Supporting Information for

Leveraging Generative Artificial Intelligence for Causal Inference with Unstructured Data

Kosuke Imai and Kentaro Nakamura

Corresponding author: Kosuke Imai
E-mail: imai@harvard.edu

This PDF file includes:

Supporting Information Text

1. A Simulation Study

We illustrate the intuition and performance of the proposed estimation procedure through a simple simulation study with 200 Monte Carlo replications.

A. Simulation setup. We generate the data so that the treatment T has highly restricted support conditional on the representations $\boldsymbol{R}$ but only part of the representation $\boldsymbol{R}$ acts as a confounder, which corresponds to confounding feature $\boldsymbol{U}$. As a result, the overlap assumption is satisfied conditional on the confounding feature. Such settings arise naturally with high-dimensional unstructured data such as text or images, where features are numerous but many are irrelevant to the outcome.

Specifically, for each observation $i \in \{1, \cdots, 5000\}$, we generate the d-dimensional representations by $\boldsymbol{R}_i \sim \text{MVN}(0, \boldsymbol{\Sigma}_d)$, where $(\boldsymbol{\Sigma}_d)_{kk} = 1$ and $(\boldsymbol{\Sigma}_d)_{kj} = 0.1$ for $j \neq k$, with $d = 4096$. We then split $\boldsymbol{R}_i$ as

$$\boldsymbol{R}_i = \begin{bmatrix} \boldsymbol{R}_i^{(t)} \\ \boldsymbol{R}_i^{(u)} \end{bmatrix}$$

where $\dim(\boldsymbol{R}_i^{(t)}) = 16$ and $\dim(\boldsymbol{R}_i^{(u)}) = 4080$. We then generate treatment T_i, confounder $\boldsymbol{U}$, and outcome Y_i as

$$T_i = \mathbf{1}\{\mathbf{1}_{16}^\top \boldsymbol{R}_i^{(t)} > 0\}, \qquad \boldsymbol{U}_i = \tanh(\boldsymbol{A}\boldsymbol{R}_i^{(u)}) \qquad Y_i = \tau T_i + 20 \times \mathbf{1}_{d_u}^\top \boldsymbol{U}_i + \epsilon_i, \qquad \epsilon_i \sim \mathcal{N}(0, 1)$$

where $d_u = \dim(\boldsymbol{U}_i) = 16$, $\tau = 1$, and $\boldsymbol{A} \in \mathbb{R}^{16 \times 4080}$ is a loading matrix where each entry is drawn from $\mathcal{N}(0, \frac{1}{4})$.

In this data generating process (DGP), the direct adjustment of $\boldsymbol{R}_i$ leads to the overlap violation, while conditioning on the confounder does not. Our goal is to properly adjust for confounder even though we do not directly observe it and must estimate it from $\boldsymbol{R}_i$. We generate the data 200 times, apply the estimation procedure described below, and then calculate performance metrics, including bias, root mean squared error (RMSE), and coverage of the 95% confidence interval.

B. Estimation Procedure. We compare four estimators on the simulated data: our GPI estimator under two architectures and two baselines. Since we know the DGP, the two GPI variants let us probe the role of architecture: (i) one sized to capture exactly the necessary and sufficient part of $\boldsymbol{R}$, and (ii) one over-parameterized to capture most of $\boldsymbol{R}$. The baseline estimators are a naive application of augmented inverse probability weighting with double machine learning (AIPW–DML) (1) and DragonNet (2, 3), which represent two standard approaches for handling high-dimensional confounders but neither of them addresses violation of the overlap assumption.

Specifically, we first apply the GPI estimation procedure, which is designed to identify the treatment effect in this setting without directly observing $\boldsymbol{U}_i$. In short, GPI identifies the treatment effect by adjusting for a deconfounder $\boldsymbol{f}(\boldsymbol{R}_i)$ that satisfies the mean independence condition

$$\mathbb{E}[Y_i \mid \boldsymbol{R}_i, T_i] = \mathbb{E}[Y_i \mid \boldsymbol{f}(\boldsymbol{R}_i), T_i]. \qquad [1]$$

We have shown that any function $\boldsymbol{f}(\boldsymbol{R}_i)$ satisfying this mean independence condition enables identification of the treatment effect. See (4) and the application sections for the theoretical properties of GPI.

To estimate the treatment effect, we use the neural network-based estimation procedure proposed in (4). Specifically, we use the Treatment-Agnostic Representation Network (TarNet) of (5), which estimates the conditional potential outcome function given the deconfounder:

$$\mu_t(\boldsymbol{f}(\boldsymbol{R}_i)) := \mathbb{E}[Y_i(t) \mid \boldsymbol{f}(\boldsymbol{R}_i)] \quad \text{for } t = 0, 1.$$

Our architecture simultaneously estimates the deconfounder and the outcome model by minimizing

$$\{\hat{\boldsymbol{\lambda}}, \hat{\boldsymbol{\theta}}_0, \hat{\boldsymbol{\theta}}_1\} := \underset{\boldsymbol{\lambda}, \boldsymbol{\theta}_0, \boldsymbol{\theta}_1}{\arg\min} \frac{1}{N} \sum_{i=1}^{N} \left\{ Y_i - \mu_{T_i}(\boldsymbol{f}(\boldsymbol{R}_i; \boldsymbol{\lambda}); \boldsymbol{\theta}_{T_i}) \right\}^2, \qquad [2]$$

where N denotes the sample size, and we make the neural network parameters explicit: $\boldsymbol{\lambda}$ denotes the parameters of the deconfounder $\boldsymbol{f}$, and $\boldsymbol{\theta}_t$ denotes the parameters of the outcome regression function μ_t. We then estimate the propensity score model given the estimated deconfounder,

$$\pi(\boldsymbol{f}(\boldsymbol{R}_i; \hat{\boldsymbol{\lambda}})) = \Pr(T_i = 1 \mid \boldsymbol{f}(\boldsymbol{R}_i; \hat{\boldsymbol{\lambda}})),$$

and apply the double machine learning procedure using the estimated outcome models $\{\hat{\mu}_1, \hat{\mu}_0\}$ and the estimated propensity score model $\hat{\pi}$.

Using a deconfounder allows us to discard information in $\boldsymbol{R}_i$ that is predictive only of treatment assignment, thereby making the positivity assumption more plausible. This feature also highlights the importance of architectural choices. The deconfounder network must be sufficiently expressive to retain the outcome-relevant variation in $\boldsymbol{R}_i$ and satisfy the mean independence condition in Eq. (1), while remaining sufficiently parsimonious to avoid retaining variation that only predicts treatment assignment. If the architecture is too restrictive, the outcome model may fail to converge to the true population

Table S1. Simulation performance across 200 Monte Carlo replications. We report bias, root mean squared error (RMSE), and the empirical coverage of the 95% confidence interval.

Estimator	Mean Absolute Bias	RMSE	95% CI Coverage
GPI (Exact)	1.692	2.146	0.965
GPI (over-parameterized)	3.269	4.212	0.760
DragonNet	3.440	4.291	0.850
AIPW–DML	4.394	6.207	0.560

outcome function. If it is too flexible or over-parameterized, however, the learned deconfounder may preserve treatment-specific information from $\boldsymbol{R}_i$, thereby reintroducing the same overlap problem that arises when directly adjusting for the full representation. Therefore, it is crucial to select architectures and hyperparameters that achieve strong out-of-sample outcome prediction performance, since treatment-specific information that is irrelevant to the outcome should not improve out-of-sample prediction.

We demonstrate the importance of choosing an appropriate architecture by comparing an exact TarNet architecture with an over-parameterized TarNet architecture. For the exact architecture, we use a one-layer neural network with output dimension 1024 for the deconfounder and a one-layer neural network with output dimension 1 for each outcome head, using ReLU activation between layers. This architecture is not over-parameterized because we know that the relationship between the confounding feature and the outcome is linear. We compare this exact architecture with an over-parameterized architecture, which uses two consecutive layers with dimensions 4096 for the deconfounder and two consecutive layers with dimensions 10 and 1 for each outcome head.

For both architectures, we optimize the network using Adam optimizer with learning rate 0.00002 and batch size 256, and early stopping is triggered if validation loss fails to improve for 5 consecutive epochs. For propensity score estimation, we used a neural network with spectral normalization (6) to enforce Lipschitz continuity. This network consisted of a two-layer MLP with ReLU activation, using 256 hidden units in the first layer and 128 in the second. The learning rate was fixed at 2×10^{-5}, and the model was trained under the same early stopping criteria. We used a two-fold cross-fitting procedure for the estimation.

We compare these GPI estimators with two baseline estimators. The first is DragonNet (2, 3), which is a widely used neural network architecture when dealing with unstructured data (3, 7). DragonNet predicts the outcome using a shared representation, while also encouraging this representation to be predictive of treatment assignment. Specifically, it minimizes the combined outcome and treatment-prediction loss function,

$$\frac{1}{N}\sum_{i=1}^{N}\left\{Y_i - Q_{T_i}(\boldsymbol{b}(\boldsymbol{R}_i))\right\}^2 - \frac{1}{N}\sum_{i=1}^{N}\left\{T_i \log g(\boldsymbol{b}(\boldsymbol{R}_i)) + (1-T_i)\log[1-g(\boldsymbol{b}(\boldsymbol{R}_i))]\right\}, \quad [3]$$

where $Q_t(\cdot)$ is the outcome model under treatment status $T_i = t$, $g(\cdot)$ is the treatment-prediction model, and $\boldsymbol{b}(\boldsymbol{R}_i)$ is the shared representation across treatment. To compare with the GPI estimator, we use a one-layer neural network with output dimension 1024 for the shared representation, a one-layer network with output dimension for each outcome head, and a two-layer network with dimension 256 and 128 for treatment head, and optimize the neural network in the exact same way as the GPI estimator. We then use the estimated outcome prediction $\{\hat{Q}_1, \hat{Q}_0\}$ and treatment prediction $\hat{g}$ for the double machine learning (DML) procedure with two-fold cross-fitting (1).

Finally, we naively implement the DML procedure with the AIPW score directly using the full representation $\boldsymbol{R}_i$ as covariates. Specifically, we estimate both the treatment and outcome models using random forests with 100 trees, a minimum leaf size of 10, and 50% of the input features considered as candidate variables at each split. We then estimate the treatment effect using two-fold cross-fitting.

C. Results. Table S1 reports the bias, RMSE, and empirical coverage of the 95% confidence interval. As expected, GPI with a properly selected TarNet mitigates the positivity violation and eliminates the resulting bias and achieves nominal coverage, whereas the two baselines, DragonNet and AIPW–DML, suffer from severe bias. GPI with an over-parameterized TarNet also exhibits non-negligible bias because it fails to adequately reduce the dimensionality of the inputs. These results highlight the importance of properly tuning the architecture and hyperparameters of TarNet within the GPI framework.

2. Text as Confounder

In this section, we provide a formal theoretical justification of our GPI methodology introduced in Section 1 of the main text and its implementation details.

A. Theoretical Properties. Our approach extends the method proposed by (4) to the case where text serves as confounder. We begin by defining the causal quantity of interest. We then establish its nonparametric identification and develop an estimation strategy.

A.1. Setup and Assumptions. Suppose we observe a sample of N independent and identically distributed (i.i.d.) units $i = 1, \ldots, N$ from a population of interest. For each unit i, we observe a treatment assignment $T_i \in \mathcal{T} \subset \mathbb{R}$ and an observed outcome $Y_i \in \mathcal{Y} \subset \mathbb{R}$, where $\mathcal{T}$ and $\mathcal{Y}$ denote the support of the treatment and that of the outcome, respectively. Additionally, we observe a set of structured confounders $\boldsymbol{Z}_i \in \mathcal{Z} \subset \mathbb{R}^d$ and unstructured high-dimensional objects (e.g., texts or images) $\boldsymbol{X}_i \in \mathcal{X} \subset \mathbb{R}^r$, some features of which also serve as confounders.

We adopt the potential outcomes framework for causal inference and assume the standard consistency assumption (8). Specifically, we assume that the potential outcome, denoted by $Y_i(t)$, depends solely on the treatment assignment of unit i itself, not on those of other units. The assumption is formalized as follows:

Assumption 1. (Consistency) *The potential outcome under the treatment $t \in \mathcal{T}$ is denoted by $Y_i(t)$, and equals the observed outcome Y_i under the realized treatment assignment T_i:*

$$Y_i = Y_i(T_i).$$

We are interested in estimating the marginal mean of the potential outcome under treatment condition $T_i = t$ for some $t \in \mathcal{T}$ (i.e., $\mathbb{E}[Y_i(t)]$), which can be used to estimate the average treatment effect (ATE). We consider the strong latent ignorability assumption; conditional on the observed covariates and certain unknown features of the high-dimensional unstructured objects, potential outcomes are independent of treatment assignment. This assumption guarantees the existence of such latent confounding features.

