

Beyond Single Transactions: D-EMAML—Dual-Edge Motif Neural Networks for Enhanced Anti-Money Laundering Detection

Dongmei Han¹⁵⁴, Min Min²⁴⁶, Yuchen Wang², Guoming Xu³, Xiaofeng Zhou^{1*}

¹School of Information Management and Engineering, Shanghai University of Finance and Economics

²School of Finance, Shanghai University of Finance and Economics

³College of Business, Shanghai University of Finance and Economics

⁴Shanghai Key Laboratory of Financial Information Technology, Shanghai University of Finance and Economics

⁵Faculty of Business Information, Shanghai Business School

⁶Shanghai University of Finance and Economics Zhejiang College

dongmeihan@shufe.edu.cn, mmin@sufe.edu.cn, wycnb@stu.sufe.edu.cn,

i@163.sufe.edu.cn, f@163.sufe.edu.cn

Abstract

Anti-money laundering (AML) detection is of vital importance in financial risk control. Although Graph Neural Networks (GNN) have yielded promising results, existing motif-based approaches primarily focus on node anomaly detection on simple graphs, which hinders the direct identification of anomalous edges in directed temporal transaction networks. Moreover, consecutive transaction relationships, termed dual-edge motifs, have rarely been considered in previous AML studies. To address these gaps, we propose the D-EMAML framework, which consists of: (1) Fast-Motif-Gen, a GPU-accelerated dual-edge motif graph generator with pruning; (2) D-EMGNN, an attention-enhanced heterogeneous GNN module that reduces motif-type information redundancy; (3) MELP, a label aggregation scheme projecting predictions from the motif graph to the original graph. Extensive experiments on real-world and synthetic datasets demonstrate significant improvements over representative baselines and validate the contribution of each component. To our knowledge, this is the first application of dual-edge motif graphs for GNN-based edge anomaly detection in AML.

Code — <https://github.com/SUFE-Finlab/D-EMAML>

Extended version — <https://paper.fin-lab.ai/paper?id=1>

1 Introduction

The escalating complexity of financial transactions has intensified the challenge of financial fraud, demanding robust detection methodologies for financial institutions and regulatory bodies (Hilal, Gadsden, and Yawney 2022; Pourhabibi et al. 2020). In response, modern Anti-Money Laundering (AML) research has prominently featured **Graph Neural Networks (GNNs)** (Hamilton, Ying, and Leskovec 2017; Kipf and Welling 2016; Xu et al. 2018), including advanced variants such as group-aware and motif-augmented models (Cheng et al. 2023; Motie and Raahemi 2024; Wu et al. 2022; Wang et al. 2023).

*Corresponding author.

Motifs, representing fundamental higher-order structures within complex networks (Milo et al. 2002), offer a powerful mechanism to abstract low-level interactions into semantically rich patterns like laundering cycles (Wang et al. 2023; Huang et al. 2021; Xiao et al. 2024). However, prevailing GNN-based approaches have critical limitations. They predominantly focus on node-centric motifs (e.g., 3- or 4-node structures) on simple directed graphs, inadequately addressing edge-level anomaly detection, which is a crucial task in transaction monitoring (Motie and Raahemi 2024). As illustrated in Figure 1, conventional node-motif methods aggregate multiple transactions, which not only fails to distinguish nuanced laundering flows but also leads to a loss of essential edge attributes. In contrast, our proposed edge-motif approach transforms individual transactions into higher-order structures. For example, in the dual-edge motif graph shown, nodes are colored by anomaly severity: black nodes represent consecutive suspicious transactions (marked in red), gray nodes represent sequences with one suspicious transaction, and white nodes represent normal sequences. This fine-grained representation effectively improves the detection of anomalous activity.

These deficiencies highlight the inadequacy of established

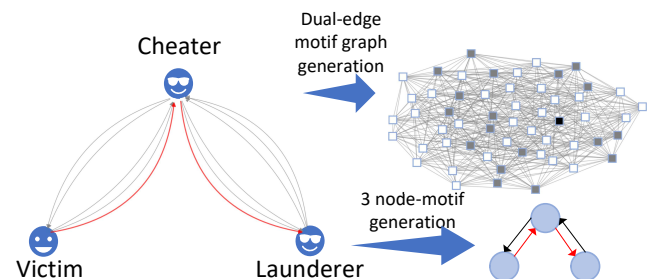


Figure 1: A comparison of node-centric motifs and our proposed dual-edge motifs. Dual-edge motifs preserve directional flow and edge attributes, enabling a more granular analysis of transaction sequences.

methods like MotifGNN (Wang et al. 2023), MADG (Yuan, Shao, and Yan 2023), and HOGAT (Huang et al. 2021). Their static, node-focused architectures are ill-suited for capturing the dynamic financial flows inherent in money laundering schemes. Thus, we introduce the concept of **dual-edge motifs**—sequential pairs of transactions that explicitly model the directional and temporal movement of funds. These structures act as a foundational “grammar” for representing sophisticated illicit tactics like structuring and layering. Despite their potential, applying such powerful, edge-centric motifs to transaction-level anomaly detection remains a largely unaddressed research frontier.

To bridge this gap, we introduce **Dual-Edge Motif Anti-Money Laundering (D-EMAML)**, a novel framework for direct edge anomaly detection. As depicted in Figure 2, our approach begins with **Fast-Motif-Gen**, an accelerated algorithm that efficiently generates and selects dual-edge motif instances using a statistics-driven pruning strategy. These instances are then used to construct a motif graph. For representation learning, we propose the **Dual-Edge Motif Graph Neural Network (D-EMGNN)**, which features a heterogeneous-bias **Motif Graph Convolutional Network (MGCN)** layer and a **Structure-Aware Information Redundancy Removal (SAIRR)** module. Finally, **Motif-to-Edge Label Propagation (MELP)** maps the learned instance-level predictions back to the original transaction edges for accurate anomaly detection.

Our main contributions are summarized as follows:

- We are the first to construct dual-edge motif graphs over multi-directed temporal networks for GNN-based edge anomaly detection, introducing D-EMAML with MELP to overcome the limitations of node-centric motifs.
- We propose D-EMGNN, which incorporates attention-based redundancy removal module (SAIRR) and heterogeneous bias mechanism (MGCN) to effectively capture motif heterogeneity. Our model achieves superior performance on both real and synthetic datasets, and ablation studies validate the effectiveness of SAIRR and MGCN.
- We design Fast-Motif-Gen, a temporal span-based generator with a pruning strategy and controlled sampling, enabling efficient and scalable motif graph construction without compromising accuracy.
- We advance AML edge analysis from single transactions to event pairs (dual-edge motifs), significantly enhancing the recognition of characteristic money laundering patterns.

