MICA: Multi-channel Representation Refinement Contrastive Learning for Graph Fraud Detection

Guifeng Wang[1](✉), Disheng Tang[2], Anatoli Shatsila[3], and Xuecang Zhang[1]

[1] Huawei Technologies Co., Ltd., Hangzhou, Zhejiang, China
wgf1109@mail.ustc.edu.cn, {wangguifeng4,zhangxuecang}@huawei.com
[2] School of Life Science, Tsinghua University, Beijing, China
dstang@mail.tsinghua.edu.cn
[3] Jagiellonian University, Kraków, Poland
anatoli.shatsila@student.uj.edu.pl

Abstract. Detecting fraudulent nodes from topological graphs is important in many real applications, such as financial fraud detection. This task is challenging due to both the class imbalance issue and the camouflaged behaviors of anomalous nodes. Recently, some graph contrastive learning (GCL) methods have been proposed to solve the above issue. However, local aggregation-based GNN encoders can not consider the long-distance nodes, leading to over-smoothing and false negative samples. Also, random perturbation data augmentation hinders separately considering camouflaged behaviors at the topological and feature levels. To address that, this paper proposes a novel contrastive learning architecture for enhancing the performance of graph fraud detection. Specifically, a context generator and a representation refinement module are embraced for mitigating the limitation of local aggregation in finding long-distance fraudsters, as well as the introduction of false negative samples in GCL. Further, a multi-channel fusion module is designed to adaptively defend against diverse camouflaged behaviors. The experimental results on real-world datasets show a significant performance improvement over baselines, which demonstrates its effectiveness.

Keywords: Graph Fraud Detection · Graph Contrastive Learning

1 Introduction

While enjoying the benefits of the surge in users under the convenience of the Internet, all walks of life have also incubated various fraudulent activities. For instance, fraudsters may create malicious accounts on payment platforms [18], spread rumors (e.g., by posting fake reviews) on e-commerce platforms [11] or make fictitious claims in the insurance industry [12]. Since many fraudulent activities are performed by multiple entities, graph-based fraud detection methods which are able to discover and incorporate structural patterns have naturally become the target of attention in both academia and industry.

X. Song et al. (Eds.): APWeb-WAIM 2023, LNCS 14334, pp. 31–46, 2024.
https://doi.org/10.1007/978-981-97-2421-5_3

The mainstream graph fraud detection methods are Graph Neural Networks (GNNs) based learning models, which have leveraged the power of message passing to learn node representations with the goal of identifying the fraudsters in the embedding space [17,30]. They work well when features and topology keep consistent with the labels [26]. However, this assumption is broken because the graph-based detectors have motivated perpetrators to artificially disturb networks to camouflage their activities. For example, fraudsters disguise themselves among a group of benign users by modifying features (e.g., camouflaged features) or adding connections (e.g., camouflaged links) to many normal users. Hence, the diverse camouflaged behaviors pose challenges to graph fraud detection.

In addition, to tackle the class imbalance issue, there are also some graph contrastive learning (GCL) attempts for fraud detection [27], which show a good performance and are becoming the trend in this area. However, there are three main limitations that severely hinder obtaining significant node representations and degrade the performance of fraud detection. First, most GCL methods employ GNNs in the form of local aggregation as encoders, which makes the representation of samples averaged or smooth in a local scope. Second, negative samples are typically randomly selected from distant nodes, and it is challenging for GNNs to capture long-distance yet similar nodes, resulting in false negative samples. Third, random perturbation way commonly used in data augmentation may lose some useful structure or attribute information, which hinders the analysis of diverse camouflaged behaviors.

Motivated by the above gaps, the purpose of this paper is to alleviate the above issues thereby improving the performance of fraud detection. To address that, based on a general contrastive learning framework, a **M**ulti-channel representation ref**I**nement **C**ontrastive learning method for fr**A**ud detection (**MICA** for abbreviation) is innovatively proposed. Specifically, we first embrace a context generator for capturing global information, hoping to learn similar camouflage patterns of fraudsters even in a long-range situation. Based on it, an augmentation-agnostic representation refinement module works on representation space, for the purpose of reducing false negative samples and mitigating over-smoothing. Then, considering that the random perturbation way may disturb the analysis of where the camouflaged behaviors come from, an adaptive and fine-grained multi-channel fusion module is designed to defend against diverse attacks. The contributions can be summarized as follows:

- To our best knowledge, this is the first paper to simultaneously explore the problems of over-smoothing, false negative samples, and the conflicts of random perturbation data augmentation under camouflage behaviors in the GCL framework for graph fraud detection.
- We proposed a general MICA model which solved the above limitations of existing GCL solutions, the adequate experiments show that our method achieves significant improvements for the fraud detection tasks.

2 Related Work

2.1 GNN-Based Fraud Detection

There are some survey works about graph anomaly or fraud detection recently [19], the key challenges referring to fraud detection include the existence of camouflage behaviors, the class imbalance issue, and data scarcity. The camouflage behavior makes the graph vulnerable to topology and features, resulting in smoother node representations obtained by GNNs, and thus fraudsters are indistinguishable from normal nodes. To solve that, some previous studies have designed some strategies such as neighbor filtering [13], adversarial learning [28] or active generative learning [7] to achieve robustness and generalization in the presence of fraudsters. As for the class imbalance issue, some undersampling [5] and data augmentation [14] methods are employed. In addition, Some recent approaches like active learning [21], meta learning [4], and data augmentation [31] are proposed to solve the data scarcity problem. Although these methods have explored fraud detection from varied perspectives and achieved effective results, few of them consider these issues simultaneously.

