Online Supplementary Appendix:

Using Large Language Model Annotations for the Social Sciences:
A General Framework of Using Predicted Variables
in Downstream Analyses

Contents

A Connection to Literature

This paper builds on several lines of work. First, this paper is motivated by the rapid and fundamental development of LLMs and, more generally, generative artificial intelligence. Over the last couple of years, many papers have shown the incredible potential of LLMs for social science research in a wide range of problems (e.g., Argyle et al., 2023; Linegar, Kocielnik and Alvarez, 2023; Palmer and Spirling, 2023; Wu et al., 2023). Among them, one of the most promising and popular use cases is text annotations by LLMs: to name a few papers, Bommasani et al. (2021); Ornstein, Blasingame and Truscott (2022); Gilardi, Alizadeh and Kubli (2023); Ollion et al. (2023); Pangakis, Wolken and Fasching (2023); Ziems et al. (2023), and this list is growing rapidly. Each paper discusses and evaluates the promise and risks of using LLM annotations in different types of social science applications. All of these papers currently only focus on assessing predictive performance, and no paper discusses how such predicted text labels can be properly used in downstream statistical analyses, which is the central focus of our paper.

Methodologically, this paper draws upon the large literature on double/debiased machine learning and doubly-robust estimation for missing data and causal inference (Robins, Rotnitzky and Zhao, 1994; Chernozhukov et al., 2018; Kennedy, 2022). In particular, our doubly robust procedure builds on foundational results on semiparametric inference with missing data (Robins and Rotnitzky, 1995; Tsiatis, 2006; Rotnitzky and Vansteelandt, 2014; Davidian, 2022) and the growing literature on doubly robust estimators for surrogate outcomes (Kallus and Mao, 2020) and semi-supervised learning (Chakrabortty and Cai, 2018; Chakrabortty, Dai and Tchetgen Tchetgen, 2022). Like these papers, we exploit the influence function to derive debiased estimators.

Our paper contributes to the growing literature on the use of predicted variables in statistical analyses. A number of papers develop methods for specific scenarios by making assumptions about the underlying data generating process. For example, Wang, McCormick and Leek (2020) take into account the predicted outcome by modeling prediction errors, Fong and Tyler (2021) address the predicted independent variables under exclusion restriction, Zhang (2021) relies on a conditional independence assumption about prediction errors, and Knox, Lucas and Cho (2022) use signed causal diagrams to compute bounds. In contrast to these papers, we only assume that researchers control the sampling process for expert annotations, and we do not make any assumption about the nature of prediction errors, which is particularly difficult to justify in applications of LLMs.

Our paper is most closely related to recent methods that build on the doubly robust estimation to deal with predicted variables. In particular, our paper extends and generalizes methods proposed in Egami et al. (2023). In particular, they only cover cases where the outcome variable requires text annotation and only discuss several models for downstream analyses. By deriving a more general result, we cover cases where any subset of the outcome and independent variables are text-based and accommodate a much wider range of downstream analyses. This methodological generalization is fundamental because about 45% of applications use text-based variables as independent variables, which is not covered in Egami et al. (2023). In addition, we

make practical contributions by providing detailed guides on LLM annotations (e.g., how to use LLM annotations in DSL) and expert annotations (e.g., how to determine the required number of expert annotations, and how to handle errors in expert annotations) using two empirical applications.

Theoretically, our paper is also closely related to two recent papers that similarly build on the literature on doubly robust methods. Prediction-powered inference (Angelopoulos et al., 2023) provides a similar framework to ours, but they have primarily focused on settings where the outcome variable is predicted while providing both asymptotic and non-asymptotic confidence intervals. Mozer and Miratrix (2023) focus on settings where the predicted outcome variable is used within randomized experiments. In contrast, our paper covers cases where any subset of the outcome and independent variables are predicted in regression analyses or in randomized experiments, while focusing on asymptotic confidence intervals. Importantly, even though all of these papers have similar methodological motivations and share the methodological foundation of doubly robust methods, they target different application areas and are complementary to each other. Katsumata and Yamauchi (2023) also develop a framework for using predicted variables while building on a different framework of control variates (Chen and Chen, 2000).

B Theoretical Properties of DSL

B.1 Notation and Assumption

Suppose researchers are interested in analyzing N documents. For each document i, we define D_i to be a vector of relevant variables we include in the downstream analyses. For regression problems, $D = (Y, X)$ where Y is the outcome variable and X is a vector of the independent variables. For the mean estimation problem, $D = Y$. For observational causal inference under conditional ignorability, $D = (Y, T, X)$ where Y is the outcome variable, T is the treatment, and X is a vector of observed covariates. For the instrumental variable method, $D = (Y, T, Z, X)$ where Y is the outcome of interest, T is the treatment, Z is the instrument, and X is a vector of observed covariates.

In text-as-data applications, we often cannot observe all relevant variables D for the entire population of documents. We decompose D into two parts $D = (D^{obs}, D^{mis})$ where D^{obs} represents variables that are observed for the entire population of documents and D^{mis} represents variables that are observed only for a subset of documents that are expert-coded. For example, when the outcome variable Y requires text annotation but $X = (X_1, \ldots, X_4)$ are observed for every document, $D^{mis} = Y$ and $D^{obs} = (X_1, X_2, X_3, X_4)$. When Y and X_1 are text-based but the remaining independent variables are observed for every document, $D^{mis} = (Y, X_1)$ and $D^{obs} = (X_2, X_3, X_4)$. This general setup allows for settings where any subset of relevant variables is text-based.

We use Q_i to denote a vector of optional document-level variables that help predict D_i^{mis}. When researchers use LLM annotations as the automated text annotation for D_i^{mis}, those LLM annotations are included in Q_i. When researchers use the classical supervised machine learning method to predict D_i^{mis}, a vector of word frequencies or word embedding is included in Q_i.

Finally, we use $R_i \in \{0, 1\}$ to denote whether document i is sampled for expert annotations. For documents with $R_i = 1$, we observe values for D_i^{mis}, but for documents with $R_i = 0$, values for D_i^{mis} are missing.

The key assumption behind DSL (Assumption 1) is that the probability of sampling documents for expert-coding π_i is decided by researchers, and π_i is larger than zero for every document. Without loss of generality, we can write $\pi_i = \pi(D_i^{obs}, Q_i)$. This notation only assumes that R_i depends on a subset of (D_i^{obs}, Q_i), so it also accommodates more common settings like random sampling where π_i does not depend on any variable or stratified sampling where π_i only depends on a small number of observed variables. Under this design-based sampling, the sampling probability π is known from the research design (i.e., not need to estimate the sampling probability), and we have

$$R_i \perp\!\!\!\perp D_i^{mis} \mid D_i^{obs}, Q_i. \tag{OA.1}$$

B.2 General Results

We first provide proof for a general DSL estimator based on convex objective functions.

Suppose researchers are interested in estimands that can be characterized as the solution to the following convex optimization problem.

$$\beta^* \;\coloneqq\; \underset{\beta \in \mathbb{R}^d}{\operatorname{argmin}}\, \mathbb{E}\left\{\ell(D; \beta)\right\} \tag{OA.2}$$

where $\ell(D; \beta)$ is the convex loss function and D represents all the relevant variables in the downstream statistical analyses. This general setup incorporates a wide range of common regression models (see Appendix B.3), such as linear regression with a continuous outcome, logistic regression with a binary outcome, multinomial logistic regression with a categorical outcome, and Poisson regression with a count outcome, as well as linear fixed-effects regression popular in causal inference.

Under mild regularity conditions, convexity allows us to express β^* as the solution to the following estimation equation.

$$\mathbb{E}\left\{m(D; \beta)\right\} \;=\; 0 \tag{OA.3}$$

where $m(D; \beta) \in \mathbb{R}^d$ is a subgradient of the loss function $\ell(D; \beta)$ with respect to β.

If researchers can observe all relevant variables D, they can directly solve the estimation equation to obtain a consistent and asymptotically normal estimator.

$$\widehat{\beta}_{\text{oracle}} \coloneqq \underset{\beta \in \mathbb{R}^d}{\operatorname{argmin}} \left\| \frac{1}{N} \sum_{i=1}^{N} m(D_i; \beta) \right\|_2^2. \tag{OA.4}$$

However, when some relevant variables are not observed for the entire population of interest, this estimator is infeasible.

DSL can estimate β even when some variables D_i^{mis} are observed only for a subset of expert-coded documents. In general, the moment function for DSL is defined as

$$m_{\text{DSL}}(D_i, Q_i, R_i; \beta, \pi, \widehat{g}) \coloneqq m(D_i^{obs}, \widehat{D}_i^{mis}; \beta) - \frac{R_i}{\pi(D_i^{obs}, Q_i)} \left(m(D_i^{obs}, \widehat{D}_i^{mis}; \beta) - m(D_i^{obs}, D_i^{mis}; \beta) \right) \tag{OA.5}$$

where $\widehat{D}_i^{mis} = \widehat{g}(D_i^{obs}, Q_i)$ and $\widehat{g}(\cdot)$ is a estimated supervised machine learning model to predict D_i^{mis} with covariates (D_i^{obs}, Q_i). Using this moment function, the proposed DSL estimator is defined as,

$$\widehat{\beta}_{\text{DSL}} \coloneqq \underset{\beta \in \mathbb{R}^d}{\text{argmin}} \left\| \frac{1}{N} \sum_{k=1}^{K} \sum_{i \in \mathcal{L}_k} m_{\text{DSL}}(D_i, Q_i, R_i; \beta, \pi, \widehat{g}_k) \right\|_2^2, \tag{OA.6}$$

where we employ a K-fold cross-fitting procedure (Chernozhukov et al., 2018). We first partition the observation indices $i = 1, \ldots, n$ into K groups $\mathcal{L}_k$ where $k = 1, \ldots, K$. We then learn the supervised machine learning model $\widehat{g}_k$ by predicting D_i^{mis} using (D_i^{obs}, Q_i) using expert-coded documents *not* in $\mathcal{L}_k$.

Proposition 1 *Under Assumption 1 and the standard regularity conditions stated below, the cross-fitted DSL estimator (equation* (OA.6)*)* $\widehat{\beta}_{DSL}$ *is consistent and asymptotically normal as sample size* N *goes to infinity.*

$$\sqrt{N}(\widehat{\beta}_{\text{DSL}} - \beta^*) \xrightarrow{d} \mathcal{N}(0, V). \tag{OA.7}$$

where

$$\begin{aligned} V &= S_V \mathbb{E}(m_{\text{DSL}}(D_i, Q_i, R_i; \beta^*, \pi, \overline{g}) m_{\text{DSL}}(D_i, Q_i, R_i; \beta^*, \pi, \overline{g})^\top) S_V, \\ S_V &= \mathbb{E}\left(\frac{\partial m_{\text{DSL}}(D_i, Q_i, R_i; \beta^*, \pi, \overline{g})}{\partial \beta} \right)^{-1} \end{aligned}$$

Here we define $\overline{g}$ *to be the probability limit of the estimated supervised machine learning function* $\widehat{g}_k$ *in the sense that for each* k, $||\widehat{g}_k - \overline{g}||_2 = o_p(1)$ *and* $\mathbb{E}_k(||m(L; \beta^*, \widehat{g}_k) - m(L; \beta^*, \overline{g})||_2^2) = o_p(1)$. *This probability limit does not need to be equal to the true conditional expectation* g^*. *Thus, we do not assume the correct specification of the estimated supervised machine learning function.*

Proof. In this proof, for the notational simplicity, we use $L_i = (D_i, Q_i, R_i)$ and omit π from the notation of the moment function. That is, we use $m_{\text{DSL}}(L_i; \beta, g)$ to denote the DSL moment function. We also use $\widehat{\beta}$ to denote $\widehat{\beta}_{\text{DSL}}$.

Using the mean value theorem, we can first expand the moment equation around β^*.

$$\frac{1}{N} \sum_{k=1}^{K} \sum_{i \in \mathcal{L}_k} m_{\text{DSL}}(L_i; \widehat{\beta}, \widehat{g}_k) = \frac{1}{N} \sum_{k=1}^{K} \sum_{i \in \mathcal{L}_k} m_{\text{DSL}}(L_i; \beta^*, \widehat{g}_k) + (\widehat{\beta} - \beta^*) \frac{1}{N} \sum_{k=1}^{K} \sum_{i \in \mathcal{L}_k} \frac{\partial m_{\text{DSL}}(L_i; \widetilde{\beta}, \widehat{g}_k)}{\partial \beta}$$

where $\widetilde{\beta}$ is a mean value, located between $\widehat{\beta}$ and β^*. For the convex objective function, the first order condition implies that we also have

$$\frac{1}{N} \sum_{k=1}^{K} \sum_{i \in \mathcal{L}_k} m_{\text{DSL}}(L_i; \widehat{\beta}, \widehat{g}_k) = 0.$$

Therefore, combining two equations, we have

$$\sqrt{N}(\widehat{\beta} - \beta^*) = \underbrace{\left(-\frac{1}{N} \sum_{k=1}^{K} \sum_{i \in \mathcal{L}_k} \frac{\partial m_{\text{DSL}}(L_i; \widetilde{\beta}, \widehat{g}_k)}{\partial \beta} \right)^{-1}}_{(a)} \times \underbrace{\frac{1}{\sqrt{N}} \sum_{k=1}^{K} \sum_{i \in \mathcal{L}_k} m_{\text{DSL}}(L_i; \beta^*, \widehat{g}_k)}_{(b)}$$

We will consider terms (a) and (b) in order.

