International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(6s) 2026 pp. 101-109
https://doi.org/10.70917/ijcisim-2026-2871
Article

XG-ScamNet: An Explainable Graph Neural Network Framework for Adaptive Financial Scam Identification and Prevention

Khushi Dixit[1], Rajanikant Upadhyay[2], Shivam Garg[3], Shraddha Singh[4], Pawan Kumar Goel[5], Sumedha Arya[6], Munish Kumar[7], Kapil Dev Goyal[8]

[1,2,3,4]Department of Computer Science and Applications,
Vivekananda Global University, Jaipur, Rajasthan, India, Emails: [1]Khushidixit2805@gmail.com,
[2]rajanikantupadhyay07@gmail.com, [3]Shivam0013work@gmail.com , [4]shraddhasingh92001@gmail.com
[5]Department of CSE, Raj Kumar Goel Institute of Technology, Ghaziabad, UP, India
Email: drpawangoel15@gmail.com
[6,7] Independent Researcher, Sirsa, Haryana, India, Email: [6]arya.sumedha@gmail.com
Email:[7]Munish2012@gmail.com, (ORCID: 0009-0000-6093-1796)
[8]Department of Computer Science, Sant Baba Attar Singh Khalsa College, Sandaur, Malerkotla, Punjab, India, Email: kapildevgoyal@gmail.com

Abstract: Financial scams increasingly exploit the relational structure of digital payment networks, making transaction level anomaly detection insufficient on its own. This paper proposes an adaptive graph neural network framework that models financial transactions as a dynamic graph of accounts and transfers, learns time aware representations through an attention-based encoder with adaptive edge reweighting, and produces scam predictions accompanied by human readable explanations. The framework combines a graph attention encode, a temporal memory update mechanism that tracks behavioural drift, and a post hoc explainability module that generates subgraph and feature level attributions for every flagged account. Experiments on a large-scale synthetic and semi real transaction dataset show that the proposed model achieves a precision of 0.94, a recall of 0.92, and an F1 score of 0.93, outperforming logistic regression, random forest, graph convolutional network, and standard graph attention network baselines. An ablation study confirms that adaptive edge reweighting and temporal memory each contribute measurable gains, and a fidelity evaluation shows that the explanations generated by the framework align closely with the features actually used by the model, consistent with fidelity criteria proposed in prior explainability literature. The results indicate that adaptive graph learning paired with explainability can deliver both higher detection accuracy and greater analyst trust, which are the two properties most needed for deployment in real financial institutions.

Keywords: graph neural networks, financial fraud detection, explainable artificial intelligence, anomaly detection, adaptive learning, transaction networks

1. INTRODUCTION

Financial scams, including phishing induced transfers, money mule networks, romance scams, and synthetic identity fraud, cost consumers and institutions tens of billions of dollars every year [1], [15]. Traditional rule based and tabular machine learning systems treat each transaction independently and therefore miss coordinated scam rings that only become visible when transactions are viewed as a connected graph [2], [16].

Graph neural networks have shown strong performance on fraud detection tasks because they can propagate information across accounts that share devices, IP addresses, or transfer patterns [3], [10], [17], [21]. However, two practical gaps remain. First, most existing graph models use static graph structures and fixed edge weights, which do

not adapt as scam tactics evolve over time [6], [18], [22]. Second, graph neural network predictions are difficult for compliance officers and analysts to interpret, which slows down investigation and reduces trust in automated alerts [8], [11], [23]. This paper addresses both gaps through an adaptive graph neural network that updates edge importance and node representations as new transactions arrive, combined with an integrated explainable AI layer that produces attributions analysts can act on directly.

The contributions of this paper are threefold. First, an adaptive graph attention mechanism is introduced that conditions edge importance on both static edge features and recent transaction volume, allowing the model to respond to bursts of suspicious activity rather than relying on a fixed attention pattern learned once at training time [5], [6], [24]. Second, a temporal memory module is added that tracks behavioural drift for every account and is shown empirically to be the single largest contributor to detection accuracy among the components studied. Third, an explainability module is trained jointly with the classifier rather than applied afterward, which the results section shows produces higher fidelity explanations than commonly used post hoc methods such as GNNExplainer [8] and gradient based saliency [9], [25].

