Parallel Graph Learning with Temporal Stamp Encoding for Fraudulent Transactions Detections

Jiacheng Ma, Sheng Xiang, Qiang Li, Liangyu Yuan, Dawei Cheng, Changjun Jiang

***Abstract*—Financial transaction systems have become the critical backbone of modern society, and the sharp increase in fraudulent transactions has become an unavoidable significant topic. Their presence poses a severe threat to financial markets, impacting the health of the economic and social welfare systems of various countries. However, most existing fraud detection methods are limited to detecting individual fraudulent entities within static transaction networks, which are neither suitable for continuously changing dynamic transaction networks nor capable of detecting the increasingly prevalent organized fraud crimes. This paper introduces a novel approach, Parallel Graph Learning with Temporal Stamp Encoding (PGLTSE). On the one hand, it designs a history information module to perform temporal dimension feature learning to adapt to the continuous changes in transaction information in Continuous-Time Dynamic Graphs (CTDG). On the other hand, it designs a gang-aware risk propagation algorithm to infer the risk of organized fraudulent activities in the global transaction relation graph. By simultaneously conducting parallel graph representation learning in both homogeneous global transaction relation graphs and heterogeneous local entity interaction graphs, it aggregates local interaction and global association information for end-to-end training. Extensive experiments on diverse real-world datasets substantiate the superior performance of PGLTSE over existing methods, demonstrating its practical efficacy in detecting complex and evolving fraudulent behaviors in financial networks.**

***Index Terms*—Fraud Detection, Continuos Time Dynamic Graphs, Graph Neural Network, Risk diffusion.**

I. Introduction

THROUGHOUT the history of financial services, the battle against fraud, and the efforts to prevent it have been an ongoing tug-of-war [1], [2]. From individual opportunists to organized crime groups, fraudsters have continually sought to exploit loopholes in financial transaction systems for illicit gain [3], [4]. As the digital revolution transformed the financial market, these malicious entities employed increasingly sophisticated methods of fraudulent transactions, posing significant challenges to market regulation and the financial security of transaction parties, and seriously challenging the integrity and safety of the market environment [5], [6].

The digitization of finance, particularly the shift from traditional platforms to online platforms, has significantly altered the landscape of financial transactions. Before the advent of e-commerce and the internet, fraud detection relied on labor-intensive manual monitoring and verification of transactions [7], [8]. For example, bank clerks would meticulously inspect checks for signs of forgery, a process fraught with inefficiencies and vulnerabilities, ill-equipped to handle the cunning of fraudsters. However, the advent of digital technology brought about a paradigm shift. By the end of the 20th century, although electronic monitoring tools were introduced, they were initially primitive and struggled to adapt to the evolving strategies of fraudsters [9]. Fraudulent actors commenced leveraging techniques such as phishing, malware injections, and Trojan attacks to conduct their schemes.

In the early 21st century, the field of fraud detection and prevention began to utilize more advanced technologies [10], [11]. Researchers developed network visualization tools to enhance the detection of suspicious online activities, completely changing the methods of monitoring and preventing fraudulent activities. By 2012, a new era of anti-fraud had arrived, enabling the construction of individual risk profiles from a broad range of data sources, predicting criminal tactics likely to be used by fraudsters through rigorous verification of real user identities and comprehensive scrutiny of user data, including age verification and transaction history. This breakthrough improved transaction monitoring, screening methods, and tracking capabilities, significantly reducing false positives. These technologies allowed for analyzing vast amounts of transaction data, using predictive analytics and machine learning to anticipate risks and eliminate threats [12]–[14]. Modern enterprises and financial institutions can not only detect fraud more accurately but also predict and prevent it beforehand.

Despite these advancements, fraudulent methods continue to innovate, with criminals adopting more complex approaches, such as device emulation, social engineering, and sophisticated identity fraud strategies to counter past anti-fraud technologies, increasingly operating in gangs involving money laundering, insurance fraud, loan fraud, and electronic currency fraud among various transaction types. These illegal activities not only harm consumer interests but also undermine the fairness and stability of financial markets. Thus, effectively identifying and preventing fraudulent transactions has become a key research topic in the fintech sector.

In recent years, machine learning technology has made significant advances, clearly outperforming traditional statis-

This work was supported by the National Key R&D Program of China (Grant no. 2022YFB4501704), the National Natural Science Foundation of China (Grant no. 62102287), and the Shanghai Science and Technology Innovation Action Plan Project (Grant no. 22YS1400600 and 22511100700).

Jiacheng Ma, Qiang Li, Liangyu Yuan, Dawei Cheng and Changjun Jiang are with the Department of Computer Science and Technology, Tongji University, Shanghai, China. Dawei Cheng and Changjun Jiang are also with the Shanghai Artificial Intelligence Laboratory, Shanghai, China (email: jcma@tongji.edu.cn; dcheng@tongji.edu.cn).

Sheng Xiang is with the Australian Artificial Intelligence Institute, University of Technology Sydney, Sydney, Australia (email: sheng.xiang@uts.edu.au).

(Corresponding author: Sheng Xiang.)

Manuscript received June 9, 2024; revised Sept. 2, 2024.

tical methods in terms of efficiency, cost-effectiveness, and accuracy [15], [16]. Initially, machine learning methods such as Random Forests and Gradient Boosting Decision Trees provided valuable insights, paving the way for innovation. Later, the explosion of deep learning brought a new paradigm to the technological frontier [17], [18], especially the use of Graph Neural Networks (GNNs) [19], [20], which have shown particular effectiveness in detecting complex patterns and anomalies in financial networks [21]–[23]. Traditional methods often regard order nodes in transaction networks as isolated data entities. In contrast, GNNs construct and learn from transaction relationship graphs, which allows them to capture the topological information within these networks. This capability enables GNNs to extract behavioral patterns of fraudsters and effectively identify and mitigate fraudulent transactions [24].

However, previous GNN methods have several limitations. They typically regard large-scale transaction data as static transaction networks, with the mining of temporal historical data relying on manually designed feature engineering (e.g., statistics within a certain time window) and lacking in mining information on the temporal dimension. Even attempts to learn temporal dimension features in transaction data are often made by converting dynamic graphs into static graphs through temporal snapshots, implicitly modeling the learning objects of graphs as Discrete Time Dynamic Graphs (DTDG), but still insufficient to handle more practical Continuous Time Dynamic Graphs (CTDG) data types. Additionally, existing graph-based fraud detection methods mostly learn only one type of relationship from entity graphs, which inevitably leads to suboptimal detection performance. Moreover, current GNN-based transaction anti-fraud methods typically only detect single fraudulent transactions represented as anomalous nodes in graphs, while the community structure—holding information related to criminal gangs—of users and merchants is largely ignored. Community structure retains the inherent high-order structural properties of graphs, for example, communities in real networks may represent real social groups, while communities in guaranteed loan networks may indicate dense loan relationships and financial institution groups. Understanding this community-level risk association information can better discover fraudulent criminal gangs.

Therefore, we propose a novel method named Parallel Graph Learning with Temporal Stamp Encoding for Fraudulent Transactions Detection (PGLTSE) with key contributions:

- To facilitate the model's capacity to adapt to the evolving dynamics of graphs, a history module has been designed to store historical information of nodes within the transaction network. The model is better adapted to evolving fraud techniques by capturing hidden temporal correlation patterns between fraudulent transactions and identifying different fraudulent behaviour patterns and trends.
- We employ parallel graph learning, namely learning simultaneously from the Homogeneous Global Transaction Relation Graph and the Heterogeneous Local Entity Interaction Graph through GNNs, to obtain better node hidden embeddings to detect rapidly changing fraud behavior patterns.
- A gang-aware risk propagation algorithm has been devised to infer the upstream and downstream transactional background of organized fraud behaviors, as part of the learning process for the global transaction relation graph. This allows us to identify group characteristics within gang fraud transactions and to discover collusive crimes and fraud gangs using graph attention layers.
- We use various real transaction data to train and validate the model, ensuring its robustness and applicability in various practical financial transaction business scenarios.

Roadmap. The rest of this paper is organized as follows: Section II defines the problem of fraud transaction detection in financial dynamic transaction networks and reviews related work. Section III details our proposed method and optimization objectives. Section IV describes how we model temporal dimension features in data to learn historical transaction information. Section V elaborates on our experimental results and case studies. Section VI concludes and discusses our research.

II. Preliminaries

A. Background and Related Works

Fraudulent transactions, characterized by deception and misleading information for illegal gains, are prevalent across various sectors including finance, e-commerce, and insurance. In the financial market, fraud manifests in diverse forms such as credit card fraud, insurance fraud, and securities fraud. For instance, fraudsters may use stolen credit card information for unauthorized purchases or fabricate accidents or illnesses to claim insurance payouts. They may also mislead investors into buying subpar stocks, thereby swindling them. Given the gravity of this issue, significant research has been directed towards anti-fraud measures in financial transactions.

1) Fraudulent Transaction Detection: Fraudulent transaction detection has been extensively studied using various machine learning and statistical techniques. Traditional methods include supervised learning approaches such as logistic regression [25], [26], decision trees [27], [28], and support vector machines [29], [30], which rely on hand-crafted features extracted from input data [31]. Recently, deep learning techniques have been employed to automatically learn complex features from raw fraudulent transaction data, improving the accuracy of fraud detection systems [32], [33]. However, these deep-learning approaches typically focus on individual entities and do not fully exploit the relationships between entities (such as users and transactions). Recently, graph-based methods have gained popularity in fraudulent transaction detection due to their ability to capture relationships between entities. These methods represent input data as graphs, where nodes represent entities (e.g., users, transactions) and edges represent relationships (e.g., transaction links). Techniques such as graph convolutional networks (GCNs) and graph attention networks (GATs) have been applied to fraud detection, showing improved performance over traditional methods [34]. CARE-GNN [35] was proposed to select better neighbors for graph-based fraud detection tasks. PC-GNN [36] was designed to solve the node label imbalance problem in fraud detection. BW-GNN [37] was proposed to address the "right-shift" issue

in the graph anomaly detection. However, these approaches often assume a static graph structure and may not fully exploit the dynamics inherent in real-world financial transaction networks. Some methods, such as GTAN [20] and DGA-GNN [38], were designed for semi-supervised fraud detection in temporal graphs. While the usage of temporal graph has been explored in existing literature, our approach combines parallel graph learning with temporal stamp encoding, offering a temporal graph learning framework.