Term (b). We begin with the main term (b), which can be decomposed into three terms.

$$\frac{1}{\sqrt{N}}\sum_{k=1}^{K}\sum_{i\in\mathcal{L}_k} m_{\mathrm{DSL}}(L_i;\beta^*,\widehat{g}_k) \quad = \quad H_1 + H_2 + H_3$$

where

$$\begin{aligned}
H_1 &\coloneqq \frac{1}{\sqrt{N}}\sum_{k=1}^{K}\sum_{i\in\mathcal{L}_k}(m_{\mathrm{DSL}}(L_i;\beta^*,\widehat{g}_k) - \mathbb{E}_k(m_{\mathrm{DSL}}(L_i;\beta^*,\widehat{g}_k))) - (m_{\mathrm{DSL}}(L_i;\beta^*,\overline{g}) - \mathbb{E}_k(m_{\mathrm{DSL}}(L_i;\beta^*,\overline{g})))\\
H_2 &\coloneqq \frac{1}{\sqrt{N}}\sum_{k=1}^{K}\sum_{i\in\mathcal{L}_k}(m_{\mathrm{DSL}}(L_i;\beta^*,\overline{g}) - \mathbb{E}_k(m_{\mathrm{DSL}}(L_i;\beta^*,\overline{g})))\\
H_3 &\coloneqq \frac{1}{\sqrt{N}}\sum_{k=1}^{K} N_k \times \mathbb{E}_k(m_{\mathrm{DSL}}(L_i;\beta^*,\widehat{g}_k)).
\end{aligned}$$

Here we use $\mathbb{E}_k$ to denote the expectation over $\mathcal{L}_k$ conditional on $\mathcal{L}_{-k}$.

H_1 is known as the empirical process term. Given that we use cross-fitting and $\mathbb{E}_k(||m_{\mathrm{DSL}}(L_i;\beta^*,\widehat{g}_k) - m_{\mathrm{DSL}}(L_i;\beta^*,\overline{g})||_2^2) = o_p(1)$, we obtain $H_1 = o_p(1)$ by Lemma 2 of Kennedy, Balakrishnan and G'Sell (2020).

Next, to examine H_2, we first show that $\mathbb{E}(m_{\mathrm{DSL}}(L_i;\beta^*,\widetilde{g})) = 0$ for any arbitrary fixed function $\widetilde{g}$.

$$\begin{aligned}
&\mathbb{E}(m_{\mathrm{DSL}}(L_i;\beta^*,\widetilde{g}))\\
=\ &\mathbb{E}\left(m(D_i^{obs},\widetilde{g}(D_i^{obs},Q_i);\beta^*) - \frac{R_i}{\pi(D_i^{obs},Q_i)}\left(m(D_i^{obs},\widetilde{g}(D_i^{obs},Q_i);\beta^*) - m(D_i^{obs},D_i^{mis};\beta^*)\right)\right)\\
=\ &\mathbb{E}\left(\left(\frac{R_i}{\pi(D_i^{obs},Q_i)}m(D_i^{obs},D_i^{mis};\beta^*)\middle| D_i^{obs},Q_i\right)\right)\\
&+\mathbb{E}\left(\left(\left(1-\frac{R_i}{\pi(D_i^{obs},Q_i)}\right)m(D_i^{obs},\widetilde{g}(D_i^{obs},Q_i);\beta^*)\middle| D_i^{obs},Q_i\right)\right)\\
=\ &\mathbb{E}\left(\frac{\mathbb{E}(R_i \mid D_i^{obs},Q_i)}{\pi(D_i^{obs},Q_i)}\mathbb{E}\left(m(D_i^{obs},D_i^{mis};\beta^*) \mid D_i^{obs},Q_i\right)\right)\\
&+\mathbb{E}\left(\left(1-\frac{\mathbb{E}(R_i \mid D_i^{obs},Q_i)}{\pi(D_i^{obs},Q_i)}\right)m(D_i^{obs},\widetilde{g}(D_i^{obs},Q_i);\beta^*)\right)\\
=\ &\mathbb{E}\left(m(D_i^{obs},D_i^{mis};\beta^*)\right)\\
=\ &0.
\end{aligned}$$

where the first equality comes from the definition of the DSL moment function, and the second equality comes from the rearrangement of the terms and the law of total expectation. The third equality comes from Assumption 1, i.e., $R_i \perp\!\!\!\perp D_i^{mis} \mid D_i^{obs},Q_i$, which implies $\mathbb{E}(R_i m(D_i^{obs},D_i^{mis};\beta^*) \mid D_i^{obs},Q_i) = \mathbb{E}(R_i \mid D_i^{obs},Q_i)\mathbb{E}(m(D_i^{obs},D_i^{mis};\beta^*) \mid D_i^{obs},Q_i)$. The fourth equality comes from the equality that $\mathbb{E}(R_i \mid D_i^{obs},Q_i) = \Pr(R_i = 1 \mid D_i^{obs},Q_i) =$

$\pi(D_i^{obs}, Q_i)$ because R_i is a binary variable. Finally, due to convexity of the objective function, $\mathbb{E}(m(D_i, \beta^*)) = 0$.

We also have

$$\begin{aligned} H_2 &\coloneqq \frac{1}{\sqrt{N}} \sum_{k=1}^{K} \sum_{i \in \mathcal{L}_k} (m_{\text{DSL}}(L_i; \beta^*, \overline{g}) - \mathbb{E}_k(m_{\text{DSL}}(L_i; \beta^*, \overline{g}))) \\ &= \frac{1}{\sqrt{N}} \sum_{k=1}^{K} \sum_{i \in \mathcal{L}_k} (m_{\text{DSL}}(L_i; \beta^*, \overline{g}) - \mathbb{E}(m_{\text{DSL}}(L_i; \beta^*, \overline{g}))) \\ &= \frac{1}{\sqrt{N}} \sum_{i=1}^{N} (m_{\text{DSL}}(L_i; \beta^*, \overline{g}) - \mathbb{E}(m_{\text{DSL}}(L_i; \beta^*, \overline{g}))) \end{aligned}$$

because $\mathbb{E}_k(m_{\text{DSL}}(L_i; \beta^*, \overline{g})) = \mathbb{E}(m_{\text{DSL}}(L_i; \beta^*, \overline{g}))$. Finally, we can use the central limit theorem to show that

$$H_2 \xrightarrow{d} \mathcal{N}(0, \mathbb{E}(m_{\text{DSL}}(L_i; \beta^*, \overline{g}) m_{\text{DSL}}(L_i; \beta^*, \overline{g})^\top)) \tag{OA.8}$$

where $\text{Var}(m_{\text{DSL}}(L_i; \beta^*, \overline{g})) = \mathbb{E}(m_{\text{DSL}}(L_i; \beta^*, \overline{g}) m_{\text{DSL}}(L_i; \beta^*, \overline{g})^\top)$ as $\mathbb{E}(m_{\text{DSL}}(L_i; \beta^*, \overline{g})) = 0$.

As for H_3, using the similar proof for $\mathbb{E}(m_{\text{DSL}}(L_i; \beta^*, \widetilde{g})) = 0$, we have $\mathbb{E}_k(m_{\text{DSL}}(L_i; \beta^*, \widehat{g}_k)) = 0$ because $\widehat{g}_k$ is a fixed function conditional on $\mathcal{L}_{-k}$. Therefore, $H_3 = 0$.

Taken together, for the main term (b), we obtain

$$\frac{1}{\sqrt{N}} \sum_{k=1}^{K} \sum_{i \in \mathcal{L}_k} m_{\text{DSL}}(L_i; \beta^*, \widehat{g}_k) \quad \xrightarrow{d} \quad \mathcal{N}(0, \mathbb{E}(m_{\text{DSL}}(L_i; \beta^*, \overline{g}) m_{\text{DSL}}(L_i; \beta^*, \overline{g})^\top)).$$

Term (a). We now consider the term (a), and we need to show that

$$\left(\frac{1}{N} \sum_{k=1}^{K} \sum_{i \in \mathcal{L}_k} \frac{\partial m_{\text{DSL}}(L_i; \widetilde{\beta}, \widehat{g}_k)}{\partial \beta} \right)^{-1} \xrightarrow{p} \mathbb{E} \left(\frac{\partial m_{\text{DSL}}(L_i; \beta^*, \overline{g})}{\partial \beta} \right)^{-1} \tag{OA.9}$$

We require the standard regularity conditions that assume the smoothness of the derivative of the moment, which holds true for the most common method of moment estimators we consider here.

Assumption 5 from Chernozhukov et al. (2022). *$\mathbb{E}\left(\partial m_{DSL}(L_i; \beta^*, \overline{g})/\partial \beta\right)$ exists and there is a neighborhood $\mathcal{N}_\beta$ of β^* such that: (i) for each k, $||\widehat{g}_k - \overline{g}||_2 = o_p(1)$; (ii) for all $||g - \overline{g}||_2$ small enough, $m_{DSL}(L; \beta, g)$ is differentiable in β on $\mathcal{N}_\beta$ with probability approaching one, and there are $C > 0$ and $\delta(D; g)$ such that, for $\beta \in \mathcal{N}_\beta$ and $||g - \overline{g}||_2$ small enough,*

$$\left|\left| \frac{\partial m_{\text{DSL}}(L; \beta, g)}{\partial \beta} - \frac{\partial m_{\text{DSL}}(L; \beta^*, g)}{\partial \beta} \right|\right|_2 \leq \delta(L, g) ||\beta - \beta^*||_2^{1/C}; \quad \mathbb{E}(\delta(L, g)) < C.$$

(iii) For each k and p and q, $\mathbb{E}(\partial m_{DSL}(L; \beta^*, \widehat{g}_k)_p/\partial \beta_q - \partial m_{DSL}(L; \beta^*, \overline{g})_p/\partial \beta_q) = o_p(1)$.

These regularity conditions are standard (Newey and McFadden, 1994). Among this regularity condition, the main requirement is that, for each k, $||\widehat{g}_k - \overline{g}||_2 = o_p(1)$. However, we

define $\overline{g}$ to be the probability limit of $\widehat{g}_k$, and thus, this automatically holds. Therefore, under this assumption and $\widehat{\beta} - \beta^* = o_p(1)$, we obtain equation (OA.9).

Finally, we combining terms (a) and (b), we have

$$\sqrt{N}(\widehat{\beta}_{\text{DSL}} - \beta^*) \xrightarrow{d} \mathcal{N}(0, V). \tag{OA.10}$$

where

$$V = \mathbb{E}\left(\frac{\partial m_{\text{DSL}}(L_i; \beta^*, \overline{g})}{\partial \beta}\right)^{-1} \mathbb{E}(m_{\text{DSL}}(L_i; \beta^*, \overline{g}) m_{\text{DSL}}(L_i; \beta^*, \overline{g})^\top) \mathbb{E}\left(\frac{\partial m_{\text{DSL}}(L_i; \beta^*, \overline{g})}{\partial \beta}\right)^{-1},$$

which completes the proof. □

B.3 Examples

The general theoretical results developed in Appendix B.2 accommodate a wide range of regression problems. To derive a new DSL estimator, researchers just need to derive a corresponding moment function (i.e., a subgradient of a given convex objective function) for each estimator.

For linear regression, the moment function is defined as,

$$\frac{1}{N}\sum_{i=1}^{N} m(D_i; \beta) \coloneqq \frac{1}{N}\sum_{i=1}^{N}(Y_i - X_i^\top \beta) X_i. \tag{OA.11}$$

For logistic regression, the moment function is

$$\frac{1}{N}\sum_{i=1}^{N} m(D_i; \beta) \coloneqq \frac{1}{N}\sum_{i=1}^{N}(Y_i - \text{expit}(X_i^\top \beta)) X_i \tag{OA.12}$$

where $\text{expit}(\cdot)$ is the inverse of the logit function.

For linear fixed-effects regression, the moment function is

$$\frac{1}{NT}\sum_{i=1}^{N}\sum_{t=1}^{T} m(D_{it}; \beta, \alpha, \gamma) \coloneqq \begin{pmatrix} \frac{1}{NT}\sum_{i=1}^{N}\sum_{t=1}^{T}(Y_{it} - \alpha_i - \gamma_t - X_{it}^\top \beta) X_{it} \\ \left\{\frac{1}{NT}\sum_{t=1}^{T}(Y_{it} - \alpha_i - \gamma_t - X_{it}^\top \beta) X_{it}\right\}_{i=1}^{N} \\ \left\{\frac{1}{NT}\sum_{i=1}^{N}(Y_{it} - \alpha_i - \gamma_t - X_{it}^\top \beta) X_{it}\right\}_{t=1}^{T} \end{pmatrix}. \tag{OA.13}$$

For multinomial logistic regression, the moment function is

$$\frac{1}{N}\sum_{i=1}^{N} m(D_i; \{\beta\}_{k=1}^{J}) \coloneqq \left\{\frac{1}{N}\sum_{i=1}^{N}(Y_{ik} - \rho_{ik}) X_i\right\}_{k=1}^{J-1} \tag{OA.14}$$

where $Y_{ik} \coloneqq \mathbf{1}\{Y_i = k\}$,

$$\rho_{ik} \coloneqq \frac{\exp(X_i^\top \beta_k)}{1 + \sum_{k=1}^{J-1}\exp(X_i^\top \beta_k)}, \tag{OA.15}$$

and $\beta_J = 0$.