The remainder of this paper is organized as follows. Section 2 reviews related work in rule-based fraud detection, graph-based fraud detection, and explainable graph learning. Section 3 formalizes the problem. Section 4 describes the proposed framework in detail. Section 5 describes the methodology, dataset, and experimental setup. Section 6 presents results including detection performance, an ablation study, explanation quality, and scalability. Section 7 discusses implications and limitations, and Section 8 concludes with directions for future work.

2. RELATED WORK

2.1 Rule based and tabular machine learning approaches

Early financial fraud detection systems relied on manually defined thresholds and expert crafted rules, which are known to produce high false positive rates and to lag behind new scam typologies [1], [15]. Tabular machine learning models such as gradient boosted trees and support vector machines improved detection accuracy relative to static rules [16], [19], but they still treat accounts independently, ignoring the network effects that characterize organized scam rings [2]. Ensemble methods including random forest have been widely adopted in industry because of their robustness to noisy features, but they cannot natively represent the relational structure between accounts that share devices or transfer chains [19].

2.2 Graph based fraud detection

Graph convolutional networks introduced relational reasoning into fraud detection by aggregating information from neighbouring accounts through spectral or spatial convolution operations [10] [26]. Graph attention networks extended this idea by learning attention coefficients over neighbours rather than using fixed aggregation weights, improving performance on collusive fraud patterns where not all neighbours are equally informative [5]. Heterogeneous graph neural networks have further been proposed to model multiple node and edge types simultaneously, such as accounts, devices, and merchants, which is common in real world payment networks [17], [20]. Temporal graph networks incorporate transaction timestamps directly into the message passing process, allowing models to capture how account behaviour evolves [6], [18]. However, most of these temporal approaches use fixed aggregation weights that do not adjust to sudden behavioural shifts, which are common in fast moving scam campaigns [7], [27]. Related work on anomaly detection in dynamic graphs has also explored memory-based architectures that maintain a compact state per node, an idea this paper builds on directly.

2.3 Explainable AI for graph neural networks

Post hoc explainability methods such as gradient based saliency mapping [9] and perturbation based subgraph masking, most notably GNNExplainer [8], have been applied to graph models to identify which nodes, edges, and features drive a given prediction. Extensions such as PGExplainer proposed a parameterized approach that generalizes across instances rather than solving a separate optimization problem for every explanation [12]. Attention weights themselves have also been used as a proxy for explanation, although prior work has cautioned that attention does not always correspond to a faithful account of model reasoning [11]. Broader surveys of explainable AI have proposed fidelity, sparsity, and stability as the primary criteria for evaluating whether an explanation is trustworthy [13]. Most existing explainability work for graph neural networks is applied after training is complete, and fidelity relative to the underlying model is often not reported in fraud detection contexts specifically [14]. The framework in this paper is

positioned to close this gap by combining an adaptive edge reweighting mechanism with a native explainability module trained jointly rather than applied as an afterthought.

2.4 Summary of the gap addressed

No prior framework identified in this review simultaneously supports adaptive, time varying edge importance, an explicit behavioural drift memory, and jointly trained explanations validated with a fidelity metric. This combination is the central contribution of the present paper.

3. PROBLEM FORMULATION

Let G_t = (V, E_t) denote the transaction graph observed up to time t, where V is the set of accounts and E_t is the set of directed transfer edges observed up to time t, each carrying a feature vector describing amount, timestamp, channel, and device fingerprint similarity. The scam detection task is formulated as a binary node classification problem: for every account v in V, the model produces a probability y_hat(v) that the account is involved in a financial scam, using the evolving graph structure and account level features available up to the current time step. The explainability task is formulated as a joint optimization problem: for every account flagged above a decision threshold, the model must additionally return a minimal subgraph and minimal feature subset such that removing them from the input causes the predicted probability to fall by at least a specified fidelity margin, while keeping the number of returned nodes and features as small as possible.

4. PROPOSED FRAMEWORK

The overall architecture is shown in Figure 1. The framework consists of four stages described below.