2.2 Graph Contrastive Learning Way

Due to the advantages of graph contrastive learning (GCL) models in various fields, some methods have also been utilized to learn node representation for fraud/anomaly detection. Such methods generally assume that abnormal users can be distinguished by structural patterns. However, there is a problem in that the structural patterns and the label semantics are not consistent, which is a vital factor that impacts the representation learning found by DCI [27] method. Hence, the DCI method injects a clustering step in the GCL scheme to reduce data inconsistency. In addition, a graph contrastive coding-based method GCCAD [2] is proposed for contrasting abnormal nodes with normal ones in terms of their distances to the global context, with scarce labels in a self-supervised way. The above methods do not consider the attributed networks, so CoLA [15] was designed to learn informative embedding from high-dimensional attributes and local structure, and measure the anomaly score for each instance pair. Although the above GCL-based studies have been proposed, the basic limitation of GCL methods for fraud detection has not been explored.

3 Methodology

In this section, we first formulate the problem and introduce the overview of the proposed MICA framework, then systematically explore each module for the fraud detection task.

3.1 Problem Definition

In fraud detection problem on graph $\mathbf{G} = (\mathcal{V}, \mathcal{E}, \mathcal{X})$ with n nodes, each node $v_i \in \mathcal{V}$ with features $x_i \in \mathcal{X} \in \mathbb{R}^{n \times d}$ represents the target entity whose suspiciousness needs to be justified. For example, to detect fake reviews on the

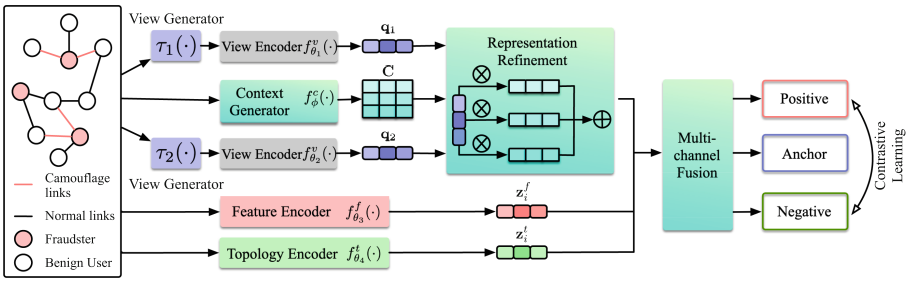


Fig. 1. The architecture of our MICA model. Based on the general GCL framework, our MICA additionally designed the context generator, representation refinement module, and multi-channel fusion module.

e-commerce platform, the target entities are genuine and fake reviews. Correspondingly, nodes have labels $\mathbf{Y} \in \{0, 1\}$, where 0 denotes benign and 1 represents suspicious. The edge $(i, j) \in \mathcal{E}$ links node v_i and v_j due to some certain relationships or shared attributes, e.g., two reviews from the same user or posted from the same devices. Hence, the fraud detection problem is a binary node classification task on the graph.

3.2 Overview of Proposed MICA

As explained above, the false negative samples and over-smoothing caused by the GNN-based view encoder, and the conflicts of random perturbation way with existing diverse camouflaged attacks pose certain challenges to GCL methods. Thanks to the superiority of the CL model for this problem [27], we make improvements based on the GCL framework. As shown in Fig. 1, we additionally design three components in our MICA model, namely context generator, representation refinement, and multi-channel fusion. The context generator is to learn a mapping function $f^c_\phi(\cdot)$ from a global scope with notable normal and fraudster patterns (Sect. 3.3). Based on it, the augmentation-agnostic representation refinement module is proposed to map each view (e.g., $\mathbf{q}_1, \mathbf{q}_2$) into a unified space and refine their distances for mitigating the averaged embedding and false negative samples (Sect. 3.4). Besides, to explore different camouflaged behaviors under random perturbation, an adaptive and fine-grained multi-channel fusion module is designed (Sect. 3.5). Finally, a supervised contrastive loss is applied on the anchor, positive and negative samples for learning downstream relevant representations (Sect. 3.6).

3.3 Context Generator

In order to alleviate the false negative samples and the over-smoothing brought by local aggregation-based GNNs, we put forward a context generator to obtain the information from a global scope, then local feature differences are easier

to be distinguished under its transformation. For example, for a fraud node that intentionally connects many normal nodes, it is easier to recognize the camouflage nature of this behavior by searching for similar cheating patterns from distant abnormal nodes.

Context Definition. The *context* represents a representation space with a global view, which could be sub-graphs of m nodes with d-dimensional node feature $\mathbf{X}^c \in \mathbb{R}^{m \times d}$ and topological information $\mathbf{A}^c \in \mathbb{R}^{m \times m}$. In our experiments, we consider context as a whole graph for a stable and global expression, which means $n = m$ in such a case. Inspired from attention mechanism [1], we calculate it via the keys $\mathbf{K} \in \mathbb{R}^{m \times k}$ and values $\mathbf{V} \in \mathbb{R}^{m \times v}$, k and v are the query/key depth and value depth respectively:

$$\mathbf{K} = \mathcal{F}_K(\mathbf{X}^c, \mathbf{A}^c) \tag{1}$$

$$\mathbf{V} = \mathcal{F}_V(\mathbf{X}^c, \mathbf{A}^c) \tag{2}$$

Here the function $\mathcal{F}_K(\cdot)$ and $\mathcal{F}_V(\cdot)$ are keys and values generators, which can be linear/non-linear functions only considering the feature information or graph-based algorithms like graph convolutional networks (GCN). We implemented several solutions for the generator, whose results could be seen in Table 2.