For Poisson regression, the moment function is

$$\frac{1}{N}\sum_{i=1}^{N} m(D_i; \beta) \coloneqq \frac{1}{N}\sum_{i=1}^{N}(Y_i - \exp(X_i^\top \beta)) X_i. \tag{OA.16}$$

B.4 Power Analysis

Increasing the number of expert annotations is equivalent to increasing the sampling probability π for each document.

From the general results we derived in Appendix B.2, we know the asymptotic variance of the DSL estimator takes the following form.

$$\begin{aligned} V &= S_V \mathbb{E}(m_{\text{DSL}}(D_i, Q_i, R_i; \beta^*, \pi, \overline{g}) m_{\text{DSL}}(D_i, Q_i, R_i; \beta^*, \pi, \overline{g})^\top) S_V, \\ S_V &= \mathbb{E}\left(\frac{\partial m_{\text{DSL}}(D_i, Q_i, R_i; \beta^*, \pi, \overline{g})}{\partial \beta}\right)^{-1} \end{aligned}$$

Based on the definition of the DSL moment, we also have

$$\mathbb{E}\left(\frac{\partial m_{\text{DSL}}(D_i, Q_i, R_i; \beta^*, \pi, \overline{g})}{\partial \beta}\right)^{-1} = \mathbb{E}\left(\frac{\partial m(D_i; \beta^*)}{\partial \beta}\right)^{-1}. \tag{OA.17}$$

Therefore, the asymptotic variance of the DSL estimator can be re-written as follows.

$$V = \mathbb{E}\left(\frac{\partial m(D; \beta^*)}{\partial \beta}\right)^{-1} \mathbb{E}(m_{\text{DSL}}(D_i, Q_i, R_i; \beta^*, \pi, \overline{g}) m_{\text{DSL}}(D_i, Q_i, R_i; \beta^*, \pi, \overline{g})^\top) \mathbb{E}\left(\frac{\partial m(D; \beta^*)}{\partial \beta}\right)^{-1}.$$

Importantly, the "sandwich" part of the variance $\mathbb{E}\left(\frac{\partial m(D; \beta^*)}{\partial \beta}\right)^{-1}$ only depends on the original moment function and is not dependent on the sampling probability π.

Therefore, increasing the number of expert annotations contributes only to the "meat" part of the variance. We can further decompose the "meat" part of the variance as follows.

$$\begin{aligned} & \mathbb{E}(m_{\text{DSL}}(D_i, Q_i, R_i; \beta^*, \pi, \overline{g}) m_{\text{DSL}}(D_i, Q_i, R_i; \beta^*, \pi, \overline{g})^\top) \\ = \; & \mathbb{E}\left(\frac{1}{\pi(D_i^{obs}, Q_i)} \left(m(D_i; \beta^*) - m(D_i^{obs}, \widehat{D}_i^{mis}; \beta^*)\right) \left(m(D_i; \beta^*) - m(D_i^{obs}, \widehat{D}_i^{mis}; \beta^*)\right)^\top\right) \\ & + \mathbb{E}\left(m(D_i^{obs}, \widehat{D}_i^{mis}; \beta^*) m(D_i^{obs}, \widehat{D}_i^{mis}; \beta^*)^\top\right) \\ & + \mathbb{E}\left(m(D_i^{obs}, \widehat{D}_i^{mis}; \beta^*) \left(m(D_i; \beta^*) - m(D_i^{obs}, \widehat{D}_i^{mis}; \beta^*)\right)^\top\right) \\ & + \mathbb{E}\left(\left(m(D_i; \beta^*) - m(D_i^{obs}, \widehat{D}_i^{mis}; \beta^*)\right) m(D_i^{obs}, \widehat{D}_i^{mis}; \beta^*)^\top\right) \end{aligned}$$

From this decomposition, we can predict the standard errors under different sampling probabilities by plugging-in different sampling probabilities into the first term. We can consistently estimate this variance under different sampling probabilities.

C Problem of Ignoring Prediction Errors

Ignoring prediction errors in the text annotation step, even if the errors are small, leads to bias, invalid confidence intervals, and wrong p-values in the subsequent statistical analyses of text-based variables. This is because prediction errors are *not completely random*—prediction errors

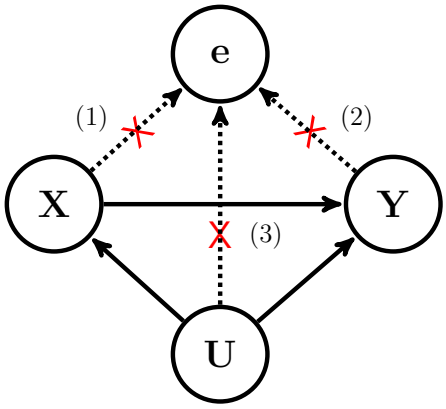


Figure OA-1: **Conditions Required to Ignore Prediction Errors.**
Note: Prediction errors e can be ignored only when e is uncorrelated with independent variables $\mathbf{X}$. This implies that (1) prediction errors e are not affected by independent variables $\mathbf{X}$, (2) prediction errors e are not affected by the outcome variable Y (to block the path $X \rightarrow Y \rightarrow e$), and (3) prediction errors e are not affected by unobserved confounders U (to block the path $X \leftarrow U \rightarrow e$).

are correlated with observed and unobserved variables we include in downstream analyses. Even small prediction errors can bias downstream analyses in any direction by any amount. Because exactly the same problem applies to the LLM-only estimation and the classical supervised learning estimation, we do not distinguish them and we discuss prediction errors in general.

To concretely illustrate the problem, we focus on a simple case where the outcome variable Y requires text annotation, and researchers want to regress Y on independent variables $\mathbf{X}$ to estimate coefficients β defined as,

$$\mathbb{E}(Y_i \mid \mathbf{X}_i) \;=\; \mathbf{X}_i^\top \beta. \tag{OA.18}$$

Researchers can easily obtain the ordinary squares estimates of β when the outcome variable of interest Y is observed for every document. However, when Y requires text annotation and Y itself is not observed for each document, researchers instead regress the predicted outcome variable $\widehat{Y}$ on independent variables $\mathbf{X}$. This linear regression with the predicted outcome variable will lead to unbiased coefficient estimation when prediction error, $e_i = \widehat{Y}_i - Y_i$, is zero on average across all different combinations of $\mathbf{X}$.

$$\mathbb{E}(e_i \mid \mathbf{X}_i) \;=\; 0. \tag{OA.19}$$

Even though this expression might seem similar to the standard exogeneity assumption, it turns out that this condition implies much stronger assumptions (see a diagram in Figure OA-1). First, prediction errors cannot be affected by any independent variable $\mathbf{X}$ included in downstream analyses. For example, in Fowler et al. (2021) where the original authors included candidate-fixed effects and a platform on which a given political ad is run as $\mathbf{X}$, this condition requires that prediction errors be the same across all candidates and platforms. This requirement

might be the most natural one—differential error rates across $\mathbf{X}$ lead to biased estimates of effects of $\mathbf{X}$. Second, prediction errors cannot be affected by the outcome of interest Y, either. For example, in Fowler et al. (2021) where Y is the tone of ads that has three categories (Attack, Contrast, Promote), this condition requires that prediction errors be the same across all categories. This condition is particularly unlikely to hold in most applications of text analyses in the social sciences because most prediction approaches (LLMs or supervised ML methods) tend to have higher prediction errors for rare categories.[1] Finally, prediction errors cannot be affected by unobserved confounders U, either. This is the case even when researchers only estimate coefficients β for descriptive analyses and do not make explicit causal claims. Unfortunately, it is not sufficient to check relationships between prediction errors and the main variables in the downstream analyses (the outcome and the independent variables). To ignore prediction errors, researchers also have to justify that prediction errors are unrelated to any unmeasured confounder, which is extremely difficult in most applications given that researchers often have limited information about unmeasured confounders.

Therefore, researchers can ignore prediction errors only when prediction errors are completely random, i.e., prediction errors are not affected by the independent variable, the outcome variable, or any unobserved confounder. Unfortunately, this condition is untenable in almost all social science applications. While we focused on one setting where Y is text-based, similar stringent conditions are required when other types of variables (e.g., independent variables) are text-based.

General Bias Formula. While we derived a condition required to ignore prediction errors above, we primarily focused on cases when Y_i is text-based. Here, using the general framework developed in Appendix B, we consider general conditions that apply to any convex optimization problem and cases where any subset of outcome and independent variables are text-based.

In general, to ignore prediction errors, researchers need to assume

$$\mathbb{E}\left(m(D_i^{obs}, \widehat{D}_i^{mis}; \beta) - m(D_i^{obs}, D_i^{mis}; \beta)\right) = 0, \tag{OA.20}$$

which means that the moment function that plugs in predicted variables is unbiased. In a special case when outcome Y_i is text based, this reduces to the bias condition derived in the main paper.

$$\begin{aligned} & \mathbb{E}\left(m(D_i^{obs}, \widehat{D}_i^{mis}; \beta) - m(D_i^{obs}, D_i^{mis}; \beta)\right) \\ = \; & \mathbb{E}\left(m(\widehat{Y}_i, X_i; \beta) - m(Y_i, X_i; \beta)\right) \\ = \; & \mathbb{E}\left(X_i(\widehat{Y}_i - X_i^\top \beta) - X_i(Y_i - X_i^\top \beta)\right) \\ = \; & \mathbb{E}\left(X_i(\widehat{Y}_i - Y_i)\right) \end{aligned}$$

which is equal to zero when $\mathbb{E}(\widehat{Y}_i - Y_i \mid X_i) = 0$.

[1] We find differential prediction errors across categories in LLM annotations for Fowler et al. (2021), which we report in Appendix G.

D Design of Simulation Studies

Here we offer the details of the simulation study we reported in Section 3.4. The main purpose of this simulation is to clearly show that ignoring prediction errors can bias downstream estimates and standard errors. We also use empirical applications in Section 5 to show the problem of ignoring prediction errors and how DSL solves the issue under realistic real-world social science data settings.

As the entire population, we generate $n = 5000$ i.i.d. observations ($i \in \{1, \ldots, 5000\}$) as follows.

- Covariates: $X_i \sim \mathcal{N}(\overrightarrow{0}, \Sigma^X)$ where $X_i = (X_{i1}, \ldots, X_{i10})$, and $\overrightarrow{0}$ is a vector of 0 with length 10. For $i \in \{1, \ldots, 10\}$, $\Sigma^X_{ii} = 1$ and for $i \neq j$, $\Sigma^X_{ij} = 0.3$. For the second covariate, we update it by binarizing $X_{i2} = \mathbf{1}\{X_{i2} > \text{qnorm}(0.8)\}$.

- Binary Outcome: $Y_i \sim \text{Bernoulli}(\text{expit}(W_i))$ where

  $$W_i = \frac{0.1}{1 + \exp(0.5X_{i3} - 0.5X_{i2})} + \frac{1.3X_{i4}}{1 + \exp(-0.1X_{i2})} + 1.5X_{i4}X_{i6} + 0.5X_{i1}X_{i2} + 1.3X_{i1} + X_{i2}$$

  This data-generating process is similar to the one in Vansteelandt and Dukes (2022). It contains various nonlinear transformation of X and it is difficult to correctly model the outcome function.

- Prediction by the automated text annotation method: $\widehat{Y}_i = P_i Y_i + (1 - P_i)(1 - Y_i)$ where $P_i \sim \text{Bernoulli}(P_q)$ and P_q controls the accuracy of the prediction. When $P_q = 0.9$, $\widehat{Y}_i = Y_i$ with 90% and $\widehat{Y}_i = 1 - Y_i$ with 10%. We vary P_q in our simulation.

  To evaluate the general statistical behavior, here we do not use a specific automated text annotation method. Rather, by directly controlling the amount of prediction errors, we can understand how prediction errors, in general, affect downstream analyses. We used a very simple flipping error as used in Clayton et al. (2023). The realistic prediction errors, as we examine in our empirical applications in Section 5, are more likely to be complex. The main idea here is to show that bias from prediction errors are substantial even for this simple prediction errors.

- Expert Annotation: We use simple random sampling of 500 documents for expert annotation. Thus, $\Pr(R_i = 1) = 0.1$ for every observation.

Our estimand of interest is the coefficients of the oracle logistic regression where we regress Y_i on $(X_{i1}, X_{i1}^2, X_{i2}, X_{i4})$. We evaluated bias, coverage, RMSE of the estimator ignoring prediction errors and DSL in Figure 1.

