How to Make Causal Inferences Using Texts*

Naoki Egami† Christian J. Fong‡ Justin Grimmer§
Margaret E. Roberts¶ Brandon M. Stewart∥

September 22, 2017

Abstract

Texts are increasingly used to make causal inferences: either with the document serving as the treatment or the outcome. We introduce a new conceptual framework to understand all text-based causal inferences, demonstrate fundamental problems that arise when using manual or computational approaches applied to text for causal inference, and provide solutions to the problems we raise. We demonstrate that all text-based causal inferences depend upon a latent representation of the text and we provide a framework to learn the latent representation. Estimating this latent representation, however, creates new risks: we may unintentionally create a dependency across observations or create opportunities to fish for large effects. To address these risks, we introduce a train/test split framework and apply it to estimate causal effects from an experiment on immigration attitudes and a study on bureaucratic responsiveness. Our work provides a rigorous foundation for text-based causal inferences, connecting two previous disparate literatures.

---

∗We thank Edo Airoldi, Gary King, Rich Nielsen, Thomas Richardson, Alex Tahk, Fredrik Sävje, Hanna Wallach and numerous seminar participants for useful discussions about making causal inference with texts. We also thank Dustin Tingley for early conversations about potential SUTVA concerns with respect to STM and sequential experiments as a possible way to combat it.

†Ph.D. Candidate, Department of Politics, Princeton University, negami@princeton.edu
‡Ph.D. Candidate, Graduate School of Business, Stanford University, cjfong@stanford.edu
§Associate Professor, Department of Political Science, University of Chicago, JustinGrimmer.org, grimmer@uchicago.edu.
¶Assistant Professor, Department of Political Science, University of California San Diego, meroberts@ucsd.edu
∥Assistant Professor, Department of Sociology, Princeton University, brandonstewart.org, bms4@princeton.edu
Text data appears in many areas of social science and frequently within causal processes. Judicial opinions, political propaganda, and campaigning are all examples of texts written to persuade or have an effect on a subject or event. Social media posts, politicians’ responses, and lawmaking are all affected by the political context, and can sometimes be direct responses to political phenomena or stimuli. Understanding how the text influences or is influenced are natural questions for social science researchers, and social science researchers have long incorporated text data into their own experiments and research designs (Lasswell, 1938; Laver, Benoit and Garry, 2003; Pennebaker, Mehl and Niederhoffer, 2003; Quinn et al., 2010).

While text data and causal inference may be a natural combination, text analysis poses many problems for causal inference because the data are high-dimensional—text can express many different concepts and can express the same concept in many different ways. As a result, the mapping between the text and the concepts expressed in the text that are of interest to the researcher is not always known before the experiment. When we collect texts in an experimental setting we may have no idea ex ante what they will contain and what concepts they will represent, and we may need to read or analyze a selection of them to understand how they map to the quantities that we are interested in measuring.

The need to discover and iteratively define measures and concepts from data is a fundamental component of social science research (Tukey, 1980), yet it also poses problems for causal inference. Because high-dimensional data like text represents many concepts and therefore can be sliced and categorized in many different ways, this discovery process could lead researchers down what Gelman and Loken (2014) have called the garden of forking paths, where we may reach false conclusions because the measures used to test the relationship between treatment and outcome are a function of the data themselves. But without the discovery process, the researcher may be left with imprecise measures of concepts of interest or may be missing important concepts all together. Thus, the researcher struggles to have it both ways: to find and discover new and appropriate measures from the data and to avoid well-known pitfalls of causal inference.

In this paper, we formalize the tradeoff between discovery and testing in causal inference with text as treatment and outcome and provide a simple solution. We define what we call the “analyst-induced SUTVA violation” (AISV) that explains why developing the mapping between the text and categories of interest in the same set of data that is used for measuring a treatment effect violates the SUTVA assumption commonly invoked in causal inference. We propose solving this problem with a simple split of the data that allows the researcher to freely define measures of interest through exploratory analysis of one dataset, then subsequently test them on the held out data to confirm that the relationships are not an artifact of exploration. This train-test split, common to machine learning and A/B tests in industry, approximates a sequential notion of science where first a finding is discovered and in a separate experiment confirmed. We apply this process of sequential science to two applications, one where text is the outcome of a survey experiment, and one where text is the treatment affecting a response.
1 Text For Causal Inference

In this section we introduce a framework for thinking about the role of text in causal inference. Our discussion here will be general, with our experimental framework incorporating several research designs—including open-ended text responses coded into categories with hand coding, conjoint experiments, and treatments with text-based appeals. To anchor our intuition, though, we focus on our two example applications of texts and causal inference: applying the Structural Topic Model (STM) to measure the dependent variable in texts (Roberts et al., 2014), and an experiment that uses texts to understand their causal effects (Fong and Grimmer, 2016). Throughout, our discussion will focus on instances where only either the dependent variable or intervention is text based—we defer to future work situations where both the dependent and independent variable are based on texts. Roberts, Stewart and Nielsen (2017) addresses the case of text as confounder.

Across our example designs, we will denote our dependent variable for each unit \( i \) (\( i = 1, 2, \ldots, N \)) with \( Y_i \), the treatment condition for unit \( i \) will be \( T_{d[i]} \) (with \( T = (T_{d[1]}, T_{d[2]}, \ldots, T_{d[N]}) \)) and any covariates for a unit will be \( X_i \), where the index \( d[i] \) refers to the index of the document that observation \( i \) has observed.\(^1\) We define the space of all possible outcomes as \( \mathcal{Y} \) and the space of all possible treatments as \( \mathcal{T} \). Note that for our framework it is possible that either the dependent variable is a text-based response or the independent variable is a text-based treatment.

To motivate our approach to causal inference and text we first focus on a randomized experiment. First, we assume that the response depends only on the assigned treatment, often called the Stable Unit Treatment Value Assumption (SUTVA). Specifically:

**Assumption 1 (SUTVA).** For all individuals \( i \), \( Y_i(T) = Y_i(T_{d[i]}) \).

Second, we will assume that our treatment is randomly assigned:

**Assumption 2 (Ignorability/Positivity).** \( Y_i(t) \perp \perp T_{d[i]} \) and \( \Pr(T_{d[i]} = t) > 0 \) for \( t \in \mathcal{T} \).

For each observation we observe only a single potential outcome corresponding to the realized treatment.

The usual causal quantity of interest is the Average Treatment Effect (Imbens and Rubin, 2015). However, with text and other high-dimensional data it is often unclear how to determine the intervention or response of interest. This is true, in part, for conceptual reasons. We might follow a large text as data literature and treat each document as a bag of words and analyze how a particular intervention affects the use of particular words. This is well defined, but it is rarely the case that a single word (or a group of words considered independently) are the quantity of interest. Similarly, it is rarely the case that we are interested in how the presence or absence of a single word affects a response. For example, suppose we are interested in how a prior career as a lawyer affects support in an election (Bonica, Chilton and Sen, 2015). One might think that we could approximate this effect by measuring how

\(^1\)This more flexible notation enables us to explicitly allow different observations to observe the same document.
much the presence or absence of the word “lawyer” affects support. But there might be many other words or phrases that convey that someone has a legal background, such as “JD”, “attorney”, and “law school.”

Rather than individual words, social scientists are often interested in some emergent property of the text—such as the topic that is discussed, the sentiment expressed, or the ideological position taken. To return to our lawyer example, a candidate biography describing a prior career as a lawyer will likely include where the candidate received her law degree, where she works, and what kind of law she practices.

There are also statistical reasons to avoid working with texts directly. Text is high-dimensional, complicated, and sparse; implying that distinct blocks of text that convey a particular topic, sentiment, or phrase are rarely replicated exactly across documents. The sparse nature of text data implies, then, that it would be difficult (if not impossible) to make a reliable inference about how interventions affect what is said or understand how various messages affect what is written because there will be insufficient data to make an inference, even in the largest data set. Rather than work directly with text, social scientists work with a low-dimensional representation of the text, though this low-dimensional representation is often only implicit (Laver, Benoit and Garry, 2003; Grimmer, Messing and Westwood, 2012; Spirling, 2012; Catalinac, 2016).

We explicitly account for this latent representation of text when writing our causal estimand. Before considering how we obtain it, we will suppose that there is a function \( g \) that maps from the text to a set of numbers. The function \( g \) represents our coding rule and is used in both machine learning settings and when hand coding. For example, if assessing the effect of an intervention on open ended responses, the process of hand coding the open ended responses into categories represents the function \( g \). And when designing text based treatments, the function \( g \) represents the researcher’s coding of how the text corresponds to particular treatment. In general, we will suppose that \( g : Y \rightarrow Z \) when the response is text and \( g : T \rightarrow Z \) when the treatment is text. The set \( Z \) is a lower-dimensional representation of the text and can take on a variety of values depending upon the study of interest. For example, if we are hand coding our documents into two categories, then \( Z = \{0, 1\} \). If we are using STM to measure the prevalence of \( K \) topics as our dependent variable, then \( Z \) is a \( K - 1 \) dimensional simplex (Roberts et al., 2014). And if we are using texts as a treatment, we might suppose that \( Z \) is the set of \( K \) binary feature vectors, representing the presence or absence of an underlying treatment (Fong and Grimmer, 2016). There are of course numerous other types of \( g \) functions that we might use—including latent scales (Slapin and Proksch, 2008), dictionary based counts of terms (Tausczik and Pennebaker, 2010), or crowd sourced measures of content (Benoit et al., 2016).

Using the function \( g \) we are able to define a causal quantity of interest. Consider first using hand coding to code text responses as belonging to one of \( K \) categories.

---

\(^2\)Of course there are counterexamples where scholars are interested in single words/phrases only, e.g. Gill and Hall (2015). This might be particularly true in the framing literature where a single word might substantially alter the content or meaning. Other experiments might make slight alterations that vary the magnitude of what is discussed.
For each unit $i$ we will write $g(Y_i(T_i)) = z_{i,T_i}$ to be the coded text for unit $i$ under treatment condition $T_i$. When our $g$ function maps to a set of $K$ categories we will write $g(Y_i(T_i)) = z_{i,T_i,k}$ to describe the value from the function for the $k^{th}$ component. We can then define the average treatment effect for category $k$ to be:

$$\text{ATE}_k = E[g(Y_i(1))_k - g(Y_i(0))_k] = E[z_{i,1,k} - z_{i,0,k}]$$

While writing the Equation 1.1 supposes that the $g$ function already exists, determining the coding rule—the discovery of a $g$ function—represents an important, though often underappreciated, component of the experimental process. In fact, most experimental designs implicitly assume it already exists and provide little guidance on how to obtain it if it does not. Rather than ignore the function or view it as a secondary issue, it is essential to place it at the center of the experimental design as one of the most important features of a valid experiment. By explicitly acknowledging the process of discovery we can develop procedures that support it and provide methodological guidance.

Obtaining the $g$ function introduces new potential problems in the estimation of causal effects. First, obtaining $g$ creates the possibility of an *Analyst Induced SUTVA Violation* (AISV). One component of SUTVA is that each observation’s potential outcome depends on only its own treatment status and we have assumed that this is true for the texts. Once we use the texts to obtain a new $g$, we can violate SUTVA, creating a dependence across observations because the particular randomization may affect the $g$ we estimate. This violation occurs because the specific treatment vector allocated affects the $g$ that we obtain, causing dependence across all observations. This violation is induced by the analyst, which is why we call it an *Analyst* induced violation.

