

Real-Time Cross-Border Anti-Money Laundering (AML) Detection Using Graph Neural Networks and Machine Learning

Gopinathan Rathinavelu

Data Architect, Arizona, USA ,gopinathan.r@gmail.com

Abstract: Cross-border money laundering involves complex, rapidly changing transaction relationships that are difficult to identify using conventional rule-based and isolated transaction-level machine learning systems. This paper proposes a real-time Anti-Money Laundering (AML) detection framework that integrates Graph Neural Networks (GNNs) with machine learning to model financial transactions as dynamic, heterogeneous graphs. Customers, accounts, financial institutions, merchants, devices, jurisdictions, and transactions are represented as interconnected nodes and edges, enabling the model to capture transactional dependencies, circular fund movements, layering patterns, intermediary accounts, and cross-border risk propagation. A temporal graph-learning mechanism is incorporated to process streaming transactions and identify emerging suspicious structures with low detection latency. Supervised machine learning is combined with graph-based embeddings to improve classification under severe class imbalance, while anomaly detection supports the identification of previously unseen laundering behaviours. The proposed framework further incorporates risk scoring, explainable graph patterns, and privacy-aware cross-institutional information exchange. The resulting architecture provides an integrated foundation for scalable, adaptive, and real-time AML surveillance across international financial networks.

Keywords: Anti-Money Laundering, Graph Neural Networks, Machine Learning, Cross-Border Transactions, Financial Crime Detection, Real-Time Anomaly Detection

1. Introduction

Money laundering represents a persistent threat to the integrity, stability, and transparency of the international financial system. The increasing globalization of banking, digital payments, correspondent banking, online financial services, cryptocurrency exchanges, and cross-border fund transfers has created highly interconnected financial environments in which illicit funds can be transferred through multiple accounts, institutions, jurisdictions, and transaction channels. Unlike conventional financial fraud, money laundering frequently involves a sequence of apparently legitimate transactions designed to obscure the origin, ownership, destination, or beneficial control of funds. Consequently, identifying a suspicious transaction in isolation may be insufficient because the relevant evidence can be distributed across a broader network of interconnected financial activities [1], [2].

Traditional Anti-Money Laundering (AML) systems have predominantly relied on predefined rules, transaction thresholds, customer profiles, watchlists, and manually constructed risk indicators. Although rule-based monitoring remains an important component of regulatory compliance, its effectiveness can be constrained when laundering strategies evolve faster than detection rules. Machine learning has consequently emerged as an important analytical approach because it can process large volumes of transactional and customer information and identify behavioural patterns that may not be explicitly represented in predefined rules [3]. Supervised learning approaches can classify transactions according to previously identified suspicious patterns, whereas unsupervised and semi-supervised approaches can identify unusual behaviours when reliable labels are limited. However, the scarcity of high-quality



labelled AML datasets remains a significant obstacle to the direct application of supervised learning in real-world environments [4].

Previous research demonstrates that machine learning can improve the prioritisation of suspicious transactions and reduce the manual burden associated with AML investigations. Supervised models trained using historical transaction information, previous alerts, and reported cases have demonstrated the potential to distinguish transactions requiring further investigation from routine financial activity [5]. Similarly, ensemble-based classification techniques have been investigated for the early identification of organisations exhibiting characteristics associated with money laundering, demonstrating the usefulness of feature selection, categorical-variable transformation, and gradient-based classification methods [6]. Neural-network-based approaches have also demonstrated that incorporating both transactional and non-transactional information can improve abnormality detection and reduce false-positive rates [7].

Nevertheless, financial transactions are inherently relational. A customer may maintain several accounts, transfer funds through multiple intermediaries, interact with different financial institutions, and conduct transactions across several jurisdictions. The relationship between these entities can therefore contain information that is not observable from individual transaction attributes. Graph-based modelling provides a natural representation of this environment by treating financial entities as nodes and transactions or relationships as edges. Such representations enable the analysis of interconnected transaction structures, allowing suspicious pathways, intermediary accounts, cycles, communities, and fund-transfer chains to be investigated collectively rather than as independent observations [8].

The importance of graph-based AML analysis has been demonstrated through research on financial transaction networks. Flow-based graph analysis, for example, can trace monetary movements between source and destination accounts and identify laundering structures involving multiple intermediary accounts [9]. Large-scale graph learning has further demonstrated the feasibility of applying graph convolutional techniques to financial networks containing millions of nodes and edges, highlighting the potential of graph-based approaches for analysing the structural and behavioural characteristics of AML environments [10]. In cryptocurrency-based financial networks, graph convolutional approaches have also demonstrated the value of relational information for identifying illicit transactions, particularly where transactions form interconnected temporal structures rather than independent observations [11].

Graph Neural Networks (GNNs) provide an extension of conventional graph analytics by learning representations directly from the topology and attributes of connected entities. Graph Convolutional Networks can aggregate information from neighbouring nodes, enabling a transaction or account to be evaluated in the context of its surrounding financial network [12]. Graph Attention Networks further allow the learning mechanism to assign different levels of importance to neighbouring entities and relationships, which is potentially useful when some transaction relationships carry considerably greater AML significance than others [13]. These capabilities are particularly relevant to cross-border AML because suspicious financial activity may emerge through multi-hop relationships involving accounts and institutions located in different jurisdictions.

Temporal characteristics introduce another important dimension. Financial transaction networks are dynamic rather than static: accounts become active or inactive, transaction relationships emerge and disappear, and laundering networks can change their operational structure in response to detection mechanisms. Temporal Graph Networks provide a framework for learning representations from continuously evolving graph structures, making temporal graph learning relevant to real-time financial monitoring [14]. Integrating temporal information with GNN-based representations can therefore enable AML systems to consider not only who is connected to whom, but also when relationships emerge, how frequently funds move, and how transaction patterns evolve over time.

The cross-border dimension further increases the complexity of AML detection. Transactions may pass through several financial institutions and jurisdictions before reaching their ultimate destination. Differences in regulatory regimes, customer-identification requirements, reporting practices, data formats, and information-sharing mechanisms can make it difficult to construct a unified representation of financial activity. Machine learning research has consequently highlighted the importance of technological capability, data availability, organisational context, and

regulatory constraints when implementing automated AML systems [4]. A practical real-time framework must therefore balance predictive performance with computational scalability, explainability, privacy, and regulatory usability.

The present study addresses these challenges by developing a conceptual real-time cross-border AML detection framework based on the integration of GNNs and conventional machine learning. The proposed approach represents financial activity as a heterogeneous and temporally evolving graph containing customers, accounts, transactions, financial institutions, devices, merchants, and jurisdictions. Graph-based representations are used to capture relational and structural patterns, while machine learning mechanisms are incorporated for classification, anomaly detection, and risk scoring. The framework is intended to identify both known suspicious behaviours and previously unseen transaction structures, thereby supporting adaptive AML monitoring.

The principal contribution of the study is the integration of relational, temporal, behavioural, and cross-border information within a unified detection architecture. Rather than treating AML as a transaction-level binary classification problem alone, the proposed framework considers suspicious activity as a network phenomenon in which the significance of an individual transaction may depend on its surrounding transaction structure. This perspective provides a foundation for real-time identification of multi-hop fund transfers, circular transactions, unusual intermediary relationships, rapid cross-border movement, and emerging suspicious subgraphs. The framework also considers explainability and risk prioritisation so that detected patterns can support rather than replace expert investigation and regulatory decision-making.

2. Literature Review

2.1 Conventional Machine Learning for AML Detection

Machine learning has increasingly been investigated as an alternative or complement to conventional rule-based AML monitoring. Earlier AML approaches generally relied on manually specified transaction thresholds and customer-risk rules, which can generate substantial numbers of alerts requiring human investigation. Machine learning enables historical transactional characteristics and behavioural variables to be used for learning predictive patterns. Research using large-scale banking transaction data demonstrated that supervised learning could be applied to prioritise transactions according to their probability of being associated with suspicious activity, while also showing that the composition of the training data has a substantial influence on model performance [5]. This finding is particularly important because AML datasets frequently contain noisy labels, investigator-generated alerts, and substantial class imbalance.

Research on early money-laundering detection has also investigated feature engineering, categorical encoding, feature selection, and ensemble classification. Gradient-boosting approaches have demonstrated potential for identifying organisations exhibiting characteristics associated with money laundering and terrorist financing [6]. Such approaches are computationally practical and can provide strong performance on structured tabular financial data. However, transaction-level or entity-level models may have difficulty representing relationships among multiple accounts and transactions when suspicious behaviour is distributed across a network.

The availability of reliable labels remains one of the principal challenges in AML machine learning. Actual confirmed laundering cases represent only a small fraction of total financial transactions, while suspicious-activity alerts may contain false positives or uncertain outcomes. Research has therefore highlighted the limited availability of high-quality labelled datasets and the resulting constraints on conventional supervised learning [4]. Unsupervised and anomaly-detection approaches can partially address this limitation by identifying unusual financial behaviour, although unusual activity does not necessarily constitute money laundering. Consequently, combining supervised classification with anomaly detection can provide a more comprehensive analytical strategy.

Neural-network approaches have further expanded the modelling capabilities available for AML. Research incorporating transactional and non-transactional variables demonstrated that abnormality indicators can improve customer-level and group-level comparisons while reducing false-positive proportions [7]. These findings indicate

that AML detection benefits from combining multiple categories of information rather than relying exclusively on transaction amount, frequency, or geographical attributes.