We found that estimators ignoring prediction errors have large biases and invalid confidence intervals, which makes it unsuitable for social science downstream analyses. DSL has low bias and proper coverage of confidence intervals regardless of the accuracy of the underlying automated text annotation method.

E Introduction to LLM Usage

To help readers incorporate LLMs into their research workflow, we include a simple guide on how to use LLMs to generate annotations. Note that this guide was written in early 2024; at the time that users are reading, various details may have changed. Our goal is to provide a helpful starting point.

E.1 Primer on Large Language Models

We begin with a high-level explanation of how LLM annotation works. Language models assign conditional probabilities over sequences of tokens (a token is basically a word or word fragment): $\Pr(x_n \mid \langle x_1, ..., x_{n-1} \rangle)$. These conditional probabilities are used autoregressively to generate tokens, successively appending each new token to the sequence and then predicting the next one.

In concrete terms, suppose we begin with the sentence "once upon", tokenized as $\langle x_1 = \text{once},\ x_2 = \text{upon} \rangle$. A language model gives the probability of the next token, x_3, conditional on the sequence so far:

$$\Pr(x_3, \mid \langle x_1 = \text{once},\ x_2 = \text{upon} \rangle).$$

Suppose that the language model returns that the most likely next token corresponds to the word "a". We set $x_3 = \text{a}$, and then proceed to calculate the probabilities for the next token in the sequence:

$$\Pr(x_4, \mid \langle x_1 = \text{once},\ x_2 = \text{upon},\ x_3 = \text{a} \rangle).$$

And so on. This recursive process is called *autoregressive language generation*, and underlies how generative language models like GPT and Llama produce text. The process sketched above is called "greedy decoding", where at each step the token with the maximum likelihood is chosen. Obviously, this strategy may lead to suboptimal sequences (i.e., where the resulting sequence is low probability), thus alternative decoding strategies exist that either incorporate stochasticity, explore multiple steps ahead before "committing", or combine aspects of both.

E.2 Using LLMs

In general, LLMs require more computational resources than a typical statistical or machine learning model that researchers can run on a laptop. Therefore, users are likely to run LLMs using computing resources located outside of their laptop (e.g., when users run GPT models, computation is not conducted in your laptop but in OpenAI's server). There are three broad solutions for this requirement.

1. **API-based Usage**: a company hosts and manages the computing equipment and models, and charges researchers for usage (e.g. per request or per token generated). For example, GPT models can be used in this way.

2. **Cloud Computing**: Researchers secure the necessary computing resources from a cloud computing provider such as AWS, Azure, Google Cloud, and so on.

3. **HPC-based Usage**: Researchers secure access to high-performance computing resources either from their own institution or via multi-institution agreements.

The second and third options also require researchers to manage deployment and interaction with the model. This requires cloud- and HPC-specific engineering skills that go beyond the scope of this guide. This guide primarily covers API-based usage of the two families of models used in this paper: GPT and Llama-2. API-based use is currently the easiest and fastest to set up.

E.3 Using GPT

GPT is a closed-source proprietary model, and therefore can only be used via an API. The main provider of the API is OpenAI, and will be the option most researchers would start at the time of writing.

OpenAI provides multiple services. At the time of writing, GPT inference is provided under their Developer Platform at `https://platform.openai.com`. OpenAI provide a detailed Quickstart guide (at `https://platform.openai.com/docs/quickstart`) that provides all the instructions needed to get up and running with GPT models with OpenAI. Official libraries for API usage are provided in Python, bash (curl) and javascript (node.js).

For `Python` users, researchers can call GPT models via `Python` when they finish setting up their OpenAI account with the aforementioned quick guide.

For `R` users, there is no official R library for OpenAI API usage. While OpenAI itself does not offer a package, there are multiple `R` packages that researchers can use to call GPT models from `R`, e.g., `https://irudnyts.github.io/openai/`, `https://github.com/joeornstein/text2data`, `https://github.com/samterfa/openai`, and `https://openair-lib.org/`. Note that the software ecosystem evolves rapidly, especially due to changes and updates from OpenAI themselves.

E.4 Using Llama

There is no "official" API for the Llama-2 family of models from Meta. The model can be downloaded after receiving permission from Meta (request access at `https://llama.meta.com/llama-downloads/`), but as noted above the largest model (Llama-2-70B) is far too large to be run on most consumer hardware without modification to the model.

E.4.1 Third-Party APIs

Several third-party API solutions exist, including:

- HuggingFace Inference Endpoints (`https://ui.endpoints.huggingface.co`): provides a simple graphical user interface to set up your own API. Most transparent solution, with the full software stack fully open source and replicable (`https://github.com/huggingface/text-generation-inference`). Billed by time.

- Azure AI Studio (`https://ai.azure.com`): provides a way to create API endpoint with open source and OpenAI models. Billed by usage.
- Replicate (`https://replicate.com`): provides an API for many open-source models. Billed by usage.

E.4.2 Google Colab

Google provides a cloud computing and research toolkit called Google Colab (`https://colab.research.google.com`). Colab provides Jupyter notebooks that can be attached to compute runtimes with an easy-to-use graphical interface. This may be a familiar and simple solution for many researchers.

It is possible to run the 70-billion parameter Llama model on a paid Google Colab runtime instance (specifically, the A100-high memory) by quantizing the model weights to 4-bit integers. We provide an example of how to do this in our replication materials.

E.4.3 Self-Hosting

For this paper, we managed our own deployment of Llama-2-70B on Princeton's Della computing cluster. Non-quantized deployment for inference required two A100-80GB Nvidia GPUs. The codebase for this deployment is provided as part of the replication code.

F Prompts Used in Social Science Applications

F.1 Examples

In this section, to show the wide applicability of LLM annotations, we provide examples of prompts used in a wide range of social science applications. As we see below, there are huge variations in how scholars provide prompts just like there are variations in how we write codebooks and train human-coders.

Please note that, in practice, users will find that the performance of LLM annotations changes when they change prompts (even if two prompts have substantively the same meaning). This black-box sensitivity is exactly one of the key challenges of directly using LLM annotations (and ignoring prediction errors) in downstream analyses. With DSL, users can obtain statistically valid estimates regardless of the choice of LLMs and prompts because DSL can fully take into account prediction errors in LLM annotations. Indeed, as we see more thoroughly in our empirical applications (Section 5), DSL estimates are stable regardless of the choice of LLMs.

- **Sentiment Classification** (Ornstein, Blasingame and Truscott, 2022)

  ⋆ Goal: Classify the sentiment of social media posts

  ⋆ Texts to be labeled:

  ```
  Congratulations to the SCOTUS. American confidence in the Supreme Court is now
  lower than at any time in history. Well done!
  ```

⋆ Prompt:

```
Decide whether a Tweet's sentiment is positive, neutral, or negative.

Tweet: Congratulations to the SCOTUS. American confidence in the Supreme Court is now lower than at any time in history. Well done!
Sentiment:
```

- **Ideological Scaling Task** (Ornstein, Blasingame and Truscott, 2022)

  ⋆ Goal: Classify the ideology of a political manifesto

  ⋆ Texts to be labeled:

  ```
  We will implement a comprehensive strategy for ending low pay, notably by the introduction of a statutory national minimum wage.
  ```

  ⋆ Prompt:

  ```
  Decide whether this sentence from a British political manifesto is Liberal, Conservative, or Neither.

  Sentence: We will implement a comprehensive strategy for ending low pay, notably by the introduction of a statutory national minimum wage.
  Classification:
  ```

- **Classification of Topics** (Rytting et al., 2023)

  ⋆ Goal: Classify a topic of news headlines

  ⋆ Texts to be labeled:

  ```
  House Panel Votes Tax Cuts, But Fight Has Barely Begun
  ```

  ⋆ Prompt:

  ```
  Using only the following categories
  """
  Macroeconomics
  Civil Rights, Minority Issues, and Civil Liberties
  Health
  Agriculture
  Labor
  Education
  Environment
  Energy
  ```

```
Immigration
Transportation
Law, Crime, and Family Issues
Social Welfare
Community Development and Housing Issues
Banking, Finance, and Domestic Commerce
Defense
Space, Science, Technology and Communications
Foreign Trade
International Affairs and Foreign Aid
Government Operations
Public Lands and Water Management
State and Local Government Administration
Weather and Natural Disasters
Fires
Arts and Entertainment
Sports and Recreation
Death Notices
Churches and Religion
Other, Miscellaneous, and Human Interest
"""
Assign the following headlines to one of the categories:
House Panel Votes Tax Cuts, But Fight Has Barely Begun ->
```

- **Hate Speech** (Ziems et al., 2023)

  ⋆ Goal: Classify a type of hate speech

  ⋆ Texts to be labeled:

  `jewish harvard professor noel ignatiev wants to abolish the white race via #wr`

  ⋆ Prompt:

```
jewish harvard professor noel ignatiev wants to abolish the white race via #wr

Which of the following categories of hate speech best describes the sentence above?
A: White Grievance (frustration over a minority group's perceived privilege and casting majority groups as the real victims of racism)
B: Incitement to Violence (flaunting in-group unity and power or elevating known hate groups and ideologies)
```

```
C: Inferiority Language (implies one group or individual is inferior to another, including dehumanization and toxification)
D: Irony (the use of sarcasm, humor, and satire to attack or demean a protected class or individual)
E: Stereotypes and Misinformation (associating a protected class with negative attributes)
F: Threatening and Intimidation (conveys a speaker commitment to a target's pain, injury, damage, loss, or violation of rights)

Constraint: Answer with one or more of the options above that is most accurate and nothing else. Always choose at least one of the options.
```

- **Attitudes toward Immigrants** (Mets et al., 2023)

  ⋆ Goal: Classify attitudes toward immigrants

  ⋆ Texts to be labeled:

  ```
  Unfortunately, by now the violence has seeped from immigrant communities to all of the society.
  ```

  ⋆ Prompt:

```
Tag the following numbered sentences as being either "supportive", "against" or "neutral" towards the topic of immigration. "Supportive" means: "supports immigration, friendly to foreigners, wants to help refugees and asylum seekers". "Against" means: "against immigration, dislikes foreigners, dislikes refugees and asylum seekers, dislikes people who help immigrants". "Neutral" means: "neutral stance, neutral facts about immigration, neutral reporting about foreigners, refugees, asylum seekers". Don't explain, output only sentence number and stance tag.

1. Unfortunately, by now the violence has seeped from immigrant communities to all of the society.
2. [truncated]
3. [truncated]
```

- **Public Opinion on Wars** (Zhu et al., 2023)

  ⋆ Goal: Classify whether a social media post is pro-Russia, Pro-Ukraine, or unbiguous.

  ⋆ Texts to be labeled:

```
International Criminal Court: Stop Putin's War Crimes - Sign the Petition!
https://t.co/NyaFp6TTNj via @Chang
```

⋆ Prompt:

```
Give the tweet about Russo-Ukrainian Sentiment a label from Pro-Russia,
Pro-Ukraine, or Not Sure.
Tweet: "International Criminal Court: Stop Putin's War Crimes - Sign the
Petition! https://t.co/NyaFp6TTNj via @Chang"
Label:
Explanation:
```

F.2 Prediction Performance in Other Applications: Figure 3

To further illustrate this wide variation in prediction performance, we also analyze a diverse set of empirical validation studies. In particular, based on a review paper by Ollion et al. (2023), we collected eight recent papers that examine the performance of LLM annotations in the social sciences and report the F-1 scores (or publicly share data so that we could compute the F-1 scores). Eight papers are as follows: Heseltine and Clemm Von Hohenberg (2023); Kuzman, Ljubešić and Mozetič (2023); Mellon et al. (2022); Mets et al. (2023); Møller et al. (2023); Rathje et al. (2023); Yang and Menczer (2023); Ziems et al. (2023). In each paper, they evaluate multiple different tasks, so we have 113 tasks in total. We find that F-1 scores range from as low as 20% to more than 95%, and many tasks show about $70 \sim 80\%$ (see Panel (c) in Figure 3). This huge variation in prediction accuracy is a common feature of LLM annotations in the social sciences.

We want to emphasize that this set of tasks is not representative of text annotation tasks that social scientists might perform. The results are probably the over-estimation of the current LLM performance given that these papers are trying to show the promise of LLM annotations. At the same time, the LLM performance is also likely to go up quickly as we have better and larger LLMs over time. This descriptive analysis is only meant to show the promise but also the risk of using LLMs.

G LLM Annotations for Fowler et al. (2021)

G.1 Specification of LLM Annotations

G.1.1 Models

We use three LLMs for our empirical application based on Fowler et al. (2021): GPT-3.5 (`gpt-3.5-turbo-0613`), GPT-4 (`gpt-4-preview-1106`) and Llama-2-70B-chat. This choice is informed primarily by leaderboard performance and popularity in the scientific community. In general, we recommend using a state-of-the-art LLM (currently, GPT-4) and one state-of-the-art open-source LLM (currently, Llama-2).

The GPT models are a collection of proprietary decoder-only transformer models from OpenAI. Architecturally, GPT-3.5 is known to be a 175 billion parameter model whereas the details

of GPT-4 are not publicly confirmed.[2] For these models, we set the temperature to 0.0 as we are predicting short sequences and are more interested in the probabilities that GPT places over the classes, and restrict the maximum number of new tokens to 10. Texts that would cause the maximum sequence length to be exceeded were truncated, and all generated texts were classified into the three target classes by searching for the final instance of one of the three class names after lowercasing the sequence. We used the GPT models via the OpenAI Python API.

