Self-Explainable Graph Transformer for Link Sign Prediction

Lu Li, Jiale Liu, Xingyu Ji, Maojun Wang*, Zeyu Zhang*

National Key Laboratory of Crop Genetic Improvement, Hubei Hongshan Laboratory, Huazhong Agricultural University
mjwang@mail.hzau.edu.cn, zhangzeyu@mail.hzau.edu.cn

Background

1. **Signed graph.** Signed graph can model friendly or antagonistic relations where edges are annotated with a positive or negative sign. Signed Graph Neural Networks (SGNNs) are widely used for learning representations of signed graphs, as shown in Figure 1.
2. **Lack of explainability.** However, SGNN models suffer from poor explainability, which limit their adoptions in critical scenarios that require under standing the rationale behind predictions. To the best of our knowledge, there is currently no research work on the explainability of the SGNN models.
3. **GNN explainability.** Research on GNN explainability falls into two main categories: post-hoc explanations and self-explainable approaches. Post-hoc explanation methods offer interpretations for trained GNN models, but these explanations may be biased and not truly reflective of the model. Therefore, the current mainstream has shifted toward self-explanatory methods, where the model can provide corresponding explanations while delivering the prediction results.

Introduction

While effective, these GNN explainability methods are primarily designed for unsigned graphs with graph or node classification tasks, making them unsuitable for SGNNs focusing on link sign prediction.

One potential approach for obtaining self-explanations in link sign prediction is to identify explainable K-nearest positive neighbors (and K-farthest negative neighbors) for each node. If the closeness between the two nodes of an edge e_{ij} is similar to the closeness between node v_i and its K-nearest positive (or K-farthest negative) neighbors, the predicted sign of the edge is positive (or negative).

The challenges of identifying the K-nearest (farthest) positive (negative) neighbors have two aspects:

Challenge 1: How to improve the quality of node representations. To prevent overfitting, current SGNN models limit networks to three layers or fewer, restricting their ability to capture multi-hop information. Meanwhile, GCN-based SGNN frameworks cannot learn proper representations from unbalanced cycles.
Challenge 2: How to find a sufficient number of negative neighbors for nodes. As is shown in Figure 2, a significant proportion of nodes have few or even no negative neighbors.

For Challenge 1, we propose a novel graph Transformer for signed graphs, leveraging signed random walk encoding to capture multi-hop neighbor information. We theoretically demonstrate that this encoding method has a stronger representation power compared to current SGNNs and other common encoding methods, such as the shortest path encoding.
For Challenge 2, we employ signed diffusion matrix based on Signed Random Walk with Restart (SRWR) algorithm to uncover potential negative relationships among nodes.

Overall, we propose a novel Self-Explainable Signed Graph Transformer (SE-SGformer) framework which predicts edge signs by identifying the K-nearest (farthest) positive (negative) neighbors and provides corresponding explanatory information.