2) Temporal Graph Learning: Despite the advancements in fraud detection, most methods were confined to static graph data [5], [34], [37], [39], with few extending to Discrete-Time Dynamic Graphs (DTDG) using snapshots, lacking consideration of temporal variables and falling short of handling continuous-time dynamic graphs. Continuous-Time Dynamic Network Embeddings (CTDNE) [40] addressed this by capturing dynamic changes in graph structures over time. Based on Node2Vec, CTDNE incorporates temporal variables into embedding vectors, enhancing fraud detection by representing evolving relationships within fraud networks. However, the increased complexity of dynamic modeling in CTDNE can result in higher computational costs, necessitating careful optimization and parameter tuning. CTDNE's focus on temporal dynamics marked a promising advancement for detecting evolving fraudulent activities. Building on CTDNE's principles, Temporal Graph Networks (TGN) [41] significantly advanced fraud detection by modeling temporal dependencies. Concurrently, Temporal Graph Attention Networks (TGAT) [42], inspired by both TGN and CTDNE, capture the dynamic evolution of graph structures over time. By incorporating attention mechanisms, TGAT enhances analytical capabilities and optimizes performance, prioritizing temporal dynamics and demonstrating exceptional proficiency in detecting dynamically evolving fraudulent activities. Incorporating graph models such as GCNs, GAT, and TGAT into fraud detection systems has greatly improved accuracy. These models, building on each other's strengths, have revolutionized understanding and combating fraudulent activities within complex networks. The introduction of temporal dynamics modeling has been particularly transformative, despite the increased computational demands. Effective resource allocation, optimization strategies, and distributed computing solutions are crucial to addressing these computational challenges.

In summary, integrating advanced graph models and temporal dynamics modeling into anti-fraud efforts constitutes a significant leap forward. Identifying evolving fraud patterns within complex networks is crucial for maintaining a proactive stance against deceptive practices. In our work, we designed a history information module to perform temporal dimension feature learning to adapt to the continuous changes in transaction information in dynamic graphs.

B. Problem Statements

In the realm of daily economic operations, a transaction usually unfolds between two distinct groups of entities: (a) *the merchants* providing services and (b) *the consumers* purchasing them. This interactive network can be conceptualized as a bipartite graph. Moreover, when a transaction implicates the same merchant or consumer, it can be perceived as interconnected, thereby modeled into a comprehensive transaction homogeneity graph.

Temporal Graph. Extract timestamps and construct a temporal graph $G = \{x(t_1), x(t_2), \ldots\}$, where $x(t_i)$ represents a change occurring in the graph at time t_i (adding or removing nodes, edges, and feature modifications, simplified to edge additions and removals in this study). Continuous events are used to generate node representations for each time t, denoted as $\mathbf{Z}(t) = (\mathbf{z}_1(t), \ldots, \mathbf{z}_n(t))$.

In this study, we define the consumer-merchant relation graph as $\mathcal{G}(\mathcal{C}, \mathcal{M}, \mathcal{E})$. Here, $\mathcal{C} = \{v_1^c, \ldots, v_{N_C}^c\}$ represents the set of consumer nodes, $\mathcal{M} = \{v_1^m, \ldots, v_{N_M}^m\}$ denotes the set of merchant nodes, and $\mathcal{E} = \{e_1^{t_1}, \ldots, e_{N_E}^{t_2}\}$ signifies different edge increase/decrease events within the graph, which correspond to the financial transactions between consumers and merchants at different moments like t_1 and t_2. The quantities of consumers, merchants, and transactions are indicated as N_C, N_M, and N_E, respectively. Concerning the neighbors within the graph, let $\mathcal{N}_v$ represent the set of nodes in node v's one-hop neighbors, such that $\mathcal{N}_{v_i^c \in \mathcal{C}} \subseteq \mathcal{M}$ and $\mathcal{N}_{v_j^m \in \mathcal{M}} \subseteq \mathcal{C}$. Each consumer node v_i^c is characterized by a d^C-dimensional feature vector $\boldsymbol{h}_{\boldsymbol{c,i}}^{\mathbf{0}} \in \mathbb{R}^{d^C}$, while each merchant node v_j^m is described by a d^M-dimensional vector $\boldsymbol{h}_{\boldsymbol{m,j}}^{\mathbf{0}} \in \mathbb{R}^{d^M}$. For the edge increase/decrease event e_k^t at time t, we define $\boldsymbol{h}_{\boldsymbol{k,t}}^{\mathbf{0}} \in \mathbb{R}^{d^E}$ as its attribute vector and let $\mathcal{Y}_e = \{0, 1\}^{N_E}$ be the set of labels for whether a transaction order is fraudulent or not, with 0 indicating normal and 1 indicating fraud.

Additionally, we construct the homogeneous transaction relation graph $\mathcal{G}_{tr}(\mathcal{V}_{tr}, \mathcal{E}_{tr})$, where each node represents a transaction. An edge e_{ij} is established between two transactions $v_i, v_j \in \mathcal{V}_{tr}$ if both transactions involve the same consumer or merchant. For each transaction, our objective is to determine the likelihood of fraud, framing our task as an edge classification problem in the consumer-merchant graph and a node classification problem in the transaction relation graph. This paper explores two detection problems: individual-level fraud transaction detection and gang-level fraud transaction detection. Essentially, this involves pattern discovery and classification within graphs, necessitating the identification of fraudulent entities in transactions.

C. Temporal Features

In real-life transactions, data often includes temporal information, leading to the abstraction of graph structures and properties that are not static but evolve over time. This corresponds to operations such as edge deletion and node attribute updates in the graph, thus introducing the concept of dynamic graph embedding.

Many fraudulent activities exhibit temporal dependencies, where a previous transaction may influence subsequent ones. At the same time, fraudsters continuously refine their strategies and tactics to evade detection. By establishing dynamic graphs considering temporal dimension features through dynamic graph embedding, we can capture these temporal dependencies, better identify potential fraud patterns, and allow us to

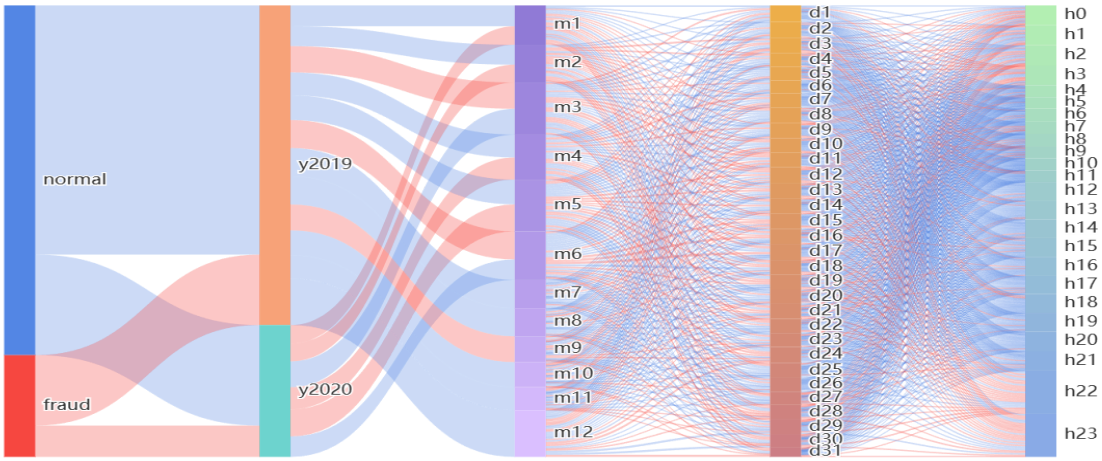


Fig. 1. A Sankey diagram has been constructed to depict the relationship between the label of whether a transaction is fraudulent and the timestamp on the Sparkov dataset [43]. On the left side, the labels indicate that the transactions have been classified as normal and fraudulent. Subsequently, these transactions flow into different time intervals, including years, months, days, and hours. The thickness of the flow paths represents the volume of transactions in specific time intervals.

continuously adjust our model to adapt to these new fraudulent behaviors. In other words, we need to find graph representation learning methods that can learn temporal dimension features on Continuous Time Dynamic Graphs (CTDGs).

Virtually most transaction datasets contain timestamps and can be modeled as CTDGs. Take the Sparkov dataset [43] as an example; it includes 120,000 credit card transactions from 1,000 customers and 800 merchants over six months, with a fraud transaction ratio of 10.6%. After downsampling the data, we observe significant differences in the temporal distribution of normal transactions and fraudulent transactions in the sankey diagram of Figure 1. For instance, we can distinctly observe that the distribution of fraud between 21:00 and 5:00 is different from other time periods, therefore, label encoding was performed based on whether the transaction occurred within this time frame. Fraudulent transactions increase during specific months, dates, and hours, possibly indicating that fraudsters are more active during these periods. This pattern may be due to fraudsters taking advantage of lax system monitoring during these times or consumers having weaker awareness of fraud prevention. Additionally, we study the temporal dimension features in the dataset using a heatmap and a violin plot(Figure 4), further demonstrating the necessity of feature learning from the perspective of temporal dimension features.

III. Methodology

We delve into the specifics of our proposed methodology for detecting fraudulent activities in financial transactions in this section. The model architecture of the PGLTSE is shown in Figure 2. We begin by outlining the learning approach within the heterogeneous entity bipartite graph, emphasizing the memory module's role in capturing time dimension features. Next, we detail the construction of the transaction relation graph and explain the representation learning process. Finally, we describe the structure of the detection network and the optimization strategy that underpins our proposed methods.

A. Local Entity Interaction Graph Learning

In the domain of everyday economic operations, interactions transpire between two distinct groups: merchants providing services and consumers procuring them. This interplay forms a bipartite graph that is instrumental in extracting concealed insights from the intricate network topology that interlinks diverse transaction entities, thereby augmenting the embedding of trade orders. Given that nodes representing both consumers and merchants undergo analogous phases of information aggregation, our attention is primarily directed toward consumer entities in this instance to clarify operational procedures. This strategy ensures the preservation of original data integrity and facilitates a deeper examination of transaction dynamics, offering a comprehensive insight into the financial ecosystem.