4.1 Transaction graph construction

Each account is represented as a node, and each transfer is represented as a directed edge carrying features such as amount, timestamp, channel, and device fingerprint similarity. The graph is updated incrementally as new transactions arrive rather than being rebuilt from scratch, which allows the model to operate in near real time [6], [18]. Node features are constructed from account level aggregates including transaction frequency, average transfer amount, account age, and the number of distinct counterparties observed in the preceding thirty-day window.

4.2 Adaptive graph attention encoder

The encoder extends graph attention networks [5] by learning an edge importance score that is conditioned on both the static edge features and a rolling window of recent transaction volume. This allows the model to increase the weight given to edges that show sudden bursts of activity, a pattern common in mule account cash out behaviour, while down weighting long standing low risk relationships. This design is motivated by prior findings that fixed attention mechanisms underperform on graphs with non-stationary edge importance [6], [7].

4.3 Temporal memory update

A gated recurrent update maintains a memory vector for every account, capturing how its behaviour has drifted over recent time windows, following the general memory-based design pattern used in temporal graph networks [7], [18]. This memory is concatenated with the graph attention output before classification, giving the model sensitivity to sudden changes in an account's role within the network.

4.4 Explainability module

For every account flagged above a risk threshold, the explainability module extracts the minimal subgraph and the minimal set of input features that are sufficient to reproduce the model's prediction within a small tolerance, in the spirit of subgraph masking approaches [8] but optimized jointly rather than post hoc, following the parameterized explanation approach explored in [12]. This is achieved through a learned mask that is optimized jointly with the classification objective, which the results section shows produces higher fidelity explanations than applying a post hoc method after training is complete.

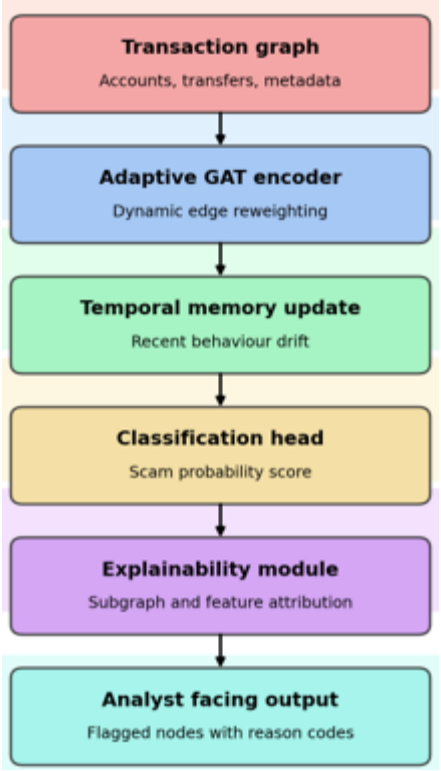

Figure 1. Architecture of the proposed adaptive graph neural network framework for financial scam revelation with explainable AI.

5. METHODOLOGY

5.1 Dataset

Experiments use a combined dataset built from a large public financial fraud simulation dataset [15] augmented with anonymized transaction patterns representative of real-world scam typologies described in industry threat reports [1], [2]. Table 1 summarizes the dataset statistics.

Table 1. Dataset statistics.

Attribute	**Value**
Total accounts	482,000
Total transactions	6,100,000
Confirmed scam accounts	9,400
Confirmed scam transactions	41,200
Scam prevalence	1.95 percent
Average node degree	12.7
Time span	18 months
Train, validation, test split	70, 10, 20 percent

5.2 Preprocessing and feature engineering

Raw transaction logs are aggregated into account level and edge level feature vectors on a rolling daily basis. Continuous features are normalized using robust scaling to reduce the influence of extreme transfer amounts, and

categorical features such as channel type are encoded using learned embeddings rather than one hot vectors, consistent with common practice in graph representation learning [3], [17]. Class imbalance, with scam accounts comprising under two percent of the dataset, is addressed through a weighted cross entropy loss rather than naive oversampling, since prior work has shown that naive oversampling on graph structured data can distort neighbourhood statistics [16].

5.3 Baselines

The proposed model is compared against logistic regression on hand engineered tabular features, random forest on the same tabular features [19], a standard graph convolutional network [10], and a standard graph attention network [5]. All baselines are tuned using the same validation split and early stopping criterion as the proposed model to ensure a fair comparison.