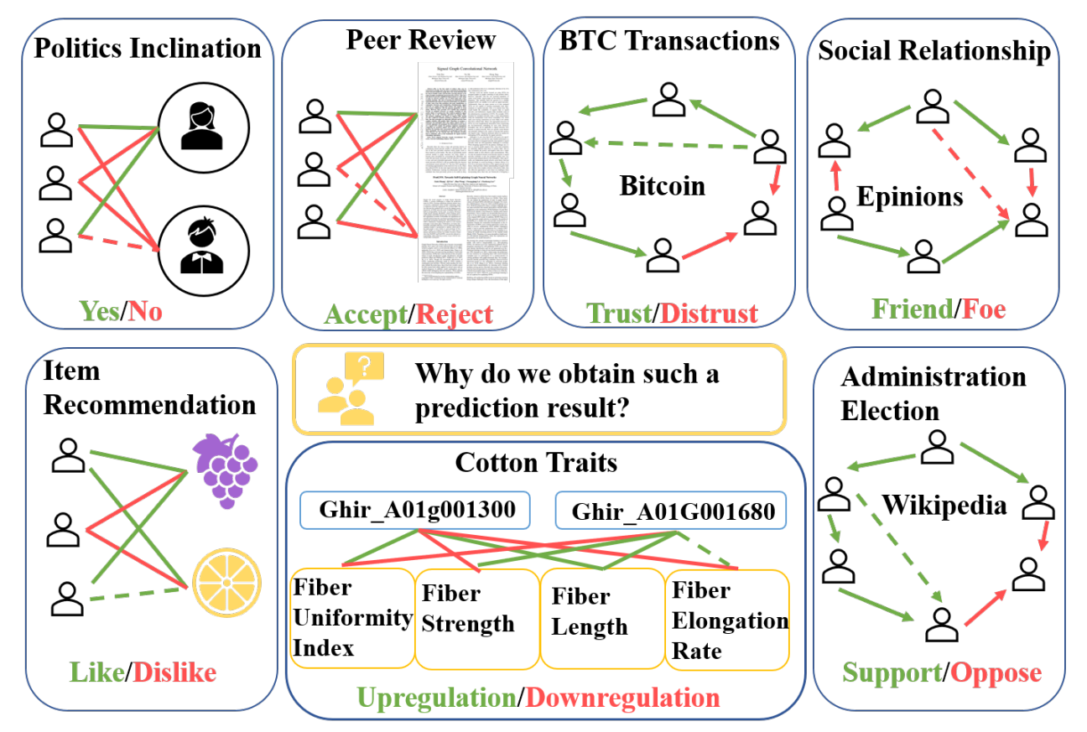
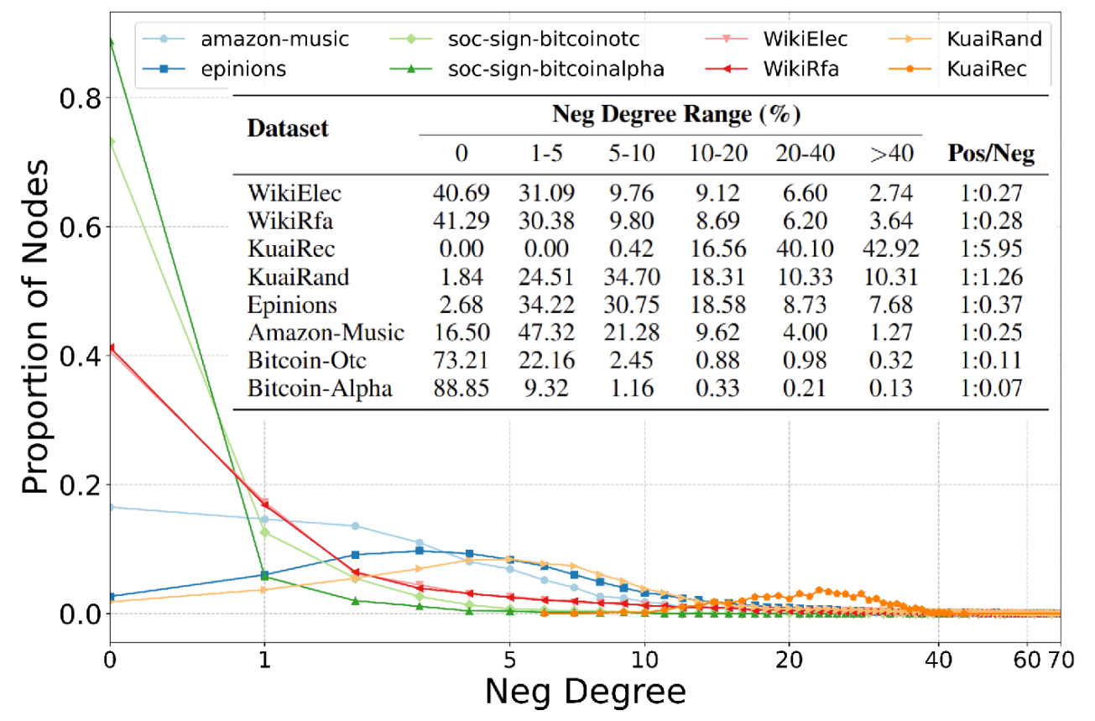


Figure 1. An illustration of signed graphs in real world.



Dataset	Neg Degree Range (%)						Pos/Neg
	0	1-5	5-10	10-20	20-40	>40	
WikiElec	40.69	31.09	9.76	9.12	6.60	2.74	1:0.27
WikiRfa	41.29	30.38	9.80	8.69	6.20	3.64	1:0.28
KuaiRec	0.00	0.00	0.42	16.56	40.10	42.92	1:5.95
KuaiRand	1.84	24.51	34.70	18.31	10.33	10.31	1:1.26
Epinions	2.68	34.22	30.75	18.58	8.73	7.68	1:0.37
Amazon-Music	16.50	47.32	21.28	9.62	4.00	1.27	1:0.25
Bitcoin-Otc	73.21	22.16	2.45	0.88	0.98	0.32	1:0.11
Bitcoin-Alpha	88.85	9.32	1.16	0.33	0.21	0.13	1:0.07

