FUDMA Journal of Sciences (FJS)
ISSN online: 2616-1370
ISSN print: 2645 - 2944
Vol. 10 No. 18, September, 2026, pp 250 – 259
DOI: https://doi.org/10.33003/fjs-2026-1018-5889

Uncertainty-Aware Credit Card Fraud Detection Using a Fuzzy Graph Attention Network with Weak-Signal Edge Preservation

Mustapha Ismail, Saadatu Zakariyyah and *Ahmed Mohammed

Department of Computer Science, Gombe State University, Gombe, Gombe State, Nigeria.

*Corresponding authors' email: ahmedmhd686@gsu.edu.ng
ORCID: https://orcid.org/0009-0009-1394-1719

Received: 05 Sep. 2026
Accepted: 14 Sep. 2026
Published: 26 Sep. 2026

ABSTRACT
Coordinated card fraud can leave only weak, partial traces between transactions, which crisp graph detectors discard. This study evaluates a Fuzzy Graph Attention Network (Fuzzy-GAT) in which Gaussian edge-membership weights modulate attention, so that weak-signal edges are preserved rather than pruned. Because the public European Credit Card Fraud benchmark contains no account, device or IP identifiers, the graph links each transaction to its ten nearest neighbours in feature space, and no relational entities were synthesised. After removing 1,081 duplicate records (283,726 transactions; 473 frauds), every model was evaluated over five stratified 70/10/20 splits, at both the default and a validation-optimised decision threshold. The Fuzzy-GAT reached an F1-score of 79.3 ± 2.9% and a ROC-AUC of 0.969 ± 0.012. It matched Logistic Regression (79.4 ± 1.8%) and a crisp Graph Convolutional Network (79.9 ± 3.0%), but Random Forest (82.1 ± 3.3%; paired t-test $p = 0.014$) and XGBoost (82.8 ± 2.4%; $p = 0.002$) outperformed it. In a three-seed ablation, fuzzy weighting gave no benefit, and preserving weak-signal edges changed F1 by −0.1 percentage points; pruning weak edges at thresholds from 0.05 to 0.80 left F1 unchanged. In a model-agnostic stress test built from perturbed copies of test frauds, no model resisted camouflage: at moderate strength the Fuzzy-GAT detected 47.2% of cases, against 56.4% for Logistic Regression. On this benchmark, fuzzy edge weighting added computational cost without measurable benefit; whether it helps where genuine relational metadata exist remains open.

Keywords: Fraud detection, Fuzzy graph theory, Graph attention network, Graph neural networks, Class imbalance, Fraudster camouflage

INTRODUCTION

The rapid digitization of financial services such as online banking, mobile payments, and automated fund transfers has expanded convenience while widening the attack surface available to fraudsters. Global payment-card fraud losses rose from $27.85 billion in 2018 to $28.65 billion in 2019 (Nilson Report, 2019, 2020), underscoring the need for adaptive detection mechanisms. The pressure is acute in Nigeria, where banks lost over ₦42.6 billion to fraud and forgery in the second quarter of 2024 alone, exceeding the total for 2023 (Okikiola et al., 2026), and where the investment-advisory sector faces a parallel escalation (Fatokun et al., 2025). Fraud in digital banking is rarely an isolated event: it typically exploits weakly defined relationships among customers, accounts, devices, and transactions, manifesting through shared devices, overlapping IP addresses, and coordinated micro-transactions designed to evade fixed detection thresholds (Bhattacharyya et al., 2011; Carneiro et al., 2017; Kirkos et al., 2007; West & Bhattacharya, 2016; Xiang et al., 2025).

Rule-based and classical statistical detection systems rely on static thresholds and expert heuristics that struggle to keep pace with evolving fraud tactics (Bolton & Hand, 2002; Ngai et al., 2011; Phua et al., 2010). Machine-learning models improved on this by learning directly from historical transaction data (Fu et al., 2016; Jurgovsky et al., 2018; Kirkos et al., 2007), but most treat transactions as independent instances, overlooking the relational structure through which coordinated fraud propagates. Graph Neural Networks (GNNs) address this gap by modeling accounts, devices, and transactions as interconnected nodes, enabling the detection of fraud rings and collusive behavior (Tian et al., 2023). However, standard GNNs assume crisp, deterministic edges, which cannot represent the partial, weak, or probabilistic relationships such as infrequent device sharing or partial identity overlap that are often the only trace left by camouflaged fraud (Dou et al., 2020; Xiang et al., 2025).