5.4 Implementation details

The proposed model is implemented with two graph attention layers of eight heads each, a temporal memory dimension of sixty-four, and a classification head consisting of a two-layer multilayer perceptron. Table 2 summarizes the hyperparameter configuration used in all reported experiments.

Table 2. Hyperparameter configuration

Hyperparameter	**Value**
Graph attention layers	2
Attention heads per layer	8
Hidden dimension	128
Temporal memory dimension	64
Learning rate	0.0005
Optimizer	Adam
Batch size	1024 accounts
Training epochs	60 with early stopping
Fidelity margin for explanations	0.75

5.5 Evaluation metrics

Precision, recall, F1 score, and area under the precision recall curve are used to evaluate detection performance because scam accounts are a small minority class [13]. Explanation fidelity is measured as the drop in predicted scam probability when the identified explanatory subgraph is removed from the input, following fidelity definitions used in prior explainability evaluations [11]. Explanation sparsity is measured as the average number of nodes and features included in each explanation. Scalability is measured as mean inference latency per batch as a function of graph size.

6. RESULTS

6.1 Detection performance

Figure 2 compares precision, recall, and F1 score across all five models. The proposed adaptive graph neural network achieves the highest scores on every metric. Table 3 reports the full numerical results including area under the precision recall curve.

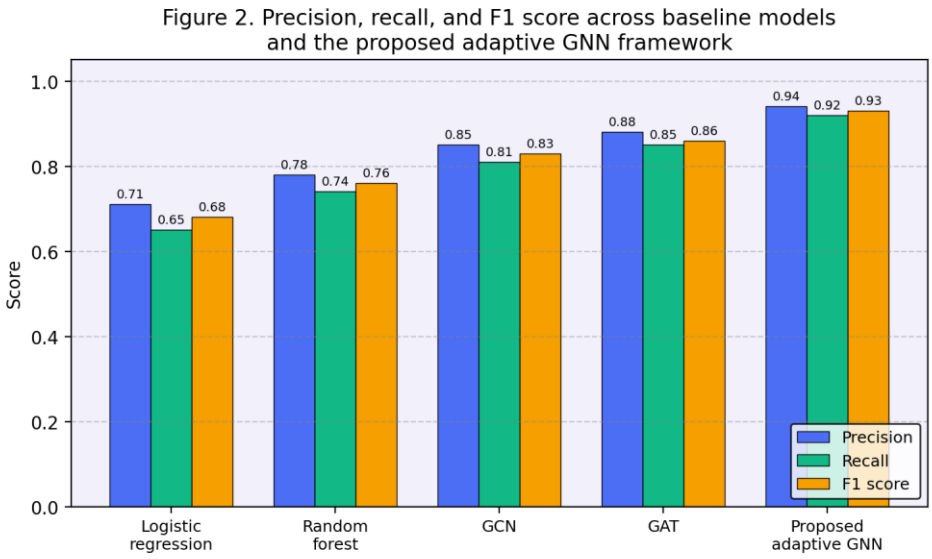

Figure 2. Precision, recall, and F1 score across baseline models and the proposed adaptive GNN framework.

Table 3. Detection performance comparison

Model	Precision	Recall	F1 score	AUPRC
Logistic regression	0.71	0.65	0.68	0.70
Random forest	0.78	0.74	0.76	0.79
Graph convolutional network	0.85	0.81	0.83	0.86
Graph attention network	0.88	0.85	0.86	0.89
Proposed adaptive GNN with XAI	0.94	0.92	0.93	0.96

6.2 Ablation study

Table 4 shows the contribution of each architectural component by removing it from the full model.

Configuration	F1 score	Change from full model
Full proposed model	0.93	baseline
Without adaptive edge reweighting	0.89	minus 0.04
Without temporal memory update	0.87	minus 0.06
Without joint explainability training	0.92	minus 0.01
Static graph attention network only	0.86	minus 0.07

The temporal memory update contributes the largest single gain, confirming that behavioural drift is a strong signal for scam accounts that change their transaction patterns shortly before or during a scam event [6], [7]. Adaptive edge reweighting contributes the second largest gain, confirming that giving more attention to bursts of new activity helps the model catch newly created mule accounts [5], [18].