Figure 2. Proportion of node negative degrees and Pos/Neg Ratios across datasets (%)

Main results

1. **Overall Framework.** The overall framework of SE-SGformer is shown in Figure 3. The basic idea of SE-SGformer is to first use a Transformer to encode the signed graph. Then, for an edge with an unknown sign from node v_i to node v_j, we identify the K-nearest (farthest) positive (negative) neighbors of v_i. The sign of the edge e_{ij} is determined based on the similarity between node v_j and the positive or negative neighbors of node v_i, with explanatory information provided simultaneously. It is mainly divided into two parts: encoding module and explainable prediction module.

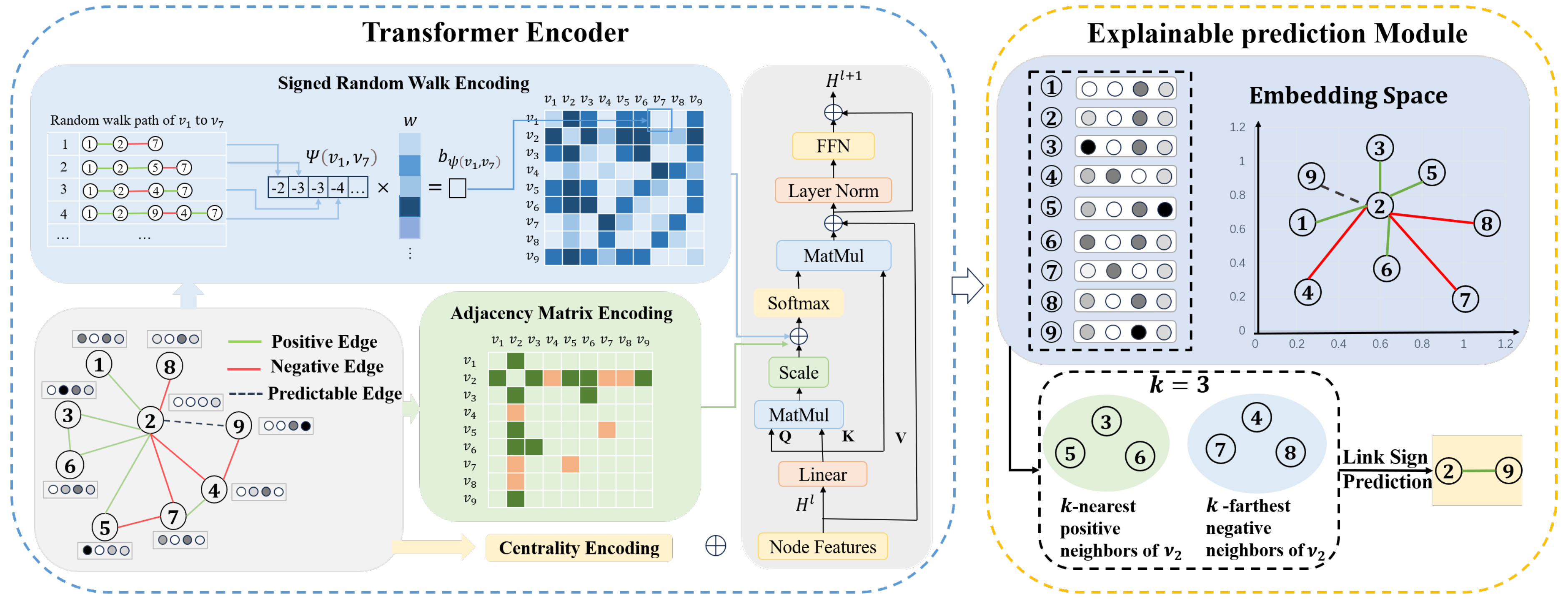

Transformer Encoder
Signed Random Walk Encoding
Random walk path of v_1 to v_7
$\Psi(v_1, v_7)$
-2 -3 -3 -4 …
×
w
=
$b_{\psi(v_1,v_7)}$
v_1 v_2 v_3 v_4 v_5 v_6 v_7 v_8 v_9
Positive Edge
Negative Edge
Predictable Edge
Adjacency Matrix Encoding
Centrality Encoding
H^{l+1}
FFN
Layer Norm
MatMul
Softmax
Scale
MatMul
Q
K
V
Linear
H^l
Node Features
Explainable prediction Module
Embedding Space
$k = 3$
k-nearest positive neighbors of v_2
k -farthest negative neighbors of v_2
Link Sign Prediction

Figure 3. The overall architecture of SE-SGformer. Firstly, the Transformer encoder, equipped with centrality encoding, adjacency matrix encoding, and random walk encoding, obtain node embeddings. Next, in the explainable prediction module, the K-nearest positive (and K-farthest negative) neighbors of the nodes are identified and used to predict the unknown link sign.