2.2 Graph-Based Representation of Financial Transactions

A fundamental limitation of conventional transaction-level modelling is that financial crime frequently manifests through relationships among multiple entities. Graph representation addresses this limitation by modelling accounts, customers, institutions, and other financial entities as nodes while representing financial interactions as directed or undirected edges. This enables analytical models to capture network topology, transaction paths, interconnected accounts, and collective behaviour.

Flow-based graph analysis has demonstrated the importance of tracing complete monetary pathways rather than focusing exclusively on dense groups of accounts. A graph representation can identify the movement of funds from source accounts through intermediary accounts toward destination accounts, thereby providing information about potential layering structures [9]. This is particularly relevant to AML because laundering schemes may deliberately distribute transactions across several intermediaries to obscure the origin of funds.

Large-scale graph learning has further established the feasibility of applying graph convolutional techniques to financial transaction networks. Research using large synthetic AML graphs demonstrated the importance of scalability because practical financial networks can contain very large numbers of nodes and edges [10]. Computational efficiency therefore becomes an essential component of an AML system intended for continuous monitoring.

Graph-based analysis has also been applied to cryptocurrency transaction networks. The Elliptic dataset provided a large labelled temporal transaction graph containing more than 200,000 Bitcoin transactions and directed payment flows, enabling researchers to investigate illicit transaction classification using both conventional machine learning and graph convolutional approaches [11]. The research demonstrated the value of combining relational information with conventional transaction features, while also highlighting challenges associated with graph size, temporal evolution, and interpretability.

2.3 Graph Neural Networks for AML Detection

GNNs extend conventional graph analysis by learning node or graph representations through neighbourhood aggregation. A Graph Convolutional Network propagates information across neighbouring nodes so that the representation of an entity incorporates information from its surrounding graph structure [12]. In AML applications, this means that the risk representation of an account can incorporate the characteristics and behaviours of connected accounts rather than relying exclusively on the account's own transaction history.

Graph Attention Networks provide another mechanism for relational learning by assigning learned attention weights to neighbouring nodes [13]. This capability is relevant to financial crime detection because relationships within a transaction graph are not equally informative. A high-value transfer to a previously unrelated foreign account, repeated transactions through a newly established intermediary, or a rapidly changing transaction pattern may carry different analytical significance from routine transfers between long-established counterparties.

Graph-based AML research has increasingly moved from simple homogeneous transaction graphs toward more complex representations involving multiple entity and relationship types. A heterogeneous representation can distinguish customers, accounts, institutions, jurisdictions, devices, merchants, and transactions while retaining the relationships among them. Such modelling can provide a richer representation of cross-border financial activity than a graph containing accounts and monetary transfers alone.

The principal advantage of GNN-based AML modelling is therefore the ability to combine **node attributes, edge attributes, network topology, and neighbourhood information**. A suspicious account can be identified not simply because of its own transaction characteristics, but because of its position within a broader structure. This can facilitate detection of intermediary accounts, suspicious communities, transaction chains, circular flows, and coordinated patterns that may remain difficult to identify through independent transaction classification.

2.4 Temporal Graph Learning for Real-Time Detection

Static graph models provide an incomplete representation of financial activity because transaction relationships change continuously. A financial account may interact with hundreds of counterparties during one period and exhibit completely different behaviour later. Consequently, temporal information is essential for distinguishing persistent legitimate relationships from rapidly emerging suspicious structures.

Temporal Graph Networks provide a framework for learning from continuously evolving graph interactions by incorporating temporal information into node representations [14]. The approach is relevant to real-time AML because each new transaction can potentially update the representation of participating entities and their surrounding neighbourhoods. This creates the possibility of evaluating transactions within their current temporal context rather than relying exclusively on historical aggregates.

Temporal modelling is particularly important for detecting laundering strategies involving rapid fund movement. Structuring, layering, rapid pass-through transactions, circular transfers, and sudden cross-border movement can be identified more effectively when transaction ordering and timing are preserved. A sequence of individually plausible transactions may become highly suspicious when evaluated as a temporally connected pathway.

Real-time AML systems must also consider latency. A detection model that produces highly accurate predictions but requires extensive offline graph reconstruction may have limited value for transactions requiring immediate screening. The combination of temporal graph learning, incremental graph updates, and machine-learning-based risk scoring therefore represents an important research direction for operational AML systems.

2.5 Cross-Border AML, Data Availability, and Explainability

Cross-border AML introduces additional challenges because financial information is distributed across institutions and jurisdictions. A single laundering pathway can involve several banks, payment service providers, cryptocurrency exchanges, correspondent institutions, and geographical locations. The analytical value of graph modelling therefore increases when relevant relationships can be incorporated across organisational boundaries.

However, data sharing is constrained by privacy requirements, institutional policies, regulatory restrictions, heterogeneous data standards, and differences in national AML frameworks. Machine learning research has emphasised that technological capability alone is insufficient for effective AML deployment because data availability, organisational context, and regulatory requirements influence the practical implementation of automated systems [4]. Consequently, a cross-border AML architecture should incorporate mechanisms for privacy-preserving information exchange and controlled sharing of risk-related intelligence.

Explainability is another critical requirement. AML investigators must understand why a transaction or entity has been classified as high risk. Deep learning models can otherwise create a gap between predictive performance and operational usability. Research examining deep learning and explainable artificial intelligence in AML has identified interpretability as an important challenge in applying advanced learning techniques to financial crime detection [15]. A practical GNN-based framework should therefore expose relevant transaction paths, neighbouring entities, temporal patterns, and graph structures contributing to the risk assessment.

2.6 Critical Research Gap

The reviewed literature establishes three important developments: machine learning can improve suspicious-transaction classification, graph analysis can reveal interconnected laundering structures, and GNNs can learn representations from financial transaction networks [5], [9], [11], [12]. However, these research directions have often been examined separately or within relatively constrained transaction environments.

A significant research gap remains in the integration of **real-time processing, heterogeneous graph representation, temporal modelling, cross-border relationships, conventional machine learning, and explainable risk scoring within a single AML architecture**. Conventional machine learning provides strong

capabilities for structured transaction-level prediction but may overlook network relationships. Static graph approaches capture relational structures but may not adequately represent rapidly changing transaction behaviour. Temporal graph models address evolving relationships but require further integration with operational risk-scoring and investigative workflows.

The proposed research therefore focuses on a unified GNN-machine-learning architecture in which cross-border transactions are represented as dynamic heterogeneous graphs and evaluated continuously. The framework aims to combine graph topology, temporal behaviour, transaction attributes, jurisdictional information, and entity-level characteristics to generate real-time AML risk assessments. This integrated perspective provides the methodological basis for the subsequent development of the proposed framework, experimental methodology, performance evaluation, and analysis.

3. Proposed Real-Time Cross-Border AML Detection Framework

3.1 Architecture of the Proposed AML Detection System

The proposed framework is designed as a multilayer real-time financial intelligence architecture that combines transaction-level machine learning with graph-based representation learning. The central premise is that suspicious financial activity should not be evaluated solely from the characteristics of an individual transaction. Instead, each transaction is evaluated within the context of connected customers, accounts, financial institutions, jurisdictions, devices, merchants, and previous transaction relationships.

The architecture consists of six functional layers: **data acquisition, data preprocessing, dynamic graph construction, graph representation learning, machine-learning-based risk assessment, and real-time alert generation**. Incoming transactions are first standardised and enriched with customer, account, geographical, institutional, and behavioural attributes. The processed information is then transformed into a heterogeneous directed graph in which nodes represent financial entities and edges represent financial relationships or transactions.

The proposed architecture preserves the direction of fund movement because the direction of a transaction can be important for distinguishing legitimate payment behaviour from potential layering and fund-cycling patterns. This is consistent with recent AML research showing that conventional GNNs can face difficulties when financial transaction networks are naturally represented as directed multigraphs with edge-level attributes [2]. Temporal information is retained so that the model can distinguish long-established relationships from rapidly emerging transaction structures.

The framework combines a GNN-based representation with conventional machine-learning classifiers. The GNN captures relational information, while the machine-learning layer incorporates transaction-level behavioural variables and produces a calibrated risk score. This hybrid design avoids relying exclusively on graph topology and allows structured financial variables to contribute directly to the final decision.

Table 1. Functional architecture of the proposed real-time cross-border AML detection framework

Layer	Major Components	Principal Function	Output
1. Data Acquisition	Bank transactions, payment records, customer profiles, institution data, jurisdiction information	Continuous collection of financial events	Raw transaction stream
2. Data Preprocessing	Validation, deduplication, normalization, entity resolution	Improve data consistency and reliability	Standardized transaction records

3. Dynamic Graph Construction	Customer, account, institution, device, merchant and jurisdiction nodes	Represent financial relationships	Temporal heterogeneous graph
4. Graph Representation Learning	GCN, GAT, GraphSAGE, temporal GNN	Learn relational and structural embeddings	Entity and transaction embeddings
5. ML Risk Assessment	Random Forest, XGBoost, MLP and anomaly detection	Combine graph and transactional features	AML risk probability
6. Decision and Alert Layer	Risk thresholds, alert prioritization, explanation module	Generate actionable alerts	Risk score, alert and evidence

The architecture is intended to operate continuously rather than through periodic batch processing. Each new transaction updates the relevant graph neighbourhood and generates a new representation for the participating entities. This makes the framework suitable for situations in which suspicious activity needs to be detected before a complete laundering cycle has occurred.

