Explainable Graph-based Fraud Detection via Neural Meta-graph Search

Zidi Qin[†]
Institute of Computing Technology, Chinese Academy of Sciences
Beijing, China
qinzidi20s@ict.ac.cn

Yang Liu[†]
Institute of Computing Technology, Chinese Academy of Sciences
Beijing, China
liuyang17z@ict.ac.cn

Qing He[†]
Institute of Computing Technology, Chinese Academy of Sciences
Beijing, China
heqing@ict.ac.cn

Xiang Ao[*†]
Institute of Computing Technology, Chinese Academy of Sciences
Beijing, China
aoxiang@ict.ac.cn

ABSTRACT

Though graph neural networks (GNNs)-based fraud detectors have received remarkable success in identifying fraudulent activities, few of them pay equal attention to models' performance and explainability. In this paper, we attempt to achieve high performance for graph-based fraud detection while considering model explainability. We propose NGS (**N**eural meta-**G**raph **S**earch), in which the message passing process of a GNN is formalized as a meta-graph, and a differentiable neural architecture search is devised to determine the optimized message passing graph structure. We further enhance the model by aggregating multiple searched meta-graphs to make the final prediction. Experimental results on two real-world datasets demonstrate that NGS outperforms state-of-the-art baselines. In addition, the searched meta-graphs concisely describe the information used for prediction and produce reasonable explanations.

CCS CONCEPTS

• **Mathematics of computing** → **Graph algorithms**; • **Computing methodologies** → **Neural networks**.

KEYWORDS

neural architecture search, graph neural network, fraud detection

ACM Reference Format:
Zidi Qin, Yang Liu, Qing He, and Xiang Ao. 2022. Explainable Graph-based Fraud Detection via Neural Meta-graph Search. In *Proceedings of the 31st ACM International Conference on Information and Knowledge Management (CIKM '22), October 17–21, 2022, Atlanta, GA, USA.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3511808.3557598

[*]Corresponding author.
[†]Key Lab of Intelligent Information Processing of Chinese Academy of Sciences (CAS). Also at University of Chinese Academy of Sciences. Xiang Ao is also at Institute of Intelligent Computing Technology, Suzhou, China.

CIKM '22, October 17–21, 2022, Atlanta, GA, USA

ACM ISBN 978-1-4503-9236-5/22/10.
https://doi.org/10.1145/3511808.3557598

1 INTRODUCTION

Fraud is an intentional deception designed to obtain financial or personal gain, which causes severe damage to our life [3, 6, 8, 18, 30]. Therefore, numerous approaches have been developed in past years for fraud detection [20, 26, 37], among which graph-based ones have escalated much attention. The possible reason might be the graph data presents rich behavioral interactions among users, offering multifaceted information for fraud detection.

Despite the success in fraud detection, most existing graph-based methods cannot explain "what drives the model to make certain predictions?", which will limit their application in critical areas such as finance. Although various self-explanatory GNNs have been proposed recently, they will face performance and efficiency limitations when deployed on real-world tasks like fraud detection. For example, SE-GNN [5] and ProtGNN [34] rely on the assumptions of K-Nearest Neighbor and prototype learning, respectively, which may not necessarily apply to real scenarios. Therefore, developing a fraud detector that simultaneously gives high-quality predictions and explanations is imperative to the deployed anti-fraud systems.

One perspective to obtain explanations for fraud detection is applying attention mechanisms that generate soft masks on the input graph [12, 29]. The masked sub-graphs are regarded as rationales to guide the model predictions. However, there is still debate over whether attention can provide explanations [1], and recent research suggests that attention-as-importance interpretations often do not work as expected [2, 15]. Another way is meta-path sampling, which employs human-defined meta-paths to aggregate information from the meta-path-aware neighborhood [11, 36]. Meta-paths illustrate what semantic information the model captures, thus offering explainability. However, it is challenging to manually design meta-paths, as it usually requires prior knowledge about task-related patterns. Moreover, the meta-path is generally arranged in a sequence that limits its capacity to capture intricate semantic proximity. For example, the skip, merge or split connection is not allowed in meta-path design. In a nutshell, how to enable model explainability while keeping high performance and efficiency for graph-based fraud detection has not been well addressed.

