Exposure Mapping Function Learning for Peer Effect Estimation

Shishir Adhikari, Sourav Medya, Elena Zheleva

University of Illinois Chicago,
Department of Computer Science
Chicago, IL, USA
{sadhik9, medya, ezheleva}@uic.edu

Abstract

In causal inference involving interacting units (e.g., individuals in a contact network), peer effects quantify how the actions or behaviors of peers (e.g., wearing a mask) affect an individual's outcome (e.g., viral infection). Measuring peer effects involves defining exposure mapping function that outputs peer exposure, a high-level causal variable summarizing peer treatments (or interventions), and estimating the difference in counterfactual outcomes for different peer exposures. Most of the existing approaches for defining exposure mapping functions consider homogeneous influence from peers and use peer exposure based on the fraction of treated peers. There is a growing interest in work that acknowledges heterogeneous influence among units (e.g., due to local neighborhood structure) and captures those influence mechanisms by automatically learning exposure mapping function. Recently, graph neural networks (GNNs) have been extensively used for causal effect estimation in networks, but their use has been mostly limited to automatic feature aggregation and addressing confounding. This work explores the capabilities of GNNs to automatically capture peer influence based on local neighborhood structure. We show GNNs using homogeneous peer exposure or GNNs learning peer exposure naively face difficulty capturing such influence mechanisms. To address this issue, we propose EGONETGNN to learn exposure mapping function by capturing peer influence mechanisms based on local neighborhood structure. We show that our approach reduces the error in estimating peer effects using synthetic network models.

Introduction

Causal inference is crucial for developing artificial intelligence (AI) systems that can make informed decisions by anticipating the consequences of actions or interventions and understanding underlying mechanisms. Decision-making in real-world scenarios often involves complex environments with interacting units, such as an online social network or an epidemiological contact network. In such environments, it is important to assess whether and to what extent a unit's outcome is influenced by the actions, behaviors or interventions of other connected units. For example, we may want to determine whether the vaccination status (treatment) of peers affects an individual's viral infection rate (outcome) in the contact network or whether the political affiliation (treatment) of peers influences one's stance on a policy issue (outcome) in the social network. In causal inference, peer effect measures the difference in a unit's outcome for different treatment regimes of peers (e.g., some contacts vaccinated versus no contact vaccinated, or observed peer political affiliations versus flipped peer affiliations). Therefore, peer effect estimation has become important for policy-making and targeted intervention design in various domains such as healthcare (Barkley et al. 2020), online advertisement (Nabi et al. 2022), and education (Patacchini, Rainone, and Zenou 2017).

Peer effect estimation requires modeling interference between units where the outcome of a unit in the network can be influenced by treatments or outcomes of their peers. The treatments could be either assigned (e.g., with randomized controlled trials (RCTs) or A/B tests) or observed from the data. The critical step in peer effect estimation is defining *peer exposure*, which summarizes treatments of a unit's peers and captures the extent to which peer treatments spill over to the unit. For example, in the contact network, peer exposure is zero if no contacts are vaccinated; however, if some contacts are vaccinated, the peer exposure could depend on the proportion of vaccinated peers or the frequency of contact with vaccinated peers. Peer effect for a unit is measured as the difference in the unit's outcome entailed by two different counterfactual peer exposure conditions. For instance, peer effect, in the contact network, is the difference in outcome, i.e., infection rate, for two exposure conditions, e.g., three fourth of peers vaccinated versus one fourth of peers vaccinated.

Peer exposure is modeled through *exposure mapping* (Aronow and Samii 2017), which is a function that maps peer treatments and other contexts to a representation that summarizes exposure to peer treatments, reduces high dimensionality, and is invariant to irrelevant contexts (e.g., permutation). Usually, domain experts define exposure mapping appropriate to the causal question and the domain of interest. Existing research has mainly considered two types of peer exposure: binary peer exposure (e.g., Bargagli-Stoffi, Tortù, and Forastiere (2020)), which captures if any friends are treated, and *homogeneous peer exposure* (e.g., based on the number or the fraction of treated peers (Ugander et al. 2013; Jiang and Sun 2022; Chen et al. 2024)). Homogeneous

peer exposure assumes all neighbors influence equally and is agnostic to the identity of the treated peers. While binary and homogeneous peer exposure assumptions make possibly unrealistic simplifying assumptions, they are intuitive to interpret and less likely to violate the *positivity assumption* in causal inference. Positivity assumption is a necessary condition for valid causal inference that requires all subpopulations to have a positive probability of receiving any level of treatment. It is imperative to design exposure mappings that capture complex peer influence mechanisms while being easy to interpret and less prone to violation of causal inference assumptions.

There is a growing interest in research that acknowledges heterogeneous influence among units (e.g., due to local neighborhood structure or tie strengths) and designs exposure mapping to capture those influence mechanisms. Prior works have considered exposure mapping that uses the weighted fraction of treated peers based on known edge weights (Forastiere, Airoldi, and Mealli 2021) or known node attributes (Qu et al. 2021). Recent research have considered learning the exposure mapping function that summarize peer exposure conditions. Zhao et al. (2022) have used attention weights to automatically learn weights in the weighted fraction of treated peers based on the similarities of the units' covariates. Ma and Tresp (2021) summarize the covariates of treated peers using a graph neural network (GNN) to learn a peer exposure embedding in addition to homogeneous peer exposure. Ma et al. (2022) employ similar method but for hypergraphs to model group interactions. (Adhikari and Zheleva 2024) use GNNs to learn peer exposure embedding by addressing unknown peer influence mechanisms, but their scope is limited to direct effect estimation, i.e., the effect of a unit's own treatment. Yuan, Altenburger, and Kooti (2021) learn peer exposure embedding based on counts of *causal network motifs* to capture heterogeneous peer influence due to local neighborhood conditions. Causal network motifs are attributed subgraphs with treatment assignments as the attributes. Counting such subgraphs can be computationally expensive, and they may not be able to capture every local structure.