2 Related Work

GNN for Transaction Laundering Detection Graph neural networks (GNNs) have shown significant promise in financial fraud detection (Motie and Raahemi 2024). Current approaches often employ techniques like graph augmentation to identify clustered illicit activities (Cheng et al. 2023) or design specialized architectures to capture directional information in transaction flows (Wu et al. 2022). However, these methods primarily focus on individual transactions or simple node-centric patterns (Cheng et al. 2023; Wu et al.

2022; Li et al. 2023; Huang et al. 2023; Egressy et al. 2024; Luo et al. 2024; Lin et al. 2023). Consequently, they neglect the richer contextual cues embedded in higher-order structural patterns, such as motifs involving sequences of multiple entities, which are crucial for understanding complex laundering schemes.

Existing Motif-based GNNs To leverage such higher-order structures, another line of research has explored motif-based GNNs. Despite their novelty, these methods are fundamentally misaligned with the specific demands of edge-level anti-money laundering (AML) detection. The majority are designed for node-level tasks, such as identifying anomalous accounts (Wang et al. 2023; Huang et al. 2021; Yuan et al. 2021; Lee et al. 2019), or graph-level classification (Xiao et al. 2024), which is too coarse to pinpoint individual illicit transactions. Other methods construct hypergraphs from motifs (Zhao et al. 2025; Long et al. 2025), but this process abstracts away the critical temporal and multi-relational details inherent in transaction flows. In essence, their static and node-centric designs render them unsuitable for the dynamic, directed, and edge-centric nature of AML, leaving a critical gap in edge-based motif anomaly detection.

In stark contrast to conventional node-centric motifs (e.g., triangles), an edge-centric design offers distinct advantages for analyzing transaction networks. By focusing on structural primitives like the **dual-edge motif**, which explicitly model **sequential transaction pairs**, such approach natively captures the **directionality, temporality, and multiplicity** characteristic of sophisticated anomalous patterns. This allows for the identification of complex behaviors that are inherently overlooked by node-centric frameworks.

3 Problem Definition

3.1 Basic Notations

We represent the transaction dataset as a **directed temporal graph** (Paranjape, Benson, and Leskovec 2017), denoted as $G = (V, \mathcal{E})$, where V is the set of nodes, and $\mathcal{E} = \{\varepsilon_i\}_{i=1}^{|\mathcal{E}|}$ is the set of transaction events. Each event $\varepsilon_i = (u_i, v_i, t_i, \chi_i, y_i)$ denotes a directed transaction from node u_i to node v_i at timestamp t_i , with $\chi_i = (\mathbf{x}_{u_i}, \mathbf{x}_{v_i}, \mathbf{x}_{uv_i})$ representing the attribute vectors of the source node, destination node, and the edge, respectively. The label $y_i \in \{0, 1\}$ indicates whether the transaction is normal ($y_i = 0$) or suspicious for money laundering ($y_i = 1$). Let $\mathbf{u}, \mathbf{v}, \mathbf{t}, \mathbf{y} \in \mathbb{R}^{|\mathcal{E}|}$ denote the vectors of source nodes, destination nodes, timestamps, and transaction labels, respectively; χ is the concatenation of vertex and edge attribute vectors. A transaction is uniquely identified by the tuple (u_i, v_i, t_i) , denoted as $\varepsilon_{uv t}$. The set $\varepsilon_{uv} = \{\varepsilon_{ij\tau} \mid i = u, j = v\}$ includes all events from u to v ; similarly, $\varepsilon_{u*} = \{\varepsilon_{ij\tau} \mid i = u\}$ and $\varepsilon_{*v} = \{\varepsilon_{ij\tau} \mid j = v\}$ denote all events originating from u and terminating at v , respectively. For an event $\varepsilon_{uv t}$, its n -hop **transaction neighbors** are denoted as $\mathcal{N}_{\varepsilon}^n(\varepsilon_{uv t})$. The set of neighbors of agent u is denoted as $\mathcal{N}(u)$.

