Proceedings of Machine Learning Research vol 275:243–259, 2025 4th Conference on Causal Learning and Reasoning

Contagion Effect Estimation Using Proximal Embeddings

Zahra Fatemi ZFATEM2@UIC.EDU
Elena Zheleva EZHELEVA@UIC.EDU
University of Illinois Chicago

Editors: Biwei Huang and Mathias Drton

Abstract

Contagion effect refers to the causal effect of peer behavior on the outcome of an individual in social networks. Contagion can be hard to estimate when it is confounded by latent homophily because nodes in a homophilic network tend to have ties to peers with similar attributes and can behave similarly without influencing one another. One way to account for latent homophily is by considering proxies for the unobserved confounders. However, as we demonstrate in this paper, existing proxy-based methods for contagion effect estimation have a very high variance when the proxies are high-dimensional. To address this issue, we introduce a novel framework, *Proximal Embeddings (ProEmb)*, that integrates variational autoencoders with adversarial networks to create low-dimensional representations of high-dimensional proxies and help with estimating contagion effects. While VAEs have been used previously for representation learning in causal inference, a novel aspect of our approach is the additional component of adversarial networks to balance the representations of different treatment groups, which is essential in causal inference from observational data where these groups typically come from different distributions. We empirically show that our method significantly increases the accuracy and reduces the variance of contagion effect estimation in observational network data compared to state-of-the-art methods. We also demonstrate its applicability to two real-world scenarios, estimating contagion on social media and in adolescent smoking behavior.

Keywords: causal inference, contagion effect, proxy variable,peer effects, interference

1. Introduction

The goal of causal inference is to estimate the effect of an intervention on individuals' outcomes. Traditionally, causal inference has relied on the assumption of no interference, which states that any individual's response to treatment depends only on their own treatment and not on the treatment of others. However, individuals can impact each other through their interactions. Contagion is a type of interference that is defined as the influence of neighbors' actions on the actions of an individual. Contagion effect estimation plays a central role in understanding how social environments shape personal actions, behavior, and attitudes (Bramoullé et al., 2009; Christakis and Fowler, 2007; Eckles et al., 2016). Some real-world applications of contagion effect estimation include studying the spread of obesity (Christakis and Fowler, 2007; Krauth, 2005), smoking behavior (Christakis and Fowler, 2008), and fake news (Torres et al., 2018).

Despite their importance, identification and estimation of contagion effects are challenging due to latent homophily Manski (1993); Shalizi and Thomas (2011); VanderWeele and An (2013), the tendency of ties to form between individuals with similar unobserved attributes. When contagion effects are confounded with latent homophily, it is hard to tell if any changes in the individual's outcome are the result of neighbors' influence or the similarity between the individual and neighbors characteristics. For example, people with similar political affiliations would be more likely to

interact on social media (e.g., Twitter) and they may express similar opinions (e.g., agree or disagree with social distancing policies during a pandemic), not because one influences the other but because they share similar political views in the first place.

To identify and estimate contagion effects in the presence of unobserved confounders, existing approaches look for observed variables that can be considered as valid proxies of the unobserved confounders (Miao et al., 2018; Tchetgen et al., 2020; Egami and Tchetgen Tchetgen, 2024). However, such approaches can perform poorly on real-world observational data, such as web and social media, in which a high-dimensional covariate space is the norm. High-dimensional control proxies (e.g., tweet words of a user) lead to a sparse vector of model parameters and higher asymptotic bias and variance of the estimation (De Luna et al., 2011). Another source of variance is selection bias (Guo et al., 2020; Shalit et al., 2017; Assaad et al., 2021). Selection bias occurs when there is a mismatch in attribute distribution between the treatment and control groups in observational data. For instance, a treatment group can comprise mostly individuals who prioritize their health and have friends who follow social distancing guidelines, while the control group comprises of individuals who do not prioritize their health and have friends who largely disregard social distancing measures. A common method for dealing with selection bias in observational studies is matching, where a balanced sample is created by identifying similar units from the opposite treatment group. However, matching tends to encounter scalability issues when applied to high-dimensional data (Abadie and Imbens, 2006; Assaad et al., 2021).

Key idea and highlights. To address high dimensionality and selection bias in real-world contagion estimation settings, we introduce *ProEmb*, a framework for inferring contagion effects in homophilic networks. ProEmb learns embeddings of high-dimensional proxies for unobserved confounders. ProEmb combines variational autoencoders (VAEs) and adversarial networks (Goodfellow et al., 2014; Mescheder et al., 2017) to map high-dimensional proxies to a probability distribution over the latent space with the goal of obtaining a balanced low-dimensional proxy representation. While the use of VAEs for causal effect estimation is not new (Grari et al., 2022; Kim et al., 2021; Louizos et al., 2017), our framework has two novel components. The first one is in defining and developing the first solution to the problem of contagion estimation with high-dimensional proxies, an important problem in real-world contagion estimation scenarios. The second one is the novel enhancement of VAEs with adversarial networks, similar to matching (Stuart, 2010), which play the important role of addressing the selection bias in treatment groups and is of independent interest for causal effect estimation beyond contagion. In addition to being meaningful for causal inference, this enhancement is crucial for the empirical performance of the estimator.

Through empirical analysis, we demonstrate that state-of-the-art methods for inferring contagion effects are prone to high bias and variance in high-dimensional scenarios, while our proposed approach exhibits remarkable performance improvements.

2. Related Work

Here, we review prior studies that focus on causal inference in observational network data. Ogburn and VanderWeele (2014) explore the role of structural causal models in causal effect estimation in the presence of different types of interference. Shalizi and Thomas (2011) show that in networks formed by latent homophily, contagion, and homophily can be confounded and the causal effect is not always identifiable. Controlling for the cluster assignment of nodes helps with identifiablity (Shalizi and McFowland III, 2016). A recent study deploys negative control outcome and exposure

variables to estimate contagion effects in low-dimensional settings (Egami and Tchetgen Tchetgen, 2024). Our work builds upon this work and focuses on estimating contagion effects when the proxies are high-dimensional.

Recently, a series of methods have been proposed to leverage representation learning to relax the strong ignorability assumption in networked data. Guo et al. (2020) map the network structure and observed node features to a latent representation space to capture the influence of hidden confounders. Veitch et al. (2019) estimate treatment effects using network embeddings by reducing the causal estimation problem to a semi-supervised prediction of the treatments and outcomes and using embedding models for the semi-supervised prediction. Cristali and Veitch (2021) use node embeddings learned from the network structure for estimating contagion effects in a different setting where covariates and the network structure are unobserved. However, these works either do not consider interference (Guo et al., 2020; Veitch et al., 2019), or selection bias (Cristali and Veitch, 2021).

Methods to improve the distribution mismatch between treatment groups include combining weighting with representation learning (Guo et al., 2020; Hassanpour and Greiner, 2019; Li and Fu, 2017), linear ridge regression with representation learning and a discriminator component (Jiang and Sun, 2022). Our approach is distinct in that it balances the proxy representations generated by VAEs with adversarial networks. Several studies have utilized VAEs to estimate proxies for confounding variables in non-network data. Louizos et al. (2017) leverage VAEs to infer latent variables proxies that help with estimating individual treatment effects. Grari et al. (2022) integrate VAEs with an adversarial training component aimed at acquiring a proxy for latent sensitive information, such as gender. Their approach differs from our framework in the sense that adversarial training focuses on guaranteeing the independence of the generated latent space from the unobserved sensitive variable. In contrast, our approach utilizes the discriminator component of an adversarial network to achieve a balance in the representation of treatment and control groups.