Fuzzy graph theory offers a natural remedy: by allowing edges to carry continuous membership values in [0, 1], a fuzzy graph represents degrees of confidence in a relationship rather than forcing a binary presence/absence decision (Pourhabibi et al., 2020; Charizanos et al., 2024). Combining this representation with attention-based graph learning creates an opportunity to build a detector that is uncertainty-aware by construction and that retains, rather than discards, the weak relational evidence through which camouflaged fraud is connected.

This paper develops a Fuzzy Graph Attention Network (Fuzzy-GAT) and tests, on the publicly available European Credit Card Fraud benchmark, whether fuzzy edge weighting improves fraud detection. The objectives are to (i) construct a fuzzy transaction graph whose edge memberships encode graded similarity; (ii) integrate that graph with a graph attention mechanism whose coefficients are modulated by edge membership; (iii) compare the model with classical machine-learning and crisp-graph baselines over five random splits, at both the default and a validation-optimised decision threshold, with paired significance tests; and (iv) isolate, through ablation and threshold-sensitivity analysis, the contribution of weak-signal edge preservation, and assess robustness under a controlled camouflage stress test. Contrary

to the motivating hypothesis, the experiments show no measurable benefit from fuzzy weighting on this benchmark.

Related Work

Machine Learning and Graph-based Fraud Detection

Classical classifiers such as decision trees, support vector machines, random forests, and gradient-boosted ensembles improved on rule-based systems by learning nonlinear patterns directly from transaction data (Juszczak et al., 2008; Kirkos et al., 2007), and comparable ensemble methods have proved effective for text-borne financial fraud such as fraudulent SMS, where a Random Forest classifier reached 97% accuracy (Dawaki et al., 2022), while deep sequential models such as CNNs and LSTMs captured temporal transaction dynamics (Fu et al., 2016; Jurgovsky et al., 2018). On the same European benchmark used here, recent studies report that a stacked Random Forest–XGBoost ensemble with SMOTE-Tomek resampling reaches an F1-score of 0.93 (Akakoh & Edegbe, 2026), that SMOTE-based preprocessing materially shifts the precision–recall trade-off among classical classifiers (Ojo et al., 2026), and that an ablation-controlled CNN attains a Matthews correlation coefficient of 0.71 under stratified five-fold cross-validation (Ohwomado et al., 2026). All of these approaches assume that observations are independent, limiting their ability to detect fraud that spans multiple linked entities. Graph-based methods correct this by modeling relational structure explicitly, and systematic reviews report that GNN-based anomaly detectors outperform non-relational models across credit-card, telecommunications, and online-payment domains (Pourhabibi et al., 2020). Graph Attention Networks (GATs) extend this further by weighting neighbors according to relevance rather than treating all connections equally (Veličković et al., 2018), which is particularly valuable in the highly imbalanced settings typical of fraud data.

Fraudster Camouflage and the Limits of Crisp Graphs

Two structural problems limit standard GNNs in adversarial fraud settings. First, over-smoothing causes node representations to converge toward indistinguishability as aggregation depth increases (Oono & Suzuki, 2020), which is especially damaging in the dense financial graphs fraudsters exploit to blend in. Second, fraudster camouflage where the deliberate mimicry of legitimate behavior and embedding within benign neighborhoods causes fraudulent nodes to inherit benign features during indiscriminate neighborhood aggregation (Dou et al., 2020). Enhanced aggregation strategies such as CARE-GNN (Dou et al., 2020) and adaptive sampling GNNs (Tian et al., 2023) mitigate these effects but continue to rely on deterministic, binary edges that cannot represent partial or ambiguous relationships (Xiang et al., 2025).

