ProtoHGC: Heterogeneous Graph Contrastive Learning with Prototype-Regularized Classification for Transaction Fraud Detection

Yuzhen Lin[1]†, Ivonne Xu[2]†, Qi Hu[3]†, Ge Zhang[4]*, Yajin Li[5], Yuan Gao[6], and Han Wu[7]

† Equal contribution.

[1] School of Information Systems and Management, Carnegie Mellon University, Pittsburgh, PA 15213, USA

[2] Department of Physics, University of Chicago, IL 60637, USA

[3] Khoury College of Computer Sciences, Northeastern University, Vancouver, BC V6B 1Z3, Canada

[4] Amazon.com Services LLC, Seattle, WA 98121, USA

zhangge208@yeah.net

[5] School of Art, City University of San Francisco, CA 94118, USA

[6] Department of Computer Sciences, University of Wisconsin-Madison, Madison, WI 53703, USA

[7] College of Liberal Arts, University of Minnesota Twin Cities, Minneapolis, MN 55414, USA

Abstract. Transaction fraud detection on digital payment platforms poses three intertwined challenges: sparse supervision, severe class imbalance, and complex relational dependencies among accounts and transaction events. This paper studies fraud detection on a heterogeneous transaction graph constructed from the public PaySim simulator. To this end, we propose ProtoHGC, a framework that integrates heterogeneous graph contrastive pre-training with prototype-regularized classification. ProtoHGC employs a multi-scale encoder that aggregates first-order neighborhood features and second-order meta-path-based context, coupled with node-level and subgraph-level contrastive objectives to enhance representation quality under limited supervision. Building on the learned embeddings, class-specific prototypes for normal and fraudulent behaviors enforce intra-class compactness and inter-class separation. Experiments on PaySim demonstrate that ProtoHGC consistently outperforms GCN, GAT, GraphSAGE, and MLP baselines across all tested hidden dimensions in terms of AUC-ROC, PR-AUC, accuracy, recall, and F1-score. We explicitly note that the current evaluation is limited to a single synthetic dataset; few-shot adaptation, cross-domain generalization, and multimedia-specific fraud scenarios remain unvalidated.

Keywords: Fraud Detection· Heterogeneous Graph Neural Networks · Graph Contrastive Learning · Prototype-Regularized Classification · Multi-Scale Message Passing

1 Introduction

Digital payment and e-commerce platforms generate massive volumes of transactions, within which fraudulent activities are highly sparse, adaptive, and inherently relational. Rather than appearing as isolated incidents, fraudulent behaviors are often embedded in complex interaction patterns involving payer accounts, payee accounts, transaction events, and temporal dynamics. Therefore, effective fraud detection requires not only the analysis of individual transaction attributes but also the modeling of broader relational context [1]. Traditional approaches based on hand-crafted rules or manually engineered features often struggle to capture these dependencies and adapt to evolving fraud strategies, motivating the development of more expressive and robust relational learning methods.

Graph neural networks (GNNs) have shown strong potential for fraud detection because they naturally model entities and their interactions as graphs. Through message passing, GNNs capture higher-order dependencies and behavioral correlations that are difficult to learn with conventional tabular methods [2]. In transaction fraud detection, this enables the exploitation of both direct transactional evidence and suspicious multi-hop relational patterns. However, many existing methods are based on homogeneous graphs, where all nodes and edges are treated uniformly. This assumption is restrictive for financial transaction data, which typically involve multiple entity types and diverse interaction semantics [3].

To better reflect the structure of transaction ecosystems, heterogeneous graph neural networks (HGNNs) have been introduced to model different node types and relation types explicitly. Nevertheless, most existing HGNN-based fraud detectors rely primarily on supervised learning and therefore require sufficient labeled examples for training [4–6]. In practice, this assumption is often unrealistic, since labeled fraud samples are scarce due to **severe class imbalance** and **the cost of manual expert annotation.**

On the one hand, contrastive learning, as an unsupervised pre-training strategy, can augment the model's discrimination ability in low-resource situations by creating positive sample paired with a negative sample for the purpose of stretching the feature distribution. Specifically, graph contrastive learning enables self-supervised training at the node or subgraph level by leveraging structural augmentations while preserving semantic consistency, improving the model's capacity to discern subtle anomalous patterns without labeled data [7]. However, purely contrastive methods lack explicit class-level structure in the embedding space and offer limited support for downstream classification with few labeled examples.

On the other hand, prototype-based learning offers a complementary mechanism for improving class discrimination and interpretability. By associating each class with representative embeddings in latent space, prototype regularization encourages samples from the same class to form compact clusters while pushing different classes farther apart [8–10]. In fraud detection, such a mechanism is particularly useful for enforcing clearer geometric separation between normal and fraudulent behaviors, which can enhance classification stability under imbalanced and low-label conditions. Despite these advantages, the integration of prototype learning with heterogeneous graph remains

relatively underexplored, especially in transaction fraud detection settings where both relational heterogeneity and supervision scarcity must be addressed simultaneously.