Generating the Context Map. With the generated keys and values, we wish to generate a linear function $f_\phi^c(\cdot) : \mathbb{R}^k \to \mathbb{R}^v$, i.e., a matrix $\mathbf{C} \in \mathbb{R}^{k \times v}$. We start by normalizing the keys across the nodes in a context via a softmax operation, which is $\overline{\mathbf{K}} = \text{softmax}(\mathbf{K}, axis = m)$, then the matrix $\mathbf{C} = \overline{\mathbf{K}}^{\mathrm{T}}\mathbf{V}$ is obtained by using the normalized keys $\overline{\mathbf{K}} \in \mathbb{R}^{m \times k}$ to aggregate the values $\mathbf{V}$.

Note that we choose to use different model parameters to generate keys and values, which can represent different levels of context. After the above operation, it is equivalent to performing screening in the context and retaining the expressive information. That is to say, some notable patterns of benign users and fraudsters remain in the context map.

3.4 Representation Refinement

With the notable patterns in the context map, to capture the long-distance but similar nodes, it is natural to refine the distance of nodes by transforming their hidden representations. Similarly, we denote n nodes as the *queries* $\mathcal{Q} = \{\mathbf{q}_1, \mathbf{q}_2, ..., \mathbf{q}_n\} \in \mathbb{R}^{n \times k}$. In a graph, although each query can be calculated by topology and features, here we think that only considering node features is more conducive to interacting with the context. Therefore, a query is generated by linearly mapping the node features to a specific space via function $\mathcal{F}_Q(\cdot)$:

$$\mathbf{Q} = \mathcal{F}_Q(\mathbf{X}) = \mathbf{X}\mathbf{W}_Q \tag{3}$$

Here $\mathbf{W}_Q \in \mathbb{R}^{d \times k}$ is the parameter transforming node features $\mathbf{X} \in \mathbb{R}^{n \times d}$ into queries. For a graph, the interactions between nodes and subgraphs as well

as the whole graph can be useful. Essentially, we can treat each row of matrix $\mathbf{C}$ as a basis, that is $\mathbf{C} = \{\mathbf{C}_1, \mathbf{C}_2, ..., \mathbf{C}_k\}$, and each basis maps each dimension of the query $\mathbf{q}_i = \{q_1, q_2, ..., q_k\}$ to get a unified representation from a global scope. Based on that, we apply the context matrix to the query $\mathbf{q}_i$ for obtaining the context-aware embedding of node v_i, which is denoted as $\mathbf{z}_i^c \in \mathbb{R}^v$:

$$\mathbf{z}_i^c = \mathbf{C}^{\mathrm{T}}\mathbf{q}_i = \mathbf{C}_1 q_1 + \mathbf{C}_2 q_2 + ... + \mathbf{C}_k q_k \tag{4}$$

The context matrix $\mathbf{C}$ is shared across all queries and is invariant to the permutation of the context elements, which helps distinct nodes with similar patterns to be close to each other in the representation space. Especially for fraudsters, their embeddings are closer even if they are not directly connected.

3.5 Multi-channel Fusion

Although the context matrix maps each node to a global space, the random permutation way of data augmentation in GCL usually randomly masks the node attributes or edges, which hinders the exploration of analyzing the effect of different elusive camouflaged behaviors, therefore we further conduct an adaptive multi-channel way to better represent the contrastive views.

Concretely, apart from the context-aware embedding $\mathbf{z}_i^c \in \mathbb{R}^v$ of node v_i, we also obtain the feature-aware embedding $\mathbf{z}_i^f$ and topology-aware embedding $\mathbf{z}_i^t$ with the same dimension via MLP operation:

$$\mathbf{z}_i^f = f_{\theta_3}^f(\mathbf{x}_i) = \mathrm{MLP}\left(\mathbf{x}_i\right) \tag{5}$$

$$\mathbf{z}_i^t = f_{\theta_4}^t(\mathbf{A}) = \mathrm{MLP}\left(\tilde{\mathbf{u}}_i\right) \tag{6}$$

For the feature encoder $f_{\theta_3}^f(\cdot)$ and topology encoder $f_{\theta_4}^t(\cdot)$, the MLP consists of three linear layers with the exponential linear unit as the activation functions. Here $\tilde{\mathbf{u}}_i$ represents the top r dimensions of the eigen vector of node v_i, which is calculated by the eigen decomposition [10] based on adjacency matrix $\mathbf{A}$, i.e., $\mathbf{A} = \mathbf{U}\Lambda\mathbf{U}^{-1}$. Except for the eigen decomposition way to obtain the topological information, other graph embedding methods such as random walks can be tried.

