we focus on a particular parameter of interest θ — the coefficients of a binary logistic regression. Specifically, for each dataset, we create a substantive task relating four independent variables $x_1 ... x_4$ to a binary outcome, y. The independent variables are either categorical or integers computed from text features. Each logistic regression, therefore, produces four coefficients $\beta_1 ... \beta_4$ and a y-intercept β_0 for a total of five parameters.See Appendix D for package use details.

Experiment 1. Our first experiment involves varying the number of expert annotations while keeping the total number of samples constant (see Figure 2, left). Our goal here is to answer the question: how do the debiasing methods improve with an increasing proportion of expert annotations? In other words, if one has a fixed number of data samples, how much budget should one allocate towards the expert annotations for debiasing?

For this experiment, we vary the number of expert samples logarithmically. We use a minimum of 200 expert annotations (below that threshold, debiasing methods became unstable). We additionally report the proportion of expert samples $\frac{n_i}{N}$ rather than the absolute number in order to compare datasets of different sizes. For each entry, dataset, and annotation procedure, we run 250 repetitions; we report 2σ confidence intervals over repetitions.

Experiment 2. In our second experiment, we instead vary the total number of samples while keeping the number of generated annotations constant (see Figure 2, right). We seek to answer: given a fixed expert annotation budget, how much does the effective sample size increase as one increases the number of generated annotations? We repeat experiments with 200, 1000, and 5000 annotations.

Like Experiment 1, we vary the number of total samples logarithmically. The minimum number of total samples is defined by the number of available expert samples. The maximum number of total samples is determined by the size of the available dataset, which varies. We report the proportion of total samples with respect to the total number of available samples to compare datasets. We use 250 repetitions to estimate the 2σ confidence interval.

Datasets and Annotations. We replicate our experiments over four datasets: Multi-domain Sentiment, Misinfo-general, Bias in Biographies, and Germeval18 (see Appendix A). We also compare performance across four LLM-model classes: BERT, DeepSeek v3, Phi-4, and Claude 3.7 Sonnet (see Appendix C). Input variables are either additional annotations available from the original dataset or quantities derived from the text, such as the text length in characters. We compare PPI, DSL, and $\theta_\dagger$ with the same number of annotations.

Evaluation Metrics. We evaluate performance of debiasing methods by comparing the respective models against reference model θ^*. Comparison between models is done using a standardized Root Mean Squared Error (sRMSE), which captures both bias and variance for a holistic performance assessment (see Appendix B). We standardize by scaling according to the reference model coefficients.

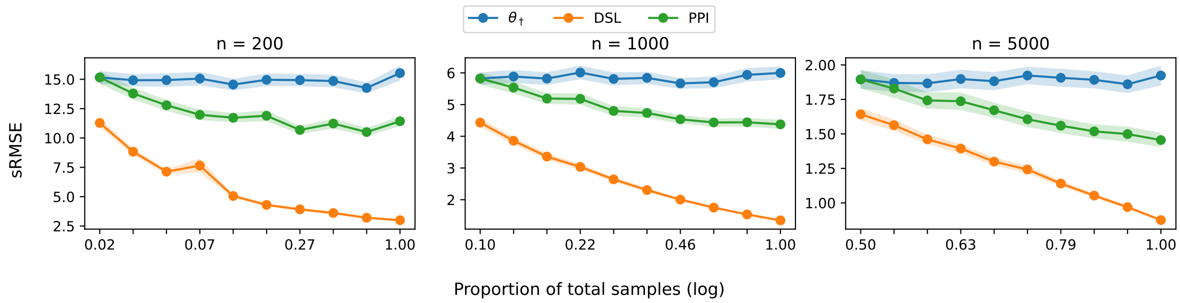

Figure 3: Results from the set of experiments varying the total number of samples, averaged over datasets and annotation methods. The x-axis shows the total number of samples (N) as a proportion of the total available samples in each dataset. The y-axis shows the sRMSE. The plots show results for $n = 200$, $n = 1000$, and $n = 5000$.

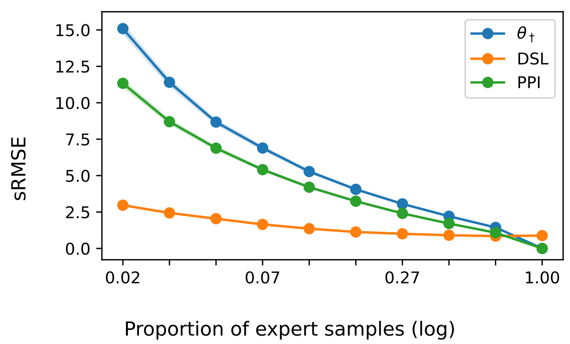

Figure 4: Results for Experiment 1 averaged over all datasets and annotation methods.