2. **Analysis of expression ability.** We theoretically analyzed the expressive power of our signed random walk encoding compared to shortest path encoding. We first define the concept of signed graph isomorphism:
Signed graph isomorphism. Two signed graphs G_1 and G_2 are isomorphic, if there exists a bijection $\phi : \mathcal{V}_{\mathcal{G}_1} \to \mathcal{V}_{\mathcal{G}_2}$, for every pair of nodes $v_1, v_2 \in V_{\mathcal{G}_1}$, $e_{ij} \in \mathcal{E}_1$, if and only if $e_{\phi(v_i),\phi(v_j)} \in \mathcal{E}_2$ and $\sigma(e_{ij}) = \sigma(e_{\phi(v_i),\phi(v_j)})$.
Then we proof the theorem 1 and Figure 4 illustrates an example where the shortest path encoding fails to distinguish two graphs as non-isomorphism, but the signed random walk encoding can successfully identify.
Theorem 1 With a sufficient number of random walks, signed random walk encoding is more expressive than that based on a fixed shortest path for signed graph.
For the issues present in Challenge 1, we further analyzed the expressive power of our graph transformer architecture based on signed random walk encoding and SGCN-based SGNN models:
Theorem 2 With a sufficient number of random walk length, the graph transformer architecture based on signed random walk encoding is more expressive than SGCN.
Signed random walk encoding combines information from multiple weighted paths to judge the relationship between nodes. This allows us to help the model obtain reasonable association information between two points within an unbalanced cycle.

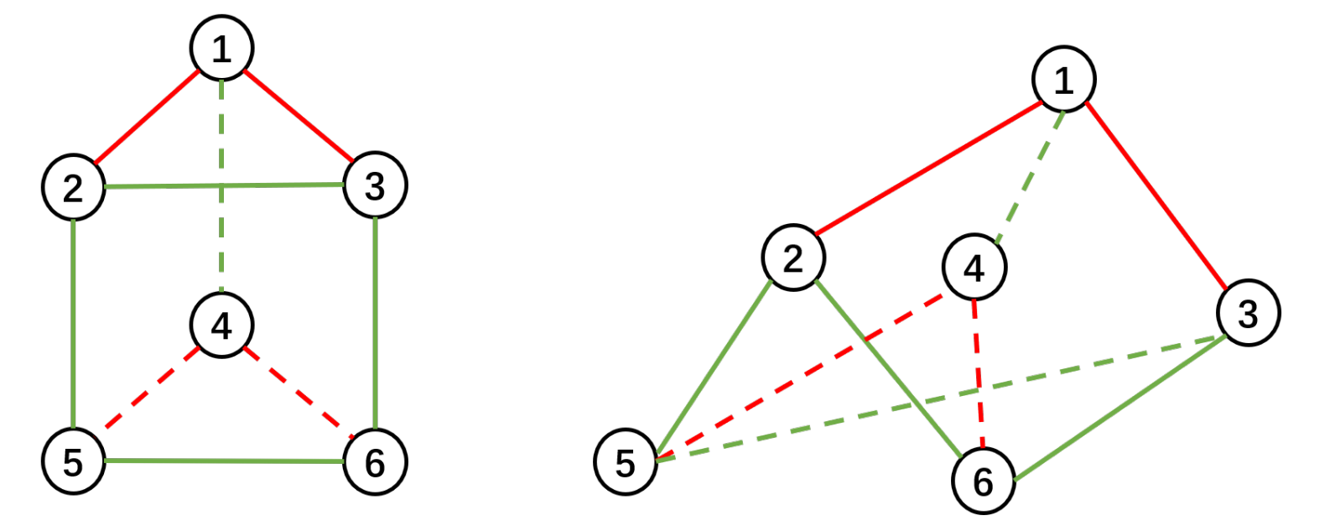


Figure 4. Encoding based on the shortest path cannot map these two graphs to different embeddings, while encoding based on the signed random walk can map them to different embeddings.