3. Problem Description

In this section, we introduce data and causal models, estimand, proxy variable types, and challenges in estimating contagion effects in high-dimensional settings.

3.1. Data model

We assume a graph $G = (\mathbf{V}, \mathbf{E})$ that consists of a set of $|\mathbf{V}|$ nodes and a set of edges $\mathbf{E} = \{e_{ij}\}$, where e_{ij} denotes that there is an edge between node $v_i \in \mathbf{V}$ and node $v_j \in \mathbf{V}$. Each node has an observed n-dimensional vector of attributes, $\mathbf{Z}_i$, unobserved characteristics, $\mathbf{U}_i$, and outcomes in two consecutive time steps, $Y_{i,t-1} \in \mathbb{R}$, and $Y_{i,t} \in \mathbb{R}$. Let $\mathbf{N}_i = \{v_j | v_j \in \mathbf{V} \ \& \ \exists \ e_{ij} \in \mathbf{E}\}$ denote the set of neighbors of node v_i and $\mathbf{A}_i$ be the adjacency vector for node v_i where $A_{ij} = 1$ if $\exists e_{ij}$. For each node, there exists a set of neighbors' hidden characteristics $\mathbf{U}_{ngb}$, a set of neighbors' observed attributes $\mathbf{Z}_{ngb}$, and two sets of neighbors' outcomes $\mathbf{Y}_{ngb,t-1}$ and $\mathbf{Y}_{ngb,t}$.

3.2. Causal Model

Following Egami and Tchetgen Tchetgen (2024), we assume the causal graph depicted in Fig. 1, where the connections ($\boldsymbol{A}_i$) are formed based on the similarity of the unobserved homophilic attributes. Latent variables are represented by dashed circles. Treatment is the set of peer outcomes

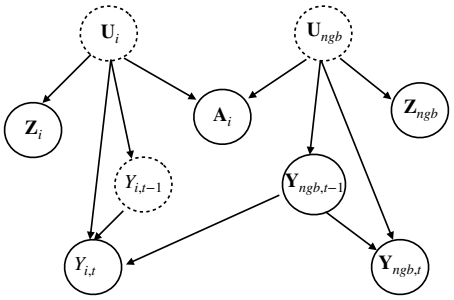


Figure 1: The causal model for the ego-network of ego v_i: $\mathbf{Z}_i$ and $\mathbf{Z}_{ngb}$ are proxies of the hidden confounders. Dashed circles show unobserved homophilic attributes.

$\mathbf{Y}_{ngb,t-1}$ and the outcome is the ego's outcome $Y_{i,t}$. Here, an ego refers to a node v_i whose contagion effects we estimate, and a peer (or neighbor) refers to a node that influences the ego's outcome. The potential outcome of node v_i under contagion effects is defined as the value that $Y_{i,t}$ would take if peer outcomes $\mathbf{Y}_{ngb,t-1}$ had been set to $\mathbf{y}$. The factual outcome $Y_{i,t}^F$ refers to the observed outcome of an individual when $\mathbf{Y}_{ngb,t-1} = \mathbf{y}$ and the counterfactual outcome $Y_{i,t}^{CF}$ shows the unobserved response of an individual when $\mathbf{Y}_{ngb,t-1} \neq \mathbf{y}$.

Given a set of activated neighbors $\hat{\mathbf{N}}_i \subseteq \mathbf{N}_i$, we define $h : \{\mathbf{Y}_{ngb,t-1}\}^{|\mathbf{N}_i|} \rightarrow \{0,1\}$ as a function which maps the neighbors' outcomes at t-1 to a binary value. We consider an ego-network connection model where multiple peers may exist ($|\mathbf{N}_i| \geq 1$). Dyads, i.e., pairs of two individuals, are a special case of the ego-networks model where for every node v_i, $|\mathbf{N}_i| = 1$.

3.3. Contagion Effect Estimation

We define *Individual Contagion Effects (ICE)* as the difference between the outcome of node v_i under two different values for the neighbors' activation $h(\mathbf{Y}_{ngb,t-1})$:

$$\tau_i = Y_{i,t}(h(\mathbf{Y}_{ngb,t-1}) = 1) - Y_{i,t}(h(\mathbf{Y}_{ngb,t-1}) = 0). \tag{1}$$

Our objective is to estimate ACE, which represents the average of ICE over all nodes. In observational data, estimating ICE is challenging because we can never simultaneously observe the factual and counterfactual outcomes of a unit.

A main assumption in causal inference from observational data is strong ignorability or no unmeasured confounding. According to this condition, the potential outcomes of a node are independent of its treatment assignment given its observed attributes (Rosenbaum and Rubin, 1983). In the causal model represented in Fig. 1, strong ignorability holds if:

$$(Y_{i,t}(1), Y_{i,t}(0)) \perp\!\!\!\perp \mathbf{Y}_{ngb,t-1} \mid \mathbf{Z}_i, \mathbf{A}_i. \tag{2}$$

However, conditioning on $\mathbf{A}_i$ introduces a dependence association between unobserved variables $\mathbf{U}_i$ and $\mathbf{U}_{ngb}$ where the unblocked backdoor path $Y_{i,t} \leftarrow \mathbf{U}_i \rightarrow \mathbf{A}_i \leftarrow \mathbf{U}_{ngb} \rightarrow \mathbf{Y}_{ngb,t-1}$ violates the ignorability assumption ($Y_{i,t} \not\perp\!\!\!\perp \mathbf{Y}_{ngb,t-1} | \mathbf{A}_i, \mathbf{Z}_i$) and makes the contagion effects unidentifiable unless proxies are available. We are interested in measuring ACE in the presence of an unobserved confounder, i.e., where the unobserved network confounder is the direct cause of the outcome of an ego and its peers.