Recently, GNNs have been extensively used for causal effect estimation in networks (Guo, Li, and Liu 2020; Jiang and Sun 2022; Chen et al. 2024), but their use has been mostly limited to automatic feature aggregation and addressing network confounding. While some methods have used exposure mapping function learning with GNNs, they have focused on summarizing covariates of treated peers. To the best of our knowledge, there is no prior work that uses GNNs to learn an exposure mapping function for explicitly capturing peer influence based on local neighborhood structure. We show that GNN-based approaches that solely rely on homogeneous peer exposure or naively learn exposure mapping lack expressiveness in capturing heterogeneous peer influence based on local neighborhood conditions. To address this gap, we propose EGONETGNN, a GNN-based approach to learn an exposure mapping function that is expressive enough to capture peer influence due to local neighborhood structure. Furthermore, EGONETGNN is designed to promote invariance to irrelevant contexts and balanced representation for adding robustness to the downstream peer effect estimation task. Experimental evaluation with synthetic network data shows the advantage of our approach in peer effect estimation when there is heterogeneous influence based on local neighborhood structure.

Peer Effect Estimation Problem Setup

We represent the network as an undirected graph $G = (V, E)$ with a set of $N = |V|$ vertices and a set of edges E. We denote node attributes with $\mathbf{X}$ and edge attributes with $\mathbf{Z}$. Let $T =< T_1, ..., T_i, ..., T_N >$ be a random variable comprising the treatment variables T_i for each node v_i in the network and Y_i be a random variable for v_i's outcome. Let $\boldsymbol{\pi} =< \pi_1, ..., \pi_i, ..., \pi_N >$ be an assignment to T with $\pi_i \in \{0, 1\}$ assigned to T_i.

Let $T_{-i} = T \setminus T_i$ and $\boldsymbol{\pi}_{-i} = \boldsymbol{\pi} \setminus \pi_i$ denote random variable and its value for treatment assignment to other units except v_i. We focus on estimating individual peer effects when the contexts for heterogeneous peer influence depend on local neighborhood structure. Let $\mathcal{Z}_i$ denote effect modifiers, which are contexts responsible for variable effects for the same level of treatment and peer exposure. $\mathcal{Z}_i$ are unknown contexts defined by some functions of node attributes $\mathbf{X}$, edge attributes $\mathbf{Z}$, and network structure G, i.e., $\mathcal{Z}_i = \phi_f(G, \mathbf{X}, \mathbf{Z})$. The *individual peer effect* (IPE) for a unit v_i, denoted as δ_i, for peer treatments $T_{-i} = \boldsymbol{\pi}_{-i}$ versus $T_{-i} = \boldsymbol{\pi}'_{-i}$ and unit's treatment $T_i = \pi_i$ given contexts $\mathcal{Z}_i$ and G is defined as

$$\begin{aligned}\delta_i = E[Y_i(T_i = \pi_i, T_{-i} = \boldsymbol{\pi}_{-i})|\mathcal{Z}_i, G]- \\ E[Y_i(T_i = \pi_i, T_{-i} = \boldsymbol{\pi}'_{-i})|\mathcal{Z}_i, G],\end{aligned} \tag{1}$$

where the counterfactual outcome of unit v_i, $Y_i(T_i = \pi_i, T_{-i} = \boldsymbol{\pi}_{-i})$, expresses the idea that the outcome is influenced by the entire treatment assignment vector $\boldsymbol{\pi}$ due to interference. A common simplifying assumption in causal inference under interference is that the counterfactual outcome of a unit is influenced only by the treatments of units in its neighborhood, rather than all other units in the network (Arbour, Garant, and Jensen 2016; Forastiere, Airoldi, and Mealli 2021; Jiang and Sun 2022; Chen et al. 2024).

Assumption 1 (Neighborhood Interference)**.** *The counterfactual outcome of a unit depends on its immediate neighborhood, i.e.,* $Y_i(T_i = \pi_i, T_{-i} = \boldsymbol{\pi}_{-i}) = Y_i(T_i = \pi_i, T_{\mathcal{N}i} = \boldsymbol{\pi}_{\mathcal{N}_i})$*, where* $T_{\mathcal{N}i}$ *denotes random variable to capture neighborhood assignments* $\boldsymbol{\pi}_{\mathcal{N}_i}$*.*

Next, we assume the counterfactual outcome actually depends on peer exposure manifested due to some underlying mechanisms based on local neighborhood structure. In the social network example, the counterfactual outcome (e.g., stance polarity) of a unit is dependent on its own treatment assignment (e.g., political affiliation) and peer exposure conditions (e.g., political polarity of friends). This assumption can be integrated with the consistency requirement in causal inference that enables equivalence between counterfactual and factual outcomes.

Assumption 2 (Consistency under heterogeneous peer influence due to local neighborhood conditions)**.** *If* $T_i = \pi_i$ *and* $T_{\mathrm{N}_i} = \boldsymbol{\pi}_{\mathcal{N}_i}$*, then* $Y_i(T_i = \pi_i, T_{\mathcal{N}i} = \boldsymbol{\pi}_{\mathcal{N}_i}) = Y_i(T_i = \pi_i, P_{\mathcal{N}i} = \phi_e(\boldsymbol{\pi}_{\mathcal{N}_i}, G, \mathbf{Z})) = Y_i$*, where* $P_{\mathcal{N}i}$ *is a random*

variable to capture peer exposure from neighborhood of v_i and ϕ_e is the exposure mapping function that takes peer treatments, network structure, and edge attributes to output peer exposure embedding.

