Signed Graph Neural Network with Latent Groups

Haoxin Liu*  Ziwei Zhang*  Peng Cui  Yafeng Zhang  Qiang Cui  Jiashuo Liu  Wenwu Zhu
Tsinghua U  Tsinghua U  Tsinghua U  Meituan  Meituan  Tsinghua U  Tsinghua U

* Equal contribution
Besides Positive Relationships

Negative relationships also play an important role

- Foes
- Disagreements
- Boycotts
- Dislike
- Distrust
- ......

How to model positive and negative relationships *simultaneously*?
Modeling: Signed Graph

Assign signs to links

unsigned graph

unsigned link
signed positive link
signed negative link

signed graph
Challenges

Unsinged graph representation learning methods not suitable

- Fundamental hypotheses in unsinged graphs fail
  - e.g. homophily

- Complex semantic relationships
  - e.g. \( (2^{k+1} - 1) \) types of relationships considering k-order neighbors
  - e.g. 38 popular signed motifs\(^2\)

Existing Signed GRL Methods

- Based on balance theory
  - SIDE, WWW 2018
  - BESIDE, CIKM 2018
  - SGCN, ICDM 2018
  - SNEA, AAAI 2020
  - ASiNE, SIGIR 2020
  - SIHG, TKDE 2020
  - SHCN, TOIS 2020
  - SGDNN, Arxiv 2020
  - SGDN, Arxiv 2020

- Not based on balance theory
  - SLF, KDD 2019
  - ROSE, WWW 2020

Signed GNN (using message passing mechanism)

Most signed GRL methods are based on balance theory
Almost all signed GNN methods are based on balance theory
The Most Popular Solution: Balance Theory

- Balance theory: a well-known social theory
  - “The friend(foe) of my friend is my friend(foe)”
  - “The friend(foe) of my foe is my foe(friend)”

- Signed GRL methods based on balance theory
  - Signed Network Embedding
    - Signed random walk based on balance theory
  - Signed Graph Neural Network
    - Aggregate layer by layer based on balance theory

---

Limitations of Balance Theory

Theoretical Analysis:

- Balance theory essentially equals to the two-conflict-groups assumption
  - Theorem A\(^1\)
    Let \( G \) be a signed undirected complete graph in which each triangle has an odd number of positive edges. Then the nodes of \( G \) can be partitioned into two sets \( A \) and \( B \) (where one of \( A \) or \( B \) may be empty), such that all edges within \( A \) and \( B \) are positive, and all edges with one end in \( A \) and the other in \( B \) are negative.
  - Theorem B\(^2\)
    A s-graph \( G \) is balanced if and only if its point set \( E \) can be partitioned into two disjoint subsets \( E_1, E_2 \), in such a way that each positive line of \( G \) joints two points of the same subset and each negative line joints two points of different subsets.

Balance theory is too ideal

---

Beyond Balance Theory: K-group Theory

- Step 0: two conflict groups (balance theory)

- Step 1: k conflict groups

- Step 2: k-group theory
  - K groups with arbitrary relations between groups
    - Negative
    - Positive
    - Netural
    - ......
Beyond Balance Theory: K-group Theory
Final Assumption & Overall Framework

- Final Assumption: Combine Global and Local View
  - Global: k-group theory
    - Capture underlying k groups with arbitrary relations between groups
  - Local: without any assumption
    - Give the model more flexibility to accommodate other factors
      - Micro-structures within groups
      - Influenced by node features
      - Individual heuristic information
        - e.g., always forms negative relations
      - Tolerate randomness/noise
      - ....

- Proposed: Group Signed GNN
  - A dual architecture
    - Global signed convolution module
    - Local signed convolution module
Overall Comparison

- **Assumptions**
  - 2 conflict groups (Balance Theory)
  - K conflict groups
  - K groups with arbitrary relation
  - Global: K groups with arbitrary relations
  - & Local: other flexible factors without any assumption

- **Methods**
  - 2 conflict groups (Balance Theory)
  - K conflict groups
  - Global model of our signed GRL model
  - Our signed GRL model: Group Signed GNN
Goal
- Discover latent community structure based on k-group theory

Challenges
- Model complex relationships between communities
- Represent nodes in a view of the groups
- Scalable
Solutions
- Denote a learnable embedding matrix $Z_C = [Z_{C_1}, Z_{C_2}, ..., Z_{C_K}] \in \mathbb{R}^{K \times d_C}$ for $K$ groups
- Complex relationships are freely modeled in the hidden space
- Node global embeddings $Z_G$ are represented as a linear combination of the group embeddings i.e., $Z_G = SZ_C$, where assignment matrix $S \in \mathbb{R}^{N \times K}$ is learned
Global Signed Convolution Module: Details