Motivated by these observations, we propose **ProtoHGC** (Prototype-enhanced Heterogeneous Graph Contrastive Network), a unified framework for transaction fraud detection on heterogeneous graphs. Built on a heterogeneous transaction graph constructed from the public PaySim [11] simulator, ProtoHGC combines heterogeneous graph contrastive pre-training with prototype-regularized classification. Specifically, the framework adopts a multi-scale encoder that captures both first-order neighborhood information and second-order meta-path-based context, enabling the model to represent local interactions and broader semantic structures in a unified way. On top of this encoder, node-level and sub-graph-level contrastive objectives are introduced to improve representation quality under limited supervision. Based on the learned embeddings, class-specific prototypes for normal and fraudulent transaction behaviors are further used to promote intra-class compactness and inter-class separability during classification [12].

Extensive experiments on the PaySim [11] dataset show that ProtoHGC consistently outperforms several representative baselines, including GCN, GAT, GraphSAGE, and MLP, across all tested hidden dimensions, demonstrating the effectiveness of combining heterogeneous graph modeling, contrastive pre-training, and prototype-based regularization for fraud detection. At the same time, we emphasize that the current evaluation is limited to a single synthetic dataset. Therefore, the framework's effectiveness for few-shot adaptation, cross-domain generalization, and multimedia-specific fraud scenarios remains to be validated in future work.

2 Related Work

Yussif et al. [13] examined the application and performance of various machine learning algorithms for fraud detection, including support vector machines, random forests, and neural networks. They highlighted the respective advantages and limitations of these methods in different fraud scenarios, offering useful theoretical and practical guidance for the development of anti-fraud models. To cope with the continually evolving forms of fraud in e-commerce and the challenge of detecting unseen fraud types, Zhang et al. [14] proposed a CGNN-based method. Their approach jointly models the representation distributions of normal and fraudulent behaviors, thereby alleviating the dependence on supervision based solely on historical fraud samples.

Furthermore, a regularity of mutual information is included to maximize the separability of normal and fraudulent behavior. Wang et al. [15] proposed a fraud detection framework for multi-relational graphs. Their method addresses challenges such as multiple relation types, class imbalance, and complex graph interactions by introducing an interactive learning strategy. By modeling both the imbalanced structure across relations and the synergistic effects among different relations, the framework improves fraud detection performance. To overcome the lack of neighboring device information, Zhu et al. [16] introduced "super-device" nodes to facilitate the propagation of structural information. In addition, a self-attention module was employed to enhance the

modeling of interactions involving leaf nodes, while a label-smoothing mechanism was adopted to reduce the adverse effect of noisy labels during training.

Dangsawang and Nuchitprasitchai [17] focused on customs fraud detection in supply chains using publicly available unstructured text from social media. They collected approximately 2.37 million commercial product records from social media platforms and categorized them into three classes for subsequent analysis. Adebowale and Akinnagbe [18] employed AI-based ETL pipelines and semantic data models to manage both structured and unstructured data, and further applied machine learning and anomaly detection techniques to identify suspicious transactions and strengthen anti-money-laundering and counter-terrorism financing efforts.

Mutemi and Bacao [19] reviewed the application of machine learning and data mining techniques to fraud detection on e-commerce platforms, providing a comprehensive analysis of common algorithmic trends, dataset characteristics, and future research directions, while also noting the increasing adoption of neural-network-based methods in practice. Mali and Upadhyay [20] applied a random forest approach to fraud detection through social media content mining. By analyzing user behavioral data, their method was able to identify potentially suspicious dynamic activities and provide preliminary risk assessments based on behavioral and content features.

In our framework, the prototype mechanism is introduced as a class-level representation constraint rather than as a dedicated few-shot learning module. It helps the model capture stable class centers for normal and fraudulent behaviors in the heterogeneous transaction graph, which may be advantageous when labeled anomalies are limited. However, since all experiments in this study are conducted under a standard supervised setting on PaySim, the current results only support the effectiveness of prototype-guided representation learning for fraud detection, rather than its applicability to formal few-shot detection scenarios.

3 Methodologies

3.1 Multi-Scale Heterogeneous Graph Encoder with Contrastive Pre-training

We construct a heterogeneous transaction graph $G = (V, E)$ based on the PaySim dataset. Because PaySim consists of transaction records rather than multimedia data, the graph is formed from the entities provided in the dataset, including account nodes, transaction nodes, transaction-type nodes, and discretized time-bucket nodes. This formulation enables the transaction system to be modeled as a heterogeneous graph that captures diverse behavioral and structural information. The corresponding node attributes are then projected into a unified representation space through node initialization, as shown in Equation (1).

$$\mathcal{G} = (\mathcal{V}, \varepsilon, \phi, \psi), \qquad x_v = Fuse\left(x_v^{text}, x_v^{img}, x_v^{vid}, x_v^{tab}\right) = \sum_m w_m f_m, \qquad (1)$$

where $\mathcal{V}$ and ε denotes the sets of nodes and edge, respectively. ϕ and ψ are the node-type and edge-type mapping functions; $x_v \in \mathbb{R}^{d_0}$ represents the initial feature of node v; f_m denotes the encoding vector of modality $m \in$ {text, img, vid, tab}; and w_m is the fusion weight of modality m, computed as $w_m = \mathrm{softmax}(-u_m)$, where $u_m$denotes

the uncertainty estimate of modality m. In this way, modalities with higher uncertainty are assigned lower fusion weights.