Fuzzy Logic for Uncertainty Modeling

Fuzzy logic (Zadeh, 1965) provides a mathematically grounded alternative to crisp classification by allowing variables including edge relationships in a graph to take graded membership values rather than binary states. Fuzzy graphs (Rosenfeld, 1975; Mordeson & Nair, 2000) extend this principle to relational data, and systematic reviews indicate that fuzzy-based fraud detectors generally achieve lower false-positive rates than rule-based or crisp models because they avoid abrupt threshold decisions (Pourhabibi et al., 2020). Charizanos et al. (2024) demonstrate an online fuzzy logistic-regression framework that handles class imbalance and non-stationarity effectively but, like most fuzzy-only approaches, lacks relational (graph) learning capability. Conversely, recent graph-based detectors such as risk-aware gated temporal attention (Xiang et al., 2025), reinforcement-learning GAT fusion (Devi et al., 2025), heterogeneous GNNs with graph attention (Sha et al., 2025), regularized memory graph attention capsule networks (Shi et al., 2025), hypergraph contrastive learning (Wang et al., 2025), and relational graph convolutional networks (Huo et al., 2025) achieve strong predictive performance but omit explicit uncertainty modeling of edges. This gap between fuzzy uncertainty handling and deep graph-relational learning motivates the framework proposed here.

MATERIALS AND METHODS

Figure 1 summarizes the five-stage pipeline; each stage is described in the subsections that follow. The study is confined to the publicly available benchmark described next; no proprietary or live transaction data were used.

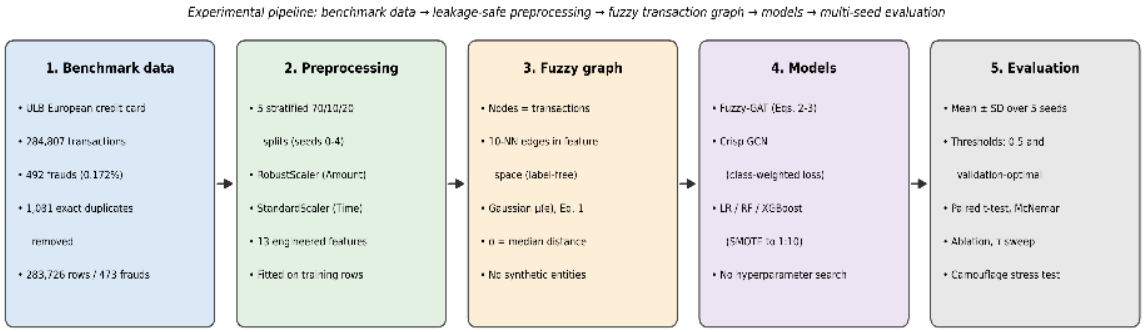

Figure 1: Overview of the Fuzzy-GAT fraud-detection pipeline from benchmark data to evaluation

Dataset

The European Credit Card Fraud Detection dataset, published by the Université Libre de Bruxelles Machine Learning Group (Dal Pozzolo et al., 2015), was used as the benchmark. It contains 284,807 transactions made by European cardholders over two days in September 2013, of which 492 (0.172%) are confirmed fraudulent, a fraud-to-legitimate ratio of approximately 1:578. With the exception of Amount and Time, all 28 features (V1–V28) are Principal Component Analysis transforms released to protect cardholder confidentiality; consequently, the dataset provides no native account, device, or IP-address identifiers, a constraint that directly shapes the graph construction described below. The severe class imbalance reflects real-world conditions and motivated the imbalance corrections described below.

Data Preprocessing

Exact duplicate records (1,081 rows, 0.38%, including 19 frauds) were removed before any splitting, leaving 283,726 transactions and 473 frauds; no values were missing. The data

were then partitioned with a stratified 70/10/20 train/validation/test split (198,608 / 28,372 / 56,746 transactions, with 95 frauds in each test partition), repeated for five random seeds (0–4). Within each split, every statistic was estimated on the training partition only. Amount was scaled with a RobustScaler and Time with a StandardScaler, and thirteen features were engineered: the Euclidean distance of V1–V28 from the legitimate-class training mean, an Isolation Forest anomaly score (Liu et al., 2008), the number of V-features lying beyond two standard deviations, the sums of absolute and of squared V-values, five pairwise products (V1·V2, V9·V10, V16·V18, V14·V17 and V12·V14), a sine–cosine encoding of time of day, and the number of transactions within ±60 s. The last feature uses timestamps only, never labels. The resulting 43 features were standardised on the training partition to form the node attributes, and the tabular baselines used the same 43 features.

Fuzzy Graph Construction

The transaction network was represented as a fuzzy graph G = (V, E, μ), where V is the set of 283,726 transactions, E the set of relational edges, and μ: E → [0, 1] a fuzzy membership function assigning a confidence value to each edge. Because the benchmark carries no account, device or IP-address identifiers, relations were defined by similarity: each transaction was joined to its k = 10 nearest neighbours in a label-free space formed by V1–V28, log (1 + Amount) and Time, standardised over all transactions, and the neighbour lists were symmetrised. This produced 2,027,144 undirected edges.