To formally define the analyst-induced SUTVA violation we rewrite our $g$ function as explicitly dependent on the other data. Specifically, we will write the value of the $g$ function for observation $i$ that received treatment $T_i$ as $g(Y(T_i), Y(T))$ where $Y(T)$ describes all respondents’ text-based responses and the population vector of treatments. Suppose now that we re-randomize treatment $T'$ and obtain new responses $Y(T')$ but we fix unit $i$’s treatment status such that $T_i = T'_i$. We can then write our new function as $g(Y(T'_i), Y(T'))$. We will say that analyst-level SUTVA holds if $g(Y(T'_i), Y(T')) = g(Y(T_i), Y(T)) = g(Y(T_i))$.

For analyst-level SUTVA to hold we require two kinds of stability after re-randomization: category stability and classification stability. Category stability holds when the underlying set of categories or scales the $g$ function implies. For example, suppose that we are using the Structural Topic Model (STM) to estimate a set of topics and the proportion of each document assigned to each topic from the responses to two randomized treatments $T$ and $T'$. Category stability for STM implies that the set of topics that we obtain after each randomization are identical. This can be a strong assumption and we have good reason to believe that it would not hold—if different treatments affect the responses provided, then we would expect that STM will discover different topics after different randomizations. If this occurs, then the categories have changed and therefore $g(Y(T'_i), Y(T')) \neq g(Y(T_i), Y(T))$. 
In other words, individual \(i\)'s inferred response depends on the treatment status of everyone else, creating a dependency that exists through the \(g\) function.

Classification stability can occur even if category stability is not an issue. For example, suppose that we are using a supervised classification method to infer the label for a response \(i\) for two different randomizations to our population \(T\) and \(T'\). If we fix a code book before analyzing our data then our categories will remain stable by assumption. Now suppose that we use a subset of our responses to train our classifier \(S_{\text{train}}\) and that we also use these observations to estimate the treatment effect. In this setting the inferred label for observation \(i\) differs across randomizations so long as the classifier we estimate varies across \(Y(T)\) and \(Y(T')\). And the result of this is that \(g(Y(T_i), Y(T'_i)) \neq g(Y(T_i), Y(T))\).

There are two ways to deal with the analyst-induced SUTVA violation. First, we might assume the problem away. But even if this assumption is valid, it leaves the researcher exposed to the practical problem that the same data used to estimate the \(g\) function is also used to estimate the causal effects, introducing an opportunity for fishing. A sufficient assumption that eliminates the possibility of an analyst-induced SUTVA violation is that \(g\) is stable across randomizations. This might be an attractive assumption, particularly when we have a large sample of texts or when we expect that there are only a few easy to discover categories likely to appear regardless of who is treated. Further, this assumption is trivially satisfied in any instance when \(g\) is determined before the randomization, such as when scholars simply count the number of words or design their own text-based treatments.

But even when we assume it is stable, learning \(g\) from the data creates a problem of false discovery. If researchers are free to modify \(g\) as they analyze the data, it becomes possible that changes in the function are made to create larger effects or to construct variables to yield more “statistically significant” findings. The ability to modify the code categories of either the dependent or independent variable creates an opportunity for a much more serious fishing problem than standard experimental analyses, though it is related to problems of recoding the dependent variable or ignoring some conditions in an experiment. Suppose scholars are interested in some statistic \(H(g(Y(T)), T)\) that depends on both the inferred latent variables and the treatment assignments. Fishing occurs if there is a specification search over the space of potential \(g\) function \(G\)—either to ensure that the statistics are below some arbitrary p-value threshold \(\{g : H(g(Y(T)), T) < 0.05\}\) or that the estimated effect is large \(g = \arg\min_{g \in G} H(g(Y(T))), T)\). Even if the fishing is done subconsciously (or with the best of intentions), it creates substantial bias in the estimates, implies that the reported confidence intervals fail to have their reported coverage rates, and leads to overfitting. Overfitting occurs when scholars tinker too much with the \(g\) function then they will be modeling noise in their data, rather than the systematic categories that matter for either the dependent or independent variable.

The result is that even if we assume the technical problem of the AISV away, we continue to have the practical problem that we are learning the mapping from text to categories on the same data we will use to measure the casual effects. Our preferred procedure is to explicitly separate the creation of \(g\) and the estimation of treatment effects. This avoids both the AISV and eliminates the opportunity for fishing. To do this, we randomly divide our data into a training set and a test set.
Specifically, we create a set of units in a training set $S_{\text{train}}$, $|S_{\text{train}}| = N_{\text{train}}$ and test set $S_{\text{test}}$, $|S_{\text{test}}| = N_{\text{test}}$. We use only the training set to estimate the $g$ function and then discard it. We then use the test set exclusively to estimate the causal effect.

This division between the training and test set avoids the AISV and ensures that we are not overfitting our data. It avoids the AISV in the test set because the function $g$ does not depend on the randomization in the test set, so that each test set unit’s response depends only on its assigned treatment status. To see formally that the train/test split provides both category and classification stability, note that there can only be a SUTVA violation if there is dependence across observations used to estimate the treatment effect. With that insight in mind, we will partition our treatment vector $T$ into $T_{i \in S_{\text{test}}}$ for the test-set observations and $T_{i \in S_{\text{train}}}$ for training-set observations (with analogous indices for re-randomized treatments $T'$).

With this division we can infer the dependent variable for response $i$ in the test set with the function $g(Y(T_i), Y(T_{i \in S_{\text{train}}}))$ and with $g(Y(T_i), Y(T'_{i \in S_{\text{train}}}))$ after re-randomization. It is obvious from the notation that the function $g$ does not depend on the other observations in the test-set, or that $g(Y(T_i), Y(T_{i \in S_{\text{train}}}), Y(T'_{i \in S_{\text{train}}})) = g(Y(T_i), Y(T_{i \in S_{\text{train}}}))$. This lack of dependence across observations used to estimate the treatment effects implies that there is no longer an analyst-induced SUTVA violation.

Not only does the training/test split avoid the analyst-induced SUTVA violation, it also avoids the problem of overfitting and fishing because of the division between the training and test set. Any fishing or overfitting in the training set will, by definition, not provide the same effects in the test set. By adopting the split design we explicitly allow and account for the process of discovery. By contrast the most prominent proposed solution to fishing, pre-analysis plans, implicitly assumes that discovery has already taken place, focusing instead on public signaling (Humphreys, Sanchez de la Sierra and Van der Windt, 2013). This is useful, but effectively limits the possibility of using contemporary data to find low-dimensional representations of text.

The training/test split in our research design brings the quantitative analysis of text in line with many best practices when scholars hand code documents. For example, a common practice is to run pre-tests on a subset of data in order to develop code books or to refine particular text-based treatments. Other times scholars doing handcoding in experiments will develop a code book on a subset of data and discard those data, using the code book to classify the remaining documents for analysis. Our approach extends the logic of these well established practices to instances where the function $g$ is learned using machine learning methods.

The particular $g$ that we obtain in our training set defines the categories that we will use for our analysis. Therefore, our goal is to obtain a consistent (or unbiased) estimator for the ATE (or other causal quantities of interest) assuming a particular $g$. By randomization of the treatment, a consistent estimator will be:

$$\hat{\text{ATE}} = E[g(Y_i(1))|T = 1] - E[g(Y_i(0))|T = 0] = \sum_{i \in S_{\text{test}}} \frac{I(T_i = 1)g(Y_i(1))}{\sum_{i \in S_{\text{test}}} I(T_i = 1)} - \sum_{i \in S_{\text{test}}} \frac{I(T_i = 0)g(Y_i(0))}{\sum_{i \in S_{\text{test}}} I(T_i = 0)}$$
Further, once the function \( g \) is fixed, we can obtain variance estimates using the usual variance estimators. Indeed, once we have obtained a \( g \) function we are able to deploy the usual best practices for experimental research (Gerber and Green, 2012). We leave the details of some of these practices including blocking, exploration of treatment effect heterogeneity and other common concerns to future work.

1.1 Statistical Properties of the \( g \) Function

Central to the use of texts for causal inference, then, is a function \( g \) that provides a mapping from the full text to a low-dimensional representation of the text. It might seem natural, then, to inquire about the properties of the estimator we use to obtain the \( g \) function. We can conceive of the procedure used to obtain the \( g \) function as an estimator \( G \) that maps from the data to the space of function \( \mathcal{G} \), \( G : \mathcal{T}, \mathcal{Y} \rightarrow \mathcal{G} \). If we suppose that there is some true function \( \hat{g} \) we might ask how well our estimator \( G \) performs—in large samples does the function \( g \) converge on \( \hat{g} \) and in small samples how discrepant is \( g \) compared to \( \hat{g} \)?

While it is certainly useful to conceive of the estimator \( G \), it is misguided to suppose that there is a true \( \hat{g} \) for some data set that our procedure is attempting to reveal. To see why it is not useful to suppose there is a true function \( \hat{g} \) consider a hypothetical experiment where we examine how people respond to a knock on the door and encouragement to vote. We might be immediately interested in whether respondents are more likely to express a positive tone about political participation. To investigate this, we might construct a \( g \) function that measures the tone of open-ended responses. But, we might also be interested in the topics that are discussed after receiving a mobilization push from a campaign, or whether individuals mention that feel their privacy was intruded upon. There is also large variation in the ways we might examine how the particular contents of the mobilization message might affect respondents. We might be interested in whether messages that have a positive tone are more likely to increase turnout, whether highlighting the threats from the a different political party causes an increase, or whether threatening the revelation of voter history to neighbors is the most effective method of increasing turnout.

This hypothetical example makes clear that there is no “true” application-independent function for obtaining either the dependent variable or treatment when making causal inferences from texts. Further, the act of estimating \( g \) implies that as the researcher we might be unsure about what properties we want \( g \) to have—making it particularly difficult to evaluate the estimator \textit{a priori}.

It is possible, however, to evaluate the \( g \) function after the fact. Given a particular function we can examine the latent features that are identified, the scales provided, and even the classification accuracy. The \textit{post hoc} validations after obtaining a \( g \) function is essential so that both the researcher and the reader are able to correctly interpret the underlying features (Grimmer and Stewart, 2013). As we describe below, our goal in the validation is to ensure that our interpretation of the labels that arise from the \( g \) function and the corresponding effect estimates align with the actual content of the function.

