Role Perceptual Augmented Temporal Graph Network for Related-party Transaction Detection

Xin Liu[1], **Yuanhang Yu**[1], **Peng Zhu**[1*], **Dawei Cheng**[1], **Changjun Jiang**[1]
[1]School of Computer Science and Technology, Tongji University, Shanghai, China
{2432042, yuanhangyu, pengzhu, dcheng, cjjiang}@tongji.edu.cn

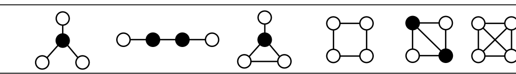
Abstract

Illegal related-party transactions (RPT) are federal felonies that pose a severe threat to the stability and integrity of modern financial systems. The increasing frequency of RPTs forms complex and dynamic networks. Existing temporal graph learning methods tend to treat entities as functionally homogeneous, ignoring the diverse and evolving structural roles of nodes. Role-based embedding methods model global structure by bridging same-role nodes, but their reliance on a unified mechanism for aggregation and evolution means they fail to distinguish the underlying logic of distinct interactions governed by structural roles. The limitations motivate us to develop a customized role-based strategy. It can also adapt to evolving RPT dynamics, thereby forming a continuous regulatory process to combat illegal activities. In this paper, we propose an innovative Role Perceptual Augmented Temporal Graph Network (RPATGN) for proactive RPT detection. We analyze the structural roles of nodes and employ a role-based message passing mechanism that adapts its aggregation strategy based on the roles of interacting nodes. We integrate a variational graph recurrent neural network, enhanced by temporal contextual attention, to explicitly model the dynamics of the roles and the overall network evolution. Extensive experiments on real-world financial datasets demonstrate the effectiveness of our approach for RPT detection. It holds practical significance for fostering robust financial systems and promoting healthy, transparent financial markets.

Code — https://github.com/Claireliu0912/RPATGN

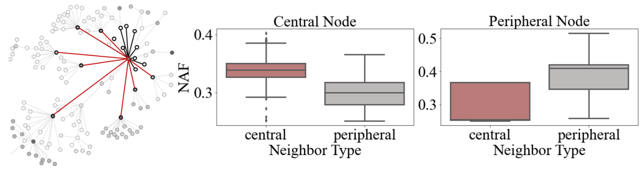
Introduction

Related-party transactions (RPT) involve the transfer of resources, services or obligations between a listed company or its subsidiaries and related parties (Muthupandian 2009), which can improve resource efficiency but pose risks of abuse for personal gain (Kohlbeck and Mayhew 2017). Safe and easily manipulated RPT appears to become the optimal choice for financial fraud to cash out (Mao et al. 2022). RPT has greatly affected the stability and fairness of the market (Kohlbeck and Mayhew 2010; Ryngaert and Thomas 2012; Dou et al. 2022). To achieve proactive supervision, it is crucial to move beyond post-hoc detection towards predicting the evolution of corporate networks. The ability to accurately forecast the emergence of intricate and dynamic relationships is a cornerstone of regulatory technology.



Graphlet						
Frequency	7,572,014	2,295,757	396,456	17,597	318	0

(a) Occurrences of all 4-node connected graphlets on the RPT network with node roles indicated in different colors.



(b) The layout of a typical RPT network. Node color denotes neighborhood aggregation factor (NAF). The box plot shows the NAF distribution for central and peripheral nodes on RPT dataset.

Figure 1: RPT network exhibits a core-periphery structure, and structural roles count in determining node interactions.

Recently, Graph Neural Networks (GNNs) have emerged as a powerful tool to unveil inherent connections in complex RPT networks (Hu et al. 2020; Zheng et al. 2023; Cheng et al. 2025). Given that financial activities unfold over time, RPT networks are inherently dynamic. New entities may emerge as transient shell companies, and interactions can shift abruptly to evade detection. Therefore, dynamic GNNs, which incorporate a temporal dimension, can provide deeper insights into the evolving RPT graphs (Cheng et al. 2018, 2020b; Xiang et al. 2022; Hu et al. 2023). However, many temporal graph learning methods fall short when applied to the RPT networks. They tend to treat nodes as homogeneous, assuming that the principles of link formation are uniform across the graph (Jin, Song, and Shi 2020; Liu et al. 2025). The assumption overlooks a fundamental driver of RPT: the heterogeneous structural roles of the entities involved, which often function differentially in the network.

RPT networks, like most financial networks, are complex relational systems shaped by human behavior and exhibit significant structural role phenomena (Barabási and PÃ3sfai 2016). As shown in Figure 1, the RPT network is domi-

*Corresponding author.

nated by a core-periphery structure, where a few nodes have multiple connections, representing financial institutions with large transaction volumes and multiple parties involved. Numerous nodes represent ordinary entities connected to the core, with a clear distinction between node roles (central and peripheral). The structural roles simultaneously define the nature of interactions and enhance the perception of local and global graph structures. Existing role-based methods treat structural roles as static attributes (Zhang and Kou 2022) or generate auxiliary features by bridging structurally similar nodes (Du et al. 2024; Wang et al. 2024). Despite capturing similarities among same-role nodes, they apply unified operations to functionally distinct role-based interactions and evolution. They fail to distinguish the business purpose of diverse RPT interactions.

Therefore, we propose Role Perceptual Augmented Temporal Graph Network (RPATGN), a novel framework designed for dynamic link prediction in RPT networks. Its core innovation lies in the ability to model not only the diverse functions governed by structural roles but also their dynamic evolution over time. (1) Spatially, a role-based adaptive message passing mechanism is introduced to perceive structural heterogeneity. It is dynamically customized based on the node's role and the role-based composition of its neighborhood, enabling the model to differentiate diverse interactions. It distinguishes the unique behaviors of neighbors with different roles to capture diverse local functions, while simultaneously leveraging the shared patterns among nodes with the same role to effectively model long-range dependencies. (2) Temporally, RPATGN employs a variational graph recurrent neural network to explicitly model the temporal transition of roles. We also utilize a temporal contextual attention mechanism that adaptively weights historical snapshots. In this way, RPATGN generates powerful representations that are sensitive to both the immediate network structure and long-term trajectory of each entity. We summarize the main contributions of this paper as follows:

- To the best of our knowledge, this is the first work that addresses the RPT supervisory challenge in the financial market by proposing a novel temporal graph learning method, which focuses on both the characteristics of RPT role-based patterns and the dynamic diffusions.
- We devise a role-based message passing mechanism that enhances the perception of both local and global structures to capture complex structural patterns and develop a temporal learning architecture to capture the graph evolution and time dependencies across multiple time steps.
- Extensive experiments on real-world datasets demonstrate the superiority of our model compared to state-of-the-art methods. The model was evaluated in the real-world RPT regulatory system, showing its applicability in providing practical technical support to maintain the stability and integrity of the financial system.

