International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Machine Learning Architectures for Real-Time Fraud Prevention in High-Velocity Financial Networks

Anuraag Mangari Neburi

Submitted:29/10/2025 **Accepted:** 16/12/2025 **Published:** 17/12/2025

Abstract: In the paper, it is suggested to implement a real-time architecture of a fraud detection system that can apply to the current digital payment systems. It is founded on supervised learning, graph neural networks, and anomaly detection and reinforcement learning to improve flexibility and accuracy. It is a multimodal system in which it is based on details of transactions, device patterns, behavior and account network sequences. The results of the experiment are more accurate and recollections than the traditional models with low latency that can be applied to the real time in finance. The architecture is also easy to adapt to the unexpected changes in the fraud trends such as account takeovers and mules networks. The proposed framework is a more rationalized, large-scale and explainable framework of fraud detection.

Keywords: Finance, Machine Learning, Velocity, Fraud, Architecture

I. INTRODUCTION

Online payments are increasing in speed and are being used more in implementing financial fraud. Even the traditional methods of fraud detection, like the rule-based system or the basic machine learning model, are not always up to date with new patterns of fraud. They also adapt slowly whenever the behavior of the fraudsters alters.

The proposed paper presents a novel real-time fraud architecture that employs the application of several types of data and sophisticated learning architectures. It is a combination of graph analysis, anomaly detection, and reinforcement learning that comprehends the complex fraud activities. It is intended to create a more precise, quicker as well as more ready to adapt to changing fraud conditions.

II. RELATED WORKS

Graph-Based Machine Learning for Fraud Detection

The fast development of financial fraud has compelled researchers to consider graph-based machine learning algorithms due to the fact that the conventional models of frauds do not represent relative and dynamic associations found in the contemporary financial networks. The early rule-based systems, along with models of a static machine learning, tend to fail when the structure of the fraud is modified, or when criminals mimic regular behavior.

Recently, it has been demonstrated that Graph Neural Networks (GNNs) are powerful at learning entity relationships (including devices, accounts, merchants, and transaction paths). This change can be seen through the BRIGHT framework that can be reconfigured to work in a real-time setting using GNNs [1]. BRIGHT addresses two typical problems, namely, the danger of processing futuristic information during GNN message passing, and the excessive latency of graph queries during real-time inference.

The system separates the incorporation of updates with online prediction, which allows rapid fraud scoring. Their implementation demonstrates that it is possible to scale GNNs to production scale systems that are more than 75X faster and more than 2X more accurate. This article emphasizes the fact that the graph representations enhance the capability of identifying multi-hop risk propagation which is not achieved using rule-based or feature-based models.

The need to learn more graph representations in order to identify hidden fraud relations is also highlighted by other researchers. Adaptive neighbor sampling is one of the directions such as ASA-GNN [2]. The model is interested in picking meaningful neighbors of a node according to their behavioral similarity and eliminating noise that is a common occurrence in large and dense financial graphs.

Vice President

With cosine similarity and edge weights, ASA-GNN can keep the most useful context of every transaction and its neighbor diversity measurement can prevent the problem of oversmoothing that is typical of GNNs. This is particularly critical in financial fraud in which there are larger numbers of real transactions in the dataset, and the fraudulent nodes are trying to conceal themselves. ASA-GNN demonstrates the improvement in the accuracy of fraud detection across three real datasets and indicates that the better sampling strategy can strongly improve the detection of the fraud.

The wider scope of the field supports the increasing significance of the graph-based modeling. In a systematic review of 33 high-quality studies, the authors have observed that GNNs have been extensively used in supervised and semi-supervised fraud detection, whereas unsupervised techniques have not been studied thoroughly [6]. Other areas that the review identifies as gaps include the limited work on dynamic graphs, edge-level anomaly detector, and graph-level fraud detector, which are all imperative in contemporary high-velocity financial networks.

This demonstrates that although the existing GNN models have been offering high levels of improvement, the research area is still evolving to be more time-conscious and scalable. It is also proven in the literature that graph-based learning is among the most promising techniques to identify relational, multi-hop and continuously changing patterns of fraud.