Initially, we implement the graph attention mechanism to determine the relevance of features of the locally adjacent merchant node v_j^m to the consumer node v_i^c, where v_j^m and v_k^m are members of the neighborhood $\mathcal{N}_{v_i^c}$. For the hidden states of nodes at the l-th layer, the attention coefficients are formulated as:

$$\alpha_{ij}^l = \frac{\exp\left(\sigma(\mathbf{u}^T\left[\mathbf{W_c}\boldsymbol{h}_{\boldsymbol{c,i}}^{\boldsymbol{l-1}} \| \mathbf{W_m}\boldsymbol{h}_{\boldsymbol{m,j}}^{\boldsymbol{l-1}}\right])\right)}{\sum_{v_k^m \in \mathcal{N}_{v_i^c}} \exp\left(\sigma(\mathbf{u}^T\left[\mathbf{W_c}\boldsymbol{h}_{\boldsymbol{c,i}}^{\boldsymbol{l-1}} \| \mathbf{W_m}\boldsymbol{h}_{\boldsymbol{m,k}}^{\boldsymbol{l-1}}\right])\right)}, \quad (1)$$

where $\mathbf{W_c} \in \mathbb{R}^{d^C \times d^C}$ and $\mathbf{W_m} \in \mathbb{R}^{d^M \times d^M}$ are the weight matrices for consumer and merchant entities, respectively, and $\boldsymbol{u} \in \mathbb{R}^{d^C + d^M}$ is the weight vector. We choose σ as the $LeakyReLU$ activation function and use $\|$ to denote the concatenation operation. This setup facilitates the generation of message representations from neighboring nodes and edges. Assuming that the edges (transactions) between the two nodes contain equivalent information, their combined messages are weighted by node attention coefficients. The set of edge change events connecting consumer v_i^c and merchant v_j^m is denoted by $\mathcal{N}_{ij}(e^t)$, with $N_{ij}^{e^t}$ representing the number of edge change events at time t. The message construction phase for the neighborhood is formulated as:

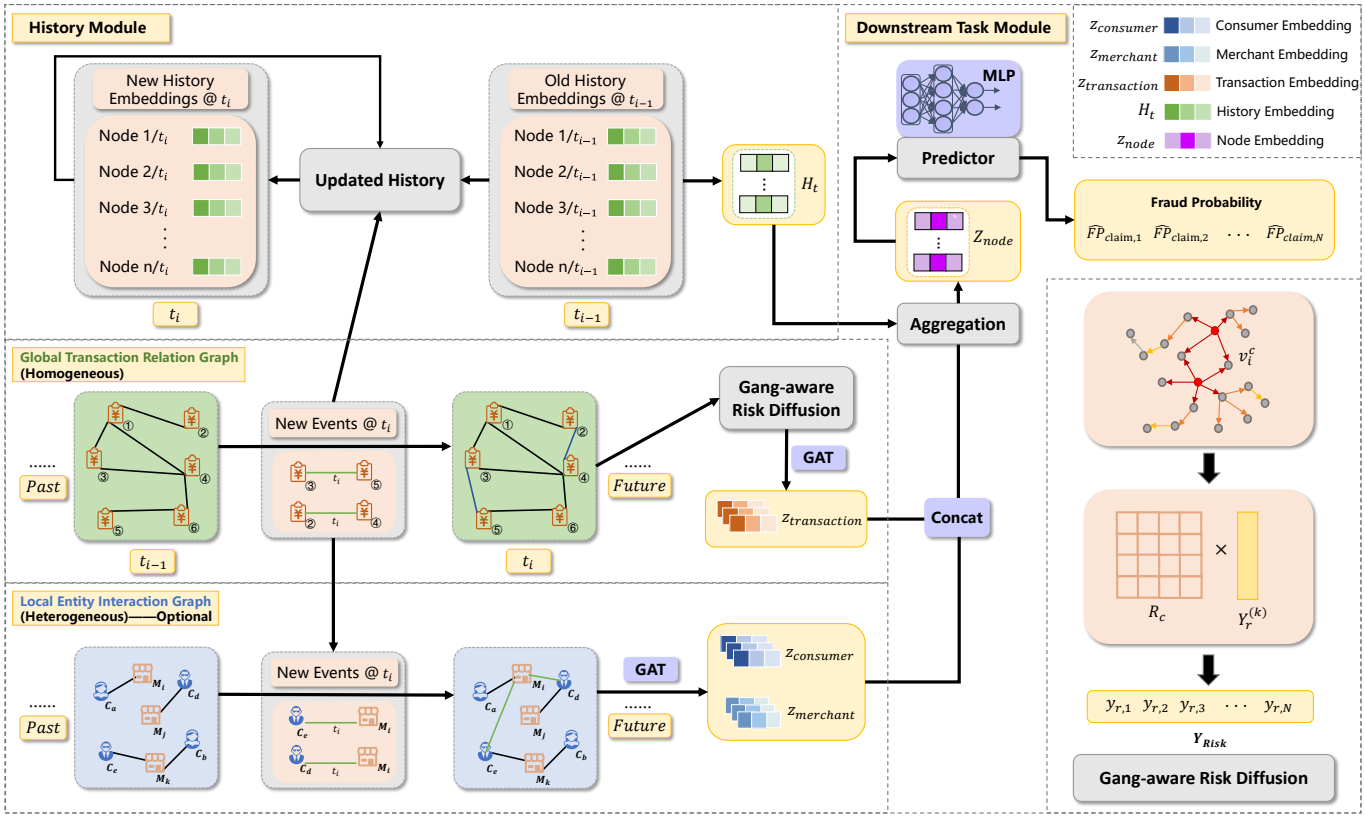


Fig. 2. The model architecture of our proposed Parallel Graph Learning with Temporal Stamp Encoding (PGLTSE) for Fraudulent Transactions Detection method. It comprises two parallel graph learning components: (a) the Homogeneous Transaction Relation Graph for *global* context learning, and (b) the Heterogeneous Entity Interaction Graph for rapidly evolving fraudster behavior's *local* context. Both graphs leverage a history module for dynamic graph learning using temporal stamps. The Gang-aware Risk Diffusion process is embedded in the Homogeneous Transaction Relation Graph. The final Downstream Task Module performs fraud probability prediction, optimizing for evolving organized fraud patterns.

$$\boldsymbol{m}^{\boldsymbol{l}}_{\boldsymbol{v}^c_i \leftarrow \boldsymbol{v}^c_i} = \sigma\big(\mathbf{W_1}\boldsymbol{h}^{\boldsymbol{l-1}}_{\boldsymbol{c,i}}\big),$$

$$\boldsymbol{m}^{\boldsymbol{l}}_{\boldsymbol{v}^c_i \leftarrow \mathcal{N}(\boldsymbol{e}^t_i)} = \sigma\Big(\sum_{v^m_k \in \mathcal{N}_{v^c_i}} \alpha^l_{ik}\big(\frac{1}{N^{e^t}_{ik}} \sum_{e_s \in \mathcal{N}_{ik}(e^t)} \mathbf{W_2}\boldsymbol{h}^{\boldsymbol{l-1}}_{\boldsymbol{e,s}}\big)\Big),$$

$$\boldsymbol{m}^{\boldsymbol{l}}_{\boldsymbol{v}^c_i \leftarrow \mathcal{N}(\boldsymbol{v}^c_i)} = \sigma\Big(\sum_{v^m_k \in \mathcal{N}_{v^c_i}} \alpha^l_{ik}\mathbf{W_3}\boldsymbol{h}^{\boldsymbol{l-1}}_{\boldsymbol{m,k}}\Big), \tag{2}$$

where $\boldsymbol{m}^{\boldsymbol{l}}_{\boldsymbol{v}^c_i \leftarrow \mathcal{N}(\boldsymbol{v}^c_i)}$, $\boldsymbol{m}^{\boldsymbol{l}}_{\boldsymbol{v}^c_i \leftarrow \mathcal{N}(\boldsymbol{e}^t_i)}$, $\boldsymbol{m}^{\boldsymbol{l}}_{\boldsymbol{v}^c_i \leftarrow \boldsymbol{v}^c_i}$ represent the aggregation message from adjacent nodes, edges of v^c_i, and the hidden state of the node in the last layer, respectively. Transformation matrices $\mathbf{W_1} \in \mathbb{R}^{d^C \times d^C}$, $\mathbf{W_2} \in \mathbb{R}^{d^C \times d^E}$, $\mathbf{W_3} \in \mathbb{R}^{d^C \times d^M}$ are employed. We integrate the propagated messages and formulate the updating paradigm as:

$$\boldsymbol{h}^{\boldsymbol{l}}_{\boldsymbol{c,i}} = \boldsymbol{m}^{\boldsymbol{l}}_{\boldsymbol{v_{c,i}} \leftarrow \boldsymbol{v_{c,i}}} + \boldsymbol{m}^{\boldsymbol{l}}_{\boldsymbol{v_{c,i}} \leftarrow \mathcal{N}(\boldsymbol{v_{c,i}})} + \boldsymbol{m}^{\boldsymbol{l}}_{\boldsymbol{v_{c,i}} \leftarrow \mathcal{N}(\boldsymbol{e^t_{c,i}})}. \tag{3}$$

The method to generate the hidden vector $\boldsymbol{h}^{\boldsymbol{l}}_{\boldsymbol{m}}$ for merchant nodes is analogous to the approach detailed previously. Through aggregation sub-layers, the model adeptly transforms the input features of node entities into advanced representations, elucidating deep structural connections.

Diverging from conventional graph neural networks, which are limited to processing nodes, our model also incorporates a function to update edge attributes, thereby generating trade order embeddings. These embeddings enrich the model with high-order data from transaction entities. For $\forall e^t \in \mathcal{E}$, the hidden states of the transaction and its associated consumer and merchant from the previous layer $l-1$ are denoted as $\boldsymbol{h}^{\boldsymbol{l-1}}_{\boldsymbol{e}}, \boldsymbol{h}^{\boldsymbol{l-1}}_{\boldsymbol{C(e)}}, \boldsymbol{h}^{\boldsymbol{l-1}}_{\boldsymbol{M(e)}}$ respectively. The aggregation function is defined as follows:

$$\boldsymbol{h}^{\boldsymbol{l}}_{\boldsymbol{e}} = \sigma\Big(\mathbf{W_e}\Big[\boldsymbol{h}^{\boldsymbol{l-1}}_{\boldsymbol{e}} || \boldsymbol{h}^{\boldsymbol{l-1}}_{\boldsymbol{C(e)}} || \boldsymbol{h}^{\boldsymbol{l-1}}_{\boldsymbol{M(e)}}\Big]\Big), \tag{4}$$

where $\mathbf{W_e}$ represents an learnable matrix for updating edges. In this context, we employ concatenation to integrate messages, allowing the network to extract more intricate features from the input spaces and enhance the encoding of local structural information through stacking multiple graph learning layers. The ultimately encoded representations for consumers and merchants are respectively labeled as $\boldsymbol{z_c}$ and $\boldsymbol{z_m}$, which are subsequently utilized in the downstream detection process.