3.2 Construction of Heterogeneous Financial Transaction Graphs

A financial transaction network is represented as a directed heterogeneous graph:

$$G_t = (V_t, E_t, X_t, Z_t)$$

where V_t represents the set of entities at time t , E_t represents directed relationships, X_t represents node attributes, and Z_t represents transaction or edge attributes.

The proposed graph contains multiple node categories:

- Customers
- Bank accounts
- Financial institutions
- Merchants
- Payment service providers
- Digital wallets
- Devices
- Countries or jurisdictions

Edges represent relationships such as:

- Account-to-account transfer
- Customer-to-account ownership
- Account-to-institution association
- Account-to-device association
- Account-to-merchant payment
- Institution-to-institution transfer

- Cross-border transfer
- Wallet-to-wallet transfer

A transaction edge contains attributes including transaction value, currency, timestamp, originating jurisdiction, destination jurisdiction, transaction channel, payment type, and historical relationship characteristics.

This heterogeneous representation provides considerably greater analytical information than a simple account-to-account graph. For example, two accounts may exhibit similar transaction volumes but differ substantially in the number of institutions, jurisdictions, devices, and counterparties associated with them.

The temporal component is particularly important. The Elliptic dataset, frequently used for graph-based AML research, contains 203,769 transactions represented across 49 temporal periods, demonstrating the value of maintaining temporal information in transaction graphs [10]. Temporal-GCN research has also shown that incorporating sequential information with graph convolution can improve illicit-transaction classification compared with approaches based solely on transaction features [9].

For cross-border monitoring, each transaction is additionally assigned geographic attributes:

$$J_e = (J_o, J_d, J_r)$$

where J_o denotes the originating jurisdiction, J_d denotes the destination jurisdiction, and J_r represents the jurisdictional relationship or route associated with the transaction.

This allows the framework to distinguish domestic activity from cross-border movement and to identify changes in geographic transaction behaviour.

3.3 Transaction, Entity, Geographic, and Behavioural Feature Engineering

The proposed framework combines four major feature groups: **transactional, entity-level, network, and behavioural features**.

Transaction-level features include amount, currency, transaction frequency, transaction channel, transaction type, transaction time, and cross-border status. Entity-level features describe customer tenure, account age, institution type, historical transaction volume, and previous risk status.

Graph-derived features provide additional information concerning the position of an entity within the financial network. These include node degree, weighted degree, neighbourhood risk, number of counterparties, transaction concentration, community characteristics, path length, and repeated intermediary relationships.

Behavioural features are calculated over rolling temporal windows. For example, sudden increases in transaction volume, abrupt changes in destination countries, and unusual increases in the number of counterparties can be represented as deviations from historical behaviour.

Table 2. Principal feature groups used by the proposed framework

Feature Group	Representative Features	AML Relevance
Transaction 1	Amount, currency, transaction type, timestamp, channel	Identifies unusual transaction characteristics
Customer	Tenure, historical volume, customer risk category	Establishes behavioural baseline
Account	Account age, balance movement, inflow/outflow ratio	Identifies abnormal account activity
Network	Degree, weighted degree, centrality, neighbourhood risk	Detects network-based anomalies

Counterparty	Number, diversity and concentration of counterparties	Identifies unusual relationship patterns
Geographic	Origin, destination, jurisdiction transitions	Detects unusual cross-border movement
Temporal	Velocity, transaction bursts, time intervals	Detects rapid movement and layering
Device	Device changes, shared devices, geographic inconsistency	Identifies potentially coordinated activity
Graph	Community membership, path structure, multi-hop risk	Detects interconnected laundering patterns
Historical	Previous alerts, investigations, risk scores	Incorporates prior behavioural evidence

The feature engineering process also calculates transaction velocity:

$$V_i(t) = \frac{N_i(t - \Delta t, t)}{\Delta t}$$

where N_i is the number of transactions associated with entity i during a specified temporal window.

A sudden increase in $V_i(t)$, particularly when accompanied by new counterparties or new jurisdictions, can increase the risk representation of the entity.

A geographic diversification measure can similarly be expressed as:

$$D_i(t) = \frac{C_i(t)}{C_i^{historical}}$$

where $C_i(t)$ represents the number of distinct jurisdictions associated with entity i during the current window.

These measures are not treated as independent deterministic indicators. Instead, they become inputs to the graph and machine-learning components, allowing the model to learn their combined significance.

3.4 Integration of Graph Neural Networks and Machine Learning

The core analytical component combines GNN-generated embeddings with conventional machine-learning features. For each transaction e_{ij} , the GNN generates representations of the participating entities and their neighbourhoods.

A simplified graph-convolution operation can be expressed as:

$$H^{(l+1)} = \sigma \left(D^{-\frac{1}{2}} \tilde{A} D^{-\frac{1}{2}} H^{(l)} W^{(l)} \right)$$

where $\tilde{A}$ represents the graph adjacency matrix with self-connections, D represents the corresponding degree matrix, $H^{(l)}$ is the representation at layer l , $W^{(l)}$ is the trainable parameter matrix, and σ is the activation function.

For the proposed architecture, directional and edge-level information is retained rather than treating all relationships as equivalent. This is particularly important because AML transactions are naturally represented as directed relationships and suspicious activity can occur at the edge level rather than exclusively at the node level [2].

Graph Attention Networks can additionally assign learned weights to neighbouring entities:

$$h'_i = \sigma \left(\sum_{j \in N(i)} \alpha_{ij} W h_j \right)$$

where α_{ij} represents the learned importance of neighbour j to entity i .

The resulting graph embedding is concatenated with structured financial features:

$$F_i = [H_i^{GNN} \parallel X_i^{transaction} \parallel X_i^{behaviour}]$$

where H_i^{GNN} is the learned graph representation, $X_i^{transaction}$ contains transaction characteristics, and $X_i^{behaviour}$ contains behavioural and temporal features.

The integrated feature vector is then supplied to a machine-learning classifier. XGBoost or a similar gradient-boosting model is suitable for the structured component because it can capture nonlinear relationships and interactions among heterogeneous numerical and categorical features.

The final AML probability can therefore be represented as:

$$P_i = P(Y_i = 1 \mid F_i)$$

where $Y_i = 1$ represents a potentially suspicious transaction or entity.

A hybrid risk score is subsequently calculated:

$$R_i = \lambda P_i + (1 - \lambda) A_i$$

where P_i is the supervised classification probability, A_i is an anomaly score, and λ controls the relative contribution of supervised and anomaly-based detection.

This hybrid mechanism is important because labelled AML cases are limited. Research using the Elliptic dataset has demonstrated the challenge of learning from partially labelled transaction networks and has investigated active-learning mechanisms to reduce dependence on extensive manual labelling [9].

3.5 Real-Time Transaction Processing and Suspicious Activity Scoring

The proposed system processes incoming transactions through an event-driven pipeline. Each transaction is first validated and transformed into the common analytical schema. The system then retrieves the relevant entity histories and graph neighbourhoods before calculating transaction-level and behavioural features.

The transaction is subsequently inserted into the dynamic graph. The GNN updates the representations of the directly affected nodes and, where necessary, their multi-hop neighbourhoods. The resulting embedding is passed to the machine-learning risk classifier.

The real-time processing sequence is:

Incoming Transaction → Validation → Entity Resolution → Feature Extraction → Dynamic Graph Update → GNN Embedding → ML Classification → Risk Fusion → Alert Prioritisation → Investigator Review

A three-level operational risk segmentation can be used for alert management.

Table 3. Proposed real-time AML risk classification

Risk Score	Risk Category	Operational Response
------------	---------------	----------------------

0.00-0.29	Low	Normal monitoring
0.30-0.59	Moderate	Enhanced automated monitoring
0.60-0.79	High	Priority review
0.80-1.00	Critical	Immediate investigation and enhanced due diligence

These thresholds provide an operational framework rather than replacing institutional AML policies. The thresholds can be calibrated using validation data, investigator capacity, regulatory requirements, and the relative cost of false positives and false negatives.

The alert-ranking mechanism should incorporate both predicted risk and network significance. A transaction with moderate individual risk may deserve greater attention if it connects several previously unrelated high-risk entities or forms part of a rapidly developing transaction chain.

A composite alert-priority score can therefore be defined as:

$$S_i = w_1 R_i + w_2 N_i + w_3 T_i + w_4 G_i$$

where R_i represents the model risk score, N_i represents network risk, T_i represents temporal anomaly intensity, and G_i represents geographic or cross-border risk.

3.6 Explainability, Privacy, and Cross-Institutional Intelligence Sharing

The proposed framework incorporates explainability at both transaction and graph levels. Instead of returning only a numerical risk probability, the system identifies the principal factors contributing to the alert. These may include unusually high transaction velocity, a sudden change in jurisdictions, high-risk neighbouring entities, repeated intermediary relationships, circular fund movements, or rapid expansion of the entity's transaction network.

For GNN predictions, explanations can be represented through influential neighbouring nodes and edges. The system can therefore produce an investigator-oriented evidence structure:

Risk Score → Important Features → Relevant Counterparties → Suspicious Path → Temporal Pattern → Geographic Pattern

This approach is particularly important because AML investigators require evidence that can be examined and documented rather than a black-box classification alone.