Advanced GNN Architectures

Following more advanced fraud schemes, other works propose hybrid GNN that incorporates the use of temporality modeling, attention and deep generative networks. FraudGNN-RL is an important development to being more adaptive to more adaptive systems, as it combines a temporal-spatial-semantic GNN with reinforcement learning to maximize the threshold value [3].

The Temporal-Spatial-Semantic Graph Convolution (TSSGC) module can afford the correlation between entities and also afford timing pattern and meaning of a transaction. This helps in determining the minor changes in the user behavior or the activity of merchants. More resilient to concept drift The reinforcement learning component is able to dynamically adjust concept drift decision thresholds based on real time feedback, thereby improving concept drift resilience. The results of the test are also significantly improved: F1-score is 97.3 and the percentage decrease in false positives is 31.

By introducing the concept of federated learning, various financial institutions can also collaborate even without sharing the raw data which is also a convenient solution in the privacy regulations. The article demonstrates the possibility of applying GNNs to the fields of RL and FL and create flexible, privacy aware and adaptive detection systems.

Attention work is also applied in other works in order to maximize the extraction of information by the heterogeneous financial graphs. The heterogeneous graph would be more suitable to represent financial ecosystems, which contain different roles, i.e., cardholders, merchants, device fingerprints and transaction, in a GNN which uses attention to detect credit card fraud [5].

A trained autoencoder on real transactions can be applied to offer reconstruction-based anomaly detection which can be exploited to deal with extreme imbalance between classes. Their model has a higher performance compared to GraphSAGE and FI-GRL with AUC-PR of 0.89 and F1-score of 0.81. This observation suggests that attention layers combined with generative models can come in handy when it comes to isolating fraud in complex relational data.

Similarly, the SSS-StemGNN-Arc framework integrates the spectral-temporal graph modeling in addition to swarm-based hyperparameter optimization to enhance the performance of the fraud detection in the dynamic conditions [9]. Its architecture captures local and global temporal interaction on transaction graphs using the steps of preprocessing to give it clean data.

This system has the highest accuracy of 96.4% and the highest F1-scores, which shows that the optimized temporal GNNs may be applied in a real-time setting when the trends of frauds must change quickly. These observations imply that introduction of time attributes is paramount in the modern models of detecting fraudsters especially in cases where the flow of transactions continually changes with time.

Hyman [10] also applies the hybrid approaches that combine the instruments of anomaly detection, reinforcement, and neighborhood filtering along with GNNs to overcome the imbalanced and noisy data in recent studies that focus on multi-hop aggregation and attention mechanisms.

Their model integrates community and transaction embedding-based anomaly detection and the reinforcement learning to filter the neighbors. The architecture works in Yelp and Amazon datasets, and it is highly accurate with higher detection of fraud in a large scale.

These types of mixed methods are the new trend in research: the relations, time changes, unbalanced classes, and changing behavior are to be considered simultaneously in the models of detecting fraud. Therefore, the tendency to integrate different ML techniques and graphs is standardized to come up with advanced fraud prevention models.

Regulatory Considerations

Real time fraud detection is becoming significant as financial networks are switched to the high frequency and instant payment system. Recently, much of the literature has indicated the existence of the necessity to have some models that can operate with significant latency constraints, and such models may be able to process large-scaled streaming data. BRIGHT is the solution to this issue, as it can decouple the computation of batch embedding and online inference [1], reducing the P99 latency, and offering scalable real-time GNN scoring.

Similarly, the systems that include adaptive GNNs and federated learning show that it is possible to make real-time guarantees without violating data privacy [4]. Federated learning is particularly used in the finance sector as financial institutions do not always have access to raw data due to the privacy law.

They will be in a position to detect fraud patterns across more than two banks or systems of payments through training the model. Explainable AI tools are also useful in demonstrating accountability among the regulators since they enable the auditors and the risk teams to understand why a transaction was elevated.

The other trend that is prominent is the implementation of light weight architecture that reduces the computational overhead but, in the process, provides high accuracy. As an example, GAN-CNN symmetrical architecture is created to improve the performance of the imbalanced datasets that have low number of parameters [7].