The goal of our paper is to learn the exposure mapping function ϕ_e and then estimate individual peer effect δ_i as

$$\begin{aligned}\delta_i = E[Y_i(T_i = \pi_i, P_{\mathcal{N}_i} = \phi_e(\boldsymbol{\pi}_{\mathcal{N}_i}, G, \mathbf{Z}))|\mathcal{Z}_i] - \\ E[Y_i(T_i = \pi_i, P_{\mathcal{N}_i} = \phi_e(\boldsymbol{\pi}'_{\mathcal{N}_i}, G, \mathbf{Z}))|\mathcal{Z}_i].\end{aligned} \tag{2}$$

For simplicity of exposition, we set $\boldsymbol{\pi}'_{\mathcal{N}_i}$ to $\vec{\mathbf{0}}$ to capture the peer exposure condition when no neighbors are treated. For example, this setting captures peer effects due to some peers being vaccinated versus none being vaccinated or peer effects due to somewhat diverse peer political affiliation versus peer political affiliations with no diversity.

Next, identification of peer effects involves expressing counterfactual expressions in terms of observational or interventional distribution. Peer effects in Eq. 2 can be expressed in terms of interventional distribution for experiments like A/B tests given the contexts Z_i are not mediators as follows (Pearl 2009):

$$\begin{aligned}\delta_i = E[Y_i|do(T_i = \pi_i, P_{\mathcal{N}_i} = \phi_e(\boldsymbol{\pi}_{\mathcal{N}_i}, G, \mathbf{Z})), \mathcal{Z}_i] - \\ E[Y_i|do(T_i = \pi_i, P_{\mathcal{N}_i} = \phi_e(\boldsymbol{\pi}'_{\mathcal{N}_i}, G, \mathbf{Z})), \mathcal{Z}_i],\end{aligned} \tag{3}$$

where $do(.)$ operator denotes intervention. Notice, $\mathcal{Z}_i$, by definition, are effect modifiers or confounders and do not mediate the treatments. Since the treatments are randomized, we do not need to worry about confounding in experimental data. But we still need to learn ϕ_e and estimate the conditional expectations in Eq. 3. To do this, we need the positivity assumption that requires every possible treatment and peer exposure condition to have non-zero probability.

Assumption 3 (Positivity)**.** *There is non-zero probability of treatment and peer exposure condition, i.e.,* $0 < P(T_i, P_{\mathcal{N}_i}) < 1$*, for every level of* T_i *and* $P_{\mathcal{N}_i}$*.*

For identification of peer effects in observational studies, we need unconfoundedness assumption that restricts the presence of hidden confounders between peer exposure conditions and the outcome as well as treatment and outcome.

Assumption 4 (Unconfoundedness for observational data)**.** *The counterfactual outcomes are independent of treatment and peer exposure conditions given the contexts* $\mathcal{Z}_i$*, i.e.,* $Y_i(T_i = \pi_i, P_{\mathcal{N}_i} = \phi_e(\boldsymbol{\pi}_{\mathcal{N}_i}, G, \mathbf{Z})), Y_i(T_i = \pi_i, P_{\mathcal{N}_i} = \phi_e(\boldsymbol{\pi}'_{\mathcal{N}_i}, G, \mathbf{Z})) \perp \{T_i, P_{\mathcal{N}_i}\}|\mathcal{Z}_i$*.*

With unconfoundedness assumption, we can rewrite Eq. 2 as follows:

$$\begin{aligned}\delta_i = E[Y_i(T_i = \pi_i, P_{\mathcal{N}_i} = \phi_e(\boldsymbol{\pi}_{\mathcal{N}_i}, G, \mathbf{Z}))|T_i, P_{\mathcal{N}_i}, \mathcal{Z}_i] - \\ E[Y_i(T_i = \pi_i, P_{\mathcal{N}_i} = \phi_e(\boldsymbol{\pi}'_{\mathcal{N}_i}, G, \mathbf{Z}))|T_i, P_{\mathcal{N}_i}, \mathcal{Z}_i],\end{aligned} \tag{4}$$

$$\begin{aligned}\delta_i = E[Y_i|T_i = \pi_i, P_{\mathcal{N}_i} = \phi_e(\boldsymbol{\pi}_{\mathcal{N}_i}, G, \mathbf{Z}), \mathcal{Z}_i] - \\ E[Y_i|T_i = \pi_i, P_{\mathcal{N}_i} = \phi_e(\boldsymbol{\pi}'_{\mathcal{N}_i}, G, \mathbf{Z}), \mathcal{Z}_i],\end{aligned} \tag{5}$$

Eq. 5 follows from the consistency assumption and its inference from observational data is valid if the positivity assumption holds. Eq. 3 can also be expressed as Eq. 5 because experimental data is guaranteed to satisfy the unconfoundedness assumption. Thus, peer effects can be estimated using the network structure, node attributes, edge attributes, treatment, and outcome as inputs by learning two functions ϕ_f for $\mathcal{Z}_i$ and ϕ_e for peer exposure along with estimating two conditional expectations.