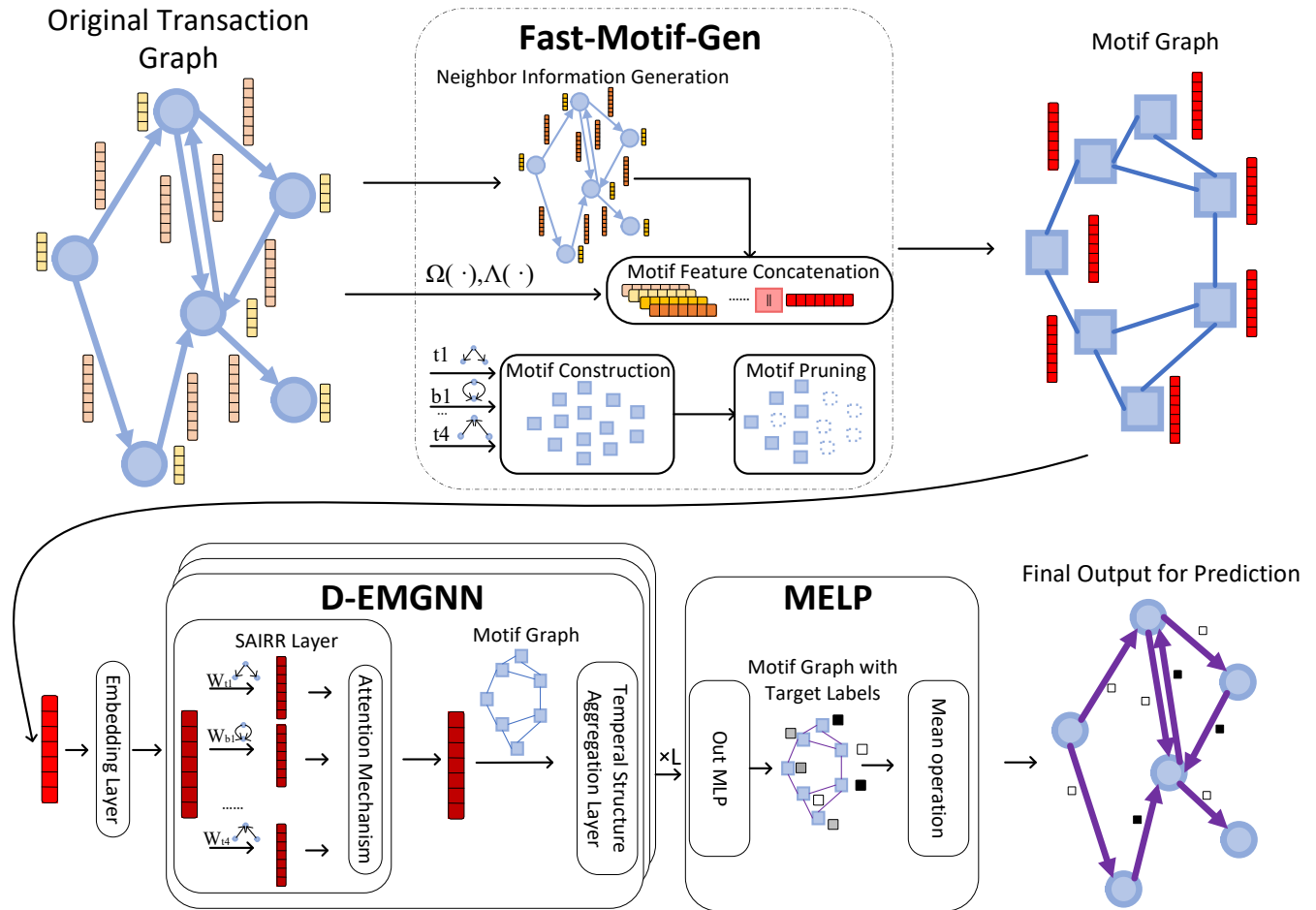


Figure 2: The overall workflow of our proposed D-EMAML framework, from motif generation and graph construction to representation learning and label propagation for edge-level anomaly detection.

3.2 Transaction Anomaly Detection

Transaction anomaly detection aims to identify events that significantly deviate from expected behavior. Given the event batch set \mathcal{E} from the whole transaction dataset and its induced neighborhood graph $G = \{\cup \mathcal{N}_\varepsilon^n(\varepsilon) \mid \varepsilon \in \mathcal{E}\}$, the task is to learn a function $\hat{\mathbf{y}} = \Phi(G)$ such that $\hat{\mathbf{y}}$ closely approximates the true label vector \mathbf{y} .

4 Methodology

Our proposed framework consists of three key components, which we detail in this section: (1) **Fast-Motif-Gen**, a systematic approach to efficiently construct a high-level motif graph from the original transaction data; (2) **D-EMGNN (Dual-Edge Motif Graph Neural Network)**, a model designed to encode the constructed motif graph using spatial attention and heterogeneous convolutions; and (3) **MELP (Motif-Edge Label Propagation)**, a module to infer labels for the original transactions based on the learned motif representations.

4.1 Fast-Motif-Gen

The core idea of Fast-Motif-Gen is to abstract the original transaction graph into a higher-level motif graph, shifting the analytical perspective from individual transactions to co-ordinated laundering tactics. In the context of AML, a single transaction might be incidental, but its significance is amplified when viewed as part of a larger pattern. By transforming laundering tactics into nodes of a new graph, a GNN can learn their interplay. For example, a model could learn that a "fan-in" pattern frequently preceding a "fan-out" pattern is a strong indicator of a sophisticated laundering scheme.

This transformation process is illustrated in Figure 3, which shows the defined motif structures and the resulting motif graph. The following sections detail the three key aspects of this process: the structure of the motif graph, an efficient generation algorithm, and a pruning mechanism for scalability.

Dual-Edge Motif Definition Examples of D-EM graph and D-EM instances are illustrated in Figure 3.

Definition 1 (Dual-Edge Motif Graph). A *motif graph* is

defined as an undirected graph $M = (\mathcal{M}, \Omega)$, where \mathcal{M} is a set of dual-edge motif instances and Ω represents the connections between them. A connection, or motif edge, exists between two motifs $m_a, m_b \in \mathcal{M}$ if and only if their constituent edge sets have a non-empty intersection, i.e., $\mathbf{E}_a \cap \mathbf{E}_b \neq \emptyset$.

Definition 2 (Dual-Edge Motif Instance). Each D-EM instance $m_i \in \mathcal{M}$ is a structured tuple encoding a specific sequential pattern of two transactions:

$$m_i = (\mathbf{V}_i, \mathbf{E}_i, \delta_i, \omega_i, \mathbf{Y}_i, \mathbf{X}_i)$$

The elements of the tuple are defined as follows:

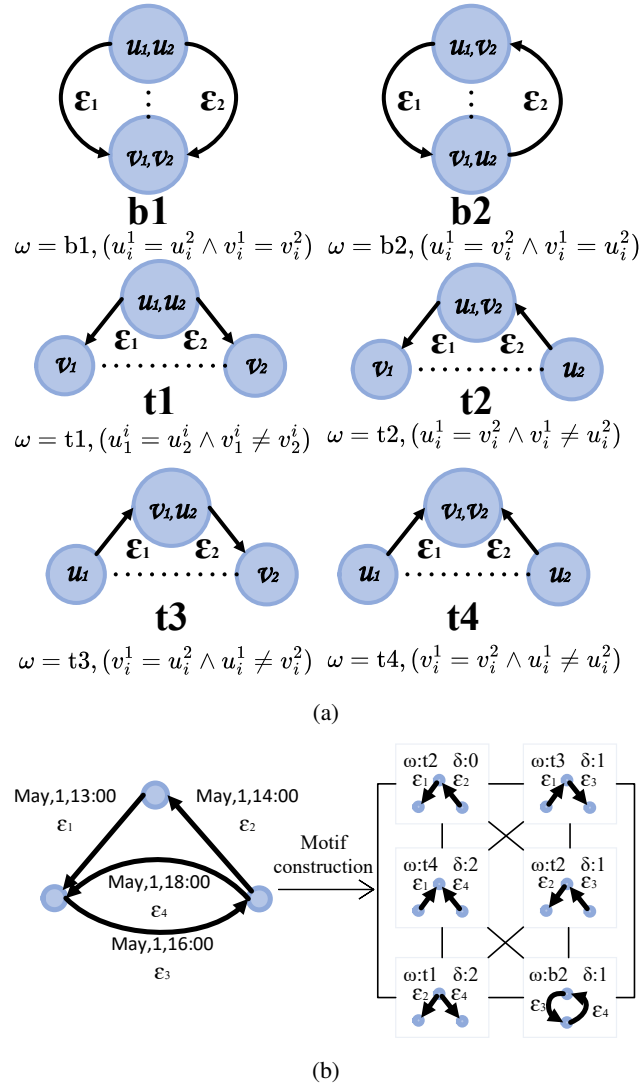


Figure 3: Overview of the motif-based graph construction. (a) Examples of dual-edge motif structures. Dotted lines represent relationships used for feature engineering. (b) Transformation from an original graph G to a motif graph M . Motif instances (squares) become nodes, and shared edges in G create links Ω between them.