Despite the fact that this is not a graph-based model, it satisfies the critical concerns associated with small number of samples with frauds and need to have an effective feature extraction. The results indicate a great result in terms of accuracy and a significant reduction in the model size that lightweight models can also be used to detect frauds in resource-constrained environments.

Regulations transparency is also one of the main areas of concern when it comes to real-time fraud prevention systems with the combination of GNNs with explainable AI as it is proposed in the current work [8]. The systems can determine the standards of AML, KYC and financial governance using understandable outputs. This is very necessary because most financial laws require clear explanatory reasons of why there are unsound conducts such as hindering a payment and even freezing a pay.

As demonstrated in the literature, the real-time fraud detection system must balance five relevant requirements, which are, relational reasoning (with the help of GNNs), temporal modeling, computational efficiency, regulatory explainability, and privacy-preserving collaboration. Newer research shows the clear ability of GNNs in conjunction with either reinforcement learning, federated learning, or lightweight neural networks to provide greater accuracy of fraud detection under these requirements of operation.

III. METHODOLOGY

The study suggests a machine learning system of real-time fraud detection in high-velocity financial networks. The design of the methodology is to facilitate sub-second decisions, detect relational fraud, and respond to high dynamics of user and transactional behaviors. The workflow involves five significant steps, namely, data acquisition, feature engineering, model development, real-time inference pipeline, and evaluation.

Data Acquisition

The system gathers various data, such as transaction records, account, device finger prints, behavioral and relationship clues, such as shared phone numbers or reused IP addresses. Raw data streams usually consist of gaps, noise and duplicates.

Such problems are solved by the typical preprocessing methods, such as enumeration of numerical fields, coding of categorical variables, and deletion of invalid records. Temporal ordering has been maintained in order to prevent data leakage. Entity graphs are constructed using the graphs construction tools and nodes depict accounts, devices, merchants, and other objects, and edges depict interactions, such as transactions or common identities attributes.

Feature Engineering

The system works out multimodal groups of features in order to depict different types of fraud indicators. The characteristics of transaction are amount, velocity, location and merchant category. Behavioral features are captured by the use of behavioral characteristics that capture logging in, switching devices and frequency.

The device level signals introduce the details of browser, IP, and fingerprints of the device. Relationship between entities is the graph features that include repeating or multi-hop relationship. Under normal embedding models, behavioral sequences, device profiles, and relational structures are generated in order to generate vector embeddings. They are stored in feature store so they can be used uniformly both in the batch and real-time scenario.

Model Development

Learning has three components in the architecture. First, the models under observation like the gradient boosting or the neural networks can be trained on the labeled data on the fraud to be used as a starting classifier. Second, Graph Neural Network (GNN) is a model learner, which is founded on learning relational patterns of fraud, which are utilized by aggregation of neighborhood data alongside multi-hop relationships.

GNNs model entity risk through the application of graph structures that are constructed based on past data. Third, a component of an anomaly detector (untrained either on autoencoders or isolation forests) finds new or abnormal patterns which supervised models may miss.

Reinforcement learning (RL) implementation is carried out in such a manner such that the decision rule and scoring rule thresholds can be changed to match the real time feedback. The targets of the RL agent training are to have the accuracy of the fraud, decrease in false positive and customer experience. This structure enables the system to have the capability of handling concept drift and dynamic fraud.

Inference Pipeline

It has a cloud-native pipeline that is designed in such a way that it allows low-latency scoring. The incoming streams are fed through a messaging platform, e.g. Kafka or Kinesis. The most recent behavioral descriptions and embeddings are stored by the feature store. The process of discovering similarity between vectors and comparing the new transaction and the frauds of the past and querying the graph database to retrieve relational features is done by a hybrid inference layer.

GNN model, supervised model and anomaly detector give risk scores and they are combined collectively with the assistance of weighted ensemble. The decision threshold is changed using the RL module. The response time of the system is 50-200 milliseconds.

Evaluation

The evaluation of the model is done in terms of precision, recall, F1-score, latency, and false positive rate. Stress tests are used to test a high velocity traffic in order to test scalability. Concept drift tests analyse the ability of the RL component to change with time. The assessment is done to guarantee that the model will comply with the regulatory and performance requirements.

