Conceptualizing Treatment Leakage in Text-based Causal Inference

Anonymous ACL submission

Abstract

Causal inference methods that control for text-based confounders are becoming increasingly important in the social sciences and other disciplines where text is readily available. However, these methods rely on a critical assumption that there is no treatment leakage: that is, the text contains only information about the confounder and no information about treatment assignment (leading to post-treatment bias). However, this assumption may be unrealistic in real-world situations involving text, as human language is rich and flexible.

We first define the leakage problem, discussing the identification and estimation challenges it raises. We also discuss the conditions under which leakage can be addressed by removing the treatment-related signal from the text in a pre-processing step we define as text distillation. Then, using simulation, we investigate the mechanics of treatment leakage on estimates of the average treatment effect (ATE).

1 Introduction

In observational settings, scholars need to collect information about potential confounders in order to estimate the causal effect (τ) of a treatment on an outcome. If we observed the set of confounders directly, we could condition on those quantities to recover unbiased causal effects. Yet, because some confounders U are difficult to measure directly, scholars are turning to alternative data sources, such as medical records, policy documents, or social media posts, to indirectly measure (proxy) confounders. Recent methodological frameworks supply ways of integrating high-dimensional text data into causal estimation (Mozer et al., 2020; Roberts et al., 2020; Feder et al., 2021).

However, prior literature has primarily assumed that documents only contain information about the confounder, and no information about the treatment—something we term the no-treatment-leakage assumption. When treatment leakage occurs after treatment assignment, its bias is equivalent to a post-treatment bias.

Treatment leakage leads to an identification challenge. The challenge is that W is both necessary for adjusting (as it is a proxy) yet it is also a post-treatment variable. Without treatment leakage, W would not be a post-treatment variable, as it does not harbour information about the treatment assignment. But because of leakage, scholars would have to accept bias arising from either adjusting on a post-treatment variable or bias arising from not adjusting for unobserved confounding that parts of W represents. Although several methodological studies develop and adapt causal-inference methods for text data (Keith et al., 2020), almost no studies examine the leakage bias dynamics.

Our work investigates the treatment-leakage challenge. It shows that if W is the only available text representing U and there exists a distillation method, f, that has the ability to transform (e.g., partition) W into its post-treatment W_T and proxy textual-components W_U, then adjusting on W_U is then the best one can do in identifying τ. As W_U is not post-treatment, we can adjust for it to reduce the bias when estimating τ. These f functions can represent a human annotator, identifying and removing parts of text (e.g., words, sentences) that belong to W_T and curating W_U; or, under additional assumptions, f can be based on supervised or unsupervised machine-learning machine learning models that transform the text or its representation.

In this paper, we define key assumptions and demonstrate the mechanics of text distillation in a simulated experiment. Using a language model, we generate synthetic documents W so that they contain information about the treatment assignment, T, and the unobserved confounding, U, imprinted paragraph-by-paragraph. Because we control which paragraph is affected by T (injecting post-treatment bias) or by U (infusing knowledge
We define treatment leakage as when the text, $W$, is affected by treatment status, $T$.

Treatment leakage: $\exists W_T, W_U \subseteq W$,
3 Text Distillation as Preprocessing

Text distillation is a form of text preprocessing. It has to target any text (e.g., tone, words, sentences) that belongs to $W_T$, and remove it from $W$. Thus, distillation ensures that the treatment signal is negated. As Figure 1, panel b, shows, if distillation is perfectly successful, it results in cutting the red arrow (from $T$ to $W$). The arrow is cut, because the distillation function has removed $W_T$ from $W$, supplying $W_U$ for causal analysis.

3.1 Assumptions for Valid Distillation

Distillation relies on $W$ and its key components being separable: that it can be decomposed into two portion (e.g., sets of paragraphs), where the first is only determined by $U$ and the second only by $T$. That is,

Separability Assumption: $W_U \cap W_T = \emptyset$

Assuming separability, a perfect distillator will produce $W^* = f(W)$ that is equivalent to the confounder, $W_U$. Perfect distillation means that the distillator $f$ identified text that contains the same information about $U$ as $W_U$ has. Thus, $W^* = W_U$, and if $W_U$ is a valid adjustment set, then $W^*$ is that as well. The separability assumption is appealing because it implies that researchers only need to find a valid partition of the text (and do not need to consider all possible text transformations).