\footnote{This estimator includes any actual statistical work, but also the work of hand coders and qualitative researchers}
The $g$ function is a researcher determined function—it does not require any assumption about the particular psychological model of respondents. Given that it is a researcher constructed concept, it makes sense that until this point we have considered a $g$ function which is constant across observations. Collquially, this implies that if two documents contain the same text they will have the same latent representation. This accords with the common intuition of a manually produced codebook which we would like to apply consistently across documents. While a completely separate function $g_i$ for each individual $i$ would be intractable, we can consider estimating $g$ functions that are depend on attributes of either the document or the person reading the document. This would be appropriate in which the meaning of a text that we intend to capture is specific to some context. For example, consider coding a speech for racism. The identity of the speaker can result in the “N”-word either being the most hateful and racist utterance in the English language or a term of endearment based. This is true in numerous other circumstances: a comment at work might be respectful or disrespectful if it is to your boss or your colleague or an emoji might be appropriate or inappropriate if its in a text or a formal letter. In each of these cases we would want to allow the $g$ function to map the same text to a different space when it is in a distinct subgroup. However, in order for the estimand to remain interpretable the $g$ function must map to a space that has a common interpretation, e.g. mapping documents to a dimension of racism even if they have different words to get there. The content covariates in the Structural Topic Model provide one way to approach this problem by forming global topic-word parameters from a sparse combination of subgroup specific words, topic specific words and subgroup-topic specific words (Roberts, Stewart and Airoldi, 2016). This can be handled in other models as well by stipulating some model for how the properties of individuals affect the global parameters.

2 Text as Outcome

In the text as outcome setting we are interested in measuring the effect of a manipulated non-text treatment on a function of the existing texts such that $g : \mathcal{Y} \rightarrow \mathcal{Z}$. Once the $g$ is defined, the process is straightforward and the standard procedures of causal inference apply. However learning the $g$ function, either statistically or manually, must be done carefully in order to avoid an Analyst Induced SUTVA violation. In this section we discuss the properties that we would like in a $g$ function (2.1), how manual content analysts develop $g$ functions (2.2), how we can learn the $g$ function statistically from the data (2.3), how the train-test split avoids the AISV violation (2.4) and the tradeoffs of train-test splits (2.5). We conclude the section with implications for past and future work (2.6).

2.1 Desirable properties of the $g$ function

Before describing the different ways that we can choose a $g$ function it is useful to describe the stakes involved and the properties that we might want to see. The choice of $g$ defines the property of the text along which we are able to measure the effect of the intervention $T$. In many ways it has the same importance as choosing
the outcome to measure in an experiment. We identify four properties (in decreasing order of importance) that we see as a starting point for choosing \( g \): interpretability, theoretical interest, label fidelity, and tractability.

First and foremost, we should choose the \( g \) function to be interpretable. If we cannot communicate what we are measuring about the text we cannot claim that it is of theoretical interest or define label fidelity; at that point, tractability becomes irrelevant. What interpretability means is specific to a given set of texts and a particular research question, but our articles must communicate to the reader what our measure is capturing. The statistically learned \( g \) functions will generally try to learn bundles of words that share semantic meaning.

The \( g \) function should also capture something of theoretical interest. It would be simple and straightforward to evaluate the causal effect of a treatment on the number of words used in the document, but this will (generally) not help advance an interesting argument. Theoretical interest will obviously be application-specific, but often it is theoretical interest that draws us towards considering bundles of words. Many of the problems we raise with the \( g \) function could be resolved by simply comparing the counts of a pre-selected word, but it can be difficult to connect to a broader theory with only a single word. Instead, we typically will claim that the \( g \) function is capturing some property of the text, such as its general subject matter, that has a particular relevance to our research program.

We also want to choose our function such that there is high fidelity in the correspondence between the label we give to what the \( g \) function is capturing and what it actually encodes. It is easy to forget that when we develop a codebook that evaluates whether a given document is about the “economy” we have not actually measured that thing, but rather a surrogate function that will form the basis of our argument. This is a common exercise in the social sciences; there is always an implicit mapping between the labels we use for our variables and the reality of what our labels measure. Text analysis, and particularly unsupervised methods, clearly expose this problem by highlighting the importance of the label. While there is no objective definition of label fidelity, we think about it as minimizing the surprise that a reader would have in going from the label to reading the text.

Finally, we want the development and deployment of the \( g \) function to be tractable. In the context of manual content analysis this could mean that the codebook can be applied accurately and reliably\(^4\) by human coders and that the number of documents to be coded is feasible for the resources available. In the case of learning the \( g \) function statistically, this would imply that we have a model which is sufficiently computationally tractable that we can estimate it and that it is able to learn a useful representation with the number of documents we possess.

An important component of all of these properties is the form of the latent representation \( Z \). Manual content analysis often chooses either a categorical representation (mutually exclusive and exhaustive categories), count of an event (a non-negative integer) or check-all that apply (a number of binary features). Statistical models use these forms as well as scales (a value in \( \mathbb{R}^D \) or number in a constrained space) or mixed membership (a value on the simplex). Other representations are possible but these have historically provided the best compromise of interpretability

\(^4\)In a statistical sense we mean they have low bias and variance.
and tractability.

There is an inherent tension between the four properties particularly between theoretical interest and label fidelity. It is often tempting to assign a label to a concept that is substantially more general than what the \( g \) function actually measures. However, allowing fidelity to decline in order to bolster theoretical interest is a way to lower the odds that the finding is replicable. We view any scientific finding as established by a sequence of studies in which a given result is replicated through multiple experiments, preferably across research labs. Under this view we are incentivised to have a \( g \) function which tightly coincides with our chosen label because it increases the chances that our result can be replicated by a different research group who may measure the same concept with a slightly different mapping function.

### 2.2 Developing a \( g \) function in handcoding

Manual content analysts have been creating \( g \) functions for years. Content analysis handbooks such as Krippendorff (2004) or Neuendorf (2016) advocate the use of codebooks which are essentially a set of instructions for humans to map from texts to a latent representation; in the language of this paper, a \( g \) function. It is beyond the scope of this paper to describe how to develop a good coding scheme for manual content analysis, but our chosen properties are generally in line with the objectives.

A manual content analysis scheme has many benefits. For example, there are many features of text which humans are quite adept at detecting which computers find more challenging (e.g. humor and sarcasm) and when these are the quantity of interest a manual content analysis scheme might be preferable. Because the investigator has more freedom to choose the component of text that interests them, they arguably have greater freedom to choose features of theoretical interest (although they are still bound by the capacity of human coders).\(^5\)

Yet the manual content scheme has to come from somewhere and we rarely talk about the process of discovery. In practice, these coding schemes are developed through iteration between coding rules and the documents to be coded. In many ways this is a reasonable practice: looking at the documents let’s us assess the ‘model fit’ and helps to ensure that the coding rules are completely specified. If we are developing a coding scheme for an open-ended survey question about ‘why did you vote for Trump?’ it would be inefficient at best and impossible at worst to enumerate every possible reason without reference to the texts. Yet engaging in this iterative process and then estimating the effect on the given set of texts results in an Analyst Induced SUTVA violation because the coding scheme is now dependent on a particular randomization (more on this in the next section).

\(^5\)Indeed the aspirations of investigators exceeding the capacity of human coders has led to trenchant critiques of the content analysis enterprise by those concerned that the coding is not able to capture the nuance of the underlying text and thus inadvertently ‘invents facts’ (Biernacki, 2012).
2.3 Learning the $g$ function

In the case of manual content analysis, the $g$ function is learned through a generally unspoken process of discovery. We can also learn the $g$ function statistically from the data. There are two strategies for this: supervised methods which ultimately produces a system close to manual content analysis and unsupervised learning which enables the inductive learning of the $g$ function.

In supervised learning we devise a manual content analysis scheme that is applied to a subset of documents in the training set. The algorithm then uses this subset of hand-coded documents in order to learn the $g$ function and predict the outcomes for the test set (see Grimmer and Stewart (2013) for more details). The threat of an Analyst Induced SUTVA violation requires that we do not use the test set in the development of our coding scheme or in the coding of the training documents that will be used to learn the $g$ function.\(^6\)

We can also use unsupervised learning methods to learn the $g$ function, statistically operationalizing the process of discovery (Grimmer and King, 2011). Unsupervised methods search for a low-dimensional representation of the data that is able to accurately reconstruct the higher dimensional representation.\(^7\) These low-dimensional representations capture covariance in the data (such as by grouping words that commonly co-occur into a single ‘topic’). The most consequential choice to make in this setting is the statistical model to be used which dictates the form of the latent representation. One popular representation of texts is provided by topic models where the latent representation, $Z$, is a point on the $K − 1$ dimensional simplex (Blei, Ng and Jordan, 2003; Blei, 2012). The representation of each document has an interpretation as proportional membership in a set of $K$ topics (Airoldi et al., 2014). Our causal estimand is specified in terms of counterfactual differences in the proportion of the document devoted to a particular topic $k$. For this paper we use the Structural Topic Model (STM) which is a particular type of topic model (Roberts, Stewart and Airoldi, 2016).\(^8\) We could also chose non-topic model representations which would give rise to other forms for $Z$ including, for example, continuous variables, single categories and binary features.

While it might be tempting to make a sharp distinction between causal inference employing supervised and unsupervised inference, both are instances of our general framework. Whether doing hand coding, supervised learning, or unsupervised methods scholars are still required to specify some function $g$ that moves from the high-dimensional text to some lower-dimensional representation of the text. Certainly how we learn that $g$ function will matter—it could introduce measurement error or bias (as we detail below). But the procedure used to discover the function does not alter the same basic framework.

Because we are now explicitly representing the process of discovery, it means at the outset we do not initially know what the outcome of our study will be.

\(^6\)This includes, for example, feature selection and representation. See the discussion of cross-validation in Grimmer and Stewart (2013) or Hastie, Tibshirani and Friedman (2013).
\(^7\)In practice this high-dimensional representation is itself already a reduction of the original text such as a document-term matrix.
\(^8\)See the appendix for a summary of the model and an explanation of how to extract the $g$ function.
We must find the representation of the text that we are interested in, connect it to our theoretical interest and estimate the appropriate $g$ function for mapping documents to that construct. This creates the risk for the Analyst Induced SUTVA violation because the particular randomization of our data may affect the discovery we make and thus our potential outcomes.\footnote{We note that the Analyst Induced SUTVA problem could be resolved if we wanted to make a technical assumption that the $g$ function was stable across all randomizations. This is a fairly heroic assumption from the point of view of the model, but also requires a commitment to a particular model choice before looking at the data, which is overly restrictive from a data analysis standpoint.} This connects the Analyst Induced SUTVA problem to overfitting or what Gelman and Loken (2014) call ‘the garden of forking paths.’ Because we are engaged in a process of discovery we want to avoid places stringent limits on the capacity of the analyst to explore other possibilities.

Once we have found a particular $g$ function that has the desirable properties we discussed above, we can apply it to a new set of documents. We emphasize that conceptually this does not imply that we have to believe there is a “true” underlying representation or that we have to assume the model, rather the causal effect in terms of the $g$ function simply becomes the estimand of interest (in the same way that choosing a given outcome measure in an experiment defines how we measure the effect of treatment). If we could collect new data, we could apply the $g$ function to that data and it would be statistically no different than if we were measuring any other standard outcome.

### 2.4 Train-Test Splits

In order to promote discovery while also avoiding an Analyst induced SUTVA violation we advocate the use of a train-test split, a technique that has an increasing number of advocates in political science (Cranmer and Desmarais, 2017; Grimmer, Messing and Westwood, 2017; Ward, Greenhill and Bakke, 2010) as well as scholars in causal inference (Wager and Athey, 2017; Chernozhukov et al., 2017; Anderson and Magruder, 2017). By confining our exploration to the training set and estimating only in the test set, we avoid the Analyst Induced SUTVA concerns and align researcher incentives. The analyst can “fish” and explore as much as they like in the train set, but are incentivized to find a robust underlying pattern. This reduces concerns about overfitting and the garden of forking paths while also allowing the analyst to develop theoretically important and precise measures.