In this paper, inspired by DiffMG [7], a recent differentiable meta-graph search for heterogeneous information networks (HINs), we

propose **N**eural meta-**G**raph **S**earch (NGS for short) for explainable graph neural network-based fraud detection. The framework consists of three steps: (1) formalizing the message passing process of GNN using meta-graph; (2) searching the meta-graph using differentiable neural architecture search (DARTS) [19]; (3) aggregating node embeddings captured by multiple meta-graphs. After searching, the model is retrained with the derived meta-graphs for final evaluation. Compared with previous works, the meta-graphs searched by NGS define the internal message passing process of the GNN, providing high-quality intrinsic explanations for the model prediction. Meanwhile, NGS employs DARTS to automatically search the meta-graphs without any prior knowledge. Moreover, meta-graphs can capture complex semantic relations and thus help the model achieve higher performance [14, 35].

We conduct extensive experiments on two real-world opinion fraud detection datasets. Experimental results show that our method exceeds state-of-the-art baselines with satisfying performance. The searched meta-graphs reveal the factors that drive the model to make specific predictions and thus allow our NGS to be capable of explainability. The demonstration is consistent with prior human experience and helps deepen our understanding of the datasets.

2 RELATED WORKS

Graph-based Fraud Detection. Graph-based methods have shown their superiority in fraud detection. For example, SemiGNN [29] applies a GNN-based hierarchical attention mechanism to detect fraudsters on Alipay. GraphConsis [22] and CARE-GNN [9] filter dissimilar neighbors before aggregation to find out camouflage fraudsters. PC-GNN [21] and AO-GNN [13] solve the label imbalance issue by node resampling and edge pruning, respectively. FRAUDRE [33] unifies four modules into a GNN to tackle the graph inconsistency and imbalance issues. H^2-FDetector [27] considers the homophilic and heterophilic connections simultaneously. However, few of them balance explainability.

Explainable GNNs. Recently, the inherent explainability of GNNs has received much attention. Self-explainable GNNs like SE-GNN [5] and ProtGNN [34] are proposed to give predictions and explanations simultaneously. However, their assumptions do not necessarily hold in real-world scenarios. For graph-based fraud detection, AMG-DP [12] and SemiGNN [29] utilize attention mechanisms to tell the essential factors for the fraud. HACUD [11] and MAHINDER [36] explain the predictions by predefined meta-paths. Nevertheless, explanations provided by attention mechanisms are not reliable, and the meta-path definition is inefficient.

3 PRELIMINARIES

Definition 3.1 (Multi-relation Graph)**.** Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A}, \mathcal{X}, \mathcal{Y})$, $\mathcal{V}$ denotes the set of nodes and $\mathcal{E} = \{E_r\}|_{r=1}^{R}$ denotes the edge set of R relations. If $R > 1$, we define $\mathcal{G}$ as a multi-relation graph. $\mathcal{A} = \{A_r\}|_{r=1}^{R}$, where A_r denotes the adjacency matrix formed by the edges of type r. $x_i \in \mathcal{X}$ represents a feature vector of node v_i and $x_i \in \mathbb{R}^d$. $\mathcal{Y}$ is the set of labels for each node in $\mathcal{V}$.

Definition 3.2 (Meta-graph)**.** A meta-graph M is a directed acyclic graph (DAG), with a single source node and a single target node. The edge connecting to nodes denotes the relation r.

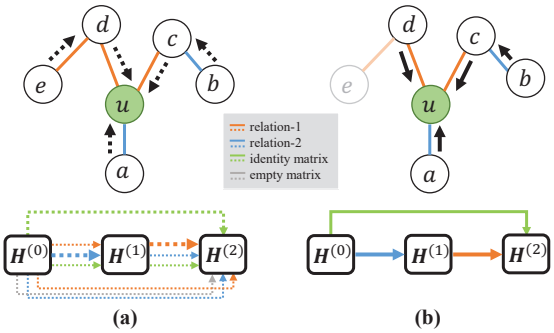


Figure 1: An overview of the search step of NGS. (a) Continuous relaxation of the search space by mixing all candidates on each edge, indicating aggregating messages coming along all possible relations. (b) The final meta-graph induced from the learned mixing probabilities and the message passing process guides on the example graph.