To capture strong signals from direct interactions among entities, we employ relation-aware first-order message passing to aggregate local neighborhood information at layer l, as described in Equations (2) and (3).

$$m_v^{(l,1)} = \sum_{r\in R} \sum_{u\in N_r(v)} \alpha_{uv}^{(l,r)} W_r^{(l)} h_u^{(l)}, \tag{2}$$

$$\alpha_{uv}^{(l,r)} = \frac{\exp\left(a_r^{\mathrm{T}}[Q_r h_v^{(l)} || K_r h_u^{(l)}]\right)}{\sum_{u'\in N_r(v)} \exp\left(a_r^{\mathrm{T}}[Q_r h_v^{(l)} || K_r h_u^{(l)}]\right)}, \tag{3}$$

where R is the set of edge types, $N_r(v)$ denotes the set of neighbors of node v under relation r, and $h_u^{(l)}$is the representation of node $u$at the l-th layer. Here, $W_r^{(l)}$, Q_r, K_r, and a_r are learnable parameters. The coefficient $\alpha_{uv}^{(l,r)}$denotes the type-aware attention weight, which assigns greater importance to relation types and neighboring nodes that are more indicative of fraudulent behavior.

To explicitly capture indirect evidence such as collusive behavior, we perform second-order aggregation over selected meta-paths and apply attention to the two-hop structural context to suppress noisy connections, denoted as Equation (4):

$$m_v^{(l,2)} = \sum_{(r_1,r_2)\in M} \sum_{u\in N_{r_1}(v)} \sum_{w\in N_{r_2}(u)} \beta_{vuw}^{(l,r_1,r_2)} W_{r_1,r_2}^{(l)} h_w^{(l)}, \tag{4}$$

where $M \subseteq R \times R$denotes a collection of selected meta-paths, such as account →transaction →account or account →transaction-type →account. The coefficient $\beta_{vuw}^{(l,r_1,r_2)}$ represents the second-order attention weight, which is obtained by applying a softmax function to the corresponding triple-wise relevance score. The matrix $W_{r_1,r_2}^{(l)}$denotes the second-order relational transformation matrix.

To integrate first-order and second-order evidence, we introduce a node-level gating mechanism to fuse multi-scale information, while residual connections and normalization are employed to stabilize deep training, as shown in Equations (5) and (6).

$$\gamma_v^{(l)} = \sigma(w_g^{\mathrm{T}}[m_v^{(l,1)} || m_v^{(l,2)}]), \tag{5}$$

$$h_v^{(l+1)} = LN(h_v^{(l)} + \sigma(W_0^{(l)} h_v^{(l)} + \gamma_v^{(l)} m_v^{(l,1)} + (1-\gamma_v^{(l)}) m_v^{(l,2)})), \tag{6}$$

where $\gamma_v^{(l)} \in (0,1)$denotes the gating coefficient, w_g and $W_0^{(l)}$are learnable parameters, $\sigma(\cdot)$is a nonlinear activation function, and $\mathrm{LN}(\cdot)$denotes layer normalization. The output representation $h_v^{(l+1)}$ is then used as the input to the next layer.

To preserve the semantic structure of fraudulent behaviors during contrastive pre-training, we apply semantic-preserving augmentation at the levels of relations, features, and sampling. Specifically, two augmented graph views are generated, encoded by a shared encoder, and projected into the contrastive space through a projection head, as defined in Equation (7).

$$\tilde{\mathcal{G}}^{(t)} \sim \mathcal{A}(\mathcal{G}), \quad h_v^{(t)} = f_\theta\left(\tilde{\mathcal{G}}^{(t)}\right), \quad z_v^{(t)} = g\left(h_v^{(t)}\right), \tag{7}$$

where $\mathcal{A}$ represents the enhancement distribution (small probability of discarding weak correlations, masking non-key subdimensions, and maintaining degree distribution of isomorphic subgraphs), f_θ is a shared heterograph encoder, $g(\cdot)$ is the projection head, $z_v^{(t)}$ is the node representation of the contrast space.

At the node level, contrastive learning uses structure-aware negative weighting to enlarge the separation between structurally similar but semantically different instances, thereby improving fine-grained fraud discrimination. At the subgraph level, the readout vector of the k-hop ego-subgraph centered at node v (S_v) is used to preserve local structural consistency across the two augmented views, helping to stabilize higher-order relational modeling, as defined in Equations (8) and (9).

$$s_v^{(t)} = READOUT\left(\{h_u^{(t)}: u \in S_v\}\right), \tag{8}$$

$$\mathcal{L}_{sub} = -\frac{1}{|V_b|}\sum_v log\frac{\exp\left(sim(s_v^{(1)}, s_v^{(2)})/\tau_s\right)}{\exp\left(\frac{sim\left(s_v^{(1)}, s_v^{(2)}\right)}{\tau_s}\right) + \sum_{u\neq v}\exp\left(sim(s_v^{(1)}, s_v^{(2)})/\tau_s\right)}, \tag{9}$$

where $\mathrm{READOUT}(\cdot)$ denotes the subgraph readout function, τ_s is the subgraph-level temperature, and $s_v^{(t)}$ is the corresponding subgraph representation.