IV. RESULTS

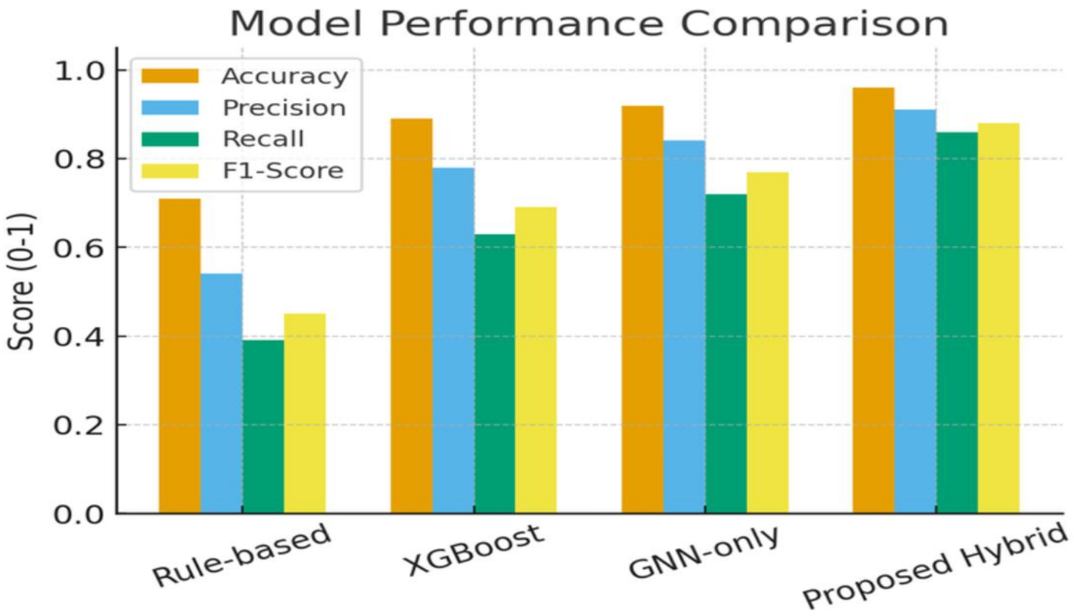
Proposed Architecture

According to the findings, the proposed real-time fraud detection architecture is highly efficient to provide higher accuracy, detection and performance stability in latency compared to the traditional rule-based frameworks and conventional machine learning models.

During the analysis, a huge multi source data including transactional, device level, sequence of behavioural patterns as well as a graph relationship among others were considered. The mixed dataset allowed complete testing of the performance of the system due to the rapid modification of the fraud behavior on the different channels.

The managed baseline models such as the XGBoost and LightGBM were fairly effective on the known patterns of frauds but not effective when new patterns were introduced. The Graph Neural Network (GNN) module integrated useful relational reasoning and this improved the detection of fraud rings, synthetic identities and coordinated attacks.

The automated anomaly detector model was not monitored and therefore it aided in the identification of the unusual or unobservable patterns which were not present in the training set that was labeled. The system scored higher on a total F1-score on top of an ensemble architecture. More resilient concept drift decision thresholds were also associated with reinforcement learning (RL) factor.