This separability assumption is particularly plausible for text data, which by its nature consists of a sequence of linguistic signifiers which can be decomposed into smaller units (e.g. paragraphs).

While plausible for many circumstances, in some cases, separability may not hold, as when the entire tone of the text is affected by the treatment. In this more complicated setting, we need a more general assumption, that the transformed text, $W^{**}$, is conditionally independent of $T$ given $U$. That is, the conditional mutual information between $W^{**}$ and $T$ given $U$ is zero, while information about $U$ in $W^{**}$ is maintained. Despite the benefits of this more general framing, because $U$ is unobserved, it may be difficult for investigators to assess whether the assumption is satisfied or whether ethically problematic information has been included in the $f$ function (e.g., race; Menon and Williamson (2018)). Unlike numerical data, as text data is readable, scholars can examine and validate whether $W^*$ still contains information about $T$.

4 Experimental Setup

We use simulation to illustrate the dynamics of text distillation and build on the framework for evaluating text-based causal inference methods introduced by Wood-Doughty et al. (2021). We generate numerical covariates from the model in Figure 1; the general procedure is described in §A, with implementation details in §B. Parameters are selected so that ATE estimates $\hat{\tau}$ are biased if the estimator does not account for the unobserved confounder $U$.

Following Wood-Doughty et al. (2021), we generate documents, $W$, by sampling from an English-language GPT-2 model (Radford et al., 2019). In contrast to their approach, text generation is conditioned not only on $U$ but also on $T$. As described in detail in §A, we define paragraph-level topics, where some topics are associated with $U$, some with $T$, and some with a residual topic related only to other background variables ($R$ in Figure 1). For a given paragraph topic, we define a number of prompts and a distribution shift that increases the probability of generating topic-related keywords.

As we simulate and record which paragraphs are affected by $T$ and by $U$, our distillator $f$ has oracle properties. We can then use $f$ to investigate three idealized distillation scenarios. The first is when a distillator was not applied or the distillator failed to do any distillation $f(W) = W$. It outputs the same corpus. The second is when it perfectly distills $W$, excluding all paragraphs affected by $T$. That is, apply $f(W) = W^*$ such that $W^* = W_U$. The third scenario is when $f$ was overly aggressive and "accidentally" removed not only $T$— but also $U$-related paragraphs, resulting in $W^{**}$. This corpus violates the proxy-faithfulness assumption that
\( W^* \) fully measures \( U \). Then, we use the three corpora, one at a time, for causal inference. We use an Inverse Propensity Weighting (IPW) estimator, fully described in §C.

5 Experiments and Results

Based on the setting described in §4, our analysis produces six estimates, three based on distillation and three based on facts about the data-generating process. Figure 2 shows all estimates.

The first estimate, \( \hat{\tau}_1 = 5.5 \), is the baseline where all information is known to the outcome model, including \( U \). Because this linear model adjusting for \( U \) and \( X \) is equivalent to the data-generating model, and the estimated effect would be equal to the true value of 5 without sampling noise. The bootstrapped 95% confidence interval (CI) is 3.4 to 7.6. The second estimate, \( \hat{\tau}_2 = -2.3 \), is obtained when \( U \) is omitted from the model to induce omitted variable bias (CI: -4.2, -0.1).

The third estimate, \( \hat{\tau}_3 \), uses IPW to estimate the ATE (see §C). Here, we use the non-distilled documents, \( W_* \), to estimate propensities. As Figure 2 shows, in the absence of distillation, the bias increases compared to conditioning on \( X \) alone, producing \( \hat{\tau}_3 = -7.0 \) (CI: -9.4, -4.6). The fourth estimate, \( \hat{\tau}_4 \), applies overly aggressive distillation. This approach gives a result similar to the unadjusted estimate: \( \hat{\tau}_4 = -2.9 \) (CI: -5.1, -0.6).