EGONETGNN: Learning Exposure Mapping Function with GNNs

Figure 1 shows the high-level overview of our peer effect estimation framework with the exposure mapping function learned with the EGONETGNN model. *First,* the attributed network is passed through a standard GNN that approximates feature mapping $\hat{\phi_f}$ to learned feature embedding $\mathcal{Z}_i$ that captures confounders or effect modifiers. *Second,* EGONETGNN approximates the exposure mapping function $\hat{\phi_e}$ by taking the ego network extracted from the attributed network and aggregating the edge attributes and peer treatments to produce peer exposure embedding. The feature embedding, exposure embedding, treatments, and outcomes are passed to an off-the-shelf peer effect estimator to get the peer effects. In this work, we demonstrate an end-to-end exposure mapping learning with EGONETGNN along with the Treatment Agnostic Representation Network (TARNet) (Shalit, Johansson, and Sontag 2017) estimator adopted for peer effect estimation.

Feature Mapping with GNN

The purpose of learning feature mapping is to capture contexts that are potentially confounders or effect modifiers. Capturing confounders ensures the estimates are unbiased and valid, while capturing effect modifiers reduces error in unit-level causal effect estimates. Prior works (Guo, Li, and Liu 2020; Jiang and Sun 2022; Adhikari and Zheleva 2024) have established GNNs are suitable for capturing such contexts in network settings. We employ a GNN similar to the one proposed by Adhikari and Zheleva (2024) because it uses feature mapping considering node and edge attributes. However, our framework is agnostic to the specific GNN architecture, i.e., any GNN (e.g., GCN (Kipf and Welling 2016) or GAT (Veličković et al. 2018)) could be used to extract the feature embedding. Let Θ denote a multi-layer perceptron (MLP) and $||$ denote a concatenation operator. The feature embedding $\mathcal{Z}_i$ is obtained for l-th layer as:

$$\mathcal{Z}_i = \Theta_0(X_i)|| \sum_{j \in \mathcal{N}_i} \Theta_l h_j^{l-1}, \text{ with } h_j^0 = X_j||Z_{ij}$$

, where $\mathcal{N}_i$ denote neighbors of node v_i.

Exposure Mapping with EGONETGNN

The reliability of an exposure mapping function ϕ_e can be assessed in terms of three key properties: 1) expressiveness, 2) invariance, and 3) bounded and balanced representation. The expressiveness property ensures the peer exposure representation $P_{\mathcal{N}_i}$ returned by the function ϕ_e is unique for different relevant contexts, while the invariance property assures the representation $P_{\mathcal{N}_i}$ does not vary due to irrelevant contexts. For example, in a social network, if the underlying peer influence depends on the number of mutual

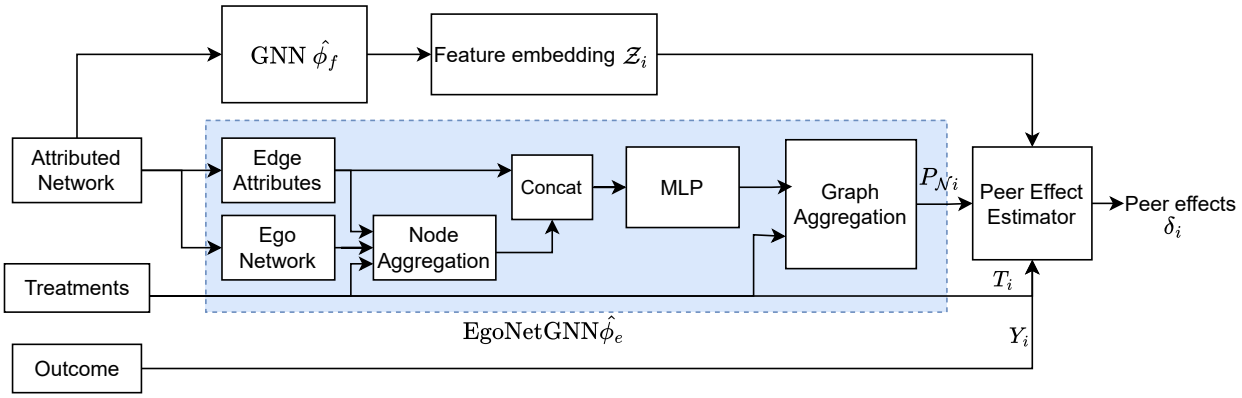

Figure 1: An overview of the proposed EGONETGNN model to learn exposure mapping function for peer effect estimation.

connections, the function ϕ_e is expressive if it can actually capture the number of mutual connections, e.g., by counting the number of triangles. For the above example, the function ϕ_e is invariant to irrelevant contexts if the difference in other features like edge weights does not change the learned representation $P_{\mathcal{N}_i}$. To satisfy the third property of bounded representation, the learned representation $P_{\mathcal{N}_i}$ should be bounded, e.g., between 0 and 1, to reflect no exposure and maximum exposure. Moreover, the representation should be balanced, which means that the learned representation $P_{\mathcal{N}_i}$ should be distributed across the entire bound.

Previous research has investigated the expressiveness of GNNs in terms of their ability to distinguish isomorphic graphs (Xu et al. 2018) or count substructures (Chen et al. 2020). Despite the flexibility of message-passing GNNs (e.g., GCN or GAN), they lack the expressiveness to count subgraphs with cycles like triangles. On the other hand, causal network motifs counts have been shown as reliable features to capture peer exposure due to local neighborhood structure (Yuan, Altenburger, and Kooti 2021). Due to the above limitation of GNNs, they cannot capture closed triad motifs (i.e., triangular motifs). Our proposed method EGONETGNN is designed to make GNNs as expressive or better than the approach of feature extraction by counting motifs. To this end, we transform the node regression task to graph regression by extracting ego networks for each unit. In an ego network, the triangle structures involving an ego node are transformed as edges, which mitigates the limitation of GNNs to capture closed triad motifs. Next, we describe the ego network construction and the architecture of our model.

