Bi-Level Selection via Meta Gradient for Graph-based Fraud Detection

Linfeng Dong¹, Yang Liu¹; Xiang Ao¹*; Jianfeng Chi²; Jinghua Feng²; Hao Yang²; Qing He¹

董临风¹; 柳阳¹; 敖翔¹*; 池剑锋²; 冯景华²; 杨浩²; 何清¹

1 Institute of Computing Technology, Chinese Academy of Sciences
2 Alibaba Group
Table of Contents

- **Background and Motivation**
  - Graph-based Fraud Detection / General Workflow of GNN / Key Challenges

- **Method-BLS**
  - Framework / Instance-level Selection / Neighborhood-level Selection

- **Experiments**
  - Overall Evaluation / Comparison with Imbalanced Learning methods / Ablation Study

- **Conclusion**
Introduction
Graph-based Fraud Detection
Introduction

General Workflow of GNN

Graph

Representation Learning Phase

Classification Phase

AGG

MLP Classifier

CE Loss
Key Challenges

- **Neighborhood-level imbalance and noise**
  - In the representation learning phase, due to the propagation mechanism on topology, excess benign neighbors will dominate the network structure and dilute the feature of fraudsters, resulting in inaccurate embeddings of fraudulent nodes.

- **Instance-level imbalance and noise**
  - In the classification phase, the majority class will dominate the training loss during the gradient descent step, leading to a biased decision boundary. Undiscovered fraudsters (noise) will contribute wrong gradient direction, thus polluting the learned classification boundary.
Method-BLS
Framework
Learn a weight $\theta_i$ for each node $v_i$ in the training set that minimizing the CE loss on an unbiased meta validation set.

$$\theta_i \propto -\beta \cdot \frac{1}{M} \sum_{i=1}^{M} \left( \frac{\partial l_j(\hat{W}_f)}{\partial \hat{W}_f} \bigg|_{\hat{W}_r = \hat{W}_r^{(t)}} \right)^\top \left( \frac{\partial l_i(W_f)}{\partial W_f} \bigg|_{W_f = W_f^{(t)}} \right)$$

$$\mathcal{L}_{GNN} = -\sum_{i=1}^{N} \theta_i \cdot (y_i \log p_i + (1 - y_i) \log(1 - p_i))$$
Neighborhood-level Node Selection

- Select neighborhood-level valuable nodes to form subset $\mathcal{N}_{i,r} \subseteq \mathcal{N}_{i,r}$
  1. Generate pseudo label
  2. Compute pseudo label affinity scores between center node and its neighbors
  3. Select top-k neighbors to participate in neighborhood aggregation

$$\hat{p}_i^{(\ell)} = G^{(\ell)}(h_i^{(\ell-1)}, W_g^{(\ell)})$$
$$S_{ij}^{(\ell)} = 1 - \|\hat{p}_i^{(\ell)} - \hat{p}_j^{(\ell)}\|_1$$
$$\mathcal{L} = \mathcal{L}_{GNN} + \sum_{\ell=1}^{L-1} \mathcal{L}_{PSE}^{(\ell)}$$
**Datasets**

- **YelpChi**
  - hotels and restaurants reviews on Yelp.com, nodes are reviews associated with 100-dimension Word2Vec embeddings as features

- **Amazon**
  - product reviews under the Musical Instrument category, nodes are users associated with 100-dimension Word2Vec embeddings as features

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#Node</th>
<th>#Edge</th>
<th>IR</th>
<th>Relation</th>
<th>#Edge-R</th>
</tr>
</thead>
<tbody>
<tr>
<td>YelpChi</td>
<td>45,954</td>
<td>3,846,979</td>
<td>5.9</td>
<td>R-U-R</td>
<td>49,315</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>R-T-R</td>
<td>573,616</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>R-S-R</td>
<td>3,402,743</td>
</tr>
<tr>
<td>Amazon</td>
<td>11,944</td>
<td>4,398,392</td>
<td>13.5</td>
<td>U-P-U</td>
<td>175,608</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>U-S-U</td>
<td>3,566,479</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>U-V-U</td>
<td>1,036,737</td>
</tr>
</tbody>
</table>
Experiments

Baselines and Evaluation Metrics

- Base models (extended for multi-relational graph + BLS)
  - GCN / GAT / GraphSAGE

- Comparison SOTA

- Evaluation Metrics
  - AUC / G-Mean


## Experiments

### Main Result

<table>
<thead>
<tr>
<th>Dataset</th>
<th>YelpChi</th>
<th>Amazon</th>
</tr>
</thead>
<tbody>
<tr>
<td>Methods</td>
<td>AUC</td>
<td>G-Mean</td>
</tr>
<tr>
<td>GCN</td>
<td>59.02±1.08</td>
<td>55.61±2.96</td>
</tr>
<tr>
<td>GraphSAGE</td>
<td>58.46±3.03</td>
<td>46.90±5.82</td>
</tr>
<tr>
<td>GAT</td>
<td>64.18±1.84</td>
<td>59.53±4.37</td>
</tr>
<tr>
<td>GraphConsis</td>
<td>69.83±3.42</td>
<td>58.57±3.85</td>
</tr>
<tr>
<td>CARE-GNN</td>
<td>78.44±0.69</td>
<td>70.13±2.17</td>
</tr>
<tr>
<td>PC-GNN</td>
<td>79.87±0.14</td>
<td>71.60±1.30</td>
</tr>
</tbody>
</table>

\[
\begin{array}{lllll}
\text{Dataset} & \text{YelpChi} & \text{Amazon} \\
\text{Methods} & \text{AUC} & \text{G-Mean} & \text{AUC} & \text{G-Mean} \\
\hline
\text{GCN} & 74.62±1.38 & 68.72±1.92 & 92.93±2.04 & 83.22±3.85 \\
\text{GraphSAGE}_M & 77.12±2.56 & 69.15±3.96 & 93.63±1.37 & 85.92±4.20 \\
\text{GAT}_M & 81.73±1.48 & 75.33±3.52 & 93.71±1.06 & 85.82±3.65 \\
\text{BLS+GCN}_M & 83.28±0.86 & 76.31±2.74 & 94.42±1.55 & 87.41±0.78 \\
\text{BLS+GraphSAGE}_M & 86.50±0.78 & 80.02±2.93 & 94.71±1.33 & 87.44±2.02 \\
\text{BLS+GAT}_M & 89.26±1.04 & 81.82±3.02 & 95.93±0.73 & 90.72±1.64 \\
\end{array}
\]
Conclusion

➢ We propose BLS, a meta gradient based algorithm to address the imbalanced and noisy label problem in graph-based fraud detection. In both representation learning and classification phase, BLS adopts a unified meta-learning paradigm to select instance-level and neighborhood-level valuable nodes.

➢ Compared to existing methods, BLS is the first work that considers the impact of class imbalance and noisy label on the message-passing mechanism of GNNs.

➢ The proposed BLS algorithm has high portability that can be applied on any GNN framework. By applying BLS on widely-used GNNs, we achieved significant improvement compared to base models and state-of-the-art on two real-world datasets.
Thanks for listening!
If you have any question, feel free to contact us at
donglinfeng19s@ict.ac.cn
aoxiang@ict.ac.cn