To explore their contributions in the final representation, we employ a gated way to adaptively adjust their weights. Concretely, we calculate the activation vectors $\boldsymbol{\alpha}_i^c, \boldsymbol{\alpha}_i^f, \boldsymbol{\alpha}_i^t$ for the context gate, feature gate, and topology gate respectively based on the embedding $\mathbf{z}_i^c, \mathbf{z}_i^f, \mathbf{z}_i^t$. For space limitation, we give the calculation process of $\boldsymbol{\alpha}_i^c$ as an example, the same process but with linear function $\mathcal{F}_\alpha^f(\cdot)$ and $\mathcal{F}_\alpha^t(\cdot)$ for the other two gated vectors:

$$\boldsymbol{\alpha}_i^c = \sigma\left(\mathcal{F}_\alpha^c\left(\mathbf{z}_i^c \| \mathbf{z}_i^f \| \mathbf{z}_i^t\right)\right) \tag{7}$$

This fine-grained way allows a good trade-off between the refined representation and original information, so as to automatically explore the camouflaged

attack under random perturbation. Then the final embedding $\mathbf{z}_i$ is the weighted sum of the above three, which serves as the contrastive samples.

$$\mathbf{z}_i = \boldsymbol{\alpha}_i^c \circ \mathbf{z}_i^c + \boldsymbol{\alpha}_i^f \circ \mathbf{z}_i^f + \boldsymbol{\alpha}_i^t \circ \mathbf{z}_i^t \tag{8}$$

Here $\mathcal{F}_\alpha^c(\cdot), \mathcal{F}_\alpha^f(\cdot), \mathcal{F}_\alpha^t(\cdot)$ are linear functions for each channel, operator $\|$ means concatenation, operator $\circ$ represents Hadamard product, and $\sigma(\cdot)$ is the Sigmoid function. After the pre-processing phase, we load the learned parameters to initialize the encoder in fine-tuning process, and the classifier is trained with a cross-entropy loss.

3.6 Supervised Contrastive Loss

On the basis of node representation, we can construct anchors and positive as well as negative samples for contrastive learning. In general, self-supervised loss functions like InfoNCE are adopted for training in CL. In our case, since the camouflage behavior disturbs the original input data, we strongly need the label information to perform some bias correction in the representation space. In addition, according to the conclusions of some recent studies [22], incorporating supervised information helps to reduce the mutual information between contrastive views while keeping task-relevant information intact.

The representation of the anchor, the positive and negative sample is separately defined as $\mathbf{z}_o, \mathbf{z}_p, \mathbf{z}_n$, then the supervised contrastive loss [8] with temperature parameter τ in the pre-training process, adapting it to the fully supervised setting for leveraging label information is formulated as:

$$\mathcal{L}_{SupCon} = -\sum_{o \in \mathcal{V}} \frac{1}{|P(o)|} \sum_{p \in P(o)} \log \frac{\exp(\mathbf{z}_o \cdot \mathbf{z}_p / \tau)}{\sum_{j \in \mathcal{V} \setminus \{o\}} \exp(\mathbf{z}_o \cdot \mathbf{z}_n / \tau)} \tag{9}$$

Here $P(o) \equiv \{p \in \mathcal{V} \setminus \{o\} : y_p = y_o\}$ is the set of indices of all positives distinct from node v_o but sharing the same label, and $|P(o)|$ is its cardinality. If the anchor is graph embedding without label information, then positive and negative nodes are selected in terms of their labels.

3.7 Model Discussion

It is worth noting that our framework is compatible with the major components in CL, whether it is data augmentation strategies, pretext tasks (e.g., the same scale views like node-node pair or cross-scale views like node-graph pair) or contrastive objectives. We explore the varied strategies in the experiments (see Sect. 4.4). Besides, the method of generating context based on the keys and values has a small number of parameters. Concretely, in regular attention mechanism, the attention map calculated by $\mathbf{Q}\mathbf{K}^\top$ has $\mathcal{O}(n * m)$ space complexity with n inputs (which is used to generate queries) and m context (which is used to generate the keys and values). While in our model, we obtain the context map by $\mathbf{K}^\top\mathbf{V}$ with $\mathcal{O}(k * v)$ space complexity. Generally, it is called local attention

Table 1. The Statistics of Datasets.

Dataset	#Nodes	#Edges	Fraud	HR
Amazon	11,944	4,404,364	6.87%	0.91
YelpChi	45,954	3,869,956	14.53%	0.77

when the context length m is smaller than the number of samples n ($m \ll n$), while global attention is when $n = m$. Since the key depth k will be set to a small value in practice, the final output dimension v is also smaller than the number of inputs, we have a much smaller memory cost. That is to say, it is an efficient calculation way compared to regular attention mechanisms.

4 Experiments

In the experiments, we mainly focus on verifying the effectiveness (in Sect. 4.2 and Sect. 4.3), expandability (in Sect. 4.4) and explainability (in Sect. 4.5).

4.1 Experimental Settings

Datasets. For the sake of fully verifying the above questions, we conduct experiments on two widely used real-world datasets (i.e., Amazon and YelpChi) whose statistics can be found in Table 1. Here we calculate the node-level homophily ratio (HR) for a better understanding of the camouflage links in the datasets.

Baselines. In order to fully analyze the performance of the model, we compare it with the following models from three categories:

Message passing GNNs have proved to be powerful in a variety of tasks on graphs. We select three architectures that are trained in an end-to-end manner. **GCN** [9], **GAT** [23] and **GPR-GNN** [3]. Specifically, GPR-GNN is a more powerful GNN that utilizes adaptive multi-hop aggregation to avoid over-smoothing and learn difficult label patterns.

GCL-based models show great potential in handling labels related issues, which is crucial in anomaly detection tasks. We choose **Deep Graph Infomax (DGI)** [24] and recently proposed **Deep Cluster Infomax (DCI)** [27] (which also falls into the Fraud Detection category). Besides, we also design a general GCL method called **GCN+SupCL** as a baseline, it uses GCN as the encoder in a supervised contrastive loss-based CL.