4 Results

In our experiments we contrasted $\theta_\dagger$, PPI, and DSL with the reference model θ^*. The only difference between $\theta_\dagger$ and the reference model θ^* is that they are trained on a different number of expert annotations — $\theta_\dagger$ is trained on only $Y_\dagger$, the expert annotations that would have been given to one of the debiasing methods . Accordingly, the smaller the proportion of expert annotations given to the debiasing methods, the more inaccurate $\theta_\dagger$ becomes, which is reflected as a high sRMSE. As we increase the proportion of expert annotations, $\theta_\dagger$ converges towards θ^*, and we observe a monotonically decreasing sRMSE. At a proportion of 1, there is no difference between $\theta_\dagger$ and θ^* (the sRMSE is 0).

Results of Experiment 1 are displayed in Figure 4. We observe that PPI has a lower sRMSE than $\theta_\dagger$ for all data points. This is expected and guaranteed by theory under assumptions. DSL exhibits a significantly lower sRMSE than both PPI and $\theta_\dagger$ for almost all data points, showing that it is able to use the expert annotations more efficiently than both. However, the crossing at the end, when virtually all expert annotations contribute to the debiasing procedure, is curious: why do the DSL and $\theta_\dagger$ curves cross? When analyzing the performance of DSL by dataset (see Appendix F), we notice that the crossing phenomenon in the performance of DSL seems dataset-dependent. In particular, for the Misinfo-general dataset, DSL performs worse than both PPI and $\theta_\dagger$ for all samples.

The exact reasons for this phenomenon are currently unknown. We have ruled out hypotheses related to poor convergence in optimization or preprocessing (e.g., centering); we have not identified obvious properties of the dataset that predict anomalous DSL estimates (e.g., agreement between expert and LLM annotations). One remaining explanation is that, although PPI debiasing via subgradients leverages less information compared to DSL (which uses external sampling design knowledge), it avoids instabilities commonly associated with weighting estimators (Zubizarreta, 2015). Future work may explore these and related explanations.

The results of Experiment 2 are displayed in Figure 3. Since $\theta_\dagger$ does not use generated annotations, its sRMSE remains constant as the dataset size grows. We also observe that, in each of the three cases, PPI and DSL both outperform $\theta_\dagger$; performance of both tends to improve as we increase the total dataset size.

5 Conclusion

This study has investigated the performance of two LLM debiasing methods. On average, both debiasing methods produce models closer to a reference model than just using a small number of expert annotations. We also observe that DSL seems to significantly outperform PPI across datasets and annotation methods. However, DSL performance appears more inconsistent and dataset-dependent.

Limitations

Our study focuses on two specific debiasing methods, DSL and PPI, leaving out several other emerging techniques such as the recently proposed predict-then-debias (e.g., Kluger et al., 2025). We only consider scenarios where the outcome variable requires annotation, restricting the scope to single-task classification. In addition, we limit our downstream analysis to logistic regression, which does not capture more complex relationships or generalize to other statistical estimators such as survival or hierarchical models.

Moreover, our experiments also concentrate on four datasets with relatively short texts in English or German, so further evaluation is needed in other languages, domains, and text lengths. Lastly, we assume expert-labeled data to be the ground truth; in practice, human annotations can also be noisy or unreliable. Future work should examine how to extend or adapt these methods when the expert labels themselves may be subject to significant measurement error or domain shifts.

References

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J Hewett, Mojan Javaheripi, Piero Kauffmann, and 1 others. 2024. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*.

Anastasios N. Angelopoulos, Stephen Bates, Clara Fannjiang, Michael I. Jordan, and Tijana Zrnic. 2023. Prediction-powered inference. *Science*, 382(6671):669–674.

Anthropic. 2025. Claude 3.7 sonnet system card. `https://assets.anthropic.com/m/785e231869ea8b3b/original/claude-3-7-sonnet-system-card.pdf`.

Lisa P. Argyle, Ethan C. Busby, Nancy Fulda, Joshua R. Gubler, Christopher Rytting, and David Wingate. 2023. Out of one, many: Using language models to simulate human samples. *Political Analysis*, 31(3):337–351.

Christopher A. Bail. 2024. Can generative AI improve social science? *Proceedings of the National Academy of Sciences*, 121(21).