3.4. Double Negative Control Proxies

One way to account for latent homophily is by considering proxies for unobserved confounders. Proxies are measurable variables that are correlated with the unobserved variable; conditioning on them enables the identification of the causal effect (Miao et al., 2018). Two groups of common proxies that make the causal effect identifiable in settings with unobserved confounders are: 1) *Negative Control Exposure* (NCE) is a variable that does not causally affect the outcome of interest, and 2) *Negative Control Outcome* (NCO) is a variable that is not causally affected by the treatment of interest. Egami and Tchetgen Tchetgen (2024) demonstrate that leveraging these two types of negative control proxies can enable the identification of contagion effects in networked data with unobserved confounders. In the causal model presented in Fig. 1, a variable $\mathbf{Z}_i$ is considered as an NCO because:

$$\mathbf{Z}_i \perp\!\!\!\perp \mathbf{Y}_{ngb,t-1} | \mathbf{U}_i, \mathbf{U}_{ngb}, \mathbf{A}_i, \tag{3}$$

and variable $\mathbf{Z}_{ngb}$ is considered as an NCE because:

$$\mathbf{Z}_{ngb} \perp\!\!\!\perp (Y_{i,t}, \mathbf{Z}_i) | \mathbf{Y}_{ngb,t-1}, \mathbf{U}_i, \mathbf{U}_{ngb}, \mathbf{A}_i. \tag{4}$$

Various estimators can be employed to infer the causal effect of interest using proxies. One commonly used approach is the *Two-stage Least Squares estimator (TSLS)*. TSLS consists of two stages (Angrist and Imbens, 1995). First, a new variable is constructed using the instrumental variables, serving as a proxy for the unobserved confounders. Then, the estimated values from the first stage replace the unobserved confounders, and an *Ordinary Least Squares (OLS)* regression is performed to estimate the causal effect. Egami and Tchetgen Tchetgen (2024) employ the TSLS estimator to quantify contagion effects by leveraging the NCE and NCO proxies as:

$$Y_{i,t} \sim \mathbf{Y}_{ngb,t-1} + \mathbf{Z}_i | \mathbf{Z}_{ngb} + \mathbf{Y}_{ngb,t-1}, \tag{5}$$

where the coefficient of $\mathbf{Y}_{ngb,t-1}$ shows the estimated ACE.

3.5. Issues with high-dimensional proxies

In the presence of high-dimensional data, the number of model parameters p exceeds the number of data samples n, a problem known as the "Large p Small n" issue in causal effect estimation using regression models (Bernardo et al., 2003). Estimating contagion effects using control proxies can be problematic when the NCO and NCE proxies are high-dimensional because the matrix of model parameters becomes sparse and exhibits a low-rank structure (Deaner, 2021). Including correlated variables in the estimation process increases the variance of the causal estimand (Abadie and Imbens, 2006; De Luna et al., 2011), which adversely affects the performance of the estimator (Chao and Swanson, 2005; Hansen et al., 2008). This issue becomes even more prominent in TSLS estimation, where the computational burden increases with the number of instruments or predictors. The goal of this paper is to solve the following problem:

Problem 1 (Contagion Effect Estimation with High-dimensional Proxies) *Let $G = (\boldsymbol{V}, \boldsymbol{E})$ be a graph evolved by latent homophily with high-dimensional double negative control proxies, associated with nodes. Our goal is to find an estimate of the average contagion effect (ACE) $\hat{\theta}$ that minimizes the expected error between $\hat{\theta}$ and the true value of ACE θ.*

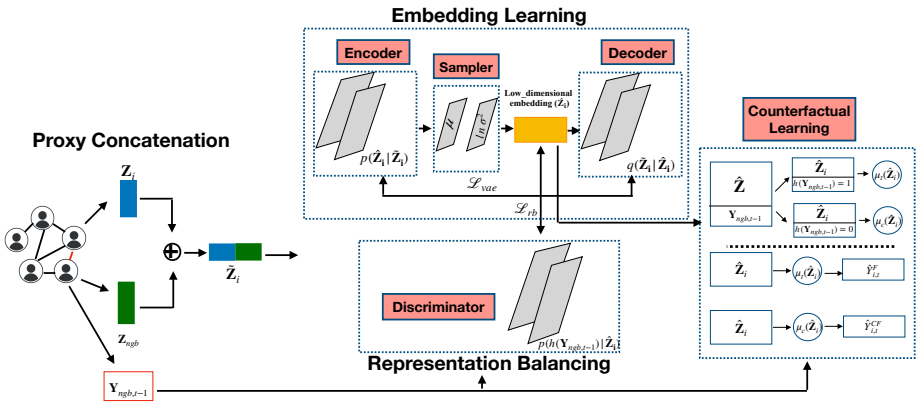


Figure 2: Illustration of the ProEmb framework.

4. Proximal Embedding Framework for Contagion Effect Estimation

To address high dimensionality and selection bias in contagion effect estimation, we introduce the ***Pro**ximal **Emb**eddings (ProEmb)* framework with three main components, shown in Fig. 2. The first component tackles issues of sparsity and high dimensionality by reducing dependent variables to uncorrelated ones, thereby improving estimator optimality (Wang et al., 2014; De Luna et al., 2011). A key technique for this is variational autoencoders (VAEs) (Kingma and Welling, 2014; Rezende et al., 2014), which we carefully adapt to our problem. Embeddings generated by VAEs can vary across different treatment groups and it can lead to confounding biases in estimating causal effects. The second component of ProEmb integrates adversarial networks to update the representation generated by VAEs and improve the distribution shift between the representations of treatment and control proxies. This updated representation is then passed on to the third component, which consists of a counterfactual learning module that measures counterfactual outcomes. To the best of our knowledge, ProEmb is the first method that integrates VAEs, adversarial networks, and meta-learners to improve causal effect estimation more generally and, more specifically, contagion effect estimation in networks with unobserved confounders. Next, we describe each component in detail.

4.1. Embedding learning

The goal of this component is to learn a low-dimensional representation of high-dimensional and sparse proxies while preserving the parts of proxies that are predictive of the outcomes. We assume that the experimenter has classified observed variables into NCO and NCE proxies based on assumptions 3-4. We use VAEs to learn low-dimensional representations for each node's proxies because of their success in dimensionality reduction and ability to both capture the underlying structure of high-dimensional data and regularize the latent space, which helps to prevent overfitting and improve generalization performance (Gregor et al., 2015; Jimenez Rezende et al., 2016).

In order to adapt VAEs to the problem of contagion estimation with high-dimensional proxies, one has to be careful about 1) how to capture latent homophily, 2) how to sample diverse low-dimensional representations from the representation space during training and inference, and 3) how to reconstruct the original high-dimensional proxy vectors, in order to evaluate and improve the performance of the model. ProEmb's VAE addresses these considerations through three parts:

1. *Probabilistic Encoder*. This component transforms high-dimensional proxies into a distribution in the latent space to infer the unobserved confounders. Since $\mathbf{Z}_i$ as an NCO and $\mathbf{Z}_{ngb}$

as an NCE variable are proxies of the unobserved homophilic attributes, we expect to recover latent features by applying a well-trained encoder model to the concatenation of these proxies. Let $\tilde{\mathbf{Z}}_i = \{z_{i,1}, ..., z_{i,n}, z_{ngb,1}, ..., z_{ngb,n}\}$ denote the concatenated vector of proxies $\mathbf{Z}_i = \{z_{i,1}, ..., z_{i,n}\}$ and $\mathbf{Z}_{ngb} = \{z_{ngb,1}, ..., z_{ngb,n}\}$ with dimension n. We use the encoder layer with L fully-connected layers to map proxies $\tilde{\mathbf{Z}}_i$ to low-dimensional latent vector $\mathbf{Z'}_i$:

$$\mathbf{Z'}_i = g(\mathbf{W}_l ... g(\mathbf{W}_1 \tilde{\mathbf{Z}}_i)), \tag{6}$$

where g indicates the activation function (e.g., Relu) and $\{\mathbf{W}_l\}, l \in \{1, ..., L\}$ represents the weight matrices of the fully connected layers of the encoder.