4 METHODOLOGY

This section presents the details of our proposed method NGS, which has three essential components: meta-graph-defined message passing, DARTS employment, and multi-meta-graph aggregation.

4.1 Meta-graph-defined Message Passing

Generally, a GNN learns node representations by utilizing the message passing scheme to aggregate information from nodes' neighbors. This process can be mathematically written as:

$$\boldsymbol{H}^{(0)} = \mathrm{MLP}\left(\boldsymbol{X}\right), \quad \boldsymbol{H}^{(l+1)} = \mathrm{Aggr}\left(\boldsymbol{H}^{(l)}; A\right) \tag{1}$$

where $\boldsymbol{X}$ denotes the node attributes, $\mathrm{Aggr}(\boldsymbol{H}^{(l)}; A)$ denotes aggregate information $\boldsymbol{H}^{(l)}$ from neighbors defined by A. Specifically, we choose the graph convolutional network (GCN) [17] as the basic GNN. $\boldsymbol{H}^{(l)}$ is the node representations in the l-th layer. However, for multi-relation graphs, Eq. (1) indiscriminately aggregates messages coming along all relations and thus loses semantic information.

Therefore, we use expressive meta-graph $M_{\mathcal{A}}$ to describe the message passing process between GNN layers by selecting edge types when aggregating neighbors' information. The meta-graph's source node and target node denote initial and final representations: $\boldsymbol{H}^{(0)}$ and $\boldsymbol{H}^{(L)}$, and the ordered nodes denote intermediate representations in the computation procedure. L is the predefined number of intermediate states. In this DAG, each intermediate node $\boldsymbol{H}^{(l)}$ $(1 \le l \le L)$ is computed based on all of its predecessors: $\boldsymbol{H}^{(l)} = \sum_{0 \le i < l} f_{i,l}\left(\boldsymbol{H}^{(i)}; \mathcal{A}_{i,l}\right)$. $f_{i,l}$ denotes passing $\boldsymbol{H}^{(i)}$ message along a particular relation type given by the corresponding edge of the meta-graph, which is selected from $\mathcal{A}_{i,l}$. $\mathcal{A}_{i,l}$ is defined as:

$$\mathcal{A}_{i,l} = \begin{cases} \mathcal{A} \cup \{\boldsymbol{I}\} & l \le L \text{ and } i = l-1 \\ \mathcal{A} \cup \{\boldsymbol{I}\} \cup \{\boldsymbol{O}\} & l \le L \text{ and } i < l-1 \end{cases} \tag{2}$$

where $\mathcal{A}$ is the adjacency matrix collection of the multi-relation graph $\mathcal{G}$. The identity matrix $\boldsymbol{I}$ and the empty matrix $\boldsymbol{O}$ allow the number of actual message passing steps and incoming links for $\boldsymbol{H}^{(l)}$ in the searched meta-graph to be flexible [7].

4.2 DARTS Employment

In order to efficiently search the meta-graph, we relax the discrete edge type selection to be continuous like DARTS [19]:

$$f_{i,l}\left(\boldsymbol{H}^{(i)};\mathcal{A}_{i,l}\right)=\sum_{A\in\mathcal{A}_{i,l}}\frac{\exp\left(\alpha_{i,l}^{A}\right)}{\sum_{A'\in\mathcal{A}_{i,l}}\exp\left(\alpha_{i,l}^{A'}\right)}\cdot\mathrm{Aggr}\left(\boldsymbol{H}^{(i)};A\right) \quad (3)$$

In Eq. (3), messages passing along all possible relations are mixed by introducing the learnable architecture parameters α, which is illustrated in Figure 1(a).