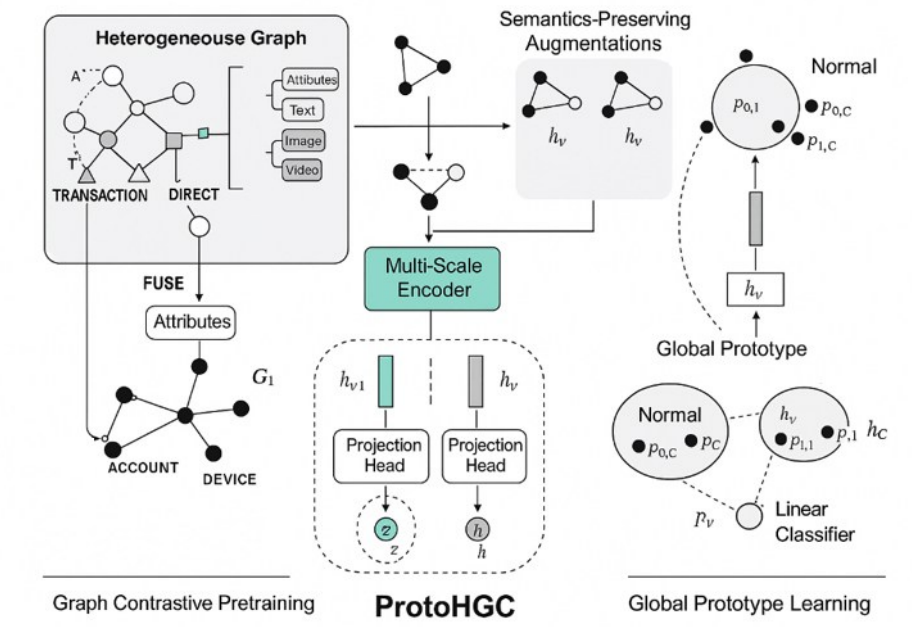


Fig. 1. Heterogeneous Graph Comparison and Prototype Learning Network for Multimedia Platform Fraud Detection.

3.2 Prototype-Regularized Classification

In the representation space learned through contrastive pre-training, we model the global patterns of normal and abnormal behaviors using multiple prototypes, and employ temperature-controlled soft assignment to adapt to long-tailed and multi-cluster anomalies, as defined in Equation (10).

$$q_{v,(c,k)} = \frac{\exp\left(-\left|\left|\hat{h}_v - \hat{p}_{c,k}\right|\right|_2^2/\gamma\right)}{\sum_{c'\in\{0,1\}}\sum_{k'}\exp\left(-\left|\left|\hat{h}_v - \hat{p}_{c',k'}\right|\right|_2^2/\gamma\right)}, \tag{10}$$

where h_v is the encoder output, $\hat{\cdot}$ denotes ℓ_2-normalization, and $c \in \{0,1\}$ represents the normal and abnormal classes, respectively. For each class, C prototypes are defined as $p_{c,k}$, $k = 1, \ldots, C$. The parameter γ is the temperature controlling the soft assignment, with smaller values leading to harder assignments.

The class-conditioned probabilities derived from multiple prototypes provide direct supervision signals. The prototype discriminative loss is formulated using cross-entropy, transforming the geometric intuition that samples should be assigned to their nearest prototypes into a statistical decision criterion, as defined in Equation (11).

$$\mathcal{L}_{proto} = -\frac{1}{|V_l|}\sum_{v\in V_l} log\frac{\sum_k \exp\left(-\left|\left|\hat{h}_v - \hat{p}_{c,k}\right|\right|_2^2/\gamma\right)}{\sum_{c'\in\{0,1\}}\sum_{k'}\exp\left(-\left|\left|\hat{h}_v - \hat{p}_{c',k'}\right|\right|_2^2/\gamma\right)}, \tag{11}$$

where $V_l$denotes the set of labeled nodes, and $y_v$is the class label of node v.

To encourage intra-class samples to cluster more tightly around their corresponding prototypes while explicitly increasing the separation between prototypes of different classes, we introduce compaction and separation regularizers as geometric priors, as defined in Equations (12) and (13).

$$\mathcal{L}_{comp} = \sum_{v\in V_l}\sum_k q_{v,(y_v,k)}\left|\left|\hat{h}_v - \hat{p}_{c,k}\right|\right|_2^2, \tag{12}$$

$$\mathcal{L}_{sep} = \sum_{k,k'}[m - \left|\left|\hat{p}_{0,k} - \hat{p}_{1,k'}\right|\right|_2^2]_+, \tag{13}$$

where L_{comp} denotes the intra-class compactness term, L_{sep} denotes the inter-class separation term. Here, $[\cdot]_+$ represents the ReLU operator, and $m > 0$ is the minimum margin imposed between prototypes of different classes.

Figure 1 illustrates the overall framework and structural composition of the proposed ProtoHGC model. The left part shows the pre-training stage, including heterogeneous graph construction and contrastive learning. Specifically, entities such as accounts, transactions, transaction types, and temporal nodes are organized into a heterogeneous graph with multiple node and edge types. The corresponding node attributes are used to initialize the graph representation. Semantic-preserving graph augmentation is then applied to generate different views, which are fed into the Multi-Scale Encoder to learn node embedding representations.