Algorithm 1: $\Gamma(G)$: Naive Motif Generation (CPU)

Input: Graph $G = (V, \mathcal{E})$

Output: Motif set \mathcal{M}

```

1: Initialize  $\mathcal{M} \leftarrow \emptyset$ 
2: for  $\varepsilon_j \in \mathcal{E}$  do
3:   for  $\varepsilon_k \in \mathcal{E}$  where  $k \neq j$  and  $t_k > t_j$  do
4:     if  $u_j, v_j$  and  $u_k, v_k$  share a node then
5:       Construct motif instance  $m_i$  from  $\varepsilon_j, \varepsilon_k$ 
6:        $\mathcal{M} \leftarrow \mathcal{M} \cup \{m_i\}$ 
7:     end if
8:   end for
9: end for
10: return  $\mathcal{M}$ 

```

- $\mathbf{V}_i = \{u_i^1, v_i^1, u_i^2, v_i^2\}$: The set of vertices that constitute the motif.
- $\mathbf{E}_i = \{\varepsilon_1^i, \varepsilon_2^i\} (t_1 \leq t_2), \varepsilon_1 \cap \varepsilon_2 \neq \emptyset$: The pair of edges that constitutes the motif.
- $\delta_i = \Delta(t_1, t_2)$: The temporal span, a scalar value measuring the duration of the motif's corresponding sub-graph.
- ω_i : The spatial class of the motif. The set of all binary motifs is denoted by $\mathcal{B} = \{b1, b2\}$ (where $|\mathbf{V}_i| = 2$), and the set of all ternary motifs is denoted by $\mathcal{T} = \{t1, t2, t3, t4\}$ (where $|\mathbf{V}_i| = 3$).
- $\mathbf{Y}_i = \mathbf{y}_1 + 2\mathbf{y}_2 \in \{0, 1, 2, 3\}$: A binary-encoded label vector associated with the motif instance.
- $\mathbf{X}_i = \text{concat}(x_i^V, x_i^V, x_i^E, x_i^E)$: A feature vector for the motif instance. It is formed by the concatenation of primary and aggregated features derived from the motif's vertices \mathbf{V}_i and edges \mathbf{E}_i .

In this paper, we adopt $\Delta(\cdot) = \lfloor \log_2(t_2 - t_1) \rfloor$ as the temporal span function.

Efficient Motif Instance Generation Generating the motif set $\mathcal{M} = \Gamma(G)$ from the original graph G can be computationally expensive. A naive approach, shown in Algorithm 1, uses nested loops to check all edge pairs for shared nodes, resulting in a prohibitive $O(|\mathcal{E}|^2)$ time complexity for large graphs.

To overcome this bottleneck, we propose a method with GPU acceleration that trades space for time. The core of this optimization is the 'AdjInd' function (Algorithm 2), which leverages tensor broadcasting to pre-compute all adjacency relationships between edge pairs in a single operation.

The output A is a boolean tensor where $A_{x,y,i,j}$ is true if the x -th node of edge i (0 for source, 1 for target) matches the y -th node of edge j . Pre-computing this tensor, despite its $O(|\mathcal{E}|^2)$ space complexity, reduces the effective time complexity of motif enumeration to $O(1)$. This trade-off enables the application of our method to large-scale graphs.

Motif Pruning for Scalability Unconstrained motif generation can lead to prohibitive memory usage for both the adjacency tensor A and the final motif set \mathcal{M} . To ensure scalability, we introduce a two-fold pruning strategy. First, we filter motifs by their temporal span δ_i , retaining only those

Algorithm 2: $A = \text{AdjInd}(e)$: GPU-based Edge Adjacency Indexing

Input: Edge tensor $e \in \mathbb{R}^{2 \times |e|}$

Output: $A \in \{0, 1\}^{2 \times 2 \times |e| \times |e|}$

- 1: Let I be the identity matrix of size $|e| \times |e|$
 - 2: $B \leftarrow$ Unsqueeze and repeat e to shape $2 \times |e| \times |e|$
 - 3: $C \leftarrow$ Unsqueeze and repeat e (transposed) to shape $2 \times |e| \times |e|$
 - 4: $A_{0,0} \leftarrow (B_0 = C_0) \wedge (\neg I)$ {Source-Source connection}
 - 5: $A_{1,0} \leftarrow (B_1 = C_0) \wedge (\neg I)$ {Target-Source connection}
 - 6: $A_{0,1} \leftarrow (B_0 = C_1) \wedge (\neg I)$ {Source-Target connection}
 - 7: $A_{1,1} \leftarrow (B_1 = C_1) \wedge (\neg I)$ {Target-Target connection}
 - 8: **return** A
-

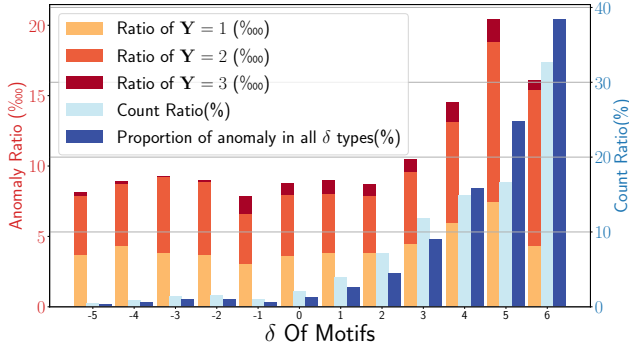


Figure 4: The anomaly ratio for different values of δ for motifs with different \mathbf{Y} , the proportion of these motifs among all motifs, and the proportion of these anomalous motifs among all types of anomalous motifs, based on statistics from subgraphs sampled from AMLWorld. Most motifs are distributed in the δ range of 3 to 6, while the proportion of anomalies is relatively high in the range of 4 and 5.

within a meaningful range $[\xi, \eta]$ based on domain knowledge or data statistics, as shown in Figure 4:

$$\mathcal{M}^* = \{m_i \in \mathcal{M} \mid \xi \leq \delta_i \leq \eta\} \quad (1)$$

Second, we control the maximum node degree d_{\max} during graph construction. The total number of dual-edge motifs is given by:

$$|\mathcal{M}| = \sum_{i=1}^{|\mathcal{V}|} \binom{d_i}{2} = \frac{1}{2} \left(\sum_{i=1}^{|\mathcal{V}|} d_i^2 \right) - |\mathcal{E}| \quad (2)$$

Given the constraint $\sum_{i=1}^{|\mathcal{V}|} d_i = 2|\mathcal{E}|$, Jensen’s inequality implies that $\sum d_i^2$ is maximized when degrees are concentrated. This leads to an upper bound for $|\mathcal{M}|$:

$$|\mathcal{M}|_{\max} \approx \left\lfloor \frac{2|\mathcal{E}|}{d_{\max}} \right\rfloor \binom{d_{\max}}{2} \leq |\mathcal{E}|(d_{\max} - 1) \quad (3)$$

This result confirms that limiting the maximum degree d_{\max} is a highly effective strategy for controlling the size of \mathcal{M} , thereby ensuring the scalability of the entire process.