Following the meta-graph guided message passing process, an MLP classifier is trained to minimize the cross-entropy loss:

$$\mathcal{L}=-\sum_{v\in\mathcal{V}}\left[y_v\log p_v+(1-y_v)\log\left(1-p_v\right)\right] \quad (4)$$

where $p_v=\mathrm{MLP}\left(h_v\right)$, and label $y_v=1$ denotes fraud while $y_v=0$ denotes benign. h_v is the final latent vector of node v obtained from $\boldsymbol{H}^{(L)}$. The model parameters (weight matrices and biases) ω and architecture parameters α are optimized in a bi-level schema:

$$\min_{\alpha}\mathcal{L}_{\mathrm{val}}\left(\omega^{*}\left(\alpha\right),\alpha\right),\ \text{s.t. }\omega^{*}\left(\alpha\right)=\arg\min_{\omega}\mathcal{L}_{\mathrm{train}}\left(\omega,\alpha\right) \quad (5)$$

We apply the first-order approximation here to solve the problem: in each iteration we first fix α and update ω by calculating $\partial\mathcal{L}_{\mathrm{train}}/\partial\omega$, then we fix ω and update α through $\partial\mathcal{L}_{\mathrm{val}}/\partial\alpha$.

After convergence, the optimal meta-graph can be achieved by choosing the edge type with the maximum α, as illusrated in Figure 1(b). Then at the evaluation phase, the model is constructed and retained following the designed meta-graph.

4.3 Multi-meta-graph Aggregation

To boost the final performance of NGS, we search for multiple meta-graphs and combine the semantic information revealed by them. For each, we first apply a meta-graph specific transformation for projected node attributes: $\boldsymbol{H}_{M}^{(0)}=\sigma\left(\mathrm{MLP}\left(\boldsymbol{H}^{(0)}\right)\right)$, where σ is an activation function such as ReLU. It encodes node attributes that may be relevant to fraudulent behavior [33]. After meta-graph guided message passing, for node $v\in\mathcal{V}$, we have $|\mathcal{V}|$ sets of latent vectors: $\{h_v^{M_1},h_v^{M_2},\ldots,h_v^{M_K}\}$, where K is the predefined number of meta-graphs. We assign different weights to these meta-graphs and aggregate them as follows.

$$\begin{aligned}e_{M_k}&=\mathrm{MLP}\left(h_v^{M_k}\right),\quad h_v=\textstyle\sum_{1\le k\le K}\beta_{M_k}\cdot h_v^{M_k}\\ \beta_{M_k}&=\frac{\exp\left(e_{M_k}\right)}{\sum_{1\le k'\le K}\exp\left(e_{M_{k'}}\right)}\end{aligned} \quad (6)$$

where h_v is the node vector used for final prediction.

5 EXPERIMENTS

In this section, we conduct a comparative evaluation of NGS against various baselines on two graph-based fraud detection datasets, exceeding or matching performance across all of them.

5.1 Experimental Setup

Datasets. We adopt two real-world fraud detection datasets to validate NGS's performance: Amazon [23] and YelpChi [25]. The nodes in the Amazon graph are users with 25-dimension features, and edges are designed by three relations: U-P-U, U-S-U, and U-V-U. The nodes in the Yelpchi are reviews with 32-dimension features, and edges are designed by three relations: R-U-R, R-T-R, and R-S-R. The descriptions and statistics of datasets can be found in [27].

Baselines. We compare NGS with various GNN baselines. We select GCN [17], GAT [28], and GraphSAGE [10] as general GNN models. We choose CARE-GNN [9], PC-GNN [21], FRAUDRE [33], AO-GNN [13], and H^2-FDetector [27] as state-of-the-art GNN-based fraud detection methods. We select ProtGNN [34] as a representative self-explainable GNN, while we do not compare our model with SE-GNN [5] as it is out of GPU memory on our machine. DiffMG [7] is a differentiable meta-graph search algorithm in HINs. $\mathrm{NGS}_{\setminus A}$ is a variant of NGS removing multi-meta-graph aggregation.

Experimental Settings and Implementation. We employ the Adam [16] optimizer for NGS. We run the model for 100 epochs at the search stage to derive meta-graphs. The learning rate of α and ω are $3e^{-4}$ and 0.005. We set $L=4$ and $K=4$ for all datasets. At the evaluation stage, we run the model for 500 epochs. DiffMG shares the same hyper-parameters but applies the author suggested training strategy. For general GNNs and ProtGNN, we set the layer to 4 and train them on homogeneous graphs where all types of edges are merged for 500 epochs. For GNN-based fraud detection methods, we use the parameters provided by the authors. We report the average score and standard deviation of 10 runs for each baseline. The division of datasets is similar to [21].