Preliminary

Background and Motivation

RPT is a common aspect of modern corporate operation, whose inherent opacity can be exploited for illicit purposes, such as asset tunneling and harming minority shareholder interests. Financial fraud cases involving RPTs, such as the Enron scandal, have resulted in significant financial losses and have undermined public trust in financial markets (Hope and Lu 2020; Bratton 2001; Han et al. 2025; Xiang et al. 2025). The sheer scale and dynamic complexity of the corporate interactions overwhelm traditional, post-hoc auditing methods. This creates an urgent need to shift the regulatory paradigm from reactive detection to proactive supervision.

The paradigm shift hinges on the technical challenges posed by the intrinsic nature of RPT networks. As illustrated in Figure 1, our empirical analysis reveals that RPT networks are dominated by a core-periphery structure and structural roles have direct functional implications. Specifically, we first counted the frequencies of all connected 4-node graphlets on the RPT dataset to uncover the fundamental connection patterns. Then, we devised a composite metric **Neighborhood Aggregation Factor (NAF)**, inspired by the neighborhood entropy and center-neighbor similarity from (Xie et al. 2020) to quantify a node's functional influence within the context of GNN message passing. The detailed empirical analysis is provided in the supplementary files. Therefore, an effective predictive model must overcome two challenges: (1) capture the dynamic evolution of structural roles, and (2) devise customized strategies for the diverse interaction patterns governed by these roles. To this end, we develop a role perceptual augmented model that can predict the formation of potential RPTs, thus enabling proactive supervision and helping to maintain financial stability.

Problem Formulation

RPT Network. In RPT networks, we treat entities involved in the transaction process as nodes and construct the network based on whether there is an RPT between the entities. RPT data is typically derived from periodic corporate disclosures (e.g., quarterly reports), presented as snapshots at a specific time step rather than a continuous data stream. Therefore, we use discrete graphs, establishing snapshots based on transaction time and duration to capture the state changes of RPT. A discrete-time temporal graph is defined as a series of observed static graph snapshots, $\mathcal{G} = \{G^1, G^2, ..., G^T\}$, where T is the number of snapshots. Each snapshot $G^t = (V^t, E^t)$ is an undirected graph with the current node set V^t and edge set E^t. In addition, each snapshot can be represented by its M-dimensional node attributes $\mathbf{X}^t \in \mathbf{R}^{N\times M}$ and the corresponding adjacency matrix $\mathbf{A}^t$. As time evolves, both the sets of nodes and edges are non-static, placing our task as inductive extrapolation of discrete graphs.

RPT Detection. Our primary objective is the proactive detection of RPTs; that is, based on the history of corporate interactions, we aim to predict which RPTs are likely to occur in the future. The objective is formally framed as a temporal graph link prediction task. The goal is to learn latent representations $\mathbf{h}^t$ by giving the snapshots till timestamp t. For each snapshot, we can use a static graph neural network f_1 to learn spatial patterns and a recurrent network f_2 to learn temporal patterns (Skarding, Gabrys, and Musial 2021):

$$\mathbf{h}^t = f_2(f_1(\mathbf{A}^t, \mathbf{X}^t), \mathbf{h}^{t-1}). \tag{1}$$

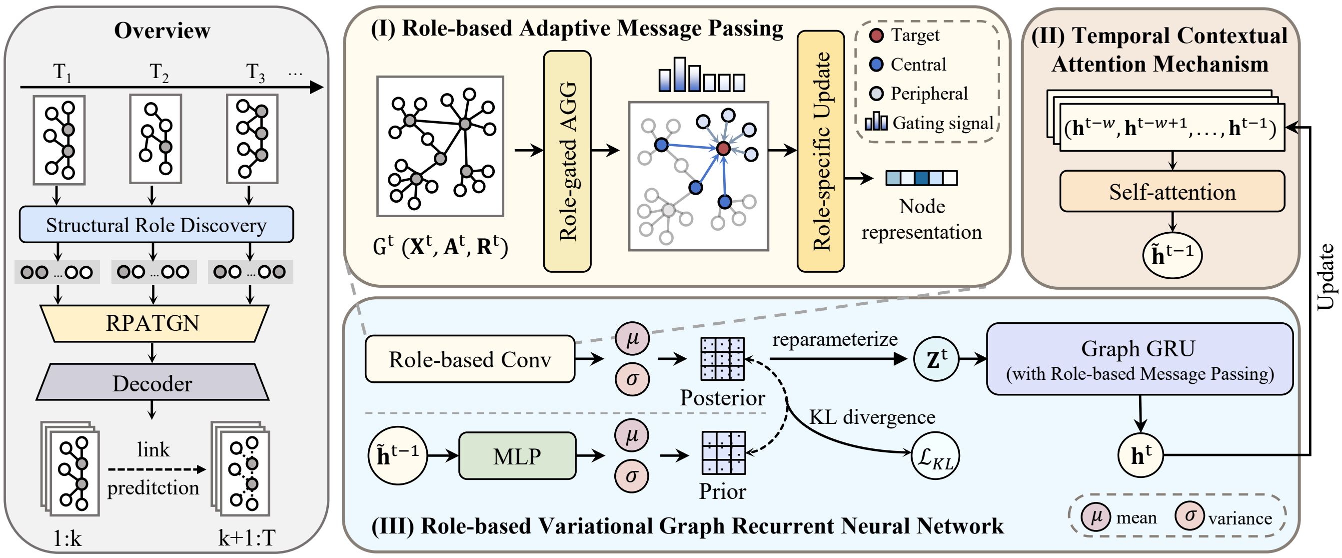


Figure 2: Overview of the RPATGN framework. The model identifies structural roles from dynamic graphs to guide (I) Role-based Adaptive Message Passing. (II) Temporal Contextual Attention Mechanism for incorporating historical context. (III) Role-based Variational Graph Recurrent Neural Network integrates structural and temporal patterns and updates hidden states.

Methodology

Based on our empirical observations of RPT networks, we propose the Role Perceptual Augmented Temporal Graph Network (RPATGN). The goal of this framework is to construct finer-grained node representations by capturing both the structural roles of nodes within a snapshot and the dynamic evolution of the graph structure across time. It takes a series of spatio-temporal graphs as input and ultimately predicts links in future snapshots. The overview of the model framework is illustrated in Figure 2.