John Blitzer, Mark Dredze, and Fernando Pereira. 2007. Biographies, Bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 440–447, Prague, Czech Republic. Association for Computational Linguistics.

Maria De-Arteaga, Alexey Romanov, Hanna Wallach, Jennifer Chayes, Christian Borgs, Alexandra Chouldechova, Sahin Geyik, Krishnaram Kenthapadi, and Adam Tauman Kalai. 2019. Bias in bios: A case study of semantic representation bias in a high-stakes setting. In *proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 120–128.

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, United States. Association for Computational Linguistics.

Naoki Egami, Musashi Hinck, Brandon Stewart, and Hanying Wei. 2023. Using imperfect surrogates for downstream inference: Design-based supervised learning for social science applications of large language models. In *Advances in Neural Information Processing Systems*, volume 36, pages 68589–68601. Curran Associates, Inc.

Naoki Egami, Musashi Hinck, Brandon M. Stewart, and Hanying Wei. 2024. Using large language model annotations for the social sciences: A general framework of using predicted variables in downstream analyses. *Preprint from November 17, 2024*.

Mahmoud El-Haj, Paul Rayson, Steve Young, Andrew Moore, Martin Walker, Thomas Schleicher, and Vasiliki Athanasakou. 2016. Learning tone and attribution for financial text mining. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC‘16)*, pages 1820–1825, Portorož, Slovenia. European Language Resources Association (ELRA).

Fabrizio Gilardi, Meysam Alizadeh, and Maël Kubli. 2023. ChatGPT outperforms crowd workers for text-annotation tasks. *Proceedings of the National Academy of Sciences*, 120(30):e2305016120.

Shiho Kino, Yu-Tien Hsu, Koichiro Shiba, Yung-Shin Chien, Carol Mita, Ichiro Kawachi, and Adel Daoud. 2021. A scoping review on the use of machine learning in research on social determinants of health: Trends and research prospects. *SSM-population Health*, 15:100836.

Dan M Kluger, Kerri Lu, Tijana Zrnic, Sherrie Wang, and Stephen Bates. 2025. Prediction-powered inference with imputed covariates and nonuniform sampling. *arXiv preprint arXiv:2501.18577*.

Hao Lin and Yongjun Zhang. 2025. The risks of using large language models for text annotation in social science research. *arXiv preprint arXiv:2503.22040*.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, and 1 others. 2024. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Daniel A McFarland and H Richard McFarland. 2015. Big data and the danger of being precisely inaccurate. *Big Data & Society*, 2(2):2053951715602495.

Priyanshu Priya, Mauajama Firdaus, and Asif Ekbal. 2024. Computational politeness in natural language processing: A survey. *ACM Computing Surveys*, 56(9):1–42.

Yanchuan Sim, Brice D. L. Acree, Justin H. Gross, and Noah A. Smith. 2013. Measuring ideological proportions in political speeches. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 91–101, Seattle, Washington, USA. Association for Computational Linguistics.

Petter Törnberg. 2024. Best practices for text annotation with large language models. *Sociologica*, 18(2):67–85.

Ivo Verhoeven, Pushkar Mishra, and Ekaterina Shutova. 2024. Yesterday's news: Benchmarking multidimensional out-of-distribution generalisation of misinformation detection models. *arXiv preprint arXiv:2410.18122*.

Michael Wiegand, Melanie Siegel, and Josef Ruppenhofer. 2018. Overview of the GermEval 2018 shared task on the identification of offensive language. In *Proceedings of GermEval 2018, 14th Conference on Natural Language Processing (KONVENS 2018)*, Vienna, Austria.

Caleb Ziems, William Held, Omar Shaikh, Jiaao Chen, Zhehao Zhang, and Diyi Yang. 2024. Can large language models transform computational social science? *Computational Linguistics*, 50(1):237–291.

José R Zubizarreta. 2015. Stable weights that balance covariates for estimation with incomplete outcome data. *Journal of the American Statistical Association*, 110(511):910–922.

A Datasets Description

We here present information about the datasets used in the analysis. All datasets are publicly available. The annotations and tabular data from the experiments will be made available on Github after review.

Multi-domain Sentiment: The Multi-domain Sentiment dataset is a corpus of product reviews taken from Amazon (Blitzer et al., 2007). The dataset was originally used to investigate domain adaptation in sentiment classifiers. We used a subset taken from 6 domains, consisting of 11 914 reviews with two sets of annotations: a binary sentiment label (positive or negative) and a domain label (books, camera, DVD, health, music, or software). The dataset is balanced both in sentiment and topic labels.