Llama-2 is a series of decoder-only transformer model trained and shared by Meta. At the time of writing, Llama-2 is available in three sizes: 7 billion, 13 billion and 70 billion parameters. We use the largest Llama-2 model because a consistent finding in the computer science literature is that larger models generally perform better. To implement Llama-2, we used the Della supercomputing cluster at Princeton University, which houses compute nodes with 4×A100 80GB NVidia GPUs. Quantizing the model to 4-bit integers (`nf4`), we were able to fit the 70-billion parameter model on a single device, enabling data and tensor parallelism to quickly generate our labels. Researchers can also use the third-party API, such as HuggingFace Inference Endpoints (`https://ui.endpoints.huggingface.co`) to implement Llama-2 without self-hosting. In order to perform batch processing with Llama-2, it is necessary to pad the sequences so that all observations in each batch are the same length. Thus, during pre-processing we padded or truncated all prompts to be the same length (4000 tokens, to leave 96 tokens for the generated text). However, including the pad tokens does affect the generated texts, meaning that the results would be different if we were to do the texts one-at-a-time without padding.

G.1.2 Prompt

LLMs can be used to obtain predicted labels for texts by including the text to be labeled in the condition and then autoregressively generating the predicted label. In general, users should include (a) a codebook, (b) texts to be labeld, and (c) an answer box as prompts.

In our empirical application based on Fowler et al. (2021), we follow the wording in the WMP codebook instructions used in the original paper.

```
In your judgment, is the primary purpose of the ad text to promote a specific
candidate, attack a candidate, or contrast the candidates? Answer either "contrast",
"promote", or "attack".

text: """
{text}
"""

Answer:
```

Here, `{text}` denotes a text to be labeled (i.e., a text of a political advertisement). The combined prompt-plus-text is then given as an input to a LLM, which generates text using

[2]Unofficially, it is thought to be a 8×220 billion parameter mixture-of-experts (MoE) model.

the autoregressive process described above. Thus, given the political ad text `Trump National Coalition Chair running for Congress here in NH`, the input to the language model would be:

```
In your judgment, is the primary purpose of the ad text to promote a specific candidate, attack a candidate, or contrast the candidates? Answer either "contrast", "promote", or "attack".

text: """
Trump National Coalition Chair running for Congress here in NH
""""

Answer:
```

G.1.3 Few-Shot Learning

In this section we consider "few-shot learning" (also known as "in-context" learning) to improve the prediction performance of LLM annotations. This consists of incorporating task-answer pairs, called "exemplars", in the prompt. The previously seen prompts are referred to as "zero-shot" because they contain zero exemplars. We now generate predictions for the same prompts with six exemplars (6-shot). To select our exemplars, we labeled a randomly sampled subset of the data and selected our exemplars from this subset.

When selecting exemplars, we chose a balanced number of exemplars from each class. To reflect different types of cases, we aimed to include diverse examples.

Incorporating an exemplar in a prompt typically consists of repeating the observation and answer portion of the prompt. For our 6-shot prompt, we have the following.

```
In your judgment, is the primary purpose of the ad text to promote a specific candidate, attack a candidate, or contrast the candidates? Answer either "contrast", "promote", or "attack".

Text:"""
On Tuesday, Nevada voters will choose between a local problem solver and a con man who funneled money from a children's charity into a failed political campaign. It's no wonder voters have rejected Danny Tarkanian 5 times.
"""
Answer: contrast

Text:"""
My opponent supports raising income taxes on you by over $2,800 a year! I believe that's just wrong and I'll fight for lower taxes so that you can keep more of
```

your hard-earned money. By now you know kim schreyer aka dr. tax you've heard about her 57 cents a gallon gas tax. i'm dino rossi and i approve this message.
"""
Answer: attack

Text"""
Donald Trump wants to roll back women's choice through the Supreme Court. I'll protect women's health care, and fully fund Planned Parenthood, in Florida.
"""
Answer: promote

Text"""
My commitment is to always be on the side of Arizona families, not big donors. As voting nears, the choice couldn't be more clear -- the politician that is focused on serving donors, or the new candidate that is focused on serving you. I ask you to stand with me.
"""
Answer: contrast

Text"""
Nobody thought a Democrat could win in Alabama. But last year, Doug Jones shocked the country and flipped a historically deep-red seat blue. Now, it is our turn.
"""
Answer: promote

Text"""
Lt Governor Dan Patrick cut public education funding by over five billion dollars, cut more than ten thousand teaching positions, and cut support for pre-K. With fewer teachers and larger class sizes, Dan Patrick won't let teachers teach and students learn. We need new leadership in Texas - vote Mike Collier for Lt Governor.
"""
Answer: attack

Text"""
{text}
"""
Answer:

G.1.4 Other Recommendations

When formulating their prompt, researchers may also want to consider model-specific factors. For instance, HuggingFace suggests that the chat variants of Llama-2 be prompted using the following template, which is based on the model's training procedure.

```
<s>[INST] <<SYS>>
{{ system_prompt }}
<</SYS>>

{{ user_message }} [/INST]
```

However, these guidelines are better thought of reasonable hypotheses extrapolated from aspects of the model training process rather than surefire ways to generate great predictions.

G.2 Additional LLM Results

In Section 2.1.2, we reported the F1 scores for the overall performance of LLM annotations. Here, we provide additional details.

Figure OA-2 shows the overall and disaggregated performance. Several points are worth noting. First, the performance can vary across models and prompts. Second, the prediction accuracy is not uniform across three categories. For all models, it is easiest to predict "Promote" and hardest to predict "Contrast". This directly means that prediction errors are affected by the outcome of interest (i.e., the tone of ads) itself, which implies that researchers cannot ignore prediction errors (see Section 3.3).

In the main paper, we reported F1 score, which is the most common measure of prediction performance in the computer science and machine learning literature, instead of the classification accuracy. It is important to remember that the classification accuracy can be high just because the category is rare. For example, when the category takes 1 only with 5%, by predicting 0 for every observation, we can get 95% accuracy automatically. This is the main reason why F1 scores are generally recommended as a measure of prediction performance when classes are imbalanced. Given this caveat, for some researchers who might be more familiar with accuracy, we also report the classification accuracy for LLM predictions in Figure OA-3.

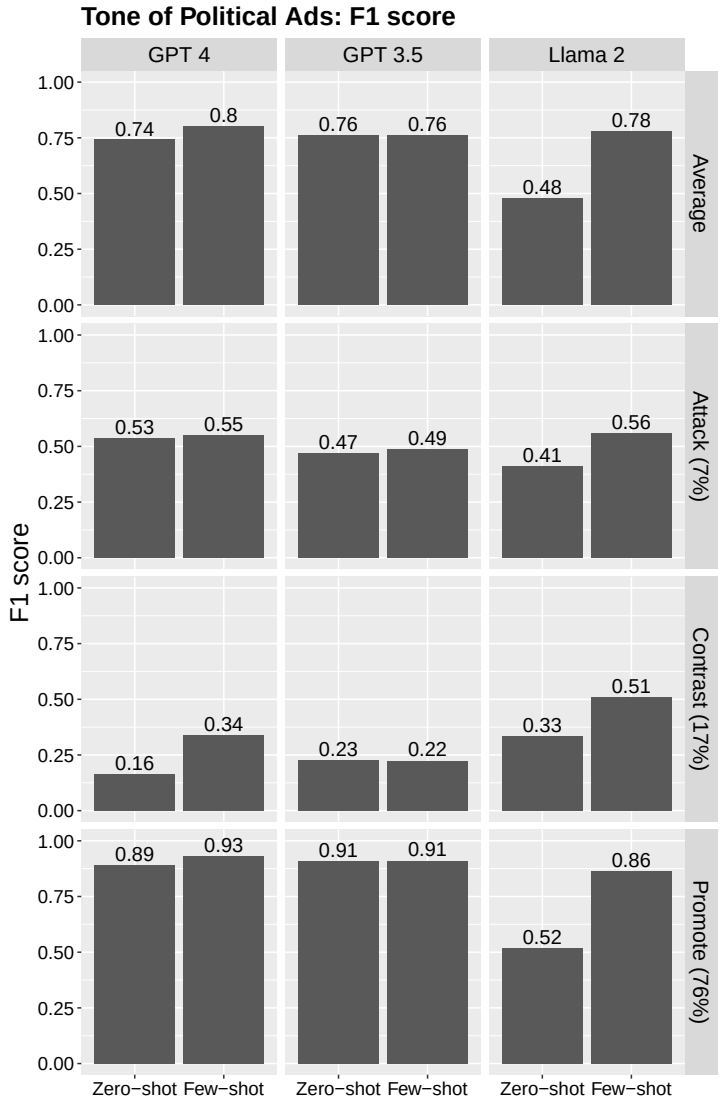


Figure OA-2: **F1 scores of LLM annotations: Fowler et al. (2021).**
Note: "Average" shows the overall performance which computes the weighted average of F1 scores for three categories: "Attack", "Contrast", and "Promote" constitute 7%, 17%, and 76% of political ads.

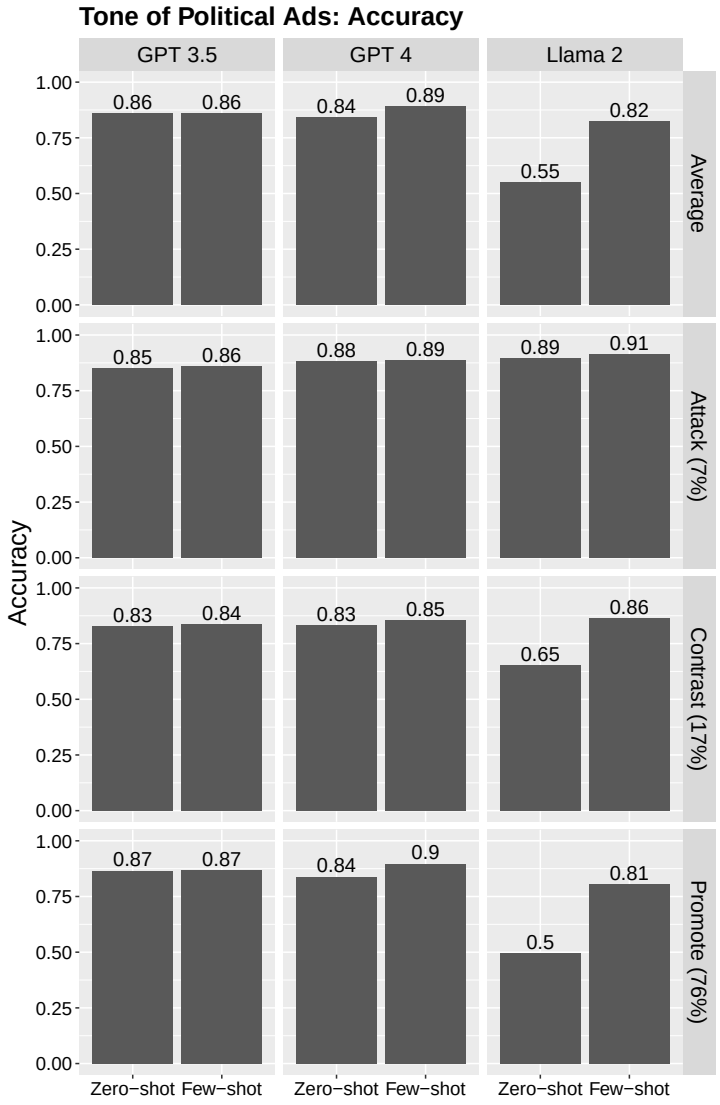


Figure OA-3: **Accuracy of LLM annotations: Fowler et al. (2021).**
Note: "Average" shows the overall performance which computes the weighted average of the classification accuracies for three categories: "Attack", "Contrast", and "Promote" constitute 7%, 17%, and 76% of political ads.

H Additional DSL Results for Fowler et al. (2021)

In the main paper, we reported the results for two outcomes, "Contrast" and "Promote". In this appendix, we also report the results for the third outcome "Attack." The main findings are similar. First, estimates from the LLM-only estimation are biased, and substantive and statistical conclusions can flip depending on which LLMs users choose. Confidence intervals are also, in general, invalid. For this outcome, the LLM-only estimation with Llama-2 has reasonable coverages, but this is a statistical coincidence without any theoretical guarantee. Indeed, if we look back at two other outcomes, the same estimator has poor coverage. Second, estimates from the classical supervised learning method are similarly biased because they ignore prediction errors, and they have invalid confidence intervals. In contrast to these existing approaches, DSL has unbiased estimates and valid confidence intervals.