Procedurally we start by creating a random partition of our document set into two portions (generally halves although not necessarily). It is important that we don’t do any preprocessing first which requires the complete dataset. For example, text scholars often remove words that are either too rare or too frequent (Grimmer and Stewart, 2013), however doing so prior to the train-test split uses data from the test set.\footnote{See Denny and Spirling (2017) for a discussion and an approach to assess the impact of these text processing decisions.} Using only the training data we are free to specify as many models as we like in our search for the outcome of interest. Only after we have made a final decision should we move to the test set.

While the result remains internally valid regardless of what we do with the training set, there are certain ways that we might want to restrict ourselves in order to...
minimize the chance that we find something which doesn’t hold up in the test set. In
general, we find it more appealing to not examine the actual treatment effects in the
training set, as doing so might unconsciously lead us to something overfit which won’t
hold up in the test set. We instead focus on the measurement itself and whether or
not we have captured something robust and interesting about the texts themselves.
We establish this through a process of validation.

Validation is an important part of the text analysis process and researchers should
apply the normal process of validation to establish that the model has a strong
correspondence between the chosen label and the actual measurement instrument
(label fidelity). These validations are often application-specific and draw on close
reading of the texts.\textsuperscript{11} These validations should be completed in the training set as
described as part of the process of discovering and labeling the \( g \) function.

The train-test split requires an additional validation procedure because we need
to ensure that the model fits nearly as well on the test set as it did on the training set.
When both the training and test sets are random draws from the same population
this will generally be true, but if we have succumbed to some level of overfitting or
we got a bad randomization, it may not be. The same types of techniques used to
validate the original model can be used in the test set as well as common measures
of model fit such as log likelihood. Unlike the validation in the training set, once we
validate in the test set we cannot return to make changes to our model. Nevertheless,
validation in the test set helps us to understand the substantive meaning of what
we have estimated and provides guidance for future experiments. When the test set
is truly a random draw from the same population as the test set, we should expect
these validations to be successful.

Mechanically, applying the \( g \) function in the test set is relatively straightforward
and is essentially the process of making a prediction for a new text. In the
appendix we provide a review of the Structural Topic Model and discuss the me-
chanics of applying the \( g \) function including how to deal with the covariate-based
prior distribution. Once we have our quantities \( g(Y) \) we can use standard estimators
appropriate to our estimand (such as the difference of means to estimate the average
treatment effect).\textsuperscript{12}

\textsuperscript{11}See Grimmer and Stewart (2013) for more detail on types of validation and the \texttt{stm}
package (Roberts, Stewart and Tingley, 2017) for tools designed to assist with validation.

\textsuperscript{12}Once we have applied the \( g \) function to our test data we can calculate confidence intervals using
usual variance estimators that capture uncertainty about our estimate given a limited sample size
conditional on the function \( g \). In prior work we have explicitly taken the view of \( g(Y) \) as a latent
variable about which there is some additional measurement uncertainty and advocated approaches
to incorporate this additional uncertainty into our confidence intervals (Roberts et al., 2014; Fong
and Grimmer, 2016). For example, Roberts et al. (2014) advocates a simulation approach to
integrate over the variational approximation to the posterior distribution which conditions on the
learned topic-word distribution, but accounts for the fact that the document-topic proportion \( \theta \)
cannot be known with certainty for a particular document because it has a finite length. Fong and
Grimmer (2016) use a bootstrap approach which captures measurement uncertainty both in the
topic-word parameters and the document-topic representation. While this approach is intuitively
appealing, it complicates the definition of \( g \) as a function because we run the risk of the same text
mapping to two different values of the latent variable (failing the vertical line test). In the interest
of simplicity we do not include this form of measurement error in this article and leave to future
work the incorporation of this uncertainty into the causal framework.
Choosing when to access the test set is a highly consequential decision because it signals the end of the discovery phase and the beginning of estimation for this particular experiment. Once we have accessed the test set, we cannot continue to refine our $g$ function as doing so would lower the value of having a held-out test set. As such we recommend a few steps before deciding to work with the test data.

1) Take One More Look at the $g$ Function

Be sure the $g$ function is capturing the aspect of the texts that you want to capture, assign labels and then validate to ensure that the conceptual gap between those labels and the representation the $g$ function produces is as small as possible. While validation approaches may vary- this necessarily involves reading documents (Krippendorff, 2004; Quinn et al., 2010; Grimmer and Stewart, 2013). It may be helpful to fix a set of human assigned labels, example documents and automated keyword labels in advance so the analyst is not tempted to adjust their description to match the estimated results.

2) Fix Your Evaluation Plan

In addition to concerns around the $g$ function you want to follow best practice in preparing for an experiment including fixing your plan for analysis in advance. The more degree of freedom you allow yourself in the test set, the easier it will be to fool yourself into a believing a finding which will not hold up under a subsequent experiment. Here we can draw from the established literature on best practices in experiments (Gerber and Green, 2012) potentially including a pre-analysis plan (Humphreys, Sanchez de la Sierra and Van der Windt, 2013).

2.5 Tradeoffs in Train-Test Split

While the train-test split addresses many of our concerns, it is not without cost. The most notable concern is the efficiency loss in splitting the data. In a 50/50 train-test split we have lost half our data for both estimating the $g$ function and estimating the treatment effect. It is difficult to assess how much data we need for either the training or the test set. The challenge in the test set is that we don’t yet know what the outcome will be when we have to make the choice. However, while we in the discovery phase we will be able to make a reasonable assessment of whether the sample size we have allotted for the test set will be able to estimate an effect the size we are interested (Gelman and Carlin, 2014).

The problem is even more complicated for setting the size of the training set because fundamentally we don’t know the power we need for discovery. As a technical matter the results on posterior contraction rates for topic models are few and far between and generally assume the data generating process as well as consider relatively small numbers of topics (Tang et al., 2014; Nguyen, 2015). Even if we trust our intuition about how many documents we need, it is difficult to know how many documents we will need to discover and estimate the most interesting outcome. In practice, we have to simply make a choice.

One natural approach would be to adopt a cross-fitting or cross-validation approach which has been extremely successful in other contexts Anderson and Ma-
gruder (2017); Chernozhukov et al. (2017). In $k$-fold cross validation the data is partitioned into $k$ equally sized partitions. The model is trained on all but one of these partitions (called the held-out set) and then model is estimated on the held-out set. Then the procedure is repeated so each of the $k$ partitions is treated as the held-out set at least once. This forms an estimate for every observation $i$ where the prediction comes from a model which was not trained on observation $i$. This is a powerful approach but relies on the idea that the predictions will be comparable across observations which is true, for example, in settings where the estimand is well-defined in advance of the split. In our setting, we have two problems. First, when a human is in the loop there is not way to separate the model fitting procedures because the human remembers the previous training-test split. Second, because the estimand is not defined in advance of the split, every fold of the cross-validation could be measuring a fundamentally different concept. Thus we expect that a cross-validation or cross-fitting strategy could only be pursued under strong assumptions about the existence of a true $g$ function or limitations on the discovery process. Any such strategy would involve aggregating over different $g$ functions which also compromises the interpretation of the exercise.

A concern we might have about this process is the stability that the entire procedure has over different train-test splits. This will change what we discover given any train-test split but it will not invalidate our results. Still there is an understandable preference to have a sufficiently large train-test split that the particular randomization of the train-test is irrelevant. We empirically explore the stability of STM under different train-test splits in the appendix. Our results suggest that a productive area of future research interest is to develop initialization procedures for the model which are more stable under different splits of the data (even perhaps at some expense of the quality of the initialization).

The train-test split offers many advantages; most notably peace of mind for both the analyst and the reader that the discovered finding holds in a held-out sample. For the analyst this allows the process of discovery to be more unconstrained. It also allows the study to make rigorous claims of having estimated a causal effect. However, the methodology we outline significantly decreases the amount of data available to us for both discovery of the $g$ function and estimating the causal effects. Whether or not the benefits are worth the costs is a function of the particular researcher’s objective.

### 2.6 Prior Work, Implications and Open Questions

There has been comparatively little work on causal inference with latent variables. Lanza, Coffman and Xu (2013) consider causal inference for latent class models but do not give a formal statement of identifying assumptions or acknowledge the set of concerns we identify as an analyst induced SUTVA violation. There is also a line of work on treatment effects for ordinal outcome variables. Volfovsky, Airoldi and Rubin (2015) present a variety of estimands and estimation strategies for causal effects where the dependent variable is ordinal. They provide approaches based both on the observed data as well as latent continuous outcomes. Volfovsky, Airoldi and Rubin (2015) express caution about the latent variable formulation due to identifi-
cation concerns and the subsequent literature (e.g., Lu, Ding and Dasgupta, 2015) has moved away from it. Unfortunately many of the strategies based directly on the observed outcomes are unavailable in the much higher dimensional setting of text analysis.

There is also a burgeoning literature on causal inference using machine learning (Van der Laan and Rose, 2011; Athey, 2015; Athey and Imbens, 2016; Bloniarz et al., 2016; Hartford et al., 2016; Chernozhukov et al., 2017; Wager and Athey, 2017; Ratkovic and Tingley, 2017). Much of this work focuses on estimating causal parameters on observed data and addressing a common set of concerns such as estimation and inference in high-dimensional settings, regularization bias and overfitting. Our work complements this literature by exploring the use of latent treatments and outcomes. Many pieces in this area call for sample splits or cross-validation for estimation and inference, providing additional justification for our preferred approach (see e.g. Chernozhukov et al., 2017). In a related piece of work, Anderson and Magruder (2017) advocate for an improved form of a sample split that minimizes false-discoveries of a treatment on a high-dimensional outcome space and compares the approach to performance under a preanalysis plan.

Although text is an obvious area of concern for Analyst Induced SUTVA violation, the points we make are applicable in any case where there is a latent construct/mapping. This could include latent measures common in political science such as measures of democracy (e.g. Polity), voting behavior (e.g. ideal points) and forms of manual content analysis. Any time a process of discovery is necessary, we should be concerned if the discovery is completed on the same units where the effect is estimated. In certain circumstances this process will be unavoidable- Polity scores were developed by looking at the full population of world democracies so there is no test set we can access, but in the development of future methods, we argue that there is substantial value in acknowledging and accounting for the discovery phase.

What then does this mean for applied work on open-ended survey experiments? The AISV concern clarifies a subtle way that one of the standard assumptions in causal inference could be violated, but it does not mean any work not employing a train-test split is invalid. However, as estimands based on latent constructs become more common in the social sciences, we hope to see an increased use of the train-test split and possibly the development of new methodologies to explicitly address the process of discovery.

3 Application: Using the Structural Topic Model for an Immigration Experiment

In this section, we illustrate our experimental process for text as outcome by applying the Structural Topic Model to an experiment about immigration and crime in the United States. The research question of interest is how a known history of violent crime for a person who has illegally entered the United States affects what how respondents believe the United States criminal justice system to respond to the knowledge that the person has illegally entered the U.S.