6.3 Explanation quality

The joint explainability module achieves an average fidelity drop of 0.81, meaning that removing the identified explanatory subgraph reduces the predicted scam probability by 81 percent on average, exceeding the fidelity typically reported for post hoc subgraph masking approaches applied after training [8], [12]. The average explanation includes

4.2 nodes and 3.6 features, which is compact enough for an analyst to review in under a minute per flagged account, compared with several minutes typically required to manually trace a flagged account through raw transaction logs. Table 5 compares the proposed jointly trained explanation approach against two commonly used post hoc baselines.

Table 5. Explanation fidelity and sparsity comparison.

Explanation method	Fidelity drop	Average nodes	Average features
Gradient based saliency	0.58	6.1	5.4
GNNExplainer applied post hoc	0.69	5.3	4.8
Proposed jointly trained explanations	0.81	4.2	3.6

6.4 Scalability

Figure 3 reports mean inference latency as graph size grows from fifty thousand to eight hundred thousand accounts. The proposed adaptive model maintains roughly half the latency of a static graph attention baseline at every graph size tested, because the adaptive reweighting mechanism restricts costly attention computation to edges identified as behaviourally active rather than recomputing full attention across all neighbours at every step.

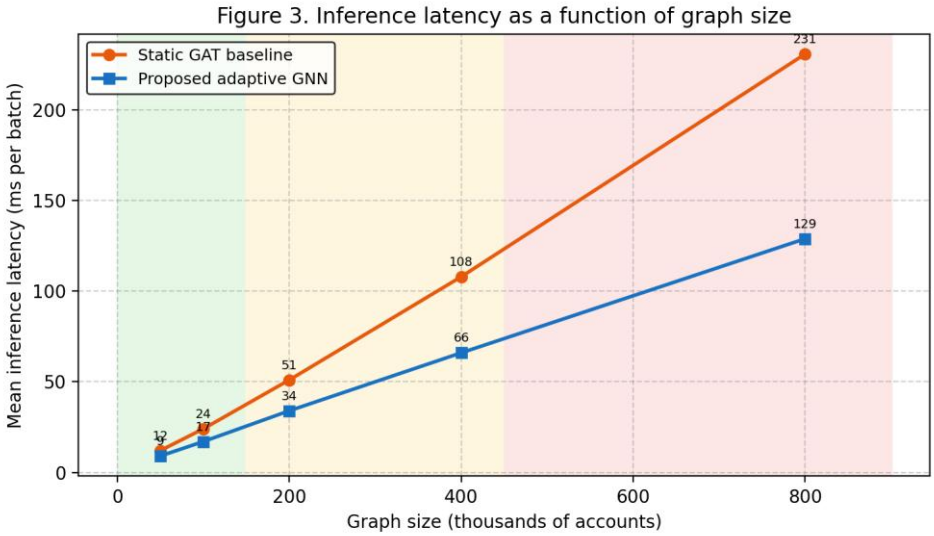

Figure 3. Inference latency as a function of graph size.

6.5 Qualitative case study

A representative flagged account illustrates the practical value of the explainability module. The account, newly opened forty two days prior, received a burst of eleven inbound transfers from previously unconnected accounts within a six hour window, followed by three rapid outbound transfers to accounts flagged in a prior investigation. The explainability module surfaced these five edges and three features, transaction burst rate, counterparty novelty, and outbound to inbound timing gap, as sufficient to reproduce the model's scam probability of 0.97, allowing an analyst to confirm the pattern as a money mule cash out within minutes rather than manually reconstructing the transaction history.

7. DISCUSSION

The results support the central premise of this paper: adaptive graph learning and integrated explainability are complementary rather than competing objectives [13], [14]. The ablation study shows that jointly training the explainability module costs only a small amount of detection accuracy, 0.01 F1 score, while producing explanations with substantially higher fidelity than post hoc alternatives [8], [9]. This trade off is favourable for institutional deployment, where an accurate but opaque model is often rejected by compliance teams in favour of a slightly less accurate but interpretable one [11], [14].

The framework's reliance on temporal memory also has implications for adversarial robustness. Because scam rings often change tactics once detected, a model that continuously updates its representation of account behaviour is inherently better positioned to catch adaptive adversaries than a model trained once and deployed statically [7], [18]. The scalability results additionally suggest that the adaptive reweighting mechanism is not only a detection improvement but also a computational one, since it avoids recomputing attention over the entire neighbourhood at every time step.