To alleviate the label imbalance problem, we apply the threshold-moving strategy [4] to NGS, DiffMG, ProtGNN, and general GNNs by setting the classification threshold to 0.2. GCN, GAT, and GraphSAGE are implemented based on DGL [31]. Other baselines are implemented based on author-provided source code.

5.2 Results

We use the same performance metrics as [21], i.e., F1-macro, AUC, and GMean. The experimental results are presented in Table 1. We have the following insightful observations from these results.

First, traditional GNNs treating all edges as single relations perform poorly on both datasets, while those baselines implemented on multi-relation graphs achieve promising results. This reflects that ignoring the semantic information encoded in different relations is harmful. NGS outperforms all baselines as it can automatically utilize task-dependent semantic information via meta-graph search.

Second, according to Table 1, compared with state-of-the-art fraud detectors, NGS boosts performance by 1%∼17% for all metrics on Amazon and 8%∼42% on YelpChi. It suggests that not all relations and neighbors are helpful for the downstream task. Those baselines fuse node representations from each relation view. Although they filter out or assign low weight to useless neighbor nodes, they are still affected by the noise from worthless relations.

Third, for ProtGNN, NGS improves the performance with 10%∼25% on Amazon and 38%∼82% on YelpChi. This is because ProtGNN treats all edges equally, which ignores semantic information and makes it hard to discover helpful prototypes. Moreover, ProtGNN skips the prototype projection because of its high computational complexity when dealing with the fraud detection task. Thus, the prototypes as explanations turn into incomprehensible vectors.

Fourth, DiffMG utilizes Gumbel-softmax sampling when searching the meta-graph in HINs. The sampling strategy inevitably affects the search space exploration, leading to a suboptimal meta-graph structure. NGS employs DARTS that tends to select proper relations for message passing to help correct the predictions [24].

Table 1: Performance comparison on Amazon and YelpChi

Method	Dataset	Amazon			YelpChi		
	Metric	F1-macro	AUC	GMean	F1-macro	AUC	GMean
Baselines	GCN	0.6571±0.0008	0.8189±0.0008	0.6629±0.0037	0.4963±0.0005	0.5504±0.0001	0.2143±0.0019
	GAT	0.5390±0.0021	0.7426±0.0020	0.3081±0.0173	0.5228±0.0070	0.5519±0.0012	0.2921±0.0193
	GraphSAGE	0.8383±0.0109	0.9149±0.0077	0.8518±0.0077	0.5781±0.0239	0.7409±0.0034	0.6815±0.0049
	CARE-GNN	0.8997±0.0064	0.9482±0.0044	0.8982±0.0015	0.6052±0.0170	0.7748±0.0008	0.7071±0.0035
	PC-GNN	0.8660±0.0164	0.9642±0.0035	0.8986±0.0203	0.6192±0.0479	0.8104±0.0057	0.7225±0.0166
	FRAUDRE	0.8519±0.1055	0.9408±0.0052	0.8847±0.0280	0.6057±0.0381	0.7582±0.0041	0.6862±0.0128
	AO-GNN*	0.8921±0.0045	0.9640±0.0020	0.9096±0.0105	0.7042±0.0051	0.8805±0.0008	0.8134±0.0232
	H^2-FDetector*	0.8392±0.0000	0.9689±0.0000	0.9203±0.0000	0.6944±0.0000	0.8877±0.0000	0.816±0.0000
	ProtGNN	0.7351±0.0112	0.8826±0.0106	0.7785±0.0126	0.5663 ±0.0024	0.6004 ±0.0056	0.4595 ±0.0196
	DiffMG	0.8826±0.0049	0.9290±0.0044	0.8855±0.0057	0.7316±0.0144	0.8799±0.0142	0.7873±0.0147
Ablation	$NGS_{\setminus A}$	**0.9234±0.0078**	0.9692±0.0136	0.9191±0.0087	0.7604±0.0227	0.9009±0.0215	0.7981±0.0279
Ours	NGS	0.9228±0.0046	**0.9736±0.0035**	**0.9218±0.0042**	**0.7828±0.0055**	**0.9218±0.0032**	**0.8351±0.0056**