Assumption 2. (Strong Latent Ignorability with Unknown Confounding Features) *There exists a deterministic function $g_{\boldsymbol{U}} : \mathcal{X} \to \mathcal{U}$ that maps an unstructured object $\boldsymbol{X}_i$ onto the low-dimensional confounding features $\boldsymbol{U}_i \in \mathcal{U}$ with $\dim(\boldsymbol{U}) \ll \dim(\boldsymbol{X})$ such that the potential outcomes are independent of the treatment assignment given the observed covariates $\boldsymbol{Z}_i$ and the confounding features $\boldsymbol{U}_i = g_{\boldsymbol{U}}(\boldsymbol{X}_i)$:*

$$\{Y_i(t)\}_{t \in \mathcal{T}} \perp\!\!\!\perp T_i \mid \boldsymbol{Z}_i = \boldsymbol{z}, \boldsymbol{U}_i = \boldsymbol{u},$$

where $\mathbb{P}(T_i = t \mid \boldsymbol{Z}_i = \boldsymbol{z}, \boldsymbol{U}_i = \boldsymbol{u}) > 0$ for all $t \in \mathcal{T}$, $\boldsymbol{z} \in \mathcal{Z}$, and $\boldsymbol{u} \in \mathcal{U}$.

Importantly, we assume the existence of low-dimensional features $\boldsymbol{U}_i$, which are deterministic functions of the unstructured objects and sufficient to satisfy the strong ignorability assumption. We do not condition directly on the high-dimensional unstructured objects $\boldsymbol{X}_i$ because doing so can often lead to perfect prediction of treatment assignment, thereby violating the positivity condition and yielding severely biased estimates (9).

Prior work either overlooks this issue (e.g., (3, 10)) or addresses it by using text-based matching (e.g., (11, 12)). Matching is not well suited for high-dimensional controls and the reliance on parametric propensity score models such as topic models leads to bias. In contrast, our method assumes only the existence of low-dimensional confounding features $\boldsymbol{U}_i$.

Finally, we assume that unstructured objects are generated by a deep generative model. When the existing images and text are of interest, we can reproduce them by appropriately prompting a deep generative model. Following (4), we adopt a broad definition of deep generative model to encompass LLMs and other foundation models.

Definition 1. (Deep Generative Model) *A deep generative model is the following possibly degenerate probabilistic model that takes prompt $\mathbb{P}_i$ as an input and generates the unstructured object $\boldsymbol{X}_i$ as an output:*

$$\mathbb{P}(\boldsymbol{X}_i \mid \boldsymbol{h}_{\boldsymbol{\gamma}}(\boldsymbol{R}_i))$$
$$\mathbb{P}(\boldsymbol{R}_i \mid \boldsymbol{P}_i)$$

where $\boldsymbol{R}_i \in \mathcal{R} \subset \mathbb{R}^d$ denotes an internal representation of $\boldsymbol{X}_i$ contained in the model and $\boldsymbol{h}_{\boldsymbol{\gamma}}(\boldsymbol{R}_i)$ is a deterministic function parameterized by $\boldsymbol{\gamma}$ that completely characterizes the conditional distribution of $\boldsymbol{X}_i$ given $\boldsymbol{R}_i$.

Our definition encompasses a wide range of existing text and image generation models. Under this definition of deep generative model, $\boldsymbol{R}$ is a lower-dimensional representation of the unstructured object $\boldsymbol{X}$ and is also a hidden representation of neural networks. We assume that the last layer of the deep generative model is a deterministic function of the internal representation $\boldsymbol{R}_i$ so that the low-dimensional features of the unstructured object $\boldsymbol{U}_i$ can be regarded as a deterministic function of the low-dimensional internal representation $\boldsymbol{R}_i$ of the deep generative model. The assumption explicitly writes a generative model with a hierarchical structure to emphasize that we can use any intermediate layer before the final layer $\boldsymbol{h}_{\boldsymbol{\gamma}}(\boldsymbol{R}_i)$ so long as $\boldsymbol{U}_i$ is a deterministic function of selected layer.

Assumption 3. (Deterministic Decoding) *The output layer of a deep generative model is deterministic. That is, $\mathbb{P}(\boldsymbol{X}_i \mid \boldsymbol{h}_{\boldsymbol{\gamma}}(\boldsymbol{R}_i))$ in Definition 1 is a degenerate distribution.*

Crucially, we only require that $\boldsymbol{R}_i$ deterministically generates the unstructured object $\boldsymbol{X}_i$. For example, some generative models, especially diffusion models, have an internal architecture that is stochastic, but the output of the decoder layer can be made deterministic so that GPI is still applicable. In the case of LLMs, (13) shows that due to updates to the internal parameters, the outputs of LLMs are not generally replicable over time. However, in this context, we are only concerned

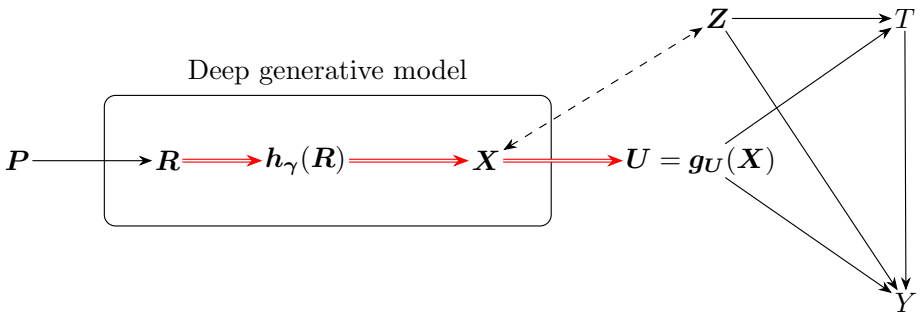

Fig. S1. Directed Acyclic Graph of the Assumed Data-Generating Process with Unstructured Confounder. An unstructured object $\boldsymbol{X}$ (e.g., an image or text) is generated via a deep generative model (shown as a box), where a prompt $\mathbb{P}$ produces an internal representation $\boldsymbol{R}$, which is then transformed into $\boldsymbol{X}$ by the function $\boldsymbol{h}_{\boldsymbol{\gamma}}(\boldsymbol{R})$. The causal relationship between treatment T and outcome Y is confounded by both observed structured covariates $\boldsymbol{Z}$ and latent confounding features $\boldsymbol{U}$ embedded within the unstructured object $\boldsymbol{X}$. Arrows with red double lines represent deterministic causal relationships while single-lined arrows denote possibly stochastic relationships and dashed bidirectional arrows indicate potential independence.

with the deterministic relationship between $\boldsymbol{R}_i$ and $\boldsymbol{X}_i$, which can be controlled by selecting appropriate hyperparameters for deterministic decoding under any given LLM.

We emphasize that researchers can apply GPI to existing text or image data by prompting deep generative models to reproduce the original unstructured inputs as faithfully as possible. For text data, this can be accomplished using instruction-tuned models with a system prompt that enforces verbatim reproduction. For example, in our text applications (4, 14), we used the prompt: "You are a text generator who simply repeats the input text," which consistently led LLMs to reproduce the original text verbatim. Although exact reproduction may occasionally fail, such cases can be readily detected by directly comparing the generated output with the original text.

A similar strategy can be applied to images using diffusion models. Because diffusion models are inherently stochastic, however, the regenerated image is typically only approximately identical to the original. In these settings, the resulting internal representations should likewise be interpreted as approximations. Importantly, deep generative models enable researchers to directly assess the quality of these approximate representations by visually inspecting how accurately the unstructured data are reconstructed.

Fig. S1 presents a directed acyclic graph (DAG) that summarizes the data generating process and assumptions described above. In this DAG, an arrow with double red-colored lines represents a deterministic causal relation while an arrow with a single line represents a stochastic causal relation. The dashed line bidirectional arrow represents the potential independence relation.

A.2. Identification. Given this setup, we establish the nonparametric identification of the mean of the potential outcome under the treatment condition $T_i = t$ for any given $t \in \mathcal{T}$. We extend the results of (4) to the case of unstructured objects as confounder, accommodating potentially non-binary discrete treatments and observed structured confounders.

Specifically, we show that there exists a deconfounder function $\boldsymbol{f}(\boldsymbol{R}_i)$ satisfying the mean independence relation $\mathbb{E}[Y_i \mid \boldsymbol{R}_i, T_i, \boldsymbol{Z}_i, \boldsymbol{f}(\boldsymbol{R}_i)] = \mathbb{E}[Y_i \mid T_i, \boldsymbol{Z}_i, \boldsymbol{f}(\boldsymbol{R}_i)]$, where $\boldsymbol{f}(\boldsymbol{R}_i)$ is a deterministic function of the internal representation $\boldsymbol{R}_i$. The existence of such a function is guaranteed by Assumptions 2 and 3, since $\boldsymbol{U}_i$ is a deterministic function of $\boldsymbol{R}_i$ and satisfies the mean independence condition. Moreover, any deconfounder function that satisfies the same mean independence relation leads to the same identification formula. The proof is given in Section 5.A.

Proposition 1. (Nonparametric Identification) *Under Assumptions 1–3, there exists a deconfounder function* $\boldsymbol{f} : \mathbb{R}^d \mapsto \mathbb{R}^q$ *with* $q \leq d$ *that satisfies the following mean independence relation:*

$$\mathbb{E}[Y_i \mid \boldsymbol{R}_i, T_i, \boldsymbol{Z}_i, \boldsymbol{f}(\boldsymbol{R}_i)] = \mathbb{E}[Y_i \mid T_i, \boldsymbol{Z}_i, \boldsymbol{f}(\boldsymbol{R}_i)]. \quad [4]$$

By adjusting for this deconfounder, we can nonparametrically identify the mean potential outcome under the treatment condition $T_i = t$ *as,*

$$\mathbb{E}[Y_i(t)] = \int_{\mathcal{R}\times\mathcal{Z}} \mathbb{E}\big[Y_i \mid T_i = t, \boldsymbol{Z}_i, \boldsymbol{f}(\boldsymbol{R}_i)\big] dF(\boldsymbol{Z}_i, \boldsymbol{R}_i). \quad [5]$$

A.3. Estimation and Inference. Given the identification formula presented in Proposition 1, we now turn to estimation and statistical inference. We extend the estimation procedure of (4) to the current setting. Our strategy relies on two key observations. First, Assumption 2 implies that the deconfounder should not vary across the treatment conditions. Second, the deconfounder must satisfy the conditional independence stated in Eq. (4).

We use a neural network architecture that extends TarNet (5) to non-binary treatment and structured confounding variables to estimate the conditional potential outcome function given the deconfounder and observed control variables, i.e.,

$$\mu_t(\boldsymbol{f}(\boldsymbol{R}_i), \boldsymbol{Z}_i) = \mathbb{E}[Y_i(t) \mid \boldsymbol{f}(\boldsymbol{R}_i), \boldsymbol{Z}_i].$$

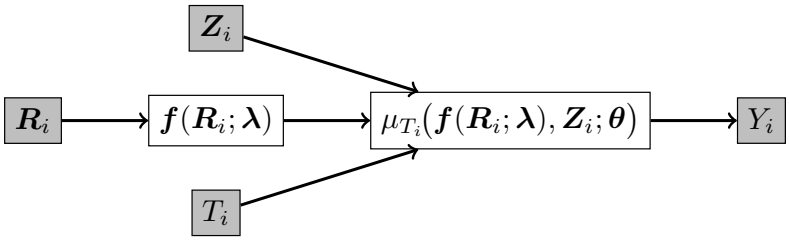

Fig. S2. Proposed Model Architecture with Unstructured Object as Confounder. The proposed model takes an internal representation of unstructured object $\boldsymbol{R}_i$ as an input, and estimates a deconfounder $\boldsymbol{f}(\boldsymbol{R}_i)$, which is a lower-dimensional representation of $\boldsymbol{R}_i$, and then uses it together with the treatment T_i and structured confounder $\boldsymbol{Z}_i$ to predict the conditional expectation of outcome $\mu_{T_i}(\boldsymbol{f}(\boldsymbol{R}_i), \boldsymbol{Z}_i) := \mathbb{E}[Y_i \mid T_i, \boldsymbol{f}(\boldsymbol{R}_i), \boldsymbol{Z}_i]$. The architecture directly encodes the mean independence relation $\mathbb{E}[Y_i \mid \boldsymbol{R}_i, T_i, \boldsymbol{Z}_i, \boldsymbol{f}(\boldsymbol{R}_i)] = \mathbb{E}[Y_i \mid T_i, \boldsymbol{Z}_i, \boldsymbol{f}(\boldsymbol{R}_i)]$.

Our proposed neural network architecture, which is summarized in Fig. S2, simultaneously estimates the deconfounder $\boldsymbol{f}(\boldsymbol{R}_i)$ and the conditional potential outcome function $\mu_{T_i}(\boldsymbol{f}(\boldsymbol{R}_i), \boldsymbol{Z}_i) := \mathbb{E}[Y_i \mid T_i, \boldsymbol{f}(\boldsymbol{R}_i), \boldsymbol{Z}_i]$, by first projecting the internal representation $\boldsymbol{R}_i$ onto a lower dimensional space of deconfounder $\boldsymbol{f}$. We then use the deconfounder, treatment T_i, and observed structured confounders $\boldsymbol{Z}_i$ to predict the conditional potential outcome function μ_{T_i} by minimizing the following loss function:

$$\{\hat{\boldsymbol{\lambda}}, \hat{\boldsymbol{\theta}}\} = \underset{\boldsymbol{\lambda},\boldsymbol{\theta}}{\arg\min} \frac{1}{N} \sum_{i=1}^{N} \{Y_i - \mu_{T_i}(\boldsymbol{f}(\boldsymbol{R}_i;\boldsymbol{\lambda}), \boldsymbol{Z}_i;\boldsymbol{\theta})\}^2, \quad [6]$$

where $\boldsymbol{\lambda}$ is the parameter vector of deconfounder function $\boldsymbol{f}(\boldsymbol{R}_i)$, and $\boldsymbol{\theta}$ is the parameter vector of conditional potential outcome function $\mu_{T_i}(\boldsymbol{f}(\boldsymbol{R}_i), \boldsymbol{Z}_i)$. We prevent the estimated deconfounder from varying across the treatment levels by using the shared representation $\boldsymbol{f}(\boldsymbol{R}_i)$ across treatment levels.