Structural Role Discovery

As established by preliminary analysis, RPT networks exhibit a distinct core-periphery structure. In such topologies, node degree serves as a simple yet effective proxy for structural features. For each temporal snapshot of the graphs $G^t = (V^t, E^t)$, a node $v \in V^t$ is assigned a structural role by comparing its degree $d(v)$ against a predefined threshold θ. The process is formalized through a mapping function Φ that assigns each node to a role in the set $\mathcal{S} = \{s_1, s_2\}$:

$$\Phi(v) = \begin{cases} s_1 & d(v) > \theta \\ s_2 & d(v) \leq \theta \end{cases} \tag{2}$$

where s_1, s_2 represent central and peripheral nodes respectively, and θ can be determined empirically (e.g., using a percentile of the degree distribution) or based on domain-specific knowledge. The definition of structural roles is flexible and can employ various mapping functions according to the context and specific application. This procedure yields a role vector $\mathbf{R}^t = \{\Phi(v) | v \in V^t\}$ for each snapshot.

Role-based Adaptive Message Passing

Based on the structural roles identified previously, a role-based adaptive message passing mechanism is designed to make the information aggregation process sensitive to the specific function of each node, allowing for a finer-grained representation and better capture of varying structural properties. The process consists of two key steps: role-gated message aggregation and role-specific node updates.

Role-gated Message Aggregation. The message function uses a role-based gating mechanism to control the message vector from a node's neighborhood. The gating mechanism emphasizes or diminishes the amount of propagated information depending on the composition of the neighbors' roles. It is learned by model training, optimizing the coefficients according to the contribution of message vectors from nodes with different roles to the prediction performance. The gating signal can be defined as:

$$g(v) = \sigma(\mathbf{R}^t_{\mathcal{N}(v)} \mathbf{W}_g + \mathbf{b}_g). \tag{3}$$

Here, $\mathbf{R}^t_{\mathcal{N}(v)}$ is the matrix of role distributions for the neighbors of node v, $\mathbf{W}_g$ and $\mathbf{b}_g$ are learnable weight matrix and bias. The message passing and aggregation function with the gating mechanism can be expressed as:

$$\mathbf{m}_v = g(v) \cdot \sum_{u \in \mathcal{N}(v)} \frac{1}{\sqrt{d(u)d(v)}} \mathbf{x}_u \mathbf{W}. \tag{4}$$

where $\mathbf{x}_u$ represents the hidden state (or feature) of node u, $\mathbf{W}$ is a learnable weight matrix, and $\frac{1}{\sqrt{d(u)d(v)}}$ is a normalization coefficient to balance the influence of node degrees.

Role-specific Node Update. After completing the message aggregation, node v updates its node representation:

$$\tilde{\mathbf{x}}_v = \sigma(\mathbf{W}_{\Phi(v)} \cdot \mathbf{x}_v + \mathbf{m}_v). \tag{5}$$

$\mathbf{W}_{\Phi(v)}$ is the role-specific weights to combine the features and aggregated information based on the node's role. This step ensures that nodes with different structural functions also have distinct feature transformation capabilities.

Based on the above, the role-based graph convolution process in matrix form can be concluded as:

$$\varphi(\mathbf{A},\mathbf{X},\mathbf{R}) = \mathbf{W}_{\Phi}\mathbf{X} + \tilde{\mathbf{D}}^{-1/2}\tilde{\mathbf{A}}_g\tilde{\mathbf{D}}^{-1/2}\mathbf{X}\mathbf{W}, \quad (6)$$

where $\tilde{\mathbf{A}}_g = g \odot \tilde{\mathbf{A}}$ is the adjacency matrix adding self-loops with gating signals, $\odot$ denotes element-wise multiplication, $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$, and $\tilde{\mathbf{D}}$ is the degree matrix for the graph. Note that in this formula, $\mathbf{A}$, $\mathbf{X}$, and $\mathbf{R}$ represent an adjacency matrix, a feature vector, and a role vector respectively.

Temporal Contextual Attention Mechanism

In dynamic graphs, node representations need to maintain structural proximity and capture temporal evolution. To achieve this, we first record historical snapshots of the network and utilize a temporal attention layer to assign higher weights to more relevant historical information. This component computes the influence weights of each historical snapshot on the current snapshot, integrating temporal information into node representations, thereby relieving the limitation of ordered modeling of RNNs (Cui et al. 2019).

Let $\mathbf{H} = [\mathbf{h}^{t-w}; \mathbf{h}^{t-w+1}; ...; \mathbf{h}^{t-1}]$ be the context matrix consisting of recent w hidden states, where w is the window width of the context. We calculate attention weights using a fully connected layer and the tanh activation function:

$$\mathbf{e} = \mathbf{r}^{\top} \tanh\left(\mathbf{Q}\mathbf{H}^{\top}\right), \quad (7)$$

where $\mathbf{r} \in \mathbb{R}^d$, $\mathbf{Q} \in \mathbb{R}^{d\times d}$ are the weight vector and matrix for attention, d is the dimension of the hidden states. Then, the softmax function is used to normalize attention weights:

$$\mathbf{a}_i = \frac{\exp(\mathbf{e}_i)}{\sum_{j=1}^{w}\exp(\mathbf{e}_j)}. \quad (8)$$

Here, $\mathbf{a}_i$ is the weight of the i-th historical snapshot. We obtain the final weighted historical representation, which encapsulates temporal evolution information:

$$\tilde{\mathbf{h}}^{t-1} = (\mathbf{a}\mathbf{H})^{\top}. \quad (9)$$

Role-based Variational Graph Recurrent Neural Network

To model the temporal evolution, we employ a role-based Variational Graph Recurrent Neural Network (VGRNN), building upon the approach in (Hajiramezanali et al. 2019). The architecture is uniquely suited for our task as it combines the sequential modeling power of RNNs with the structural learning capability of GNNs, all within a probabilistic framework that can handle the inherent uncertainty of financial data. Using this framework, we integrate structural role information by conditioning the VGRNN's graph encoder. In this way, it can fully capture the complex dependencies between topological evolution and time-varying node attributes, as well as the variations of structural roles.