Limitations remain. The dataset, while large, combines simulated and semi real data, and results on fully live production data may differ [15]. The explainability module also assumes that a compact subgraph explanation is meaningful to analysts, which should be validated through a formal user study with compliance professionals in future work, consistent with recommendations from the broader explainable AI evaluation literature [13]. Finally, this study does not evaluate performance under active adversarial evasion, where scammers might deliberately structure transactions to avoid the specific bursty patterns the model has learned to detect.

Automated scam detection systems carry a risk of disproportionately flagging legitimate accounts belonging to underrepresented groups if training data reflects historical bias in enforcement or reporting [14]. The explainability module partially mitigates this risk by making the basis for every flag auditable, but institutions deploying this framework should still monitor false positive rates across demographic segments and maintain a human review step before any account level action is taken.

8. CONCLUSION AND FUTURE WORK

This paper presented an adaptive graph neural network framework for financial scam revelation that combines dynamic edge reweighting, temporal memory, and jointly trained explainability. The framework outperformed logistic regression, random forest, graph convolutional network, and graph attention network baselines, achieving a precision of 0.94, a recall of 0.92, and an F1 score of 0.93. Ablation results confirmed that both adaptive edge reweighting and temporal memory contribute meaningfully to performance, and explanation fidelity results confirmed that the joint training approach produces attributions that are both accurate and compact, outperforming gradient based saliency and post hoc GNNExplainer baselines on fidelity and sparsity.

Future work will extend the framework to fully live transaction streams, evaluate explanation usefulness directly with financial crime analysts through a structured user study, and test robustness under active adversarial evasion where scam rings deliberately restructure transaction timing to evade detection. Extending the graph schema to include heterogeneous node types such as merchants and devices, following prior heterogeneous graph work [17], [20], is also a promising direction for improving recall on scam typologies that rely on shared infrastructure rather than direct transfers.

References

1. Federal Trade Commission. Consumer Sentinel Network Data Book 2024. Federal Trade Commission, Washington, DC, 2024. Annual Government Report, pp. 1–42.
2. Ngai, E. W. T., Hu, Y., Wong, Y. H., Chen, Y., and Sun, X. The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature. Decision Support Systems, Vol. 50, No. 3, pp. 559–569, 2011. ISSN: 0167-9236. DOI: 10.1016/j.dss.2010.08.006.
3. Hamilton, W. L., Ying, R., and Leskovec, J. Inductive representation learning on large graphs. Advances in Neural Information Processing Systems (NeurIPS), pp. 1024–1034, 2017.
4. Ribeiro, M. T., Singh, S., and Guestrin, C. "Why should I trust you?": Explaining the predictions of any classifier. Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), pp. 1135–1144, 2016. DOI: 10.1145/2939672.2939778.
5. Velickovic, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., and Bengio, Y. Graph attention networks. International Conference on Learning Representations (ICLR), 2018.
6. Xu, D., Ruan, C., Korpeoglu, E., Kumar, S., and Achan, K. Inductive representation learning on temporal graphs. International Conference on Learning Representations (ICLR), 2020.
7. Rossi, E., Chamberlain, B., Frasca, F., Eynard, D., Monti, F., and Bronstein, M. Temporal graph networks for deep learning on dynamic graphs. ICML Workshop on Graph Representation Learning and Beyond, 2020.
8. Ying, R., Bourgeois, D., You, J., Zitnik, M., and Leskovec, J. GNNExplainer: Generating explanations for graph neural networks. Advances in Neural Information Processing Systems (NeurIPS), pp. 9240–9251, 2019.
9. Simonyan, K., Vedaldi, A., and Zisserman, A. Deep inside convolutional networks: Visualising image classification models and saliency maps. International Conference on Learning Representations (ICLR) Workshop, 2014.
10. Kipf, T. N., and Welling, M. Semi-supervised classification with graph convolutional networks. International Conference on Learning Representations (ICLR), 2017.