B. Global Transaction Relation Graph Learning

To integrate inter-dependent knowledge specific to each transaction into our model and assess the global gang-level risk, we construct a transaction-transaction graph $\mathcal{G}_{tr}(\mathcal{V}_{tr}, \mathcal{E}_{tr})$. An edge $e^t_{i,j} \in \mathcal{E}_{tr}$ is established between nodes v^{tr}_i and v^{tr}_j within $\mathcal{V}_{tr}$ when they share a common consumer or merchant. This graph models a gang-level network enabling the modeling of organized fraud patterns more effectively.

1) Gang-aware Risk Diffusion: To capture extensive patterns of conspiracy fraud, we introduce a risk diffusion model within the global transaction relational graph. Using fraud classification labels of identified nodes as targets, a label propagation algorithm computes the disseminated risks. Conventional label propagation (LP) algorithms rely on an adjacency matrix, only accounting for the 1-hop neighborhood of the initial node. To address this, we introduce an advanced gang-aware risk diffusion algorithm, utilizing a risk matrix (RM) $\mathbf{R_{tr}}$, expanding upon the original adjacency matrix $\mathbf{A_{tr}}$ of the transaction relation graph.

In a graph structure, the impact of direct information diffusion is the same as multiple indirect diffusion. By accessing the multi-hop neighborhood of a node v and estimating their immediate risk diffusion intensity, the diffusion range can be scaled during the iteration. We select nodes associated with known gang-related fraudulent transactions from the training dataset, identified as starting nodes of risk contagion S_{tr}. These nodes serve as source nodes for establishing contagion links. Starting from these source nodes, a biased random walk performs a depth-first traversal of their neighborhoods, propagating the risk of organized transaction fraud to surrounding nodes. Direct connections between source nodes and their multi-hop neighbors extend the adjacency matrix, denoted as $\mathbf{A}_{\mathbf{tr}}^{'}$, recording the original hop distances. For $\forall v_i, v_j \in \mathcal{V}_{tr}$, the extended risk matrix is computed as:

$$\mathbf{R_{tr}} = \gamma \mathbf{A}_{\mathbf{tr}}^{'} + (1-\gamma)\mathbf{A_{tr}}, \tag{5}$$

where γ is a hyperparameter balancing the effects of multi-hop neighbors and direct neighborhood. We employ the extended risk matrix $\mathbf{R_{tr}}$ for multi-hop label propagation, addressing the deficiency in short-range perception inherent to traditional label propagation algorithms. During risk contagion, the propagation step is updated iteratively as:

$$\mathbf{H_{tr}}^{(t+1)} = (1-\lambda)\mathbf{R_{tr}}\mathbf{H_{tr}}^{(t)} + \lambda \mathbf{Y_{tr}}, \tag{6}$$

where $\mathbf{Y_{tr}}$ denotes the initialized risk labels, $\mathbf{H_{tr}}^{(t)}$ is the risk propagation vector of nodes in step t, and λ is the teleportation probability. The convergence of the iterative update results in the risk vector $\mathbf{H_{tr}}$ encoding the node-level risk attributes, leveraging the global structural features of the transaction relation graph.

C. Fraud Detection Network

Utilizing node embeddings from local and global perspectives and their respective propagation attributes, we construct a detection network to assess transaction risks. The locally encoded consumer and merchant representations $\boldsymbol{z_c}, \boldsymbol{z_m}$, and globally encoded representation $\boldsymbol{H_{tr}}$ undergo concatenation to form an input vector $\boldsymbol{X_{input}}$:

$$\boldsymbol{X_{input}} = [\boldsymbol{z_c}||\boldsymbol{z_m}||\boldsymbol{H_{tr}}]\,. \tag{7}$$

The fraud detection network, implemented as a multi-layer perceptron (MLP), maps $\boldsymbol{X_{input}}$ to output the probability of fraud for each transaction, expressed as:

$$\hat{y} = \sigma\left(\mathbf{W_o} \cdot \boldsymbol{X_{input}} + \mathbf{b_o}\right), \tag{8}$$

where $\mathbf{W_o}, \mathbf{b_o}$ are the weights and biases of the MLP, respectively, and σ denotes the activation function.

D. Model Training and Optimization

To optimize the fraud detection network, we adopt a binary cross-entropy loss function, given by:

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log\left(\hat{y}_i\right) + (1-y_i)\log\left(1-\hat{y}_i\right)\right], \tag{9}$$

where y_i and $\hat{y}_i$ are the actual and predicted labels, respectively, for each transaction in the training set. We employ the Adam optimizer with early stopping criteria to minimize the loss function and enhance the model's generalization ability. The training process iteratively updates the model parameters until convergence, ensuring robust detection performance.

IV. Temporal Graph Learning

A. Overview

Fraud detection within financial networks demands an agile response to evolving transaction patterns. Our approach involves extending traditional static graph neural networks to accommodate dynamic scenarios by introducing a History Module. This advanced module captures the ongoing temporal changes in nodes and edges, crucial for detecting evolving fraudulent activities. By integrating temporal dynamics, the model does not merely analyze discrete events but understands them as part of a continuum, enhancing prediction accuracy and sensitivity to new fraud strategies.

B. Temporal Graph Representation

We conceptualize financial transactions as an evolving series of timestamped events that continuously alter the topology of a global transaction graph. Each event, whether it is an edge modification, node update, or both, modifies the graph's existing structure, emulating real-world transaction dynamics. This continuous modification approach helps the model to grasp complex patterns over time, avoiding the oversimplification associated with static or snapshot-based models. The temporal graph is formalized as $G = \{x(t_1), x(t_2), \ldots\}$, where each $x(t_i)$ represents a time-specific event influencing the graph structure, like additions or deletions of edges. At any given moment t, node representations are captured as $\mathbf{Z}(t) = (\mathbf{z}_1(t), \ldots, \mathbf{z}_n(t))$, providing a comprehensive view of the graph's state.

C. Node Representation Learning

Node representations are pivotal in understanding the current and historical context of transactions. At each timestep t, the total representation for node i, denoted as $z_i(t)$, is computed by considering both the immediate attributes and the interconnected past activities of its neighbors. This is achieved through:

$$z_i(t) = \mathrm{ENC}(\mathbf{z}_i(t), \{\mathbf{z}_j(t-1)\}_{j\in\mathcal{N}(i)}, h_i(t-1)) \tag{10}$$

Here, ENC is a sophisticated neural network encoder that integrates the current node attributes $\mathbf{z}_i(t)$, the previous timestep's

embeddings of its neighbors $\{\mathbf{z}_j(t-1)\}_{j\in\mathcal{N}(i)}$, and the node's historical embedding $h_i(t-1)$. This encoding process is designed to preserve temporal continuity and enrich the node features with a depth of historical insights, facilitating more accurate predictions.

D. History Module

The dynamism of financial fraud necessitates a model that adapts to continuous changes in transaction patterns. The History Module is engineered to update the historical embeddings of each node, incorporating new transaction data while retaining valuable past information. This dual focus on past and present data ensures robust adaptability and learning efficacy.

1) Embedding Update Mechanism: Our model employs a recurrent neural network (RNN) to manage the sequential update of history embeddings. Specifically, we use a Gated Recurrent Unit (GRU) as the RNN architecture, which efficiently captures the temporal dependencies in sequential data. Each node's embedding is updated iteratively, processing each transaction event to refine the node's historical context:

$$h_i(t) = \text{RNN}(h_i(t-1), z_i(t)) \tag{11}$$

This GRU-based updating mechanism ensures that new information is integrated smoothly with existing historical data, maintaining a continuous timeline of node behavior without information leakage during graph representation learning.

2) Embedding Utilization for Fraud Detection: The refined embeddings are crucial for identifying potential fraud:

$$\hat{y}^c_{ij}(t) = \text{PRED}(\mathbf{z}_i(t), \mathbf{z}_j(t), h_i(t), h_j(t)) \tag{12}$$

Here, PRED refers to a two-layer perceptron (MLP) that takes the embeddings as input and predicts the probability of fraudulent transactions.

$$\mathcal{L}(t) = \text{Loss}(\hat{y}^c_{ij}(t), y^c_{ij}(t)) \tag{13}$$

The Loss function used is the binary cross-entropy, which measures the difference between the predicted probabilities and the actual labels, providing a robust metric for model optimization.

E. Batch Interaction and Memory Update

The concluding phase involves updating the History Module based on batch interactions, which helps reinforce the learned patterns and prepare the model for future predictions. The updating process encompasses:

$$\text{NodeUpdate}(h_i(t), e_{ij}(t)) \tag{14}$$

Here, NodeUpdate refers to the process where the historical embedding $h_i(t)$ of node i is updated by incorporating the current interaction $e_{ij}(t)$ with node j. This ensures that the embedding reflects the most recent transaction details.

$$\text{Aggregate}(h_i(t), \{h_j(t)\}_{j\in\mathcal{N}(i)}) \tag{15}$$

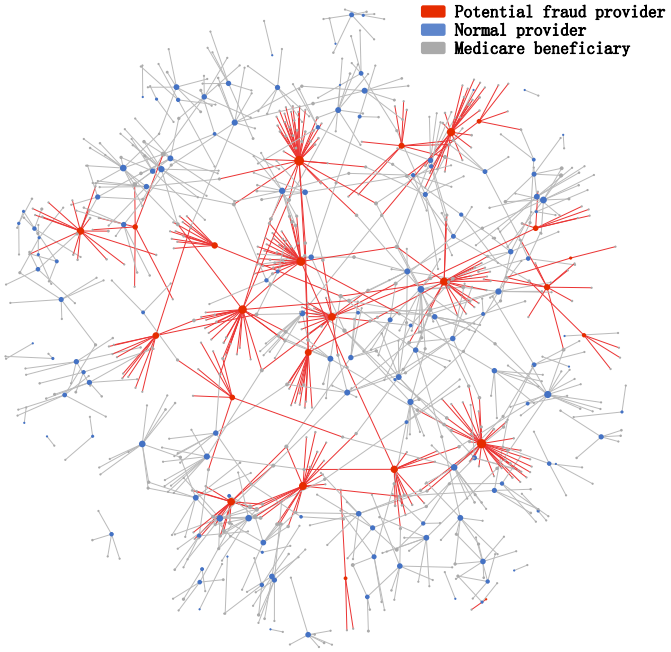


Fig. 3. Visualization of the healthcare dataset. This dataset comprises over 500,000 records of medical insurance claims used for identifying and analyzing medical insurance fraud. The visualization of this dataset reveals a certain degree of gang clustering among transactions, which further underscores the necessity of detecting fraudulent groups.