First, using the weighted historical representation $\tilde{\mathbf{h}}^{t-1}$ after passing through fully connected layers, we derive the prior distribution, which follows a normal distribution N:

$$p(\mathbf{Z}^t|\mathbf{A}^{<t},\mathbf{X}^{<t}) = \prod_{i=1}^{n} N(\mu_{i,\text{prior}}^t, \text{diag}((\sigma_{i,\text{prior}}^t)^2), \quad (10)$$

where μ_{prior}^t, σ_{prior}^t are the parameters of the conditional prior distribution, representing the mean and variance respectively. n denotes the number of nodes. Then, we utilize the defined φ in Eq.(6) as the encoder, yielding the parameters of the approximated posterior μ_{enc}^t and σ_{enc}^t:

$$\mu_{\text{enc}}^t = \varphi_{\mu}([\mathbf{X}^t \parallel \tilde{\mathbf{h}}^{t-1}], \mathbf{A}^t, \mathbf{R}^t), \quad (11)$$

$$\sigma_{\text{enc}}^t = \varphi_{\sigma}([\mathbf{X}^t \parallel \tilde{\mathbf{h}}^{t-1}], \mathbf{A}^t, \mathbf{R}^t), \quad (12)$$

where $\parallel$ denotes the concatenation operation. And we can derive the posterior distribution of $\mathbf{Z}^t$:

$$q(\mathbf{Z}^t|\mathbf{A}^t,\mathbf{X}^t,\tilde{\mathbf{h}}^{t-1}) = \prod_{i=1}^{n} N(\mu_{i,\text{enc}}^t, \text{diag}((\sigma_{i,\text{enc}}^t)^2)). \quad (13)$$

We update the hidden states using the recurrence equation using a non-probabilistic deep neural network f that integrates GCN and RNNs, where the deep layers in the recursive network are replaced by graph convolutional layers:

$$\mathbf{h}^t = f(\mathbf{A}^t, [\mathbf{X}^t \parallel \mathbf{Z}^t], \mathbf{R}^t, \tilde{\mathbf{h}}^{t-1}). \quad (14)$$

Here, we also adopt a role-based message passing mechanism in the message passing module of the graph convolutional layer. This allows us to capture the variations in node roles and the dynamic changes in the graph structure.

Loss Function and Optimization

The loss function consists of two parts: the reconstruction loss and the Kullback-Leibler divergence (Kingma and Welling 2013) between the prior and posterior. Since link prediction is generally seen as a binary classification task, a common form of reconstruction loss is based on binary cross-entropy. The joint loss function $\mathcal{L}$ is formulated as:

$$\mathcal{L} = \sum_{t=1}^{T}\sum_{(i,j)\in E^t}\left[\mathbf{A}_{ij}^t \log \hat{\mathbf{A}}_{ij}^t + (1-\mathbf{A}_{ij}^t)\log(1-\hat{\mathbf{A}}_{ij}^t)\right] - KL(q\left(\mathbf{Z}^t|\mathbf{A}^{\le t},\mathbf{X}^{\le t},\mathbf{Z}^{<t}\right)||p\left(\mathbf{Z}^t \mid \mathbf{A}^{<t},\mathbf{X}^{<t},\mathbf{Z}^{<t}\right)). \quad (15)$$

Here, E^t denotes all node pairs at time step t. The model can be optimized through standard stochastic gradient descent-based methods. We use a default Adam optimizer (Kingma and Ba 2014) with the learning rate of 10^{-2} and the weight decay of 10^{-5} to execute the optimization process.

Complexity Analysis

We analyze the time complexity of the main components in each timestamp. Let $|V|$ be the number of nodes, $|E|$ the number of edges, d the feature dimension, and w the context window size. In the structural role discovery module, calculating node degrees and assigning roles has a complexity of $O(|V| + |E|)$. For the role-based adaptive message passing module, computing gating signals is $O(|V| \cdot d)$, and message aggregation and node update are $O(|E| \cdot d)$. The temporal contextual attention mechanism involves constructing the context matrix $O(w \cdot |V| \cdot d)$ and computing attention weights $O(w \cdot |V| \cdot d^2)$, resulting in a total complexity of

$O(w \cdot |V| \cdot d^2)$. Lastly, the role-based variational graph recurrent Neural Network has complexities of $O(|V| \cdot d^2)$ for computing distribution parameters, encoding node representations, and updating hidden states. The overall complexity for each timestamp is $O(max(w \cdot |V| \cdot d^2, |E| \cdot d))$.

Experiment

In this section, we conduct experiments to answer the following four research questions:

- **RQ1**: How is the overall performance of RPATGN compared with state-of-the-art methods?
- **RQ2**: What is the individual contribution of the various components within RPATGN to its overall performance?
- **RQ3**: How do hyper-parameter configurations affect the performance of RPATGN?
- **RQ4**: What qualitative evidence demonstrates that RPATGN's ability to capture the evolution of node roles in real-world business scenarios?

Experimental Settings

Datasets. We conducted experiments on four real-world financial transaction networks that commonly exhibit core-periphery structure patterns. Table 1 shows detailed statistics of the datasets. Specifically, the RPT dataset was collected from regularly disclosed RPT data in China's nationwide financial market, spanning from January 2015 to December 2021. It includes profiles of over 5,000 listed companies and more than 30,000 related parties, RPT operation information, and financial exchanges involved in these transactions. It includes profiles of over 5,000 listed companies and more than 30,000 related parties. After data cleaning, we obtained over 670,000 high-quality RPT records. We constructed a dynamic graph where nodes represent entities and an edge at a given time indicates an RPT. We are working closely with our partners on data anonymization and open-sourcing, aiming to contribute this valuable industry-level dataset to the academic community and promote research on RPT. In addition, we selected three other open-source datasets for our experiments: Elliptic, Bitcoin OTC, and Bitcoin Alpha. For all datasets, we partitioned the data into a sequence of graph snapshots at quarterly intervals.

Baselines. We compare the performance of our proposed method with the state-of-the-art baselines, including two advanced static network embedding models GAE (Kipf and Welling 2016) and VGAE (Kipf and Welling 2016) and several temporal graph learning models DySAT (Sankar et al. 2020), EvolveGCN (Pareja et al. 2020), GRUGCN (Seo et al. 2018), VGRNN (Hajiramezanali et al. 2019), HTGN (Yang et al. 2021), DGCN (Gao et al. 2022), HGWaveNet (Bai et al. 2023), HMPTGN (Le and Ta 2024), RTGCN (Du et al. 2024), Hawkes-GNN (Qi et al. 2025). The detailed description of the datasets and baseline method is summarized in the supplementary materials at GitHub.