- Model: a novel prototype GNN

- Assignment Initialization
  - Generate the initial assignment probability matrix $S \in \mathbb{R}^{N \times K}$

$$X'_v = \text{MLP}(X_v) \quad (1)$$

$$Q^{(0)}_{v,C_i} = Z^T_{C_i} X'_v, S^{(0)}_{v,C_i} = \frac{\exp \left( Q^{(0)}_{v,C_i} \right)}{\sum_{j=1}^{K} \exp \left( Q^{(0)}_{v,C_j} \right)} \quad (2)$$
Global Signed Convolution Module: Details

- Model: a novel prototype GNN

- Assignment Propagation and Aggregation

\[ p_v^{(m)} = \sum_{u \in \mathcal{N}_v^+} S_u^{(m-1)}, \quad n_v^{(m)} = \sum_{u \in \mathcal{N}_v^-} S_u^{(m-1)} \]

*sum aggregator*
Global Signed Convolution Module: Details

- **Model**: a novel prototype GNN

- **Assignment Updating**

\[
S_v^{(m)} = \mathcal{F}^{(m)} \left( S_v^{(m-1)}, p_v^{(m)}, n_v^{(m)} \right) \\
= \text{softmax} \left( \sigma \left( \left[ S_v^{(m-1)}, p_v^{(m)}, n_v^{(m)} \right] W_G^{(m)'}, W_G^{(m)} \right) \right)
\]
Global Signed Convolution Module: Details

- Model: a novel prototype GNN

- Obtaining Global Representations \( Z_G \)
  - Repeating the 2nd and 3rd steps for \( M_G \) time, i.e., adopting \( M_G \) message-passing layers
  - Obtaining the final assignment matrix \( S \)
    \[ S = S^{(M_G)} \]
  - Using the linear combination of group embeddings
    \[ Z_G = SZ_C \]
Goal

- Give the model more flexibility to accommodate other factors
- Without any assumption

Solution

- Treat self connections, positive links, and negative links as three relations

Model

- A multi-relational GNN
- Conduct message-passing for $M_L$ layers, the final node local embeddings are $Z_L$

$$h_v^{(m)} = \left[ \sum_{u \in N_v^+} z_u^{(m-1)}, \sum_{u \in N_v^-} z_u^{(m-1)} \right], \quad z_v^{(m)} = \sigma \left( \left[ z_v^{(m-1)}, h_v^{(m)} \right] W_L^{(m)} + b_L^{(m)} \right)$$
Proposed Model: Group Signed GNN (GS-GNN)

- **Final Embedding**
  - Concat global embeddings $Z_G$ and local embeddings $Z_L$
  - $Z = [Z_G, Z_L]$

- **Objective function**
  - e.g. for link sign prediction task, i.e., predicting the polar of the given links
  - $\mathcal{L} = -\frac{1}{|\mathcal{E}^+ \cup \mathcal{E}^-|} \left( \sum_{(u,v) \in \mathcal{E}^+} \log p(u, v) + \sum_{(u,v) \in \mathcal{E}^-} (1 - \log p(u, v)) \right) + \lambda \mathcal{L}_{\text{reg}}$
Experimental Setting: Datasets, Task & Metrics

- **Datasets**
  - Four public real-world signed graphs
    - Bitcoin-Alpha, Bitcoin-OTC: two signed graphs extracted from bitcoin trading platforms
    - Slashdot: a technology-related news website
    - Epinions: a consumer review site
  - Synthetic datasets

- **Task**
  - Link sign prediction

- **Evaluation Metrics**
  - Four metrics
    - AUC: area under curve
    - Macro-F1: macro-averaged F1 score
    - Micro-F1: micro-average F1 score
    - Binary-F1: binary average F1 score
  - Higher value indicates better performance