2. *Sampler*. The sampler plays a crucial role in generating latent vectors from the learned distribution in the latent space. These vectors are randomly sampled from the distribution $p(\hat{\mathbf{Z}}_i|\tilde{\mathbf{Z}}_i)$, utilizing the mean and log-variance values obtained from the encoder's output. The latent layer is represented by two sets of neurons: one representing the means of the latent space, and one representing the log-variances, measured as:

$$\mu = \mathbf{W}_\mu \mathbf{Z'_i} + \mathbf{b}^\mu, \;\; ln\, \delta^2 = \mathbf{W}_\delta \mathbf{Z'_i} + \mathbf{b}^\delta, \tag{7}$$

where $\mathbf{b}^\mu$ and $\mathbf{b}^\delta$ are bias vectors. A proxy representation is sampled from the latent space:

$$\hat{\mathbf{Z}}_i \sim p(\hat{\mathbf{Z}}_i|\tilde{\mathbf{Z}}_i) = \mathcal{N}(\mu, exp(ln\;\; \delta^2)). \tag{8}$$

$\hat{\mathbf{Z}}_i$ contains the low-dimensional representation of the proxies, later utilized by the counterfactual learning module for estimating contagion effects.

3. *Probabilistic Decoder*. The decoder attempts to reconstruct the original proxy vector $\tilde{\mathbf{Z}}_i$ from the proxy representation $\hat{\mathbf{Z}}_\mathbf{i}$. The decoder uses $\hat{L}$ fully-connected layers to map $\hat{\mathbf{Z}}_\mathbf{i}$ to $\tilde{\mathbf{Z}}_\mathbf{i}$, i.e.,

$$\mathbf{Z}''_i = f(\hat{\mathbf{W}}_l ... f(\hat{\mathbf{W}}_1 \hat{\mathbf{Z}}_i)), \tag{9}$$

where $\mathbf{Z}''_i$ shows the reconstructed representation, f indicates the activation function, and $\{\hat{\mathbf{W}}_l\}, l \in 1, ..., \hat{L}$ denotes the weight matrix of the fully connected layers.

The loss function of VAEs consists of two main parts: 1) the reconstruction loss which measures the dissimilarity between the original data and the data reconstructed by the VAEs, and 2) The *Kullback–Leibler (KL)* divergence, acting as a regularizer by quantifying disparities between the inferred distribution $p(\hat{\mathbf{Z}}|\tilde{\mathbf{Z}})$ and the desire prior distribution $p(\hat{\mathbf{Z}})$. More specifically:

$$\mathcal{L}_{vae} = \frac{1}{|\mathbf{V}|} \sum_{i=1}^{|\mathbf{V}|} |z'_i - z_i|^2 + KL(p(\hat{\mathbf{Z}}_i|\tilde{\mathbf{Z}}_i)|p(\hat{\mathbf{Z}}_i))_{i=1}^{|\mathbf{V}|}. \tag{10}$$

4.2. Representation balancing

Since the embedding learning models are trained on the factual outcomes and used to predict the counterfactual outcomes, minimizing the error in factual outcomes $Y_{i,t}^F$ does not guarantee the simultaneous error reduction in counterfactual outcomes $Y_{i,t}^{CF}$. In this particular component, we focus on enhancing proxy representation to achieve similarity between the induced distributions for

treated and control nodes. Inspired by Jiang and Sun (2022), we employ the discriminator component of Generative Adversarial Networks (Goodfellow et al., 2014) to address the imbalance proxy representations generated by VAEs.

Let $\mathcal{D} : \hat{\mathbf{Z}}_i \rightarrow \{0, 1\}$ denote the discriminator function, mapping the latent representation $\hat{\mathbf{Z}}_i$ to $h(\mathbf{Y}_{ngb,t-1})$. We train the discriminator to maximize the probability of accurately predicting $h(\mathbf{Y}_{ngb,t-1})$ from the latent representation by optimizing the discriminator loss function:

$$\mathcal{L}_{\mathcal{D}} = \frac{1}{|\mathbf{V}|} \sum_{i=1}^{|\mathbf{V}|} (h(\mathbf{Y}_{ngb,t-1})\ log\ \mathcal{D}(\hat{\mathbf{Z}}_i) + (1 - h(\mathbf{Y}_{ngb,t-1}))\ log(1 - \mathcal{D}(\hat{\mathbf{Z}}_i)).$$

The latent representation $\hat{\mathbf{Z}}_i$ is adjusted to achieve a uniform distribution for $p(h(\mathbf{Y}_{ngb,t-1})|\hat{\mathbf{Z}}_i)$. Given the binary nature of $Y_{ngb,t-1}$, this distribution implies $p(h(\mathbf{Y}_{ngb,t-1}) = 1|\hat{\mathbf{Z}}_i) = p(h(\mathbf{Y}_{ngb,t-1}) = 0|\hat{\mathbf{Z}}_i) = 0.5$. The regularization loss is defined as:

$$\mathcal{L}_{rb} = \frac{1}{|V|} \sum_{i=1}^{|V|} (\mathcal{D}(\hat{\mathbf{Z}}_i) - 0.5)^2. \tag{11}$$

The regularization loss $\mathcal{L}_{rb}$ is then backpropagated to the encoding part of the VAEs, enabling the update of the latent representation $\hat{\mathbf{Z}}_i$ such that the discriminator $\mathcal{D}$ cannot accurately predict $Y_{ngb,t-1}$. This leads to a more balanced and unbiased latent representation for proxies.

4.3. Counterfactual learning

This component focuses on training models to infer the counterfactual outcomes from low-dimensional embeddings of proxies $\hat{\mathbf{Z}}_\mathbf{i} \in R^m$. The factual outcomes are used to train the models. The objective function of this component during training is to minimize the error of the inferred factual outcomes defined as $\frac{1}{n} \sum_{i=1}^{n} \left(\hat{Y}_{i,t} - Y_{i,t}\right)^2$ where $\hat{Y}_{i,t}$ indicates the predicted factual outcome by ProEmb. To make this process more concrete, we demonstrate how our framework would use a common Heterogeneous Treatment Effect (HTE) estimation algorithm, the T-learner. However, our framework could leverage other HTE estimation algorithms as well. T-learner meta-learning algorithm is an example of such estimators and is used to measure Conditional Average Treatment Effect (CATE). A meta-learner is a framework to estimate the Individual Treatment Effects (ITE) using any supervised machine learning estimators known as base-learners (Künzel et al., 2019). In T-learner, two base-learners are trained with treatment (μ_t) and control nodes (μ_c) to estimate the conditional expectations of the outcomes given observed attributes (in our case $\mathbf{Z}$). μ_t and μ_c are employed to predict the counterfactual outcomes of control and treatment nodes, respectively. The difference between the predicted outcomes by treatment and control models shows ITE.

5. Experiments

We evaluate the performance of different methods for contagion effect estimation and demonstrate the applicability of our approach for detecting contagion effects in two real-world datasets.