Ego network construction. First, an ego network $\bar{G}_i(\bar{V}_i, \bar{E}_i)$ is extracted from $G(V, E)$ for each node v_i such that node set $\bar{V}_i$ consists neighbors of v_i, i.e., $\bar{V}_i = \{v_j : e_{ij} \in E \wedge v_j \in V\}$ and edge set $\bar{E}_i$ consists edges between neighbors of v_i, i.e., $\bar{E}_i = \{e_{jk} : e_{jk} \subset E \wedge v_j \in \bar{V}_i \wedge v_k \in \bar{V}_i\}$.

Node aggregation. Next, node attribute $\bar{X}_j$ of node $v_j \in \bar{V}_i$ in the ego network is set using edge attributes of ego node v_i and peer v_j, i.e., $\bar{X}_j = Z_{ij}$. Here, we consider transforming an ego's edge attributes as node attributes of peers in the ego network because the ego node itself is not present in the ego network, and we want to capture the heterogeneous influence due to local neighborhood conditions. The node aggregation is performed in the ego network $\bar{G}_i$for l layers as:

$$h_j^l = \sum_{k \in \mathcal{N}_j} h_k^{l-1}, \text{ with } h_j^0 = T_j || \bar{X}_j.$$

Encoder MLP. Now, the aggregated representation and raw edge attributes are passed into an encoder MLP to extract a low dimensional embedding. The goal of this module is to capture complex mechanism based on the local neighborhood. Formally, the output embedding h_j^{exp} is obtained as follows:

$$h_j^{exp} = ReLU(\Theta_{exp}(tanh(\Theta_{enc}(\bar{X}_j || h_j^l))))$$

Here, the intermediate layer uses $tanh$ activation function to capture mechanisms that may involve proportions (i.e., multiplication or division) and $tanh$ helps the subsequent MLP to learn it by bounding the input.

Graph aggregation. Finally, the low-dimensional representation output from the MLP module is aggregated on the entire ego network. The peer exposure embedding is obtained as follows:

$$P_{\mathcal{N}_i} = \frac{\sum_j (T_j \times h_j^{exp})}{\sum_j h_j^{exp}} || 1 - e^{-\sum_j (T_j \times h_j^{exp})}.$$

We consider two aggregations such that the peer exposure embedding is bounded between zero and one, with zero being the case of no peer exposure. The first aggregation is similar to the fraction of treated peers, but we weight each peer by h_j^{exp} learned by the preceding layer. The second aggregation is analogous to the number of treated peers, except that each peer is weighted by h_j^{exp}.

Peer Effect Estimation with TARNet

The TARNet architecture (Shalit, Johansson, and Sontag 2017) consists of a single MLP with two predictors predicting counterfactual outcomes under treatment and control, i.e.,

$$h_i^{emb} = \Theta_{emb}(\mathcal{Z}_i),$$

$$Y_i(0, P_{\mathcal{N}_i}) = \Theta_{Y(0)}(h_i^{emb}||P_{\mathcal{N}_i}), and$$

$$Y_i(1, P_{\mathcal{N}_i}) = \Theta_{Y(1)}(h_i^{emb}||P_{\mathcal{N}_i}).$$

The peer effect for observed or assigned treatments is obtained as $\hat{\delta}_i = Y_i(0, P_{\mathcal{N}_i}) - Y_i(0, \vec{\mathbf{0}})$ if $T_i = 0$ and $\hat{\delta}_i = Y_i(1, P_{\mathcal{N}_i}) - Y_i(1, \vec{\mathbf{0}})$ if $T_i = 1$. However, one could query for arbitrary peer effect with $Y_i(\pi_i, P_{\mathcal{N}_i}) - Y_i(\pi_i, P'_{\mathcal{N}_i})$.

End-to-end Learning

For the end-to-end learning of the exposure mapping function and the counterfactual outcomes using TARNet, we minimize the standard TARNet loss function to minimize mean square error (MSE) in factual outcome prediction along with the other loss functions designed for EGONET-GNN. These custom loss functions introduce priors to make the learned exposure mapping function stable (Balance loss) and reliable (Bound loss).

TARNet outcome prediction loss. This loss function minimizes the MSE error between predicted outcome and observed outcome, i.e., $L_{pred} = (Y_i - \hat{Y}_i)^2$, where $\hat{Y}_i = Y(1, P_{\mathcal{N}_i})$ if $T_i = 1$ else $Y(0, P_{\mathcal{N}_i})$.

Balance loss. For stability, we use a prior that encourages a balanced distribution of the learned peer exposure embedding. This loss function checks how far the learned peer embedding distribution is from a continuous uniform distribution between 0 and 1, i.e., $L_{bal} = (mean(P_{\mathcal{N}_i}) - 0.5)^2 + (var(P_{\mathcal{N}_i}) - \frac{1}{12})^2 + (range(P_{\mathcal{N}_i}) - 1)^2$. Here, we consider MSE of mean, variance, and range of learned embedding $P_{\mathcal{N}_i}$ against corresponding value of the uniform distribution.

Bound loss. For reliability, we use a prior that peer effects for the instances with no exposure are zero. This loss function checks if peer effects for the instances with $P_{\mathcal{N}_i} = 0$ are zero, i.e., $L_{bound} = (Y_i(\pi_i, P_{\mathcal{N}_i}) - Y_i(\pi_i, \vec{\mathbf{0}}))^2$ if $P_{\mathcal{N}_i} = 0$ else 0. This is required for the reliability of the EGONET-GNN framework and for preserving the interpretation that $P_{\mathcal{N}_i} = 0$ means no peer exposure. Notice the second term $Y_i(\pi_i, \vec{\mathbf{0}})$ represents a counterfactual setting with no peer exposure for all units and it is a significant distribution shift from the observed peer exposure conditions. This loss function aims to mitigate the effect of distribution shifts.