Experiments

We compare the performance of SE-SGformer with baselines on real-world datasets in terms of link sign prediction. All datasets were experimented with five times, and the link sign prediction accuracy and standard deviation are shown in Table 1. Then, we evaluate the quality of the explanatory information of the k- nearest positive (farthest negative) neighbors identified in the explainable decision-making process. We set K = 40 and compared our method with the baseline on three datasets. The precise@40 and standard deviation are shown in Table 2.

Dataset	GCN	GAT	SGCN	SNEA	SGCL	SIGformer	SE-SGformer
Amazon-music	63.87 ± 3.57	65.39 ± 2.88	70.63 ± 0.69	70.48 ± 0.05	78.26 ± 1.52	58.64 ± 0.64	**79.20 ± 0.23**†
Epinions	71.07 ± 1.26	73.97 ± 1.64	**86.97 ± 1.53**	82.26 ± 0.57	70.83 ± 5.10	57.07 ± 0.38	72.84 ± 1.78
KuaiRand	44.35 ± 0.00	51.63 ± 2.84	**62.85 ± 0.05**	61.95 ± 0.13	60.68 ± 0.99	61.40 ± 0.47	56.89 ± 0.12
KuaiRec	61.56 ± 0.42	65.73 ± 0.74	85.11 ± 0.11	79.69 ± 0.01	79.84 ± 3.13	61.31 ± 1.37	**85.60 ± 0.05**†
WikiRfa	70.79 ± 6.02	71.31 ± 3.56	78.69 ± 1.06	75.20 ± 0.13	75.02 ± 4.33	65.60 ± 0.94	**79.99 ± 0.08**†
WikiElec	66.21 ± 1.50	66.50 ± 1.76	79.14 ± 0.48	77.10 ± 0.68	79.63 ± 2.82	65.74 ± 2.72	**80.63 ± 0.08**†
Bitcoin-OTC	83.77 ± 0.60	86.37 ± 1.24	88.22 ± 0.69	86.05 ± 0.46	87.65 ± 1.74	80.30 ± 2.32	**90.03 ± 0.35**‡
Bitcoin-Alpha	83.99 ± 1.61	86.25 ± 0.38	87.96 ± 0.38	87.95 ± 0.15	83.01 ± 3.79	73.82 ± 3.55	**89.88 ± 0.40**‡

Table 1. Comparison of Accuracy(%) across Different Models. The best scores are in bold, and the second-best ones are underlined. ”†” and ”‡” indicate the statistically significant improvements with $p < 0.05$ and $p < 0.01$ (one-sided paired t-test) over the best baseline, respectively.

Next, we conduct a detailed sensitivity analysis of the key hyper-parameters max degree (D), dim (d), and layer (L). D represents the maximum value of positive or negative degrees, d refers to the dimension of the node embedding, and L indicates the number of Transformer layers. The results are shown in Figure 5.

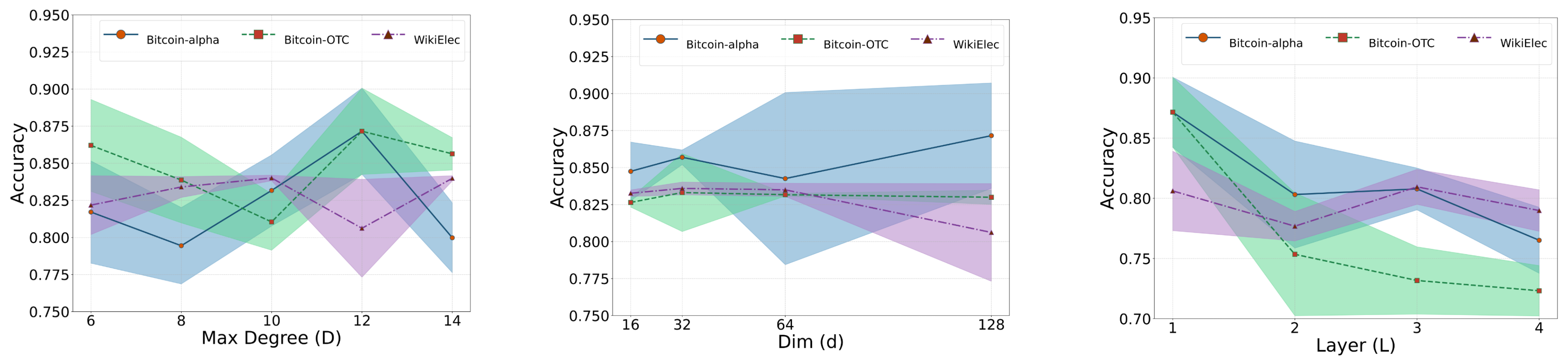


Figure 5. Parameter sensitivity analysis.