4 Experiments

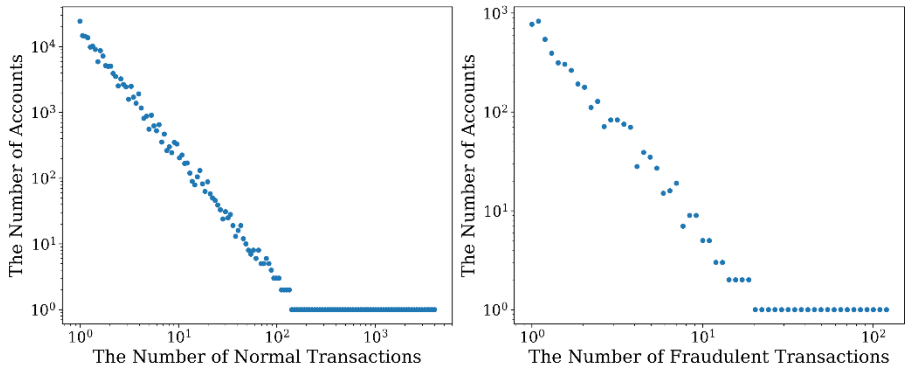
4.1 Experimental Setup

We utilize the publicly available PaySim dataset [11], which is widely used in fraud detection and simulates large-scale financial transaction scenarios based on real mobile payment logs. PaySim contains multiple types of transaction records, including both normal and fraudulent activities, making it suitable for evaluating fraud detection methods. During data preprocessing, entities such as accounts and transactions are modeled as nodes, while different transactional interactions are represented as edges. In addition,

transaction attributes, such as amount and timestamp, are incorporated as node or edge features to construct a heterogeneous graph for downstream learning.

We do not restrict the number of labeled fraudulent samples to a small support set, nor do we compare against few-shot fraud detection baselines under episodic or low-shot settings. As a result, the prototype component should not be interpreted as providing experimentally verified few-shot capability in the current version of the paper. Instead, the reported results demonstrate that prototype-based regularization can improve fraud detection performance in a conventional supervised setting.

Figure 2 illustrates the distribution of account-level transaction frequencies for normal and fraudulent behaviors. The left panel shows the distribution of normal transactions, while the right panel presents the distribution of fraudulent transactions. In both subfigures, the horizontal axis represents the number of transactions associated with an individual account, and the vertical axis denotes the number of accounts with the corresponding transaction frequency.



(a) Normal transactions (b) Fraudulent transactions

Fig. 2. Distribution of Account-Level Transaction Frequencies for Normal and Fraudulent Behaviors.

In the model, ProtoHGC's multiscale encoder has its infrastructure as a two-layer heterography attention network (HAN), and its projection head is a two-layer feedforward neural network, with embedding dimension set to 128. The optimizer used in the model is Adam, with a learning rate of 0.001 and a batch size of 512. In the training process, the model combines the comparative learning loss and the prototype alignment loss for joint optimization, and the hyperparameter search on the validation set was used to determine the loss weight in the model to improve accuracy and robustness in the financial fraud detection tasks. This paper selects five representative comparison models including:

- GCN (Graph Convolutional Network): A classic graph neural network method that aggregates the features of neighboring nodes through convolutional operations, and is commonly used in financial fraud detection to capture local structure information of accounts and transactions.

- GAT (Graph Attention Network): Introduce an attention mechanism based on graph convolution, assigning different weights to different neighbors to improve the discrimination ability of node representation.
- Metapath2Vec: A heterogeneous graph-based embedding method that performs random walks and learns node embeddings by defining meta-paths (e.g., "account-deal-account").
- GraphSAGE: A sampling and aggregation framework that can efficiently process large-scale graph data. In fraud detection tasks, GraphSAGE can improve the modeling capabilities of large-scale transaction graphs while ensuring computational efficiency.
- CARE-GNN (CAmouflage-REsistant Graph Neural Network): A graph neural network model designed for fraud detection, considering that fraudulent nodes may be connected to legitimate nodes through "camouflage".

4.2 Experimental Analysis

1) Detection accuracy measures the proportion of correctly classified samples among all samples and is widely used to evaluate classification performance. However, for highly imbalanced tasks such as fraud detection, accuracy alone can be misleading. In terms of accuracy, Figure 3 shows that ProtoHGC consistently outperforms GCN, GAT, GraphSAGE, and MLP across all tested hidden dimensions. Its accuracy remains above 0.89 throughout and reaches a peak of 0.95. These results demonstrate the effectiveness of ProtoHGC in capturing abnormal patterns within complex heterogeneous graph structures.

GCN and GAT exhibit similar performance at smaller hidden dimensions, but their accuracy tends to plateau as the hidden dimension increases, suggesting limited capacity to exploit more complex relational information. GraphSAGE achieves relatively high accuracy at lower dimensions, yet shows only marginal improvement as the hidden dimension grows. In contrast, MLP performs substantially worse than all graph-based models across all settings, indicating that a model without explicit graph structural modeling is inadequate for this task.