### Table 1: The Statistics of Real-world Datasets

<table>
<thead>
<tr>
<th>Datasets</th>
<th># Nodes</th>
<th># Links</th>
<th># Positive Links (Ratios)</th>
<th># Negative Links (Ratios)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bitcoin-Alpha</td>
<td>3,775</td>
<td>14,120</td>
<td>12,721 (90.09%)</td>
<td>1,399 (9.91%)</td>
</tr>
<tr>
<td>Bitcoin-OTC</td>
<td>5,875</td>
<td>21,489</td>
<td>18,230 (84.83%)</td>
<td>3,259 (15.17%)</td>
</tr>
<tr>
<td>Slashdot</td>
<td>37,626</td>
<td>419,072</td>
<td>313,543 (74.82%)</td>
<td>105,529 (25.14%)</td>
</tr>
<tr>
<td>Epinions</td>
<td>45,003</td>
<td>616,031</td>
<td>513,851 (83.41%)</td>
<td>102,180 (16.59%)</td>
</tr>
</tbody>
</table>
Experimental Setting: Baselines

- **Baselines**
  - Signed graph clustering method
    - SPONGE (AISTATS 2019)
  - Signed network embedding based on balance theory
    - SIDE (WWW 2018)
  - Signed network embedding not based on balance theory
    - SLF (KDD 2019)
  - Unsigned GNN
    - GCN (ICLR 2017)
  - Signed GNN based on balance theory
    - SGCN (ICDM 2018)
    - SNEA (AAAI 2020): +attention
    - SGDN (Arxiv 2020): +diffusion

- **Our method: GS-GNN**
Results on Synthetic Dataset

Question 1
- Can GS-GNN fully utilize the k-group theory and discover the underlying structure of signed graphs?

Synthetic dataset
- Using the signed stochastic block model (SSBM) to generate $K_S$ conflict groups with random noise

Results of Macro-F1

<table>
<thead>
<tr>
<th>Assumption</th>
<th>Method</th>
<th>$K_S = 2$</th>
<th>$K_S = 3$</th>
<th>$K_S = 4$</th>
<th>$K_S = 5$</th>
<th>$K_S = 6$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Balance Theory</td>
<td>SGCN</td>
<td>0.442</td>
<td>0.398</td>
<td>0.362</td>
<td>0.334</td>
<td>0.357</td>
</tr>
<tr>
<td></td>
<td>SGDNN</td>
<td>0.791</td>
<td>0.682</td>
<td>0.612</td>
<td>0.530</td>
<td>0.495</td>
</tr>
<tr>
<td>K-Group</td>
<td>SPONGE</td>
<td>0.983</td>
<td>0.989</td>
<td>0.990</td>
<td>0.990</td>
<td>0.989</td>
</tr>
<tr>
<td>$K=K_S$</td>
<td>GS-GNN</td>
<td>0.984</td>
<td>0.991</td>
<td>0.991</td>
<td>0.989</td>
<td>0.982</td>
</tr>
<tr>
<td>$K=2$</td>
<td>SPONGE</td>
<td>0.983</td>
<td>0.853</td>
<td>0.749</td>
<td>0.670</td>
<td>0.600</td>
</tr>
<tr>
<td></td>
<td>GS-GNN</td>
<td>0.984</td>
<td>0.991</td>
<td>0.988</td>
<td>0.984</td>
<td>0.889</td>
</tr>
<tr>
<td>$K=6$</td>
<td>SPONGE</td>
<td>0.463</td>
<td>0.662</td>
<td>0.848</td>
<td>0.940</td>
<td>0.989</td>
</tr>
<tr>
<td></td>
<td>GS-GNN</td>
<td>0.986</td>
<td>0.988</td>
<td>0.990</td>
<td>0.989</td>
<td>0.980</td>
</tr>
</tbody>
</table>

Conclusion
- Demonstrate the superiority of GS-GNN in utilizing the k-group theory
- GS-GNN even outperforms the SPONGE
Results on Real Graphs

Question 2
- How does GS-GNN perform on different real graphs which is usually complicated, compared with other state-of-the-art signed graphs representation learning methods?

Results
- Observations
  - SPONGE fails on real-world graphs
  - Unsigned GNN outperforms other baselines in some cases
  - GS-GNN consistently outperforms all the baselines on all datasets with all evaluation metrics
  - 5.14%~10.13% improvement of Macro-F1 indicates that GS-GNN can better model the negative links