Also, we conducte ablation experiments to explore the impact of the three encodings of Transformer. SE-SGformer$_{\text{w/o-CE}}$ denotes the variant without centrality encoding. SE-SGformer$_{\text{w/o-RE}}$ denotes the variant without signed random walk encoding. SE-SGformer$_{\text{w/o-AE}}$ denotes the variant without adjacency matrix encoding. The experiment results on Bitcoin-OTC and Bitcoin-Alpha datasets are reported in Figure 6.

Model	Bitcoin-OTC	Bitcoin-Alpha	Amazon-music
SGCN	57.25 ± 0.15	54.88 ± 0.14	60.78 ± 0.07
SNEA	55.09 ± 0.16	55.43 ± 0.15	60.29 ± 0.07
SGCL	54.10 ± 0.16	54.59 ± 0.15	61.70 ± 0.08
SIGformer	53.33 ± 0.14	51.27 ± 0.09	58.76 ± 0.05
SE-SGformer	**75.19 ± 0.13**	**94.47 ± 0.58**	**76.07 ± 0.20**

Table 2. The metric precision@40 (%) of baselines on different datasets.

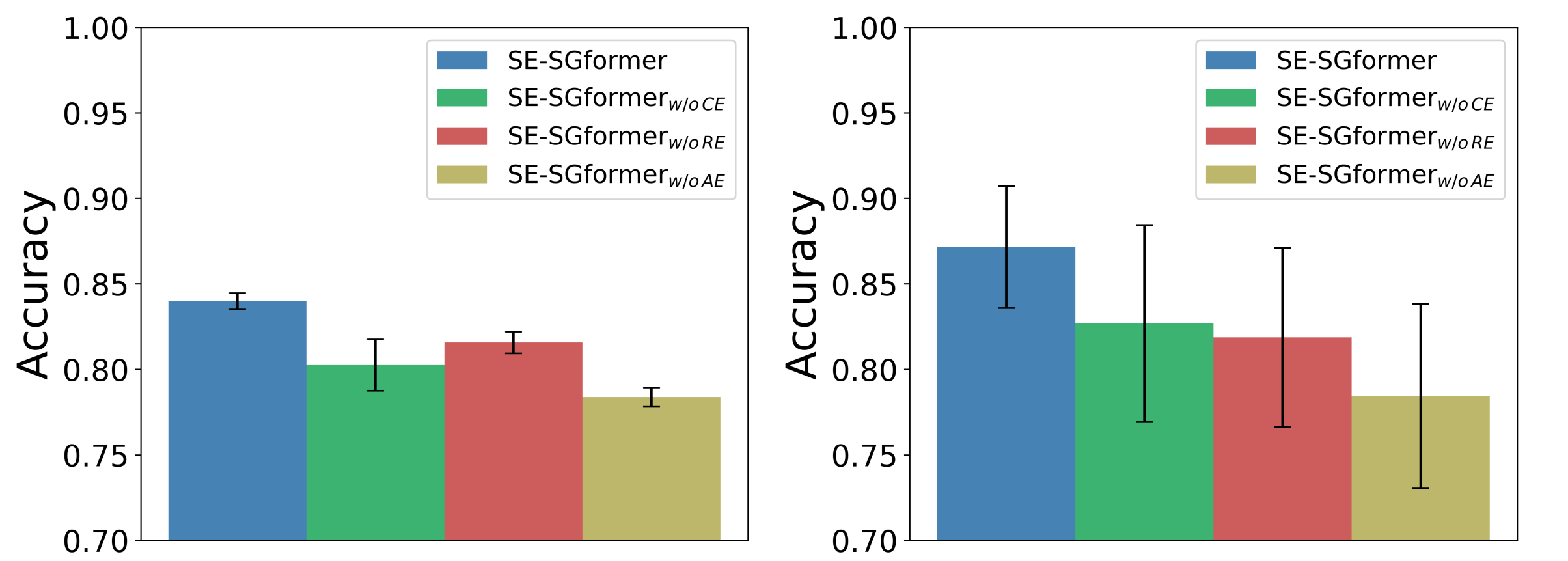


Figure 6. Ablation study on Bitcoin-OTC (left) and Bitcoin-Alpha (right) datasets.

AAAI 2025.

lu123@webmail.hzau.edu.cn