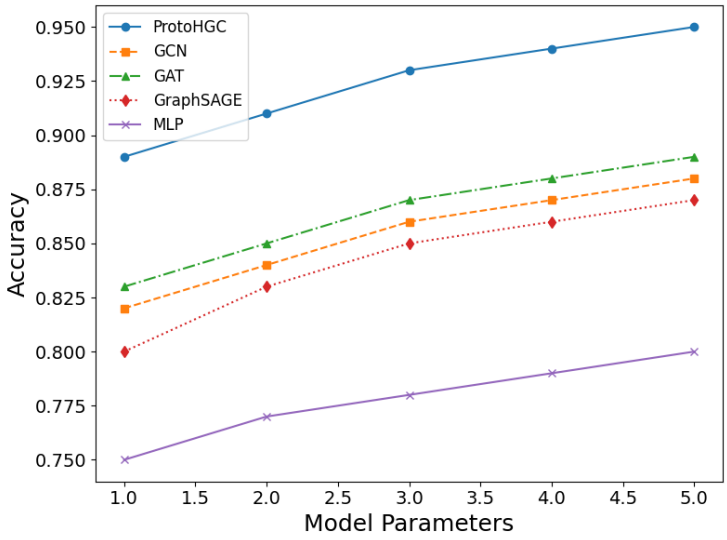


Fig. 3. Accuracy Comparison of ProtoHGC for Fraud Detection on Heterogeneous Graphs.

2) Recall measures the proportion of truly fraudulent transactions that are correctly identified by the model. In the recall evaluation, we compared five representative methods, namely ProtoHGC, GCN, GAT, GraphSAGE, and MLP, under different hidden dimensions (64, 128, 256, 512, and 1024). As shown in Figure 4, ProtoHGC consistently outperforms the other models across all parameter settings, achieving recall values ranging from 0.84 to 0.94. These results indicate that ProtoHGC is more effective at identifying potential fraudulent samples, particularly in settings where fraudulent transactions are sparse or difficult to distinguish from normal behavior.

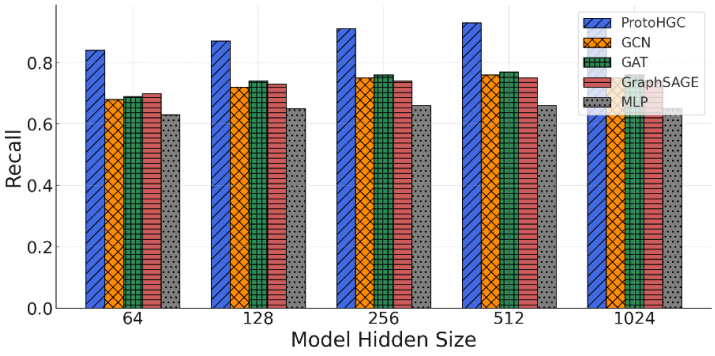


Fig. 4. Recall Performance Comparison of Fraud Detection Models.

In contrast, GCN and GAT exhibit only slight performance gains and plateau at approximately 0.76–0.77 recall at the highest hidden dimension, indicating limited capacity to model complex graph structures and long-range dependencies. GraphSAGE attains relatively competitive performance at medium dimensions, but its recall improves

only marginally at larger dimensions. MLP remains the weakest baseline across all settings, further demonstrating that models lacking explicit graph structural information are inadequate for capturing the complexity of fraud detection data.

3) F1-score is the harmonic mean of precision and recall, making it particularly suitable for imbalanced classification tasks. It provides a more comprehensive assessment of the model's ability to identify anomalous classes.

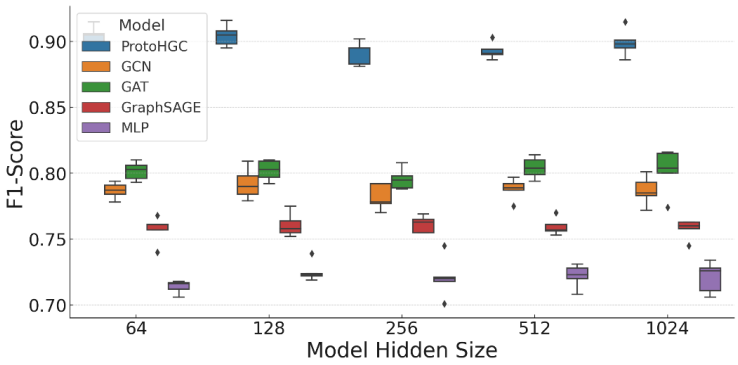


Fig. 5. F1-Score Comparison Across Models with Varying Hidden Sizes.

Figure 5 compares the F1-scores of different models under five hidden dimensions (64, 128, 256, 512, and 1024). The results show that ProtoHGC consistently achieves the best performance, maintaining a stable F1-score in the range of 0.89–0.91 across all parameter settings. This demonstrates the robustness and consistency of the proposed model under limited-supervision conditions. In contrast, GAT and GCN also perform relatively well, but their F1-scores remain at a lower level, approximately between 0.78 and 0.81 across different settings.

Although GAT and GCN achieve comparable performance under some parameter settings, both remain consistently below ProtoHGC on average. GraphSAGE shows noticeable variation across different hidden dimensions and even performs relatively poorly in certain cases. This may be attributed to limitations in its neighbor aggregation mechanism when modeling complex behavioral relationships. MLP yields the lowest F1-scores overall, ranging from 0.70 to 0.74. Its inferior performance further suggests that models without explicit graph structural modeling are not well suited to fraud detection on complex heterogeneous graphs.