Aggregate represents a function that combines the historical embeddings $h_j(t)$ of all neighboring nodes $j \in \mathcal{N}(i)$. This aggregation captures the broader network context surrounding node i, providing a richer representation that considers both local and global network dynamics.

$$h_i(t+1) = \text{MemoryUpdate}(h_i(t), \text{Aggregate}) \tag{16}$$

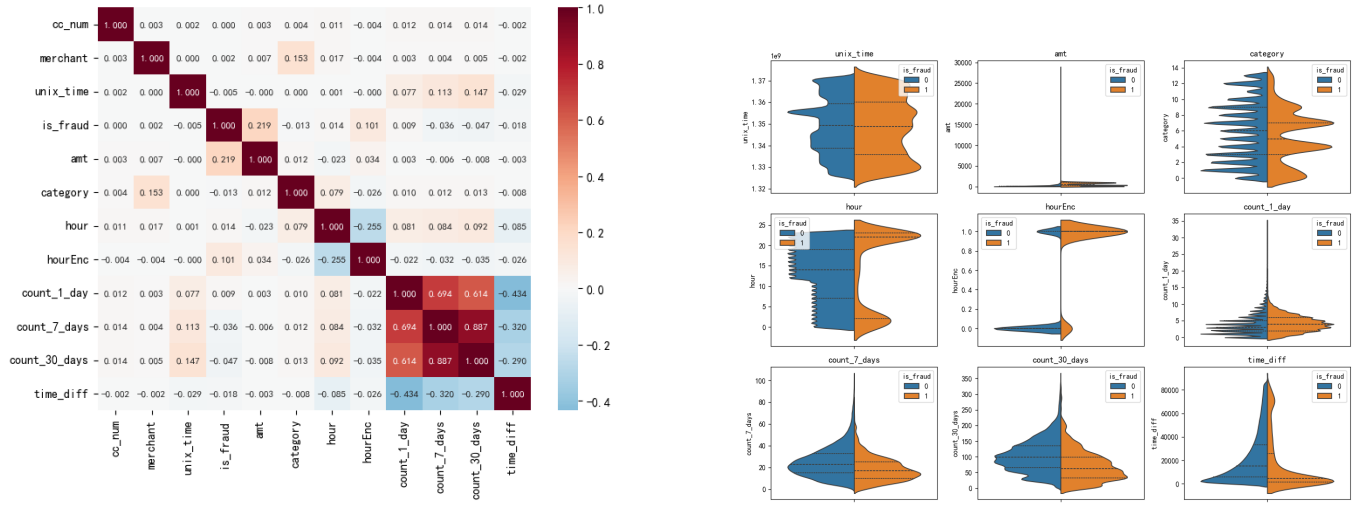
MemoryUpdate is the final step where the historical embedding $h_i(t)$ is updated to $h_i(t+1)$ by integrating the aggregated information. This step ensures that the node's memory is refreshed with the latest network interactions, maintaining an up-to-date historical context for each node.

V. Experiments

A. Datasets

In this study, we selected three datasets from different transaction types: healthcare claims fraud (Healthcare dataset [44]), e-commerce payment fraud (IEEE-CIS dataset [45]), and credit card fraud (Sparkov dataset [43]). The rationale behind choosing these datasets is to cover a broad range of financial transaction scenarios, allowing for a comprehensive evaluation of the proposed model's robustness and applicability. To ensure the model's efficacy across various real-world financial transaction scenarios, we conducted detailed statistical analysis and data preprocessing on each dataset.

The Healthcare dataset comprises over 500,000 records of medical insurance claims used for identifying and analyzing medical insurance fraud. We visualized part of the transactions (as shown in Figure 3), revealing a certain degree of gang clustering among transactions, which further underscores



(a) Heatmap (b) Violin Plot

Fig. 4. The heatmap and violin Plot of the Sparkov dataset [43]. The vertical axis of the heatmap represents the number of related transactions in each order, while the horizontal axis represents the months. Each block's heat value indicates the probability of fraud in the transaction.

the necessity of detecting fraudulent groups. The IEEE-CIS dataset contains approximately 10.9 million records of e-commerce payment transactions, encompassing various fraud patterns. The Sparkov dataset includes about 284,000 records of credit card transactions, focusing on identifying credit card fraud. We used heatmaps and violin plots in Figure 4 to better select features and created Sankey diagrams like Figure 1 based on different timestamps, illustrating the necessity of incorporating temporal feature extraction.

In the data preprocessing stage, we first filtered all fraudulent records as the positive dataset. Since the number of users with no unauthorized transactions greatly exceeds the number of affected users, we downsampled the negative (legitimate) part to alleviate the imbalance problem. For example, users maintaining multiple credit cards were combined into one user ID, and inactive users with fewer than ten records per month were excluded. We then used user-level downsampling for normal users, ensuring that for any user with multiple cards, all cards were either excluded or included in the dataset. Finally, we encoded categorical data, such as user ID and location code, into one-hot representations, rounded the time records to a standard DateTime format (yyyy-MM-dd HH:mm:ss), and normalized the amount attribute, which typically exhibits a long-tail distribution. These preprocessing steps ensured the quality and consistency of the data for subsequent analysis and modeling.

By analyzing and processing these three datasets, we ensured that the proposed model is robust and applicable to various types of financial transaction data. The following provides a brief introduction to each dataset:

- *Healthcare dataset.* This dataset comprises detailed healthcare claims records from various providers, which are utilized to detect medical insurance fraud.
- *IEEE-CIS dataset.* This dataset contains comprehensive records of e-commerce payment transactions, including detailed information and fraud labels, used for analyzing and detecting e-commerce payment fraud.
- *Sparkov dataset.* This dataset includes credit card transaction records, detailing the characteristics and labels of each transaction, which are instrumental in studying and identifying credit card fraud.

B. Compared Methods

We employ the following methods as baselines on our benchmark dataset to highlight the effectiveness of the proposed PGLTSE. In these experiments, the tasks are learned independently. These baselines include:

- *Node2vec [46].* Node2vec is a graph embedding algorithm that effectively generates low-dimensional vector representations of nodes in a graph. It learns node embeddings by optimizing the co-occurrence probability of node sequences generated by random walks, thus capturing the structural information of the graph.
- *CTDNE [40].* CTDNE is a dynamic network embedding method that captures the temporal evolution of networks by modeling event sequences in continuous time. This allows CTDNE to generate node embeddings in dynamic graphs, thereby capturing the dynamics of the graph.
- *DropEdge [47].* DropEdge is a regularization strategy for graph neural networks that prevent overfitting by randomly dropping edges of the graph during training. This strategy improves the model's generalization ability, thereby enhancing the model's performance on unseen data.
- *GAT [48].* Graph Attention Networks (GAT) is a type of graph neural network that introduces an attention mechanism to weigh the contributions of neighbor nodes, thereby better capturing the structural information of the graph. This allows GAT to better consider the relationships between nodes when processing graph data.
- *GraphSAGE [49].* GraphSAGE is a type of graph neural network that generates node embeddings by sampling and aggregating local neighbor information of nodes. This method allows GraphSAGE to handle large-scale and

dynamic graphs, making it suitable for various practical applications.

- *DCI [50].* Dynamic Continuous-time Information Network (DCI) is a dynamic graph embedding method that captures the dynamics of networks by modeling the continuous-time propagation of information in the network. This allows DCI to generate node embeddings in dynamic graphs, thereby capturing the dynamics of the graph.
- *CARE-GNN [39].* CARE-GNN is a graph neural network designed to enhance fraud detection by addressing the challenges of camouflaged fraudsters. It incorporates modules to handle feature and relation camouflage, improving the performance of GNNs in detecting fraud in networks with deceptive behaviors.
- *PC-GNN [5].* PC-GNN is a graph neural network that addresses the class imbalance issue in fraud detection by using a label-balanced sampler and a neighborhood sampler to ensure that minority classes are adequately represented in the model training, improving the model's ability to detect fraudulent activities.
- *TGAT [42].* Temporal Graph Attention Networks (TGAT) is a type of dynamic graph neural network that introduces a temporal attention mechanism to consider the temporal dependency of node interactions. This allows TGAT to better consider temporal information when processing dynamic graphs.
- *DyRep [51].* DyRep is a dynamic graph embedding method that models the structural and temporal evolution of the graph to predict future edges and times. This allows DyRep to generate embeddings for nodes and edges in dynamic graphs, thereby capturing the dynamics of the graph.
- *Jodie [52].* Jodie is a model for predicting future user behavior in dynamic user-item interaction graphs. It models the dynamic embeddings of users and items to capture the temporal patterns of interactions. This allows Jodie to better consider temporal information when processing user-item interaction data.
- *TGN [41].* Temporal Graph Networks (TGN) is a type of dynamic graph neural network that effectively utilizes temporal dimension features by combining a memory module and graph-based operators. This allows TGN to better consider temporal information when processing dynamic graphs.
- *PGLTSE.* The full proposed parallel graph learning model with temporal stamp encoding is presented in this paper.

C. Parameter Settings and Evaluation Metrics

In our implementation, we initially pre-train a two-layer Graph Convolutional Network (GCN) to generate risk embeddings with a dimension of 32. The number of attentional layers within the entity graph is set to 2, and the dimensions of the output representations z_b, z_c, z_p are 16, 32, and 16, respectively. During the parallel training phase, we set the maximum number of epochs at 100 and employ a dropout rate of 0.6 to prevent overfitting. Our framework is implemented using PyTorch 1.12.1 with CUDA 11.3 and Python 3.7 as the backend. For other components, a two-layer Graph Attention Network (GAT) is utilized to obtain embedding representations. The model training is conducted on a server equipped with two 32GB NVIDIA Tesla V100 GPUs.

In terms of evaluation, we leverage four widely recognized metrics: Area Under the Curve (AUC), Area Under the Precision-Recall Curve (AUPRC), recall, and F1 score. These metrics are chosen to comprehensively assess the effectiveness of our model. The AUC measures the model's ability to discriminate between classes at various threshold settings. The AUPRC is particularly useful for comparisons in datasets with an imbalanced distribution of classes. Recall focuses on the model's ability to identify all positive samples, which is crucial for fraud detection where missing a fraudulent transaction can be costly. The F1 score provides a balance between precision and recall, indicating the overall accuracy and robustness of the model. For all four metrics, a higher score signifies superior performance.