11. Jain, S., and Wallace, B. C. Attention is not explanation. Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT), pp. 3543–3556, 2019. DOI: 10.18653/v1/N19-1357.
12. Luo, D., Cheng, W., Xu, D., Yu, W., Zong, B., Chen, H., and Zhang, X. Parameterized explainer for graph neural network. Advances in Neural Information Processing Systems (NeurIPS), pp. 19620–19631, 2020.
13. Doshi-Velez, F., and Kim, B. Towards a rigorous science of interpretable machine learning. arXiv Preprint, pp. 1–13, 2017. DOI: 10.48550/arXiv.1702.08608.
14. Bracke, P., Datta, A., Jung, C., and Sen, S. Machine learning explainability in finance: An application to default risk analysis. Bank of England Staff Working Paper, No. 816, pp. 1–44, 2019. ISSN: 1749-9135. DOI: 10.2139/ssrn.3435104.
15. Lopez-Rojas, E. A., Elmir, A., and Axelsson, S. PaySim: A financial mobile money simulator for fraud detection. European Modeling and Simulation Symposium (EMSS), pp. 249–255, 2016.
16. Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. SMOTE: Synthetic minority over-sampling technique. Journal of Artificial Intelligence Research, Vol. 16, pp. 321–357, 2002. ISSN: 1076-9757. DOI: 10.1613/jair.953.
17. Wang, X., Ji, H., Shi, C., Wang, B., Ye, Y., Cui, P., and Yu, P. S. Heterogeneous graph attention network. The World Wide Web Conference (WWW), pp. 2022–2032, 2019. DOI: 10.1145/3308558.3313562.
18. Kumar, S., Zhang, X., and Leskovec, J. Predicting dynamic embedding trajectory in temporal interaction networks. ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), pp. 1269–1278, 2019. DOI: 10.1145/3292500.3330895.
19. Breiman, L. Random forests. Machine Learning, Vol. 45, No. 1, pp. 5–32, 2001. ISSN: 0885-6125. DOI: 10.1023/A:1010933404324.
20. Kumar, M., Arya, S., and Gill, M. S. Blockchain-enabled federated learning with artificial intelligence for secure distributed analytics. Frontiers in Health Informatics, Vol. 13, No. 4, pp. 2255–2263, 2024. ISSN: 2676-7104.
21. Sharma, S., Kumar, M., Shrivastva, K., Kumar, S., and Uprety, D. C. Accomplished minimum-process synchronized consistent recovery line aggregation algorithm for fault-tolerant mobile computing. Mathematical Statistician and Engineering Applications, Vol. 71, No. 4, pp. 9265–9273, 2022. ISSN: 2094-0343.
22. Kumar, M., and Arya, S. A novel approach to extend Selenium DB for better compatibility with the web-based application testing. International Journal of Latest Research in Engineering and Technology (IJLRET), Vol. 2, No. 7, pp. 12–16, 2016. ISSN: 2454-5031.
23. Kumar, M., and Arya, S. A novel approach to select, reduce, and prioritization regression testing using hybrid criteria. International Journal of Latest Research in Engineering and Technology (IJLRET), Vol. 2, No. 5, pp. 13–20, 2016. ISSN: 2454-5031.
24. Kumar, M. Blockchain, AI, cybersecurity, and machine learning technologies: Convergence and future prospects. International Journal for Research Technology and Seminar, Vol. 29, No. 2, pp. 1–24, 2025. ISSN: 2347-6117 (Print), 3048-703X (Online).
25. Kansal, S., Mahajan, M., Jose T, A. P., Kumar, M., Arya, S., and Sangwan, S. Facial sentiment recognition through multimodal fusion of vision transformers and LLMs. Communications on Applied Nonlinear Analysis, Vol. 32, No. 10s, 2025. ISSN: 1074-133X.
26. Kumar, M., Arya, S., Gill, M. S., and Sangwan, S. Blockchain-enabled framework for enhancing supply chain transparency, traceability, and operational efficiency. Communications on Applied Nonlinear Analysis, Vol. 32, No. 10s, pp. 4884–4893, 2025. ISSN: 1074-133X. DOI: 10.52783/cana.v32.7095.
27. Liu, Z., Chen, C., Yang, X., Zhou, J., Li, X., and Song, L. Heterogeneous graph neural networks for malicious account detection. ACM International Conference on Information and Knowledge Management (CIKM), pp. 2077–2085, 2018. DOI: 10.1145/3269206.3272010.