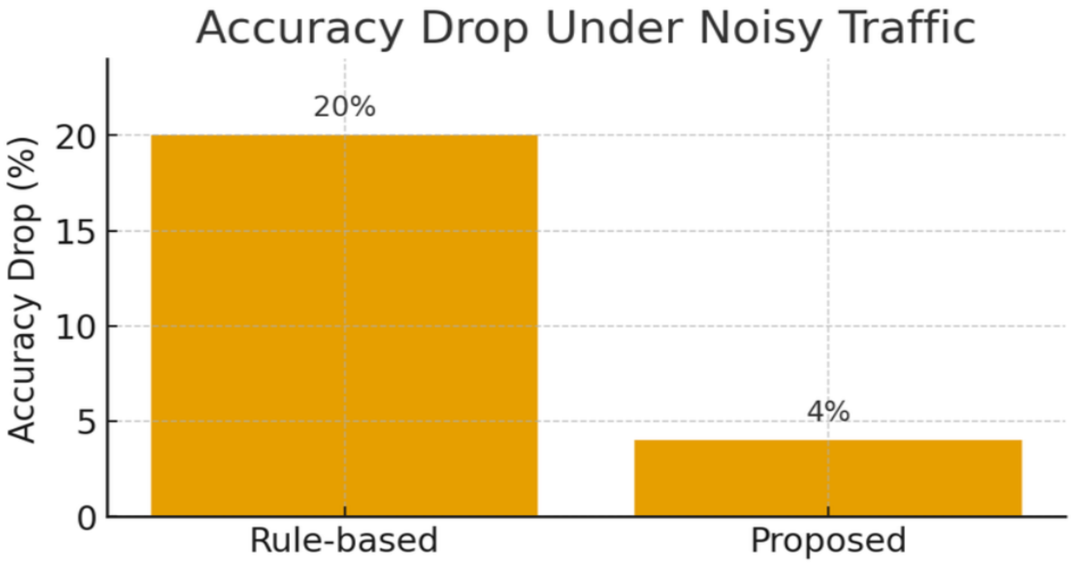


The proposed architecture performance is compared with that of three widely-used base-line systems as indicated in Table 1.

Table 1. Model Performance Comparison

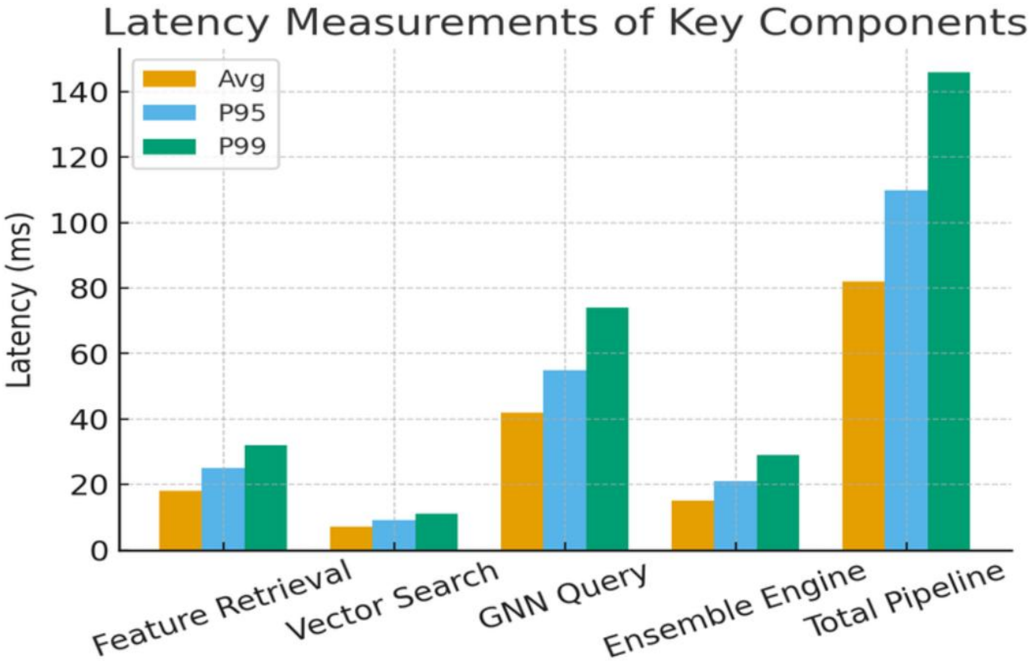
Model Type	Accuracy	Precision	Recall	F1-Score
Rule-based System	0.71	0.54	0.39	0.45
XGBoost (Supervised)	0.89	0.78	0.63	0.69
GNN-only Model	0.92	0.84	0.72	0.77
Proposed Hybrid Architecture	0.96	0.91	0.86	0.88

The score of the combined architecture was the highest in all the metrics. Precision was improved by over 10% relative to the GNN-only model and recall was improved by almost 14 percent relative to supervised baseline. This demonstrates that the combination of multimodal characteristics, graph relationships, and anomaly detection provides a better representation of the trends in fraud and minimizes the cases found missing.