Overall loss. We combine the above losses to obtain overall loss function $\mathcal{L}$ to minimize as

$$\mathcal{L} = L_{pred} + \lambda_{bal} \times L_{bal} + \lambda_{bound} \times L_{bound} + \lambda_{L1} \times ||\mathbf{\Theta}_{gnn}||_1, \tag{6}$$

where $\mathbf{\Theta}_{gnn}$ denote overall parameters in feature mapping GNN and EGONETGNN, and the last term is L_1 loss to promote invariance to irrelevant contexts by preferring sparse weights. λ_{bound} and λ_{bal} are hyperparameters to weigh bound loss and balance loss, respectively.

Experiments and Results

Here, we describe the datasets and experimental setup for the evaluation of our method, EGONETGNN. Then, we present the main takeaways from the results.

Dataset

Similar to other works in causal inference, we rely on synthetic and semi-synthetic data for the evaluation. We adapt the dataset used by Adhikari and Zheleva (2024) for the evaluation of peer effect estimation. We consider two synthetic network models with different edge densities: (1) the Watts Strogatz (WS) network (Watts and Strogatz 1998), which models small-world phenomena, and (2) the Barabási Albert (BA) network (Albert and Barabási 2002), which models preferential attachment phenomena. We generate both networks by fixing the number of nodes to 3000 and controlling the density of edges. For the BA model, the preferential attachment parameter p_{ba} of $[1, 5, 10]$ is used to generate sparse to dense networks, where a new node connects to p_{ba} existing nodes to form the network. For the WS model, we use mean degree parameters p_{ws} to $[0.002N, 0.005N, 0.01N]$ with fixed rewiring probability of 0.5, similar to prior works (Yuan, Altenburger, and Kooti 2021; Adhikari and Zheleva 2024).

Treatment model. The treatment assignments depend on the unit's covariates as well as peer covariates and some edge attribute. We generate treatment T_i for a unit v_i as $T_i \sim \theta(a(\tau_c \mathbf{W}_T \times \frac{\sum_{j \in \mathcal{N}_i} \mathbf{X^c}_j}{\sum_{j \in \mathcal{N}_i} Z^c_{ij}}) + (1 - \tau_c)\mathbf{W}_T \cdot \mathbf{X^c}_i)$, where θ denotes Bernoulli distribution, $a : \mathbb{R} \mapsto [0, 1]$ is an activation function, $\tau_c \in [0, 1]$ controls spillover influence from unit v_i's peers, $\mathbf{X^c} \subset \mathbf{X}$ is a subset of node attributes, $Z^c \in \mathbf{Z}$ is an edge attribute, and $\mathbf{W}_T$ is a weight matrix.

Outcome model. The outcomes depend on unit's treatment, peer treatments based on local neighborhood condition, and confounders. We generate outcome Y_i for a unit v_i as:

$$\begin{aligned} Y_i = &(\delta_{exp} + \delta_{em} \times T_i) \times \phi_e(G, \mathbf{Z}, T_{-i}) + \\ &(\tau_d + \tau_{em} \times \phi_{em}(G, \mathbf{X}, \mathbf{Z})) \times T_i + g(\mathbf{X_c}, Z_c, G) + \epsilon. \end{aligned} \tag{7}$$

Here, the first term $(\delta_{exp} + \delta_{em} \times T_i) \times \phi_e(G, \mathbf{Z}, T_{-i})$ captures peer effects, where $\phi_e(G, \mathbf{Z}, T_{-i})$ captures peer exposure that depends on local neighborhood condition (e.g., the number of mutual connections between treated peers and ego unit) and δ_{exp} and δ_{em} are coefficients controlling magnitude/direction of peer effects. The term $g(\mathbf{X_c}, Z_c, G)$ captures confounding and $\epsilon \sim \mathcal{N}(0, 1)$ is random noise. The remaining term captures direct effect due to unit's own treatment with effect modification by some contexts.

Experimental Setup

We design our experimental setup to answer the following research questions (RQ).

RQ1. How well do methods for peer effect estimation perform when peer exposure mechanisms depend on local neighborhood conditions? RQ1 investigates the performance of peer estimation baseline methods, including those considering homogeneous or heterogeneous peer influence, compared to our method, when peer influence mechanisms are based on local neighborhood conditions. We generate synthetic networks, BA and WS, with low, medium, and high edge density. For each network, we generate treatment

and outcome according to treatment and outcome models above. For the outcome model, we consider three mechanisms for true peer exposure conditions ($\phi_e(G, \mathbf{Z}, T_{-i})$): 1) peer exposure is given by a weighted fraction of treated peers with weights depending on the number of mutual connections; 2) peer exposure is the clustering coefficient between the treated peers; and 3) peer exposure depends on the number of connected components among treated peers. Here, the only challenge is detecting peer effects. Therefore, the coefficients τ_d and τ_{em} in the outcome model (Eq. 7) are set to zero. The coefficients scaling peer effects δ_{exp} and δ_{exp} are set to 20 for the first two mechanisms and 1 for the third mechanism because true peer exposure in the first two are bounded from 0 to 1 while the last one is unbounded.