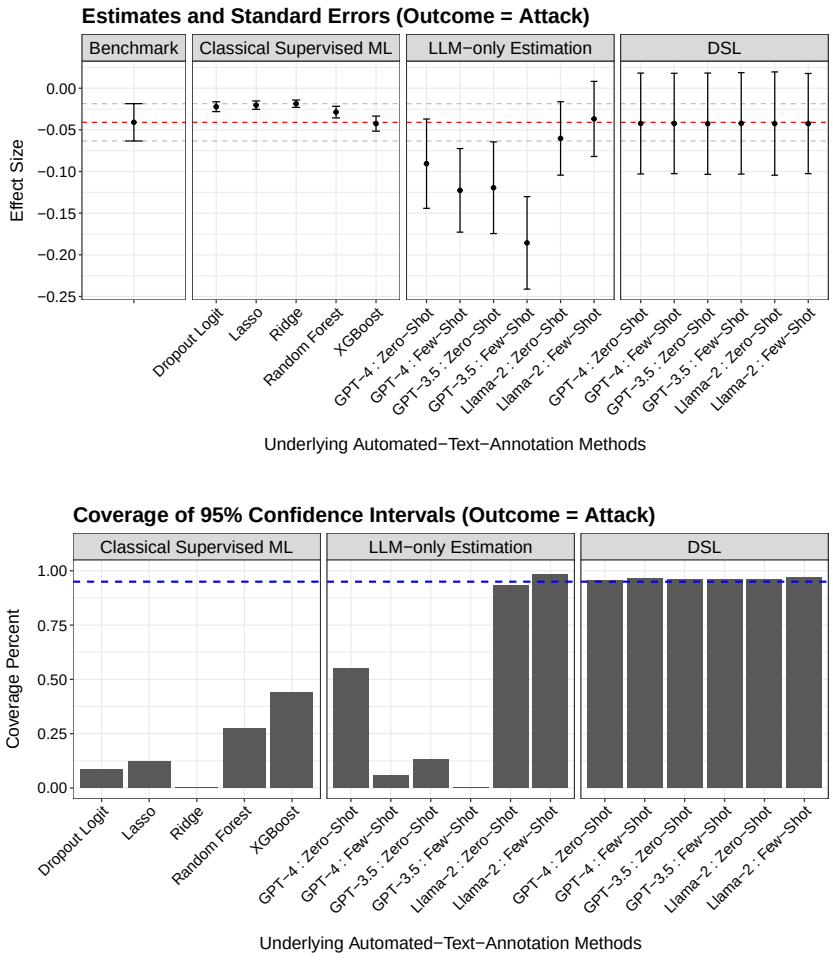


Figure OA-4: **Results for Outcome = "Attack" in Fowler et al. (2021).** *Note*: In the top panel, red dotted lines represent point estimates of the "Benchmark" estimates and gray dotted lines represent their 95% confidence intervals. To show the average performance across random sampling of expert-coding, we report the average point estimates and standard errors across 500 repeated sampling. In the bottom panel, blue dotted lines represent 95%.

I LLM Annotations for Pan and Chen (2018)

I.1 Specification of LLM Annotations

I.1.1 Models

As we did for the first application, we use three LLMs for our empirical application based on Pan and Chen (2018): GPT-3.5, GPT-4 and Llama-2-70B-chat.

I.1.2 Prompt

In this empirical application, we annotate two different variables: *Prefecture Wrongdoing* (whether each citizen complaint accuses of prefecture-level wrongdoing) and *County Wrongdoing* (whether each citizen complaint accuses of county-level wrongdoing). For *Prefecture Wrongdoing*, we used the following prompt for GPTs. For Llama-2, we found that the performance based on Chinese prompts is so poor that we decided to use the English version of the same prompt.

```
下面的中文文本是发生在中国江西省九江市的一起投诉，判断帖子是否指控了九江市地级层面的政府官员或政府机构有不当行为，包括指控腐败和暴力，以及违反法律和法规。请注意，评估仅考虑针对九江市地级的指控，不包括针对下辖地区及其子地区的任何指控（这些下辖地区包括'濂溪区'、'浔阳区'、'柴桑区'、'瑞昌市'、'共青城市'、'庐山市'、'武宁县'、'修水县'、'永修县'、'德安县'、'都昌县'、'湖口县'、'彭泽县'）。

如果帖子包含对九江市地级官员或机构的上述性质的指控，请返回'1'。否则，请返回'0'。

文本："""
{text}
"""
回答：
```

where {text} denotes that a text to be labeled (i.e., each online complaint). The combined prompt-plus-text is then given as an input to a LLM.

For *County Wrongdoing*, we used the following prompt for GPTs. Again, for Llama-2, we found that the performance based on Chinese prompts is so poor that we decided to use the English version of the same prompt.

```
下面的文本是发生在中国江西省九江市的一起投诉。判断帖子是否指控了九江市下辖地区及其子地区的政府官员或政府机构有不当行为，这些不当行为包括指控腐败和暴力，以及违反法律和法规。请注意，评估仅考虑针对九江市地级下辖地区或其子地区的指控（下辖地区包括濂溪区、浔阳区、柴桑区、瑞昌市、共青城市、庐山市、武宁县、修水县、永修县、德安县、都昌县、湖口县、彭泽县），不包括任何针对九江市地级的指控。

如果帖子包含针对九江市地级市下辖地区或其子地区官员或机构的上述性质的指控，
```

请返回'1'。否则，请返回'0'。

文本："""
{text}
"""
回答:

I.1.3 Few-Shot Learning

As in the first example, we also consider few-shot learning by adding diverse examples. For *Prefecture Wrongdoing*, we used the following prompt.

下面的中文文本是发生在中国江西省九江市的一起投诉，判断帖子是否指控了九江市地级层面的政府官员或政府机构有不当行为，包括指控腐败和暴力，以及违反法律和法规。请注意，评估仅考虑针对九江市地级的指控，不包括针对下辖地区及其子地区的任何指控（这些下辖地区包括'濂溪区'、'浔阳区'、'柴桑区'、'瑞昌市'、'共青城市'、'庐山市'、'武宁县'、'修水县'、'永修县'、'德安县'、'都昌县'、'湖口县'、'彭泽县'）。

如果帖子包含对九江市地级官员或机构的上述性质的指控，请返回'1'。否则，请返回'0'。

文本: """
媒体报道：各地突击提拔干部透视：一把手权力过于集中所致。文中提及，今年6月,江西省委常委、秘书长赵智勇涉嫌违纪被免职后,就被媒体曝出,他2006年离开九江前1个月,曾突击提拔了一批女干部,有的学校老师直接被提拔为区团委副书记,不少属于破格提拔,但后来接任的领导在接到群众反映后,又把提拔的一部分女干部打回原单位。（人民网等36家媒体）"""
回答: 1

文本: """
网友举报九江市工商局局长原局长孔祥华贪污腐化，包养情妇等问题。（天涯社区）"""
回答: 1

文本: """
网曝九江市公路管理局九江分局局长李广金长期公车私用，出入酒店。（九江论坛）"""
回答: 1

文本: """
网友质疑永修县医保局敛财，2012年初办理医保时，永修县医保局强迫必须多缴两年无编制期间的医保统筹金，共计2700多元，否则不予办理医保。统筹金不打入个人医保帐户，单位也没有任何补助，卡里没有一分钱，等于是白交，还不开任何发票凭证。（问政江

西）”””
回答: 0

文本: ”””
网友举报武宁县船滩镇党委委员黄少华违规违纪，以搭干股、圈矿山用地、贩卖土方石料的方式聚敛钱财，经常出入于大小宾馆酒店聚众赌博，拉帮结派。（大江论坛）”””
回答: 0

文本：”””
{text}
”””
回答:

For *County Wrongdoing*, we used the following prompt.

下面的文本是发生在中国江西省九江市的一起投诉。判断帖子是否指控了九江市下辖地区及其子地区的政府官员或政府机构有不当行为，这些不当行为包括指控腐败和暴力，以及违反法律和法规。请注意，评估仅考虑针对九江市地级下辖地区或其子地区的指控（下辖地区包括濂溪区、浔阳区、柴桑区、瑞昌市、共青城市、庐山市、武宁县、修水县、永修县、德安县、都昌县、湖口县、彭泽县），不包括任何针对九江市地级的指控。

如果帖子包含针对九江市地级市下辖地区或其子地区官员或机构的上述性质的指控，请返回'1'。否则，请返回'0'。

文本: ”””
网友质疑永修县医保局敛财，2012年初办理医保时，永修县医保局强迫必须多缴两年无编制期间的医保统筹金，共计2700多元，否则不予办理医保。统筹金不打入个人医保帐户，单位也没有任何补助，卡里没有一分钱，等于是白交，还不开任何发票凭证。（问政江西）
”””
回答: 1

文本: ”””
网友举报武宁县船滩镇党委委员黄少华违规违纪，以搭干股、圈矿山用地、贩卖土方石料的方式聚敛钱财，经常出入于大小宾馆酒店聚众赌博，拉帮结派。（大江论坛）
”””
回答: 1

文本: ”””
都昌县汪墩乡茅垅村质疑信访事项答复意见书，称镇领导在征收农田过程中大发横财。

（天涯社区）
"""
回答: 1

文本: """
网曝九江市公路管理局九江分局局长李广金长期公车私用，出入酒店。（九江论坛）
"""
回答: 0

文本: """
网友举报九江市工商局局长原局长孔祥华贪污腐化，包养情妇等问题。（天涯社区）
"""
回答: 0

文本："""
{text}
"""
回答:

I.2 Additional LLM Results

In Section 2.1.2, we reported the F1 scores for the overall performance of LLM annotations. Here, we provide additional details.

Figure OA-2 shows F1 scores and the classification accuracy for both "Prefecture Wrongdoing" and "County Wrongdoing". Several points are worth noting. First, the performance varies across two tasks. It is easier to predict "Prefecture Wrongdoing" than to predict "County Wrongdoing" across models and prompts. Second, in this application, the prediction performance is relatively stable across models and prompts. However, as we see in Section 5, even though the average prediction performance is relatively similar, because prediction errors are non-random, LLM-only estimation using different LLM annotations produces very different results. This highlights the methodological problem of ignoring prediction errors.

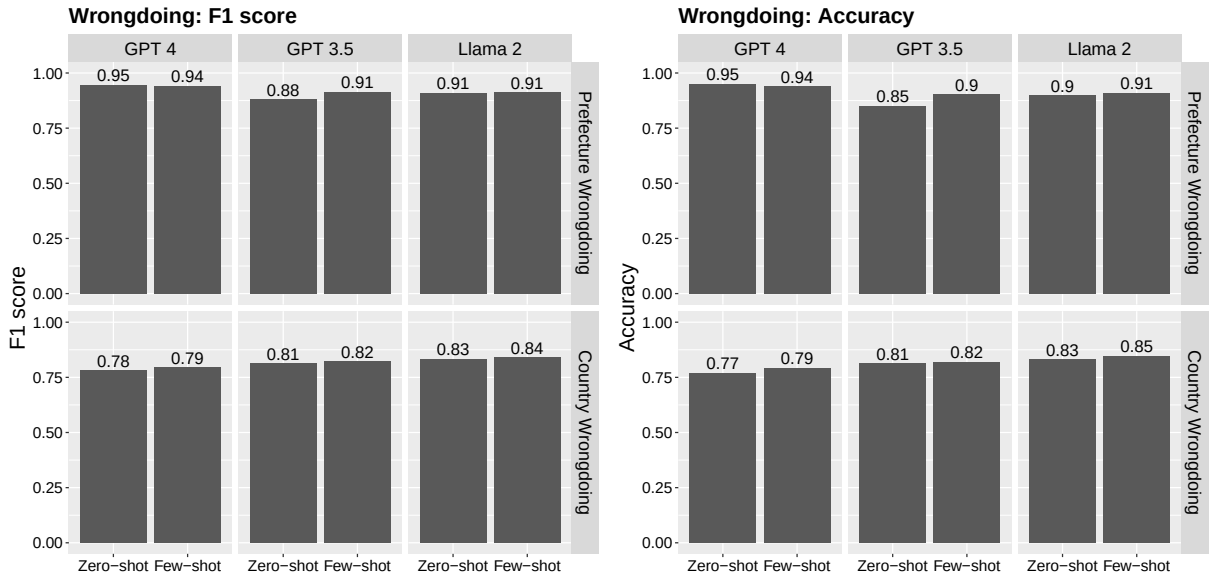


Figure OA-5: **Prediction Performance of LLM annotations: Pan and Chen (2018).** *Note*: The left panel shows F1 scores and the right panel shows the classification accuracy.

J Additional DSL Results for Pan and Chen (2018)

Setup

DSL requires simple four steps. First, we generate LLM annotations for the entire population of documents. As we discussed in Section 2, we here consider six versions: GPT 4, GPT 3.5, and Llama 2 with zero-shot and few-shot learning. In the second step, we randomly sample 500 documents for expert-coding.[3] In the third step, using the expert-coded data, we further improve LLM predictions by cross-fitting the generalized random forest (Athey, Tibshirani and Wager, 2019) to predict the expert-coded labels with LLM annotations produced in the first step. Finally, we combine expert-coded labels and predicted labels in the DSL logistic regression with exactly the same specification in the original paper. In particular, we regress the upward reporting (i.e., whether a given complaint is reported upward to provincial-level officials) on the aforementioned two independent variables (*Prefecture Wrongdoing* and *County Wrongdoing*) and other control variables with interactions.

$$\begin{aligned}\text{Upward Reporting} \quad \sim \quad & \text{Prefecture Wrongdoing} + \text{County Wrongdoing} + \text{Connections} + \\ & \text{County Wrongdoing} \times \text{Connections} + \text{Other Controls}\end{aligned}$$

where "Other Controls" include prevalence, group issue, sentiment, personal experience, collective action, petitions, and provincial jurisdiction. See Column (3) in Table 3 of the original paper. Importantly, "Prefecture Wrongdoing" and "County Wrongdoing" are text-based.