D. Overall Performance Comparison

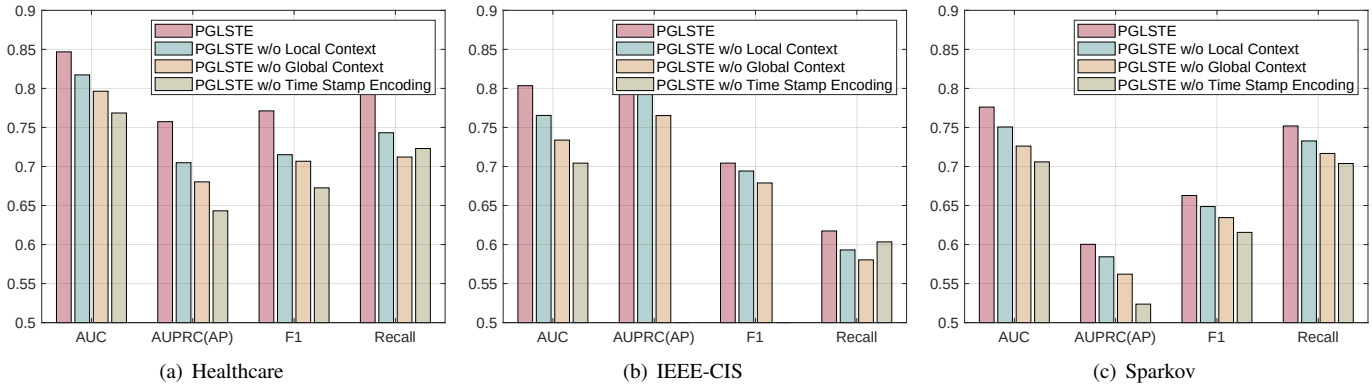
We evaluated the performance of different models for fraud detection across three datasets: Healthcare, IEEE-CIS, and Sparkov. The results, summarized in Table I, highlight the effectiveness of our proposed method, PGLTSE, in comparison to various state-of-the-art baselines.

The initial part of Table I shows the performance of traditional models like Node2vec, CTDNE, and DropEdge. For example, Node2vec's AUC scores are 0.6543, 0.5629, and 0.6221 on the Healthcare, IEEE-CIS, and Sparkov datasets, respectively, suggesting its difficulty in capturing intricate fraud patterns. Advanced models such as GAT, GraphSAGE, and DCI offer better results but do not match the performance of PGLTSE. For instance, GAT achieves an AUC of 0.7743 on the Healthcare dataset and 0.7242 on the Sparkov dataset, indicating the benefits of attention mechanisms. Models like CARE-GNN and PC-GNN, which are designed to address challenges such as camouflaged fraudsters and class imbalance, respectively, demonstrate notable performance gains. CARE-GNN achieves an AUC of 0.6288 on Healthcare, showing its effectiveness in dealing with camouflaged fraudulent behavior, while PC-GNN records an AUC of 0.768 on the same dataset, underscoring its ability to handle class imbalance in fraud detection tasks. However, these models are limited by their static graph approaches and lack of temporal analysis. Methods such as DyRep, Jodie, and TGAT, which incorporate temporal feature learning modules, show further improvements compared to previous methods, but they still do not outperform our approach.

Our method, PGLTSE, demonstrates superior performance compared to other models across the evaluated datasets. Notably, PGLTSE achieves the highest AUC scores of 0.8468 on Healthcare, 0.7935 on IEEE-CIS, and 0.7761 on Sparkov. Furthermore, PGLTSE excels in AUPRC, F1, and Recall metrics. For instance, on the Healthcare dataset, PGLTSE records an AUPRC of 0.7574, an F1 score of 0.7712, and a Recall of 0.8097, outperforming all other models. These

TABLE I
RESULTS OF THE FRAUD DETECTION EXPERIMENT IN THREE DIFFERENT DATASETS.

Dataset	Healthcare				IEEE-CIS				Sparkov			
Metric / Model	AUC	AUPRC(AP)	F1	Recall	AUC	AUPRC(AP)	F1	Recall	AUC	AUPRC(AP)	F1	Recall
Node2vec	0.6543	0.5528	0.6042	0.5318	0.5629	0.2234	0.1722	0.3479	0.6221	0.4128	0.5351	0.6112
CTDNE	0.7024	0.5953	0.5126	0.3911	0.6024	0.2748	0.3352	0.2688	0.6754	0.5829	0.4988	0.6132
DropEdge	0.693	0.5443	0.622	0.5271	0.7235	0.4074	0.3697	0.2676	0.6214	0.4407	0.5027	0.6425
GAT	0.7743	0.6689	0.7108	0.7558	0.692	0.206	0.3927	0.5968	0.7242	0.5482	0.6027	0.6912
GraphSAGE	0.5283	0.4261	0.4844	0.5274	0.5324	0.1784	0.3216	0.4433	0.6021	0.3224	0.4829	0.4243
DCI	0.7167	0.5686	0.6791	0.7309	0.7183	0.3815	0.493	0.5951	0.7359	0.4428	0.6024	0.6687
CARE-GNN	0.6288	0.4264	0.4844	0.5274	0.6019	0.3419	0.6059	0.5584	0.7627	0.2398	0.5719	0.7148
PC-GNN	0.768	0.4935	0.6226	0.5239	0.7346	0.3122	0.5929	0.5601	0.661	0.1984	0.519	0.5175
TGAT	0.7096	0.601	0.6045	0.7778	0.7282	0.7441	0.6252	0.5482	0.7699	0.5939	0.6623	0.7487
DyRep	0.7189	0.5638	0.5508	0.5386	0.6838	0.6642	0.5662	0.5169	0.5204	0.4052	0.3919	0.3933
Jodie	0.7435	0.6132	0.5251	0.4628	0.6555	0.6111	0.5432	0.5145	0.612	0.5891	0.5841	0.7506
TGN	0.8171	0.7462	0.6818	0.7453	0.7617	0.7891	0.6558	0.5698	0.7447	0.5752	0.6413	0.7331
PGLTSE	**0.8468**	**0.7574**	**0.7712**	**0.8097**	**0.7935**	**0.8007**	**0.6843**	**0.5974**	**0.7761**	**0.6003**	**0.6628**	**0.7519**



(a) Healthcare (b) IEEE-CIS (c) Sparkov

Fig. 5. Ablation experiments on three distinct fraud detection datasets: Healthcare, IEEE-CIS, and Sparkov.

results underscore PGLTSE's capability to capture complex and evolving fraud patterns through parallel graph learning and temporal stamp encoding. The outstanding performance of PGLTSE across various fraud detection scenarios emphasizes its robustness and broad applicability in the financial sector.

E. Ablation Studies

To comprehensively evaluate the effectiveness of each component in our proposed approach, we conducted ablation experiments on three datasets: Healthcare, IEEE-CIS Fraud Detection, and Sparkov. The ablation experiments focus on three specific components: the heterogeneous local entity interaction graph learning, the gang-aware risk propagation algorithm in the homogeneous transaction relation graph, and the historical information module for time stamp encoding. Each experiment group compares the full implementation of PGLTSE against three variants with specific components removed, including:

- **PGLTSE w/o Local Context**: Removing the heterogeneous local entity interaction graph learning during the whole parallel graph learning.
- **PGLTSE w/o Global Context**: Replacing the gang-aware risk diffusion algorithm in a homogeneous transaction relation graph with the traditional label propagation algorithm for the pre-training task.
- **PGLTSE w/o Time Stamp Encoding**: Removing the history module and replacing it with the normal message passing and aggregation function algorithm for graph representation learning.

The results in Figure 5 generally show the best performance across all metrics (AUC, AUPRC, F1, Recall) with the complete PGLTSE model, emphasizing the importance

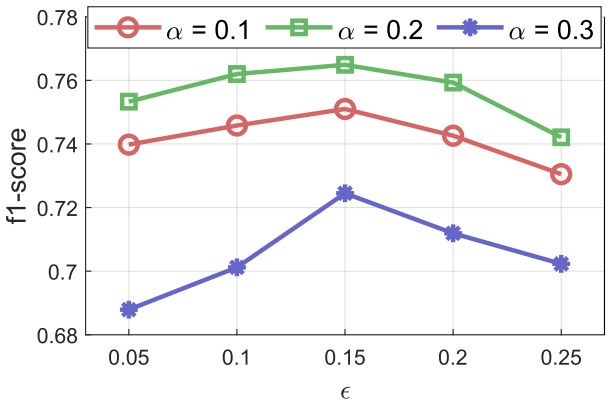


Fig. 6. The results of parameter sensitivity experiments in the medical insurance claims dataset show the impact of the propagation coefficient α and the initial potential risk ϵ on the detection of organized fraud.

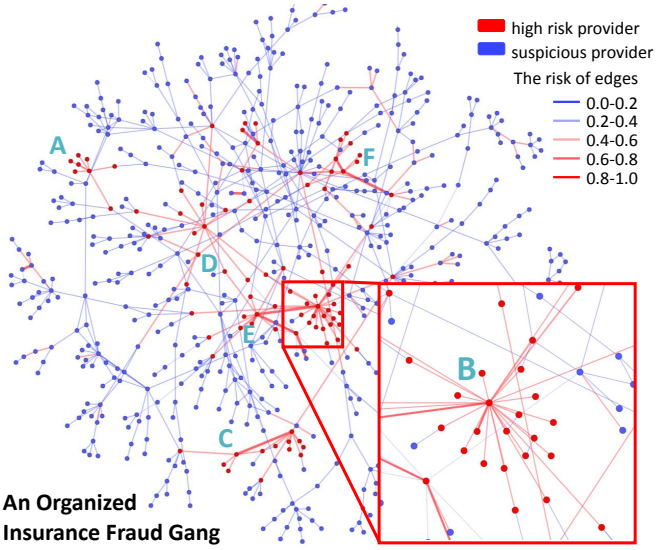


Fig. 7. The structure of a typical medical service provider graph is depicted below. We have identified six criminal factions, labeled as A, B, C, D, E, and F. The efficacy of our approach in identifying most organized fraud claims is demonstrated, with the edge colors representing the likelihood of fraudulent activity.

of integrating local and global contexts along with temporal information. Specifically:

The removal of the local entity interaction graph significantly lowered performance across all datasets, especially on the F1 and Recall metrics, highlighting the critical role of local transaction information in capturing the complexities and local characteristics of fraudulent behaviors.