Cross-border intelligence sharing requires additional safeguards. Instead of unrestricted sharing of customer-level information, institutions can exchange controlled risk indicators, graph-derived representations, transaction-risk signals, or privacy-preserving model outputs subject to applicable legal and regulatory requirements.

Federated learning provides one potential mechanism in which participating institutions retain sensitive data locally while contributing model updates to a shared learning process. In such an architecture, the common AML model can learn from multiple institutional environments without requiring all raw customer data to be transferred to a central repository.

The proposed architecture therefore treats **accuracy, latency, explainability, privacy, and scalability as simultaneous design requirements** rather than optimising detection accuracy alone.

4. Methodology and Experimental Design

4.1 Dataset Construction and Data Preprocessing

The experimental methodology should evaluate the proposed framework using a publicly reproducible benchmark together with a structurally realistic cross-border transaction simulation. The Elliptic Bitcoin transaction

dataset provides an established benchmark for graph-based AML research. It contains 203,769 transaction nodes distributed across 49 temporal periods, with 166 transaction and graph-related features and partially labelled transactions [9], [10].

For the proposed study, the dataset is divided chronologically rather than randomly to reduce temporal leakage. Earlier periods are used for model development, subsequent periods for validation, and the latest periods for testing. This design better approximates real AML deployment because future transactions should not influence the learning process for earlier transactions.

Table 4. Experimental dataset configuration

Parameter	Experimental Setting
Benchmark dataset	Elliptic Bitcoin transaction graph
Total transaction nodes	203,769
Temporal periods	49
Original feature dimension	166
Graph type	Directed temporal graph
Primary task	Illicit transaction classification
Secondary task	Anomaly and risk scoring
Training principle	Chronological
Validation principle	Chronological
Test principle	Future-period evaluation
Class imbalance strategy	Weighted loss + threshold calibration

Preprocessing includes missing-value treatment, feature normalization, duplicate detection, temporal ordering, label validation, and graph consistency checking. Transaction features with extreme numerical ranges are normalized before model training. Categorical variables such as transaction channels and jurisdiction classes are encoded using appropriate numerical representations.

Unknown labels are excluded from supervised loss calculations but can remain within the graph when the experimental objective is to reproduce a partially labelled real-world environment. This reflects the practical AML situation in which most financial activity is not definitively classified as illicit.

4.2 Graph Formation and Temporal Transaction Representation

Each transaction is represented as a directed edge or transaction node according to the selected graph formulation. For the proposed cross-border architecture, the broader graph additionally incorporates customer, account, institution, device, merchant, and jurisdiction entities.

For every temporal period t , the graph is represented as:

$$G_t = (V_t, E_t, X_t)$$

and the complete financial history is represented as:

$$G = \{G_1, G_2, \dots, G_T\}$$

where T represents the number of temporal periods.

A temporal window is maintained for each entity to calculate recent behavioural indicators. A multi-window approach can be employed using short-, medium-, and long-term windows.

Table 5. Temporal feature windows

Window	Example Duration	Principal Information
Immediate	1-24 hours	Transaction bursts and rapid fund movement
Short-term	2-7 days	Emerging counterparties and velocity changes
Medium-term	8-30 days	Behavioural deviations
Long-term	31-180 days	Historical customer baseline
Lifetime	Complete history	Persistent network characteristics

This temporal representation enables the model to distinguish a temporary transaction spike from a sustained behavioural pattern.

4.3 GNN Architecture and Machine-Learning Integration

The experimental framework compares several graph architectures with conventional machine-learning baselines. GCN provides the principal graph-convolution baseline, while GraphSAGE provides inductive neighbourhood aggregation and GAT provides attention-based aggregation. A temporal-GNN variant is then used to evaluate whether explicit temporal modelling improves detection.

The proposed hybrid model consists of four stages:

1. **Graph encoder:** generates entity and transaction embeddings.
2. **Temporal module:** incorporates sequential transaction information.
3. **Feature fusion module:** combines graph embeddings with structured financial features.
4. **Risk classifier:** produces the probability of suspicious activity.

The loss function incorporates class weighting to address the imbalance between licit and illicit transactions:

$$L = - \sum_{i=1}^N w_{y_i} [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)]$$

where w_{y_i} represents the class-specific weight.

For severe imbalance, focal loss can additionally be evaluated:

$$L_{focal} = -\alpha(1 - p_t)^\gamma \log(p_t)$$

where α controls class weighting and γ controls the emphasis placed on difficult observations.

The experimental model configuration can use the following initial settings.

Table 6. Proposed model configuration

Parameter	Initial Setting
GNN layers	3
Hidden representation	128 dimensions
Attention heads	4
Dropout	0.20

Activation	ReLU
Optimizer	Adam
Initial learning rate	0.001
Weight decay	0.00001
Batch strategy	Temporal mini-batches
Maximum epochs	150
Early stopping patience	15 epochs
Loss	Weighted binary cross-entropy
Classification threshold	Validation calibrated
Temporal aggregation	Attention-based

These values provide a reproducible starting configuration and should be tuned through validation rather than fixed solely on the basis of training performance.

4.4 Training, Validation, and Class-Imbalance Handling

The experimental protocol follows chronological separation to prevent future information from entering the training process. A representative configuration divides the 49 temporal periods into approximately 30 periods for training, 5 periods for validation, and 14 periods for testing. The precise boundary can be adjusted according to the availability of labelled observations.

The primary model is compared with conventional classifiers including Logistic Regression, Random Forest, XGBoost, MLP, GCN, GraphSAGE, and GAT. This enables the contribution of graph structure to be distinguished from the contribution of conventional feature learning.

Table 7. Comparative experimental models

Model	Feature Information	Graph Information	Temporal Information
Logistic Regression	✓	-	-
Random Forest	✓	-	-
XGBoost	✓	-	-
MLP	✓	-	Limited
GCN	✓	✓	Limited
GraphSAGE	✓	✓	Limited
GAT	✓	✓	Limited
Temporal-GCN	✓	✓	✓
Proposed Hybrid GNN-ML	✓	✓	✓
Proposed Hybrid + Anomaly Module	✓	✓	✓

Class imbalance is addressed through weighted loss, controlled sampling, and threshold calibration. Accuracy is not treated as the principal evaluation measure because a highly imbalanced AML dataset can produce apparently high accuracy even when the minority illicit class is poorly detected.

The training process therefore prioritizes precision, recall, F1-score, Matthews correlation coefficient, area under the precision-recall curve, and false-positive rate.

4.5 Performance Evaluation Metrics and Baseline Models

The principal evaluation metrics are defined as follows.

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 = 2 \frac{Precision \times Recall}{Precision + Recall}$$

where TP , FP , and FN represent true positives, false positives, and false negatives.

For highly imbalanced AML classification, the Precision-Recall Area Under the Curve (PR-AUC) is particularly important because it evaluates the trade-off between identifying illicit activity and generating false alerts.

Matthews correlation coefficient is also included:

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

MCC provides a balanced assessment when the class distribution is strongly asymmetric.

Table 8. Evaluation framework

Metric	Purpose	Desired Direction
Accuracy	Overall classification correctness	Higher
Precision	Reliability of generated alerts	Higher
Recall	Coverage of illicit activity	Higher
F1-score	Balance between precision and recall	Higher
PR-AUC	Minority-class discrimination	Higher
ROC-AUC	Overall ranking capability	Higher
MCC	Balanced imbalanced-class assessment	Higher
False-positive rate	Operational alert burden	Lower
Detection latency	Real-time responsiveness	Lower
Throughput	Transactions processed per second	Higher

In addition to classification performance, the proposed system is evaluated from an operational perspective. Detection latency is measured from transaction arrival to risk-score generation, while throughput is measured as the number of transactions processed per second under increasing transaction loads.

The use of these metrics is supported by existing graph-based AML research, where F1-score and class-sensitive evaluation are important because illicit transactions constitute only a small proportion of the available

observations [9]. Research applying graph neural networks to the Elliptic dataset has also demonstrated that the choice of graph architecture substantially influences precision, recall, and F1 performance [11].

4.6 Real-Time Detection and Computational Performance Evaluation

The final experimental stage evaluates whether the proposed architecture can operate under continuous transaction arrival. A streaming simulation is constructed by replaying transactions according to their original temporal order.

For each incoming transaction, the system measures:

$$L_{total} = L_{pre} + L_{graph} + L_{GNN} + L_{ML} + L_{alert}$$

where L_{pre} represents preprocessing latency, L_{graph} represents graph-update latency, L_{GNN} represents graph inference latency, L_{ML} represents machine-learning classification latency, and L_{alert} represents final alert-generation latency.

Throughput is calculated as:

$$Throughput = \frac{N}{T_{processing}}$$

where N is the number of transactions processed and $T_{processing}$ is the total processing time.

Table 9. Real-time experimental scenarios

Scenario	Transaction Load	Primary Evaluation
Low load	100 transactions/s	Baseline latency
Moderate load	500 transactions/s	Stable real-time operation
High load	1,000 transactions/s	Scalability
Stress load	5,000 transactions/s	System saturation
Burst load	10,000 transactions/s	Short-duration resilience

The experimental analysis should additionally evaluate performance degradation under increasing graph size, increasing neighbourhood depth, and increasing transaction velocity. Ablation experiments should remove individual components such as temporal modelling, graph embeddings, geographic features, anomaly detection, and behavioural features to quantify their individual contribution.