The fifth estimate, \( \hat{\tau}_5 \), applies oracle distillation by removing the paragraphs we know were affected by \( T \). Using \( W^* \), the bias is reduced substantially, yielding an estimate \( \hat{\tau}_5 = 3.5 \) (CI: 1.2, 5.8). As the CI of this \( \hat{\tau} \) includes the true \( \tau = 5 \), we conclude that distillation successfully recovers \( \tau \). However, we note that this recovery is not perfect and will be affected by sampling and modeling parameters.

The sixth estimate, \( \hat{\tau}_6 \), demonstrates the impact of model selection for the propensity estimator. Using the true (simulated) propensity, the IPW estimate is \( \hat{\tau}_6 = 4.9 \) (CI: 2.2, 7.6). This result shows that further gains could be made by careful model selection (Chernozhukov et al., 2018).

Figure 3 shows distributions of propensity values for \( \hat{\tau}_3, \hat{\tau}_5, \) and \( \hat{\tau}_6 \). Without distillation (red), the estimated propensities cluster near 0 and 1. \( T \) is predicted almost perfectly, as mentioned in §2.1.1, causing the IPW estimate to be similar to the unweighted one. Conversely, with distillation, the predicted probabilities are now similar to the data-generating propensities, and thereby, the resulting causal estimate is improved.

6 Discussion

This paper shows the critical role of the no-treatment-leakage assumption when using text for causal inference. While text is becoming an established data source, it may harbour valuable information about a confounder but also contaminating information about post-treatment effects. This issue has seen little discussion in text-based causal inference literature, but has the potential to severely bias causal estimates, potentially leading to false discoveries or invalid policy recommendations.

Our study has limitations. First, more work is required to show how the no-treatment-leakage assumption operates under different covariance structures. Second, a larger simulation framework is needed to decompose estimator bias and variance.

For extensions, we recommend two paths. First, while this paper focuses on treatment leakage, there are other types of leakage when a single document is a function of multiple causal nodes. Thus, a generalization of the no-treatment-leakage assumption is the no-node-leakage assumption. Second, researchers need a framework when human partitioning of text is not possible due to corpus size. Automatic distillation could be attempted with additional assumptions, perhaps building from the literature on removing sensitive information in data (Bolukbasi et al., 2016; Ravfogel et al., 2020).
References

To summarize the general approach in this section and provide details for the simulation in §5 in the next section.

For each document $i$, we first draw observed and unobserved confounders $X_i$ and $U_i$, and then the treatment $T_i$. For each paragraph $j$ in the document, we draw a paragraph topic $Z_{ij}$, depending on the values of $U_i$ and $T_i$, and then a prompt $W_{ij}$ depending on the value of $Z_{ij}$. Finally, we sample from the GPT-2 language model to generate the text:

We first draw observed and unobserved confounders $X_i$ and $U_i$, and then the treatment $T_i$. For each paragraph $j$ in the document, we draw a paragraph topic $Z_{ij}$, depending on the values of $U_i$ and $T_i$, and then a prompt $W_{ij}$ depending on the value of $Z_{ij}$. Finally, we sample from the GPT-2 language model to generate the text:

We used the implementation from the HuggingFace repository, https://huggingface.co/gpt2.
paragraph text $W_{ij}$, starting from the prompt $W_{ij}^0$ and with a vocabulary distribution shift defined by $Z_{ij}$. Algorithm 1 shows the pseudocode.

**Algorithm 1** Generation of synthetic data.

```
for $i \in 1, \ldots, N$
    $X_i \sim f_X$
    $U_i \sim f_U$
    $T_i \sim \text{Bernoulli}(\text{sigmoid}(f_T(X_i, U_i)))$
    $Y_i \sim f_Y(X_i, U_i, T_i)$
for $j \in 1, \ldots, K$
    $Z_{ij} \sim \text{Categorical}(f_Z(U_i, T_i))$
    $W_{ij}^0 \sim \text{Categorical}(f_{W0}(Z_{ij}))$
    $W_{ij} \sim \text{LM}(W_{ij}^0, Z_{ij})$
```

In the pseudocode above, the functions $f_X$, $f_U$, $f_T$, and $f_Y$ define the distributions of the observed confounders, unobserved confounder, treatment and outcome, respectively. On the paragraph level, the function $f_Z$ defines a categorical distribution over paragraph topics, and $f_{W0}$ a categorical distribution over prompts.