Similarly, the removal of the gang-aware risk diffusion algorithm also led to a decline in performance, particularly on AUC and AUPRC metrics, underscoring the effectiveness of the gang-aware risk propagation algorithm in understanding global patterns of fraudulent behaviors within the entire transaction network.

Moreover, the removal of the history module adversely affected all metrics across all datasets, validating the essential role of the history module in enhancing the model's accuracy in capturing transaction temporal pattern embeddings.

By conducting ablation studies across three distinct datasets, the results demonstrate not only the effectiveness of each component of PGLTSE but also the model's robustness and broad applicability. Whether dealing with medical insurance claims, e-commerce fraud, or complex financial services fraud, PGLTSE effectively identifies and predicts fraudulent behaviors. The consistency of results across datasets underscores the importance of adopting graph-based approaches and integrating various contextual and temporal dimensions to adapt to evolving fraud strategies and complex transaction patterns. This highlights PGLTSE's practical applicability and superior performance in real-world scenarios.

F. Parameter Sensitivity

Take the healthcare transaction Dataset as an example, we further explore the generalization performance of the model concerning the hyper-parameters of initial potential risk ϵ and propagation parameter α. Specifically, we assess their impact on risk diffusion in organized fraud detection tasks and present the averaged F1 score of performance on inpatient and outpatient test data in Figure 6. The initial potential risk ϵ for normal nodes is varied from 0.05 to 0.25 in increments of 0.05, while the propagation coefficient α is selected from the set $\{0.1, 0.2, 0.3\}$. Our results show that PGLTSE performs better as ϵ increases from 0.05 to 0.15, with the average F1 score peaking when ϵ is 0.15 and α is 0.2. However, the performance declines when ϵ exceeds 0.15. This decline may be attributed to the overestimation of risk from legal behavior, causing the model to become overly sensitive in identifying fraudulent transactions. Additionally, as α increases from 0.2 to 0.3, performance drops more rapidly, likely because a higher α limits the diffusion scope, thereby impairing the model's ability to learn community-level risk representation.

G. Case Studies

This section presents two case studies demonstrating how the PGLTSE model addresses critical challenges in fraud detection: Fraud Gang Detection and Temporal Fraud Pattern Discovery. The model integrates a community-level risk propagation algorithm that assesses interconnected risks across organized fraud activities, enhancing the identification of complex fraud schemes. Additionally, it has a history module to capitalize on transactional history information, enabling the detection of evolving fraudulent patterns.

1) Fraud Gang Detection: Figure 7 shows a complex network of 663 medical service providers, illustrating the intricate relationships facilitated through shared claims with identical beneficiaries. In this network, connections between any two providers are denoted by edges whose color and thickness are indicative of the fraud probability, as assessed by our method. Providers connected by claims with a higher likelihood of fraud are categorized as high-risk, while others are considered low-risk.

PGLTSE has delineated six prominent fraud gangs within this network, identified as Groups A through F. These factions are characterized by a pronounced clustering pattern, which

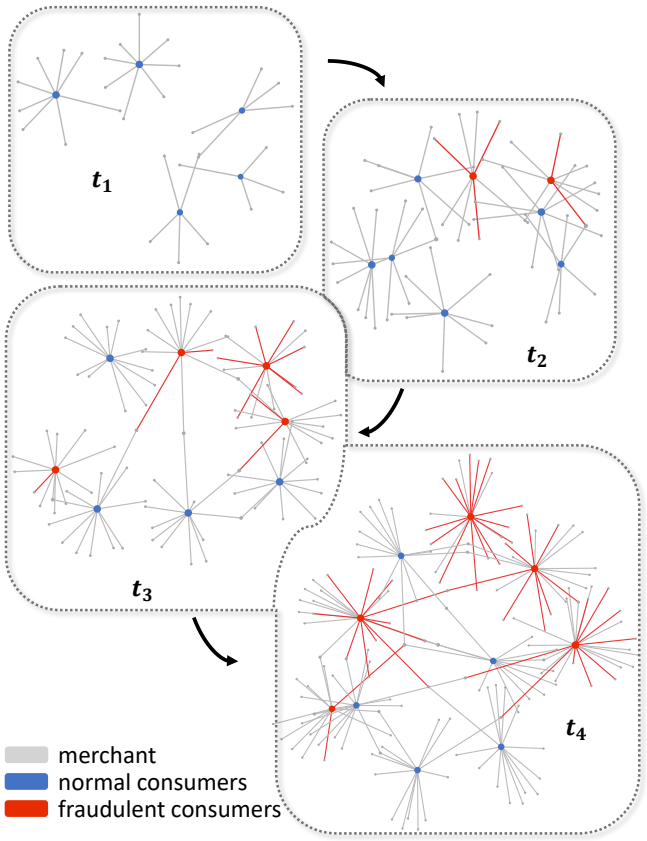


Fig. 8. Evolution of a transaction network over time, illustrating the progression of normal and fraudulent consumer behaviors. The image shows transactions from the Sparkov dataset at four different timestamps: t_1, t_2, t_3, and t_4. Nodes are categorized as merchants (gray), normal consumers (blue), and fraudulent consumers (red).

our method effectively identifies, underscoring the prevalence of fraudulent activities within these clusters. Particularly noteworthy is Group B, highlighted in red squares, where a significant number of adjacent nodes are implicated in systematic fraud, forming a conspicuous cluster. This observation affirms the effectiveness of our community-aware risk diffusion strategy, which is meticulously designed to detect and delineate such organized fraud schemes.

Further scrutiny of Group B and adjacent areas reveals that the likelihood of fraudulent claims is intricately linked to the behaviors of the connected entities. This connection highlights the necessity of integrating diverse entity information to enhance the detection precision. The capability of our model to synthesize such information allows for more accurate identification of organized fraud patterns, showcasing its utility in dissecting and understanding complex fraudulent networks.

2) Temporal Fraud Pattern Discovery: In Figure 8, we scrutinize the dynamic behavior of consumer transactions within the Sparkov dataset across four distinct timestamps, from t_1 to t_4. The primary objective is to trace how typically normal consumers transition into fraudulent activities and to identify the emergence of new fraudulent entities within the network. This investigation affirms the capability of our temporal graph learning model to effectively detect and predict these evolving fraudulent behaviors.

Initially, at t_1, the network predominantly exhibits regular transactional activities, with minimal indications of fraud. As the timeline progresses to t_2, we begin to observe the initial signs of fraudulent activities. Between t_2 and t_3, a noteworthy shift occurs—nodes that previously engaged in normal transactions start to participate in fraudulent behaviors. This transition not only underscores the adaptability of fraudsters, who initially integrate into the network as benign entities but also highlights the critical period during which these entities begin exhibiting fraudulent intentions.

The expansion from t_3 to t_4 is marked by a significant proliferation of fraudulent activities, with a substantial increase in the number of nodes involved in fraud. This phase reveals that new fraudulent entities, not previously detected at earlier timestamps, have been introduced into the network. The observation of these new nodes at t_4 suggests that the network is persistently targeted by incoming fraudsters, reflecting a continuous threat landscape.

Our model leverages the history module, enhanced with temporal dimension information, to meticulously map out these transitions. It excels in identifying not just the consumers who transition from normal to fraudulent behaviors from one timestamp to the next, but also in detecting the influx of new fraudulent consumers into the network. The analytical depth provided by continuous monitoring of transaction networks through our model significantly enriches the accuracy of fraud predictions and facilitates proactive fraud management strategies. This case study not only validates the effectiveness of our approach but also illustrates the critical importance of understanding and adapting to the temporal dynamics of fraud within transaction networks.

VI. Conclusion and Discussion

The research presented in this paper marks a significant advancement in the domain of fraud detection within financial transaction networks. Our proposed model, PGLTSE, integrates parallel graph learning with temporal stamp encoding to effectively capture the dynamic nature of fraud. By utilizing a history module and a community-level risk propagation algorithm, the model excels in identifying and predicting fraudulent activities across various transaction types and scenarios. The experimental results showcase the model's robustness and superiority in handling real-world data, outperforming conventional methods. The ability to dynamically adapt to new fraud patterns and its high accuracy in recognizing complex fraud structures make PGLTSE a valuable tool for financial institutions aiming to safeguard their operations against fraud. In the future, we plan to enhance the model's scalability and deploy it across larger-scale and more diverse datasets. This will involve refining the model's architecture to better handle extensive data and exploring methods to optimize its performance in large-scale operational environments. These efforts will further solidify PGLTSE's role as a critical asset in the fight against financial fraud.

REFERENCES

[1] J. West and M. Bhattacharya, "Intelligent financial fraud detection: a comprehensive review," *Computers & security*, vol. 57, pp. 47–66, 2016.

[2] C. Phua, V. Lee, K. Smith, and R. Gayler, "A comprehensive survey of data mining-based fraud detection research," *arXiv preprint arXiv:1009.6119*, 2010.

[3] S. Marchal and S. Szyller, "Detecting organized ecommerce fraud using scalable categorical clustering," in *Proceedings of the 35th Annual Computer Security Applications Conference*, 2019, pp. 215–228.

[4] D. Cheng, Y. Ye, S. Xiang, Z. Ma, Y. Zhang, and C. Jiang, "Anti-money laundering by group-aware deep graph learning," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–13, 2023.

[5] Y. Liu, X. Ao, Z. Qin, J. Chi, J. Feng, H. Yang, and Q. He, "Pick and choose: a gnn-based imbalanced learning approach for fraud detection," in *Proceedings of the web conference 2021*, 2021, pp. 3168–3177.

[6] X. Zhu, X. Ao, Z. Qin, Y. Chang, Y. Liu, Q. He, and J. Li, "Intelligent financial fraud detection practices in post-pandemic era," *The Innovation*, vol. 2, no. 4, 2021.

[7] J. A. Major and D. R. Riedinger, "Efd: A hybrid knowledge/statistical-based system for the detection of fraud," *International Journal of Intelligent Systems*, vol. 7, no. 7, pp. 687–703, 1992.

[8] Y. Kou, C.-T. Lu, S. Sirwongwattana, and Y.-P. Huang, "Survey of fraud detection techniques," in *IEEE international conference on networking, sensing and control, 2004*, vol. 2. IEEE, 2004, pp. 749–754.

[9] Ravelin. (2023) Machine learning for fraud detection.