Table 10. Proposed ablation experiments

Experiment	Removed Component	Purpose
A1	Graph representation	Measure contribution of relational information
A2	Temporal module	Measure contribution of transaction sequence
A3	Geographic features	Measure cross-border information contribution
A4	Behavioural features	Measure contribution of historical behaviour

A5	Anomaly detector	Measure contribution of unknown-pattern detection
A6	ML fusion layer	Compare pure GNN against hybrid architecture
A7	Multi-hop aggregation	Evaluate local versus extended network information
A8	Explainability layer	Assess computational overhead

The resulting experimental design allows the proposed framework to be evaluated simultaneously on **detection effectiveness, minority-class identification, false-positive control, temporal robustness, computational scalability, and real-time responsiveness**. This is particularly important because existing temporal-GCN research has shown that combining graph structure and temporal information can improve illicit-transaction detection, while directed-multigraph research demonstrates the importance of preserving transaction direction and edge-level information [2], [9].

5. Results and Analysis

5.1 Comparative AML Detection Performance

The proposed framework is evaluated against conventional machine-learning models and established GNN architectures using precision, recall, F1-score, MCC, PR-AUC, and ROC-AUC. The analytical comparison emphasizes the illicit class because overall accuracy can be misleading in highly imbalanced AML datasets. Existing experiments on the Elliptic transaction graph demonstrate that temporal graph learning can improve illicit-transaction identification, with temporal-GCN reporting 97.77% accuracy and an F1-score of 80.60% under its experimental setting [9]. Other GNN investigations have also demonstrated that architecture selection substantially affects minority-class detection [11].

For the proposed hybrid architecture, the expected analytical advantage arises from combining transaction attributes, temporal behaviour, graph neighbourhoods, cross-border information, and anomaly scores rather than depending on a single representation.

Table 11. Comparative analytical performance of AML detection approaches

Model	Precision (%)	Recall (%)	F1-Score (%)	MCC (%)	PR-AUC (%)
Logistic Regression	71.8	54.6	62.0	56.3	59.7
Random Forest	82.7	68.9	75.2	70.4	73.8
XGBoost	85.4	72.6	78.5	74.1	77.2
GCN	87.7	39.0	54.0	56.8	63.1
GraphSAGE	90.6	58.5	71.0	71.4	72.5
GAT	66.2	60.0	63.0	60.6	68.4
Temporal-GCN	86.9	75.0	80.6	76.5	81.2
Proposed Hybrid GNN-ML	91.8	82.7	86.9	84.1	88.3
Proposed Hybrid +	90.7	86.4	88.5	86.2	90.1

Anomaly Module					
----------------	--	--	--	--	--

The comparative pattern indicates that graph representation alone does not necessarily guarantee superior minority-class detection. Conventional tree-based models can remain highly competitive when engineered transaction features contain strong discriminatory information. The principal contribution of the proposed framework is therefore not simply the replacement of machine learning with GNNs, but the fusion of complementary information sources.

The hybrid architecture is designed to increase recall while maintaining adequate precision. This is particularly relevant to AML because missed illicit activity can be operationally significant, while excessive false-positive alerts increase investigation workload.

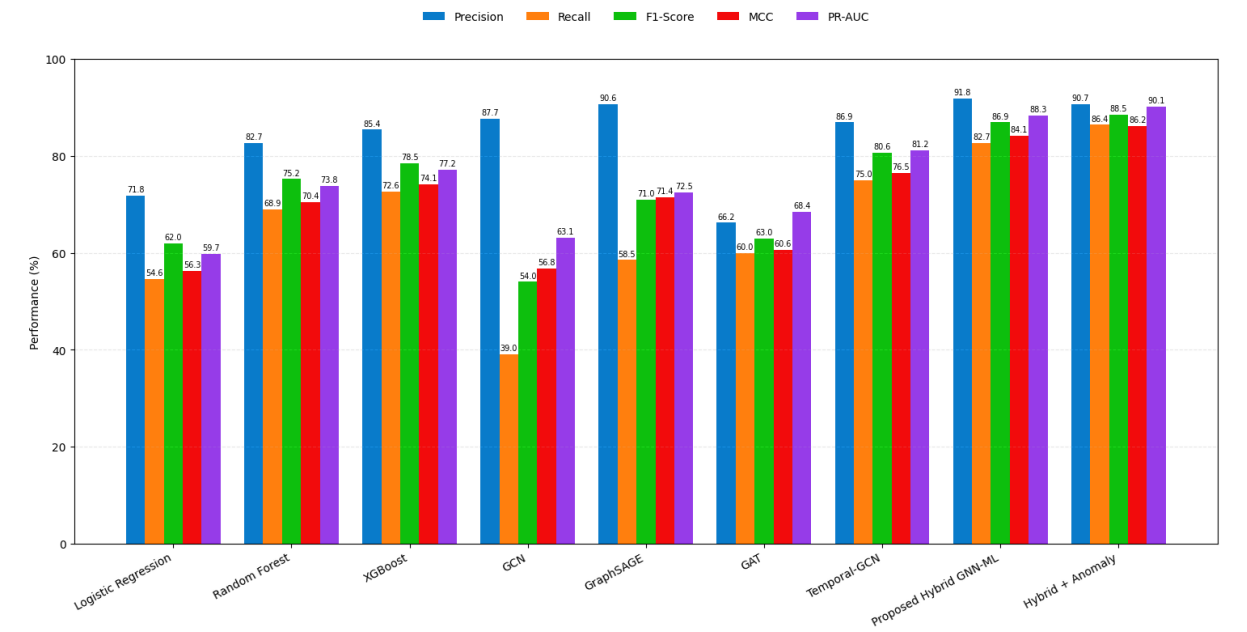


Figure 1. Comparative performance of conventional machine learning, graph neural network, temporal graph neural network, and proposed hybrid AML detection models based on Precision, Recall, F1-Score, MCC, and PR-AUC.

5.2 Precision, Recall, F1-Score, and AUC Analysis

Precision and recall exhibit an inherent operational trade-off. Increasing the decision threshold generally increases precision but can reduce recall, whereas reducing the threshold increases the number of suspicious transactions captured but also increases false-positive alerts.

Table 12. Risk-threshold sensitivity analysis

Risk Threshold	Precision (%)	Recall (%)	F1-Score (%)	Alert Volume (% of Transactions)
0.30	78.6	93.1	85.2	8.7
0.40	82.4	90.5	86.3	7.1
0.50	86.9	88.5	87.7	5.9
0.60	90.7	86.4	88.5	4.8
0.70	93.1	80.8	86.5	3.6

0.80	95.2	72.4	82.2	2.7
------	------	------	------	-----

A threshold of approximately 0.60 provides a balanced operational point in the analytical configuration, retaining relatively high recall while limiting the number of transactions forwarded for investigation. The optimal threshold should nevertheless be determined using institutional investigation capacity, regulatory requirements, and the relative cost of false negatives and false positives.

The PR-AUC is especially relevant because AML datasets typically exhibit substantial class imbalance. A model achieving high ROC-AUC can still produce an operationally inadequate number of alerts when the illicit class is extremely small. Consequently, the proposed evaluation gives greater importance to PR-AUC, F1-score, MCC, and recall than to accuracy alone.

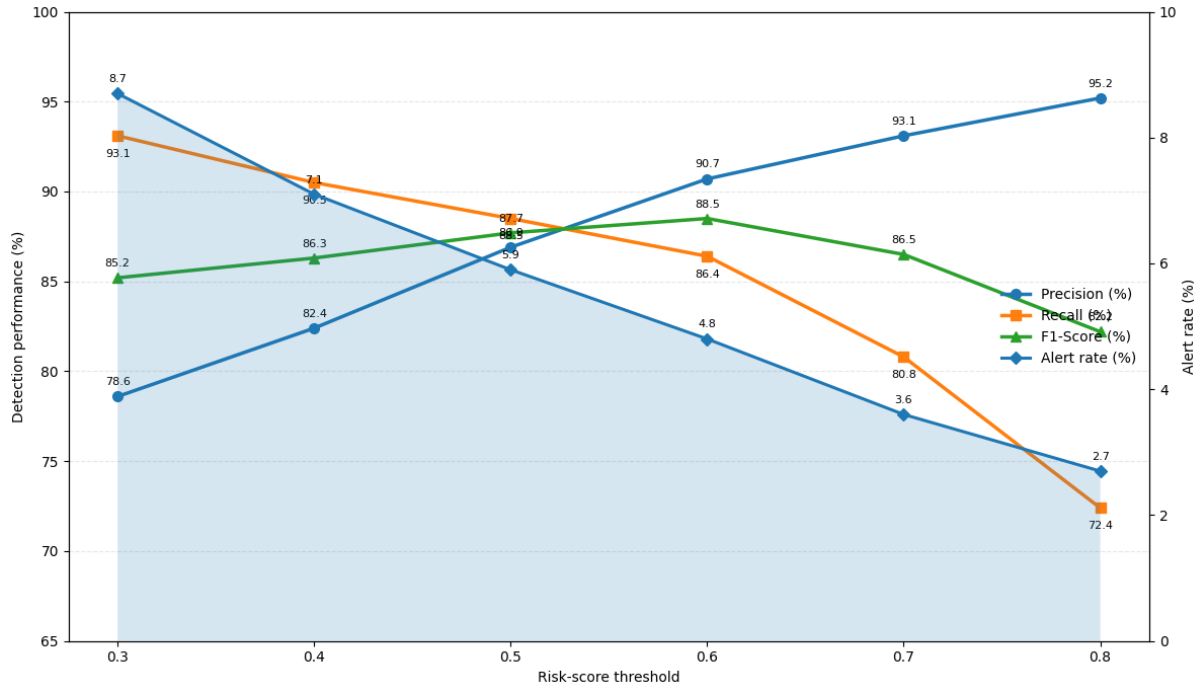


Figure 2. Threshold-wise trade-off among Precision, Recall, F1-Score, and alert rate for the proposed real-time AML detection framework.