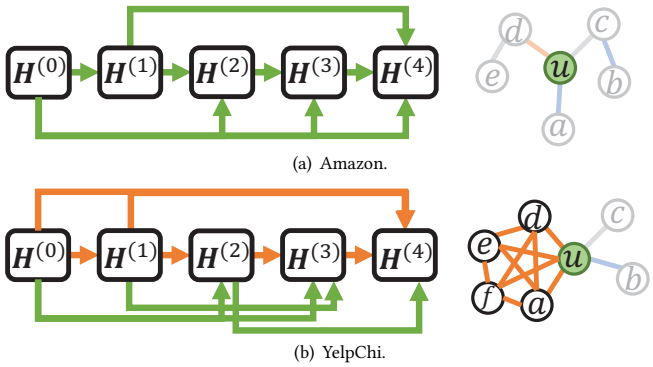
* The results are obtained from previous work, and H^2-FDetector does not report the variance in its experiment.

Experimental results demonstrate that NGS outperforms DiffMG, with 4%∼5% improvement in Amazon and 5%∼7% YelpChi.

Finally, for ablation, NGS exceeds or matches its variant for all metrics on two datasets. This is because multi-meta-graph aggregation can improve fault tolerance by assigning low weight to the meaningless meta-graph during evaluation, which offsets the impact of the previous search stage. The lower variance of NGS supports the conclusion too. Meanwhile, it also allows for more options in deciding which relations to propagate information along.

5.3 NGS Explainability

We visualize meta-graph instances discovered by NGS on Amazon and YelpChi in Figure 2. In Figure 2(a), we observe that no relation is involved in the meta-graph of Amazon, suggesting that the model does not aggregate neighborhood information from any relations, and user attributes are the key to identifying fraudsters in Amazon. In Figure 2(b), we observe that the R-U-R relation is highly relevant to fraud detection. It is consistent with the results of AO-GNN [13], with merely 0.04% R-U-R edges considered to be noise and pruned. According to the definition, the nodes in the R-U-R subgraph are distributed like clusters. In each cluster, the nodes are connected to each other as they are reviews posted by the same user. The searched meta-graph shows that the message is constantly passed and aggregated within the R-U-R cluster. It suggests that if a review is fraudulent, then most of the other reviews sent by the same poster are also fraudulent, reflecting a typical default phenomenon: click farming [32]. It should be stressed that the nodes' prediction depends only on the neighboring nodes involved in the meta-graph suggested message passing process, and that's where the explainability comes from. Visualization is only a means to show the meta-graph. The explanations suggest that when designing models in the future, the feature attributes of Amazon and the R-U-R relation of YelpChi need to be concerned. YelpChi may be more appropriate when evaluating graph-based fraud detection methods, and that is why our model has more significant improvements on the YelpChi.



(a) Amazon.

(b) YelpChi.

Figure 2: Meta-graphs searched by NGS and their corresponding message passing process on the example graph. The edge color denotes the relation type: green denotes the identity matrix I, and particularly in (b) orange denotes the R-U-R.

6 CONCLUSION

In this paper, we propose an explainable graph-based fraud detection method, NGS, which balances the performance and explainability. The framework of NGS can be decoupled into three components: formalizing the message passing of GNN as a meta-graph, employing DARTS to search the meta-graph, aggregating multiple meta-graphs to enhance stability and expressiveness. Experiments on two real-world datasets demonstrate that NGS outperforms previous state-of-the-arts. Moreover, the resulting explainable meta-graphs illustrate what factors lead to users being predicted as fraudulent, giving interesting intuitions regarding the tasks.

ACKNOWLEDGMENTS

The research work is supported by the National Natural Science Foundation of China under Grant (No.61976204, 92046003, U1811461). Xiang Ao is also supported by the Project of Youth Innovation Promotion Association CAS, Beijing Nova Program Z201100006820062. Yang Liu is also supported by China Scholarship Council.