Importantly, GPI differs from the existing methods of causal inference for texts such as (3) that directly predict treatment assignment using text representation. Doing so often leads to the violation of the overlap assumption because each text is unique and can perfectly predict treatment assignment. Under this standard approach, adding a prediction head for the treatment assignment encourages the deconfounder to capture nearly all the information predictive of treatment, thereby exacerbating the violation of overlap assumption.

GPI avoids this problem by optimizing only the outcome model for the best out-of-sample prediction, discarding the information that is predictive only of the treatment. While the optimization problem in Eq. (6) does not explicitly penalize overlap violations, we address them by learning a low-dimensional representation (i.e., deconfounder), which is only predictive of the outcome, through dimensionality reduction in the neural network architecture and careful hyperparameter tuning (see SI Section 1 for a simulation study that illustrates this point).

Given the proposed neural network architecture, we estimate the Average Treatment Effect for the Treated (ATT) using double machine learning (DML) (1). We estimate ATT to compare our result with the original analysis based on text matching. Here, we propose to estimate the propensity score model as a function of the estimated deconfounder, i.e., $\pi_t(\boldsymbol{f}(\boldsymbol{R}_i), \boldsymbol{Z}_i) = \mathbb{P}(T_i = t \mid \boldsymbol{f}(\boldsymbol{R}_i), \boldsymbol{Z}_i)$ so that we do not directly condition on the high-dimensional unstructured objects.

To ensure the asymptotic normality discussed later in this section, the propensity score model must be Lipschitz-continuous; accordingly, we employ a neural network with spectral normalization (6). Practical implementation requires careful tuning of the hyperparameters for the deconfounder and the outcome models, including network width and depth, learning rate, batch size, dropout rate, and number of epochs. We automate this with advanced hyperparameter-optimization tools such as Optuna (15).

As an example, we present the entire estimation procedure in the case of binary treatment $\mathcal{T} = \{0, 1\}$. Denote the observed data by $\mathcal{D} := \{\mathcal{D}_i\}_{i=1}^N$ where $\mathcal{D}_i := \{Y_i, T_i, \boldsymbol{R}_i, \boldsymbol{Z}_i\}$. We use the following K-fold cross-fitting procedure, assuming that N is divisible by K.

1. Randomly partition the data into K folds of equal size where the size of each fold is $n = N/K$. The observation index is denoted by $I(i) \in \{1, \ldots, K\}$ where $I(i) = k$ implies that the ith observation belongs to the kth fold.

2. For each fold $k \in \{1, \cdots, K\}$, use observations with $I(i) \neq k$ as training data:

   (a) further split the training data into two folds, $I_1^{(-k)}$ and $I_2^{(-k)}$

   (b) simultaneously obtain an estimated deconfounder and an estimated conditional outcome function on the first fold, which are denoted by $\hat{\boldsymbol{f}}^{(-k)}(\{\boldsymbol{R}_i, \boldsymbol{Z}_i\}_{i \in I_1^{(-k)}}) := \boldsymbol{f}(\{\boldsymbol{R}_i, \boldsymbol{Z}_i\}_{i \in I_1^{(-k)}}; \hat{\boldsymbol{\lambda}}^{(-k)})$ and $\hat{\mu}_t^{(-k)}(\{\boldsymbol{R}_i, \boldsymbol{Z}_i\}_{i \in I_1^{(-k)}}) := \mu_t(\boldsymbol{f}(\{\boldsymbol{R}_i\}_{i \in I_1^{(-k)}}, \boldsymbol{Z}_i; \hat{\boldsymbol{\lambda}}^{(-k)}); \hat{\boldsymbol{\theta}}^{(-k)})$, respectively, by solving the optimization problem given in Equation Eq. (6), and

   (c) obtain an estimated propensity score given the estimated deconfounder on the second fold, which is denoted by $\hat{\pi}^{(-k)}(\hat{\boldsymbol{f}}^{(-k)}(\{\boldsymbol{R}_i\}_{i \in I_2^{(-k)}}), \boldsymbol{Z}_i) := \hat{\pi}^{(-k)}(\boldsymbol{f}(\{\boldsymbol{R}_i\}_{i \in I_2^{(-k)}}; \hat{\boldsymbol{\lambda}}^{(-k)}), \boldsymbol{Z}_i)$.

3. Compute the ATT estimator $\hat{\tau}$ as a solution to:

$$\frac{1}{nK}\sum_{k=1}^{K}\sum_{i:I(i)=k}\psi(\mathcal{D}_i;\hat{\tau},\hat{\boldsymbol{f}}^{(-k)},\mu_0^{(-k)},\hat{\pi}^{(-k)}) \;=\; 0,$$

where

$$\psi(\mathcal{D}_i;\tau,\boldsymbol{f},\mu_0,\pi) = \frac{T_i(Y_i-\mu_0(\boldsymbol{f}(\boldsymbol{R}_i),\boldsymbol{Z}_i))}{\mathbb{P}(T_i=1)} - \frac{\pi(\boldsymbol{f}(\boldsymbol{R}_i),\boldsymbol{Z}_i)(1-T_i)(Y_i-\mu_0(\boldsymbol{f}(\boldsymbol{R}_i),\boldsymbol{Z}_i))}{\mathbb{P}(T_i=1)(1-\pi(\boldsymbol{f}(\boldsymbol{R}_i),\boldsymbol{Z}_i))} - \frac{T_i\tau}{\mathbb{P}(T_i=1)}. \quad [7]$$

Finally, we establish the asymptotic properties of the proposed estimator in the case of binary treatment. We assume that the following regularity conditions hold. These conditions are similar to those in (4) except that we also condition on the observed covariates $\boldsymbol{Z}_i$.

Assumption 4. (Regularity Conditions) *Let c_1, c_2, and $q>2$ be positive constants and δ_n be a sequence of positive constants approaching zero as the sample size n increases. Then, the following conditions hold.*

(a) (Primitive conditions)

$$\mathbb{E}[|Y_i|^q]^{1/q}\le c_1, \quad \sup_{\boldsymbol{r}\in\mathcal{R},\boldsymbol{z}\in\mathcal{Z}}\mathbb{E}[|Y_i-\mu_{T_i}(\boldsymbol{f}(\boldsymbol{r}),\boldsymbol{z})|^2 \mid \boldsymbol{R}_i=\boldsymbol{r},\boldsymbol{Z}_i=\boldsymbol{z}]\le c_1,$$
$$\mathbb{E}[|Y_i-\mu_{T_i}(\boldsymbol{f}(\boldsymbol{R}_i),\boldsymbol{Z}_i)|^2]^{1/2}\ge c_2.$$

(b) (Outcome model estimation)

$$\mathbb{E}[|\hat{\mu}_{T_i}(\hat{\boldsymbol{f}}(\boldsymbol{R}_i),\boldsymbol{Z}_i)-\mu_{T_i}(\boldsymbol{f}(\boldsymbol{R}_i),\boldsymbol{Z}_i)|^q]^{1/q}\le c_1, \quad \mathbb{E}[|\hat{\mu}_{T_i}(\hat{\boldsymbol{f}}(\boldsymbol{R}_i),\boldsymbol{Z}_i)-\mu_{T_i}(\boldsymbol{f}(\boldsymbol{R}_i),\boldsymbol{Z}_i)|^2]^{1/2}\le \delta_n n^{-1/4}.$$

(c) (Deconfounder estimation)

$$\mathbb{E}\left[\|\hat{\boldsymbol{f}}(\boldsymbol{R}_i)-\boldsymbol{f}(\boldsymbol{R}_i)\|^q\right]^{1/q}\le c_1, \quad \mathbb{E}\left[\|\hat{\boldsymbol{f}}(\boldsymbol{R}_i)-\boldsymbol{f}(\boldsymbol{R}_i)\|^2\right]^{1/2}\le \delta_n n^{-1/4}$$

(d) (Propensity score estimation) $\pi(\cdot)$ is Lipschitz continuous at the every point of its support, and satisfies:

$$\mathbb{E}[|\hat{\pi}(\boldsymbol{f}(\boldsymbol{R}_i),\boldsymbol{Z}_i)-\pi(\boldsymbol{f}(\boldsymbol{R}_i),\boldsymbol{Z}_i)|^q]^{1/q}\le c_1, \quad \mathbb{E}[|\hat{\pi}(\boldsymbol{f}(\boldsymbol{R}_i),\boldsymbol{Z}_i)-\pi(\boldsymbol{f}(\boldsymbol{R}_i),\boldsymbol{Z}_i)|^2]^{1/2}\le \delta_n n^{-1/4}.$$

These regularity conditions are standard in the DML literature, and it is known that the neural network architecture with an appropriate depth and width can achieve the required convergence rates (16). The propensity score requires the Lipschitz continuity condition, which can be satisfied by using a regularized neural network architecture (6).

Given the above assumptions, the asymptotic normality of the proposed estimator follows immediately from the DML theory.

Theorem 1. (Asymptotic Normality of the ATT Estimator) *Under Assumptions 1–4, the estimator $\hat{\tau}$ obtained from the influence function ψ satisfies asymptotic normality:*

$$\frac{\sqrt{N}(\hat{\tau}-\tau)}{\sigma}\xrightarrow{d}\mathcal{N}(0,1)$$

where $\sigma^2=\mathbb{E}[\psi(\mathcal{D}_i;\tau,\boldsymbol{f},\mu_0,\pi)^2]$.

We omit the proof as it is identical to that of Theorem 1 given in (4).

When we have internal representations from different GenAI models, we can combine the estimates based on them to improve statistical efficiency. The following proposition establishes the asymptotic normality of such combined estimator in the case of the two GenAI models. The extension to more than two GenAI models is straightforward.

Proposition 2. (Asymptotic Normality of the Combined Estimator) *Suppose that we have two different GenAI models that give two different internal representations. Let $\mathcal{D}_{ij}=\{Y_i,T_i,\boldsymbol{R}_{ij},\boldsymbol{Z}_i\}$ be the observed data obtained from the jth GenAI model for $j=1,2$, where $\boldsymbol{R}_{ij}$ is the internal representation obtained from the jth GenAI model. Let $\hat{\tau}_j$ be the ATT estimators obtained from the jth GenAI model and $\psi_j(\mathcal{D}_{ij})=\psi_j(\mathcal{D}_{ij};\tau,\boldsymbol{f}_j,\mu_{1j},\mu_{0j},\pi_{1j})$ be the influence function for $\hat{\tau}_j$ for $j=1,2$, where $\boldsymbol{f}_j$, μ_{tj}, and π_{1j} are the true deconfounder, outcome model, and propensity score model for the jth GenAI model, respectively. Under Assumptions 1–4 for each GenAI model, the combined estimator $\hat{\tau}_{\text{comb}}=\omega\hat{\tau}_1+(1-\omega)\hat{\tau}_2$ for any weight ω satisfies asymptotic normality:*

$$\sqrt{N}\frac{(\hat{\tau}_{\text{comb}}-\tau)}{\sigma_{\text{comb}}}\xrightarrow{d}\mathcal{N}(0,1)$$

Model	Outcome	Learning Rate	Dropout	Outcome Model	Deconfounder
LLaMA3 (8B)	Number of Posts	1.188×10^{-7}	0.051	[100, 1]	[1024, 512]
	Rate of Censorship	9.974×10^{-5}	0.161	[100, 1]	[1024, 512]
	Rate of Missing Posts	9.858×10^{-5}	0.118	[50, 1]	[1024, 512]
Gemma3 (1B)	Number of Posts	2.726×10^{-7}	0.094	[100, 1]	[1024, 512]
	Rate of Censorship	7.678×10^{-5}	0.276	[50,1]	[1024, 512]
	Rate of Missing Posts	7.499×10^{-5}	0.154	[50,1]	[512, 256]
Qwen (0.6B)	Number of Posts	6.100×10^{-5}	0.160	[100, 1]	[1024, 512]
	Rate of Censorship	9.679×10^{-5}	0.195	[200, 1]	[256, 128]
	Rate of Missing Posts	5.970×10^{-5}	0.094	[100, 1]	[256, 128]
Sentence-BERT	Number of Posts	6.728×10^{-6}	0.066	[100, 1]	[1024, 512]
	Rate of Censorship	2.055×10^{-5}	0.276	[100,1]	[512, 256]
	Rate of Missing Posts	2.641×10^{-5}	0.071	[50,1]	[512, 256]

Table S2. Optimal hyperparameters selected for each outcome measure using Optuna. The table lists the learning rate, dropout rate, outcome model architecture, and deconfounder architecture applied to the reanalysis of (11).

where $\sigma^2_{\text{comb}} = \mathbb{E}[(\omega\psi_1(\mathcal{D}_{i1}) + (1-\omega)\psi_2(\mathcal{D}_{i2}))^2]$. *Moreover, the combined estimator achieves the minimum asymptotic variance at the optimal weight*

$$\omega^* = \frac{\mathbb{E}[\psi_2(\mathcal{D}_{i2})^2] - \mathbb{E}[\psi_1(\mathcal{D}_{i1})\psi_2(\mathcal{D}_{i2})]}{\mathbb{E}[\psi_1(\mathcal{D}_{i1})^2] + \mathbb{E}[\psi_2(\mathcal{D}_{i2})^2] - 2\mathbb{E}[\psi_1(\mathcal{D}_{i1})\psi_2(\mathcal{D}_{i2})]}.$$

See SI Section 5.B for the proof. The combined estimator can substantially reduce variance when the correlation between the two influence functions is low, indicating that the different representations provide complementary information about the treatment effect.