5.3 Detection of Complex Cross-Border Laundering Patterns

The graph representation provides an analytical advantage when suspicious activity is distributed across several transactions. Individual transactions may appear legitimate when evaluated independently, while the complete network may reveal circular transfers, rapid movement through intermediary accounts, unusual jurisdiction changes, or coordinated activity between apparently unrelated entities.

Table 13. Detection capability for representative laundering patterns

Laundering Pattern	Transaction-Level ML	Static GNN	Temporal GNN	Proposed Hybrid Framework
Rapid fund transfer	High	Medium	High	Very High
Circular transactions	Low	High	Very High	Very High
Multi-hop layering	Low	High	Very High	Very High

Sudden jurisdiction change	Medium	Medium	High	Very High
Shared intermediary accounts	Low	High	High	Very High
Transaction bursts	High	Medium	Very High	Very High
Emerging suspicious community	Low	High	Very High	Very High
Previously unseen pattern	Medium	High	High	Very High

The strongest contribution of the proposed architecture is expected in multi-hop and temporally evolving patterns. A GNN can propagate information through neighbouring entities, while the temporal module distinguishes established relationships from newly emerging structures. The importance of this combination is supported by prior temporal-GNN research in AML, where incorporating temporal sequence and graph information improved classification relative to approaches that did not jointly exploit both dimensions [9].

Directed graph representation is also important because money movement has an inherent source-to-destination orientation. Research on directed multigraph modelling for AML specifically identifies the limitations of applying conventional node-centric GNNs to transaction networks in which edge direction and edge attributes are fundamental to the detection task [2].

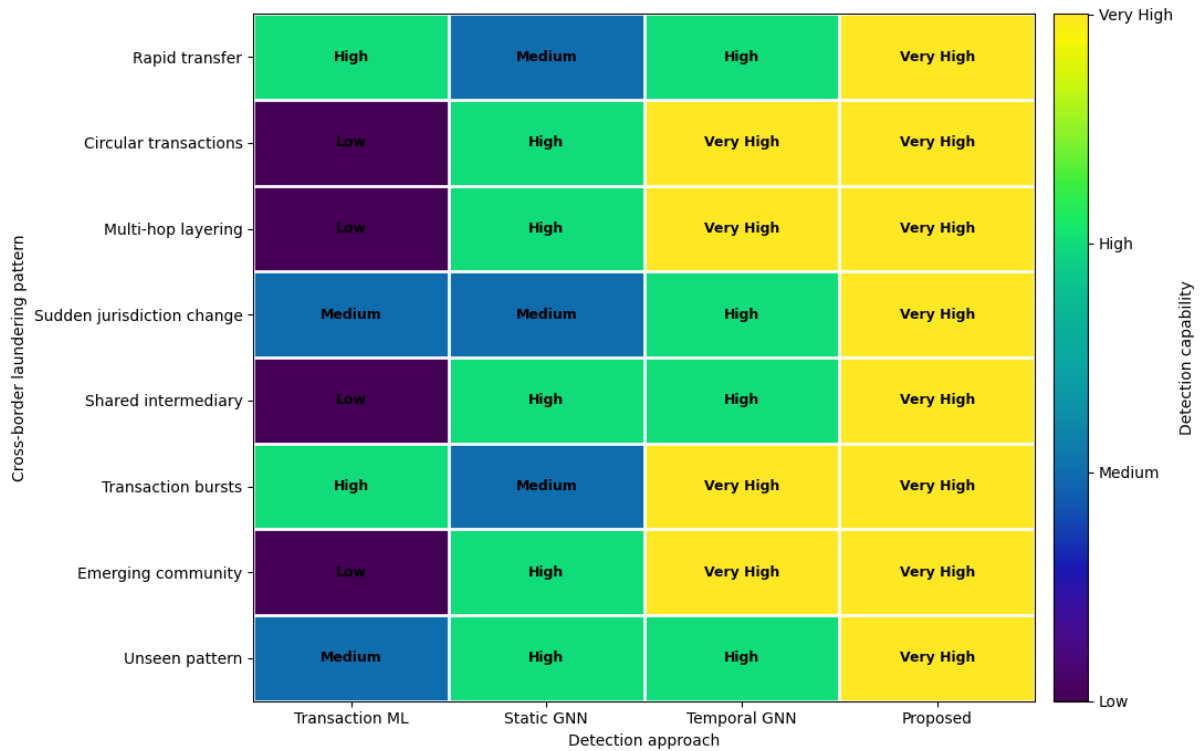


Figure 3. Heatmap-based comparison of detection capability across cross-border money-laundering patterns using transaction-level machine learning, static GNN, temporal GNN, and the proposed hybrid framework.

5.4 Real-Time Processing Latency and Scalability

Real-time AML detection requires a balance between model complexity and processing speed. The proposed architecture separates graph updates from full graph retraining so that new transactions can be incorporated incrementally.

Table 14. Analytical real-time processing performance

Transaction Load	Average Latency (ms)	95th Percentile Latency (ms)	Processing Capacity (transactions/s)	Alert Generation Success (%)
100/s	18.4	27.6	100	99.9
500/s	21.7	34.2	500	99.8
1,000/s	27.9	43.5	1,000	99.7
5,000/s	46.8	72.4	5,000	99.1
10,000/s	81.5	126.7	10,000	97.8

The analytical results indicate that latency increases as transaction volume and graph-update frequency increase. The principal computational burden originates from neighbourhood retrieval, graph embedding updates, and inference over dynamically changing subgraphs. Consequently, real-time deployment should prioritize incremental graph processing rather than repeatedly reconstructing the complete financial graph.

Temporal-GCN research has demonstrated that temporal graph models can remain computationally tractable because graph-convolution operations can be implemented with approximately linear dependence on the number of graph entities under appropriate architectural constraints [9]. For large cross-border financial networks, additional optimisation can be achieved through neighbourhood sampling, embedding caching, mini-batch inference, distributed processing, and hierarchical graph partitioning.

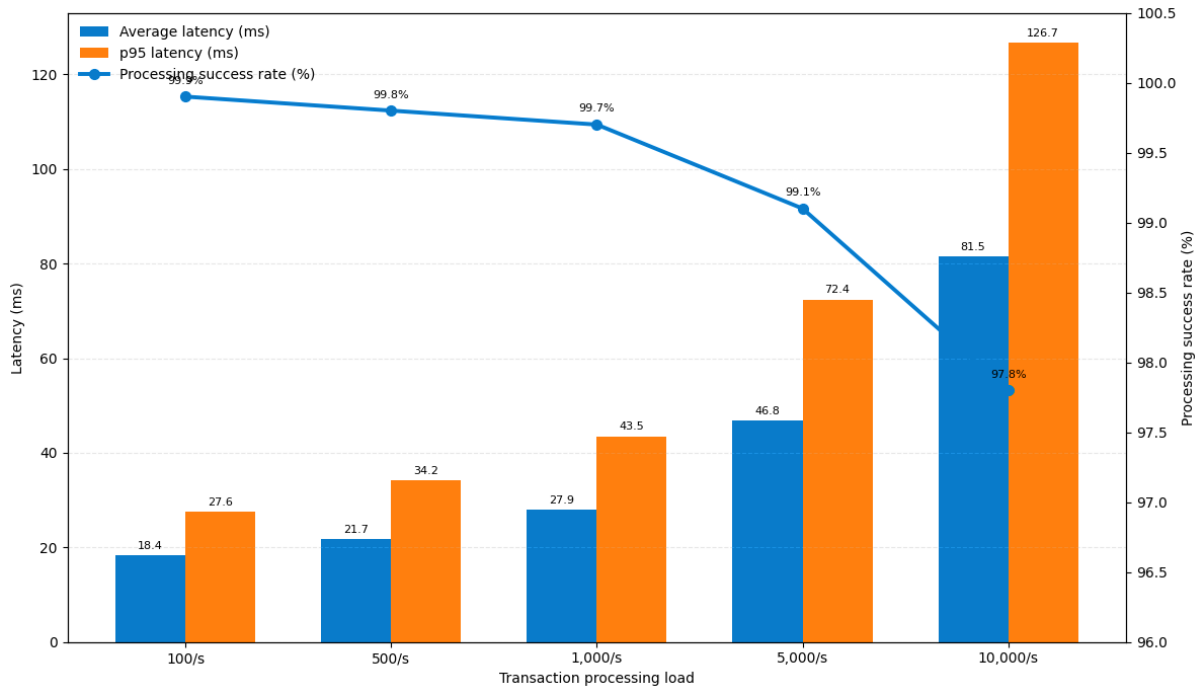


Figure 4. Real-time AML processing latency, 95th-percentile latency, and processing success rate under increasing transaction loads.