No account, device or IP entities were synthesised. The graph is built from transaction features alone and never uses labels, so its construction cannot encode knowledge of which transactions are fraudulent. Because all transactions, including validation and test transactions, are present as unlabelled nodes, the setting is transductive: the loss is computed on training nodes only, and validation and test nodes contribute features but no labels during training. The graph captures feature-space similarity, not verified links between customers or devices. The experiments therefore test whether fuzzy weighting of similarity edges helps; they do not test fraud-ring detection on real relational data.

Edge weights were assigned with a Gaussian membership function,

$$\mu(e) = \exp\left(\frac{-d(e)^2}{2\sigma^2}\right) \quad (1)$$

Where d(e) is the Euclidean distance between the two transactions joined by edge e in the neighbour space, and σ = 1.386 is the median neighbour distance. Edges were categorised as high-confidence (μ ≥ 0.8), moderate-confidence (0.5 ≤ μ < 0.8) or weak-signal (μ < 0.5), and the full model retained all of them. Of the undirected edges, 26.2% were high-confidence, 27.0% moderate and 46.7% weak-signal; these shares follow from the choice of σ and describe the construction rather than a finding. Each node also received a self-loop with membership 1.

Fuzzy Graph Attention Network

A Graph Attention Network (Veličković et al., 2018) was selected as the relational learning backbone because its attention mechanism is directly compatible with fuzzy edge weighting. For a node i and a neighbor j, the unnormalized attention logit e_{ij} is given by Equation (2) and the fuzzy-weighted attention coefficient α_{ij} by Equation (3):

$$e_{ij} = \text{LeakyReLU}\left(a^T\left[Wh_i \parallel Wh_j\right]\right) \quad (2)$$

$$\alpha_{ij} = \frac{\mu(i,j)\cdot\exp(e_{ij})}{\sum_{k\in N(i)}\mu(i,k)\cdot\exp(e_{ik})} \quad (3)$$

Where h_i is the feature vector of node i, W a shared linear transformation, a the attention vector, ‖ concatenation, and N(i) the neighbourhood of node i including its self-loop. Multiplying each softmax term by μ(i, j) lets high-confidence edges dominate aggregation, while weak-signal edges contribute in proportion to their membership instead of being discarded. The design aims to temper indiscriminate neighbourhood aggregation, a known weakness of deep graph networks (Oono & Suzuki, 2020). The network (Figure 2) has three attention layers: 8 heads of 16 units (128 units concatenated), 4 heads of 16 units (64), and 1 head of 32 units. It uses ELU activations, dropout of 0.3 after the first two layers, and a linear output unit who's sigmoid gives the fraud score. Training was full-batch on the training nodes, with binary cross-entropy whose positive-class weight equals the legitimate-to-fraud ratio of the training partition (about 577), so that missed frauds cost more than false alarms (Bahnsen et al., 2016). The optimiser was Adam (learning rate 0.001, weight decay 5×10^{-4}) with cosine annealing over at most 150 epochs. Training stopped when validation PR-AUC had not improved for 20 epochs, and the weights from the best epoch were restored.

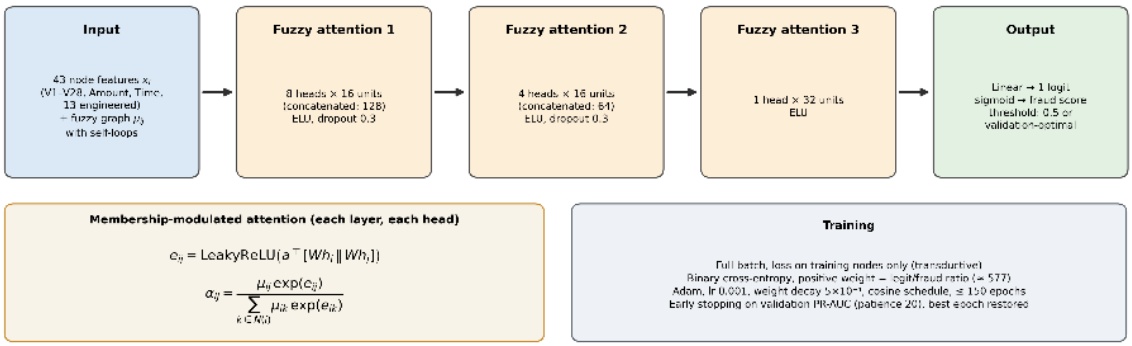

Figure 2: Fuzzy Graph Attention Network architecture, with 43 node features, three membership-modulated attention layers and a sigmoid output