B. Implementation Details for the Empirical Application. We use the dataset analyzed by (11), which is based on the data from the Weiboscope project (17). This project collected Weibo posts in real time and subsequently revisited them to determine whether they had been censored. The original analysis focused on users who experienced at least one instance of censorship during the first half of 2012, and examined their subsequent posts in the second half of that year. To construct the control group, the authors selected posts that had a cosine similarity greater than 0.5 to a censored post and were posted on the same day. Posts shorter than 15 characters were excluded. The final dataset we analyze consists of 75,324 posts from 4,155 Weibo users.

Based on the treatment and control "focal" posts, three outcome variables were constructed:

1. The number of posts made by the same user within the four weeks following the focal post
2. The proportion of those posts that were censored (as indicated by a "permission denied" message)
3. The proportion of posts that went missing (as indicated by a "Weibo does not exist" message)

It is important to note that the second and third outcomes capture different types of post removals. As noted by (17), a "permission denied" message is a clear signal of censorship, whereas "Weibo does not exist" may reflect either censorship or voluntary deletion by the user. To adjust for users' baseline behavior, the same three measures were also computed for the four-week period preceding each focal post and included as confounding variables.

We apply the GPI methodology. First, we use LLaMA3 (eight billion parameters) and Gemma3 (one billion parameters) to reproduce all posts in the dataset and extract the internal representation of the last token. Due to the autoregressive nature of generative models, this final-token representation captures most of the information of each post (18–20). The resulting internal representation, denoted by $\boldsymbol{R}_i$, has a dimensionality of 4,096 for LLaMA3 and 1,152 for Gemma3. Using these representations, we estimate the deconfounder $\boldsymbol{f}(\boldsymbol{R}_i)$ and the outcome model $\mu_{T_i}(\boldsymbol{f}(\boldsymbol{R}_i), \boldsymbol{Z}_i)$ via a neural network architecture described in the previous section.

To train the model for each outcome variable, we employed two-fold cross-validation on the full sample, following the recommendation of (21). The hyperparameter search space includes:

- Learning rate: $[10^{-7}, 10^{-4}]$
- Dropout rate: $[0.05, 0.3]$
- Outcome model architecture: single-layer MLP with ReLU activation and 50, 100, or 200 hidden units
- Deconfounder architecture: two-layer MLP with ReLU activation and hidden unit configurations of (256, 128), (512, 256), or (1024, 512)

We automated hyperparameter optimization using Optuna (15), selecting the best parameters based on validation loss across 100 trials. For both hyperparameter tuning and nuisance function estimation, models were trained for up to 10,000 epochs, with early stopping triggered if validation loss failed to improve for five consecutive epochs. We used the Adam optimizer with a batch size of 256 and a weight decay of 10^{-8} to prevent overfitting. To stabilize training, we applied gradient clipping with a maximum norm of 1.0. For the sake of completeness, Table S2 reports the optimal hyperparameter configurations for each outcome measure.

After selecting the optimal hyperparameters, we estimated the ATT for each outcome using the DML procedure described in the previous section. Specifically, we implemented two-fold cross-fitting to estimate the nuisance components: the deconfounder $\boldsymbol{f}(\boldsymbol{R}_i)$, the outcome model $\mu_{T_i}(\boldsymbol{f}(\boldsymbol{R}_i), \boldsymbol{Z}_i)$, and the propensity score $\pi(\boldsymbol{f}(\boldsymbol{R}_i), \boldsymbol{Z}_i)$.

The deconfounder and outcome model were trained using the selected hyperparameters for up to 10,000 epochs, with early stopping triggered if the validation loss did not improve for five consecutive epochs. For propensity score estimation, we used a neural network with spectral normalization (6) to enforce Lipschitz continuity. This network consisted of a two-layer MLP with ReLU activation, using 128 hidden units in the first layer and 64 in the second. The learning rate was fixed at 2×10^{-5}, and the model was trained under the same early stopping criteria.

The ATT was then computed using Eq. (3), with standard errors derived from the influence function given in Eq. (7). To account for within-user correlation, we clustered standard errors at the user level. For consistency with the original study's text matching approach, we estimated the ATT using both the full sample and the matched sample from the original analysis. We also consider the optimal combinations of the estimates from the two LLMs using the optimal weights in Theorem 2 so that we can have more efficient results.

For comparison, we replicated the text matching procedure proposed by (11), following their publicly available replication materials. Specifically, we fit a structural topic model with 100 topics, incorporating the censorship indicator (treatment) as a covariate. From this model, we obtained both the estimated topic proportions and treatment projections for each post.

Next, we applied coarsened exact matching (CEM) to pair censored posts with uncensored ones based on four criteria: (1) topic proportions, (2) treatment projection, (3) post date, and (4) prior censorship history. This matching procedure yielded a final sample of 879 posts from 628 users. Using the matched sample and weights generated by CEM, we compared treatment and control posts across the three outcome measures described earlier. As in the main analysis, standard errors were clustered at the user level to account for within-user correlation.

In addition, we replicated three representation-based approaches for comparison. First, following **(author?)** (3, 7), we fine-tuned a pre-trained language model to construct text representations for causal inference. Specifically, we used BERT embeddings (22) as the initial text representation and jointly optimized predictive losses for the treatment assignment and outcome. Once the outcome model was estimated, following (7), we fit the propensity score models conditional on the estimated outcome model under each treatment arm, and finally estimated the treatment effect using the influence function given in Eq. (7). Because fine-tuning BERT is computationally expensive, we adopted the default hyperparameters from (7), except for the learning rate, and trained the model for 30 epochs. We set the learning rate to 0.001 to accelerate convergence, so that the relatively small number of training epochs would not materially affect the results.

Second, we used the same neural network architecture as in Fig. S2, but replaced the input representations with alternative pre-trained language model representations. Specifically, we used the multilingual sentence embedding model `paraphrase-multilingual-MiniLM-L12-v2` from the `sentence-transformers` library, which produces 384-dimensional representations. We also used the Qwen3 embedding model with 0.6 billion parameters, which produces 1024-dimensional representations; this model is based on a large language model fine-tuned for embedding tasks such as information retrieval and semantic similarity. Finally, we used the same pre-trained embedding representations from Qwen3 and Sentence-BERT but used the DragonNet (2), which added the treatment prediction head based on the shared representation of the neural network in Fig. S2 while minimizing the loss function in Eq. (3).

Unlike the internal representations used in GPI, these embeddings are not designed to exactly reconstruct the input text, which makes them cheaper to compute but may discard information relevant to downstream causal estimation. By comparing these results with the original GPI estimates, we can assess whether the performance gains arise from better internal representations. For each pre-trained embedding, we fine-tuned using the same procedure as GPI and used the exact same architecture for the sake of comparison. Table S2 shows the selected hyperparameter values.

C. Additional Empirical Results. Table S3 presents the results for the GPI estimators based on LLaMA3, Gemma3, and their optimal combination, while Table S4 reports results using alternative text representations and benchmark methods, including Qwen3–0.6B embeddings, Sentence-BERT, fine-tuned-BERT+DragonNet (3), and text matching.

The fine-tuned BERT model with DragonNet proposed by (3) produces better estimates, but these estimates are still statistically significantly different from those obtained using the GPI methods for the rate of censorship and the rate of missing posts.

In addition to the findings summarized in Section 1 of the main text, we find that the relative efficiency of GPI and text matching varies across outcomes. Although GPI yields smaller standard errors for the number of posts after censorship, text matching yields smaller standard errors for the rate of censorship and the rate of missing posts. This pattern, however, should be interpreted with caution. The text-matching estimator treats the topic-model representations as fixed and therefore does not account for the uncertainty involved in estimating these representations. As a result, its reported standard errors may understate the true uncertainty. By contrast, GPI incorporates the uncertainty from estimating the deconfounder within a

Table S3. Estimated Average Treatment Effects for the Treated (ATT) using GPI with LLaMA3 (8B), Gemma3 (1B), and their optimal combination. Standard errors (reported in parentheses) are clustered at the user level. Results are shown for both the full sample and the matched sample.

	GPI (LLaMA3-8B)		GPI (Gemma3-1B)		GPI (Combined)	
Outcome	Full	Matched	Full	Matched	Full	Matched
Rate of censorship	0.012	0.018	0.011	0.015	0.011	0.017
	(0.000)	(0.002)	(0.000)	(0.002)	(0.000)	(0.002)
Rate of missing posts	0.065	0.100	0.070	0.099	0.067	0.099
	(0.003)	(0.022)	(0.003)	(0.023)	(0.003)	(0.021)
Number of posts	-16.200	-14.700	-14.400	-4.330	-15.600	-7.690
	(2.600)	(21.900)	(2.690)	(21.300)	(2.560)	(21.100)

Table S4. Estimated Average Treatment Effects for the Treated (ATT) using alternative text representations and benchmark methods. Standard errors (reported in parentheses) are clustered at the user level. Results are shown for both the full sample and the matched sample whenever applicable.

	Qwen Embedding + TarNet		Qwen Embedding + DragonNet		Sentence-BERT + TarNet		Sentence-BERT + DragonNet		BERT (Fine-tuned) + DragonNet		Text Matching
Outcome	Full	Matched	Full	Matched	Full	Matched	Full	Matched	Full	Matched	Matched
Rate of censorship	0.010	0.018	0.007	0.009	-0.049	-0.101	0.008	0.012	0.014	-0.001	0.004
	(0.000)	(0.005)	(0.000)	(0.002)	(0.027)	(0.312)	(0.000)	(0.002)	(0.001)	(0.002)	(0.001)
Rate of missing posts	0.053	0.077	0.058	0.091	0.049	0.082	0.064	0.069	0.057	-0.030	0.050
	(0.002)	(0.026)	(0.003)	(0.022)	(0.003)	(0.030)	(0.003)	(0.026)	(0.003)	(0.021)	(0.016)
Number of posts	-16.441	9.223	-18.015	0.432	-23.000	-10.621	-23.938	-7.516	-22.788	8.635	11.500
	(2.313)	(23.648)	(2.520)	(19.179)	(2.520)	(23.545)	(2.516)	(17.791)	(3.806)	(21.758)	(41.200)

semiparametric estimation framework. At the same time, GPI tunes the representation to optimize out-of-sample outcome prediction, prioritizing features that explain outcome variation.

We also find that adding the treatment head changes the point estimates for the rate of censorship. For the Qwen embedding, the estimated effect is attenuated by 30% in the full sample and by 50% in the matched sample. The differences are statistically significant for both Qwen and Sentence-BERT. By contrast, DragonNet does not materially change the results for the other two outcomes for either pre-trained embedding model.

D. Details of the Robustness Analysis. To further assess the robustness of our approach, we examine how much the point estimates change when we additionally adjust for textual features that might act as a confounder. Specifically, we use a set of 60 keywords independently identified by (17): (i) 30 keywords with the highest relative frequency in censored posts compared to uncensored ones, and (ii) 30 keywords most frequent among self-censoring users relative to others (as reported in Appendix Tables 1a and 1b of (17)). Then, our candidate confounder is the proportion of these keywords in each post.

We estimate both the GPI and text matching methods with and without directly controlling these text features, and calculate the relative absolute bias (i.e., the absolute change in point estimates divided by the absolute value of the original point estimates). For text matching, we additionally control for these textual features by using the same structural topic model specification while incorporating the 60 keywords as controls in the coarsened exact matching (CEM). For GPI, we calculate the standard error of the relative absolute bias using the delta method, as shown below. For text matching, since no analytic variance formula is available, we estimate uncertainty using the bootstrap with 200 replications though the validity of this uncertainty estimate is not guaranteed (23).

Formally, let τ_{long} and τ_{short} be the parameter of interest obtained from GPI with and without controlling the text features, respectively, and let C_i be the text-based confounding feature for post i. Then, the relative absolute bias is defined as

$$R = \frac{|\tau_{\text{long}} - \tau_{\text{short}}|}{|\tau_{\text{short}}|}.$$

Let ψ_{long} and ψ_{short} be the influence functions of τ_{long} and τ_{short}, respectively. Because they are influence functions, we have

$$\sqrt{N}\begin{pmatrix}\hat{\tau}_{\text{long}} - \tau_{\text{long}} \\ \hat{\tau}_{\text{short}} - \tau_{\text{short}}\end{pmatrix} = \frac{1}{\sqrt{N}}\sum_{i=1}^{n}\begin{pmatrix}\psi_{\text{long}}(\mathcal{D}_i, C_i; \tau_{\text{long}}, \boldsymbol{f}_{\text{long}}, \mu_{1,\text{long}}, \mu_{0,\text{long}}, \pi_{\text{long}}) \\ \psi_{\text{short}}(\mathcal{D}_i; \tau_{\text{short}}, \boldsymbol{f}_{\text{short}}, \mu_{1,\text{short}}, \mu_{0,\text{short}}, \pi_{\text{short}})\end{pmatrix} + o_P(1)$$

where $\boldsymbol{f}_{\text{long}}$, $\mu_{t,\text{long}}$, and π_{long} are the nuisance functions estimated when controlling the text features, and $\boldsymbol{f}_{\text{short}}$, $\mu_{t,\text{short}}$, and π_{short} are those estimated without controlling the text features. By the multivariate central limit theorem, we have

$$\sqrt{N}\begin{pmatrix}\hat{\tau}_{\text{long}} - \tau_{\text{long}} \\ \hat{\tau}_{\text{short}} - \tau_{\text{short}}\end{pmatrix} \xrightarrow{d} \mathcal{N}\left(\begin{pmatrix}0 \\ 0\end{pmatrix}, \begin{pmatrix}\sigma^2_{\text{long}} & \sigma_{\text{long,short}} \\ \sigma_{\text{long,short}} & \sigma^2_{\text{short}}\end{pmatrix}\right)$$

Table S5. Correlation between estimated influence functions from GPI using LLaMA3-8B and Gemma3-1B models. Results are reported for both the full and matched samples.