Fraud detection schemes include models which were proven to perform well in anomaly detection tasks. In this category we consider **CARE-GNN** [5], **GeniePath** (with its variant **GeniePathLazy**) [17], **FRAUDRE** [29] and **PC-GNN** [13]. CARE-GNN leverage reinforcement learning (RL) to effectively deal with camouflaged fraudsters and GeniePath utilizes multi-hop attention to learn receptive paths and propagate signals in the graph in a more effective fashion.

Table 2. Overall evaluation on YelpChi and Amazon datasets.

Method		YelpChi				Amazon			
		AUC (%)	AP (%)	Recall (%)	Acc. (%)	AUC (%)	AP (%)	Recall (%)	Acc. (%)
Baselines	GCN	61.31	25.32	50.02	85.47	74.40	20.72	50.02	93.12
	GAT	63.31	26.72	50.81	85.59	78.19	26.77	51.77	93.12
	GPR-GNN	83.65	52.77	63.07	87.48	96.57	84.54	88.48	97.99
	GCN+SupCL	59.99	22.37	50.07	85.47	85.35	39.35	54.11	93.36
	DGI	66.88	29.64	53.71	85.71	83.24	40.47	58.36	93.60
	DCI	66.72	29.44	53.42	85.63	83.29	42.08	64.64	93.32
	CARE-GNN	79.21	42.24	72.00	71.16	95.20	86.30	88.47	96.19
	GeniePath	76.90	39.78	56.70	86.13	77.40	33.33	57.75	94.11
	GeniePathLazy	84.93	56.61	64.64	88.05	96.65	85.82	88.56	**98.10**
	FRAUDRE	83.82	52.35	75.00	72.72	95.88	87.68	89.86	96.23
	PC-GNN	83.76	55.05	70.00	68.43	94.96	81.86	88.58	86.86
Variants	ICA(GPR)	77.81	42.90	62.54	85.97	94.30	82.15	87.76	97.72
	ICA(MLP)	83.47	51.69	63.26	87.33	95.53	83.13	86.92	97.72
	ICA(GCN)	86.67	62.28	69.53	89.03	97.60	87.58	87.46	97.88
	MICA(topo.)	86.92	62.28	69.74	88.84	97.76	87.51	87.46	97.76
	MICA(feat.)	88.90	66.97	74.52	**89.57**	97.79	88.07	**91.22**	97.75
	MICA	**89.36**	**67.94**	**75.33**	89.56	**98.01**	**88.89**	90.95	97.84

FRAUDRE considered the multi-relation among users and unified the graph-agnostic embedding and fraud-aware graph convolution module into a GNN framework, while PC-GNN proposed a pick-and-choose step to sample neighborhoods and getting the final node embeddings by aggregating neighbor information under different relations. Although there are more architectures designed for anomaly detection (e.g., Player2Vec [30], SemiGNN [25] and GraphConsis [16]), we decided not to include them in our study because these recently published methods outperform them significantly on both datasets.

Evaluation Metrics. In fraud detection problems we are naturally more interested in correctly identifying fraudsters (positive instances). Moreover, as we mentioned before, the classes are naturally imbalanced, which is reflected in both datasets. For the above reasons, we utilize four metrics: ROC-AUC (AUC), Average precision score (AP), Recall and Accuracy (Acc.).

Parameters Settings. For fair comparison, we separately adopt 40% and 60% of whole data as training set and testing set, and use the following unified setup for all models: hidden dimensions = 64, number of epochs = 1000 on YelpChi and 300 on Amazon, batch size = 1024 for YelpChi and 256 for Amazon, learning rate = 0.002 and L2 regularization weight = $5e^{-4}$. In our MICA model, the representation dimension of query/key $k = 16$ and the representation dimension of value $v = 64$. All of the models are trained with Adam optimizer, the results are averaged on 20 run times. We implement code based on PyTorch Geometric [6]. The code is available at https://github.com/goiter/anomaly_detection.git.

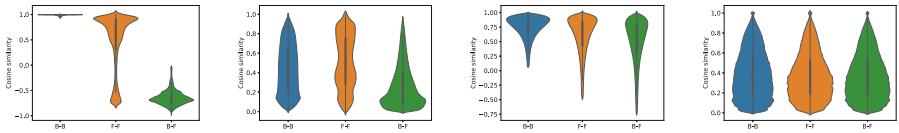
4.2 Overall Evaluation

In addition to conducting experiments on the baselines, we also implement some variants of our models from two perspectives.

- **ICA**: In order to focus on verifying the performance of the context generator and representation refinement module, we name a method as ICA which is our MICA model without the multi-channel fusion module. Besides, we implement the same function but not sharing parameters for keys and values generator (i.e., the function $\mathcal{F}_K(\cdot)$ and $\mathcal{F}_V(\cdot)$ in Eq. (1)) in three ways: GPR-GNN, MLP and GCN. We refer to these models as **ICA(GPR)**, **ICA(MLP)** and **ICA(GCN)** respectively.
- **MICA**: This is our proposed model which includes all modules in Fig. 1 with the keys (and values) generator implemented as GCN. While **MICA(topo.)** and **MICA(feat.)** are the variants of MICA fused with only the topology-aware embedding and the feature-aware embedding respectively.