Baseline Model Configuration

Logistic Regression (with feature standardisation), Random Forest (Breiman, 2001) and XGBoost (Chen & Guestrin, 2016) were trained on the 43 tabular features with library default hyperparameters. Their training partitions were oversampled with SMOTE (Chawla et al., 2002) to a 1:10 fraud-to-legitimate ratio, and no class weighting was applied. The graph baseline was a three-layer Graph Convolutional Network (Kipf & Welling, 2017) with the same layer widths, dropout, loss and optimiser as the Fuzzy-GAT, operating on the crisp graph (edges with μ ≥ 0.5, all weighted 1). Each model family therefore uses exactly one imbalance

correction: SMOTE for the tabular models, and a class-weighted loss for the graph models, because synthetic SMOTE samples cannot join a graph without inventing their edges. Neither the baselines nor the Fuzzy-GAT received a hyperparameter search. To avoid comparing an untuned threshold with a tuned one, every model is reported at two operating points: the default threshold of 0.5, and the F1-optimal threshold selected on the validation partition and applied unchanged to the test partition (Branco et al., 2016).

Evaluation Protocol

Performance on each test partition was measured by precision, recall, F1-score, ROC-AUC, PR-AUC (average precision), the Matthews's correlation coefficient (MCC) and accuracy. A classifier that labels every transaction legitimate already scores 99.83% accuracy, so precision, recall, F1-score and PR-AUC carry the comparison (Fawcett, 2006; Saito & Rehmsmeier, 2015); recent studies on this benchmark likewise rely on minority-class metrics (Akakoh & Edegbe, 2026; Ohwomado et al., 2026). Results are reported as mean ± standard deviation over the five seeds. Differences between the Fuzzy-GAT and each baseline were tested with a paired t-test on the per-seed F1-scores and, for seed 0, with McNemar's test on the test-set predictions (McNemar, 1947; Dietterich, 1998). The ablation compared four graph variants over the same three seeds (0–2): crisp edges, all edges with uniform weight, fuzzy weights with weak-signal edges removed, and the full model. The sensitivity analysis retrained the Fuzzy-GAT on seed 0 with edges of $\mu < \tau$ removed, for $\tau \in \{0.05, 0.20, 0.35, 0.50, 0.65, 0.80\}$. All experiments ran on an 8-core CPU without a GPU, using PyTorch 2.12, scikit-learn 1.8, imbalanced-learn 0.14 and XGBoost 3.4. A Fuzzy-GAT training run typically took 17–33 minutes, against 4–8 minutes for the GCN.

Robustness to camouflage was assessed with a model-agnostic stress test. Each of the 95 test frauds of seed 0 was moved a fraction λ of the way toward the legitimate-class training mean in V1–V28, and its Amount toward the dataset median, with $\lambda \in \{0, 0.25, 0.50, 0.75\}$. Four further copies of each fraud received small Gaussian noise (0.05 standard deviations per feature), giving 475 cases per λ. perturbed transactions entered the graph through their ten nearest existing transactions, and detection was measured as recall at each model's validation-optimised threshold. The perturbation acts on features only, applies identically to every model, and was defined before the experiments were run, so it does not favour the fuzzy architecture. It remains a synthetic approximation of camouflage (Dou et al., 2020), and the replicates are not independent frauds.

RESULTS AND DISCUSSION

Dataset Characteristics and Fuzzy Graph Structure

Exploratory analysis confirmed the severe class imbalance (1:578) and showed that fraudulent transactions cluster at lower amounts (median 9.25, against 22.00 for legitimate transactions), consistent with card-testing behaviour. The PCA features are mutually uncorrelated, and V17, V14, V12 and V10 are the most strongly associated with the class (Figure 3). In the fuzzy graph, fraud is strongly homophilous: 72.7% of the neighbours of fraudulent transactions are themselves fraudulent. Figure 4 shows the neighbourhood of one correctly detected test fraud: a crisp graph keeps a single edge, whereas the fuzzy graph keeps all ten, most with low membership. Fraud–fraud edges form nine connected clusters, each containing at least one weak-signal edge, and 80.5% of fraudulent transactions have a weak-signal edge to another fraud (Figure 5). Weak edges are therefore common around fraud, but, as the ablation below shows, keeping them did not change detection.