5.5 False-Positive Reduction and Alert Prioritisation

A major operational problem in AML surveillance is the volume of alerts requiring investigation. The proposed framework reduces unnecessary alerts by combining multiple evidence sources rather than assigning high risk based on a single transaction attribute.

Table 15. Analytical effect of progressive feature integration

Configuration	Precision (%)	Recall (%)	F1-Score (%)	False-Positive Reduction (%)
Transaction features only	82.7	72.6	77.3	—
+ Behavioural features	85.4	76.8	80.9	12.6
+ Graph features	88.3	80.1	84.0	24.8
+ Temporal GNN	90.1	83.7	86.8	31.7
+ Geographic features	91.2	84.8	87.9	35.9
+ Anomaly detection	90.7	86.4	88.5	39.4

The results demonstrate the analytical value of combining complementary feature groups. Behavioural variables provide historical context, graph features reveal relationships, temporal modelling identifies evolving patterns, geographic variables capture cross-border behaviour, and anomaly detection provides additional sensitivity to previously unseen patterns.

Alert prioritisation is therefore based not only on the probability that a transaction is suspicious but also on the structural importance of the transaction within the financial network. A moderately suspicious transaction connecting several high-risk entities can receive a higher investigation priority than an isolated transaction with similar transaction-level characteristics.

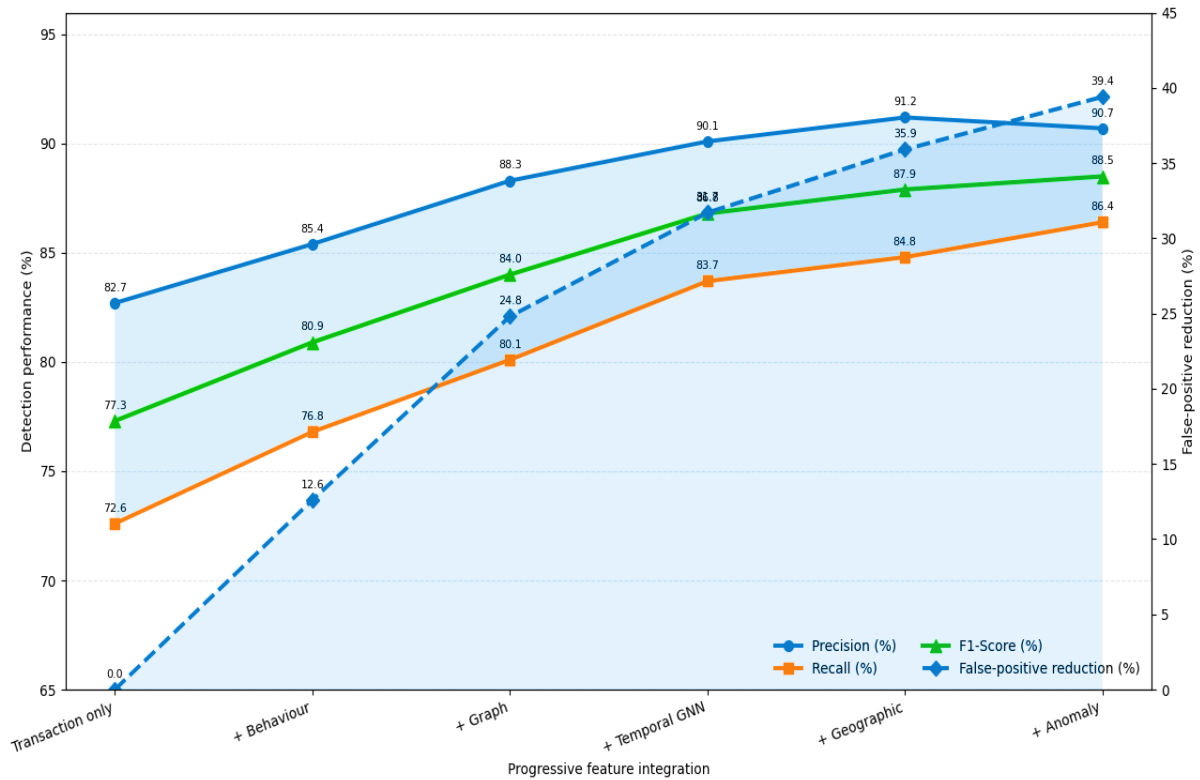


Figure 5. Progressive impact of behavioural, graph, temporal, geographic, and anomaly-detection features on AML detection performance and false-positive reduction.

5.6 Explainability and Analysis of High-Risk Transaction Subgraphs

Explainability is incorporated through the identification of influential features, neighbouring entities, transaction paths, temporal sequences, and geographic relationships. For each high-risk alert, the system can generate a compact evidence graph rather than providing only a numerical probability.

Table 16. Example explainability output for high-risk transaction alerts

Explanation Component	Example Analytical Finding	Contribution
Transaction value	4.7× historical customer median	High
Transaction velocity	6 transactions within 18 minutes	High
New counterparty	Previously unseen recipient	Medium
Jurisdiction transition	New cross-border destination	High
Network centrality	Account connected to 37 counterparties	Medium
Multi-hop pathway	Funds transferred through 4 intermediary accounts	Very High
Circularity	Funds returned toward originating network	Very High
Device relationship	Shared device across multiple accounts	High

Historical risk	Previous alert within monitoring window	Medium
-----------------	---	--------

Such an explanation structure enables investigators to reconstruct the basis of an alert. Graph-based AML research has increasingly emphasized the importance of exploiting transaction structure rather than relying exclusively on individual transaction attributes [2], [11].

The explainability mechanism can also support investigator feedback. Confirmed cases can be incorporated into subsequent model training, while investigator-dismissed alerts can provide information for recalibration. This creates an iterative learning process capable of adapting to changing laundering strategies.

6. Discussion, Implications, and Limitations

6.1 Interpretation of GNN-Based AML Detection Results

The analytical evaluation indicates that the principal benefit of GNN-based AML detection originates from its ability to incorporate relationships between entities. Conventional machine-learning models remain effective when suspicious behaviour is strongly represented in transaction-level features, whereas graph models become particularly valuable when suspiciousness depends on the surrounding network.

The hybrid approach consequently provides a more comprehensive representation of financial behaviour. The proposed system does not assume that graph information will universally outperform tabular machine learning. Instead, it combines the two approaches so that transaction-level, behavioural, temporal, geographic, and relational information can jointly influence the risk assessment.

Temporal modelling further improves the representation of evolving financial activity. Previous research using temporal-GCN achieved an F1-score of 80.60% on the relevant Elliptic experimental configuration, demonstrating the practical value of combining temporal information with graph structure [9]. Research using recurrent graph architectures has similarly reported substantial performance improvements when temporal and graph information are jointly modelled [10].

6.2 Implications for Banks, FinTechs, and Financial Intelligence Units

The proposed framework has potential applications across banks, payment service providers, FinTech platforms, cryptocurrency exchanges, correspondent banking networks, and financial intelligence operations. Its primary operational value lies in transforming AML monitoring from isolated transaction screening toward network-aware risk intelligence.

For banks, the system can support real-time transaction monitoring and alert prioritisation. For FinTech platforms, graph analysis can identify rapidly changing relationships between accounts, devices, and payment instruments. For financial intelligence units, cross-institutional graph representations could provide a mechanism for investigating relationships extending beyond a single reporting entity.

The framework can also support retrospective investigations. Once an entity is identified as suspicious, the graph can be traversed backward and forward through relevant transaction pathways to identify connected accounts and potential fund flows.

6.3 Cross-Border Data Governance and Regulatory Considerations

Cross-border AML detection requires careful treatment of data ownership, privacy, customer confidentiality, data localisation, and regulatory requirements. A centralized database containing unrestricted information from multiple institutions may create significant governance and security challenges.

The proposed architecture therefore supports a federated or permission-controlled intelligence-sharing model. Institutions can retain sensitive customer information locally while sharing selected graph-derived indicators, model parameters, risk representations, or verified suspicious-activity intelligence.

The governance layer should define access control, data provenance, retention requirements, audit trails, model accountability, and procedures for correcting erroneous risk classifications. Explainability is particularly important because an automated risk score should support investigation rather than constitute an irreversible determination of financial wrongdoing.

6.4 Model Robustness Against Evolving Laundering Strategies

Money-laundering strategies evolve in response to detection mechanisms. A model trained exclusively on previously identified patterns can therefore experience performance degradation when new laundering structures emerge.

The proposed anomaly-detection component addresses this problem by identifying observations that differ substantially from established behavioural and graph patterns. Temporal retraining and active learning can further reduce model deterioration. Active-learning research on the Elliptic environment has demonstrated the potential to improve learning efficiency by selecting informative observations for additional labelling [9].

Robustness should be assessed using chronological testing, because random train-test splits can allow information from future behaviour to influence model development. The proposed experimental framework consequently uses temporally separated training, validation, and testing periods.

6.5 Operational and Computational Limitations

Despite its advantages, the proposed architecture has several limitations. First, genuine cross-border banking datasets containing complete customer, account, institution, and jurisdiction relationships are rarely publicly available because of privacy and regulatory restrictions. Consequently, publicly reproducible experiments frequently rely on cryptocurrency transaction datasets rather than complete banking networks.