4) AUROC evaluates the ability of a binary classifier to distinguish positive samples from negative ones over all possible decision thresholds. In fraud detection, it provides a threshold-independent assessment of how well a model separates fraudulent transactions from normal transactions. Table 1 compares the AUROC performance of ProtoHGC, GCN, GAT, GraphSAGE, and MLP under different hidden dimensions. As shown in the table, ProtoHGC consistently obtains the highest AUROC values across all settings, demonstrating superior discriminative capability. By contrast, GCN and GAT perform slightly worse, while MLP remains the weakest baseline. These results

further confirm that incorporating graph structural information is critical for effective fraud detection.

Table 1. AUC-ROC Performance Comparison Across Models.

Hidden Size	ProtoHGC	GCN	GAT	GraphSAGE	MLP
64	0.9431	0.8865	0.8726	0.8361	0.769
128	0.9486	0.8740	0.8655	0.8556	0.7854
256	0.9452	0.8828	0.8638	0.8368	0.7666
512	0.944	0.8892	0.8752	0.8538	0.7649
1024	0.9589	0.8967	0.877	0.8431	0.7761

Table 2. PR-AUC Performance Comparison on Imbalanced Data.

Hidden Size	ProtoHGC	GCN	GAT	GraphSAGE	MLP
64	0.9108	0.8531	0.8399	0.8186	0.738
128	0.9259	0.8554	0.8536	0.8148	0.7324
256	0.9120	0.8429	0.8479	0.8032	0.7405
512	0.9161	0.8545	0.8416	0.8174	0.7510
1024	0.9250	0.8502	0.8428	0.8233	0.7586

5) PR-AUC is a particularly important metric in fraud detection tasks with severe class imbalance. Table 2 presents the PR-AUC results. Because fraudulent transactions account for only a small fraction of the data, PR-AUC provides a more informative evaluation of how effectively a model balances precision and recall for the fraud class. The results show that ProtoHGC consistently outperforms the baseline methods on this metric, highlighting its strong ability to detect fraudulent transactions under sparse-label and imbalanced conditions.

By comparison, traditional models such as GraphSAGE and MLP achieve markedly lower PR-AUC values, suggesting limited ability to identify minority-class fraud patterns. The superior performance of ProtoHGC indicates that prototype-based regularization together with contrastive pre-training can more effectively enhance anomaly discrimination and improve fraud detection performance.

4.3 Ablation Study

1) Embedding Visualization Before and After Knowledge Enhancement. Figure 6 illustrates the distribution of learned embeddings before and after the incorporation of generative multi-domain knowledge. In Figure 6(a), the original embedding space exhibits only partial separation between abnormal and normal samples, and the target sample remains difficult to distinguish from nearby instances. After enhancement, as shown in Figure 6(b), abnormal samples become more compactly clustered and more clearly separated in the embedding space, while normal samples retain a relatively well-structured distribution.

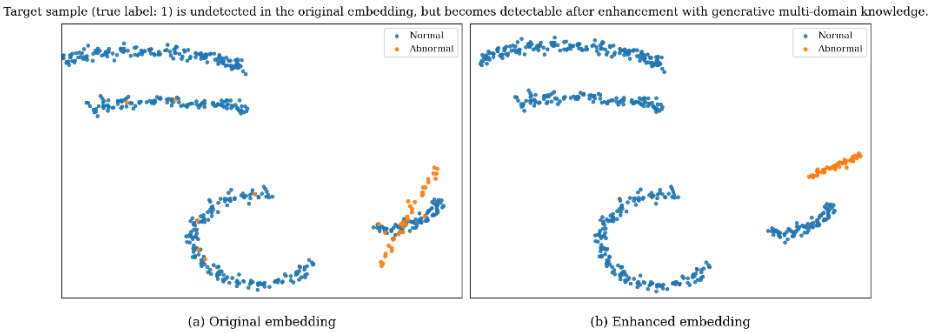


Fig. 6. Visualization of Embedding Enhancement for Abnormal Transaction Detection.

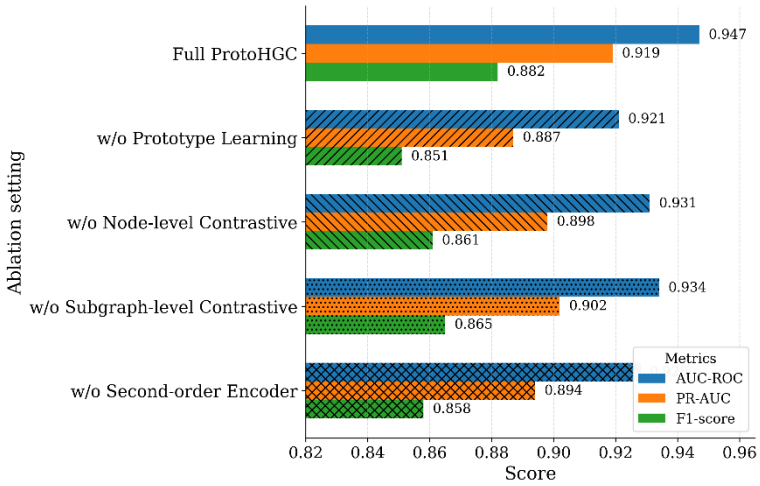


Fig. 7. Ablation Study of ProtoHGC for Transaction Fraud Detection.