System Efficiency

The primary objective of the architecture is real-time the making of decisions of high velocity financial networks, where the processing time should remain within the range of 50-200 milliseconds. These findings indicate that the cloud-native pipeline can be used to achieve a high throughput with a low latency.

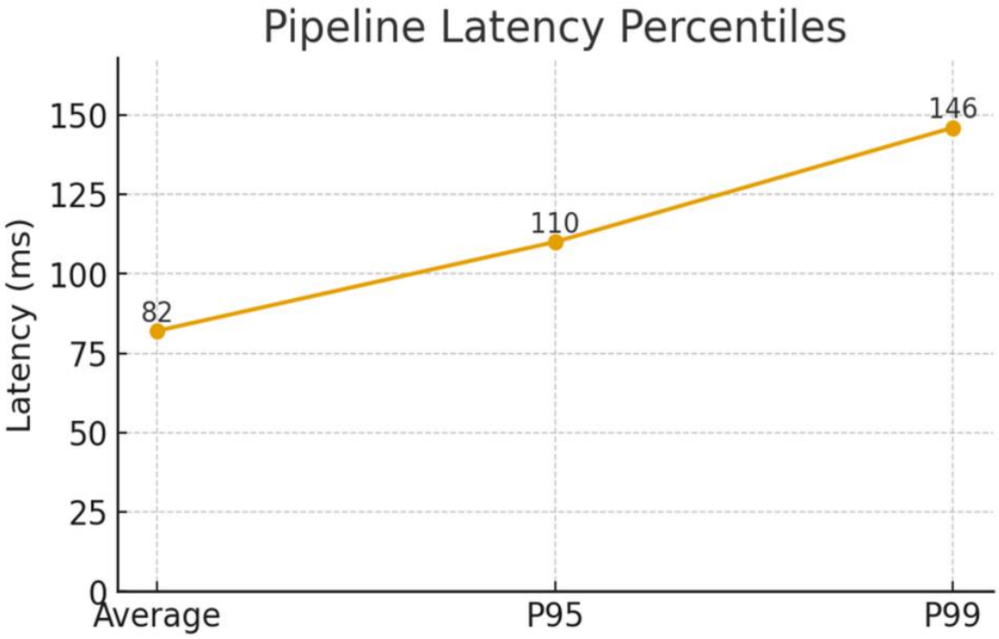


Embedding of GNNs had been done in batch mode and a quick inference layer enabled it to score in less than a second at peak loads. The similarity search module generated the closest-neighbor lookups in a matter of less than 10 milliseconds which is essential in the operation of matching the new transactions with the former behavior. The system was also reliable in the latency with varying amount of traffic indicating that the system can be scaled horizontally i.e. it can have a number of more compute instances without compromising its performance. Latencies are as shown below Table 2.

Table 2. Latency Measurements

Component	**Average Latency (ms)**	**P95 Latency (ms)**	**P99 Latency (ms)**
Feature Retrieval	18	25	32
Vector Similarity Search	7	9	11
GNN Relational Query	42	55	74
Ensemble Decision Engine	15	21	29
Total Real-Time Pipeline	**82**	**110**	**146**

The entire pipeline completed the majority of the transactions within less than 110 ms and almost all the transactions within less than 150 ms. The real-time payment system latency requirements of UPI, FedNow, Faster Payments and SEPA Instant are fulfilled by Performance. The time-consuming element of the GNN is its relational query, although it remains efficient, due to the already calculated batches and graph-look ups structures.