Similarly to Wood-Doughty et al. (2021), we use two mechanisms to condition the generation of a paragraph on a topic $Z$: a prompt and a vocabulary distribution shift. The distribution shift is designed to promote a set of keywords related to the topic and we implement it by multiplying the language model probabilities by a topic-specific vector $\theta_Z$ of scale factors:

$$P'(w|\text{context}, Z) \propto P_{\text{LM}}(w|\text{context}) \cdot \theta_Z(w)$$

**B Parameterization Used in §5**

In §5, we generated $N = 10,000$ instances, each consisting of numerical values and a document. We used the following distributions to generate the document-level variables: $f_X$ was a 3-dimensional isotropic Gaussian; $f_U$ was an even coin toss; $f_T$ was linear in $X_i$ and $U_i$; $f_Y$ was Gaussian with a mean defined by a linear function of $X_i$, $U_i$, and $T_i$ and a fixed standard deviation.

Each document consisted of $K = 20$ paragraphs. For the paragraph generation, we defined five different topics: two corresponding to positive and negative treatment values; two corresponding to positive and negative values of the unobserved confounder; one general background topic that was unrelated to $U$ or $T$ (but conceptually thought of as controlled by other “residual” variables $R$). For a document with given values of $U$ and $T$, we set the topic distribution $f_Z$ to select the $U$ topic with a probability of 0.2, the $T$ topic with a probability of 0.2, and the general topic with a probability of 0.6.

The generated texts were designed to simulate a hypothetical use case where the researchers want to investigate the effect of IMF programs on some country-level indicator (cf. Daoud et al., 2019). The treatment variable $T$ represents the presence or absence of an IMF program; the unobserved confounder $U$ represents the political situation of the country with respect to the IMF. For each topic except the general topic, we define four different prompts: for instance, for a positive treatment value, one of the prompts was *The International Monetary Fund mandates the deregulation of [COUNTRY]’s labor market*. In the analysis, “[COUNTRY]” is substituted by randomly sampled country names.

All topics except the general topic defined a distribution shift used when generating from the language model. We used 8 topic keywords for each of these topics. For these keywords, the corresponding entries in the vocabulary distribution shift vector $\log \theta_Z$ were set to a value that defines the strength of the effect of $T$ on $W$; for all other words except these keywords, $\log \theta_Z$ was 0. Since our focus in this paper is on a clear-cut use case where the effects are strong, we set the strength parameter to a value of 4, which gives a noticeable effect on the generated texts.

The text generation model was run on a single GPU (NVIDIA GeForce GTX TITAN X), generating the 10,000 documents took around 10 hours. The generation of random text is within the intended use of the GPT-2 model.²

**C IPW Details**

**C.1 Background**

The ATE is defined as $\tau = \mathbb{E}[Y_i(1) - Y_i(0)]$, where $Y_i(t)$ is the potential outcome for unit $i$ under treatment $t$. It can be identified in randomized experiments (Rubin, 1974). However, the situation is more complicated in the observational setting, where the treatment is not randomized to units but could be correlated with confounders, $X_i$, that are associated with the treatment and the outcome. In that setting, we can, with additional assumptions, still recover the ATE using Inverse Propensity Weighting (IPW) or related robust methods (Funk et al., 2011), where observations are weighted by

²https://huggingface.co/gpt2
the inverse of their estimated treatment probabilities $\hat{\pi}(X_i) = \Pr(T_i = 1|X_i)$ (Rosenbaum and Rubin, 1983): $
abla = \frac{1}{n} \sum_{i=1}^{n} \left\{ \frac{T_i Y_i}{\hat{\pi}(X_i)} - \frac{(1-T_i)Y_i}{1-\hat{\pi}(X_i)} \right\}$.

C.2 Estimation

ATE estimates based on Inverse Propensity Weighting (see §C.1) require the estimation of the propensity scores, $\Pr(T|X, W)$. To estimate these scores, we applied a $L_1$-regularized logistic regression model using the glmnet package in R. The regularization strength ($\lambda$) was set automatically via 10-fold cross-validation. When estimating propensities, we represented the (non-distilled or distilled) document as an $L_2$-normalized TF-IDF vector using the 256 most frequent terms in the vocabulary, while the numerical covariates $X$ were standardized.