Evaluations. We obtain node representations that can be applied to various downstream tasks. Our goal is to predict future RPT, focusing on the task of dynamic graph link prediction. Typically, the task involves training on a given sequence of k snapshots and predicting the link in the next or multiple future snapshots. We choose the first 50% snapshots as the training set and the rest of the snapshots as the test set. For static models, we use all the edges in the training shots for training and the remaining as the test set. We employ two common evaluation metrics: area under the ROC curve (AUC) and average precision (AP) to comprehensively assess the model's ability to correctly classify true and false edges. Following previous work (Yang et al. 2021), we label all known edges in the test snapshots as true and sample an equal number of non-links as false.

Dataset	#Nodes	#Edges	#Snapshots
RPT	37,651	675,935	28
Elliptic	203,769	234,355	49
Bitcoin OTC	5,881	35,592	20
Bitcoin Alpha	3,783	24,186	20

Table 1: Statistics including the number of nodes, edges, and snapshots of the selected dataset.

Parameter Settings. In our implementation, we conducted the experiment five times and reported the average results. For datasets without node features, we use one-hot encoding as the input feature. For all baselines, we take the recommended parameter settings unless otherwise noted. For our proposed RPATGN, we set the embeddings dimension as 128 for RPT, Elliptic datasets and 32 for Bitcoin-OTC, Bitcoin-Alpha. We uniformly set the historical snapshot recording window to 3. During training, we used early stopping with patience of 50 and a maximum of 200 iterations to prevent overfitting and optimize performance. Our method is implemented using PyTorch 1.12.1 with CUDA 11.3 and Python 3.7 as the backend. The model is trained on a server with four 32GB NVIDIA Tesla V100 GPUs.

Overall Performance

Table 2 presents the main experimental results across all four real-world datasets. The findings demonstrate that our proposed RPATGN consistently outperforms all baseline methods. The improvements are statistically significant, as validated by t-tests. On average, RPATGN achieves a relative gain of 3.7% in AP over the strongest baseline, showcasing its superior ability to capture network dynamics and learn expressive node representations. The strong performance across diverse financial networks, including RPTs, blockchain, and bitcoin transactions, underscores the broad applicability and effectiveness of our approach. More specifically, our RPT dataset contains over 600,000 transaction records involving over 748 billion USD. A mere 1% improvement can translate to detecting transactions worth over 7.48 billion USD, showcasing the immense potential of our method for enhancing financial regulation. Additionally, it is worth mentioning that compared to RTGCN's high memory use due to constructing structural role-based hypergraphs,

Dataset	RPT		Elliptic		Bitcoin OTC		Bitcoin Alpha	
Metric	AUC	AP	AUC	AP	AUC	AP	AUC	AP
GAE	77.65 ± 0.62	83.50 ± 0.38	81.77 ± 1.91	81.22 ± 1.90	83.20 ± 0.98	86.05 ± 0.71	85.36 ± 2.06	88.74 ± 1.56
VGAE	81.86 ± 0.08	84.59 ± 0.29	78.35 ± 0.23	74.68 ± 0.22	81.50 ± 0.72	85.98 ± 0.59	84.11 ± 1.33	87.46 ± 1.12
DySAT	OOM	OOM	OOM	OOM	85.67 ± 0.09	83.23 ± 0.62	84.32 ± 0.28	82.21 ± 0.35
EvolveGCN	OOM	OOM	OOM	OOM	83.68 ± 2.98	86.27 ± 2.83	85.01 ± 1.23	87.06 ± 1.65
GRUGCN	83.27 ± 0.07	83.73 ± 0.31	82.39 ± 0.22	85.72 ± 0.33	86.66 ± 1.90	86.32 ± 1.57	88.72 ± 2.59	87.68 ± 3.79
VGRNN	84.56 ± 0.11	85.15 ± 0.23	79.40 ± 0.32	83.59 ± 1.94	85.67 ± 0.31	86.95 ± 0.33	88.08 ± 0.60	89.30 ± 0.05
HTGN	85.93 ± 0.06	80.48 ± 0.04	80.08 ± 0.11	86.24 ± 0.06	88.67 ± 0.28	79.99 ± 0.14	92.23 ± 0.33	86.68 ± 0.36
DGCN	86.90 ± 0.12	85.46 ± 0.20	84.75 ± 0.03	83.49 ± 0.04	88.58 ± 0.03	87.68 ± 0.01	86.46 ± 0.04	84.88 ± 0.01
HGWaveNet	86.63 ± 0.04	82.05 ± 0.22	81.22 ± 0.18	73.09 ± 0.37	87.18 ± 0.03	79.05 ± 0.03	87.27 ± 0.15	84.69 ± 0.51
HMPTGN	86.27 ± 0.06	85.09 ± 0.01	84.04 ± 0.03	77.97 ± 0.03	85.91 ± 0.21	78.02 ± 0.31	92.09 ± 0.01	86.91 ± 0.01
RTGCN	OOM	OOM	OOM	OOM	87.66 ± 0.20	81.77 ± 0.18	89.27 ± 0.29	89.50 ± 0.43
Hawkes-GNN	83.21 ± 0.21	83.37 ± 0.11	79.05 ± 0.46	77.43 ± 0.80	84.59 ± 0.21	85.53 ± 0.11	83.76 ± 0.43	82.90 ± 2.42
RPATGN(Ours)	**87.62 ± 0.07**	**88.76 ± 0.17**	**85.89 ± 0.12**	**87.43 ± 0.24**	**91.59 ± 0.33**	**92.11 ± 0.30**	**93.12 ± 0.14**	**93.37 ± 0.09**

Table 2: AUC and AP scores (% ± standard deviation) of dynamic link prediction task on real-world datasets. OOM: Out Of Memory. The **bold** results represent the best methods for each dataset, and the underline results represent the second-best.

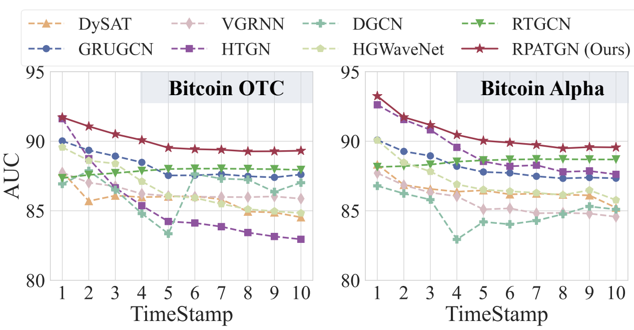


Figure 3: AUC of link prediction at each test snapshot on the Bitcoin OTC and Bitcoin Alpha datasets.

our method modifies message passing based on structural roles, enhancing structure representation and boosting efficiency on large-scale graphs.