We compare DSL against the classical supervised learning approach and the LLM-only estimation. For the classical supervised learning approach, we examine five widely used supervised ML methods: lasso, ridge, random forest, and XGBoost. We use a term-document matrix used in the original paper, which has more than 5000 variables, as predictors. For the LLM-only estimation, we consider the same six versions of LLM annotations. We expect that these existing approaches can provide unbiased estimates with valid confidence intervals, only when prediction errors are completely random, while DSL provides valid statistical guarantees even with arbitrary prediction errors.

Results

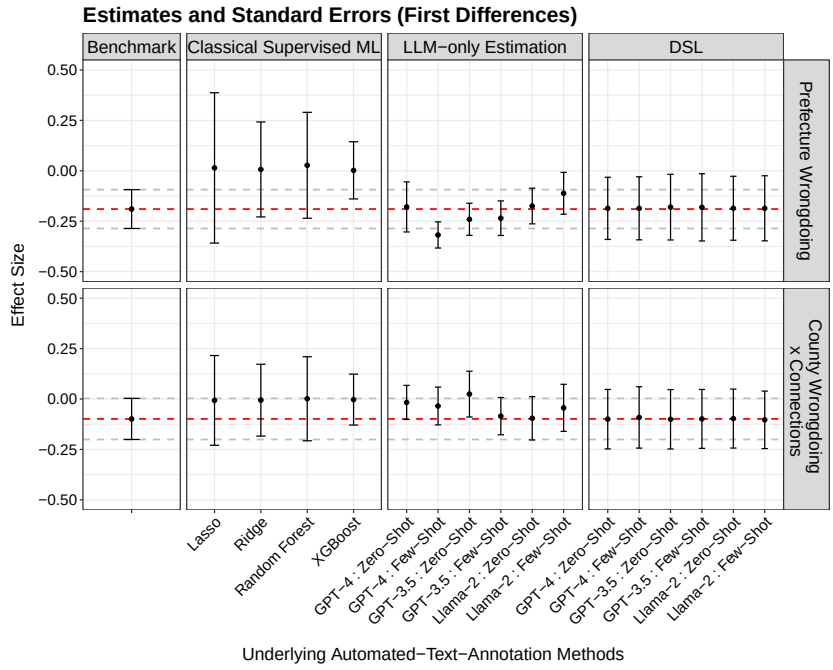
In the main paper, we reported the results for two main coefficients, "Prefecture Wrongdoing" and "County Wrongdoing". In this appendix, we also report the results based on the first differences because coefficients of logistic regression tend to be difficult to interpret and it is often recommended to report results based on the differences in predicted probabilities. As we emphasized in the paper, researchers can apply DSL and then use estimated coefficients to compute the first differences or any other function of estimated coefficients.

Here we specifically focus on the two effects that the original authors focused on most: the effect of "Prefecture Wrongdoing" and the effect of "Connection" for posts accusing of "Country

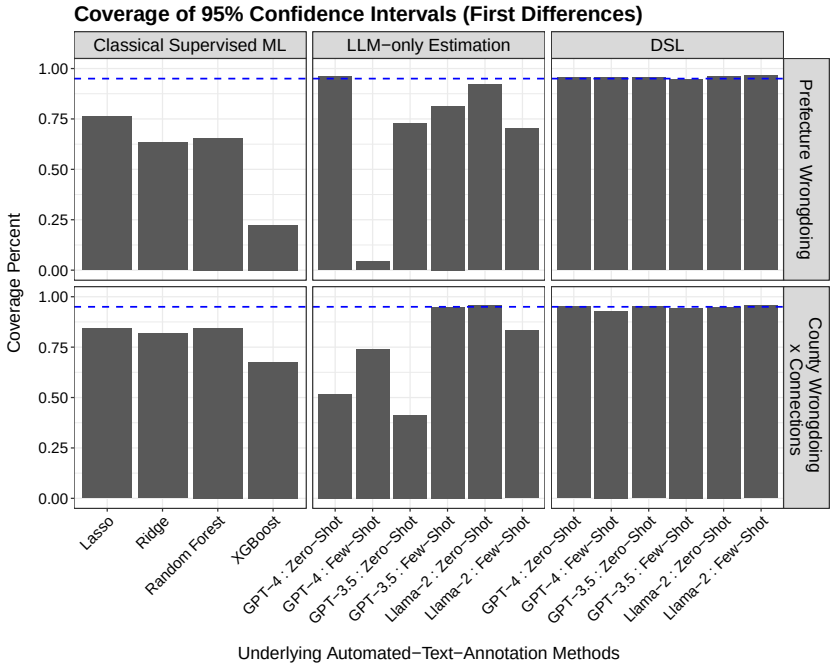
[3] In this empirical validation, we rely on expert-coding from the original authors, so we simply reveal expert-coding for sampled documents.

Wrongdoing". We report the results in Figure OA-6.

The main findings are similar. First, estimates from the LLM-only estimation are biased, and substantive and statistical conclusions can flip depending on which LLMs users choose. Confidence intervals are also, in general, invalid. Some LLM-only estimators have reasonable coverages for at least one outcome, but no LLM-only estimator has 95% coverages for both effects because they do not have any theoretical guarantees. Second, estimates from the classical supervised learning method are similarly biased because they ignore prediction errors, and they have invalid confidence intervals. In contrast to these existing approaches, DSL has unbiased estimates and valid confidence intervals, regardless of the underlying automated text annotation method.



(a)



(b)

Figure OA-6: **Comparisons of DSL and Existing Approaches in terms of First Differences using Pan and Chen (2018).**
Note: In Panel (a), red dotted lines represent point estimates of the "Benchmark" estimates and gray dotted lines represent their 95% confidence intervals. To show the average performance across random sampling of expert-coding, we report the average point estimates and standard errors across 500 repeated sampling. In Panel (b), blue dotted lines represent 95%.

K Literature Review

To evaluate the current practice of text annotations in text-as-data applications in political science, we conducted a review of academic articles published in the top 10 political science journals: American Political Science Review (APSR), American Journal of Political Science (AJPS), Journal of Politics (JOP), Political Behavior (PB), Quarterly Journal of Political Science (QJPS), British Journal of Political Science (BJPS), Comparative Political Studies (CPS), World Politics (WP), International Organization (IO), and Journal of Experimental Political Science (JEPS). These journals represent a group of highly cited and influential journals in political science. For example, these 10 journals together have total citations of over 7,800 on average as compared to the 1,315 average total citation counts across all academic journals in the field of political science. Furthermore, the 5-year journal impact factor among these 10 journals is 5.8 on average, more than twice as large as the average score across all political science journals.[4]

We first searched for all articles published in the years 2015 through 2022 (inclusive) using a keyword "text as data" and "text analysis" in Web of Science. We also included the articles published in the above-mentioned top journals that cite Grimmer and Stewart (2013). In total, we reviewed 88 papers. We note that this number is the lower bound of the actual number of papers using text-as-data methods because some papers do not explicitly use terminologies, such as "text as data" and "text analysis".

We then manually coded the following information for each paper. (1) Whether a paper uses some forms of text annotations (if so, we continue to code for the remaining items): (2) A type of downstream analyses that use text-based variables: (3) Whether text-based variables are used as the outcome and/or independent variables in the downstream analyses: (4) Whether a paper explicitly acknowledges the potential biases due to prediction errors: (5) Whether a paper statistically addresses the potential biases due to prediction errors: (6) If a paper uses the classical supervised learning approach, what is the F-1 score and the classification accuracy of the text classification step?

L Practical Guide: Additional Recommendations

In this section, we summarize our practical recommendations in each step of DSL.

L.1 Step 1: Predict Text Labels with LLMs

In the first step, researchers use LLMs to predict text labels for the entire population of documents. See our examples of prompts in Section 2 and Appendix F.

L.1.1 Which LLMs should we use?

Researchers can often start with zero-shot learning (i.e., no exemplar) using the state-of-the-art LLM. We also recommend implementing at least one open-source LLM like Llama-2. Researchers

[4]These values are based on a total of 307 political science journals recorded in the Journal Citation Reports provided by Web of Science.

can also consider few-shot learning (i.e., adding exemplars). Note that specifics of LLM implementations are likely to evolve quickly over time given the speed of the LLM development.

More importantly, researchers do not need to choose one specific LLM in the proposed DSL framework. If researchers have multiple high-performing LLM annotations, they can incorporate all of them in DSL estimators. This is because DSL only uses LLM annotations as predictors for expert-coded labels, and we do not make any assumptions about errors in LLM annotations.

L.1.2 What if LLM annotations are Very Bad?

If LLM annotations have extremely low predictive performance, DSL is going to have large standard errors because bias-correction terms are large, even though it will still maintain statistical validity. In this case, it is recommended to retrain the automated text annotation step (e.g., using different LLMs or training the classical supervised machine learning method), which is the same advice as in the classical supervised learning literature. Using low-quality LLM annotations will not invalidate the statistical properties of DSL, but having higher performing automated text annotation will reduce standard errors and the required number of expert annotations.

L.1.3 Should we spend time on improving LLMs or increasing the number of expert annotations if we want to reduce standard errors?

In general, we recommend researchers should spend time on increasing the number of expert annotations as long as the predictive performance of the underlying automated text annotation method is reasonably high (around $80 \sim 90\%$, as in our examples in Section 2). When the predictive performance becomes moderately high, it is often difficult to further improve the predictive accuracy or F1 scores by more than 5 percentage points. Standard errors of DSL do not often reduce much even if the predictive accuracy of the underlying text annotation methods improves by 1 or 2 percentage points. In contrast, increasing the number of expert annotations is theoretically guaranteed to reduce standard errors of the downstream analyses. Researchers can also use a power analysis to explicitly predict how much standard errors will decrease by adding a certain number of expert annotations (see Section 5.1.3).

L.2 Step 2: Sample Documents for Expert Annotation

In the second step, we sample a subset of documents for expert annotations. See Section 6 of the main texts for recommendations about how to handle errors in expert annotations.

L.2.1 Required Number of Expert Annotations

How many documents experts need to annotate depends on applications. To answer this question in each specific application, we develop a data-driven power analysis: after annotating a small number of documents, users can predict how many more documents they need to annotate in order to achieve a user-specified size of standard error. For example, as in traditional power analysis, suppose researchers expect the main treatment effect to be about 4 percentage points and would like to design their study such that standard errors are about 2 percentage points in order to detect the expected effect size with a conventional threshold for statistical significance.

In this case, for instance, after annotating 250 political ads, our data-driven power analysis can predict how many additional expert-annotated documents are required to make standard errors below 2 percentage points. See our empirical application in Section 5.1.3.

L.2.2 Construct Validity and Prediction Errors

In this paper, we developed a method to account for prediction errors, which is the discrepancy between expert annotations and automated text annotations. An equally important problem is the construct validity, which is a question about a mapping between a theoretical concept of interest and expert annotations. The proposed method can only account for prediction errors, and this does *not* replace careful, theoretical considerations about how operationalization in a user-specified codebook relates to the main theoretical concept. Rather, our method *augments* theoretical thinking about the construct validity by allowing researchers to focus on expert annotations and removing any additional error from the use of automated text annotation.

L.2.3 How to Sample Documents for Expert Annotations

DSL can allow for any sampling method for expert annotations as long as it is controlled by researchers, including the most common random sampling and any sampling method that depends on document-level observed covariates. In practice, researchers can often start with random sampling with equal probabilities. If researchers wish to over-sample documents that are difficult to annotate, one possible approach is to increase the sampling probability for documents that LLMs are more uncertain about (e.g., Li et al., 2023). Active learning is also a promising area of research to improve text sampling in the classical supervised learning settings (Bosley et al., 2022).

L.3 Step 3: Train an ML model to further improve LLM-prediction

In this third step, we fit a supervised ML model via cross-fitting where we predict the expert-coded labels with predictors that include LLM annotations generated in Step 1 and any other variables that are predictive (e.g., term-document matrices). This step helps to calibrate LLM annotations using the expert-coded labels (similar to fine-tuning in LLMs). Our companion `R` package `dsl` can implement this third and the next fourth steps with one function.

DSL estimator does not assume the correct specification of the underlying prediction model. Thus, DSL estimates are consistent and have valid confidence intervals regardless of the choice of ML models.[5] This step is practically important because if researchers have multiple high-performing LLMs, they can include all of them as predictors in this step.

L.4 Step 4: Fitting DSL Regression

The final step combines the expert annotations and automated annotations within the DSL framework. Researchers can apply the DSL framework to a large class of generalized linear models commonly used in social science applications (e.g., linear, logistic, multinomial-logistic,

[5]In our companion `R` package `dsl`, we use random forest as the default method, while users can always confirm that their results are indeed stable when changing ML models, as our theoretical results imply.