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Metric</th>
<th>SPONGE</th>
<th>SLF</th>
<th>SIDE</th>
<th>GCN</th>
<th>SGCN</th>
<th>SNEA</th>
<th>SGDN</th>
<th>GS-GNN</th>
<th>Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bitcoin-Alpha</td>
<td>AUC</td>
<td>0.513</td>
<td>0.847</td>
<td>0.797</td>
<td>0.806</td>
<td>0.858</td>
<td>0.866</td>
<td>0.840</td>
<td>0.893</td>
<td>+3.12%</td>
</tr>
<tr>
<td></td>
<td>Macro-F1</td>
<td>0.504</td>
<td>0.668</td>
<td>0.665</td>
<td>0.546</td>
<td>0.706</td>
<td>0.727</td>
<td>0.663</td>
<td>0.793</td>
<td>+9.08%</td>
</tr>
<tr>
<td></td>
<td>Micro-F1</td>
<td>0.901</td>
<td>0.819</td>
<td>0.824</td>
<td>0.902</td>
<td>0.864</td>
<td>0.873</td>
<td>0.894</td>
<td>0.930</td>
<td>+3.10%</td>
</tr>
<tr>
<td></td>
<td>Binary-F1</td>
<td>0.948</td>
<td>0.892</td>
<td>0.896</td>
<td>0.948</td>
<td>0.921</td>
<td>0.926</td>
<td>0.942</td>
<td>0.961</td>
<td>+1.37%</td>
</tr>
<tr>
<td>Bitcoin-OTC</td>
<td>AUC</td>
<td>0.700</td>
<td>0.873</td>
<td>0.828</td>
<td>0.845</td>
<td>0.871</td>
<td>0.863</td>
<td>0.863</td>
<td>0.915</td>
<td>+4.81%</td>
</tr>
<tr>
<td></td>
<td>Macro-F1</td>
<td>0.644</td>
<td>0.735</td>
<td>0.713</td>
<td>0.675</td>
<td>0.754</td>
<td>0.760</td>
<td>0.734</td>
<td>0.837</td>
<td>+10.13%</td>
</tr>
<tr>
<td></td>
<td>Micro-F1</td>
<td>0.763</td>
<td>0.828</td>
<td>0.820</td>
<td>0.875</td>
<td>0.850</td>
<td>0.858</td>
<td>0.871</td>
<td>0.920</td>
<td>+5.14%</td>
</tr>
<tr>
<td></td>
<td>Binary-F1</td>
<td>0.850</td>
<td>0.892</td>
<td>0.889</td>
<td>0.928</td>
<td>0.908</td>
<td>0.914</td>
<td>0.926</td>
<td>0.952</td>
<td>+2.59%</td>
</tr>
<tr>
<td>Slashdot</td>
<td>AUC</td>
<td>0.500</td>
<td>0.888</td>
<td>0.820</td>
<td>0.819</td>
<td>0.873</td>
<td>0.888</td>
<td>0.887</td>
<td>0.916</td>
<td>+3.15%</td>
</tr>
<tr>
<td></td>
<td>Macro-F1</td>
<td>0.432</td>
<td>0.772</td>
<td>0.725</td>
<td>0.670</td>
<td>0.760</td>
<td>0.769</td>
<td>0.769</td>
<td>0.812</td>
<td>+5.18%</td>
</tr>
<tr>
<td></td>
<td>Micro-F1</td>
<td>0.752</td>
<td>0.812</td>
<td>0.773</td>
<td>0.797</td>
<td>0.802</td>
<td>0.812</td>
<td>0.838</td>
<td>0.865</td>
<td>+3.22%</td>
</tr>
<tr>
<td></td>
<td>Binary-F1</td>
<td>0.861</td>
<td>0.867</td>
<td>0.840</td>
<td>0.875</td>
<td>0.859</td>
<td>0.868</td>
<td>0.896</td>
<td>0.915</td>
<td>+2.12%</td>
</tr>
<tr>
<td>Epinions</td>
<td>AUC</td>
<td>0.508</td>
<td>0.928</td>
<td>0.878</td>
<td>0.869</td>
<td>0.925</td>
<td>0.931</td>
<td>0.930</td>
<td>0.959</td>
<td>+3.01%</td>
</tr>
<tr>
<td></td>
<td>Macro-F1</td>
<td>0.474</td>
<td>0.795</td>
<td>0.746</td>
<td>0.685</td>
<td>0.800</td>
<td>0.819</td>
<td>0.819</td>
<td>0.865</td>
<td>+5.62%</td>
</tr>
<tr>
<td></td>
<td>Micro-F1</td>
<td>0.832</td>
<td>0.865</td>
<td>0.829</td>
<td>0.864</td>
<td>0.872</td>
<td>0.888</td>
<td>0.903</td>
<td>0.931</td>
<td>+3.10%</td>
</tr>
<tr>
<td></td>
<td>Binary-F1</td>
<td>0.908</td>
<td>0.915</td>
<td>0.891</td>
<td>0.922</td>
<td>0.920</td>
<td>0.931</td>
<td>0.942</td>
<td>0.961</td>
<td>+2.02%</td>
</tr>
</tbody>
</table>
Ablation Study Results

Question 3
- Does sum aggregator contribute to our proposed GS-GNN method?