4.2 Dual-Edge Motif Graph Neural Networks (D-EMGNN)

Our proposed D-EMGNN model processes the motif graph using two core modules: the Spatial Attentive Information Redundancy Reduction (SAIRR) module and the Motif Graph Convolution Network (MGCN). The process begins by obtaining initial node embeddings through a linear transformation of the motif features, $\mathbf{h}_i^{(0)} = \mathbf{W}_{\text{in}} \mathbf{X}_i + b_{\text{in}}$.

Spatial Attentive Information Redundancy Reduction (SAIRR) In an AML context, different motif types (e.g., a binary ‘pass-through’ vs. a ternary ‘fan-in’) can involve the same financial accounts, leading to informational redundancy. SAIRR is designed to purify the unique structural signal of each motif type. First, it maps the hidden representation $\mathbf{h}_i^{(l)}$ at layer l into type-specific spaces:

$$\mathbf{h}_{\omega,i}^{(l)} = \mathbf{W}_{\omega}^{(l)} \mathbf{h}_i^{(l)} + b_{\omega}^{(l)}, \omega \in \{\mathcal{B}, \mathcal{T}\} \quad (4)$$

Then, to minimize redundancy, it computes attention coefficients across these heterogeneous spaces, allowing it to dynamically weigh the incremental contribution of each motif type.

$$\mathbf{Q}^{(l)} = \mathbf{h}_{\omega_i,i}^{(l)} \mathbf{W}_{\text{att},\omega_i}^{(l)} \quad (5)$$

$$\mathbf{K}_{\omega}^{(l)} = \mathbf{h}_{\omega,i}^{(l)} \mathbf{W}_{\text{att},\omega}^{(l)}, \omega \neq \omega_i \quad (6)$$

$$\alpha_{\omega} = \frac{\exp \left(\frac{\mathbf{Q}^{(l)} \mathbf{K}_{\omega}^{(l)\top}}{\sqrt{|\mathcal{T}|+|\mathcal{B}|}} \right)}{\sum_{\omega' \neq \omega_i} \exp \left(\frac{\mathbf{Q}^{(l)} \mathbf{K}_{\omega'}^{(l)\top}}{\sqrt{|\mathcal{T}|+|\mathcal{B}|}} \right)} \quad (7)$$

The redundancy-suppressed representation $\tilde{\mathbf{h}}_i^{(l)}$ is then calculated as:

$$\tilde{\mathbf{h}}_i^{(l)} = \sigma \left(\beta_i^{(l)} \left(\mathbf{h}_{\omega_i,i}^{(l)} - \sum_{\omega \neq \omega_i} \alpha_{\omega} \mathbf{h}_{\omega,i}^{(l)} \right) \right) \quad (8)$$

$$\beta_i^{(l)} = \sigma \left(\mathbf{h}_{\omega_i,i}^{(l)\top} \sum_{\omega \neq \omega_i} \alpha_{\omega} \mathbf{h}_{\omega,i}^{(l)} \right) \quad (9)$$

Here, the gating scalar $\beta_i^{(l)}$, defined in (9), measures the similarity between a motif’s own type-specific embedding and the aggregated embeddings of other types. By adaptively subtracting weighted redundant information, SAIRR forces the model to focus on the distinct topological roles of each laundering tactic. This approach improves upon prior work (Chen et al. 2023) on motif-based method by using a more nuanced, attention-driven reduction mechanism instead of simple linear subtraction. The workflow is illustrated in Figure 5.

Motif Graph Convolution Network (MGCN) The proposed MGCN layer aggregates information from neighboring motifs, extending the standard Graph Convolutional Network (GCN) (Kipf and Welling 2016) to handle heterogeneous motif interactions. In sophisticated AML schemes, the interaction between two ‘fan-in’ motifs may signify a different behavior than the interaction between a ‘fan-in’ and a

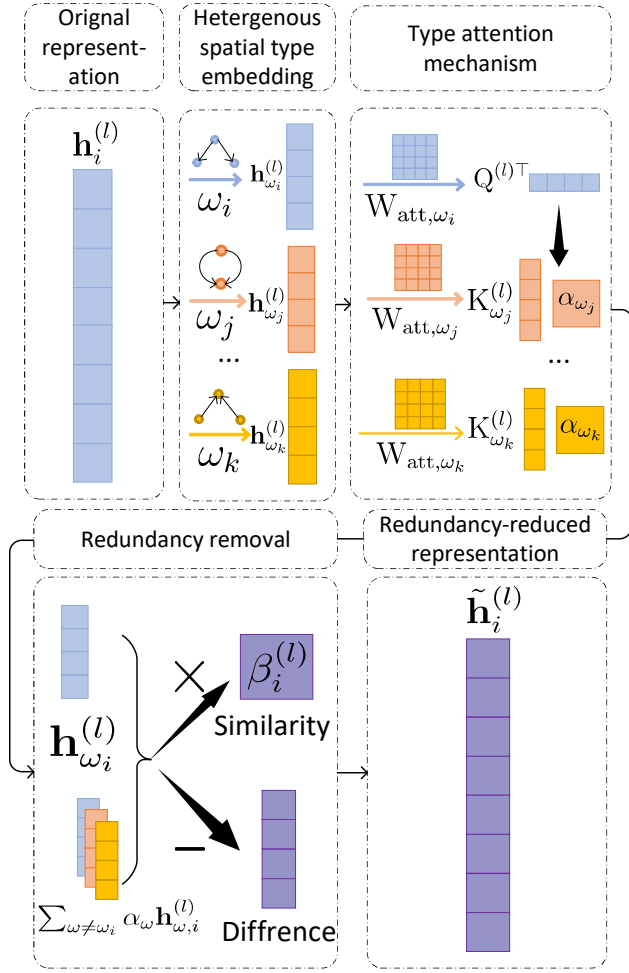


Figure 5: Workflow of the Spatial Attentive Information Redundancy Reduction (SAIRR) module. SAIRR purifies motif representations by using an attention mechanism to identify and subtract redundant information from different motif types.

‘cycle’ motif. To capture this “grammar” of laundering tactics, MGCN incorporates learnable, type-aware biases:

$$\mathbf{h}_i^{(l+1)} = \sigma \left(\sum_{j \in \mathcal{N}(m_i) \cup \{m_i\}} \frac{\mathbf{W} \tilde{\mathbf{h}}_j^{(l)}}{\sqrt{d_i d_j}} + b_{\omega_i \omega_j} \right), \quad l < L \quad (10)$$

$$d_i = |\mathcal{N}(m_i)| + 1 \quad (11)$$

Here, $b_{\omega_i \omega_j}$ is a bias vector indexed by the types of both the source (ω_i) and target (ω_j) motifs. Unlike a standard GCN, which treats all connections uniformly, MGCN learns distinct transformations for each pair of interacting motif types. This enables a more powerful and structurally-aware aggregation that respects the diverse composition of complex laundering operations.