Poisson, and linear fixed-effects regression). Our framework also accommodates settings where any subset of the outcome variable and independent variables are text-based.

While we so far focused on regression analyses, which are the most common downstream text analyses, the DSL framework can be used for a broader range of statistical analyses in the social sciences. Whenever researchers have high-quality expensive annotations (e.g., expert annotations) and large-scale lower-quality annotations (e.g., automated text annotations), there are risks of prediction errors, and DSL can be employed to properly take into account prediction errors. Specifically, we outline the two common research questions here.

L.4.1 Estimation of Category Proportions over Time or across Groups

Many scholars are interested in estimating the proportion of all documents in each user-specified category (e.g., Hopkins and King, 2010; Keith and O'Connor, 2018; Card and Smith, 2018; Jerzak, King and Strezhnev, 2023). For example, we might study how the proportion of censored documents changes over time or how the proportion of social media posts containing hate speech differs across groups, such as Democrats and Republicans.

These questions can be analyzed within the DSL framework, too. Specifically, researchers can regress a text category on time or groups. For example, using whether a document is censored as the outcome and time indicators as the independent variable in the DSL linear regression, researchers can estimate how the proportion of censored documents changes over time. When users include time-fixed effects, this estimation method is non-parametric and is equivalent to estimating the proportion of censored documents in each time period separately.

L.4.2 Causal Inference with Texts

An increasing number of scholars make causal inference with textual data (Fong and Grimmer, 2021; Egami et al., 2022; Feder et al., 2022; Mozer and Miratrix, 2023). DSL can be used in causal inference applications where the outcome, treatment, or confounders are text-based. In randomized experiments, researchers can use the DSL regression to perform the difference-in-means or covariate-adjusted linear regression for estimating the average treatment effect.[6] In observational studies, under corresponding causal identification assumptions, researchers can apply the DSL two-stage-least squares for the instrumental variable design, the DSL local linear regression for the regression discontinuity design, and the DSL two-way fixed effects estimator for the difference-in-differences design.

[6]Previous studies (e.g., Fong and Grimmer, 2021; Egami et al., 2022) have clarified the challenges of inferring a codebook and causal estimates from the same data, especially when using unsupervised learning approaches. In contrast, this paper relies on a supervised learning framework where a codebook is given by researchers rather than estimated by a model.

References

Angelopoulos, Anastasios N., Stephen Bates, Clara Fannjiang, Michael I. Jordan and Tijana Zrnic. 2023. "Prediction-powered inference." *Science* 382(6671):669–674.
URL: *https://www.science.org/doi/abs/10.1126/science.adi6000*

Argyle, Lisa P, 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.

Athey, Susan, Julie Tibshirani and Stefan Wager. 2019. "Generalized Random Forests." *The Annals of Statistics* 47(2):1148 – 1178.
URL: *https://doi.org/10.1214/18-AOS1709*

Bommasani, Rishi, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill et al. 2021. "On the Opportunities and Risks of Foundation Models." *arXiv preprint arXiv:2108.07258* .

Bosley, Mitchell, Saki Kuzushima, Ted Enamorado and Yuki Shiraito. 2022. "Improving Probabilistic Models in Text Classification via Active Learning." *arXiv preprint arXiv:2202.02629* .

Card, Dallas and Noah A Smith. 2018. The importance of calibration for estimating proportions from annotations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. pp. 1636–1646.

Chakrabortty, Abhishek, Guorong Dai and Eric Tchetgen Tchetgen. 2022. "A General Framework for Treatment Effect Estimation in Semi-Supervised and High Dimensional Settings." *arXiv preprint arXiv:2201.00468* .

Chakrabortty, Abhishek and Tianxi Cai. 2018. "Efficient and adaptive linear regression in semi-supervised settings." *Annals of Statistics* .

Chen, Yi-Hau and Hung Chen. 2000. "A Unified Approach to Regression Analysis under Double-Sampling Designs." *Journal of the Royal Statistical Society Series B: Statistical Methodology* 62(3):449–460.

Chernozhukov, Victor, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey and James Robins. 2018. "Double/Debiased Machine Learning for Treatment and Structural Parameters." *Econometrics Journal* 21:C1 – C68.

Chernozhukov, Victor, Juan Carlos Escanciano, Hidehiko Ichimura, Whitney K Newey and James M Robins. 2022. "Locally robust semiparametric estimation." *Econometrica* 90(4):1501–1535.

Clayton, Katherine, Yusaku Horiuchi, Aaron R Kaufman, Gary King and Mayya Komisarchik. 2023. Correcting Measurement Error Bias in Conjoint Survey Experiments. Technical report Working Paper.

Davidian, Marie. 2022. "Methods based on semiparametric theory for analysis in the presence of missing data." *Annual Review of Statistics and Its Application* 9:167–196.

Egami, Naoki, 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. **URL:** *https://www.science.org/doi/abs/10.1126/sciadv.abg2652*

Egami, Naoki, 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." *Advances in Neural Information Processing Systems* 36.

Feder, Amir, Katherine A Keith, Emaad Manzoor, Reid Pryzant, Dhanya Sridhar, Zach Wood-Doughty, Jacob Eisenstein, Justin Grimmer, Roi Reichart, Margaret E Roberts et al. 2022. "Causal inference in natural language processing: Estimation, prediction, interpretation and beyond." *Transactions of the Association for Computational Linguistics* 10:1138–1158.

Fong, Christian and Justin Grimmer. 2021. "Causal Inference with Latent Treatments." *American Journal of Political Science* .

Fong, Christian and Matthew Tyler. 2021. "Machine learning predictions as regression covariates." *Political Analysis* 29(4):467–484.

Fowler, Erika Franklin, Michael M Franz, Gregory J Martin, Zachary Peskowitz and Travis N Ridout. 2021. "Political Advertising Online and Offline." *American Political Science Review* 115(1):130–149.

Gilardi, Fabrizio, Meysam Alizadeh and Maël Kubli. 2023. "ChatGPT Outperforms Crowd-Workers for Text-Annotation Tasks." *arXiv preprint arXiv:2303.15056* .

Grimmer, Justin 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.

Heseltine, Michael and B Clemm Von Hohenberg. 2023. "Large Language Models as A Substitute for Human Experts in Annotating Political Text." *preprint SocArxiv: cx752* .

Hopkins, Daniel J and Gary King. 2010. "A Method of Automated Nonparametric Content Analysis for Social Science." *American Journal of Political Science* 54(1):229–247.

Jerzak, Connor T, Gary King and Anton Strezhnev. 2023. "An improved method of automated nonparametric content analysis for social science." *Political Analysis* 31(1):42–58.

Kallus, Nathan and Xiaojie Mao. 2020. "On the role of surrogates in the efficient estimation of treatment effects with limited outcome data." *arXiv preprint arXiv:2003.12408* .

Katsumata, Hiroto and Soichiro Yamauchi. 2023. Statistical Analysis with Machine Learning Predicted Variables. Technical report Working Paper.

Keith, Katherine and Brendan O'Connor. 2018. Uncertainty-aware generative models for inferring document class prevalence. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics pp. 4575–4585.
URL: *https://aclanthology.org/D18-1487*

Kennedy, Edward H. 2022. "Semiparametric doubly robust targeted double machine learning: a review." *arXiv preprint arXiv:2203.06469* .

Kennedy, Edward H, Sivaraman Balakrishnan and Max G'Sell. 2020. "Sharp instruments for classifying compliers and generalizing causal effects." *Annals of Statistics* .

Knox, Dean, Christopher Lucas and Wendy K Tam Cho. 2022. "Testing Causal Theories with Learned Proxies." *Annual Review of Political Science* 25:419–441.

Kuzman, Taja, Nikola Ljubešić and Igor Mozetič. 2023. "ChatGpt: Beginning of An End of Manual Annotation? Use Case of Automatic Genre Identification." *arXiv preprint arXiv:2303.03953* .

Li, Minzhi, Taiwei Shi, Caleb Ziems, Min-Yen Kan, Nancy F Chen, Zhengyuan Liu and Diyi Yang. 2023. "CoAnnotating: Uncertainty-guided Work Allocation between Human and Large Language Models for Data Annotation." *arXiv preprint arXiv:2310.15638* .

Linegar, Mitchell, Rafal Kocielnik and R Michael Alvarez. 2023. "Large Language Models and Political Science." *Frontiers in Political Science* 5:1257092.

Mellon, Jonathan, 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." *SSRN preprint: 4310154* .

Mets, Mark, Andres Karjus, Indrek Ibrus and Maximilian Schich. 2023. "Automated Stance Detection in Complex Topics and Small Languages: The Challenging Case of Immigration in Polarizing News Media." *arXiv preprint arXiv:2305.13047* .

Møller, Anders Giovanni, Jacob Aarup Dalsgaard, Arianna Pera and Luca Maria Aiello. 2023. "Is A Prompt and A Few Samples All You Need? Using GPT-4 for Data Augmentation in Low-Resource Classification Tasks." *arXiv preprint arXiv:2304.13861* .

Mozer, Reagan and Luke Miratrix. 2023. "Decreasing the Human Coding Burden in Randomized Trials with Text-based Outcomes via Model-Assisted Impact Analysis." *arXiv preprint arXiv:2309.13666* .

Newey, Whitney K and Daniel McFadden. 1994. "Large Sample Estimation and Hypothesis Testing." *Handbook of econometrics* 4:2111–2245.

Ollion, Etienne, Rubing Shen, Ana Macanovic and Arnault Chatelain. 2023. "Chatgpt for Text Annotation? Mind the Hype!" *SocArXiv. October* 4.

Ornstein, Joseph T, Elise N Blasingame and Jake S Truscott. 2022. How to Train Your Stochastic Parrot: Large Language Models for Political Texts. Technical report Working Paper.

Palmer, Alexis and Arthur Spirling. 2023. Large Language Models Can Argue in Convincing and Novel Ways About Politics: Evidence from Experiments and Human Judgement. Technical report Working paper.

Pan, Jennifer and Kaiping Chen. 2018. "Concealing Corruption: How Chinese Officials Distort Upward Reporting of Online Grievances." *American Political Science Review* 112(3):602–620.

Pangakis, Nicholas, Samuel Wolken and Neil Fasching. 2023. "Automated Annotation with Generative AI Requires Validation." *arXiv preprint arXiv:2306.00176* .

Rathje, Steve, Dan-Mircea Mirea, Ilia Sucholutsky, Raja Marjieh, Claire Robertson and Jay J Van Bavel. 2023. "GPT is An Effective Tool for Multilingual Psychological Text Analysis.".

Robins, James M and Andrea Rotnitzky. 1995. "Semiparametric efficiency in multivariate regression models with missing data." *Journal of the American Statistical Association* 90(429):122–129.

Robins, James M, Andrea Rotnitzky and Lue Ping Zhao. 1994. "Estimation of Regression Coefficients When Some Regressors Are Not Always Observed." *Journal of the American Statistical Association* 89(427):846–866.

Rotnitzky, Andrea and Stijn Vansteelandt. 2014. Double-robust methods. In *Handbook of missing data methodology*. CRC Press pp. 185–212.

Rytting, Christopher Michael, Taylor Sorensen, Lisa Argyle, Ethan Busby, Nancy Fulda, Joshua Gubler and David Wingate. 2023. "Towards Coding Social Science Datasets with Language Models." *arXiv preprint arXiv:2306.02177* .

Tsiatis, Anastasios A. 2006. *Semiparametric theory and missing data*. Springer.

Vansteelandt, Stijn and Oliver Dukes. 2022. "Assumption-lean Inference for Generalised Linear Model Parameters." *Journal of the Royal Statistical Society Series B: Statistical Methodology* 84(3):657–685.
URL: *https://doi.org/10.1111/rssb.12504*

Wang, Siruo, Tyler H McCormick and Jeffrey T Leek. 2020. "Methods for Correcting Inference based on Outcomes Predicted by Machine Learning." *Proceedings of the National Academy of Sciences* 117(48):30266–30275.

Wu, Patrick Y, Joshua A Tucker, Jonathan Nagler and Solomon Messing. 2023. "Large Language Models Can be Used to Estimate the Ideologies of Politicians in A Zero-Shot Learning Setting." *arXiv preprint arXiv:2303.12057* .

Yang, Kai-Cheng and Filippo Menczer. 2023. "Large Language Models Can Rate News Outlet Credibility." *arXiv preprint arXiv:2304.00228* .

Zhang, Han. 2021. "How Using Machine Learning Classification as a Variable in Regression Leads to Attenuation Bias and What to Do About It." SocArXiv.

Zhu, Yiming, Peixian Zhang, Ehsan-Ul Haq, Pan Hui and Gareth Tyson. 2023. "Can ChatGpt Reproduce Human-Generated Labels? A Study of Social Computing Tasks." *arXiv preprint arXiv:2304.10145* .

Ziems, Caleb, William Held, Omar Shaikh, Jiaao Chen, Zhehao Zhang and Diyi Yang. 2023. "Can Large Language Models Transform Computational Social Science?" *arXiv preprint arXiv:2305.03514* .