Should Graph Convolution Trust Neighbors?  
A Simple Causal Inference Method

Fuli Feng$^{12}$, Weiran Huang$^3$, Xiangnan He$^4$, Xin Xin$^5$, Qifan Wang$^6$, Tat-Seng Chua$^2$

$^1$Sea-NExT Joint Lab, $^2$National University of Singapore, $^3$The Chinese University of Hong Kong
$^4$University of Science and Technology of China, $^5$University of Glasgow, $^6$Google US

fulifeng93@gmail.com
Graph has become the default choice for relational data modeling in many IR applications.

**Social network:** Node $\rightarrow$ user; Edge $\rightarrow$ following

**IR applications:** user profiling, rumour detection, targeted advertising, etc.

**User behavior:** Node $\rightarrow$ user/item; Edge $\rightarrow$ click/buy

**IR applications:** recommendation.

Graph-based learning: leverage the graph structure to make better predictions.

- Node features are propagated over the graph structure.
- **Node 3** $\leftarrow \{\text{Node 1, Node 2, Node 4}\} + \text{Node 3}$
- Node prediction is made after the aggregation.

Graph Convolutional Network (GCN)

**Graph Convolutional Network (GCN)**

GCN is being increasingly used in IR applications, ranging from search engines, recommender systems to question-answering systems.

Local Structure Discrepancy Issue

1) Should GCN always trust the neighbors?
   - **Node 1**: Yes
   - **Node 2**: No!

2) The distribution of cross-category edges is not consistent over nodes
   - **Distribution drift**
Existing Solutions

Model Training: Mitigate the impact of the discrepancy issue.

- **Denoising**
  - Edge classification: identify & remove the cross-category edges
  - Spectral filtering: filter out the high-frequency signal in the adjacency matrix

- **Graph attention**
  - Neighbor attention: adjust the contribution of neighbors
  - Hop/layer attention: adjust the contribution of neighbors at difference hops

1) Not easy to be trained well in practice; and 2) Hard to generalize well to testing nodes.
Handling Discrepancy During Model Inference

**Existing method:**
One-pass inference, indiscriminate for Node 1 and Node 2.

**Our expectation:**
Node specific inference, trust neighbor less when making prediction for Node 2.

How does the neighbors affect the prediction?
Causal Effect & Causal Intervention

- **Causal Graph:**
  Graphical models used to encode assumptions about the data-generating process.

- **Intervention on X** [term: do(X=x)]
  Study specific causal relationships between X and the target variable.
  Randomized controlled trial.
  In graph: Cut off the paths that point into X

- **Causal Effect:**
  \[ P(Y | \text{do}(X=x)) - P(Y | \text{do}(X=x_{ref})) \]
  measures the expected increase in Y as the treatment changes from \( X = x \) to \( X = x_{ref} \)

---

Causal GCN Inference Mechanism

A causal view of generating node prediction

Should GCN trust the neighbors?

- Unmeasured confounder
- Drift

Training a simple binary classifier (choice model) to make choice

\[ \text{choose}(\hat{y}, \hat{y}^s) \]

Original Post intervention prediction prediction
Factors for Making Choice & Causal Uncertainty

Factors for making choice: prediction confidence, category transition, causal uncertainty.

- Causal effect of $\mathcal{N}$
  
  $e = f(x, \mathcal{N}(x) | \hat{\theta}) - f(x, do(N = \emptyset) | \hat{\theta})$,  
  
  $= f(x, \mathcal{N}(x) | \hat{\theta}) - f(x, \emptyset | \hat{\theta})$,  
  
  $= \hat{y} - \hat{y}^s$.

- Variance of causal effect
  
  $\nu = var(\{f(x, \mathcal{N}(x)_k | \hat{\theta}) | k \leq K\})$,

$\mathcal{N}(x)_k$ is a sample of the neighbors by randomly dropping edges.
A New Schema for Training GCN

Algorithm 1 Applying CGI to GCN

**Input:** Training data $X, A, Y$.

/* Training */

1. Optimize Equation (4), obtaining GCN ($\hat{\theta}$);  \hfill \triangleright \text{GCN training}
2. Construct $\mathcal{D}$;  \hfill \triangleright \text{Causal intervention}
3. Optimize Equation (10), obtaining choice model ($\hat{\eta}$); \hfill \triangleright \text{CGI training}
4. Return $\hat{\theta}$ and $\hat{\eta}$.