	Full	Matched
Rate of censorship	0.947	0.941
Rate of missing posts	0.747	0.663
Number of Posts	0.884	0.922

Table S6. Relative absolute bias for the GPI estimators. Relative absolute bias is defined as the absolute change in point estimates from including a text-based confounder, relative to the original estimates. For the text-based confounder, we use the proportion of 60 keywords related to censorship or self-censorship. Standard errors, reported in parentheses, are calculated by the delta method. "Full" refers to estimates using the entire sample, while "Matched" refers to estimates based on the matched sample.

	GPI (LLaMA3-8B)		GPI (Gemma3-1B)	
Outcome	Full	Matched	Full	Matched
Rate of censorship	0.007	0.196	0.037	0.132
	(0.003)	(0.140)	(0.033)	(0.097)
Rate of missing posts	0.040	0.279	0.124	0.142
	(0.049)	(0.346)	(0.042)	(0.212)
Number of posts	0.084	0.242	0.062	0.370
	(0.068)	(0.682)	(0.067)	(3.638)

Table S7. Relative absolute bias for embedding-based estimators and text matching. Relative absolute bias is defined as the absolute change in point estimates from including a text-based confounder, relative to the original estimates. For the text-based confounder, we use the proportion of 60 keywords related to censorship or self-censorship. Standard errors, reported in parentheses, are calculated by the delta method for embedding-based estimators and by bootstrap for text matching, as an analytic formula is not available for text matching. "Full" refers to estimates using the entire sample, while "Matched" refers to estimates based on the matched sample.

	Qwen Embedding + TarNet		Qwen Embedding + DragonNet		Sentence-BERT + TarNet		Sentence-BERT + DragonNet		BERT (Fine-tuned) + DragonNet		Text matching
Outcome	Full	Matched	Full	Matched	Full	Matched	Full	Matched	Full	Matched	Matched
Rate of censorship	0.096	0.095	0.045	0.083	0.110	0.806	1.403	6.906	0.133	1.093	0.202
	(0.021)	(0.189)	(0.020)	(0.090)	(1.362)	(0.505)	(2.770)	(40.592)	(0.017)	(0.191)	(0.277)
Rate of missing posts	0.008	0.030	0.086	0.072	0.037	4.086	0.032	0.236	0.082	2.016	0.627
	(0.021)	(0.177)	(0.019)	(0.096)	(0.322)	(8.079)	(0.022)	(0.231)	(0.042)	(1.602)	(3.416)
Number of posts	0.095	0.467	0.047	5.957	0.022	0.068	0.089	0.032	0.799	0.900	5.350
	(0.039)	(2.291)	(0.045)	(259.396)	(0.021)	(0.476)	(0.032)	(1.934)	(0.339)	(4.429)	(33.794)

where $\sigma^2_{\text{long}} = \mathbb{E}[\psi^2_{\text{long}}]$, $\sigma^2_{\text{short}} = \mathbb{E}[\psi^2_{\text{short}}]$, and $\sigma_{\text{long,short}} = \mathbb{E}[\psi_{\text{long}}\psi_{\text{short}}]$. Then, under the assumption that $\tau_{\text{short}} \neq 0$ and $\tau_{\text{long}} - \tau_{\text{short}} \neq 0$, by the delta method, we have

$$\sqrt{n}(\hat{R} - R) \xrightarrow{d} \mathcal{N}(0, V) \quad \text{where} \quad V = \nabla g^\top \begin{pmatrix} \sigma^2_{\text{long}} & \sigma_{\text{long,short}} \\ \sigma_{\text{long,short}} & \sigma^2_{\text{short}} \end{pmatrix} \nabla g$$

with $g(x, y) = |x - y|/|y|$ and $\nabla g = (\partial g/\partial x, \partial g/\partial y)^\top$ evaluated at $(\tau_{\text{long}}, \tau_{\text{short}})$.

Table S6 reports the results for the GPI estimators, while Table S7 reports the results using alternative representations and benchmark methods. We find that the GPI methods achieve relatively small absolute bias though the results based on Qwen embedding are similar. The estimates based on Sentence-BERT with TarNet are also similar, though we see that those estimates are different from the ones from GPI as reported in Table S4. The text matching is the least robust to the inclusion of additional confounding features, suggesting that in high-dimensional settings matching methods have a difficulty in adjusting for confounding features.

3. Image as Treatment

In this section, we provide a formal theoretical justification of our methodology introduced in Section 2 of the main text and describe its implementation details.

A. Theoretical Properties. Our identification result is nearly identical to that of (4). The key difference is that we focus on image rather than text. We also derive testable implications of our key identification condition (Assumption 9). As in the previous section, we define the causal estimand of interest, establish its nonparametric identification, and then develop an appropriate estimation strategy.

A.1. Assumptions and Identification. Suppose we observe a sample of N i.i.d. units $i = 1, \ldots, N$ from a population of interest. For each unit i, we observe an unstructured treatment object $\boldsymbol{X}_i \in \mathcal{X} \subset \mathbb{R}^r$ (e.g., images or texts) and an observed outcome $Y_i \in \mathcal{Y} \subset \mathbb{R}$. As in the previous section, we assume that the treatment object $\boldsymbol{X}_i$ is generated by a deep generative model defined in Definition 1 and that the last layer of the deep generative model is a deterministic function of the internal representation $\boldsymbol{R}_i$ (Assumption 3).

As before, we adopt the potential outcome framework. We denote the potential outcome under the treatment object $\boldsymbol{x} \in \mathcal{X}$ by $Y_i(\boldsymbol{x})$, which is defined as the outcome that would be realized if unit i were to receive treatment object $\boldsymbol{x}$. The observed outcome is denoted by Y_i and is defined as the potential outcome under the realized treatment object $\boldsymbol{X}_i$, i.e., $Y_i = Y_i(\boldsymbol{X}_i)$. This notation assumes no interference between units and no hidden multiple version of treatment objects (8). In our setup, since the same annotator might code the same images multiple times, this assumption implies the absence of carry-over effects. This is formalized as follows:

Assumption 5. (Consistency) *The potential outcome under the treatment object $\boldsymbol{x} \in \mathcal{X}$ is denoted by $Y_i(\boldsymbol{x})$ and equals the observed outcome Y_i under the realized treatment object $\boldsymbol{X}_i$:*

$$Y_i \;=\; Y_i(\boldsymbol{X}_i).$$

Since we are working in the observational studies setting without randomization, we rely on the standard ignorability assumption for causal identification. Specifically, we assume that the assignment of the treatment object is independent of the potential outcomes. This is satisfied in our application if the annotators do not select which images to annotate. The assumption is formalized as follows:

Assumption 6. (Ignorability) *The assignment of treatment object $\boldsymbol{X}$ is independent of potential outcomes,*

$$Y_i(\boldsymbol{x}) \;\perp\!\!\!\perp\; \boldsymbol{X}_i,$$

where $0 < \mathbb{P}(\boldsymbol{X}_i = \boldsymbol{x}) < 1$ for all $i = 1, \ldots, N$, $\boldsymbol{x} \in \mathcal{X}$.

We are interested in estimating the causal effect of a particular feature that is a part of the treatment object. We assume that this treatment feature T is determined solely by the treatment object $\boldsymbol{X}$.

Assumption 7. (Treatment Feature) *There exists a deterministic function $g_T : \mathcal{X} \to \mathcal{T}$ that maps a treatment object $\boldsymbol{X}_i$ to a treatment feature of interest $T_i \in \mathcal{T}$, i.e.,*

$$T_i \;=\; g_T(\boldsymbol{X}_i).$$

The assumption implies that the treatment variable is a function of the treatment object and does not vary across respondents.

Next, we define confounding features, which represent all features of $\boldsymbol{X}$ other than the treatment feature T that influence the outcome Y. These confounding features, denoted by $\boldsymbol{U}_i$, are based on a vector-valued deterministic function of $\boldsymbol{X}$ and are denoted by $\boldsymbol{U} \in \mathcal{U}$ where $\mathcal{U}$ denotes their support. As in the previous section, however, this confounding feature is not observed.

Assumption 8. (Confounding Features) *There exists an unknown vector-valued deterministic function $\boldsymbol{g}_{\boldsymbol{U}} : \mathcal{X} \to \mathcal{U}$ that maps an unstructured object $\boldsymbol{X}_i \in \mathcal{X}$ to the confounding features $\boldsymbol{U}_i \in \mathcal{U}$, i.e.,*

$$\boldsymbol{U}_i \;=\; \boldsymbol{g}_{\boldsymbol{U}}(\boldsymbol{X}_i),$$

where $\dim(\boldsymbol{U}_i) \ll \dim(\boldsymbol{X}_i)$.

We note that Assumptions 7–8 are identical to the ones presented in (4).

Finally, we introduce our key identification assumption, which states that it is possible to intervene the treatment feature without changing the confounding features. Specifically, we assume that the treatment feature cannot be represented as a deterministic function of the confounding features. In addition, the confounding features should not be a function of treatment feature either so that we only adjust for pretreatment variables (24). We formalize this assumption as follows.

Assumption 9. (Separability of Treatment and Confounding Features) *The potential outcome is a function of the treatment feature of interest t and another separate function of the confounding features $\boldsymbol{u}$. Specifically, for any given $\boldsymbol{x} \in \mathcal{X}$ and all i, we have:*

$$Y_i(\boldsymbol{x}) = Y_i(t, \boldsymbol{u}) = Y_i(g_T(\boldsymbol{x}), \boldsymbol{g}_{\boldsymbol{U}}(\boldsymbol{x})),$$

where $t = g_T(\boldsymbol{x}) \in \mathcal{T}$ and $\boldsymbol{u} = \boldsymbol{g}_{\boldsymbol{U}}(\boldsymbol{x}) \in \mathcal{U}$. In addition, g_T and $\boldsymbol{g}_{\boldsymbol{U}}$ are separable. That is, there exists no deterministic function $\tilde{g}_t : \mathcal{U} \to \{0, 1\}$, which satisfies $\mathbf{1}\{g_T(\boldsymbol{x}) = t\} = \tilde{g}_t\big(\boldsymbol{g}_{\boldsymbol{U}}(\boldsymbol{x})\big)$ for all $\boldsymbol{x} \in \mathcal{X}$ and any $t \in \mathcal{T}$. Similarly, there exist no deterministic functions $\boldsymbol{g}' : \mathcal{X} \to \mathcal{X}'$ and $\tilde{\boldsymbol{g}}_{\boldsymbol{U}} : \mathcal{T} \times \mathcal{X}' \to \mathcal{U}$ that depend nontrivially on the first argument, which satisfy $\boldsymbol{g}_{\boldsymbol{U}}(\boldsymbol{x}) = \tilde{\boldsymbol{g}}_{\boldsymbol{U}}(g_T(\boldsymbol{x}), \boldsymbol{g}'(\boldsymbol{x}))$ for all $\boldsymbol{x} \in \mathcal{X}$ and $\tilde{\boldsymbol{g}}_{\boldsymbol{U}}(t, \boldsymbol{g}'(\boldsymbol{x}')) \neq \tilde{\boldsymbol{g}}_{\boldsymbol{U}}(t', \boldsymbol{g}'(\boldsymbol{x}'))$ for some $t \neq t'$ with $t, t' \in \mathcal{T}$ and some $\boldsymbol{x}' \in \mathcal{X}$.

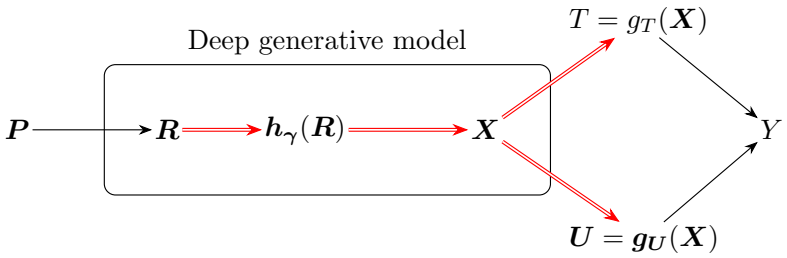

Fig. S3. Directed Acyclic Graph of the Assumed Data Generating Process when the Treatment is an Unstructured Object. A treatment object $\boldsymbol{X}$ (e.g., image) is generated using a deep generative model (rectangle) with a prompt $\mathbb{P}$, producing an internal representation $\boldsymbol{R}$ that generates $\boldsymbol{X}$ through a deterministic function $\boldsymbol{h}_{\gamma}(\boldsymbol{R})$. The treatment object affects the outcome Y through its treatment feature of interest T and other confounding features $\boldsymbol{U}$. Treatment object is assumed to be independent of potential outcomes. An arrow with red double lines represents a deterministic causal relation while an arrow with a single line indicates a possibly stochastic relationship.