5.1. Semi-synthetic data generation

In this section, we describe the semi-synthetic datasets we generated for our experiments. It is important to note that the generation doesn't consider embeddings and is therefore not biased towards

an embedding-based solution. We utilize four real-world datasets: 1) *Hateful Users*, which is a sample of 5,000 hateful and normal tweets (Ribeiro et al., 2018), 2) *Stay-at-Home (SAH)*, which is a sample of 30,000 tweets reflecting users' attitudes toward stay-at-home orders during the COVID-19 pandemic (Fatemi et al., 2022), 3) *BlogCatalog*, which is a sample of 5,196 bloggers from an online blog community, and 4) *Flickr* which is a sample of 7,575 users who share photos on Flickr social media platform (Guo et al., 2020).

In the first two datasets, each tweet exhibits a unique distribution over several topics, reflecting the hidden semantic structure of the tweet. We consider the topic distribution of each tweet as the unobserved confounder $\mathbf{U}_i$. To extract the topic distribution of each tweet, we employ Latent Dirichlet Allocation (LDA) (Blei et al., 2003) and measure the coherence score to determine the optimal number of topics. We obtain 20 topics for SAH and 50 topics for the Hateful Users dataset. For the BlogCatalog and Flickr datasets, we follow Guo et al. (2020) and learn 50 topics.

Ego-Network model. Since our causal model relies on the assumption that ties form between nodes by latent homophily, we generate the connections synthetically. We consider data for both ego-networks and dyads. In our network model, we assume that activated neighbors may activate an inactive ego with probability of 0.3. In the dyadic model, each node in the graph is connected to only one other node. More details are provided in Appendix.

Counterfactual model. We generate the outcome of each node in two consecutive time steps. $Y_{i,t-1}$ is generated as:

$$Y_{i,t-1} = \alpha_u \mathbf{U}_i + \epsilon, \tag{12}$$

where $\epsilon \sim \mathcal{N}(0, , 1)$, and α_u is the vector of unobserved confounder coefficients with the size of $\mathbf{U}_i$. We generate the factual and counterfactual outcomes as:

$$Y^F_{i,t} = \beta_u \mathbf{U}_i + \beta_y Y_{i,t-1} + \tau h(\mathbf{Y}_{ngb,t-1}) + \epsilon, \tag{13}$$

$$Y^{CF}_{i,t} = \beta_u \mathbf{U}_i + \beta_y Y_{i,t-1} + \tau (1 - h(\mathbf{Y}_{ngb,t-1})) + \epsilon, \tag{14}$$

where β_u is the unobserved confounder coefficient vector. In our experiments, we utilize both the max() and sigmoid(mean()) functions for h().

5.2. Experimental setup

We consider two types of attributes $\mathbf{Z}_i$. We use bag-of-words (BoW) to represent documents as vectors (vector size of 4,939 for SAH, 13,146 for Hateful Users, 8,189 for BlogCatalg, and 12,047 for Flickr). To understand whether there is value in VAE or using embedding representation is sufficient, we also experiment with simple embeddings derived from BoW for the datasets for which original text is available, BlogCatalog and Flickr. We consider GloVe-200d model (Pennington et al., 2014) and *Bidirectional Encoder Representations from Transformers (BERT)* (Devlin et al., 2019). We further fine-tune the BERT model for $1,000$ steps to obtain new embeddings specific to each dataset (*BERT-ft* model).

To understand the value of ProEmb using different base-learners, we employ three types of base-learners for the T-Learner estimator: 1) ProEmb with Linear Regression (PE-LR), 2) ProEmb with Gradient Boosted Trees (PE-GB), and 3) ProEmb with Multi-layer Perceptrons (PE-NN). We set the embedding dimension of the VAEs as the dimension of the unobserved confounder variable (20 in SAH and 50 in the Hateful Users, BlogCatalog, and Flickr datasets). The hyperparameter tuning is described in the Appendix.

Table 1: RMSE of ACE using BoW feature representation in networked datasets. Numbers following ± indicate the standard deviation of the estimates.

	h=max()					h=mean()				
Dataset	TSLS	CEVAE	NetD	T-GB	PE-GB	TSLS	CEVAE	NetD	T-GB	PE-GB
SAH	2.75 ± 1.35	2.27 ± 0.09	0.87 ± 0.7	0.61 ± 0.13	**0.4 ± 0.1**	5.42 ± 3.6	2.42 ± 0.09	0.85 ±0.45	0.65±0.21	**0.47± 0.24**
Hateful Users	3.28 ± 1.96	2.6± 0.08	0.88± 0.16	0.58± 0.07	**0.41± 0.08**	4.6 ± 2.8	2.51± 0.06	0.66± 0.16	0.62 ± 0.11	**0.47 ± 0.15**
BlogCatalog	207 ± 109	1.83± 0.12	0.38± 0.13	0.27± 0.06	**0.09±0.03**	620±481	3.41±0.25	0.23 ± 0.12	0.19±0.09	**0.11±0.06**
Flickr	128 ± 105	2.12±0.13	0.46±0.27	0.35±0.11	**0.12±0.04**	160±120	2.76±0.18	0.36±0.21	0.28±0.12	**0.13 ± 0.07**

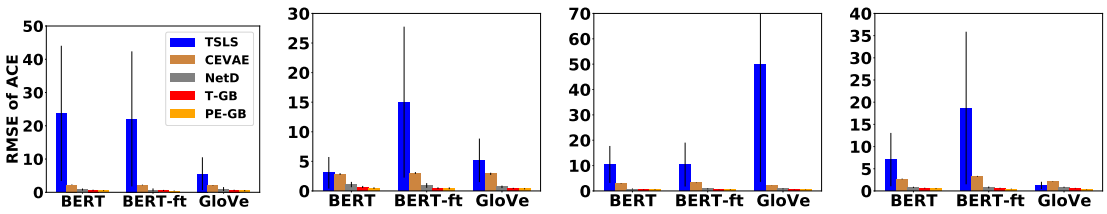
To report the estimation error of different models, we measure the *Root Mean Squared Error (RMSE)* of contagion effects over 10 runs. We consider the BoW or word embedding vector of each user's tweet as an NCO proxy and the BoW or word embedding vector of the peer's tweet as an NCE proxy of the hidden topic distributions. Following Egami and Tchetgen Tchetgen (2024), we set $\beta_y = 0.2$ in Eq. 13 and Eq.14. In addition, we vary the strength of unobserved confounding coefficient vector β_u with two different distributions $\beta_u \sim \mathcal{N}(5, 2)$ and $\beta_u \sim \mathcal{N}(0, 3)$ and $\alpha_u \sim \mathcal{N}(0, 1)$.

Baselines: We compare the performance of ProEmb variants against four different baselines. TSLS is the only existing and state-of-the-art method that makes contagion effects identifiable in network data with unobserved confounders using negative control proxies (Egami and Tchetgen Tchetgen, 2024). *Causal Effect Variational Autoencoder (CEVAE)* is a VAEs-based model for inferring ITE with unobserved confounders (Louizos et al., 2017). Although this model is primarily intended for non-network datasets, we adapt it to network data by concatenating available proxies for the unobserved confounders ($\mathbf{Z}_i$ and $\mathbf{Z}_{ngb}$) as the noisy proxy vector for each node. *Network Deconfounder (NetD)* exploits *Graph Convolutional Networks (GCNs)* to learn representations of hidden confounders by mapping features and network structure into a shared representation space (Guo et al., 2020). We also consider only a T-Learner with Linear Regression (*T-LR*), Gradient Boosted Tree (*T-GB*), and MLP (*T-NN*) as the base-learners.