/* Testing */

5. Calculate $f(x, N(x)|\hat{\theta})$; \hfill \triangleright \text{Original prediction}

- Training for the choice model

- Two-pass GCN inference

- Choice model inference

6. Calculate $f(x, \emptyset|\theta)$; \hfill \triangleright \text{Post-intervention prediction}

   Calculate final classification with Equation (8);
EXP1: Semi-supervised Setting (Discrepancy)

<table>
<thead>
<tr>
<th>Dataset</th>
<th>CiteSeer(10%)</th>
<th>CiteSeer(30%)</th>
<th>CiteSeer(50%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPNP</td>
<td>71.0%</td>
<td>64.4%</td>
<td>64.2%</td>
</tr>
<tr>
<td>APPNP_Self</td>
<td>65.1%</td>
<td>62.9%</td>
<td>64.3%</td>
</tr>
<tr>
<td>APPNP_CGI</td>
<td><strong>71.8%</strong></td>
<td><strong>66.9%</strong></td>
<td><strong>68.6%</strong></td>
</tr>
<tr>
<td>RI</td>
<td>1.1%</td>
<td>3.9%</td>
<td>7.2%</td>
</tr>
</tbody>
</table>

Table 1: Performance of APPNP’s original prediction, post-intervention prediction, and CGI prediction on the three synthetic datasets w.r.t. accuracy. RI means the relative improvement over APPNP achieved by APPNP_CGI.

- The causal GCN inference mechanism indeed mitigates the discrepancy issue.
- The relative improvement increases when facing more severe discrepancy.
### EXP2: Semi-supervised Setting (Random Split)

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Cora</th>
<th>Citeseer</th>
<th>Pubmed</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPNP</td>
<td>81.8%</td>
<td>72.6%</td>
<td>79.8%</td>
</tr>
<tr>
<td>APPNP_Self</td>
<td>69.3%</td>
<td>66.5%</td>
<td>75.9%</td>
</tr>
<tr>
<td>APPNP_Ensemble</td>
<td>78.0%</td>
<td>71.4%</td>
<td>79.2%</td>
</tr>
<tr>
<td>APPNP_CGI</td>
<td>82.3%</td>
<td>73.7%</td>
<td>81.0%</td>
</tr>
<tr>
<td>RI</td>
<td>5.5%</td>
<td>2.8%</td>
<td>2.3%</td>
</tr>
</tbody>
</table>

**Table 2: Performance of APPNP with different inference mechanisms on three semi-supervised node classification datasets w.r.t. the classification accuracy. RI means the relative improvement of APPNP_CGI over APPNP_Ensemble.**

- The causal GCN inference mechanism is effective in the conventional setting.
  - Insufficient labels: 20-shot per class
EXP3: Full supervised Setting

<table>
<thead>
<tr>
<th>RoBERTa (768)</th>
<th>JKNet</th>
<th>MLP</th>
<th>DAGNN</th>
<th>APPNP</th>
<th>APPNP_Self</th>
<th>APPNP_Ensemble</th>
<th>APPNP_CGI</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>75.59%</td>
<td>72.26%</td>
<td>74.93%</td>
<td>75.74%</td>
<td>73.43%</td>
<td>76.26%</td>
<td>76.52%</td>
</tr>
<tr>
<td></td>
<td>75.54%</td>
<td>72.26%</td>
<td>74.83%</td>
<td>75.61%</td>
<td>73.38%</td>
<td>75.86%</td>
<td>76.07%</td>
</tr>
</tbody>
</table>

Table 3: Performance comparison under full-supervised settings. We use bold font and underline to highlight the best and second best performance under each setting.

- The causal GCN inference mechanism is effective in the conventional setting.
  - Chronological split
EXP4: Causal Uncertainty

- The causal uncertainty reveals the correctness of a prediction
- Causal uncertainty is complementary to classification confidence
Conclusion & Future Work

- Solving the local structure discrepancy issue during GCN inference
- The one-pass model inference might be insufficient
- Incorporating causal intervention is beneficial
- More causal inference techniques, e.g., counterfactual inference
- Eliminating the bias in GCN, e.g., degree bias