According to the results shown in Table 2, we summarize our conclusions as follows: (1) **The proposed model MICA contributes to fraud detection significantly.** Even without the multi-channel fusion module, our ICA(GCN) outperforms the baselines on almost all evaluation metrics. The performance improvement between ICA methods and GCL-based methods like DCI indicates that the context generator and representation refinement module have a certain effect on solving the problem of false negative samples. (2) **The multi-channel fusion module is more advantageous in heterophilic datasets**. By comparing the results of ICA(GCN) and the variants of MICA, it is obvious that the multi-channel module plays an important role in the overall performance of YelpChi. We hypothesize that the main reason for that is the dataset being relatively heterophilic (with 0.77 homophily ratio [20]). That is to say, there is a considerable portion of connected nodes belong to different classes. In this situation, it may be necessary to pay more attention to the context and feature information. (3) **The results under different context generators reflect that the context should retain as much information as possible from a global scope.** By comparing the results under the variants of ICA, we can see that the GCN-based context generator performs best. Although the GPR-GNN method achieves impressive results in a supervised manner, the performance of ICA(GPR) is not as good as that of ICA(MLP), indicating that the context in the pre-training phrase requires as comprehensive information as possible.

Besides, according to the results of the baselines, we can draw some observations: (1) The widely used GNN methods (i.e., GCN and GAT) fail to achieve good results on fraud detection datasets, while the results of GPR-GNN are impressive due to its ability to adaptively learn the weights in order to optimize nodes features and topological information extraction process. (2) The performance of the GCL-based models (i.e., GCN+SupCL, DGI, and DCI) on these two datasets is not very satisfactory. The results of GCN+SupCL on the YelpChi dataset are worse than the GCN method but better on the Amazon dataset, which illustrates that graph contrastive learning methods are not necessarily better than GNNs, and more powerful view encoders are needed. As for DGI



(a) MICA (Amazon) (b) DCI (Amazon) (c) MICA (YelpChi) (d) DCI (YelpChi)

Fig. 2. Comparison of cosine similarity of MICA and DCI between benign-benign (B-B), fraudster-fraudster (F-F) and benign-fraudster (B-F) users. Sub-figure (a) and (c) are the results of MICA while Sub-figure (b) and (d) are from DCI.

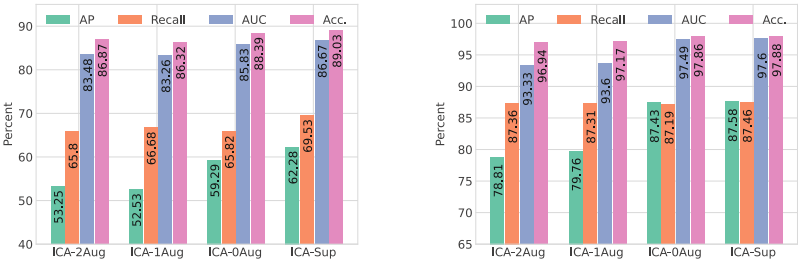
and DCI, we believe that for larger datasets, contrasting each node to the full graph or some feature-based clusters is not good enough to separate them with camouflaged behaviors. (3) CARE-GNN, GeniePathLazy, FRAUDRE, and PC-GNN also perform well. By selecting neighbors and considering the multi-relation on edges, CARE-GNN and PC-GNN deeply mitigate the attack brought by the topological camouflage behavior. FRAUDRE which investigated aspects of the features, topologies, and relations proved the ability in solving the heterophily issue. GeniePathLazy with the individual feature map in the multi-hop attention has achieved better performance than GeniePath, which also verifies the necessity of our proposed multi-channel fusion module.

4.3 Visualization on Distinguishable Representations

In order to verify the motivation and intuitively prove that our MICA model learned better representations and mitigated the over-smoothing, we compared the pairwise cosine similarity of node embeddings with DCI methods. Figure 2 shows the violin plot of inner-class (B-B, F-F) and inter-class (B-F) similarities, which is a box plot with the addition of a rotated probability density and the average value represented as white dots. On one hand, according to Fig. 2(a) and Fig. 2(b), we can clearly observe that the inner-class similarity of MICA on the Amazon dataset is higher than that of DCI, and the opposite for inter-class. That is to say, MICA has distinct distributions for inner- and inter-class similarities, while the distributions of DCI are closer and more difficult to distinguish. On the other hand, the YelpChi dataset has a lower homophily ratio, implying the prevalence of structural anomaly compared with the Amazon dataset. According to Fig. 2(c) and Fig. 2(d), though with overlapped distributions, for MICA there's a clear difference between the means of inner- and inter-class distributions, while we can barely observe any difference between the counterparts from DCI. In a word, our MICA model makes node representation more distinguishable on inner- and inter-class, which proved that MICA alleviated the over-smoothing issue and is more beneficial to detect camouflaged behaviors.