Evaluation metrics. To evaluate the performance of individual peer effect (IPE) estimation, we use the *Precision in the Estimation of Heterogeneous Effects* (ϵ_{PEHE}) (Hill 2011) metric defined as $\epsilon_{PEHE} = \sqrt{\frac{1}{N}\sum_i(\delta_i - \hat{\delta}_i)^2}$, where δ_i and $\hat{\delta}_i = Y_i(\hat{\pi_i}, P_{\mathcal{N}_i}) - Y_i(\hat{\pi_i}, \vec{\mathbf{0}})$ are true and estimated IPEs. ϵ_{PEHE} (lower better) measures the deviation of estimated IPEs from true IPEs.

Baselines. We compare our proposed approach, EGONETGNN, with state-of-the-art (SOTA) peer estimation methods. The approaches DWR (Zhao et al. 2022) and 1-GNN-HSIC (Ma and Tresp 2021) use neural-network or GNN-based method to learn peer exposure embedding. NetEst (Jiang and Sun 2022) and TNet (Chen et al. 2024) use the fraction of treated peers as peer exposure but the estimator is based on adversarial learning and doubly robust method, respectively, for robustness. We also consider GNN-TARNet-Motifs and GNN-CFR-Motifs approaches that consider manually extracted causal motifs (Yuan, Altenburger, and Kooti 2021) as peer exposure and TARNet and counterfactual regression (CFR) estimators (Shalit, Johansson, and Sontag 2017) as strong baselines. GNN-TARNet-Motifs and GNN-CFR-Motifs serve as references to check whether the exposure mapping function learned by our method is as good as or better than manually extracted causal motifs. We also include INE-TARNet (Adhikari and Zheleva 2024) adapted for peer effect estimation as a baseline, although it was developed for direct effect estimation.

Hyperparameters and model selection. For the experiments, we choose $\lambda_{bound} = 1$ and $\lambda_{bal} = 0.1$ for the loss function. Moreover, we perform grid search hyperparameter tuning by varying GNN learning rate $\{0.02, 0.05\}$ and $\lambda_{L1} = \{0.1, 1\}$, and setting TARNet learning rate to 0.01. A 20% held-out dataset is used for model selection, where model with lowest $L_{pred} + L_{bal} + L_{bound}$ is chosen for reporting. The baselines INE-TARNet, GNN-TARNet-Motifs, and GNN-CFR-Motifs are also tuned similarly. Other baselines are tuned by varing the learning rate $\{0.02, 0.01\}$, keeping other hyperparameters default. DWR is calibrated for 5 epochs to balance representation. We set the output embedding dimension of encoder MLP to 1 giving two-dimensional peer exposure.

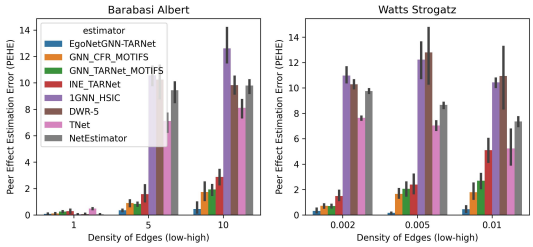

Figure 2: Peer effect estimation error when true peer exposure depends on number of mutual connections. Our method significantly outperforms all baselines showing its capability to count triangles in the ego network.

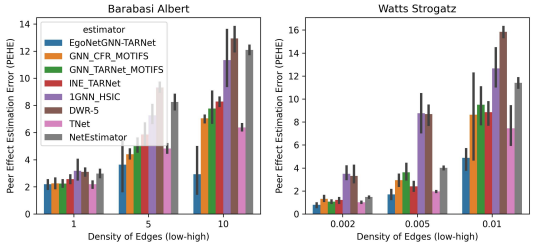

Figure 3: Peer effect estimation error when true peer exposure depends on connected components among treated peers. Our method performs well compared to all baselines when underlying peer exposure mechanism cannot be explained totally with motifs structures only.

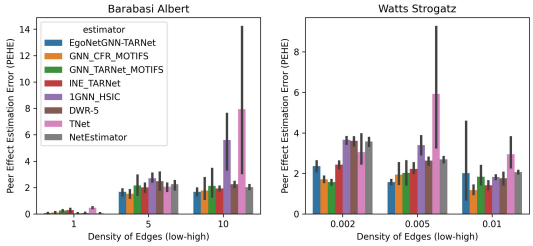

Figure 4: Peer effect estimation error when true peer exposure depends on clustering coefficient among treated peers. Our method is better than or competitive to motif-count based baseline when the underlying peer exposure mechanism can be explained by causal motif counts.

Results

Figures 2 to 4 depict results for the research question and reveal our model performs reliably well in estimating peer effects when peer exposure depends on local neighborhood structure. Each figure shows the performance of peer estimation approaches in terms of the PEHE metric (lower is better) for BA and WS network models, each generated using three different edge density parameters. For each setting, the experiment is repeated for 5 seeds, and we show the mean value and standard deviation as error bars. The $x - axis$ shows the edge density parameters (low to high) used for generating the networks. The performance of our method

EGONETGNN with TARNet estimator is shown as blue bar. It is evident from the figures that our method is better than all of the baselines across most of the settings, and it is competitive with approaches that use causal motif counts.

In Figure 2, our method significantly outperforms all baselines showing its capability to count triangles in the ego network and hence capture the number of mutual connections between an ego and other peers. In Figure 4, our method performs well compared to all baselines when underlying peer exposure mechanism, i.e., based on the number of connected components, cannot be explained totally with motifs structures only. In Figure 3, our method is better than or competitive to motif-count based baselines when the underlying peer exposure mechanism can be explained by causal motif counts. By construction, the BA network with p=1 does not have mutual friends, and the clustering coefficient is zero. So, all methods perform comparatively well for this network in Figure 2 and 4 because there is homogeneous or no peer exposure. By construction, the BA network with p=1 does not have mutual friends, and the clustering coefficient is zero. So, all methods perform comparatively well for this network in Figures 2 and 4 because there is homogeneous or no peer exposure. For other settings, most baselines perform poorly.