To evaluate the model's long-term and short-term prediction capabilities, we compared its performance with selected baseline models across different test snapshots over time. As illustrated in Figure 3, some competitive baselines, such as HTGN, perform well initially but show a noticeable performance decline as they predict over the long term. In contrast, our RPATGN demonstrates exceptional stability, maintaining a consistently high performance margin across nearly all snapshots. This highlights our model's robust ability to capture both long-term and short-term dynamics, a substantial advantage for reliable dynamic link prediction tasks.

Ablation Study

We conduct ablation experiments to validate the effectiveness of the proposed model's main components. We further investigate three variants of RPATGN, which are w/o RAMP, w/o TCAM, and w/o RVGRNN, removing the role-based adaptive message passing mechanism, the temporal contextual attention mechanism, and the role-based variational graph recurrent neural network respectively from the original model. Figure 4 presents the results of the ablation study, showing that performance decreases when each component is removed. Notably, the most significant performance drop occurs in the w/o RAMP variant, especially on the Elliptic dataset. This is because the Elliptic network exhibits a prominent core-periphery structure, where the role distinction in local structures is pronounced. It suggests that the customized aggregation strategy truly counts in capturing structural role differences, significantly impacting the model's performance.

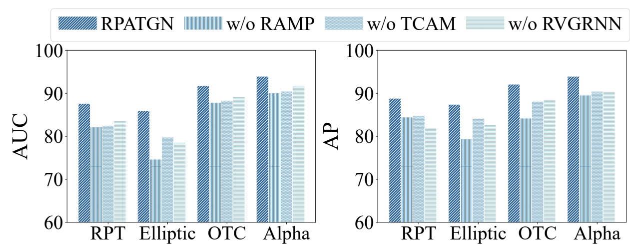


Figure 4: Ablation studies for different RPATGN framework variants, in terms of AUC and AP.

Parameter Sensitivity Experiment

We further investigate the model generalization performance on hyper-parameters of varying embedding dimension d from 16 to 256 and historical window size w from 1 to 10. As illustrated in Figure 5, the AUC score is consistently higher at d =32 and d =64. Smaller dimensions capture fine-grained, local relationships but may miss sufficient detail, especially in larger snapshot windows. Larger dimensions can overfit, capturing excessive details that may not generalize well, especially in smaller datasets. Furthermore, the model is relatively stable across different w, with slight improvements around w =5. Larger windows including more historical information can help capture long-term trends but may introduce noise and redundancy, potentially degrading performance in ever-changing scenarios.

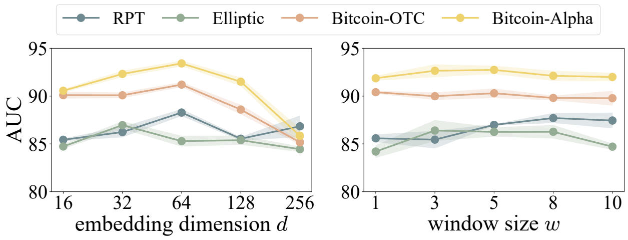


Figure 5: AUC scores of varying hyperparameter.

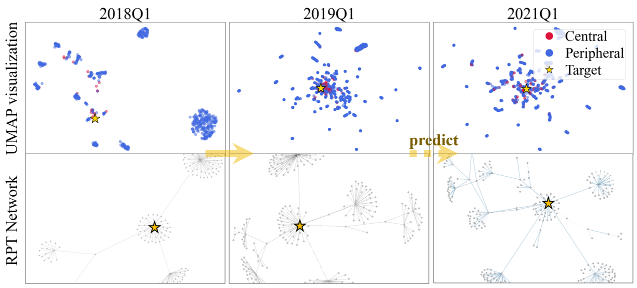


Figure 6: Visualizing the evolution of RPT networks and highlighting the trajectory of company A, where the edge color denotes the predicted probability.

Case Study

To demonstrate RPATGN's ability to capture real-world business dynamics, we present a case study on a single entity, Company A. Figure 6 displays the UMAP projection of the node embeddings around Company A and the corresponding network structure from 2018 to 2021. According to public records, Company A underwent a period of expansion between 2018 and 2019, establishing numerous new RPTs with other large companies (central nodes in the network). In this way, it actively formed the very core-periphery structures our initial analysis identified as the dominant topological pattern in these networks. By 2021, Company A's expansion and strategic repositioning had stabilized. The predicted result of our model in 2021 Q1 accurately reflects the stabilization of Company A's RPT. By capturing that Company A was transitioning into a more central role, RPATGN could accurately predict its new, stable set of RPTs with other major companies in 2021. It dynamically models the change in the company's functional influence by observing its interactions with nodes of different structural roles. This case vividly demonstrates RPATGN's capacity in identifying significant links and predicting the evolution of RPT network, which could provide a powerful tool for forward-looking regulatory analysis in a complex and dynamic financial context. More details are provided in the supplementary files.

Related Works

Graph Learning for Temporal Networks

Many real-world applications involve dynamic graphs where both structure and inputs change over time. Spatio-temporal Graph Neural Networks are crucial for capturing this dynamism (Cheng et al. 2020a; Xiang et al. 2023; Zhu et al. 2025). They integrate graph convolution for spatial dependencies and use RNNs or CNNs for temporal dependencies (Yu, Yin, and Zhu 2017; Yan, Xiong, and Lin 2018; You, Du, and Leskovec 2022). Building upon this, recent works incorporate additional techniques like hyperbolic graph neural network (Yang et al. 2021; Bai et al. 2023), causal inference (Hu et al. 2023; Zhao and Zhang 2024), graph heterogeneity (Jiang et al. 2023; Zhou et al. 2023), motif-based learning (Liu and Sariyüce 2023; Zhang et al. 2023), topological data analysis (Shamsi et al. 2024), etc. While effective at capturing temporal evolution, these methods fundamentally lack role-perceptual capabilities. It renders them unable to model either the collective behaviors of same-role nodes or the conditional logic of interactions between different roles.