4.3 Motif-Edge Label Propagation (MELP)

As shown in Figure 6, MELP introduces a novel framework for predicting edge labels by leveraging motif-level information embedded within graph structures. In this approach, motif representations $\mathbf{h}_i^{(L)}$ are projected into a space of dimension \mathbf{Y} via a multilayer perceptron (MLP), and motif label probabilities are obtained using the softmax function:

$$\mathbf{P}_i^{\text{edge}} = \text{MLP}(\mathbf{h}_i^{(L)}) \quad (12)$$

$$p_{ij}^{\text{edge}} = \frac{\exp(\mathbf{P}_{ij}^{\text{edge}})}{\sum_{k=0}^{2^{|\mathcal{E}|-1}} \exp(\mathbf{P}_{ik}^{\text{edge}})}. \quad (13)$$

Here, p_{ij}^{edge} denotes the probability that motif i is assigned to class j .

Edge label inference The label probability for each edge within a motif is inferred by marginalizing the motif class probabilities. For a dual-edge motif, this is calculated as:

$$P(\mathbf{y}_1^i = 1) = p_{i1} + p_{i3} \quad (14)$$

$$P(\mathbf{y}_2^i = 1) = p_{i2} + p_{i3} \quad (15)$$

The final predicted label probability $\hat{\mathbf{y}}_{\varepsilon_k}$ for an edge ε_k in the original graph is computed by averaging the inferred probabilities from all motifs that contain it:

$$\begin{aligned} \hat{\mathbf{y}}_{\varepsilon_k} &= P(\mathbf{y}_{\varepsilon_k} = 1) \\ &= \frac{\sum_{m_i \in \mathcal{M}, \varepsilon_k = \varepsilon_1^i} P(\mathbf{y}_1^i = 1)}{|\{m_i : \varepsilon_k \in m_i\}|} \\ &\quad + \frac{\sum_{m_i \in \mathcal{M}, \varepsilon_k = \varepsilon_2^i} P(\mathbf{y}_2^i = 1)}{|\{m_i : \varepsilon_k \in m_i\}|} \end{aligned} \quad (16)$$

The model is trained by minimizing the binary cross-entropy loss over all edges in the original graph:

$$L(\mathbf{y}, \hat{\mathbf{y}}) = -\frac{1}{|\mathcal{E}|} \sum_{i=1}^{|\mathcal{E}|} \mathbf{y}_i \log(\hat{\mathbf{y}}_i) + (1 - \mathbf{y}_i) \log(1 - \hat{\mathbf{y}}_i). \quad (17)$$

5 Experiments

5.1 Datasets

Real-world Anti-Money Laundering (AML) datasets are exceptionally rare due to stringent privacy and regulatory constraints. Our evaluation therefore strategically combines a proprietary, real-world banking dataset with two large-scale public benchmarks that serve as crucial proxies by emulating complex financial transactions.

- **TransBank**: A proprietary transactional dataset from an anonymous bank, containing a curated subset of records with anomalous behaviors. Adhering to strict regulatory compliance, all personally identifiable information (PII) was removed and the data were securely destroyed post-experiment.
- **AMLWorld**: A large-scale synthetic benchmark from the AMLWorld simulator (Altman et al. 2023), designed to realistically model inter-bank transaction flows for AML research.

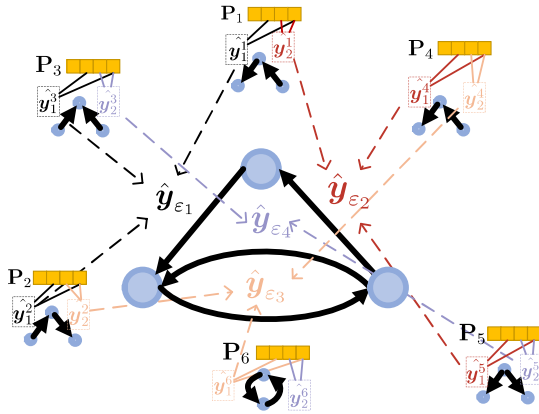


Figure 6: Overview of the Motif-Edge Label Propagation (MELP) process. Motif-level predictions, generated from the final GNN embeddings via an MLP, are propagated back to their constituent edges in the original graph. The final label for an edge is determined by aggregating predictions from all motifs to which it belongs. The example motif graph is the same as in Figure 3b.

- **DgraphE**: A real-world transaction dataset introduced by (Huang et al. 2022). We repurposed its original node classification task for edge-centric detection (identifying anomalous transactions), creating the version we term DgraphE.

Table 1 summarizes the statistics of these datasets.

Dataset	$ V $	$ \mathcal{E} $	\dim_{x_u}	$\dim_{x_{uv}}$	Anomaly Ratio(%)
TransBank	130M	4.5B	17	289	0.02
AMLWorld	0.51M	5.07M	-	47	0.15
DgraphE	3.7M	4.3M	17	-	14.5

Table 1: Statistics of the datasets used in experiments.

5.2 Settings

Environment Experiments were conducted on either an Ubuntu 24.04 LTS server equipped with 2 AMD 9654 CPUs, 2 NVIDIA Tesla H800 (80GB) GPUs, and 1.5TB of RAM, or an Ubuntu 24.04 LTS server equipped with 2 Intel Xeon Gold 6430 CPUs, 8 NVIDIA RTX 4090 GPUs, and 512GB of RAM, depending on resource availability to expedite the process. All implementations were based on the `torch-geometric` Python package (Fey and Lenssen 2019). It is noted that due to bank compliance regulations and confidentiality agreements, the TransBank dataset and some of its corresponding experimental details cannot be publicly disclosed.

Baselines To comprehensively evaluate the effectiveness of our proposed dual-edge motif graph, we conduct a comparative analysis. The core of our evaluation is to compare the performance of representative GNN architectures on the original graph versus their counterparts on our motif-based graph. To distinguish these two settings, we append

an “e” suffix to models operating on the original graph (e.g., GINe) and an “m” suffix for those on the dual-edge motif graph (e.g., GINm). Our selection of fundamental baselines includes widely-recognized GNN architectures: GIN (Xu et al. 2018), GAT (Veličković et al. 2017), and PNA (Corso et al. 2020), all of which support learnable edge attributes. Notably, PNA’s architecture relies on pre-computed graph statistics (e.g., degree distribution) that are not well-defined or directly applicable to our dual-edge motif graph. Therefore, for a comparison on the motif graph, we substitute PNA with an adapted GCN (Kipf and Welling 2016) model, denoted as GCNm. Furthermore, to benchmark against state-of-the-art methods relevant to higher-order structures, we incorporate HOGAT (Huang et al. 2021) and GUIDE (Yuan et al. 2021) due to official code availability. We have carefully adapted these models to align with our edge-level prediction task and methodology, referring to them as HOGATm and GUIDEm respectively. Importantly, our experimental model features an end-to-end nature. Consequently, our experimental results differ from those of other methods that precompute and enhance features (e.g., ego-ids) from the entire dataset, which we consider less realistic.