Despite the relevance of this question to current events within the United States,
as a starting point we analyze data from an existing survey experiment conducted and made available by Cohen, Rust and Steen (2004). The survey experiment was administered in the context of a larger study of public perceptions of the criminal justice system. The survey was conducted in 2000 by telephone random-digit dial and includes 1,300 respondents.\(^\text{13}\)

In the survey experiment of interest, respondents were given two scenarios of a criminal offense. In both conditions, the crime was illegal entry into the United States. In the treatment condition, respondents were told that the person had previously committed a violent crime and had previously been deported. In the control condition, respondents were told that the person had never been imprisoned before.\(^\text{14}\)

In the treatment condition, respondents were told:

```
“A 28-year-old single man, a citizen of another country, was convicted of illegally entering the United States. Prior to this offense, he had served two previous prison sentences each more than a year. One of these previous sentences was for a violent crime and he had been deported back to his home country.”
```

In the control condition, respondents were told:

```
“A 28-year-old single man, a citizen of another country, was convicted of illegally entering the United States. Prior to this offense, he had never been imprisoned before.”
```

Respondents were then asked a close-ended question about whether or not the person should go to jail. If they responded that the person should not go to jail, they were asked to respond to an open-ended question, “Why?” Based on the subsequent questions in the survey, the researchers who designed the survey experiment were interested in estimating whether or not the respondent believed that the person should not go to jail because he should be deported, or whether the respondent believed that the person should not go to jail for some other reason.\(^\text{15}\)

The challenge for analyzing the open-ended survey experiment like this one is to define a mapping \(g()\) from the text to the categories of interest. The \(g()\) mapping

\(^{13}\)More details about the survey are available in Cohen, Rust and Steen (2002).

\(^{14}\)The treatment and control conditions were placed in a slightly different order in the survey and we are aware that there may be order effects. To address some of the features of the experiment that we would design differently, we are currently re-running the survey experiment in the U.S. on Mechanical Turk.

\(^{15}\)Immediately following the experiment, the survey enumerator was told to follow up with a few other questions if the respondent used the word “deport” in their open ended response.
is challenging to define before seeing the open-ended responses. The researcher may know some of the categories of interest – like deportation – before the fact, but may not know the function that maps between the text and these categories. For example, in this survey, many respondents explained that they thought the person in the scenario should be deported, but many did not use the word ‘deport’; saying instead, “He should be sent home.” The researcher would need to understand the many different ways that a person could express the same concept before defining the \( g \) function.

In addition, the researcher may not know some of the categories before collecting the open-ended responses, and the categories themselves need to be known before the mapping \( g() \) can be defined. For example, many people within the survey responded that they did not think that the person in the vignette had committed a crime. Because both scenarios explain that the person has been “convicted” of illegally entering the country, the researcher may not have anticipated that the respondents would disagree with the court conviction entirely.

Because the researcher may not know the mapping from the text to the categories of interest or even the categories themselves, the discovery phase of the research process allows the researcher to explore and define the \( g() \) function before testing the treatment effect. In this example, we split the data into two equally sized training and test sets. Using the training set, we fit many structural topic models with treatment as a prevalence covariate using various preprocessing techniques and numbers of topics. This allows us to explore a variety of mappings and decide on one that best measures the concepts of interest and reflects the range of ideas expressed within the data. We settled on an STM specification of ten topics that we believed best measured the concepts within the open-ended responses. The ten topics estimated within the training set as well as representative documents from each topic are reported in Tables 1 and 2.

<table>
<thead>
<tr>
<th>Topic</th>
<th>Label</th>
<th>Highest Probability Words</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topic 1</td>
<td>He wants a better life</td>
<td>didnt, want, pay, better, life, probabl, isnt</td>
</tr>
<tr>
<td>Topic 2</td>
<td>Send him back</td>
<td>back, countri, send, home, well, charg, troubl</td>
</tr>
<tr>
<td>Topic 3</td>
<td>Small punishment</td>
<td>offens, reason, like, chanc, first, can, citizen</td>
</tr>
<tr>
<td>Topic 4</td>
<td>Depends on circumstances</td>
<td>come, depend, doesnt, free, feel, law, shouldnt</td>
</tr>
<tr>
<td>Topic 5</td>
<td>Crime was not violent</td>
<td>crime, commit, violent, immigr, wasnt, look, never</td>
</tr>
<tr>
<td>Topic 6</td>
<td>Deport</td>
<td>deport, that, give, counti, peopl, look, guilti</td>
</tr>
<tr>
<td>Topic 7</td>
<td>Prison is too strict</td>
<td>enter, anyth, right, live, realli, illeg, anybodi</td>
</tr>
<tr>
<td>Topic 8</td>
<td>Right to freedom</td>
<td>just, tri, get, hes, came, freedom, put</td>
</tr>
<tr>
<td>Topic 9</td>
<td>Deport bc overcrowded</td>
<td>sent, prison, think, alreadi, anoth, done, hasnt</td>
</tr>
<tr>
<td>Topic 10</td>
<td>Deport bc expense</td>
<td>dont, think, know, time, need, serv, crimin</td>
</tr>
</tbody>
</table>

Table 1: Words most representative of topics.

To create a final estimate, we cluster the topics into two main categories – topics
Table 2: Representative documents of each topic.

<table>
<thead>
<tr>
<th>Topic</th>
<th>Label</th>
<th>Representative Document</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topic 1</td>
<td>He wants a better life</td>
<td>we’re the land of opportunity everybody wants a better life</td>
</tr>
<tr>
<td>Topic 2</td>
<td>Send him back</td>
<td>send him back to his country</td>
</tr>
<tr>
<td>Topic 3</td>
<td>Small punishment</td>
<td>”it was his first offense, didn’t hurt anybody, maybe a fine though, probation or something. that’s nice serious like murder or robbery”</td>
</tr>
<tr>
<td>Topic 4</td>
<td>Depends on circumstances</td>
<td>it depends on reason why he is coming into state if he was coming to better himself it’s ok if he has a record he should be disbarred or deported</td>
</tr>
<tr>
<td>Topic 5</td>
<td>Crime was not violent</td>
<td>because he didn’t commit a crime that was effecting someone else’s individual liberties</td>
</tr>
<tr>
<td>Topic 6</td>
<td>Deport</td>
<td>he should be deported</td>
</tr>
<tr>
<td>Topic 7</td>
<td>Prison is too strict</td>
<td>because he didn’t do anything except illegally enter</td>
</tr>
<tr>
<td>Topic 8</td>
<td>Right to freedom</td>
<td>Because he’s just trying to get his freedom. Maybe he’s trying to away from a tough situation/that country-maybe it’s not good for him.</td>
</tr>
<tr>
<td>Topic 9</td>
<td>Deport bc overcrowded</td>
<td>he should be sent to prison in another country our prisons are over crowded already</td>
</tr>
<tr>
<td>Topic 10</td>
<td>Deport bc expense</td>
<td>because i think he should be deported-p-i don’t think he should be supported in our prison system and i don’t think he should be allowed to immigrate here</td>
</tr>
</tbody>
</table>

related to deportation (Topics 2, 6, 9, and 10) and topics that indicate the respondent would neither deport nor jail the person in the vignette (Topics 1, 3, 4, 5, 7, and 8). This allows us to estimate the relationship between treatment and more general topical categories in the training set on the combination of the closed and open ended responses, reported in Table 3. Within the training set, the treated group (where the respondents were told that the person had previous criminal history) is more likely to report that they think the hypothetical person should go to jail, second most likely to report that the hypothetical person should be deported, and unlikely to report that the hypothetical person should be neither deported or sent to jail. Within the control group, fewer people thought the person should be sent to jail and instead thought the person should either be deported or another course of action should be considered.

<table>
<thead>
<tr>
<th></th>
<th>Jail</th>
<th>Deport</th>
<th>Neither</th>
</tr>
</thead>
<tbody>
<tr>
<td>Treatment</td>
<td>0.66</td>
<td>0.18</td>
<td>0.14</td>
</tr>
<tr>
<td>Control</td>
<td>0.23</td>
<td>0.38</td>
<td>0.39</td>
</tr>
</tbody>
</table>

Table 3: Impact of previous crime treatment on perceptions of just outcomes, training set.

<table>
<thead>
<tr>
<th></th>
<th>Jail</th>
<th>Deport</th>
<th>Neither</th>
</tr>
</thead>
<tbody>
<tr>
<td>Treatment</td>
<td>0.66</td>
<td>0.20</td>
<td>0.14</td>
</tr>
<tr>
<td>Control</td>
<td>0.23</td>
<td>0.38</td>
<td>0.39</td>
</tr>
</tbody>
</table>

Table 4: Impact of previous crime treatment on perceptions of just outcomes, test set.
Figure 1: Comparison of the relationship between treatment and topics in the training and test sets.
To ensure that this relationship is not simply an artifact of the data within the training set, we apply the fitted STM model we decided on within the training set onto the data in the test set that was not used to estimate the mapping $g()$. We compare the relationship between treatment and topics in Figure 1. For the most part, the relationships between treatment and topics are similar in the training and test set. The aggregated topic results are also very similar, as shown in Table 4. Not only do the training and test sets have similar results, these results match up almost exactly with the hand coding of the open ended responses from the authors’ original paper (Cohen, Rust and Steen, 2002, pg 30). The evidence from this full analysis suggests that the treatment condition increases the probability that a respondent will think that the person in the vignette should be deported or put in jail, and that the control condition makes it more likely that a respondent will think that the person should be neither put in jail nor deported.

### 3.1 Test Set

<table>
<thead>
<tr>
<th>Label</th>
<th>Highest Probability Words</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topic 1</td>
<td>Prison, committed crimes</td>
</tr>
<tr>
<td>Topic 2</td>
<td>Prison, repeat offender</td>
</tr>
<tr>
<td>Topic 3</td>
<td>Prison, because of the law</td>
</tr>
<tr>
<td>Topic 4</td>
<td>Prison, but depends on circumstances</td>
</tr>
<tr>
<td>Topic 5</td>
<td>Looking for better life</td>
</tr>
<tr>
<td>Topic 6</td>
<td>No prison bc taxpayers</td>
</tr>
<tr>
<td>Topic 7</td>
<td>Stay, but depends on circumstances</td>
</tr>
<tr>
<td>Topic 8</td>
<td>Deport, if needed prison</td>
</tr>
</tbody>
</table>

### 4 Text-Based Treatments

We now consider the case where text is the intervention in an experiment and the dependent variable is a uni-dimensional variable. For example, say that individuals are all randomly assigned different text-based descriptions of the same candidate for a political office. After reading the description they are then asked to rate their evaluation on a feeling thermometer scale from 1 to 100. In this case the treatment is the description of the candidate assigned to the respondents $T_d[i]$. The potential outcomes $Y_i(T_d[i])$ describes respondent $i$’s rating of the candidate under the $d[i]^{th}$ version of the treatment.