[10] S. Ghosh and D. L. Reilly, "Credit card fraud detection with a neural-network," in *System Sciences, 1994. Proceedings of the Twenty-Seventh Hawaii International Conference on*, vol. 3. IEEE, 1994, pp. 621–630.

[11] J. Schiller, "The impact of insurance fraud detection systems," *Journal of Risk and Insurance*, vol. 73, no. 3, pp. 421–438, 2006.

[12] R. Patidar, L. Sharma *et al.*, "Credit card fraud detection using neural network," *International Journal of Soft Computing and Engineering (IJSCE)*, vol. 1, no. 32-38, 2011.

[13] D. Cheng, F. Yang, S. Xiang, and J. Liu, "Financial time series forecasting with multi-modality graph neural network," *Pattern Recognition*, vol. 121, p. 108218, 2022.

[14] D. Cheng, S. Xiang, C. Shang, Y. Zhang, F. Yang, and L. Zhang, "Spatio-temporal attention-based neural network for credit card fraud detection," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 01, 2020, pp. 362–369.

[15] P. Dua and S. Bais, "Supervised learning methods for fraud detection in healthcare insurance," pp. 261–285, 2014.

[16] R. A. Bauder and T. M. Khoshgoftaar, "Medicare fraud detection using machine learning methods," in *16th IEEE international conference on machine learning and applications (ICMLA)*. IEEE, 2017, pp. 858–865.

[17] Y. Pandey, "Credit card fraud detection using deep learning." *International Journal of Advanced Research in Computer Science*, vol. 8, no. 5, 2017.

[18] A. Pumsirirat and Y. Liu, "Credit card fraud detection using deep learning based on auto-encoder and restricted boltzmann machine," *International Journal of advanced computer science and applications*, vol. 9, no. 1, 2018.

[19] D. Lim, X. Li, F. Hohne, and S.-N. Lim, "New benchmarks for learning on non-homophilous graphs," *arXiv preprint arXiv:2104.01404*, 2021.

[20] S. Xiang, M. Zhu, D. Cheng, E. Li, R. Zhao, Y. Ouyang, L. Chen, and Y. Zheng, "Semi-supervised credit card fraud detection via attribute-driven graph representation," in *AAAI*, 2023, pp. 1–8.

[21] D. Cheng, X. Wang, Y. Zhang, and L. Zhang, "Risk guarantee prediction in networked-loans," in *IJCAI International Joint Conference on Artificial Intelligence*, 2020, pp. 1–7.

[22] B. Xu, H. Shen, B. Sun, R. An, Q. Cao, and X. Cheng, "Towards consumer loan fraud detection: Graph neural networks with role-constrained conditional random field," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 5, 2021, pp. 4537–4545.

[23] D. Cheng, Z. Niu, J. Li, and C. Jiang, "Regulating systemic crises: Stemming the contagion risk in networked-loans through deep graph learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, pp. 6278–6289, 2023.

[24] D. Cheng, X. Wang, Y. Zhang, and L. Zhang, "Graph neural network for fraud detection via spatial-temporal attention," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 8, pp. 3800–3813, 2020.

[25] F. Itoo, Meenakshi, and S. Singh, "Comparison and analysis of logistic regression, naïve bayes and knn machine learning algorithms for credit card fraud detection," *International Journal of Information Technology*, vol. 13, pp. 1503 – 1511, 2020.

[26] H. Z. Alenzi and N. O. Aljehane, "Fraud detection in credit cards using logistic regression," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 12, 2020.

[27] P. Hájek, M. Z. Abedin, and U. Sivarajah, "Fraud detection in mobile payment systems using an xgboost-based framework," *Information Systems Frontiers*, pp. 1 – 19, 2022.

[28] P. Save, P. Tiwarekar, K. N. Jain, and N. Mahyavanshi, "A novel idea for credit card fraud detection using decision tree," *International Journal of Computer Applications*, vol. 161, no. 13, pp. 6–9, 2017.

[29] N. Rtayli and N. Enneya, "Enhanced credit card fraud detection based on svm-recursive feature elimination and hyper-parameters optimization," *J. Inf. Secur. Appl.*, vol. 55, p. 102596, 2020.

[30] S. Kumar, V. K. Gunjan, M. D. Ansari, and R. Pathak, "Credit card fraud detection using support vector machine," in *Proceedings of the 2nd International Conference on Recent Trends in Machine Learning, IoT, Smart Cities and Applications: ICMISC 2021*. Springer, 2022, pp. 27–37.

[31] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "Lightgbm: A highly efficient gradient boosting decision tree," in *Neural Information Processing Systems*, 2017.

[32] K. Fu, D. Cheng, Y. Tu, and L. Zhang, "Credit card fraud detection using convolutional neural networks," in *Neural Information Processing: 23rd International Conference, ICONIP 2016, Kyoto, Japan, October 16–21, 2016, Proceedings, Part III 23*. Springer, 2016, pp. 483–490.

[33] J. Jurgovsky, M. Granitzer, K. Ziegler, S. Calabretto, P.-E. Portier, L. He-Guelton, and O. Caelen, "Sequence classification for credit-card fraud detection," *Expert systems with applications*, vol. 100, pp. 234–245, 2018.

[34] D. Wang, Y. Qi, J. Lin, P. Cui, Q. Jia, Z. Wang, Y. Fang, Q. Yu, J. Zhou, and S. Yang, "A semi-supervised graph attentive network for financial fraud detection," *2019 IEEE International Conference on Data Mining (ICDM)*, pp. 598–607, 2019.

[35] Y. Dou, Z. Liu, L. Sun, Y. Deng, H. Peng, and P. S. Yu, "Enhancing graph neural network-based fraud detectors against camouflaged fraudsters," *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 2020.

[36] Y. Liu, X. Ao, Z. Qin, J. Chi, J. Feng, H. Yang, and Q. He, "Pick and choose: A gnn-based imbalanced learning approach for fraud detection," *Proceedings of the Web Conference 2021*, 2021.

[37] J. Tang, J. Li, Z. Gao, and J. Li, "Rethinking graph neural networks for anomaly detection," in *International Conference on Machine Learning*. PMLR, 2022, pp. 21 076–21 089.

[38] M. Duan, T. Zheng, Y. Gao, G. Wang, Z. Feng, and X. Wang, "Dga-gnn: Dynamic grouping aggregation gnn for fraud detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 10, 2024, pp. 11 820–11 828.

[39] Y. Dou, Z. Liu, L. Sun, Y. Deng, H. Peng, and P. S. Yu, "Enhancing graph neural network-based fraud detectors against camouflaged fraudsters," in *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 2020, pp. 315–324.

[40] G. H. Nguyen, J. B. Lee, R. A. Rossi, N. K. Ahmed, E. Koh, and S. Kim, "Continuous-time dynamic network embeddings," *ACM Transactions on the Web (TWEB)*, vol. 12, no. 4, p. Article 19, 2018.

[41] E. Rossi, B. Chamberlain, F. Frasca, D. Eynard, F. Monti, and M. Bronstein, "Temporal graph networks for deep learning on dynamic graphs," in *ICML 2020 Workshop on Graph Representation Learning*, 2020.

[42] D. Xu, C. Ruan, E. Korpeoglu, S. Kumar, and K. Achan, "Inductive representation learning on temporal graphs," in *International Conference on Learning Representations (ICLR)*, 2020.

[43] K. Shenoy, "Credit card transactions fraud detection dataset."

[44] R. A. Gupta, "Healthcare provider fraud detection analysis," https://www.kaggle.com/datasets/rohitrox/healthcare-provider-fraud-detection-analysis, 2020, accessed: 2021-01-25.

[45] H. Addison, B.-M. Bernadette, I. C. inversion, L. John, Lynn@Vesta, Marcus2010, and P. H. Abbass, "Ieee-cis fraud detection," 2019.

[46] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016.

[47] Y. Rong, W. Huang, T. Xu, and J. Huang, "Dropedge: Towards deep graph convolutional networks on node classification," in *International Conference on Learning Representations*, 2020.

[48] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph Attention Networks," *International Conference on Learning Representations*, 2018, accepted as poster.

[49] W. L. Hamilton, R. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *NIPS*, 2017.

[50] Y. Wang, J. Zhang, S. Guo, H. Yin, C. Li, and H. Chen, "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, pp. 1239–1248.

[51] R. Trivedi, M. Farajtabar, P. Biswal, and H. Zha, "Dyrep: Learning representations over dynamic graphs," in *International conference on learning representations*, 2019.

[52] S. Kumar, X. Zhang, and J. Leskovec, "Predicting dynamic embedding trajectory in temporal interaction networks," in *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 2019, pp. 1269–1278.

Jiacheng Ma is a Master candidate in the Fintech-Lab, majoring computer intelligence technology in Tongji University, Shanghai, China. He received his BSc in Software Engineering from Nanjing University of Aeronautics and Astronautics in China. His research interests include data mining, graph machine learning in finance, and graph-based fraud detection.

Sheng Xiang is a PhD candidate in the Australian Artificial Intelligence Institute (AAII), major in Computer Science, University of Technology, Sydney (UTS). He received his BSc degree from Shanghai Jiao Tong University. His research interests include graph machine learning in finance, graph generative algorithms, temporal graph processing, graph-based fraud detection, and heterogeneous graphs.

Qiang Li is currently pursuing the Bachelor's degree in Computer Science and Technology at Tongji University, Shanghai, China. He is conducting research internships at the Fin-techLab, primarily focusing on research related to financial transaction fraud detection. His research interests include data mining, graph neural networks, financial big data, and multimodal sentiment analysis.

Liangyu Yuan is an undergraduate in the Department of Computer Science and Technology at Tongji University, Shanghai, China. He is currently engaged as a research intern at the Tongji Fintech Lab. His research focuses on fraud detection and generative models for graphs.

Dawei Cheng received the PhD degree in computer science from Shanghai Jiao Tong University, Shanghai, China. He is an associate professor with the Department of Computer Science and Technology, Tongji University, Shanghai, China. Before that, he was a postdoctoral associate with MoE Key Laboratory of Artificial Intelligence, Department of Computer Science, Shanghai Jiao Tong University. His research fields include data mining, graph learning, and Big Data in finance.

Changjun Jiang received the PhD degree from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 1995. He is currently a professor at the Department of Computer Science and Technology, Tongji University, Shanghai, China. His current research interests include concurrency theory, formal verification of software, service-oriented computing, Big Data in finance, intelligent systems, financial risk management, and Big Data computing.