5.3. Results

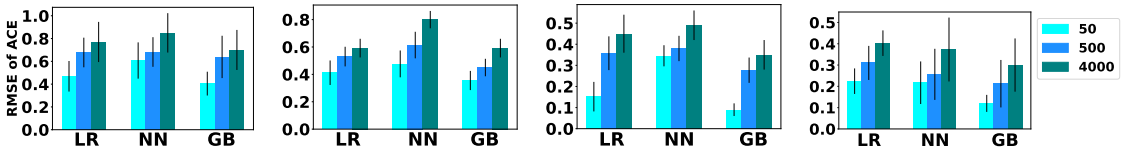
5.3.1. COMPARISON TO ALL BASELINES

In Table 1, we provide a comparison of the best of our three method variants, PE-GB, with all baseline models (*TSLS*, *CEVAE*, *NetD*, and *T-GB*), assessing their performance in estimating ACE using BoW features as proxy variables across all datasets. We employ both the max() and mean() activation functions in the ego-networks model. The results show that in all datasets *TSLS* consistently achieves significantly higher error and variance compared to the other models, especially our proposed method *PE-GB*. This was one of the most surprising results in our study since *TSLS* is a well-established estimation method in causal inference. It's worth noting that CEVAE, a method that utilizes VAEs for causal effect inference in non-network data, demonstrates worse performance when contrasted with our approach, *PE-GB*. This observation highlights the significance of the counterfactual model the discriminator component of our model. As represented in Fig. 5 in the Appendix, we obtain consistent results with datasets on dyads. We also perform an ablation study to assess the impact of integrating VAEs and a discriminator module, confirming that PE-GB outperforms all other models. Further details are provided in the Appendix.



(*a*) SAH, $\beta_u \sim \mathcal{N}(0,3)$ (*b*) SAH, $\beta_u \sim \mathcal{N}(5,2)$ (*c*) Hateful, $\beta_u \sim \mathcal{N}(0,3)$ (*d*) Hateful, $\beta_u \sim \mathcal{N}(5,2)$

Figure 3: RMSE of ACE employing the max() activation function in networked datasets.



(*a*) SAH (*b*) Hateful Users (*c*) BlogCatalog (*d*) Flickr

Figure 4: RMSE of ACE in ProEmb with varying embedding vector dimensions and $\beta_u \sim \mathcal{N}(0,3)$ employing max() activation function. The x-axis represents different types of base-learners used in the ProEmb framework.

5.3.2. SENSITIVITY TO WORD EMBEDDING METHODS

In this experiment, we evaluate the performance of baseline methods using embedding representations instead of BoW and with different unobserved confounding coefficients. As depicted in Fig. 3, our observations consistently align with those obtained using the BoW method in the ego-network model. TSLS exhibits the highest levels of bias and variance while NetD outperforms both TSLS and CAVAE, it doesn't quite reach the level of effectiveness demonstrated by our proposed method, PE-GB. With dyadic data, we observe consistent results.

5.3.3. SENSITIVITY TO THE DIMENSION OF THE EMBEDDING

To investigate the impact of the embedding vector dimension on the estimation error of ProEmb variants, we train the ProEmb models with BoW features and different numbers of VAE embedding dimensions from 20 to 4000. As the number of dimensions increases (Fig. 4), the estimation error also increases for all ProEmb variants, with *PE-GB* achieving the lowest error among all ProEmb variants. The results are consistent across different datasets, with network and dyads ego-network, and when utilizing mean() activation functions.

5.3.4. REAL-WORLD DEMONSTRATION

One of the main challenges in social studies is measuring the strength of peer effects in different domains. As a demonstration of the applicability of our approach to detecting contagion effects in real-world scenarios, we analyze two datasets: 1) French Election, and 2) Peer Smoking. French Election is a Twitter dataset about the 2017 French presidential election (Burghardt et al., 2023). This dataset comprises of 5.3M tweets related to the election, encompassing attitudes, concerns, and emotions expressed in each tweet. Our objective is to measure the extent to which a friend's tweet with a specific emotion or attitude influences a user's decision to post a tweet with the same emotion or attitude. Details on filtering the dataset are in the Appendix.

Since a user may have multiple retweets, we consider the average of each user's tweets' Bag of Words (BoW) representation, which has a vector size of 7,573, as the NCO proxy. Additionally,

we calculate the average of each user's retweet embeddings and use them as the NCE proxy. We employ the BoW representation because our approach yields the lowest estimation error when it is utilized. We use the mean() activation function in this experiment. We report the estimation of the contagion effect using PE-GB because it achieves the best performance in almost all datasets.

We report on four different outcomes: 1) vote against which represents the author's attitude toward voting against a candidate, 2) anger emotion, 3) love emotion, and 4) religious concern:

- Friends' tweets about voting against a candidate have a small negative effect, meaning that they are less likely to tweet about it themselves ($\hat{\theta}_{PE-GB} = -0.013$, P-value=0.001).

- Our method does not reveal a significant contagion effect between users regarding concerns related to religion ($\hat{\theta}_{PE-GB} = -0.002$, P-value = 0) or love emotion ($\hat{\theta}_{PE-GB} = 0.007$, P-value=0.019).

- The anger emotions expressed by peers in their tweets have a small negative impact on the emotional tone of users who retweet those posts, leading to a tendency for opposite emotions to be reflected in their retweets ($\hat{\theta}_{PE-GB} = -0.016$, P-value=0).

The Peer Smoking dataset comprises 1,263 9th and 10th graders from 16 high schools in the Chicago area, observed across three distinct waves (Mermelstein et al., 2009). Our primary objective with this dataset is to assess the influence of peer smoking behaviors during Wave I on an individual's smoking habits during Wave II. We filter the dataset for youth who do not smoke cigarettes in Wave I. We investigate two scenarios: 1) To what extent does an individual's boyfriend or girlfriend's smoking behavior affect their own smoking habits? 2) How does the smoking behavior of the group of friends an individual hang out with influence their own smoking habits? We examine both cigarette and marijuana smoking habits as outcomes. To prepare the dataset for analysis, we transform categorical demographic features using one-hot encoding. Our findings include:

- The cigarette smoking habits of boyfriends or girlfriends have a positive effect on the individual's cigarette smoking behavior ($\hat{\theta}_{PE-GB} = 0.112$, P-value= 0.0005).

- The cigarette smoking habits of the individual's circle of close friends, have a lower but also positive effect on the cigarette behavior of the individual ($\hat{\theta}_{PE-GB} = 0.061$, P-value=0.0001).

6. Conclusion

In this paper, we introduce the Proximal Embeddings (ProEmb) framework for increasing the accuracy of contagion effect estimation in network data affected by latent homophily and selection bias. Our framework comprises three key components: 1) embedding learning, which utilizes variational autoencoders to map high-dimensional proxies to low-dimensional representations and capture latent homophily, 2) representation balancing, which leverages adversarial networks to address the representation mismatch between treatment groups' proxy representations, and 3) counterfactual learning, which employs meta-learners to estimate counterfactual outcomes. Our results demonstrate the compelling performance of the ProEmb framework compared to the baselines in reducing the contagion effect estimation error. A potential future direction is developing a framework to measure multi-hop contagion effects in networks with latent confounders.