#### 4.1 Comparing Text Treatments Without a Latent Representation: A/B Tests

A/B tests, a widely used methodology in industry applications, rely on testing and comparing different versions of text based treatments without a latent-feature representation. In these cases, the average treatment effect comparing two treatments

---

16To be clear in a true application we would advocate simply showing the effect in the test set. Here we show the train/test comparison solely to demonstrate that the effects are relatively stable.
is straightforward: $E[Y_i(T_{d[i]} = 1) - Y_i(T_{d[i]} = 0)]$. Randomly assigning the texts identifies the average treatment effect of comparing one treatment to another. More generally, we can compare any pair of treatments through similar random assignment of texts to individuals.

This methodology is a workhorse tool across industry. A variety of industry methodologies attempt to determine the optimal treatment to deliver, or the optimal treatment based on a set of categories. To do this, a set of texts (or other high-dimensional data, such as a website design) are provided and then a response is evaluated.

A/B tests are useful for applications where a best text is needed to use for an application; they are less useful for social science research. Social scientists are rarely interested in obtaining the “best” version of a text or in comparing two messages without reference to some underlying feature.

### 4.2 Binary Latent Treatments

Rather than A/B tests to estimate the response to an entire block of text, social scientists are often interested in how some underlying feature of a document affects
responses—that is the researcher is interested in estimating on how an aspect or latent value of the text influences the outcome. For example, the researcher might be interested in whether including military service in the description has an impact on the respondents’ ratings of the candidate. Military service is a latent variable – there are many ways that the text could describe military service that all would count as the inclusion of military service and many ways that the text could omit military service that all would count as the absence of the latent variable. The researcher might assign 100 different candidate descriptions, some which mention the candidate’s military service and some which do not. In this case, the treatment of interest is $Z_i = g(T_{d[i]})$ which maps the treatment text to an indicator variable that indicates whether or not the text contains a description of the candidate’s military service.

The latent representations of the text-based treatment presents complications to causal inference and not just if machine learning methods are used to infer the $g$ function.

### 4.2.1 Analyst-induced SUTVA violation and overfitting

First, the latent-representation of text creates the possibility of an analyst-induced SUTVA violation (AISV). As we described above, if the researcher decides on the mapping $g$ between text based intervention $T_{d[i]}$ and binary treatments $Z_i$ based on the full text of all $T$s and responses, then the inferred treatments for each unit, $Z_i$, will depend on the texts allocated to all other units. Further, if researchers are able to search for different $g$ functions then they will be able to fish for significant treatments across a much broader search than standard experiments.

Therefore, when a researcher decides to represent treatment as a latent variable that is a function of the text – whether through hand coding or statistical modeling – they should take precautions to avoid analyst-induced SUTVA assumptions and overfitting by ensuring that $g$ is not estimated based on the full corpus of treatment texts or on the full outcome variables $Y$. Either the researcher should define $g$ from the outset or should split the data into a training and test set and estimate $g$ only from the training set, as discussed above. This addresses both the problem of analyst-induced SUTVA violations and of overfitting.

### 4.2.2 Sufficiency of Latent Treatments

Focusing on inferring the effect of latent treatments, rather than entire texts, enables social scientific inferences from text-based experiments. The tradeoff, however, is that randomization is often done at the text level. In order to make an inference about the underlying latent treatments after a text-based randomization, we have to assume that our $g$ function captures all the information relevant to the response in $T$ is contained in $Z$. We present two versions of this assumption that is necessary to identify the causal effect. (For proofs see Fong and Grimmer (2016) and Fong et al. (2017)). The stronger assumption is the most intuitive and that is that $Y_i(T_{d[i]}) = Y_i(g(T_{d[i]}))$ for all documents and for all respondents. In words, this assumption requires that the potential outcome response to the text be identical to the potential outcome response to all documents with the same latent feature representation. This
assumption is strong because it requires that there is no other information contained in the text that matters for the response beyond what is contained in the latent feature representation. In our running example about military service, this would mean that the inclusion or exclusion of military service is the only aspect relevant to the effect of the document on the individual’s rating. Particularly for text, we could imagine that this assumption could easily be violated. If both versions of the treatment contain “The candidate served in the military”, but one also adds “The candidate was dishonorably discharged” we might expect that this additional text added in addition $Z$ may be relevant to the responses.

We can weaken this strong assumption to allow for some features of the text not contained in the latent feature representation to matter for the potential outcome. Rather than assume all responses are equal, Fong et al. (2017) shows we can assume that for any two treatments $T, T' \in \mathcal{T}$, with the same feature representation ($g(T) = g(T')$), then $E_i[Y_i(T)] = E_i[Y_i(T')]$. In words, this assumption states that for any individual the response to two documents with the same latent feature representation might differ, but on average over individuals the responses are the same. Fong et al. (2017) shows that this assumption is equivalent to supposing that the components of the document that affect the response and are not included in the latent feature representation are orthogonal to the latent feature representation. Technically, we can write $\epsilon_i(T) = Y_i(T) - Y_i(g(T))$. Then this more general assumption is equivalent to assuming that $E_i[\epsilon_i(T)] = 0$ for all $T$.

4.2.3 No Hidden Values of Treatment

Even if we avoid AISVs and the latent treatments are sufficient, the latent variable representation of treatment also requires that $Z$ does not contain hidden values of the treatment. This can emerge because a description of a particular treatment might obscure some underlying variation in the treatment. For example, suppose we create a dictionary based coding rule that considers a candidate’s biography to refer to military experience if it mentions the word army or military. Then this will include individuals who have served in the military, but also to candidates whose parents served in the military, or even candidates who worked for the Salvation Army.

In this case we can still obtain an unbiased estimate the treatment effect, though the estimated effect will differ from the effect that we think we are estimating. This is problematic for interpretation and for applications to other experiments that attempt to build on the first round. With careful validation of the text-based treatment we can limit the risk of hidden versions of the treatment in confounding our analysis.

4.3 Multidimensional Latent Treatments

With complicated interventions, such as text, it is likely that there are more than just one latent treatment. The presence of multiple interventions requires defining new causal quantities of interest and suggests new potential questions we can ask about how the features of a text affect a response. With multidimensional latent treatments
we can go beyond asking what effects including or excluding military service has on perceptions of a candidate, we can also learn how military service interacts with other components of the text to influence responses—such as a candidates career background, education prior to military service, or life after the military. We might think of texts as combinations of multiple treatments - a candidate description could include military service, occupation description, or a description of their family. In this case, we map the treatment text $T$ to a vector of $K$ binary treatments $Z_j \in \mathcal{Z}$ where $\mathcal{Z}$ represents all $2^K$ possible combinations of the treatments.

Scholars often create the multidimensional treatments explicitly. For example, Grimmer, Messing and Westwood (2017) examine how claiming credit for funding in a district with a press release affects the way constituents allocate credit to elected officials. Grimmer, Messing and Westwood (2017) design an experiment that varies many aspects of legislators’ messages to their participants. In the treatment condition of the experiment, the authors are able to vary many different aspects of credit claiming. Treatment $T_j$ is the combination of all of the texts. This maps directly into a binary treatment vector $Z_j$ which describes the components of the text to which the participant was assigned.

By using a multidimensional treatment, researchers are able to identify the aspects of the text that are most relevant to the outcome variable. The researchers can estimate the Average Marginal Component Effect (AMCE) for each individual component $k$, using methods developed for conjoint analysis (Hainmueller, Hopkins and Yamamoto, 2013). The AMCE is the average effect of the component $k$, averaged over all other possible combinations:

$$AMCE_k = \int_{Z_{-k}} E[Y(Z_k = 1, Z_{-k}) - Y(Z_k = 0, Z_{-k})]m(Z_{-k})dZ_{-k}$$

The $AMCE_k$ describes the average effect of component $k$, summed over all other values of $k$, weighted by $m(Z_{-k})$, or the distribution of $Z_{-k}$. The AMCE can be thought of as an estimate of the effect of component $k$, given the distribution of other components in the population—therefore providing a sense of how an intervention will matter given other characteristics.

The analyst might also be interested in estimating the effect of an interaction between two components $k$ and $l$. For example, the researcher might be interested if including military service into a candidate profile has a different effect on candidate ratings if the profile also includes that the candidate is female. This could be estimated as the Average Component Interaction Effect (ACIE) (Hainmueller, Hopkins and Yamamoto, 2013):

$$ACIE_{k,l} = \int_{Z_{-k,-l}} E[(Y(Z_k = 1, Z_l = 1, Z_{-k,-l}) - Y(Z_k = 1, Z_l = 0, Z_{-k,-l}))
- (Y(Z_k = 0, Z_l = 1, Z_{-k,-l}) - Y(Z_k = 0, Z_l = 0, Z_{-k,-l})]m(Z_{-k,-m})dZ_{-k,-m}$$

The ACIE will be the difference between the AMCE for military service for a candidate description that includes information that the candidate is female and the AMCE for military service for a candidate description that does not include this information.
Note that the three complications from the last section also pertain to the case of multidimensional treatments. If the mapping $g()$ between $T$ and $Z$ is not known before defining and reading the treatment texts or the outcome is used in the estimation of these mapping, then an AISV will occur. Even when using hand coding, researchers should either use a pre-test to determine their coding scheme or use a training/test split to first learn a coding scheme using the responses and then separately estimate the treatment effects.

### 4.4 Unsupervised Discovery of Latent Treatments

In the cases described above, the researchers came up with their own representation of $g$ either through their substantive knowledge, a test of a theoretical argument, or through careful reading of documents in the training set. They then constructed the text treatments to reflect the treatments they wanted to test. In many cases, researchers will know that they want to estimate the effect of including military service or the effect of including the amount of funding raised on the outcome. However, in other cases, researchers may not have a clear sense of the treatments they would like to test, or would like some guidance from the data about what the interesting treatments to examine might be. This is equivalent to saying that researchers are not sure what $Z$ should be, or if they have texts of interest in hand, they are not sure of the mapping from the set of texts $T$ to the set of treatments $Z$.

One approach to estimating the function $g$ would be to estimate features that explain the texts well, such as the principal components of a document-term matrix. Using these estimates for $Z$, the impact of $Z$ on $Y$ can be estimated directly. The use of principal components to estimate causal quantities of interest leads to several issues. First, the set of features that are discovered best explain the texts, but do not condition on the responses. This can lead to the discovery of features that explain the content of texts but do not explain the response to those texts. Second, the use of principal components leads to a continuous treatment. This might be reasonable, however, estimating the response to a continuous treatment requires strong functional form assumptions and often a great deal of data to estimate more than a linear response. Alternative approaches, such as an Indian Buffet Process (Griffiths and Ghahramani, 2011), yield a binary feature vector about the treatments that are present or absent in a text, but fail to include information about the responses.

Certainly focusing on the texts alone is a reasonable approach, but including information about the response can improve the latent features that the $g$ function maps texts to. Fong and Grimmer (2016) create an unsupervised method for estimating treatments from text data. They develop a supervised Indian Buffet Process (sIBP) that discovers the topics within the documents that are related to the outcome. The authors assume that the proportion of documents in each latent feature $k$ is $\pi_k$, where $\pi_k$ is generated by a stick-breaking algorithm (Doshi et al., 2009). Each document can be summarized by treatment vector $Z_j$ where $z_{j,k} \sim \text{Bernoulli}(\pi_k)$. Note that because each individual $z_{j,k}$ is drawn from a Bernoulli that a treatment document can have more than one latent feature, allowing for multi-dimensional treatments.