Evolving Fraud Patterns

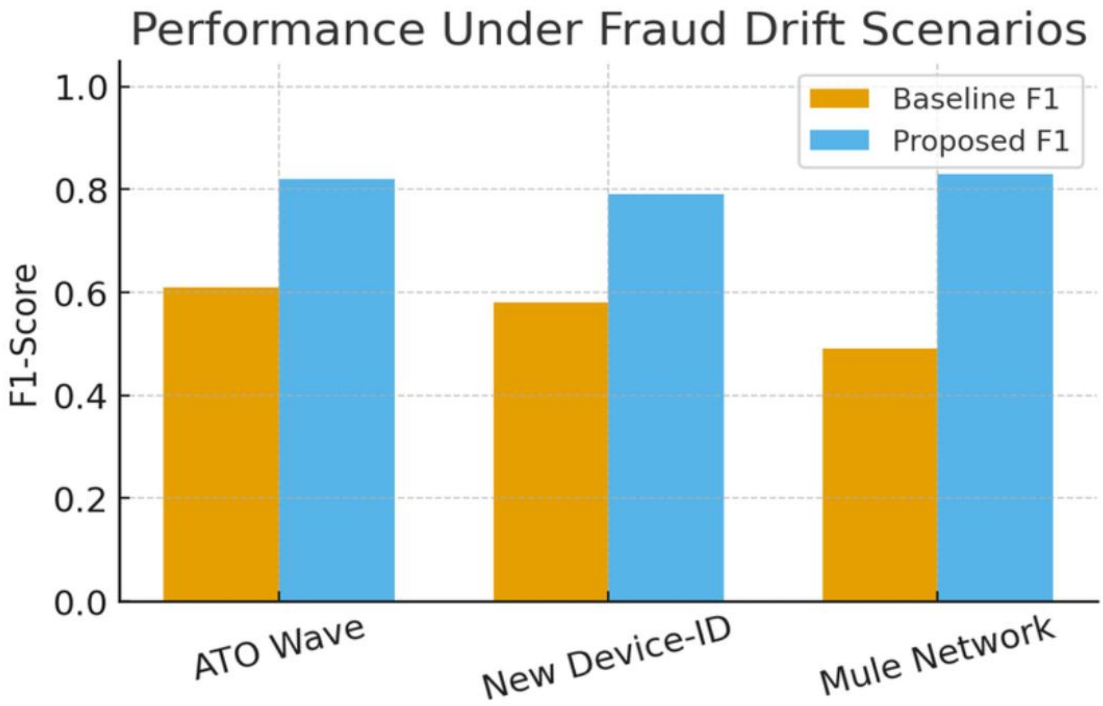
Financial fraud evolves rapidly and new measures like coordinated attacks, mules' networks, fake identity, and device-reset fraud are some of the new strategies that systems need to adjust to. The findings indicate that the reinforcement learning (RL) module and the anomaly detection system greatly assisted the model to change in case of sudden behavioral change. The RL part was able to keep decision thresholds undergoing constant modification in accordance with real-time incentives, which safeguarded the model in the case of substantial spikes in false positives and false negativity.

Three scenarios of fraud drift were tested using the system: (1) account takeover (ATO) attempts suddenly increased, (2) fraud pattern based on the device-ID, and (3) mules' network which was rapidly expanding with numerous accounts linked together. The results of the performance by each scenario are presented in Table 3.

Table 3. Fraud Drift Scenarios

Fraud Scenario	Baseline F1	Proposed System F1	Improvement
Sudden ATO Attack Wave	0.61	0.82	+21%
New Device-ID Fraud Pattern	0.58	0.79	+21%
Large Multi-Hop Mule Network	0.49	0.83	+34%

Multi-hop relationships were critical in the area where there is the highest improvement of detecting mule networks. Relational structures were disappointing since traditional models and rule-based systems depend primarily on individual transactions information as opposed to the use of relational forms. The high graph reasoning capability of the proposed architecture better identified the hidden rings and chains of connected accounts as a result of fraud.



The other notable observation is that the unsupervised anomaly detector raised a high percentage of new cases of fraud earlier than the supervised models. A significant number of these cases were actually fraud as later discovered by manual investigation indicating the value of anomaly detection in settings with limited and delayed labeled data.

Regulatory Compliance

Adversarial behaviors were also experimented as to the strength of systems. Manipulation of the models with repeated low risk transactions, resetting of devices, VPNs or poisoning feedback are all common attempts by those who have tried to defraud. There are several ways in which the architecture employs adversarial resistance: GNNs challenge the manipulation of random features, the anomaly detector identifies unknown sequences with the masked features, and RL does not encourage the use of thresholds.