4.4 Expandability and Ablation Study on ICA

In Sect. 4.2 we analyze the performance of our ICA model and its variants with different context generators, which is one of the ablation studies. Besides, due to



(a) Variants of ICA (YelpChi) (b) Variants of ICA (Amazon)

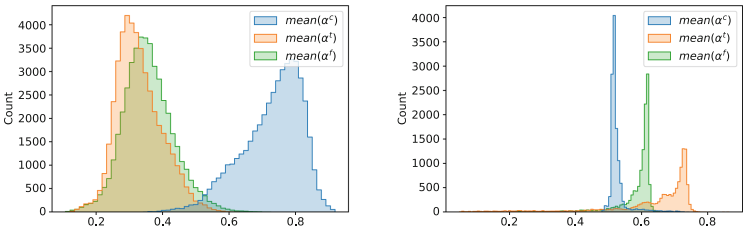
ICA Variants	Contrastive Modes	Contrastive Losses	Augmentation Way		
			Anchor	Positives	Negatives
ICA-2Aug	node-graph pair	self-supervised	aug1	aug1	aug2
ICA-1Aug	node-graph pair	self-supervised	-	-	aug
ICA-0Aug	node-graph pair	supervised	-	-	-
ICA-Sup	node-node pair	supervised	-	-	-

Fig. 3. Comparison of our ICA model on varied contrastive modes, training losses and data augmentation strategies for contrastive views. The different settings of the ICA variants are listed in the table.

the expandability of our proposed framework, it supports different contrastive modes, contrastive losses and data augmentation strategies for contrastive views. Therefore, in this section we analyze the effects of them. For fair comparison, we merely conduct variant experiments in our ICA model.

As the table in Fig. 3 shows, we implement the ICA method with varied contrastive modes, contrastive losses and data augmentation strategies. Concretely, the node-graph pair contrastive mode is to treat the graph embedding as the anchor and node embeddings as positive/negative samples. The graph embedding is obtained by sum pooling on node embeddings. Similarly, the anchor and contrastive samples in node-node pair contrastive mode are all base on node embeddings. The samples in ICA-2Aug are all obtained by two kinds of data augmentation methods (i.e., aug1 and aug2). Differently, the data augmented nodes as negative samples in ICA-1Aug, while the original nodes and their corresponding graph representation are used as positive samples and the anchor. Note that we implement the data augmentation by randomly removing the edges with a 0.2 dropout rate and masking the features with a 0.1 probability.

By observing Fig. 3 we can draw the following conclusions. Firstly, the methods training with supervised contrastive loss (i.e., ICA-0Aug and ICA-Sup) outperforms the ones with self-supervised loss (i.e., ICA-2Aug and ICA-1Aug), which verifies that leveraging label information in pre-training makes representations keeping task-relevant information. Secondly, compared with the results of ICA-Sup and ICA-0Aug, it shows that the node-node pair contrastive mode may be more suitable for this node-level fraud detection task than the node-graph



(a) Gates Distribution (YelpChi) (b) Gates Distribution (Amazon)

Fig. 4. Distributions of the context gate, feature gate, and topology gate in MICA. Each data point in the x-axis is obtained by averaging over the feature dimension.

pair. In addition, it may also be due to the design of the context generator and representation refinement module enriching the node samples with notable pattern information, making it more effective and intuitive when doing the sample contrastive learning. Thirdly, by comparing the results on simply designed data augmentation strategies, using different data augmentation strategies in ICA-2Aug and ICA-1Aug does not seem to have much effect on the results, which implies that the representation refinement module is augmentation-agnostic. In a summary, all the above ablation studies show the expandability and effectiveness of our proposed ICA architecture.

4.5 Explainability of Multi-channel Fusion Module

Table 2 indicates that the feature-aware embedding alone (MICA(feat.)) brings the most significant improvement to our ICA model, and it will continue to advance with the addition of topology-aware embedding (MICA). However, it is impossible to directly measure the contributions of each module solely depending on Table 2, since the gating module relies upon the coupling of the involved embeddings. Therefore, we draw Fig. 4 to analyze which part of information is more crucial for fraud detection. Specifically, we draw the distribution of three gates (i.e., $\boldsymbol{\alpha}^c, \boldsymbol{\alpha}^f, \boldsymbol{\alpha}^t$) based on their average values over feature dimensions.

From the distribution figures, we can infer that the contribution of context-aware embedding is the highest on YelpChi dataset, and the role of feature-aware embedding is slightly more important than that of topology-aware embedding. While on Amazon dataset, the contributions of them are less different, with the topology-aware embedding being the most important component. These results imply that the designed multi-channel module enables our MICA model to be applicable to varied datasets, and be able to interpret the results.

5 Conclusion

In this paper, we aim to design a new solution for enhancing prediction performance in fraud detection tasks. Considering the node representations learned

from GCL methods are hampered by the over-smoothing and false negative samples under various camouflage behaviors, we propose a general contrastive learning model to improve the representational power. From a global camouflage patterns perspective, nodes are transformed into a unified representation space via the generated context and representation refinement module. Then the multi-channel fusion module is considered to mitigate the conflicts of random perturbation with various camouflaged behaviors. Finally, a supervised training loss is adopted for learning a downstream relevant embedding. All of the designed modules greatly improve the distinguishability of the fraudster and benign user in the representation space. The implemented experiments fully verify the effectiveness of the proposed solution in fraud detection. Due to the generality of our proposed framework, we believe that in the future, designing a different context map for each application scenario is a worthy direction to explore.