The authors assume a mapping from $Z_i$ to the standardized term-document ma-
trix $X_i$ through the D-dimensional vector $A_k$, where $X_i \sim \text{MvtNormal}(Z_i, A, \sigma^2 \pi I_D)$. The latent feature vector $Z_i$ also affects the response $Y_i$ through the normal, $Y_i \sim \text{Normal}(Z_i \beta, \tau^{-1})$ where $\tau \sim \text{Gamma}(a, b)$. Thus with the model the authors both want to discover the latent treatments $Z_i$ and estimate their influence on the outcome by estimating $\beta$. The authors use variational approximation to estimate these parameters.

Fong and Grimmer (2016) apply the sIBP to the training data in order to learn a $g$ function. In the test set Fong and Grimmer (2016) use the $g$ function to infer the treatments that are present in a particular text, but alter the inference to avoid conditioning on the dependent variable. They do this because otherwise the inferred treatments present in the test set will depend upon the observation’s response to that text, which creates obvious problems for causal inference.

Once the latent treatments are inferred in the test set documents, their effect can be estimated using any procedure that might be used to analyze an experiment. Fong and Grimmer (2016) use a simple linear regression with each of the latent features as the regressors to estimate the effects of the treatments. More complicated models could be used to estimate interactions or to extrapolate effects to a different population of documents.

5 Application: Consumer Financial Protection Bureau

To illustrate the use of text-based treatments, we analyze the determinants of a timely resolution to complaints filed at the Consumer Financial Protection Bureau (CFPB). The CFPB is a product of Dodd-Frank legislation and is (in part) charged with offering protections to consumers. The CFPB solicits complaints from consumers across a variety of financial products and then addresses those complaints. It also has the power to secure payments for consumers from companies, impose fines on firms found to have acted illegally, or both.

The CFPB is particularly compelling for our analysis because it provides a massive database on the text of the complaint from the consumer and how the company responded. If the person filing the complaint consents, the CFPB posts the text of the complaint in their database, along with a variety of other data about the nature of the complaint. For example, one person filed a complaint stating that “the service representative was harsh and not listening to my questions. Attempting to collect on a debt I thought was in a grace period ... They were aggressive and unwilling to hear it” and asked for remedy. The CFPB also records whether a business offers a timely response once the CFPB raises the complaint to the business. In total, we use a collection of 113,424 total complaints downloaded from the CFPB’s public website.

While the texts are not randomly assigned to the CFPB, the way complaints are recorded at the CFPB makes it plausible that we have all the observable features necessary to understand the determinants. The bureaucrats at the CFPB have little information other than what is in the complaints and have little capacity to collect additional information. Given these limitations, the assumption that the texts pro-
Table 5: Consumer Financial Protection Bureau Latent Treatments

<table>
<thead>
<tr>
<th>No.</th>
<th>Automatic Keywords</th>
<th>Manual Keyword</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>payment, payments, amount, interest, balance, paid, month</td>
<td>loan</td>
</tr>
<tr>
<td>2</td>
<td>card, called, call, branch, money, deposit, credit_card, told</td>
<td>bank</td>
</tr>
<tr>
<td>3</td>
<td>debt, debt_collection, account, number, validation, dispute, collection</td>
<td>debt collection</td>
</tr>
<tr>
<td>4</td>
<td>xxxx, account, time xxxx_xxxx, request, copy, received, letter</td>
<td>detailed complaint</td>
</tr>
<tr>
<td>5</td>
<td>payment, payments, pay, told, amount, month, called</td>
<td>disputed payment</td>
</tr>
<tr>
<td>6</td>
<td>loan, mortgage, modification, house foreclosure, payments</td>
<td>mortgage</td>
</tr>
<tr>
<td>7</td>
<td>debt, debt_collection, collection, credit_reporting, proof, credit_report</td>
<td>threat</td>
</tr>
<tr>
<td>8</td>
<td>fcra, credit_report, credit_reporting, reporting, debt, violation, law</td>
<td>credit report</td>
</tr>
</tbody>
</table>

provide all the information for the outcome seems reasonable. It could be violated if there are other non-textual factors that correlate with the text content. For example, if working with the CFPB directly to resolve the complaint were important and individuals who submitted certain kinds of complaints were less well equipped to assist the CFPB, then we would have concern the selection on observables assumption holds.

We are interested in learning both the salient features that determine a salient response and then the effect of those features on the probability of a response. To do this, we first randomly divide the data, placing 10% in the training set and 90% of the data in the test set. We place more data in the test set because our large sample provides ample opportunity to discover the latent-treatments in the training set and to provide greater power when estimating effects in the test set. In the training set we apply the sIBP process to the text of the complaints and whether there was a timely response to infer the latent treatments. We use an extensive search to determine the number of features to include and the particular model run to use. The sIBP is a nonparametric Bayesian method, so it will estimate the number of features to include in the model, though the number estimated from a nonparametric method rarely corresponds to the optimal number for a particular application. To select a final model we then evaluate the candidate model fits utilizing a model fit statistic introduced in Fong and Grimmer (2016) that provides a quantitative measure of model fit. And because we are using a training set, we can also refit the model several times choosing the model that provides the features that provide the best substantive insights.

Once we have fit the model in the training set, we use it to the infer the treatments in the test set. Table 5 provides the inferred latent treatments from the CFPB complaint data. The Automatic Keywords are the words with the largest values in the estimate latent factors for each treatment, and the manual keyword is a phrase that we assign to each category after assessing the categories. Using these features we can then infer their presence or absence in the treated documents and then estimate their effect. To do this we use the regression procedure from Fong and Grimmer (2016) and then use a bootstrap to capture uncertainty from estimation.

Figure 2 shows the effects of each latent feature on the probability of a timely response. The black dots are point estimates and the line are 95-percent confidence intervals. Figure 2 reveals that when consumers offer more detailed feedback (Treatment 4) and when complaints are made about payments to repay a loan. In contrast, the CFPB is much less successful at obtaining prompt responses from debt collectors—either when those collectors are explicitly attempting to collect a debt
(Treatment 3) or when the debt collectors are threatening credit reports (Treatment 7). The inability to obtain a prompt response from debt collectors is perhaps not surprising—debt collection companies exist to successfully recoup funds and are likely less concerned with their perceived reputation with debtors. It also demonstrates that it can be harder to remedy consumer complaints in some areas than others, even if the FSPB is generally able to assist complaints.

6 How do we know we’ve learned something?

With data as complicated as high-dimensional as text and so many different possible mappings between text and the latent variables, we might be concerned about the generalizability of our inferences and with validating our conclusions in other data and domains. Do the findings hold only in this dataset? Did our randomly drawn training/test split result in a specific analysis that would not occur in other dataset? Might other researchers have chosen a different $g$ function that would have provided different conclusions?

We argue that the only way to ensure generalizable findings is through a variety of replications. Each step in the research process offers some place where different decisions could have been made or different strategies could have been pursued. While this “garden of forking paths” is often used to criticize empirical research, recognizing the forked paths and varying as many of the decisions as possible will provide guidance on the robustness of our findings. We argue for replication because results achieved with a training/test split are internally valid, results that hold after replicating all stages of the research process are more general than those that do not withstand replication in new datasets or to new models. Here we introduce various levels of replication that serve as levels of validation of the generalizability of the
findings for the types of experiments we discuss above.

**Fit the model on new test data.** The researcher might use the model they obtained through exploration in the training set to estimate the impact of treatment in a different test set than their original test set. This new test data could simply be another randomly split subset of the original data. Or it could be test data collected in a different sample or even in a different domain. If the results were similar in a number of varied test sets, we might be more certain that the findings generalize more broadly.

**Estimation of the model – model selection** The researcher might re-estimate the model in the original training set and provide a set of model estimates based off of a variety of models. This would show that the results were not specific to a particular coding scheme $g$, regardless of whether that $g$ came from hand-coding or a topic model.

**Training split** The researcher might re-sample the training/test split. This would require re-estimating the model and validating the results in a new test set. If the results were robust to the training/test split it would show that the result could be discovered under multiple training/test splits. Re-estimating by repeatedly drawing new training and test splits introduces the possibility for an AISV. But even when confronted with that violation, redrawing the split (and following some fixed procedure) enables an assessment of the stability of the algorithm.

**Collect data** The researcher might completely recollect the data. This would automatically generate a new training/test split because the data would all be completely different. This would show that the results are robust to variation in the implementation of data collection and to a different set of individuals.

**Choose a domain/context** The researcher might decide to ask the same question in a different domain or context, i.e. in a different population or time period. Replication in this context would show the most broad generalizability of the findings as the result would hold in broader populations of individuals.

### 6.1 Conclusion

Making causal inferences with textual data is difficult to do, even within an experimental context. While high-dimensional data like text requires the researcher to discover the mapping between the data and the quantities of interest, this process of discovery undermines the researcher’s ability to make causal inferences. We clarify this issue by describing the Analyst-induced SUTVA violation and show that this type of SUTVA violation can be solved with a simple split of the dataset into a training set used for discovery and a test set used for testing. More broadly, we advocate for research designs that allow for sequential experiments that explicitly set aside research degrees of freedom for discovery of interesting measures, while
rigorously testing relationships within experiments once these measures are defined explicitly.

While we set out the basic framework for doing causal inferences with text data here, there is much more work to be done to understand the optimal research design within this framework, for example we need a deeper understanding of the tradeoffs in train/test splits. Very little research has explicitly tackled methods for discovery of new measures, and we hope that future work will help spell out the types and amount of data best suited for discovery from data. Last, the analyst-induced SUTVA assumption does not only apply to text data, but to any latent representation of treatment and outcome variables. The framework of the train/test split should be considered in the context of other types of data that require a low-dimensional representation of high-dimensional data.
A Using the Structural Topic Model for Text as Outcome

This appendix provides a short review of the Structural Topic Model, explains how to obtain and use the $g$ function and provides an analysis of stability of topic model results across train-test splits. In this appendix we deviate slightly from the notation in the main paper to use bold-faced text to indicate vectors or matrices.

A.1 A Brief Review of the Structural Topic Model

The Structural Topic Model is a mixed membership model of texts related to Latent Dirichlet Allocation (Blei, 2012) which was developed in Roberts et al. (2014); Roberts, Stewart and Airoldi (2016) and implemented in Roberts, Stewart and Tingley (2017). It allows for the analyst to incorporate observed document metadata which is able to affect either topical prevalence (the amount which a topic is discussed) and topical content (the way in which a topic is discussed). In this paper we consider the case in which a set of observed metadata which includes the treatment and pre-treatment covariates are allowed to affect topic prevalence and there are no topical content covariates. Denoting the pretreatment covariates for document $d$ as $X_d$ and the treatment as $T_d$, the generative process can be given as:

$$\eta_d \sim \text{Normal}(X_d \gamma_X + T_d \gamma_T, \Sigma)$$

$$\theta_{d,k} = \frac{\exp(\eta_{d,k})}{\sum_{j=1}^{K} \exp(\eta_{d,j})}$$

$$z_{d,n} \sim \text{Categorical}(\theta_d)$$

$$w_{d,n} \sim \text{Categorical}(\beta_{z_{d,n}})$$

Where $\theta_d$ is a $K$-dimensional vector on the simplex indicating the proportion of the document allocated to each topic formed by applying the softmax function to $\eta_d$ a vector in $\mathcal{R}^{K-1}$ where the $K$-th element is fixed to zero. $z_{d,n}$ is a token level latent variable containing the assignment for token $n$ of document $d$. $\beta$ is a $K$ by $V$ dimensional matrix where each row contains the conditional probability of seeing word $v$ given that is about topic $k$. The model differs from Latent Dirichlet Allocation in its use of a logistic normal prior distribution for the document-topic proportions and through the ability to have that prior centered at a document-specific location determined by the document metadata.