REFERENCES

[1] Bing Bai, Jian Liang, Guanhua Zhang, Hao Li, Kun Bai, and Fei Wang. 2021. Why Attentions May Not Be Interpretable?. In *KDD*. 25–34.

[2] Jasmijn Bastings and Katja Filippova. 2020. The elephant in the interpretability room: Why use attention as explanation when we have saliency methods?. In *EMNLP (Workshop)*. 149–155.

[3] Bernardo Branco, Pedro Abreu, Ana Sofia Gomes, Mariana S. C. Almeida, João Tiago Ascensão, and Pedro Bizarro. 2020. Interleaved Sequence RNNs for Fraud Detection. In *KDD*. 3101–3109.

[4] Guillem Collell, Drazen Prelec, and Kaustubh R Patil. 2018. A simple plug-in bagging ensemble based on threshold-moving for classifying binary and multiclass imbalanced data. *Neurocomputing* 275 (2018), 330–340.

[5] Enyan Dai and Suhang Wang. 2021. Towards Self-Explainable Graph Neural Network. In *CIKM*. 302–311.

[6] Sarthika Dhawan, Siva Charan Reddy Gangireddy, Shiv Kumar, and Tanmoy Chakraborty. 2019. Spotting Collective Behaviour of Online Frauds in Customer Reviews. In *IJCAI*. 245–251.

[7] Yuhui Ding, Quanming Yao, Huan Zhao, and Tong Zhang. 2021. DiffMG: Differentiable Meta Graph Search for Heterogeneous Graph Neural Networks. In *KDD*. 279–288.

[8] Linfeng Dong, Yang Liu, Xiang Ao, Jianfeng Chi, Jinghua Feng, Hao Yang, and Qing He. 2022. Bi-Level Selection via Meta Gradient for Graph-Based Fraud Detection. In *Database Systems for Advanced Applications*. 387–394.

[9] Yingtong Dou, Zhiwei Liu, Li Sun, Yutong Deng, Hao Peng, and Philip S Yu. 2020. Enhancing graph neural network-based fraud detectors against camouflaged fraudsters. In *CIKM*. 315–324.

[10] Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive Representation Learning on Large Graphs. In *NeurIPS*.

[11] Binbin Hu, Zhiqiang Zhang, Chuan Shi, Jun Zhou, Xiaolong Li, and Yuan Qi. 2019. Cash-out user detection based on attributed heterogeneous information network with a hierarchical attention mechanism. In *AAAI*. 946–953.

[12] Binbin Hu, Zhiqiang Zhang, Jun Zhou, Jingli Fang, Quanhui Jia, Yanming Fang, Quan Yu, and Yuan Qi. 2020. Loan Default Analysis with Multiplex Graph Learning. In *CIKM*. 2525–2532.

[13] Mengda Huang, Yang Liu, Xiang Ao, Kuan Li, Jianfeng Chi, Jinghua Feng, Hao Yang, and Qing He. 2022. AUC-Oriented Graph Neural Network for Fraud Detection. In *WWW*. 1311–1321.

[14] Zhipeng Huang, Yudian Zheng, Reynold Cheng, Yizhou Sun, Nikos Mamoulis, and Xiang Li. 2016. Meta structure: Computing relevance in large heterogeneous information networks. In *KDD*. 1595–1604.

[15] Sarthak Jain and Byron C. Wallace. 2019. Attention is not Explanation. In *NAACL*. 3543–3556.

[16] Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR*.

[17] Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *ICLR*.

[18] Kuan Li, Yang Liu, Xiang Ao, Jianfeng Chi, Jinghua Feng, Hao Yang, and Qing He. 2022. Reliable Representations Make A Stronger Defender: Unsupervised Structure Refinement for Robust GNN. In *KDD*. 925–935.

[19] Hanxiao Liu, Karen Simonyan, and Yiming Yang. 2019. DARTS: Differentiable Architecture Search. In *ICLR*.

[20] Yang Liu, Xiang Ao, Fuli Feng, and Qing He. 2022. UD-GNN: Uncertainty-Aware Debiased Training on Semi-Homophilous Graphs. In *KDD*. 1131–1140.