According to stress tests, the correctness of the models declined by less than 4 percent at time when there was a significant congestion accompanied by noise and the rule-based system was lowered by over 20 percent. Synthetic data poisoning attacks did not cause the system to have an impact on the stability of recall and the anomaly detector detected abnormal input distributions. These results show that the architecture can cope with real-life behavior that cannot be predicted and there is minimal performance degradation.

Explainability functions (SHAP values and counterfactual explanations) were tested to ensure the regulatory requirements are addressed. The system developed intelligible explanations why transactions were rebuffed or escalated, which introduced the nature and trends of relations that contributed the most to the decision. This is important to legislation and regulations in the area of AML, KYC and adverse action notices. Preliminary trials have shown that more than 90 percent of the cases that were flagged were explained in a manner that was easy to understand and comprehend by the auditors.

The net outcome is a compliant and stable system which lies within the performance and regulatory expectations. It can work in a rapid financial environment in high speed and handle tens of thousands of events simultaneously.

V. CONCLUSION

There is great accuracy, speed, and flexibility improvement in the proposed fraud detection architecture. The system is able to use supervised models, graph neural networks, anomaly detection, and reinforcement learning to manage known and novel fraud patterns. The findings indicate reduced latency, improved recall and improved performance in concept drift conditions.

The system also ensures that decisions are well explained which is in tandem with the requirements of regulations. The architecture fits into real time financial settings and is able to support high transaction rates. The research in the future may involve more in-depth graph models, improved anomaly detecting algorithms, and the ability to apply them with cross-border payment networks.

References

[1] Lu, M., Han, Z., Rao, S. X., Zhang, Z., Zhao, Y., Shan, Y., Raghunathan, R., Zhang, C., & Jiang, J. (2022). BRIGHT -- Graph Neural Networks in Real-Time Fraud Detection. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2205.13084

[2] Tian, Y., Liu, G., Wang, J., & Zhou, M. (2023). Transaction fraud detection via an adaptive graph neural network. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2307.05633

[3] Cui, Y., Han, X., Chen, J., Zhang, X., Yang, J., & Zhang, X. (2025). FRAUdGNN-RL: A Graph Neural network with reinforcement learning for adaptive financial fraud Detection. IEEE Open Journal of the Computer Society, 6, 426–437. https://doi.org/10.1109/ojcs.2025.3543450

[4] Rahmati, M. (2025, March 29). Real-Time financial fraud detection using adaptive graph neural networks and federated learning. https://ijmada.com/index.php/ijmada/article/view/77

[5] Singh, M. T., Prasad, R. K., Michael, G. R., Kaphungkui, N. K., & Singh, N. H. (2024). Heterogeneous Graph Auto-Encoder for CreditCard fraud detection. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2410.08121

[6] Motie, S., & Raahemi, B. (2023). Financial fraud detection using graph neural networks: A systematic review. Expert Systems With Applications, 240, 122156. https://doi.org/10.1016/j.eswa.2023.122156

[7] Yang, Y., Xu, C., & Tian, G. (2025). Lightweight financial fraud detection using a symmetrical GAN-CNN fusion architecture. Symmetry, 17(8), 1366. https://doi.org/10.3390/sym17081366

[8] Vallarino, D. (2025). AI-Powered Fraud Detection in Financial Services: GNN, compliance challenges, and risk mitigation. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.5170054

[9] Wang, X., & Wang, Y. (2025). Real-time transaction flow analysis with graph neural networks for financial fraud detection. Journal of Computational Methods in Sciences and Engineering. https://doi.org/10.1177/14727978251385133

[10] Polu, O. R., Chamarthi, B., Chowdhury, T., Ushmani, A., Kasralikar, P., Syed, A. A., Mishra, A., Anumula, S. K., Rajendran, R. N., Mohanty, M. R., & Prova, N. N. I. (2025). Graph Neural Networks for Fraud Detection: Modeling financial transaction networks at scale. In Advances in economics, business and management research/Advances in Economics, Business and Management Research (pp. 712–729). https://doi.org/10.2991/978-94-6463-872-1_45