The model is estimated using partially-collapsed, non-conjugate, variational inference. $\gamma$ and $\Sigma$ are given regularizing priors of the user’s choice and $\beta$ is point estimated. The model optimization problem is non-convex and so a careful initialization strategy is necessary (Roberts, Stewart and Tingley, 2016). Roberts, Stewart and Tingley (2016) advocate a deterministic initialization based on the spectral method of moments (Arora et al., 2013) which we refer to below as the ‘Spectral Initialization.’
A.2 Obtaining and Using the $g$ Function

In this section we walk through the procedure for obtaining and using the $g$ function with the Structural Topic Model. The experimental protocol is summarized below. Note that any other method can be used for the discovery stage but here we demonstrate with the STM for simplicity.

- Create the train-test split
- In the training set (discovery)
  - explore the documents as desired using STM
  - choose an estimand (including assigning and validating a label)
  - Identify the mapping function $g$ such that
    $$\hat{\theta}_i = g(Y_i, \beta, \hat{\mu}_i, \hat{\Sigma})$$
- In the test set (evaluation)
  - Using the $g$ function, obtain our transformed outcome for each document. (see below for details)
  - Estimate treatment effects (using for example the difference of means)
  - Validate model fit and label fidelity in the test set.

Application of the $g$ function in STM is equivalent to predicting $\theta_i$ for a held-out document $i$. This can be accomplished with the recently added `fitNewDocuments` function in the `stm` package. In the STM model, the latent variable $\theta_i$ is a function of a global prior $(\mu, \Sigma)$, the topic word parameters $\beta$ and the observed words $W_d$. The token-level latent variables $Z$ are integrated out. We have estimated $\beta$ in the train set and in many ways this communicates what the topics substantively contain. We must also decide how to set our priors $\mu$ and $\Sigma$.

The `stm` package offers three options: no prior, the covariate-specific prior and the average prior. The ‘no prior’ setting sets $\mu$ to a vector of zeroes and $\Sigma$ to be a diagonal matrix with very large diagonals. The covariate-specific prior uses the observed covariates in the new documents to construct the document-specific prior. The average prior averages over the values of $\mu$ in the training set and provides a single average prior for all documents.\textsuperscript{17}

If we have used only pre-treatment covariates in the STM model we can use any of these strategies. In our application we do include the treatment and so we cannot use the covariate-specific prior because then the same text would yield two different values of the outcome depending on the treatment assignment. For our application we use the average prior. In future work we plan to implement functionality in `stm` that will allow us to marginalize out a single covariate such as treatment so that we can use the covariate-specific prior.

\textsuperscript{17}More specifically we take the column means of the $D$ by $K-1$ matrix $\mu$ in the training set which we call $\bar{\mu}$. We then recalculate $\Sigma$ as though the update had been made using the new value of $\mu$. The update is then $\Sigma = \Sigma - (\sum_d(\eta_d - \bar{\mu}_d)(\eta_d - \bar{\mu}_d)^T) + (\sum_d(\eta_d - \bar{\mu}_d)(\eta_d - \bar{\mu}_d)^T)$. 

34
can still make use of pre-treatment covariates in the test set. This still requires us to make a choice about the what distribution of the treatment to marginalize over.

When using a version of the \( g \) function which is not the covariate-specific prior, we recommend that analysts assess effects in the training set using the same procedure as in the test set. While the effects will generally not be very different (particularly for long documents), maintaining the same procedure should provide a better expectation of test set behavior. For example, in our application Figure 3 compares our training set estimates using both the covariate-specific prior and the averaged prior and compares them to the test set (which uses the averaged prior). Using the average prior to make predictions in the training set before calculating effect estimates gives us a better indication of what we will eventually observe in the test set.

### A.3 Stability Across Train-Test Splits

Our approach does not require stability of analysis across different train-test splits. Different train-test splits might lead to different discovery phases which in turn yield different estimands and test sets where we can evaluate that estimand. Nevertheless, we might be slightly uncomfortable with the idea that the particular randomization into the train-test spit yield quite different estimands (and papers) at the end of the
process. As such we wanted to evaluate the stability of the STM under different samples of a fixed population.

In a formal sense we are interested in studying the posterior contraction rates of the model, a problem taken up analytically in Tang et al. (2014) for the related Latent Dirichlet Allocation model. However, we are far more interested in understanding performance in practice and whether different train-test splits lead to substantively different topic-word distributions ($\beta$), different document-topic proportions ($\theta$) and different covariate effects. As the number of documents increase or the topics are more sharply defined stability will improve. For this demonstration we use the Poliblog data (Eisenstein and Xing, 2010), a collection of around 13,246 posts from six different political blogs in the runup to the 2008 American presidential election. We use this because it is readily available for use with the stm package and is roughly representative of the document lengths that we often see in stm applications overall. We would expect that the diversity of topics in political blogs would make the problem harder than the more focused open-ended response case, but the length of the documents would make it easier.

We started by estimating the model on the full set of documents with 20 topics using the spectral initialization. We consider this to be the “truth” because the unattainable stability ideal would be that the subsamples provide the same answer as the full set of documents. We then choose two prominent topics to be our “outcomes” a topic about Obama and a topic about War (particularly Iraq and Afghanistan). In each simulation we choose the topic that most closely approximates our two chosen outcomes, emphasizing that the labels ‘Obama’ and ‘War’ may well not be good approximations for the topic in the subsample.

Because of the multimodality problem in topic models, instability could arise from two sources: differences in the local mode discovered and differences in the data observed. We investigate this by considering three different initialization strategies:

1) Cold Spectral Start
   Using the spectral initialization on the subsample. This is reflective of current practice.

2) Warm Spectral Start
   Use the complete data to initialize the model. This would create an analyst-induced SUTVA violation as it shares information from the test set. However, it is suggestive of what might be achievable by providing more stable initializations.

3) Warm Oracle Start
   Use the results of the converged model on the full sample to initialize each subsample. This is an infeasible estimator. The instability in this estimate cannot be reduced by a better initialization strategy.

In each case we run the model on the sample sizes 100 times and plot the results along with the ‘truth’ as defined by the full document set. Figure 4 shows the results for the average proportion of the topic use in the corpus.

As we can see the results are reasonably stable at 5000 documents for a corpus of this complexity and less so with 1000. The warm spectral start shows considerable
improvement for the 5000 document case suggesting that at least at this scale, we might see substantial gains from an initialization specifically designed to be more stable across splits. The near perfect stability of the warm oracle start for the 5000 document case suggests it is a matter of the initialization and not necessarily the data itself, where for 1000 documents there is evidence that some level of the instability is unavoidable given the model.

We can also examine the word-distributions themselves. Figure 5 shows the proportion of mass associated with each of the top ten words in the topic (as chosen by the full model). The horizontal tickmarks on the right show the estimate in the full data.

Generally speaking the models correctly preserve the rank ordering of the most prominent words in each topic, but the estimates can often be substantially incorrect. We do emphasize that there is relatively little information with which to
estimate these parameters and so we would expect to see more instability than in the simulations for $\theta$.

Finally we present estimates of “treatment effects.” Here we use the binary rating variable (indicating whether the blog is liberal or conservative) as a treatment. This is clearly not randomly assigned and we use it simply because it is a binary covariate we would expect to influence the outcome in some way. We plot the estimate with a 95% confidence interval in Figure 6 along with the estimate in the complete dataset shown in blue.

Once again we can see some substantial variability that appears to be reducible via a more stable initializations. We emphasize that we should not expect these confidence intervals to have proper coverage as in every case the estimand is different. Indeed the high variance on the Obama topic of the warm spectral start is a good indication that the estimand is changing substantially in each different split. What
the relatively tight set of estimates in the two warm starts suggest is that this might be avoidable with a different initialization.

In summary, there is a significant degree of instability across splits. Again, this is not a problem in a technical sense as the $g$ function applied to the test set will still provide a proper estimator of that specific estimand. What these simulations do suggest is that further research into more stable initialization strategies might substantially reduce the amount of instability across train-test splits.

There are several limitations to this simulation study: most notably that we neither know the actual truth nor can we be sure what the scope conditions are
for these results to apply to other datasets. We also cannot simulate the stability of the entire discovery process, only that a particular model is comparable across subsamples. Hoping for stability in discovery may be quixotic as the very idea of discovery itself may imply some level of instability.
References


Gill, Michael and Andrew B Hall. 2015. “How Judicial Identity Changes The Text Of Legal Rulings.”


Lasswell, Harold Dwight. 1938. “Propaganda technique in the world war.”.


Pennebaker, James W, Matthias R Mehl and Kate G Niederhoffer. 2003. “Psycho-
logical aspects of natural language use: Our words, our selves.” Annual review of
psychology 54(1):547–577.

Quinn, Kevin M, Burt L Monroe, Michael Colaresi, Michael H Crespin and
Dragomir R Radev. 2010. “How to analyze political attention with minimal as-

Ratkovic, Marc and Dustin Tingley. 2017. “Causal Inference through the Method

Roberts, Margaret E., Brandon M. Stewart and Dustin Tingley. 2016. Navigating
the Local Modes of Big Data: The Case of Topic Models. In Computational Social
University Press chapter 2, pp. 51–97.

Roberts, Margaret E., Brandon M. Stewart and Dustin Tingley. 2017. stm: R
Package for Structural Topic Models. R package version 1.2.3.

Roberts, Margaret E, Brandon M. Stewart, Dustin Tingley, Christopher Lucas, Jet-
son Leder-Luis, Shana Kushner Gadarian, Bethany Albertson and David G Rand.

Roberts, Margaret E, Brandon M. Stewart and Edoardo M Airoldi. 2016. “A model
of text for experimentation in the social sciences.” Journal of the American Sta-
tistical Association 111(515):988–1003.

methods for high-dimensional data with applications to text. Technical report
Working paper.

time-series party positions from texts.” American Journal of Political Science

change and relative power, 1784–1911.” American Journal of Political Science

Tang, Jian, Zhaoshi Meng, Xuanlong Nguyen, Qiaozhu Mei and Ming Zhang. 2014.
Understanding the limiting factors of topic modeling via posterior contraction

Tausczik, Yla R and James W Pennebaker. 2010. “The psychological meaning of
words: LIWC and computerized text analysis methods.” Journal of language and

Tukey, John W. 1980. “We need both exploratory and confirmatory.” The American