Evaluation Metrics We employ F1-score and Area Under the Precision-Recall Curve (AUPRC) as our evaluation metrics, specifically chosen to handle the severe class imbalance inherent in anti-money laundering (AML) tasks. The F1-score offers a balanced measure of precision and recall at a single decision threshold. However, AUPRC provides a more comprehensive and robust assessment by evaluating performance across all thresholds, which is crucial as it better reflects a model’s ability to identify rare positive instances. Consequently, AUPRC serves as our primary evaluation criterion.

Training Details. We first chronologically split the data into training, validation, and test sets. All models were trained for up to 50 epochs with subsampling strategies. We saved the model checkpoint that achieved the best F1-score on the validation set. This model was then evaluated once on the test set to report final performance. To ensure a fair comparison, all models shared the same input features and a 2-layer GNN architecture with a 256-dim hidden layer. We employed a 2-hop neighbor sampling strategy, sampling up to 100 neighbors per hop. The learning rate was tuned from $\{10^{-5}, 5 \times 10^{-4}, 10^{-4}\}$, and the batch size was selected from $\{16, 32, 64, 128\}$.

5.3 Performance

The experimental results are reported as the mean \pm standard deviation over 5 independent runs. Each entry is presented in this format.

Model Comparison The performance of all models is detailed in Table 2. The results highlight three critical points. **1) Value of Motif Representation:** A direct comparison between the first two blocks of the table (e.g., GINe vs. GINm) reveals that simply switching to the dual-edge motif graph yields substantial performance gains for all GNN backbones. This confirms the benefit of encoding higher-order

Dataset	TransBank		AMLWorld		DgraphE	
	F1(%)	AUPRC (%)	F1(%)	AUPRC(%)	F1(%)	AUPRC(%)
GINe	7.84±0.52	4.69±0.95	10.8±0.81	7.69±0.91	25.3±1.06	48.4±0.08
GATe	12.3±1.56	6.42±0.85	16.8±2.68	8.78±0.58	20.7±0.87	48.5±0.25
PNAe	13.9±1.47	7.87±1.24	19.4±2.44	11.8±1.89	21.2±1.45	46.4±0.01
GINm	14.5±1.22	5.84±1.67	16.2±1.68	7.58±1.94	32.4±1.84	53.8±0.16
GATm	16.8±1.97	9.47±1.59	24.4±0.91	12.9±1.18	33.2±1.57	54.1±0.21
GCNm	17.4±1.58	10.2±1.48	23.1±1.13	11.9±0.77	33.1±1.33	52.7±0.37
HOGATm	4.37±0.32	2.62±0.39	10.7±2.15	7.23±0.89	23.2±1.38	47.7±0.02
GUIDEm	7.62±0.72	3.02±0.39	12.4±1.73	8.64±0.76	19.6±1.18	46.3±0.17
D-EMGNN	18.6±1.24	14.2±1.77	26.4±1.55	15.2±1.96	35.6±1.85	56.2±0.03
w/o SAIRR	15.4±1.06	8.79±0.87	21.2±0.44	10.8±0.71	31.9±1.29	54.3±0.24
w/o MGCN	11.2±1.26	6.24±0.47	16.8±1.87	7.36±1.05	24.9±1.67	47.5±0.08
w/o both	12.7±1.15	7.48±0.58	18.0±1.09	8.39±0.91	28.3±1.73	51.7±0.12

Table 2: Performance comparison as well as ablation study results on datasets. Best performance on the test set are bolded.

η	3	4	5	6
F1(%)	7.89±0.45	21.5±0.80	24.7±0.90	26.4±1.55
AUPRC(%)	2.00±0.31	10.3±1.44	13.5±0.98	15.2±1.96
Runtime(min)	15	30	65	75
Max.batch size	128	64	32	16

Table 3: Performance of D-EMAML on AMLWorld with varying η values ($\xi = 0$). The best performances on the test set are bolded.

patterns. **2) D-EMGNN’s Superiority:** Our D-EMGNN model consistently outperforms all other models across the three datasets. Its lead is particularly pronounced in the AUPRC metric, which is vital for imbalanced data. **3) Comparison with Higher-Order Methods:** D-EMGNN’s performance stands in stark contrast to other higher-order methods like HOGATm and GUIDEm, which performed poorly in the edge detection task. This underscores the effectiveness and robustness of our proposed message-passing scheme on the dual-edge motif graph.

Ablation Study The last block of Table 2 reports the results of module ablation experiments on datasets. The complete D-EMGNN model consistently achieves the best performance across both datasets. Removing either SAIRR or the $b_{\omega_i\omega_j}$ term in MGCN leads to noticeable drops in both F1 and AUPRC, highlighting the essential role of each component. Further, simultaneously removing both modules causes a substantial additional decline in performance compared to removing SAIRR alone, demonstrating the importance of both modules to the overall model effectiveness. Interestingly, when both modules are removed, the model degenerates into a homogeneous variant, which performs better than the variant with only the $b_{\omega_i\omega_j}$ term ablated. This suggests that learning heterogeneous embedding representations with a homogeneous message passing mechanism may actually be detrimental to model performance.

Hyperparameter Analysis The performance of our D-EMAML model for different values of η is shown in Table 3, with ξ fixed at 0 on the AMLWorld dataset. As η increases, both F1 score and AUPRC initially improve, with optimal results achieved at $\eta = 6$, beyond which performance plateaus. At the same time, the training time per epoch increases, and the maximum feasible batch size decreases. Furthermore, we observe that the performance at $\eta = 5$ is not significantly different from that at $\eta = 6$. These results indicate that the proposed pruning strategy allows the model to focus on salient motif instances, enables scaling up of the training set, and reduces training time overhead.

6 Conclusion

We introduce D-EMAML, a novel framework for anti-money laundering that fundamentally advances edge-level anomaly detection by embedding financial domain knowledge directly into a high-order graph representation. Our dual-edge motif graph materializes this by modeling money laundering patterns as foundational building blocks, bridging the critical gap between structural analysis and direct edge-level prediction where traditional methods falter. Powered by tailored SAIRR and MGCN modules, D-EMAML’s motif-aware representation enables GNNs to achieve a substantial performance leap over counterparts on the original graph, as verified on large-scale datasets. Our efficient pruning strategy ensures this sophisticated analysis is tractable for real-world deployment. Ultimately, D-EMAML establishes a new paradigm for fusing domain expertise with the very structure of graph learning models, providing a robust foundation for future research in critical applications. This approach not only delivers a powerful model for financial security but also charts a course for developing more interpretable and domain-aware AI systems in other high-stakes fields. The framework’s primary limitation is its reduced sensitivity to isolated laundering transactions. Future work will prioritize enhancing algorithmic scalability via memory optimization and improving model interpretability.