References

1. Bello, I.: LambdaNetworks: modeling long-range interactions without attention. In: International Conference on Learning Representations (2020)
2. Chen, B., et al.: GCCAD: graph contrastive coding for anomaly detection. arXiv: abs/2108.07516 (2021)
3. Chien, E., Peng, J., Li, P., Milenkovic, O.: Adaptive universal generalized PageRank graph neural network. In: International Conference on Learning Representations (2020)
4. Ding, K., Zhou, Q., Tong, H., Liu, H.: Few-shot network anomaly detection via cross-network meta-learning. In: Proceedings of the Web Conference (2021)
5. Dou, Y., Liu, Z., Sun, L., Deng, Y., Peng, H., Yu, P.S.: Enhancing graph neural network-based fraud detectors against camouflaged fraudsters. In: Proceedings of the 29th ACM International Conference on Information & Knowledge Management (2020)
6. Fey, M., Lenssen, J.E.: Fast graph representation learning with PyTorch Geometric. In: ICLR Workshop on Representation Learning on Graphs and Manifolds (2019)
7. Jiang, Z., et al.: Camouflaged Chinese spam content detection with semi-supervised generative active learning. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 3080–3085 (2020)
8. Khosla, P., et al.: Supervised contrastive learning. In: Advances in Neural Information Processing Systems, vol. 33, pp. 18661–18673 (2020)
9. Kipf, T., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv: abs/1609.02907 (2017)
10. Kreuzer, D., Beaini, D., Hamilton, W.L., L'etourneau, V., Tossou, P.: Rethinking graph transformers with spectral attention. In: Advances in Neural Information Processing Systems, vol. 34 (2021)
11. Li, A., Qin, Z., Liu, R., Yang, Y., Li, D.: Spam review detection with graph convolutional networks. In: Proceedings of the 28th ACM International Conference on Information & Knowledge Management (2019)
12. Liang, C., et al.: Uncovering insurance fraud conspiracy with network learning. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (2019)

13. Liu, Y., et al.: Pick and choose: a GNN-based imbalanced learning approach for fraud detection. In: Proceedings of the Web Conference (2021)
14. Liu, Y., Ao, X., Zhong, Q., Feng, J., Tang, J., He, Q.: Alike and unlike: resolving class imbalance problem in financial credit risk assessment. In: Proceedings of the 29th ACM International Conference on Information & Knowledge Management (2020)
15. Liu, Y., Li, Z., Pan, S., Gong, C., Zhou, C., Karypis, G.: Anomaly detection on attributed networks via contrastive self-supervised learning. IEEE Trans. Neural Netw. Learn. Syst. **33**(6), 2378–2392 (2021)
16. Liu, Z., Dou, Y., Yu, P.S., Deng, Y., Peng, H.: Alleviating the inconsistency problem of applying graph neural network to fraud detection. In: Proceedings of the 43nd International ACM SIGIR Conference on Research and Development in Information Retrieval (2020)
17. Liu, Z., Chen, C., Li, L., Zhou, J., Li, X., Song, L.: GeniePath: graph neural networks with adaptive receptive paths. In: Proceedings of the 33rd AAAI Conference on Artificial Intelligence (2019)
18. Liu, Z., Chen, C., Yang, X., Zhou, J., Li, X., Song, L.: Heterogeneous graph neural networks for malicious account detection. In: Proceedings of the 27th ACM International Conference on Information & Knowledge Management (2018)
19. Ma, X., Wu, J., Xue, S., Yang, J., Sheng, Q.Z., Xiong, H.: A comprehensive survey on graph anomaly detection with deep learning. IEEE Trans. Knowl. Data Eng. **35**(12), 12012–12038 (2021)
20. Pei, H., Wei, B., Chang, K.C.C., Lei, Y., Yang, B.: Geom-GCN: geometric graph convolutional networks. In: International Conference on Learning Representations (2020)
21. Ren, Y., Wang, B., Zhang, J., Chang, Y.: Adversarial active learning based heterogeneous graph neural network for fake news detection. In: 2020 IEEE International Conference on Data Mining (ICDM), pp. 452–461 (2020)
22. Tian, Y., Sun, C., Poole, B., Krishnan, D., Schmid, C., Isola, P.: What makes for good views for contrastive learning. In: Advances in Neural Information Processing Systems, vol. 33, pp. 6827–6839 (2020)
23. Velickovic, P., Cucurull, G., Casanova, A., Romero, A., Lio', P., Bengio, Y.: Graph attention networks. In: International Conference on Learning Representations (2018)
24. Velickovic, P., Fedus, W., Hamilton, W.L., Lio', P., Bengio, Y., Hjelm, R.D.: Deep graph infomax. In: International Conference on Learning Representations (2018)
25. Wang, D., et al.: A semi-supervised graph attentive network for financial fraud detection. In: IEEE International Conference on Data Mining (ICDM), pp. 598–607 (2019)
26. Wang, X., Zhu, M., Bo, D., Cui, P., Shi, C., Pei, J.: AM-GCN: adaptive multi-channel graph convolutional networks. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (2020)
27. Wang, Y., Zhang, J., Guo, S., Yin, H., Li, C., Chen, H.: Decoupling representation learning and classification for GNN-based anomaly detection. In: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (2021)
28. Yang, X., Lyu, Y., Tian, T., Liu, Y., Liu, Y., Zhang, X.: Rumor detection on social media with graph structured adversarial learning. In: Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence, pp. 1417–1423 (2021)

29. Zhang, G., et al.: FRAUDRE: fraud detection dual-resistant to graph inconsistency and imbalance. In: 2021 IEEE International Conference on Data Mining (ICDM), pp. 867–876 (2021)
30. Zhang, Y., Fan, Y., Ye, Y., Zhao, L., Shi, C.: Key player identification in underground forums over attributed heterogeneous information network embedding framework. In: Proceedings of the 28th ACM International Conference on Information & Knowledge Management (CIKM 2019), pp. 549–558 (2019)
31. Zhao, T., Ni, B., Yu, W., Guo, Z., Shah, N., Jiang, M.: Action sequence augmentation for early graph-based anomaly detection. In: Proceedings of the 30th ACM International Conference on Information & Knowledge Management (2021)