For the substantive task, we use the sentiment label as the outcome variable. The independent variables are: the domain label (transformed to numeric values 0-5), the number of characters in the review, the number of space-separated words in the review, and the number of repetitions of the word "I" in the review.

Misinfo-general: The Misinfo-general dataset is a large corpus of British newspaper articles (Verhoeven et al., 2024) originally used to benchmark out-of-distribution performance of misinformation models. For our experiments, we selected articles from 2022 that were published in one of two venues: The Guardian UK or The Sun. We then balanced the dataset to have 5000 articles in each class.

For the substantive task, we use the venue as the binary outcome variable. The independent variables are: the number of characters in the article, the number of space-separated words in the article, the number of capital letters in the article, and the number of characters in the title of the article.

Bias in Biographies: The Bias in Biographies dataset is a corpus of short biographies originally used to study gender bias in occupational classification (De-Arteaga et al., 2019). The corpus consists of English-language online biographies from the Common Crawl, annotated with self-identified binary gender and occupation labels (with 28 categories), enabling analysis of implicit gender biases in textual representations. Here, $N = 10000$.

For the substantive task, we use the gender label as the outcome variable. This variable is balanced. Independent variables are: the occupation label (transformed to numeric value, 0-27), the number of characters in the biography, the number of space-separated words in the biography, and the number of capital letters in the biography.

`Germeval18`: The Germeval18 dataset is a corpus of German tweets. It was used in the GermEval shared task on the identification of offensive language in 2018 (Wiegand et al., 2018). It is composed of a training and test set of documents with an associated toxicity label, for a total of 5676 documents. We use a balanced subset of the data.

For the substantive task, we use the binary toxicity label as the outcome variable. The independent variables are: the number of characters in the tweet, the number of space-separated words in the tweet, the number of capital letters in the tweet, and the number of "@" characters in the tweet.

B Details of Evaluation Metrics

We define:

$$\mathrm{RMSE}(\theta; d) = \sqrt{\mathbb{E}\left[\left(\frac{\theta - \theta_d^*}{\theta_d^*}\right)^2\right]}.$$

where θ are the coefficients from the model under test and θ_d^* are the coefficients from the reference model for dataset d.

C Model Details

BERT + logistic regression. As a representative of supervised approaches, we fine-tune a pretrained BERT encoder (Devlin et al., 2019) on the expert-labeled subset to obtain contextual representations $\mathbf{h}_i = \mathrm{BERT}(d_i)$, which are then passed to a logistic regression head trained to predict y_i.

Large Language Models. We also generate annotations with three language models: Microsoft Phi-4 (Abdin et al., 2024), DeepSeek v3 (Liu et al., 2024), and Claude 3.7 Sonnet (Anthropic, 2025). Phi-4 is a 14B open-weights model which we ran locally. We used the paid DeepSeek and Anthropic APIs to access DeepSeek v3 and Claude 3.7 Sonnet, respectively. We paid approximately 10$ for the DeepSeek API and approximately $100 for the Anthropic API. The prompts used to generate the labels are available in Appendix E. In some cases, the annotations generated for a small number of the documents did not fit the annotation schema. These samples were ignored.

D Package and Code Details

For the classical logistic regression we use the `scikit-learn` Python package. We use no regularization and set the maximum iterations to 1000.

For DSL, we use the `dsl` R package from the original paper authors for both experiments. We leave the parameters to their default settings.

For PPI, we use the `ppi_py` Python package from the original paper authors for both experiments. We also leave the parameters to their default settings.

The source code for the experiments will be made available on Github after review.

E Prompts

Figures 5, 6, 7, and 8 show the prompt templates used to make prompts for LLM annotation. The prompt templates were specialized for each dataset since each dataset corresponds to a different annotation task. However, the structure of the prompt templates was kept the same: first, a short description of the task, then an explanation of the formatting with two simple examples, and finally the document to classify. For each dataset, we also include a system prompt (see Table 1).