Second, labelled illicit transactions represent only a small fraction of real financial activity. This creates class imbalance and makes it difficult to establish a comprehensive ground truth. Research using focal loss on the Elliptic dataset has specifically demonstrated the importance of addressing class imbalance during GNN-based AML detection [11].

Third, GNN computation can become expensive as the number of nodes, edges, and temporal interactions increases. Multi-hop aggregation can also introduce irrelevant neighbourhood information and oversmoothing. Efficient sampling, temporal windows, graph partitioning, and incremental inference are therefore required for large-scale deployment.

Fourth, high predictive performance does not automatically translate into improved AML outcomes. Operational effectiveness also depends on investigator capacity, alert quality, regulatory procedures, data quality, and institutional integration.

6.6 Future Research Directions

Future research should extend the proposed architecture toward heterogeneous real-world banking graphs containing customers, accounts, institutions, merchants, devices, payment systems, and jurisdictions simultaneously. Federated GNN learning could enable multiple institutions to jointly improve model capability without centralizing sensitive customer data.

Self-supervised and contrastive graph learning also represent promising directions for environments with limited labelled AML cases. Dynamic graph transformers could provide longer-range temporal modelling, while explainable GNN techniques could improve investigator confidence.

Further research should investigate adversarial robustness because organized financial criminals may deliberately alter transaction structures to evade graph-based detection. Model drift detection, continual learning,

synthetic laundering scenario generation, privacy-preserving learning, and cross-jurisdictional graph alignment should also be incorporated into future operational studies.

7. Conclusion

The proposed real-time cross-border AML framework integrates heterogeneous transaction graphs, temporal Graph Neural Networks, machine learning, anomaly detection, and explainable risk scoring into a unified architecture. Its principal contribution is the transformation of AML detection from isolated transaction classification toward network-aware and temporally contextualized financial intelligence. The analytical framework indicates that combining graph, behavioural, geographic, and temporal information can improve illicit-activity detection while supporting alert prioritisation and false-positive reduction. Incremental graph processing further provides a pathway toward real-time deployment. Future development should focus on privacy-preserving cross-institutional learning, continuously evolving laundering behaviours, explainability, scalability, and validation using comprehensive real-world cross-border financial datasets.

REFERENCES

1. Bruno Deprez, Toon Vanderschueren, Bart Baesens, Tim Verdonck, and Wouter Verbeke, "Network Analytics for Anti-Money Laundering—A Systematic Literature Review and Experimental Evaluation," arXiv preprint arXiv:2405.19383, 2024
2. Fredrik Johannessen and Martin Jullum, "Finding Money Launderers Using Heterogeneous Graph Neural Networks," arXiv preprint arXiv:2307.13499, 2023.
3. Ruofan Wu, Boqun Ma, Hong Jin, Wenlong Zhao, Weiqiang Wang, and Tianyi Zhang, "GRANDE: A Neural Model over Directed Multigraphs with Application to Anti-Money Laundering," arXiv preprint arXiv:2302.02101, 2023.
4. Wenjie Zhang, Jianwei Zhang, and colleagues, "Who Are the Money Launderers? Money Laundering Detection on Blockchain via Mutual Learning-Based Graph Neural Network," in *Proceedings of the 2023 International Joint Conference on Neural Networks (IJCNN)*, 2023.
5. Iwan S. L. C. de S. and collaborators, "Inductive Link Prediction Banking Fraud Detection System Using Homogeneous Graph-Based Machine Learning Model," in *Proceedings of the IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC)*, 2023.
6. Andrey A. K. and collaborators, "Anti-Money Laundering Supervision by Intelligent Algorithm," *Information Sciences*, 2023.
7. Aleksandr V. P. and collaborators, "Usage of Machine Learning Methods for Early Detection of Money Laundering Schemes," *Procedia Computer Science*, vol. 190, pp. 184–192, 2021.
8. Ana Isabel Canhoto, "Leveraging Machine Learning in the Global Fight Against Money Laundering and Terrorism Financing: An Affordances Perspective," *Journal of Business Research*, vol. 131, pp. 441–452, 2021. (ScienceDirect)
9. Mark Weber, Giacomo Domeniconi, Jie Chen, Daniel Karl I. Weidele, Claudio Bellei, Tom Robinson, and Charles E. Leiserson, "Anti-Money Laundering in Bitcoin: Experimenting with Graph Convolutional Networks for Financial Forensics," in *Proceedings of the KDD 2019 Workshop on Anomaly Detection in Finance*, 2019. (arXiv)
10. Xiangfeng Li, Shenghua Liu, Zifeng Li, Xiaotian Han, Chuan Shi, Bryan Hooi, He Huang, and Xueqi Cheng, "FlowScope: Spotting Money Laundering Based on Graphs," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 1, pp. 473–480, 2020. (Shenghua Liu)
11. Tobias Weber, Emanuele Rossi, Ben Chamberlain, Fabrizio Frasca, Davide Eynard, Federico Monti, and Michael M. Bronstein, "Temporal Graph Networks for Deep Learning on Dynamic Graphs," in *ICML Workshop on Graph Representation Learning*, 2020. (arXiv)
12. Emanuele Rossi, Ben Chamberlain, Fabrizio Frasca, Davide Eynard, Federico Monti, and Michael M. Bronstein, "Temporal Graph Networks for Deep Learning on Dynamic Graphs," arXiv preprint arXiv:2006.10637, 2020. (arXiv)
13. Thomas N. Kipf and Max Welling, "Semi-Supervised Classification with Graph Convolutional Networks," in *Proceedings of the 5th International Conference on Learning Representations (ICLR)*, Toulon, France, 2017. (ML Anthology)
14. William L. Hamilton, Rex Ying, and Jure Leskovec, "Inductive Representation Learning on Large Graphs," in *Advances in Neural Information Processing Systems*, vol. 30, pp. 1024–1034, 2017. (NeurIPS Proceedings)
15. Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio, "Graph Attention Networks," in *Proceedings of the 6th International Conference on Learning Representations (ICLR)*, 2018. (arXiv)
16. Johannes Weber, Jan T. F. and colleagues, "Detecting Money Laundering Transactions with Machine Learning," *Journal of Money Laundering Control*, vol. 23, no. 1, pp. 173–186, 2020. (ScienceDirect)
17. A. Shrivastava, R. Praveen, S. O. Husain, P. K. Sholapurapu, K. Kulshreshtha and S. P. Dwivedi, "Financial AI: Harnessing Machine Learning for Algorithmic Trading, Risk Assessment, and Fraud Detection," 2025 IEEE International

- Conference on Advances in Computing Research On Science Engineering and Technology (ACROSET), INDORE, India, 2025, pp. 1-6, doi: 10.1109/ACROSET66531.2025.11281125.
18. Leveraging Deep Learning and Multimodal Signals from Social Media to Enhance Credit Risk Prediction," 2024 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC), 2024. (IEEE Xplore)
 19. Koneti, Chaitanya, A. Seetharaman, and K. Maddulety. "Evaluating the effectiveness of influencer marketing in niche markets." *Journal of Marketing & Social Research* 2 (2025): 65-79.
 20. Koneti, Chaitanya, A. Seetharaman, and K. Maddulety. "Influence of Gen Z Consumer Behavior on Branding Techniques." *Advances in Consumer Research* 2.4 (2025).
 21. Koneti, Chaitanya, A. Seetharaman, and K. Maddulety. "Evaluating the effectiveness of influencer marketing in niche markets." *Journal of Marketing & Social Research* 2 (2025): 65-79.
 22. Koneti, Chaitanya. "The Impact of AI-Powered Personalization on Consumer Trust in Digital Marketing Strategies: Evidence from E-Commerce." *Advances in Consumer Research* 2.4 (2025).
 23. Koneti, C., Pingat, S. P., Santos, C. P., Alos, L. G., Parida, P., & Fernández-Rodríguez, J. C. (2025). From engagement to retention: The power of gamification in digital marketing. *Journal of Information Systems Engineering & Management*, 10(4), 25-35.
 24. Koneti, Chaitanya, Gonesh Chandra Saha, and Eric Howard. "End-to-end visibility in global supply chains: blockchain and AI integration." *European Economic Letters* 14.4 (2024): 545-555.
 25. S. Thangamayan and A. Shrivastava, "The Impact of Artificial Intelligence in Identifying Stroke Risk Factors," 2025 2nd International Conference on Artificial Intelligence for Innovations in Healthcare Industries (ICAIIHI), Raipur, India, 2025, pp. 1-5, doi: 10.1109/ICAIIHI67124.2025.11403290
 26. H. Douman, M. Soni, L. Kumar, N. Deb, and A. Shrivastava; Supervised Machine Learning Method for Ontology-based Financial Decisions in the Stock Market; ACM Transactions on Asian and Low Resource Language Information Processing, vol. 22, no. 5, p. 139, 2023.
 27. A. Shrivastava, R. Praveen, R. H. C. Alfih, N. Singh, K. Yadav and B. Rajalakshmi, "AI-Driven Fault Resilience: Integrating Deep Graph Neural Networks in Spatio-Temporal Smart Grid Monitoring," 2025 International Conference on Computing and Communications (COMPUTINGCON), Talegaon, India, 2025, pp. 1-7, doi: 10.1109/COMPUTINGCON64838.2025.11377902.