[21] Yang Liu, Xiang Ao, Zidi Qin, Jianfeng Chi, Jinghua Feng, Hao Yang, and Qing He. 2021. Pick and Choose: A GNN-based Imbalanced Learning Approach for Fraud Detection. In *WWW*. 3168–3177.

[22] Zhiwei Liu, Yingtong Dou, Philip S Yu, Yutong Deng, and Hao Peng. 2020. Alleviating the inconsistency problem of applying graph neural network to fraud detection. In *SIGIR*. 1569–1572.

[23] Julian John McAuley and Jure Leskovec. 2013. From Amateurs to Connoisseurs: Modeling the Evolution of User Expertise through Online Reviews. In *WWW*. 897–908.

[24] Yijian Qin, Xin Wang, Zeyang Zhang, and Wenwu Zhu. 2021. Graph Differentiable Architecture Search with Structure Learning. In *NeurIPS*.

[25] Shebuti Rayana and Leman Akoglu. 2015. Collective Opinion Spam Detection: Bridging Review Networks and Metadata. In *KDD*. 985–994.

[26] Nick F. Ryman-Tubb, Paul Krause, and Wolfgang Garn. 2018. How Artificial Intelligence and machine learning research impacts payment card fraud detection: A survey and industry benchmark. *Engineering Applications of Artificial Intelligence* 76 (2018), 130–157.

[27] Fengzhao Shi, Yanan Cao, Yanmin Shang, Yuchen Zhou, Chuan Zhou, and Jia Wu. 2022. H2-FDetector: A GNN-Based Fraud Detector with Homophilic and Heterophilic Connections. In *WWW*. 1486–1494.

[28] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. In *ICLR*.

[29] Daixin Wang, Jianbin Lin, Peng Cui, Quanhui Jia, Zhen Wang, Yanming Fang, Quan Yu, Jun Zhou, Shuang Yang, and Yuan Qi. 2019. A semi-supervised graph attentive network for financial fraud detection. In *ICDM*. 598–607.

[30] Jianyu Wang, Rui Wen, Chunming Wu, Yu Huang, and Jian Xion. 2019. FdGars: Fraudster Detection via Graph Convolutional Networks in Online App Review System. In *WWW*. 310–316.

[31] Minjie Wang, Da Zheng, Zihao Ye, Quan Gan, Mufei Li, Xiang Song, Jinjing Zhou, Chao Ma, Lingfan Yu, Yu Gai, et al. 2019. Deep graph library: A graph-centric, highly-performant package for graph neural networks. *arXiv preprint arXiv:1909.01315* (2019).

[32] Yuanyuan Wu, Eric W.T. Ngai, Pengkun Wu, and Chong Wu. 2020. Fake online reviews: Literature review, synthesis, and directions for future research. *Decision Support Systems* 132 (2020), 113280.

[33] Ge Zhang, Jia Wu, Jian Yang, Amin Beheshti, Shan Xue, Chuan Zhou, and Quan Z Sheng. 2021. FRAUDRE: Fraud Detection Dual-Resistant to Graph Inconsistency and Imbalance. In *ICDM*. 867–876.

[34] Zaixi Zhang, Qi Liu, Hao Wang, Chengqiang Lu, and Cheekong Lee. 2022. ProtGNN: Towards Self-Explaining Graph Neural Networks. In *AAAI*.

[35] Huan Zhao, Quanming Yao, Jianda Li, Yangqiu Song, and Dik Lun Lee. 2017. Meta-Graph Based Recommendation Fusion over Heterogeneous Information Networks. In *KDD*. 635–644.

[36] Qiwei Zhong, Yang Liu, Xiang Ao, Binbin Hu, Jinghua Feng, Jiayu Tang, and Qing He. 2020. Financial defaulter detection on online credit payment via multi-view attributed heterogeneous information network. In *WWW*. 785–795.

[37] Xiaoqian Zhu, Xiang Ao, Zidi Qin, Yanpeng Chang, Yang Liu, Qing He, and Jianping Li. 2021. Intelligent financial fraud detection practices in post-pandemic era. *The Innovation* 2, 4 (2021), 100176.