Results
- The ablation study results of sum aggregator

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Metric</th>
<th>GS-GNN_mean</th>
<th>GS-GNN_sum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bitcoin-Alpha</td>
<td>AUC</td>
<td>0.844</td>
<td>0.893</td>
</tr>
<tr>
<td></td>
<td>Macro-F1</td>
<td>0.712</td>
<td>0.793</td>
</tr>
<tr>
<td></td>
<td>Micro-F1</td>
<td>0.915</td>
<td>0.930</td>
</tr>
<tr>
<td></td>
<td>Binary-F1</td>
<td>0.954</td>
<td>0.961</td>
</tr>
<tr>
<td>Bitcoin-OTC</td>
<td>AUC</td>
<td>0.900</td>
<td>0.915</td>
</tr>
<tr>
<td></td>
<td>Macro-F1</td>
<td>0.812</td>
<td>0.837</td>
</tr>
<tr>
<td></td>
<td>Micro-F1</td>
<td>0.914</td>
<td>0.920</td>
</tr>
<tr>
<td></td>
<td>Binary-F1</td>
<td>0.950</td>
<td>0.952</td>
</tr>
</tbody>
</table>

Conclusion
- Using the sum aggregator for positive and negative neighbors separately in signed graphs is important
Ablation Study Results

- **Question 3**
  - Do both the global and local representation contribute to our proposed GS-GNN method?

- **Results**
  - The ablation study results of local and global representation

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Metric</th>
<th>GS-GNN_L</th>
<th>GS-GNN_G</th>
<th>GS-GNN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bitcoin-Alpha</td>
<td>AUC</td>
<td>0.875</td>
<td>0.889</td>
<td>0.893</td>
</tr>
<tr>
<td></td>
<td>Macro-F1</td>
<td>0.754</td>
<td>0.731</td>
<td>0.793</td>
</tr>
<tr>
<td></td>
<td>Micro-F1</td>
<td>0.922</td>
<td>0.916</td>
<td>0.930</td>
</tr>
<tr>
<td></td>
<td>Binary-F1</td>
<td>0.958</td>
<td>0.954</td>
<td>0.961</td>
</tr>
<tr>
<td>Bitcoin-OTC</td>
<td>AUC</td>
<td>0.891</td>
<td>0.906</td>
<td>0.915</td>
</tr>
<tr>
<td></td>
<td>Macro-F1</td>
<td>0.801</td>
<td>0.786</td>
<td>0.837</td>
</tr>
<tr>
<td></td>
<td>Micro-F1</td>
<td>0.901</td>
<td>0.899</td>
<td>0.920</td>
</tr>
<tr>
<td></td>
<td>Binary-F1</td>
<td>0.946</td>
<td>0.942</td>
<td>0.952</td>
</tr>
</tbody>
</table>

- **Conclusion**
  - Both modules contribute to GS-GNN
  - The local and global representations of nodes are complementary
Parameter Sensitivities Results

- Question 4
  - How do essential parameters affect the model?

- Results
  - Varying the number of layers $M_L$ in the local module

- Conclusion
  - 2 local layers is a suitable choice
Parameter Sensitivities Results

- **Question 4**
  - How do essential parameters affect the model?

- **Results**
  - Varying the number of layers $M_G$ in the global module

```

```

- **Conclusion**
  - 5 global layers is a suitable choice
Parameter Sensitivities Results

- **Question 4**
  - How do essential parameters affect the model?

- **Results**
  - Varying the number of groups $K$

- **Conclusion**
  - Setting $K$ from 3 to 5 leads to the best results
Conclusion

- Study representation learning methods for signed graphs
  - Most existing methods are based on balance theory, ignore its serious limitation
  - We propose the k-group theory
    - a general and more realistic assumption beyond the usual balance theory

- Propose a novel signed GNN with a dual architecture (GS-GNN)
  - Simultaneously learn global and local representations.
    - fully leverage the k-group theory
    - with the flexibility to capture extra information beyond k-group theory
  - Simple and effective

- Extensive experimental results on synthetic and real signed graphs
  - Demonstrate the superiority of our proposed assumption and method
  - Achieves new state-of-the-art, to the best of our knowledge
Thanks!

Haixin Liu, Tsinghua University
liu-hx20@mails.tsinghua.edu.cn