We formulate the separability assumption in terms of deterministic functions rather than stochastic relationships because both the treatment and confounding features are assumed to be deterministic functions of the same unstructured data. The first requirement of separability states that the treatment feature should not be a deterministic function of the confounding features, which implies the overlap assumption (see Lemma 1 of (4)). The second requirement states that the confounding features should not be a function of the treatment feature, which corresponds to the standard assumption that confounders are pre-treatment variables. We consider $\boldsymbol{g}_{\boldsymbol{U}}(\boldsymbol{x}) = \tilde{\boldsymbol{g}}_{\boldsymbol{U}}(g_T(\boldsymbol{x}), \boldsymbol{g}'(\boldsymbol{x}))$ rather than $\boldsymbol{g}_{\boldsymbol{U}}(\boldsymbol{x}) = \tilde{\boldsymbol{g}}_{\boldsymbol{U}}(g_T(\boldsymbol{x}))$ to allow for the possibility that the confounding features $\boldsymbol{g}_{\boldsymbol{U}}$ may depend on the unstructured object $\boldsymbol{X}$ other than through the treatment feature g_T alone.

We show that this separability assumption implies the independence of support, which has appeared as an operationalization of disentanglement in the causal representation learning literature (25).

Proposition 3. (Separability implies independent support) *Under Assumption 9, for every $t \in \mathcal{T}$,*

$$\text{supp}(\boldsymbol{U}_i \mid T_i = t) = \text{supp}(\boldsymbol{U}_i),$$

where supp$(\cdot)$ *denotes the support of the corresponding random variable.*

The proof is given in SI Section 5.C. This result is analogous to Theorem 9 of (25) except that our result concerns deterministic functions, while theirs is formulated for stochastic mappings.

Under this setup, we are interested in estimating the following average causal effect of the treatment feature (ATE) while adjusting for the confounding features,

$$\tau \;:=\; \mathbb{E}[Y_i(1, \boldsymbol{U}_i) - Y_i(0, \boldsymbol{U}_i)]. \qquad [8]$$

Fig. S3 presents a DAG that summarizes the data generating process and required identification assumptions (i.e., Assumptions 3, and 5–9). As in Fig. S1, an arrow with double lines represents a deterministic causal relationship, while an arrow with a single line indicates a possibly stochastic causal relationship.

Given this setup, we establish the nonparametric identification of the mean mean of the potential outcome.

Proposition 4. (Nonparametric Identification of the Marginal Mean of Potential Outcome) *Under Assumptions 3, 5–9, there exists a deconfounder function $\boldsymbol{f} : \mathcal{R} \to \mathcal{Q} \subset \mathbb{R}^{d_Q}$ with $d_Q = \dim(\mathcal{Q}) \leq d_R = \dim(\mathcal{R})$ that satisfies the following conditional mean independence relation:*

$$\mathbb{E}[Y_i \mid \boldsymbol{R}_i, T_i, \boldsymbol{f}(\boldsymbol{R}_i)] = \mathbb{E}[Y_i \mid T_i, \boldsymbol{f}(\boldsymbol{R}_i)] \qquad [9]$$

where $0 < \mathbb{P}(T_i = t \mid \boldsymbol{f}(\boldsymbol{R}_i) = \boldsymbol{q}) < 1$ for all $t \in \mathcal{T}$ and $\boldsymbol{q} \in \mathcal{Q}$. In addition, the treatment feature and a deconfounder are separable. By adjusting for such a deconfounder, we can uniquely and nonparametrically identify the marginal mean of the potential outcome under the treatment condition $T_i = t$ for $t \in \mathcal{T}$ as:

$$\mathbb{E}[Y_i(t, \boldsymbol{U}_i)] \;=\; \int_{\mathcal{C}} \int_{\mathcal{R}} \mathbb{E}[Y_i \mid T_i = t, \boldsymbol{f}(\boldsymbol{R}_i)] dF(\boldsymbol{R}_i).$$

We omit the proof since it is similar to that of Proposition 1 of (4). We also do not present the estimation procedure details, which are almost identical to the previous application.

B. Implementation Details for the Empirical Application. We analyze the UCLA Protest Image Dataset (26), which contains 11,659 protest images in total. We focus on a subset of 9,316 images with human annotations for multiple attributes collected via Amazon Mechanical Turk (MTurk). For objective binary attributes, such as the presence of protest activities or specific visual scene characteristics, each image was independently annotated by two MTurk workers, with disagreements resolved by a third annotator. In contrast, perceived violence was measured through pairwise comparisons because it is inherently

Table S8. Hyperparameter values selected using Optuna for the image application.

Model	Learning Rate	Dropout	CNN Channels	CNN Dropout	CNN Batch Norm	Outcome Model	Deconfounder
Stable Diffusion v1.5	8.996×10^{-5}	0.276	$[32, 64, 128]$	2.903×10^{-4}	False	$[200, 1]$	$[512, 256]$
Stable Diffusion v2.1	6.109×10^{-5}	0.088	$[32, 64]$	0.008	True	$[50, 1]$	$[256, 128]$

subjective and continuous. Specifically, the original study randomly sampled 58,295 image pairs, with each image appearing in approximately 10 comparisons. For each pair, 10 MTurk workers selected the image they perceived as more violent.

The authors then used the Bradley-Terry model to estimate continuous image-level perceived violence scores, assigning each image a real-valued score between 0 and 1. See (26) for additional details about the dataset.

In our analysis, we treat nighttime (binary) as the treatment and perceived violence (continuous, ranging from 0 to 1) as the outcome. We expect the true causal effect of nighttime on perceived violence to be small because an image can, in principle, be transformed from nighttime to daytime without changing the underlying level of violence, although such a transformation may still influence how violent the image is perceived to be. Consequently, after adjusting for other image features, the estimated effect of nighttime on perceived violence should be substantially attenuated. This is because nighttime can influence the outcome only through annotators' perceptions rather than through any change in the actual level of violence depicted in the image.

We implemented our proposed methodology as follows. We first regenerated the protest images using Stable Diffusion versions 1.5 and 2.1 (27), then extracted the internal representation from the final layer of the diffusion model, immediately prior to the decoder of variational autoencoder. Consistent with Assumption 3, we verified that this decoder is a deterministic function of the internal representation. Because image dimensions vary across the dataset, we padded all images to 512×512 before passing them through the Stable Diffusion models. The resulting internal representation has dimensionality of $16{,}384 = 64 \times 64 \times 4$, which is substantially smaller than the original image dimensionality of $786{,}432 = 512 \times 512 \times 3$.

We fitted the neural network model that includes the internal representation $\boldsymbol{R}$, the treatment feature of interest (nighttime), and the outcome variable (perceived violence). Unlike the text-based application, we did not apply a pooling operation to reduce dimensionality. Instead, we used a convolutional neural network (CNN) as an encoder to further reduce the dimensionality of the internal representation and then applied the same downstream neural network architecture as in the previous application. For the CNN layers, we used kernel size 3, stride 1, padding 1, and pooling kernel size 2 for the two versions of Stable Diffusion. The tuning procedure followed the same protocol as before, with additional tuning for the CNN layers. Specifically, the hyperparameter search space for the CNN layers included:

- CNN channel architecture: either $[32, 64]$, $[64, 128]$, or $[32, 64, 128]$;
- CNN dropout: from 0.0 to 0.3;
- CNN batch normalization: either true or false.

Table S8 reports the selected optimal hyperparameters using Optuna.

To estimate the average effect of nighttime scene, we applied the methodology described earlier, using two-fold cross-fitting in combination with the selected hyperparameters to estimate both the deconfounder and outcome models. All optimization settings, including the choice of optimizer, batch size, number of training epochs, gradient clipping, and early stopping, were held consistent with the prior application. We also combined the estimates from the two different generative models (Stable Diffusion versions 1.5 and 2.1) by the optimal linear combination method discussed earlier (see Proposition 2).

4. Structural Model of Texts

In this section, we provide a formal theoretical justification for the methodology introduced in Section 3 of the main text. This application is intended to demonstrate how our GPI framework can be integrated into an existing structural model. Specifically, we tailor the proposed methodology to the experimental setting of (28), which focuses on estimating the latent persuasiveness of political rhetoric. Unlike the previous two applications, this analysis employs a model-based inferential approach. We begin by outlining the experimental design, then introduce our proposed semiparametric model, and finally detail the implementation of the empirical analysis.

A. Experimental Design. To estimate the latent persuasiveness of political rhetoric, (28) constructed a set of hypothetical political arguments and randomly assigned pairs of these arguments to survey respondents. The authors designed a total of 336 distinct arguments, which varied systematically across the following 12 policy issues in contemporary British politics.

- Building a third runway at Heathrow
- Closing large retail stores on Boxing Day
- Extending the Right to Buy
- Extension of surveillance powers in the UK

Table S9. Two examples of political rhetorics about the policy issue regarding "Building a third runway at Heathrow."

Appeal to authority / **For** The Airports Commission, an independent body established to study the issue, have argued that expanding Heathrow is "the most effective option to address the UK's aviation capacity challenge"
Appeal to history / **Against** History show us that most large infrastructure projects do not lead to significant economic growth, which suggests that the expansion of Heathrow will fail to pay for itself.

- Fracking in the UK
- Nationalization of the railways in the UK
- Quotas for women on corporate boards
- Reducing the legal restrictions on cannabis use
- Reducing university tuition fees
- Renewing Trident
- Spending 0.7% of GDP on overseas aid
- Sugar tax in the UK

In addition, the authors used the following 14 rhetorical elements:

- Ad hominem
- Appeal to authority
- Appeal to fairness
- Appeal to history
- Appeal to national greatness
- Appeal to populism
- Common sense
- Cost and benefit
- Country comparison
- Crisis
- Metaphor
- Morality
- Public opinion
- Side effects of policy

Lastly, the authors generated two versions of each argument, corresponding to the "for" and "against" positions.

Once the set of arguments was generated, they were randomly assigned to respondents. In the experiment, each respondent was presented with a pair of arguments on the same topic—one in favor and one against a given policy issue—that differed in their rhetorical elements. Respondents were asked to indicate which argument they found more persuasive, or whether they found both equally persuasive. A total of 3,317 respondents participated in the study, each evaluating four randomly selected issue pairs, resulting in 13,268 pairwise observations.

Our goal is to estimate the latent persuasiveness of each rhetorical element. The key methodological challenge is that, although argument pairs are randomly assigned to respondents, the rhetorical elements themselves may still be correlated with other features of the texts. Table S9 provides two example arguments on the same policy issue ("Building a third runway at Heathrow") used in the experiment. While both address the same topic, they differ in multiple dimensions beyond rhetorical style or stance, highlighting the potential for confounding.

To address this issue, (28) identified seven additional textual features—argument length, readability, positive and negative tone, overall emotional language, fact-based language, and whether the argument was sourced from parliamentary speeches in Hansard or authored by the researchers—and included them as additional covariates in their robustness check. However, there is no theoretical guarantee that this list of covariates fully captures all relevant sources of variation, leaving open the possibility of residual confounding.

B. Model of Latent Persuasiveness. We model the persuasiveness of the political rhetoric in the way similar to the original analysis of (28), but instead of a parametric model, we use a semiparametric model by leveraging the internal representation of the deep generative model to adjust for the argument-level confounders. Suppose that we have J arguments indexed by $j = 1, \cdots, J$ and N respondents indexed by $i = 1, \cdots, N$. Let $\boldsymbol{X}_j$ denote the text of argument j. We regenerate this text using a deep generative model as defined in Definition 1 and obtain its internal representation $\boldsymbol{R}_j$. As before, we assume that the last layer of the deep generative model is a deterministic function of the internal representation $\boldsymbol{R}_j$ (Assumption 3).

Each argument text $\boldsymbol{X}_j$ contains three features of interest: a rhetorical element $T_j \in \{1, \cdots, 14\}$, a side of the argument $S_j \in \{1, 2\}$ (for or against), and a policy issue $P_j \in \{1, \cdots, 12\}$. All features of interest are solely based on the argument text $\boldsymbol{X}_j$, and thus can be written as deterministic functions of the internal representation $\boldsymbol{R}_j$.

Each respondent i is presented with a pair of arguments $(\boldsymbol{X}_j, \boldsymbol{X}_{j'})$ on the same topic $P_j = P_{j'}$ but with the opposite sides $S_j \neq S_{j'}$ and then asked to answer the question about persuasiveness. Let $Y_{ijj'}$ denote the outcome variable, representing respondent i's answer to this question. Then, we can define the outcome as follows:

$$Y_{ijj'} = \begin{cases} 0 & \text{if respondent } i \text{ answers that argument } j \text{ is more persuasive than argument } j' \\ 1 & \text{if respondent } i \text{ answers that both arguments are equally persuasive} \\ 2 & \text{if respondent } i \text{ answers that argument } j' \text{ is more persuasive than argument } j \end{cases}$$

We use $\mathcal{J}(i)$ to represent a set of argument pairs assigned to respondent i.