2) Ablation Study of ProtoHGC Components. Figure 7 presents the ablation study results of ProtoHGC under different component removal settings, including the exclusion of prototype learning, node-level contrastive learning, subgraph-level contrastive learning, and the second-order encoder. The performance is evaluated using three metrics, namely AUC-ROC, PR-AUC, and F1-score. It can be observed that the full ProtoHGC model achieves the best performance across all three metrics, indicating the effectiveness of the complete framework for transaction fraud detection.

5 Conclusion

In conclusion, we propose ProtoHGC, a heterogeneous graph neural network framework that integrates graph contrastive learning and prototype-based regularization for fraud detection. By constructing a heterogeneous transaction graph that captures relationships among entities such as accounts and transactions, ProtoHGC jointly models

structural dependencies and attribute information. The framework further enhances representation learning through contrastive pre-training under limited supervision, while the prototype module promotes intra-class compactness and inter-class separability between normal and fraudulent behaviors. In addition, the multi-scale encoder effectively combines first- and second-order relational information to capture fraud patterns at different structural levels. Future work will focus on improving the model's cross-domain generalization, interpretability, and practical applicability, as well as exploring incremental learning and human-in-the-loop mechanisms for more robust fraud detection.

References

1. Jiang, Nan, et al. "MAFI: GNN-based multiple aggregators and feature interactions network for fraud detection over heterogeneous graph." IEEE Transactions on Big Data 8.4 (2021): 905-919.
2. Liu, Yajing, Zhengya Sun, and Wensheng Zhang. "Improving fraud detection via hierarchical attention-based graph neural network." Journal of Information Security and Applications 72 (2023): 103399.
3. Zhou, Hangjun, et al. "Internet financial fraud detection based on a distributed big data approach with node2vec." Ieee Access 9 (2021): 43378-43386.
4. Bonett, Stephen, et al. "Assessing and improving data integrity in web-based surveys: Comparison of fraud detection systems in a COVID-19 study." JMIR formative research 8 (2024): e47091.
5. El-Shafai, Walid, et al. "A comprehensive taxonomy on multimedia video forgery detection techniques: challenges and novel trends." Multimedia Tools and Applications 83.2 (2024): 4241-4307.
6. Odufisan, Oluwaseun Isaac, Osekhonmen Victory Abhulimen, and Erastus Olarenwaju Ogunti. "Harnessing Artificial Intelligence and Machine Learning for Fraud Detection and Prevention in Nigeria." Journal of Economic Criminology (2025): 100127.
7. Lyu, Qiuyun, et al. "BCFDPS: A Blockchain-Based Click Fraud Detection and Prevention Scheme for Online Advertising." Security and Communication Networks 2022.1 (2022): 3043489.
8. Sharma, Dilip Kumar, et al. "Sarcasm detection over social media platforms using hybrid auto-encoder-based model." Electronics 11.18 (2022): 2844.
9. Rawat, Romil, et al. "Applications of social network analysis to managing the investigation of suspicious activities in social media platforms." Advances in Cybersecurity Management. Cham: Springer International Publishing, 2021. 315-335.
10. Sharma, Dilip Kumar, et al. "A survey of detection and mitigation for fake images on social media platforms." Applied Sciences 13.19 (2023): 10980.
11. Lopez-Rojas E, Elmir A, Axelsson S. PaySim: A financial mobile money simulator for fraud detection. 28th European modeling and simulation symposium, EMSS, Larnaca. Dime University of Genoa, 2016: 249-255.
12. Goyal, Bharti, et al. "Detection of fake accounts on social media using multimodal data with deep learning." IEEE Transactions on Computational Social Systems (2023).
13. Yussiff, Alimatu-Saadia, et al. "The best machine learning model for fraud detection on e-platforms: a systematic literature review." Computer Science and Information Technologies 5.2 (2024): 195-204.

14. Zhang, Ge, et al. "efraudcom: An e-commerce fraud detection system via competitive graph neural networks." ACM Transactions on Information Systems (TOIS) 40.3 (2022): 1-29.
15. Wang, Xiaodi, et al. "Fraud detection on multi-relation graphs via imbalanced and interactive learning." Information Sciences 642 (2023): 119153.
16. Zhu, Yadong, et al. "Botspot++: A hierarchical deep ensemble model for bots install fraud detection in mobile advertising." ACM Transactions on Information Systems (TOIS) 40.3 (2021): 1-28.
17. Dangsawang, Bundidth, and Siranee Nuchitprasitchai. "A machine learning approach for detecting customs fraud through unstructured data analysis in social media." Decision Analytics Journal 10 (2024): 100408.
18. Adebowale, Adelakun Matthew, and Olayiwola Blessing Akinnagbe. "Cross-platform financial data unification to strengthen compliance, fraud detection and risk controls." World J Adv Res Rev 20.3 (2023): 2326-2343.
19. Mutemi, Abed, and Fernando Bacao. "E-commerce fraud detection based on machine learning techniques: Systematic literature review." Big Data Mining and Analytics 7.2 (2024): 419-444.
20. Mali, Yogesh, and Tejal Upadhyay. "Fraud detection in online content mining relies on the random forest algorithm." SciWaveBulletin 1.3 (2023): 13-20.