F Results by Dataset

Figure 9 showcases the results for Experiment 1 broken down by dataset.

```
Classify the following review as either:
- POSITIVE if the review indicates an overall positive sentiment
- NEGATIVE if the review indicates an overall negative sentiment

Give no other explanation for your classification, only output the label.

Here are two examples of the formatting I would like you to use, where
< REVIEW_TEXT > is a stand-in for the article text:

< REVIEW_TEXT >

CLASSIFICATION: POSITIVE

< REVIEW_TEXT >

CLASSIFICATION: NEGATIVE

Here's the review to classify:

{text}

CLASSIFICATION:
```

Figure 5: The prompt template used to annotate documents from the Multi-domain Sentiment dataset, where `{text}` is substituted with the document in question.

```
Classify the following article as either:
- THESUN if it is likely to have been published in the British tabloid newspaper
  The Sun
- THEGUARDIAN if it is likely to have been published in the British daily
  newspaper The Guardian

Give no other explanation for your classification, only output the label.

Here are two examples of the formatting I would like you use, where
< ARTICLE_TEXT > is a stand-in for the article text:

< ARTICLE_TEXT >

CLASSIFICATION: THESUN

< ARTICLE_TEXT >

CLASSIFICATION: THEGUARDIAN

Here's the article I would like you to classify:

{text}

CLASSIFICATION:
```

Figure 6: The prompt template used to annotate documents from the Misinfo-general dataset, where `{text}` is substituted for the document in question

```
Classify the following textual biographies as either:
- MALE if the subject is likely to be male
- FEMALE if the subject is likely to be female

Give no other explanation for your classification, only output the label.

Here are two examples of the formatting I would like you use, where < BIOGRAPHY_TEXT >
is a stand-in for the textual biography:

< BIOGRAPHY_TEXT >

CLASSIFICATION: MALE

< BIOGRAPHY_TEXT >

CLASSIFICATION: FEMALE

Here's the textual biography I would like you to classify:

{text}

CLASSIFICATION:
```

Figure 7: The prompt template used to annotate documents from the Bias in Biographies dataset, where `{text}` is substituted for the document in question

```
Classify the following German tweets as either:
- OFFENSIVE if the tweet is likely to contain an offense or be offensive
- OTHER if the tweet is _not_ likely to contain an offense or be offensive

Give no other explanation for your classification, only output the label.

Here are two examples of the formatting I would like you use, where < TWEET_TEXT >
is a stand-in for the text of the tweet:

< TWEET_TEXT >

CLASSIFICATION: OFFENSIVE

< TWEET_TEXT >

CLASSIFICATION: OTHER

{make_examples(examples)}

Here's the German tweet I would like you to classify:

{text}

CLASSIFICATION:
```

Figure 8: The prompt template used to annotate documents from the Germeval18 dataset, where `{text}` is substituted for the document in question

Dataset	System Prompt
Multi-domain Sentiment	“You are a perfect sentiment classification system”
Misinfo-general	“You are a perfect newspaper article classification system”
Bias in Biographies	“You are a perfect biography classification system”
Germeval18	“You are a perfect German tweet classification system”

Table 1: The system prompts used to annotate the various datasets

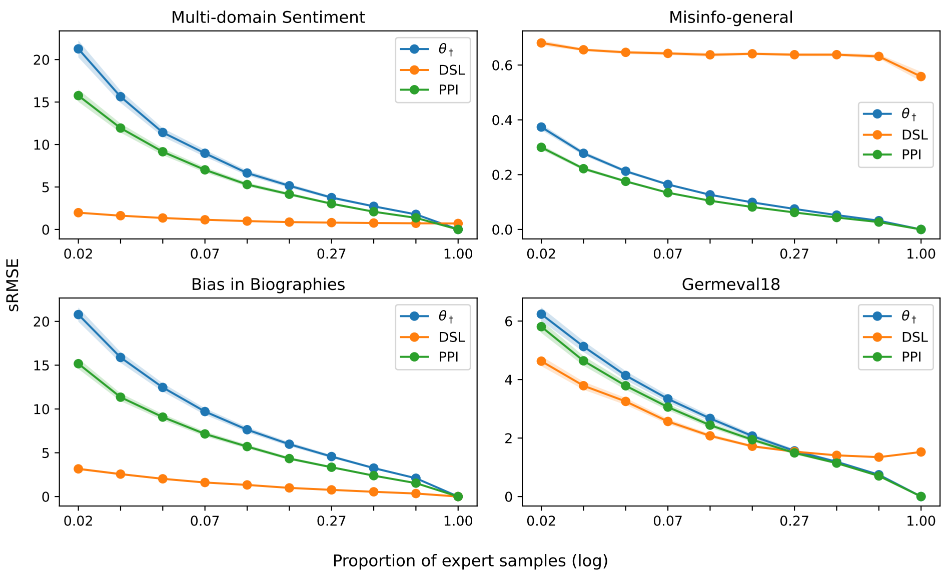

Figure 9: Results from the set of experiments varying the proportion of expert samples, aggregated per dataset.