In the original analysis, (28) employed a variant of the Bradley–Terry model (29) to estimate the latent persuasiveness of rhetorical elements. The model assumes the following data-generating process,

$$\log\left[\frac{\mathbb{P}(Y_{ijj'} \leq y)}{\mathbb{P}(Y_{ijj'} > y)}\right] = \delta_k + (\alpha_{P_j S_j} + \beta_{T_j} + \gamma_j) - (\alpha_{P_{j'} S_{j'}} + \beta_{T_{j'}} + \gamma_{j'}), \quad [10]$$

where $\beta_t \sim \mathcal{N}(0, \omega)$ and $\gamma_j \sim \mathcal{N}(0, \sigma_{T_j})$. Here, δ_k denotes the intercept for response category y, α_{ps} represents the fixed effect associated with policy issue $P_j = p$ and argument side $S_j = s$, β_t captures the average effect of rhetorical element $T_j = t$, and γ_j denotes the random effect for argument j with variance σ_{T_j}, which is allowed to vary across rhetorical elements.

Importantly, this model adjusts only for policy issue and argument side, and therefore does not account for other potentially confounding features of the text. In addition, the model assumes independent normal prior distributions for both β_t and γ_j. Neither assumption is implied by the experimental design. Our goal is to apply GPI in order to relax these restrictive assumptions.

Specifically, we estimate the persuasiveness of rhetorical element T_j in argument j while adjusting for confounding text features $\boldsymbol{X}_j$. As before, let $\boldsymbol{U}_j$ denote the confounding features of argument j that are not captured by the rhetorical element T_j. We assume that the confounding features $\boldsymbol{U}_j$ are a deterministic function of the argument text $\boldsymbol{X}_j$, i.e., $\boldsymbol{U}_j = \boldsymbol{g}_{\boldsymbol{U}}(\boldsymbol{X}_j)$, where $\boldsymbol{g}_{\boldsymbol{U}}$ is an unknown vector-valued function (Assumption 8). These confounding features are further assumed to be separable from the rhetorical element T_j (Assumption 9).

We model the outcome by adjusting for all the other relevant features in the texts. Specifically, we assume the following semiparametric structural model:

$$\log\left[\frac{\mathbb{P}(Y_{ijj'} \leq y)}{\mathbb{P}(Y_{ijj'} > y)}\right] = \delta_y + \beta_{T_j} - \beta_{T_{j'}} + h(\boldsymbol{R}_j) - h(\boldsymbol{R}_{j'}) \quad [11]$$

where we set $\sum_{j=1}^{J} \beta_j = 0$ and $\sum_{j=1}^{J} h(\boldsymbol{R}_j) = 0$ for identification, β_{T_j} represents the persuasiveness of rhetorical element T_j, and $h(\boldsymbol{R}_j)$ is the effect of confounding features. This model is semiparametric because we do not restrict the functional form of $h(\boldsymbol{R}_j)$ while assuming that the persuasiveness of each rhetorical element $T_j = t \in \{1, \cdots, 14\}$ additively enters the model on the log-odds scale. This additive specification is necessary because the number of unique arguments is limited (i.e., $J = 336$), making the estimation of high-dimensional interactions among textual features statistically unstable in this relatively small sample.

To estimate this model, we propose fitting the model with neural network architecture. Specifically, we use the following semiparametric structural model for estimation:

$$\log\left[\frac{\mathbb{P}(Y_{ijj'} \leq y)}{\mathbb{P}(Y_{ijj'} > y)}\right] = \delta_y + \beta_{T_j} - \beta_{T_{j'}} + h(\boldsymbol{R}_j; \boldsymbol{\theta}) - h(\boldsymbol{R}_{j'}; \boldsymbol{\theta}). \quad [12]$$

where $\boldsymbol{\theta}$ is the parameters of the persuasiveness of confounding features $h(\boldsymbol{R}_j; \boldsymbol{\theta})$, and $\sum_{j=1}^{14} \beta_j = 0$ and $\sum_{j=1}^{J} h(\boldsymbol{R}_j; \boldsymbol{\theta}) = 0$. This structural model is feasible because we observe the internal representation $\boldsymbol{R}_j$ and confounding feature $\boldsymbol{U}_j$ is a deterministic function of internal representations.

We model $h(\boldsymbol{R}_j; \boldsymbol{\theta})$ as a neural network function of $\boldsymbol{R}_j$ using the standard feed-forward neural network architecture. We minimize the following loss function to estimate the parameters:

$$\begin{aligned} \{\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\delta}}\} = \underset{\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\delta}}{\arg\min} & - \sum_{i=1}^{N} \sum_{(j,j') \in \mathcal{J}(i)} \sum_{y=0}^{1} [\mathbf{1}\{Y_{ijj'} \leq y\} \log \sigma(\delta_y + \beta_{jj'} + h_{jj'}) + \mathbf{1}\{Y_{ijj'} > y\} \log\{1 - \sigma(\delta_y + \beta_{jj'} + h_{jj'})\}] \\ \text{s.t.} & \quad \sum_{j=1}^{14} \beta_j = 0, \qquad \sum_{j=1}^{J} h(\boldsymbol{R}_j; \boldsymbol{\theta}) = 0, \end{aligned} \quad [13]$$

Table S10. Optimal hyperparameter values selected for the semiparametric structural model.

Model	$\dim(\boldsymbol{R})$	Learning Rate	Dropout	Outcome Model	Deconfounder
LLaMA3 (8B)	4096	6.073×10^{-5}	0.108	[50, 1]	[512, 256]
LLaMA3.3 (70B)	8192	2.897×10^{-5}	0.281	[50, 1]	[1024, 512]
Gemma3 (1B)	1152	9.827×10^{-5}	0.280	[50, 1]	[1024, 512]

where $\boldsymbol{\beta} = (\beta_1, \cdots, \beta_{14})$, $\boldsymbol{\delta} = (\delta_0, \delta_1)$, $\mathcal{J}(i)$ is the pair of arguments that the respondent i received, $\beta_{jj'} = \beta_{T_j} - \beta_{T_{j'}}$, $h_{jj'} = h(\boldsymbol{R}_j; \boldsymbol{\theta}) - h(\boldsymbol{R}_{j'}; \boldsymbol{\theta})$, and $\sigma(x) = 1/(1 + \exp(-x))$ is the logistic function. We quantify the uncertainty of the estimated persuasiveness parameters β_t for $t \in \{1, \cdots, 14\}$ by estimating standard errors via the bootstrap and constructing confidence intervals using the normal approximation.

C. Implementation Details of the Empirical Application. To implement our proposed methodology, we first regenerated all 336 arguments using the three different LLMs: (1) LLaMA3 with eight billion parameters, (2) LLaMA3.3 with seventy billion parameters, and (3) Gemma3 with one billion parameters. Similar to the first application, we extracted the internal representation of each argument. The dimensions of internal representation are different across models: 4096 for LLaMA3–8B, 8192 for LLaMA3.3–70B, and 1152 for Gemma3–1B.

Even though the dimensions are different, since all internal representations deterministically reproduce each argument, they should contain all the information that are needed to reproduce the texts and should yield similar estimates. More generally, the efficiency of GPI estimators is governed by two competing factors: (i) the dimensionality of the internal representation and (ii) the informativeness of that representation. More capable foundation models (e.g., LLaMA) often produce higher-dimensional representations, which can reduce statistical efficiency by increasing estimation variance. At the same time, these representations may contain richer information that improves the prediction of outcomes and nuisance functions. The resulting trade-off is therefore application-specific: in some settings, the additional information outweighs the cost of higher dimensionality, whereas in others it does not.

We fine-tuned the feed-forward neural networks within the semiparametric structural model (see Eq. (12)). We implemented the constraint in the optimization problem given in Eq. (13) by demeaning the estimated outcome model, ensuring its average is zero. For fine-tuning, we employed the same hyperparameter search space and procedure as in the previous two applications, except for the neural network architecture, and optimized the hyperparameters using Optuna. For the architecture, because the feed-forward network $\tilde{h}$ models both the deconfounder $\boldsymbol{f}$ and the partially linear component h (i.e., $\tilde{h} = h \circ \boldsymbol{f}$), we separately optimize the deconfounder and outcome-model components. For both fine-tuning and the entire structural model fitting, we used the Adam optimizer with batch size of 256, and we applied gradient clipping with a maximum norm of 1.0. Table S10 presents the optimal hyperparameters selected by Optuna. We then fitted the entire structural model using the selected hyperparameters, as described above. Once we had fitted the outcome model, we estimated the latent persuasiveness β_{T_j} and its uncertainty by estimating the standard error via the bootstrap with 200 bootstrap replications.

Fig. S4 shows that the estimates of latent persuasiveness are consistent across the three LLMs. The Pearson correlations of the estimated latent persuasiveness of rhetorical elements across models exceed 0.99 for all pairs of LLMs. These results suggest that even though the internal representations are different, the three LLMs yield similar estimates under the deterministic decoding assumption (Assumption 3).

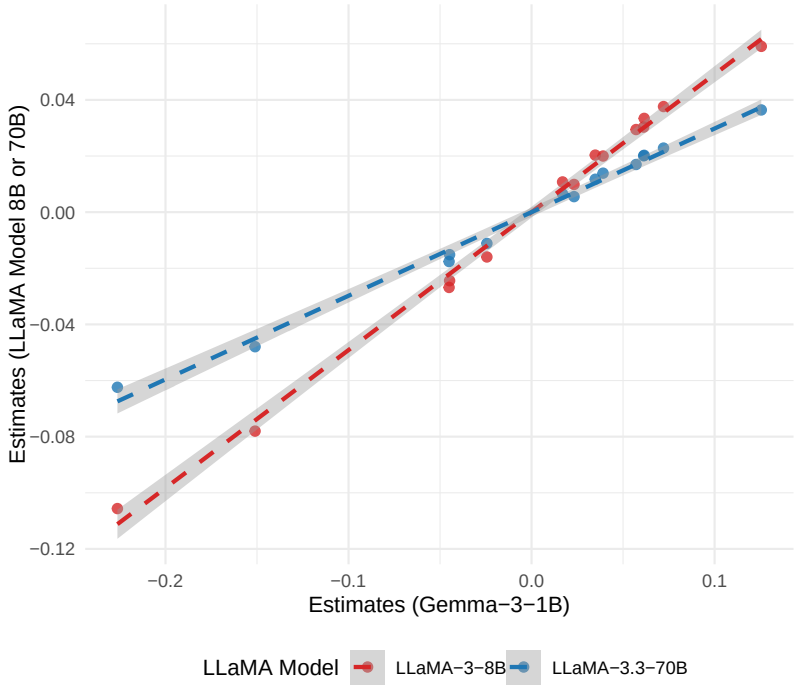

Fig. S4. The estimates of latent persuasiveness across three LLMs (LLaMA3 with eight billion parameters, LLaMA3.3 with seventy billion parameters, and Gemma3 with one billion parameters). The dashed lines represent the fitted regression lines and the gray area represents the 95% confidence interval. The results show that the estimates are highly correlated.

5. Proofs

A. Proof of Proposition 1.

Proof. Under the definition of the deep generative model (see Definition 1) and Assumption 3, we can write $\boldsymbol{U}_i = \boldsymbol{f}^*(\boldsymbol{R}_i)$ with some deterministic function $\boldsymbol{f}^*$. Then,

$$
\begin{aligned}
\mathbb{E}[Y_i(t)] &= \int_{\mathcal{Z}\times\mathcal{U}} \mathbb{E}\big[Y_i(t) \mid \boldsymbol{Z}_i, \boldsymbol{U}_i\big] dF(\boldsymbol{U}_i, \boldsymbol{Z}_i) \\
&= \int_{\mathcal{Z}\times\mathcal{U}} \mathbb{E}\big[Y_i(t) \mid T_i = t, \boldsymbol{Z}_i, \boldsymbol{U}_i\big] dF(\boldsymbol{U}_i, \boldsymbol{Z}_i) \\
&= \int_{\mathcal{Z}\times\mathcal{U}} \mathbb{E}\big[Y_i \mid T_i = t, \boldsymbol{Z}_i, \boldsymbol{U}_i\big] dF(\boldsymbol{U}_i, \boldsymbol{Z}_i) \\
&= \int_{\mathcal{Z}\times\mathcal{U}} \mathbb{E}\big[Y_i \mid T_i = t, \boldsymbol{Z}_i, \boldsymbol{f}^*(\boldsymbol{R}_i)\big] dF(\boldsymbol{f}^*(\boldsymbol{R}_i), \boldsymbol{Z}_i) \\
&= \int_{\mathcal{Z}\times\mathcal{R}} \mathbb{E}\big[Y_i \mid T_i = t, \boldsymbol{Z}_i, \boldsymbol{f}^*(\boldsymbol{R}_i)\big] dF(\boldsymbol{R}_i, \boldsymbol{Z}_i)
\end{aligned}
$$

where the second equality follows from Assumption 2, the third equality follows from Assumption 1. Finally, suppose that there exists another function $\boldsymbol{f}(\boldsymbol{R}_i)$ that satisfies $\mathbb{E}[Y_i \mid \boldsymbol{R}_i, T_i = t, \boldsymbol{Z}_i, \boldsymbol{f}(\boldsymbol{R}_i)] = \mathbb{E}[Y_i \mid T_i = t, \boldsymbol{Z}_i, \boldsymbol{f}(\boldsymbol{R}_i)]$. Then,

$$
\begin{aligned}
&\int_{\mathcal{Z}\times\mathcal{R}} \mathbb{E}\big[Y_i \mid T_i = t, \boldsymbol{Z}_i, \boldsymbol{f}^*(\boldsymbol{R}_i)\big] dF(\boldsymbol{R}_i, \boldsymbol{Z}_i) \\
&= \int_{\mathcal{Z}\times\mathcal{R}} \mathbb{E}\big[Y_i \mid T_i = t, \boldsymbol{Z}_i, \boldsymbol{f}(\boldsymbol{R}_i), \boldsymbol{R}_i\big] dF(\boldsymbol{R}_i, \boldsymbol{Z}_i) \\
&= \int_{\mathcal{Z}\times\mathcal{R}} \mathbb{E}\big[Y_i \mid T_i = t, \boldsymbol{Z}_i, \boldsymbol{f}^*(\boldsymbol{R}_i), \boldsymbol{R}_i\big] dF(\boldsymbol{R}_i, \boldsymbol{Z}_i) \\
&= \int_{\mathcal{Z}\times\mathcal{R}} \mathbb{E}\big[Y_i \mid T_i = t, \boldsymbol{Z}_i, \boldsymbol{f}^*(\boldsymbol{R}_i)\big] dF(\boldsymbol{R}_i, \boldsymbol{Z}_i)
\end{aligned}
$$