Role-based Representation Learning

Intuitively, roles represent the main node-level connectivity patterns within their neighborhoods (Rossi and Ahmed 2014). Role-based representation learning methods have been proposed to capture structural similarity, demonstrating that structural roles can enhance the quality of node embeddings (Ahmed et al. 2020; Zhang et al. 2021; Sankar et al. 2022; Cheng et al. 2023). RMNE (Zhang and Kou 2022) further extends role-based random walk methods to multiplex networks. Recently, (Lee et al. 2024) found that global structural features can be extracted using the node roles. RTGCN (Du et al. 2024) leverages structural role information on temporal graphs to capture graph evolving patterns. However, these methods treat roles either as static attributes or auxiliary features by bridging structurally similar nodes. They apply unified operations for information aggregation and evolution to functionally distinct roles, thus failing to differentially model diverse interactions.

In contrast, we integrate role-based patterns and adopt customized aggregation strategies, where message passing is dynamically conditioned on the structural roles. The dual focus on role-specific aggregation and dynamic adaptation allows for a more nuanced modeling of node functions and enhances the overall perception of the complex network.

Conclusion

In this paper, we uncover the core-periphery structure in RPT networks and analyze the structural role patterns. We validate the impact of different structural roles on the aggregation and propagation of node information. Based on the insights, we propose the RPATGN, which tailors message passing strategies for different structural roles, enhancing the perception of both local and global graph structures. By integrating a temporal contextual attention mechanism and variational recurrent graph neural networks, our model strengthens the ability to capture role transition and graph dynamics. This enables accurate identification of potential RPT behaviors in complex and dynamic financial data scenarios. Extensive experiments demonstrate the effectiveness of RPATGN. The work provides a powerful tool for proactive financial supervision and helps to formulate regulatory policies and risk prevention strategies. It has positive practical significance and value for the financial system and contributes to a more stable and transparent financial market.

Acknowledgments

This work was supported by the National Science Fundation of China (Grant no. 62472317), the Fundamental Research Funds for the Central Universities, the Shanghai Science and Technology Innovation Action Plan Project (Grant no. 22YS1400600).

References

Ahmed, N. K.; Rossi, R. A.; Lee, J. B.; Willke, T. L.; Zhou, R.; Kong, X.; and Eldardiry, H. 2020. Role-based graph embeddings. *IEEE Transactions on Knowledge and Data Engineering*, 34(5): 2401–2415.

Bai, Q.; Nie, C.; Zhang, H.; Zhao, D.; and Yuan, X. 2023. Hgwavenet: A hyperbolic graph neural network for temporal link prediction. In *Proceedings of the ACM Web Conference*, 523–532.

Barabási, A.; and PÃ3sfai, M. 2016. *Network Science*. Cambridge University Press. ISBN 9781107076266.

Bratton, W. W. 2001. Enron and the dark side of shareholder value. *Tul. L. Rev.*, 76: 1275.

Cheng, D.; Tu, Y.; Niu, Z.; and Zhang, L. 2018. Learning temporal relationships between financial signals. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2641–2645. IEEE.

Cheng, D.; Wang, X.; Zhang, Y.; and Zhang, L. 2020a. Graph neural network for fraud detection via spatial-temporal attention. *IEEE Transactions on Knowledge and Data Engineering*, 34(8): 3800–3813.

Cheng, D.; Xiang, S.; Shang, C.; Zhang, Y.; Yang, F.; and Zhang, L. 2020b. Spatio-temporal attention-based neural network for credit card fraud detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 362–369.

Cheng, D.; Ye, Y.; Xiang, S.; Ma, Z.; Zhang, Y.; and Jiang, C. 2023. Anti-money laundering by group-aware deep graph learning. *IEEE Transactions on Knowledge and Data Engineering*, 35(12): 12444–12457.

Cheng, D.; Zou, Y.; Xiang, S.; and Jiang, C. 2025. Graph neural networks for financial fraud detection: a review. *Frontiers of Computer Science*, 19(9): 199609.

Cui, Q.; Wu, S.; Huang, Y.; and Wang, L. 2019. A hierarchical contextual attention-based network for sequential recommendation. *Neurocomputing*, 358: 141–149.

Dou, H.; Liu, Y.; Shi, Y.; and Xu, H. 2022. Are related-party transactions beneficial or detrimental in emerging markets? New evidence of financial services agreements from China. *International Review of Financial Analysis*, 81: 102144.

Du, H.; Shi, L.; Chen, X.; Zhao, Y.; Zhang, H.; Yang, C.; Zhuang, F.; and Kou, G. 2024. Representation Learning of Temporal Graphs with Structural Roles. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 654–665.

Gao, C.; Zhu, J.; Zhang, F.; Wang, Z.; and Li, X. 2022. A novel representation learning for dynamic graphs based on graph convolutional networks. *IEEE Transactions on Cybernetics*, 53(6): 3599–3612.

Hajiramezanali, E.; Hasanzadeh, A.; Narayanan, K.; Duffield, N.; Zhou, M.; and Qian, X. 2019. Variational graph recurrent neural networks. *Advances in neural information processing systems*, 32.

Han, L.; Wang, L.; Cheng, Z.; Wang, B.; Yang, G.; Cheng, D.; and Lin, X. 2025. Mitigating the tail effect in fraud detection by community enhanced multi-relation graph neural networks. *IEEE Transactions on Knowledge and Data Engineering*.

Hope, O.-K.; and Lu, H. 2020. Economic consequences of corporate governance disclosure: Evidence from the 2006 SEC regulation on related-party transactions. *The Accounting Review*, 95(4): 263–290.

Hu, J.; Liang, Y.; Fan, Z.; Chen, H.; Zheng, Y.; and Zimmermann, R. 2023. Graph Neural Processes for Spatio-Temporal Extrapolation. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 752–763.

Hu, Z.; Dong, Y.; Wang, K.; and Sun, Y. 2020. Heterogeneous graph transformer. In *Proceedings of the web conference*, 2704–2710.

Jiang, R.; Wang, Z.; Yong, J.; Jeph, P.; Chen, Q.; Kobayashi, Y.; Song, X.; Fukushima, S.; and Suzumura, T. 2023. Spatio-temporal meta-graph learning for traffic forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, 8078–8086.

Jin, Y.; Song, G.; and Shi, C. 2020. Gralsp: Graph neural networks with local structural patterns. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 4361–4368.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kingma, D. P.; and Welling, M. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Kipf, T. N.; and Welling, M. 2016. Variational graph auto-encoders. *arXiv preprint arXiv:1611.07308*.

Kohlbeck, M.; and Mayhew, B. W. 2010. Valuation of firms that disclose related party transactions. *Journal of Accounting and Public Policy*, 29(2): 115–137.