Acknowledgments

All the authors contribute to this work equally. Xiaofeng Zhou is the corresponding author. This work is supported by the Key Special Funds of the National Natural Science Foundation of China (Grant No. 72342034 & 72342021). This work is also supported by Innovation Team of Philosophy and Social Sciences (Financial Technology and Security Governance), MOE of China.

References

- Altman, E.; Blanuša, J.; Von Niederhäusern, L.; Egressy, B.; Anghel, A.; and Atasu, K. 2023. Realistic synthetic financial transactions for anti-money laundering models. *Advances in Neural Information Processing Systems*, 36: 29851–29874.
- Chen, X.; Cai, R.; Fang, Y.; Wu, M.; Li, Z.; and Hao, Z. 2023. Motif graph neural network. *IEEE Transactions on Neural Networks and Learning Systems*.
- 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.
- Corso, G.; Cavalleri, L.; Beaini, D.; Liò, P.; and Veličković, P. 2020. Principal neighbourhood aggregation for graph nets. *Advances in neural information processing systems*, 33: 13260–13271.
- Egressy, B.; Von Niederhäusern, L.; Blanuša, J.; Altman, E.; Wattenhofer, R.; and Atasu, K. 2024. Provably powerful graph neural networks for directed multigraphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38(10), 11838–11846.
- Fey, M.; and Lenssen, J. E. 2019. Fast graph representation learning with PyTorch Geometric. *arXiv preprint arXiv:1903.02428*.
- Hamilton, W.; Ying, Z.; and Leskovec, J. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30.
- Hilal, W.; Gadsden, S. A.; and Yawney, J. 2022. Financial fraud: a review of anomaly detection techniques and recent advances. *Expert systems With applications*, 193: 116429.
- Huang, L.; Zhu, Y.; Gao, Y.; Liu, T.; Chang, C.; Liu, C.; Tang, Y.; and Wang, C.-D. 2021. Hybrid-order anomaly detection on attributed networks. *IEEE Transactions on Knowledge and Data Engineering*, 35(12): 12249–12263.
- Huang, X.; Yang, Y.; Wang, Y.; Wang, C.; Zhang, Z.; Xu, J.; Chen, L.; and Vazirgiannis, M. 2022. Dgraph: A large-scale financial dataset for graph anomaly detection. *Advances in Neural Information Processing Systems*, 35: 22765–22777.
- Huang, Z.; Huang, Y.; Qian, P.; Chen, J.; and He, Q. 2023. Demystifying bitcoin address behavior via graph neural networks. In *2023 IEEE 39th International Conference on Data Engineering (ICDE)*, 1747–1760. IEEE.
- Kipf, T. N.; and Welling, M. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.
- Lee, J. B.; Rossi, R. A.; Kong, X.; Kim, S.; Koh, E.; and Rao, A. 2019. Graph convolutional networks with motif-based attention. In *Proceedings of the 28th ACM international conference on information and knowledge management*, 499–508.
- Li, X.; Li, Y.; Mo, X.; Xiao, H.; Shen, Y.; and Chen, L. 2023. Diga: Guided diffusion model for graph recovery in anti-money laundering. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 4404–4413.
- Lin, D.; Wu, J.; Huang, T.; Lin, K.; and Zheng, Z. 2023. Who is who on Ethereum? Account labeling using heterophilic graph convolutional network. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 54(3): 1541–1553.
- Long, J.; Ji, S.; Wang, Z.; Chen, T.; and Yang, L. 2025. A Hypergraph Neural Network with Motif Interaction Enhancement. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 383–394. Springer.
- Luo, X.; Han, X.; Zuo, W.; Wu, X.; and Liu, W. 2024. MLaD²: A Semi-Supervised Money Laundering Detection Framework Based on Decoupling Training. *IEEE Transactions on Information Forensics and Security*, 19: 4518–4533.
- Milo, R.; Shen-Orr, S.; Itzkovitz, S.; Kashtan, N.; Chklovskii, D.; and Alon, U. 2002. Network motifs: simple building blocks of complex networks. *Science*, 298(5594): 824–827.
- Motie, S.; and Raahemi, B. 2024. Financial fraud detection using graph neural networks: A systematic review. *Expert Systems with Applications*, 240: 122156.
- Paranjape, A.; Benson, A. R.; and Leskovec, J. 2017. Motifs in temporal networks. In *Proceedings of the tenth ACM international conference on web search and data mining*, 601–610.
- Pourhabibi, T.; Ong, K.-L.; Kam, B. H.; and Boo, Y. L. 2020. Fraud detection: A systematic literature review of graph-based anomaly detection approaches. *Decision Support Systems*, 133: 113303.
- Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; and Bengio, Y. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.
- Wang, D.; Zhang, Z.; Zhao, Y.; Huang, K.; Kang, Y.; and Zhou, J. 2023. Financial default prediction via motif-preserving graph neural network with curriculum learning. In *Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining*, 2233–2242.
- Wu, R.; Ma, B.; Jin, H.; Zhao, W.; Wang, W.; and Zhang, T. 2022. Grande: a neural model over directed multigraphs with application to anti-money laundering. In *2022 IEEE International Conference on Data Mining (ICDM)*, 558–567. IEEE.
- Xiao, C.; Pang, S.; Tai, W.; Huang, Y.; Trajcevski, G.; and Zhou, F. 2024. Motif-Consistent Counterfactuals with Adversarial Refinement for Graph-Level Anomaly Detection. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 3518–3526.

- Xu, K.; Hu, W.; Leskovec, J.; and Jegelka, S. 2018. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*.
- Yuan, X.; Zhou, N.; Yu, S.; Huang, H.; Chen, Z.; and Xia, F. 2021. Higher-order structure based anomaly detection on attributed networks. In *2021 IEEE International Conference on Big Data (Big Data)*, 2691–2700. IEEE.
- Yuan, Z.; Shao, M.; and Yan, Q. 2023. Motif-level anomaly detection in dynamic graphs. *IEEE Transactions on Information Forensics and Security*, 18: 2870–2882.
- Zhao, J.-L.; Zhang, X.-J.; Ding, X.; Zhang, X.; and Zhang, H.-F. 2025. Enhancing Graph Structure Learning Via Motif-Driven Hypergraph Construction. *IEEE Transactions on Network Science and Engineering*.