Acknowledgments

This research was funded in part by NSF under grant no. 2047899 and DARPA under contract number HR001121C0168. We thank Dr. Mermelstein for providing the Peer Smoking dataset.

References

Alberto Abadie and Guido W Imbens. Large sample properties of matching estimators for average treatment effects. *econometrica*, 74(1):235–267, 2006.

Joshua D Angrist and Guido W Imbens. Two-stage least squares estimation of average causal effects in models with variable treatment intensity. *JASA*, 90(430):431–442, 1995.

Serge Assaad, Shuxi Zeng, Chenyang Tao, Shounak Datta, Nikhil Mehta, Ricardo Henao, Fan Li, and Lawrence Carin. Counterfactual representation learning with balancing weights. In *ICAIS*, 2021.

JM Bernardo, MJ Bayarri, JO Berger, AP Dawid, D Heckerman, AFM Smith, and M West. Bayesian factor regression models in the "large p, small n" paradigm. *Bayesian statistics*, 7:733–742, 2003.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet Allocation. *JMLR*, 2003.

Yann Bramoullé, Habiba Djebbari, and Bernard Fortin. Identification of peer effects through social networks. *Journal of econometrics*, 150(1):41–55, 2009.

Keith Burghardt, Ashwin Rao, Siyi Guo, Zihao He, Georgios Chochlakis, Baruah Sabyasachee, Andrew Rojecki, Shri Narayanan, and Kristina Lerman. Socio-linguistic characteristics of coordinated inauthentic accounts. *arXiv:2305.11867*, 2023.

John C Chao and Norman R Swanson. Consistent estimation with a large number of weak instruments. *Econometrica*, 73(5):1673–1692, 2005.

Nicholas A Christakis and James H Fowler. The spread of obesity in a large social network over 32 years. *New England journal of medicine*, 357(4):370–379, 2007.

Nicholas A Christakis and James H Fowler. The collective dynamics of smoking in a large social network. *New England journal of medicine*, 358(21):2249–2258, 2008.

Irina Cristali and Victor Veitch. Using embeddings to estimate peer influence on social networks. In *NeurIPS*, 2021.

Xavier De Luna, Ingeborg Waernbaum, and Thomas S Richardson. Covariate selection for the nonparametric estimation of an average treatment effect. *Biometrika*, 98(4), 2011.

Ben Deaner. Many proxy controls. *arXiv preprint arXiv:2110.03973*, 2021.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *ACL*, 2019.

D. Eckles, R. Kizilcec, and E. Bakshy. Estimating peer effects in networks with peer encouragement designs. *PNAS*, 2016.

Naoki Egami and Eric J Tchetgen Tchetgen. Identification and estimation of causal peer effects using double negative controls for unmeasured network confounding. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 86:487–511, April 2024.

Zahra Fatemi, Abari Bhattacharya, Andrew Wentzel, Vipul Dhariwal, Lauren Levine, Andrew Rojecki, G Elisabeta Marai, Barbara Di Eugenio, and Elena Zheleva. Understanding stay-at-home attitudes through framing analysis of tweets. *DSAA*, 2022.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, 2014.

Vincent Grari, Sylvain Lamprier, and Marcin Detyniecki. Fairness without the sensitive attribute via causal variational autoencoder. In *IJCAI*, pages 696–702, 2022.

Karol Gregor, Ivo Danihelka, Alex Graves, Danilo Rezende, and Daan Wierstra. Draw: A recurrent neural network for image generation. In *ICML*, 2015.

Ruocheng Guo, Jundong Li, and Huan Liu. Learning individual causal effects from networked observational data. In *WSDM*, pages 232–240, 2020.

Christian Hansen, Jerry Hausman, and Whitney Newey. Estimation with many instrumental variables. *Journal of Business & Economic Statistics*, 26(4):398–422, 2008.

Negar Hassanpour and Russell Greiner. Counterfactual regression with importance sampling weights. In *IJCAI*, 2019.

Song Jiang and Yizhou Sun. Estimating causal effects on networked observational data via representation learning. In *CIKM*, 2022.

Danilo Jimenez Rezende, SM Eslami, Shakir Mohamed, Peter Battaglia, Max Jaderberg, and Nicolas Heess. Unsupervised learning of 3d structure from images. In *NeurIPS*, 2016.

Hyemi Kim, Seungjae Shin, JoonHo Jang, Kyungwoo Song, Weonyoung Joo, Wanmo Kang, and Il-Chul Moon. Counterfactual fairness with disentangled causal effect var. autoencoder. In *AAAI*, 2021.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *stat*, 1050:1, 2014.

Brian V Krauth. Peer effects and selection effects on smoking among canadian youth. *Canadian Journal of Economics/Revue canadienne d'économique*, 38(3), 2005.

Sören R Künzel, Jasjeet S Sekhon, Peter J Bickel, and Bin Yu. Metalearners for estimating heterogeneous treatment effects using machine learning. *PNAS*, 116(10), 2019.

Sheng Li and Yun Fu. Matching on balanced nonlinear representations for treatment effects estimation. *NeurIPS*, 2017.

Christos Louizos, Uri Shalit, Joris M Mooij, David Sontag, Richard Zemel, and Max Welling. Causal effect inference with deep latent-variable models. *NeurIPS*, 2017.

Charles F Manski. Identification of endogenous social effects: The reflection problem. *The review of economic studies*, 60(3):531–542, 1993.

Robin J Mermelstein, Peter J Colvin, and Sven D Klingemann. Dating and changes in adolescent cigarette smoking: does partner smoking behavior matter? *Nicotine & Tobacco Research*, 11 (10):1226–1230, 2009.

Lars Mescheder, Sebastian Nowozin, and Andreas Geiger. Adversarial variational bayes: Unifying variational autoencoders and generative adversarial networks. In *ICML*, 2017.

Wang Miao, Zhi Geng, Eric J Tchetgen Tchetgen, et al. Identifying causal effects with proxy variables of an unmeasured confounder. *Biometrika*, 105(4):987–993, 2018.

Elizabeth L Ogburn and Tyler J VanderWeele. Causal diagrams for interference. *Statistical science*, 29(4):559–578, 2014.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543, 2014.

Gabriel G Piva, Fabiano L Ribeiro, and Angélica S Mata. Networks with growth and preferential attachment: modelling and applications. *Journal of Complex Networks*, 9(1):cnab008, 2021.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *ICML*, 2014.

Manoel Horta Ribeiro, Pedro H Calais, Yuri A Santos, Virgílio AF Almeida, and Wagner Meira Jr. Characterizing and detecting hateful users on twitter. In *ICWSM*, 2018.

Paul R Rosenbaum and Donald B Rubin. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70(1):41–55, 1983.

Uri Shalit, Fredrik D Johansson, and David Sontag. Estimating individual treatment effect: generalization bounds and algorithms. In *ICML*, 2017.

Cosma Rohilla Shalizi and Edward McFowland III. Estimating causal peer influence in homophilous social networks by inferring latent locations. *arXiv:1607.06565*, 2016.