Kohlbeck, M.; and Mayhew, B. W. 2017. Are related party transactions red flags? *Contemporary Accounting Research*, 34(2): 900–928.

Le, V. Q.; and Ta, V. C. 2024. Toward a manifold-preserving temporal graph network in hyperbolic space. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, 4380–4388.

Lee, O.-J.; et al. 2024. Transitivity-Preserving Graph Representation Learning for Bridging Local Connectivity and Role-Based Similarity. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 12456–12465.

Liu, P.; and Sariyüce, A. E. 2023. Using motif transitions for temporal graph generation. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 1501–1511.

Liu, X.; Rui, H.; Cheng, D.; Han, L.; Zhou, Z.; and Zhao, G. 2025. Multi-Granularity Augmented Graph Learning for

Spoofing Transaction Detection. In *Proceedings of the ACM on Web Conference*, 5151–5160.

Mao, X.; Sun, H.; Zhu, X.; and Li, J. 2022. Financial fraud detection using the related-party transaction knowledge graph. *Procedia Computer Science*, 199: 733–740.

Muthupandian, K. 2009. IAS 24, Related Party Disclosures-A Closer Look.

Pareja, A.; Domeniconi, G.; Chen, J.; Ma, T.; Suzumura, T.; Kanezashi, H.; Kaler, T.; Schardl, T.; and Leiserson, C. 2020. Evolvegcn: Evolving graph convolutional networks for dynamic graphs. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 5363–5370.

Qi, Q.; Chen, H.; Cheng, M.; and Liu, H. 2025. Input Snapshots Fusion for Scalable Discrete-Time Dynamic Graph Neural Networks. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 1138–1149.

Rossi, R. A.; and Ahmed, N. K. 2014. Role discovery in networks. *IEEE Transactions on Knowledge and Data Engineering*, 27(4): 1112–1131.

Ryngaert, M.; and Thomas, S. 2012. Not all related party transactions (RPTs) are the same: Ex ante versus ex post RPTs. *Journal of Accounting Research*, 50(3): 845–882.

Sankar, A.; Wang, J.; Krishnan, A.; and Sundaram, H. 2022. Self-supervised role learning for graph neural networks. *Knowledge and Information Systems*, 64(8): 2091–2121.

Sankar, A.; Wu, Y.; Gou, L.; Zhang, W.; and Yang, H. 2020. Dysat: Deep neural representation learning on dynamic graphs via self-attention networks. In *Proceedings of the 13th international conference on web search and data mining*, 519–527.

Seo, Y.; Defferrard, M.; Vandergheynst, P.; and Bresson, X. 2018. Structured sequence modeling with graph convolutional recurrent networks. In *International conference on neural information processing*, 362–373. Springer.

Shamsi, K.; Poursafaei, F.; Huang, S.; Ngo, B. T. G.; Coskunuzer, B.; and Akcora, C. G. 2024. GraphPulse: Topological representations for temporal graph property prediction. In *The Twelfth International Conference on Learning Representations*.

Skarding, J.; Gabrys, B.; and Musial, K. 2021. Foundations and modeling of dynamic networks using dynamic graph neural networks: A survey. *iEEE Access*, 9: 79143–79168.

Wang, Y.; Lipka, N.; Zhang, R.; Siu, A.; Zhao, Y.; Ni, B.; Wang, X.; Rossi, R.; and Derr, T. 2024. Topology-aware retrieval augmentation for text generation. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, 2442–2452.

Xiang, S.; Cheng, D.; Shang, C.; Zhang, Y.; and Liang, Y. 2022. Temporal and heterogeneous graph neural network for financial time series prediction. In *Proceedings of the 31st ACM international conference on information & knowledge management*, 3584–3593.

Xiang, S.; Zhang, G.; Cheng, D.; and Zhang, Y. 2025. Enhancing Attribute-Driven Fraud Detection With Risk-Aware Graph Representation. *IEEE Transactions on Knowledge and Data Engineering*.

Xiang, S.; Zhu, M.; Cheng, D.; Li, E.; Zhao, R.; Ouyang, Y.; Chen, L.; and Zheng, Y. 2023. Semi-supervised credit card fraud detection via attribute-driven graph representation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 14557–14565.

Xie, Y.; Li, S.; Yang, C.; Wong, R. C.-W.; and Han, J. 2020. When do gnns work: Understanding and improving neighborhood aggregation. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, volume 2020.

Yan, S.; Xiong, Y.; and Lin, D. 2018. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Yang, M.; Zhou, M.; Kalander, M.; Huang, Z.; and King, I. 2021. Discrete-time temporal network embedding via implicit hierarchical learning in hyperbolic space. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 1975–1985.

You, J.; Du, T.; and Leskovec, J. 2022. ROLAND: graph learning framework for dynamic graphs. In *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, 2358–2366.

Yu, B.; Yin, H.; and Zhu, Z. 2017. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv preprint arXiv:1709.04875*.

Zhang, G.; Ye, T.; Jin, D.; and Li, Y. 2023. An attentional multi-scale co-evolving model for dynamic link prediction. In *Proceedings of the ACM Web Conference*, 429–437.

Zhang, H.; and Kou, G. 2022. Role-based multiplex network embedding. In *International Conference on Machine Learning*, 26265–26280. PMLR.

Zhang, W.; Guo, X.; Wang, W.; Tian, Q.; Pan, L.; and Jiao, P. 2021. Role-based network embedding via structural features reconstruction with degree-regularized constraint. *Knowledge-Based Systems*, 218: 106872.

Zhao, K.; and Zhang, L. 2024. Causality-Inspired Spatial-Temporal Explanations for Dynamic Graph Neural Networks. In *The Twelfth International Conference on Learning Representations*.

Zheng, W.; Xu, B.; Lu, E.; Li, Y.; Cao, Q.; Zong, X.; and Shen, H. 2023. MIDLG: Mutual Information based Dual Level GNN for Transaction Fraud Complaint Verification. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 5685–5694.

Zhou, Z.; Huang, Q.; Lin, G.; Yang, K.; Bai, L.; and Wang, Y. 2023. Greto: Remedying dynamic graph topology-task discordance via target homophily. In *International Conference on Learning Representations*.

Zhu, P.; Li, Y.; Liu, Q.; Cheng, D.; and Jiang, C. 2025. Financial Time Series Prediction With Multi-Granularity Graph Augmented Learning. *IEEE Transactions on Knowledge and Data Engineering*.