Discussion

This work motivates the problem of learning exposure mapping function for peer effect estimation and proposes EGONETGNN for addressing influence due to local neighborhood structure. Our experiments demonstrate increased expressiveness of our method to capture complex local neighborhood exposure conditions. Ongoing work is exploring the generalizability of the method to semi-synthetic data and ablation studies. This work can be applied to the estimation of other network effects like direct effects and total effects. Future work should extend the method to capture generic unknown influence mechanisms for peer effect estimation by addressing the invariance to irrelevant contexts. Another extension should consider relaxing the assumption of neighborhood interference condition.

References

Adhikari, S.; and Zheleva, E. 2024. Inferring Individual Direct Causal Effects Under Heterogeneous Peer Influence. *Machine Learning Journal*.

Albert, R.; and Barabási, A.-L. 2002. Statistical mechanics of complex networks. *Reviews of modern physics*, 74(1): 47.

Arbour, D.; Garant, D.; and Jensen, D. 2016. Inferring network effects from observational data. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 715–724.

Aronow, P. M.; and Samii, C. 2017. Estimating average causal effects under general interference, with application to a social network experiment. *The Annals of Applied Statistics*, 11(4): 1912–1947.

Bargagli-Stoffi, F. J.; Tortù, C.; and Forastiere, L. 2020. Heterogeneous Treatment and Spillover Effects under Clustered Network Interference. *arXiv preprint arXiv:2008.00707*.

Barkley, B. G.; Hudgens, M. G.; Clemens, J. D.; Ali, M.; and Emch, M. E. 2020. Causal inference from observational studies with clustered interference, with application to a cholera vaccine study. *Annals of Applied Statistics*, 14(3): 1432–1448.

Chen, W.; Cai, R.; Yang, Z.; Qiao, J.; Yan, Y.; Li, Z.; and Hao, Z. 2024. Doubly Robust Causal Effect Estimation under Networked Interference via Targeted Learning. In *Forty-first International Conference on Machine Learning*.

Chen, Z.; Chen, L.; Villar, S.; and Bruna, J. 2020. Can graph neural networks count substructures? *Advances in neural information processing systems*, 33: 10383–10395.

Forastiere, L.; Airoldi, E. M.; and Mealli, F. 2021. Identification and estimation of treatment and interference effects in observational studies on networks. *Journal of the American Statistical Association*, 116(534): 901–918.

Guo, R.; Li, J.; and Liu, H. 2020. Learning individual causal effects from networked observational data. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, 232–240.

Hill, J. L. 2011. Bayesian nonparametric modeling for causal inference. *Journal of Computational and Graphical Statistics*, 20(1): 217–240.

Jiang, S.; and Sun, Y. 2022. Estimating Causal Effects on Networked Observational Data via Representation Learning. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 852–861.

Kipf, T. N.; and Welling, M. 2016. Semi-Supervised Classification with Graph Convolutional Networks. In *International Conference on Learning Representations*.

Ma, J.; Wan, M.; Yang, L.; Li, J.; Hecht, B.; and Teevan, J. 2022. Learning causal effects on hypergraphs. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 1202–1212.

Ma, Y.; and Tresp, V. 2021. Causal inference under networked interference and intervention policy enhancement. In *International Conference on Artificial Intelligence and Statistics*, 3700–3708. PMLR.

Nabi, R.; Pfeiffer, J.; Charles, D.; and Kıcıman, E. 2022. Causal inference in the presence of interference in sponsored search advertising. *Frontiers in big Data*, 5.

Patacchini, E.; Rainone, E.; and Zenou, Y. 2017. Heterogeneous peer effects in education. *Journal of Economic Behavior & Organization*, 134: 190–227.

Pearl, J. 2009. *Causality*. Cambridge university press.

Qu, Z.; Xiong, R.; Liu, J.; and Imbens, G. 2021. Efficient Treatment Effect Estimation in Observational Studies under Heterogeneous Partial Interference. *arXiv preprint arXiv:2107.12420*.

Shalit, U.; Johansson, F. D.; and Sontag, D. 2017. Estimating individual treatment effect: generalization bounds and algorithms. In *International Conference on Machine Learning*, 3076–3085. PMLR.

Ugander, J.; Karrer, B.; Backstrom, L.; and Kleinberg, J. 2013. Graph cluster randomization: Network exposure to multiple universes. In *Proceedings of the 19th ACM*

SIGKDD international conference on Knowledge discovery and data mining, 329–337.

Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; and Bengio, Y. 2018. Graph Attention Networks. In *International Conference on Learning Representations*.

Watts, D. J.; and Strogatz, S. H. 1998. Collective dynamics of 'small-world'networks. *nature*, 393(6684): 440–442.

Xu, K.; Hu, W.; Leskovec, J.; and Jegelka, S. 2018. How Powerful are Graph Neural Networks? In *International Conference on Learning Representations*.

Yuan, Y.; Altenburger, K.; and Kooti, F. 2021. Causal Network Motifs: Identifying Heterogeneous Spillover Effects in A/B Tests. In *Proceedings of the Web Conference 2021*, 3359–3370.

Zhao, Z.; Kuang, K.; Xiong, R.; and Wu, F. 2022. Learning Individual Treatment Effects under Heterogeneous Interference in Networks. *arXiv preprint arXiv:2210.14080*.