Cosma Rohilla Shalizi and Andrew C Thomas. Homophily and contagion are generically confounded in observational social network studies. *Sociological methods & research*, 40(2):211–239, 2011.

Elizabeth A Stuart. Matching methods for causal inference: A review and a look forward. *Statistical science*, 25(1):1, 2010.

Eric J Tchetgen Tchetgen, Andrew Ying, Yifan Cui, Xu Shi, and Wang Miao. An introduction to proximal causal learning. *arXiv:2009.10982*, 2020.

Russell Torres, Natalie Gerhart, and Arash Negahban. Epistemology in the era of fake news: An exploration of information verification behaviors among social networking site users. *SIGMIS Database*, 49(3):78–97, jul 2018. ISSN 0095-0033.

Tyler J VanderWeele and Weihua An. Social networks and causal inference. *Handbook of causal analysis for social research*, pages 353–374, 2013.

Victor Veitch, Yixin Wang, and David Blei. Using embeddings to correct for unobserved confounding in networks. *NeurIPS*, 32, 2019.

Wei Wang, Yan Huang, Yizhou Wang, and Liang Wang. Generalized autoencoder: A neural network framework for dimensionality reduction. In *CVPR workshops*, 2014.

7. Appendix

7.1. Experiments

7.1.1. SEMI-SYNTHETIC DATA GENERATION

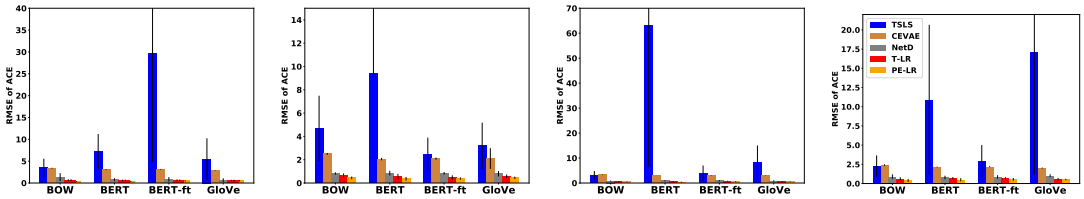
The advantage of considering both dyadic and network data is that it allows us to examine scenarios where a node is influenced by either a single activated neighbor or multiple activated neighbors. By considering dyadic data, we can focus on the interactions between pairs of nodes and gain insights into how one node's activation affects its immediate neighbor. This analysis provides valuable information about the dynamics at the micro-level. On the other hand, analyzing network data allows us to capture the broader influence of multiple activated neighbors on a node. The probability of an edge forming between node v_i and v_j is determined by the cosine similarity of their latent attribute vectors $\mathbf{U}_i$ and $\mathbf{U}_j$. This means that individuals with similar latent attributes are more likely to be connected. In the network model, we aim to generate networks growing based on latent homophily and preferential attachment. We start with $m_0 = 3$ fully connected seed nodes. At each time step, a new node v_j connects to $m = 3$ existing nodes, selected randomly with a probability proportional to the node's degree k_i (Piva et al., 2021):

$$\pi(k_i|v_j) = \frac{\cos(\mathbf{U}_i, \mathbf{U}_j)k_i}{\sum_n \cos(\mathbf{U}_i, \mathbf{U}_n)k_n}. \tag{15}$$

where $\cos(\mathbf{U}_i, \mathbf{U}_j)$ is used as the module of the similarity between node v_j and v_j.

7.2. Experimental setup

To train the VAEs, discriminators, and MLP models, we conduct a hyperparameter search for the learning rate and the number of epochs. The learning rate is searched within the set $\{0.1, 0.01, 0.001, 0.0001\}$, while the number of epochs is searched within $\{10, 30, 50, 70, 100\}$. The best results are achieved with a learning rate of 0.001 and 50 epochs for both models. For the VAEs, we search the number of hidden units of the hidden layers in $\{100, 200, 300\}$ and the number of encoder and decoder layers in $1, 2, 3, 4$. We select a network with 100 hidden nodes, a 3-layer encoder, and a 3-layer decoder with a ReLU activation function. In the discriminator component, after hyperparameter search, we determine that four hidden layers, with linear activation functions, produce the best performance. The output layer utilizes a Sigmoid function. Regarding the MLP, we search for the number of hidden units and the number of fully connected layers. Ultimately, we train an MLP model with two fully connected layers, each containing 125 hidden units.



(*a*) SAH, $\beta_u \sim \mathcal{N}(0,3)$ (*b*) SAH, $\beta_u \sim \mathcal{N}(5,2)$ (*c*) Hateful, $\beta_u \sim \mathcal{N}(0,3)$ (*d*) Hateful, $\beta_u \sim \mathcal{N}(5,2)$

Figure 5: Comparison of RMSE of ACE using various baseline methods in dyadic data. Error bars represent the standard deviation of the estimated effects.

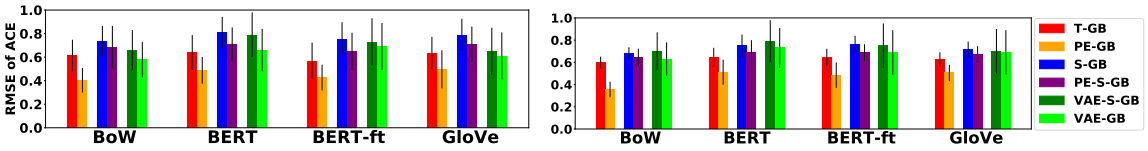


Figure 6: RMSE of ACE in SAH (left) and Hateful Users dataset (right) in ablation study, considering network data and utilizing the max() activation function, with $\beta_u \sim \mathcal{N}(0,3)$.

7.3. Results

7.3.1. SENSITIVITY TO WORD EMBEDDING METHODS

We also evaluate the performance of various methods used to estimate peer contagion effects in the SAH and Hateful Users datasets, based on dyad data and observe that the findings align consistently with the results from the network data (Fig. 5).

7.3.2. ABLATION STUDY

In this experiment, we systematically modify components of the ProEmb framework to investigate their individual importance. In Fig. 6, we denote *S-GB* as the S-Learner estimator, where one GB classifier is trained using both treatment and control nodes (unlike T-learner which has two classifiers). *PE-S-GB* represents ProEmb with both the S-Learner and GB for the counterfactual model. For *VAE-GB*, we utilize a variational autoencoder to reduce the dimensionality of the proxies, followed by a T-Learner with GB for the counterfactual model. In contrast, *VAE-S-GB* employs an S-Learner for the counterfactual model. Our results highlight the importance of integrating VAEs and a discriminator module to mitigate representation mismatches between treatment and control nodes, thereby enhancing estimation accuracy. Our findings demonstrate that T-Learner outperforms the S-Learner as a meta-learner, and *PE-GB* exhibits superior performance compared to all models.

7.3.3. REAL-WORLD DEMONSTRATION

We begin by filtering this dataset to include only tweets and retweets that were posted before the second election date (May 2023), resulting in 4.2M tweets. Then, we construct the retweet network containing 3.1M connections. Following this, we filter the dataset for tweets from users who tweeted at least one tweet after retweeting a tweet. This process yields a total of 13k users with 190k tweets.