Thus, any function of $\boldsymbol{R}_i$ that satisfies the aforementioned mean independence condition leads to the same identification formula. □

B. Proof of Proposition 2.

Proof. By Theorem 1, we know that each estimator $\hat{\tau}_j$ $(j = 1, 2)$ has the influence function representation:

$$\sqrt{N}(\hat{\tau}_j - \tau) = \frac{1}{\sqrt{N}} \sum_{i=1}^{N} \psi_j(\mathcal{D}_{ij}) + o_P(1)$$

where $\psi_j(\mathcal{D}_{ij})$ is the influence function for the estimator $\hat{\tau}_j$ at the observation $\mathcal{D}_{ij}$. Therefore, for any weight ω, the estimator $\hat{\tau} = \omega\hat{\tau}_1 + (1-\omega)\hat{\tau}_2$ has the influence function representation:

$$\sqrt{N}(\hat{\tau} - \tau) = \frac{1}{\sqrt{N}} \sum_{i=1}^{N} \{\omega\psi_1(\mathcal{D}_{i1}) + (1-\omega)\psi_2(\mathcal{D}_{i2})\} + o_P(1)$$

Therefore, by central limit theorem, for any ω, we have

$$\sqrt{N}(\hat{\tau} - \tau) \xrightarrow{d} \mathcal{N}(0, V(\omega))$$

where $V(\omega) = \mathbb{E}[(\omega\psi_1(\mathcal{D}_{i1}) + (1-\omega)\psi_2(\mathcal{D}_{i2}))^2]$.

Then, since $V(\omega)$ is a quadratic function of ω, we can minimize the asymptotic variance $V(\omega)$ with respect to ω as follows,

$$\begin{aligned} V(\omega) &= \omega^2\mathbb{E}[\psi_1(\mathcal{D}_{i1})^2] + (1-\omega)^2\mathbb{E}[\psi_2(\mathcal{D}_{i2})^2] + 2\omega(1-\omega)\mathbb{E}[\psi_1(\mathcal{D}_{i1})\psi_2(\mathcal{D}_{i2})] \\ &= \{\mathbb{E}[\psi_1(\mathcal{D}_{i1})^2] + \mathbb{E}[\psi_2(\mathcal{D}_{i2})^2] - 2\mathbb{E}[\psi_1(\mathcal{D}_{i1})\psi_2(\mathcal{D}_{i2})]\}\omega^2 \\ &\quad + 2\{\mathbb{E}[\psi_1(\mathcal{D}_{i1})\psi_2(\mathcal{D}_{i2})] - \mathbb{E}[\psi_2(\mathcal{D}_{i2})^2]\}\omega + \mathbb{E}[\psi_2(\mathcal{D}_{i2})^2]. \end{aligned}$$

As it is a quadratic function of ω, it is minimized at

$$\omega^* = \frac{\mathbb{E}[\psi_2(\mathcal{D}_{i2})^2] - \mathbb{E}[\psi_1(\mathcal{D}_{i1})\psi_2(\mathcal{D}_{i2})]}{\mathbb{E}[\psi_1(\mathcal{D}_{i1})^2] + \mathbb{E}[\psi_2(\mathcal{D}_{i2})^2] - 2\mathbb{E}[\psi_1(\mathcal{D}_{i1})\psi_2(\mathcal{D}_{i2})]}$$

□

C. Proof of Proposition 3.

Proof. We prove it by contradiction. Suppose, for contradiction, that

$$\mathcal{S}_{TU} \neq \mathcal{T} \times \mathcal{U}.$$

where $S_{TU} = \{(g_T(\boldsymbol{x}), \boldsymbol{g}_{\boldsymbol{U}}(\boldsymbol{x})) : \boldsymbol{x} \in \mathcal{X}\}$ is the joint support of treatment feature and confounding feature. This means that there exists a pair $(t^\dagger, \boldsymbol{u}^\dagger) \in \mathcal{T} \times \mathcal{U}$ such that $(t^\dagger, \boldsymbol{u}^\dagger) \notin \mathcal{S}_{TU}$. Because $\boldsymbol{u}^\dagger \in \mathcal{U}$, there exists some $\boldsymbol{x}_0 \in \mathcal{X}$ satisfying $\boldsymbol{g}_{\boldsymbol{U}}(\boldsymbol{x}_0) = \boldsymbol{u}^\dagger$ and $t_0 := g_T(\boldsymbol{x}_0) \neq t^\dagger$. Now, we consider $\boldsymbol{g}'$ and $\tilde{\boldsymbol{g}}_{\boldsymbol{U}}$ as $\boldsymbol{g}'(\boldsymbol{x}) = \boldsymbol{g}_{\boldsymbol{U}}(\boldsymbol{x})$ and

$$\tilde{\boldsymbol{g}}_{\boldsymbol{U}}(t, \boldsymbol{u}) = \begin{cases} \boldsymbol{u} & \text{if } (t, \boldsymbol{u}) \neq (t^\dagger, \boldsymbol{u}^\dagger) \\ \boldsymbol{c} & \text{if } (t, \boldsymbol{u}) = (t^\dagger, \boldsymbol{u}^\dagger) \end{cases}$$

where $\boldsymbol{c} \neq \boldsymbol{u}^\dagger$. This is well-defined because $(t^\dagger, \boldsymbol{u}^\dagger) \notin \mathcal{S}_{TU}$ (so the value of $\tilde{\boldsymbol{g}}_{\boldsymbol{U}}$ at this point is unconstrained by the requirement that it reproduce the observed data; the independence of support means that we can change T and $\boldsymbol{U}$ separately). Now, by definition of $\tilde{\boldsymbol{g}}_{\boldsymbol{U}}$,

$$\tilde{\boldsymbol{g}}_{\boldsymbol{U}}(t_0, \boldsymbol{g}'(\boldsymbol{x}_0)) = \tilde{\boldsymbol{g}}_{\boldsymbol{U}}(t_0, \boldsymbol{u}^\dagger) = \boldsymbol{u}^\dagger,$$

while

$$\tilde{\boldsymbol{g}}_{\boldsymbol{U}}(t^\dagger, \boldsymbol{g}'(\boldsymbol{x}_0)) = \tilde{\boldsymbol{g}}_{\boldsymbol{U}}(t^\dagger, \boldsymbol{u}^\dagger) = \boldsymbol{c} \neq \boldsymbol{u}^\dagger.$$

Thus, $\tilde{\boldsymbol{g}}_{\boldsymbol{U}}(t, \boldsymbol{g}'(\boldsymbol{x}_0))$ changes with t while holding $g'(\boldsymbol{x}_0)$ fixed. Therefore, $\boldsymbol{U}$ can be represented as a nontrivial deterministic function of T and another feature $g'(\boldsymbol{x})$, contradicting the separability assumption. Hence no such unattainable pair $(t^\dagger, \boldsymbol{u}^\dagger)$ can exist under separability. Therefore, $\mathcal{S}_{TU} = \mathcal{T} \times \mathcal{U}$. □

References

1. V Chernozhukov, et al., Double/debiased machine learning for treatment and structural parameters. *The Econom. J.* **21**, C1–C68 (2018).
2. C Shi, D Blei, V Veitch, Adapting Neural Networks for the Estimation of Treatment Effects in *Advances in Neural Information Processing Systems*. (Curran Associates, Inc.), Vol. 32, (2019).
3. V Veitch, D Sridhar, DM Blei, Adapting Text Embeddings for Causal Inference (2020) arXiv:1905.12741 [cs, stat].
4. K Imai, K Nakamura, Causal inference with generative artificial intelligence: Application to texts as treatments. *J. Am. Stat. Assoc.* (2026) Forthcoming.
5. U Shalit, FD Johansson, D Sontag, Estimating individual treatment effect: generalization bounds and algorithms (2017) arXiv:1606.03976 [cs, stat].
6. H Gouk, E Frank, B Pfahringer, MJ Cree, Regularisation of neural networks by enforcing lipschitz continuity. *Mach. Learn.* **110**, 393–416 (2021).
7. L Gui, V Veitch, Causal Estimation for Text Data with (Apparent) Overlap Violations (2023) arXiv:2210.00079 [cs, stat].
8. DB Rubin, Comments on "On the application of probability theory to agricultural experiments. Essay on principles. Section 9" by J. Splawa-Neyman translated from the Polish and edited by D. M. Dabrowska and T. P. Speed. *Stat. Sci.* **5**, 472–480 (1990).
9. A D'Amour, P Ding, A Feller, L Lei, J Sekhon, Overlap in observational studies with high-dimensional covariates. *J. Econom.* **221**, 644–654 (2021).
10. S Klaassen, et al., DoubleMLDeep: Estimation of Causal Effects with Multimodal Data (2024) arXiv:2402.01785 [cs, econ, stat].
11. ME Roberts, BM Stewart, RA Nielsen, Adjusting for Confounding with Text Matching. *Am. J. Polit. Sci.* **64**, 887–903 (2020) _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/ajps.12526.
12. R Mozer, L Miratrix, AR Kaufman, LJ Anastasopoulos, Matching with Text Data: An Experimental Evaluation of Methods for Matching Documents and of Measuring Match Quality. *Polit. Analysis* **28**, 445–468 (2020) Publisher: Cambridge University Press.
13. C Barrie, A Palmer, A Spirling, Replication for language models problems, principles, and best practice for political science. *Work. Pap.* (2024).
14. K Nakamura, K Imai, Genai powered dynamic causal inference with unstructured data. *arXiv preprint arXiv:2605.07834* (2026).
15. T Akiba, S Sano, T Yanase, T Ohta, M Koyama, Optuna: A Next-generation Hyperparameter Optimization Framework in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19. (Association for Computing Machinery, New York, NY, USA), pp. 2623–2631 (2019).
16. MH Farrell, T Liang, S Misra, Deep Neural Networks for Estimation and Inference. *Econometrica* **89**, 181–213 (2021) _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.3982/ECTA16901.
17. Kw Fu, Ch Chan, M Chau, Assessing censorship on microblogs in china: Discriminatory keyword analysis and the real-name registration policy. *IEEE internet computing* **17**, 42–50 (2013).
18. A Neelakantan, et al., Text and code embeddings by contrastive pre-training. *arXiv preprint arXiv:2201.10005* (2022).
19. X Ma, L Wang, N Yang, F Wei, J Lin, Fine-tuning llama for multi-stage text retrieval in *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*. pp. 2421–2425 (2024).
20. T Jiang, S Huang, Z Luan, D Wang, F Zhuang, Scaling sentence embeddings with large language models. *arXiv preprint arXiv:2307.16645* (2023).
21. P Bach, O Schacht, V Chernozhukov, S Klaassen, M Spindler, Hyperparameter tuning for causal inference with double machine learning: A simulation study in *Causal Learning and Reasoning*. (PMLR), pp. 1065–1117 (2024).
22. J Devlin, MW Chang, K Lee, K Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding in *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*. pp. 4171–4186 (2019).
23. A Abadie, GW Imbens, Matching on the estimated propensity score. *Econometrica* **84**, 781–807 (2016).
24. A Daoud, CT Jerzak, R Johansson, Conceptualizing Treatment Leakage in Text-based Causal Inference (2022) arXiv:2205.00465 [cs].
25. Y Wang, MI Jordan, Desiderata for representation learning: A causal perspective. *J. Mach. Learn. Res.* **25**, 1–65 (2024).
26. D Won, ZC Steinert-Threlkeld, J Joo, Protest activity detection and perceived violence estimation from social media images in *Proceedings of the 25th ACM international conference on Multimedia*. pp. 786–794 (2017).
27. R Rombach, A Blattmann, D Lorenz, P Esser, B Ommer, High-Resolution Image Synthesis With Latent Diffusion Models. pp. 10684–10695 (2022).
28. J Blumenau, BE Lauderdale, The Variable Persuasiveness of Political Rhetoric. *Am. J. Polit. Sci.* **n/a** (2022) _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/ajps.12703.
29. RA Bradley, ME Terry, Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika* **39**, 324–345 (1952).