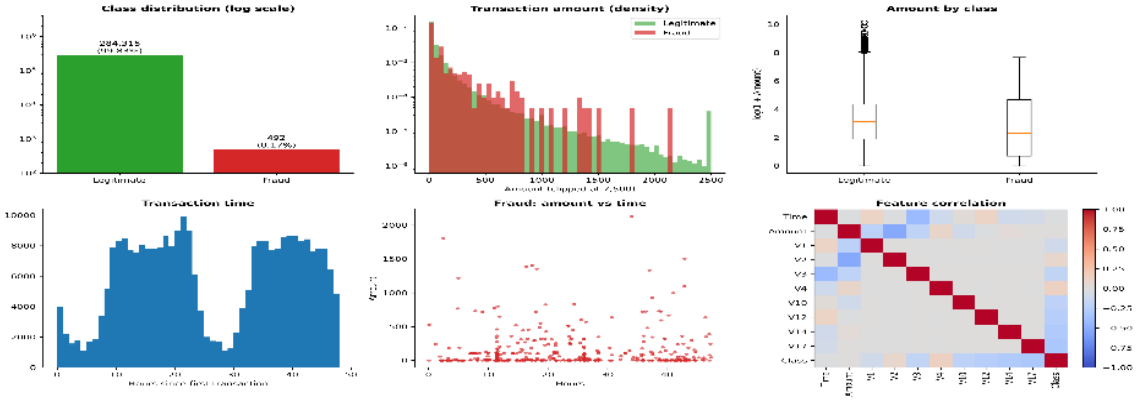

Figure 3: Exploratory data analysis: class distribution, transaction amount, amount by class, transaction time, fraud amount versus time, and feature correlations

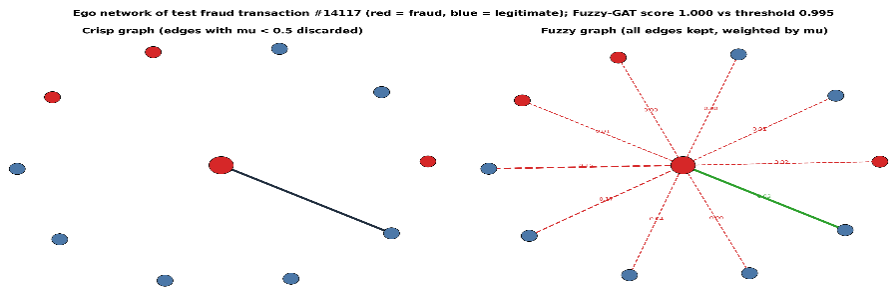

Figure 4: Ten-nearest-neighbour ego network of a correctly detected test fraud (red = fraud, blue = legitimate) in the crisp graph (left; edges with $\mu < 0.5$ discarded) and the fuzzy graph (right; green $\mu \geq 0.8$, orange $0.5 \leq \mu < 0.8$, red dashed $\mu < 0.5$)

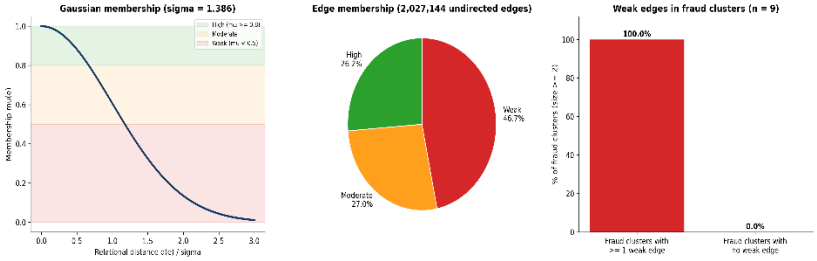

Figure 5: Gaussian membership function (left), distribution of edge memberships (centre), and share of fraud clusters containing a weak-signal edge (right)

Comparative Model Performance

Table 1 and Figure 6 compare the five models over five seeds. At the validation-optimised threshold, XGBoost (82.8 ± 2.4%) and Random Forest (82.1 ± 3.3%) achieved the highest F1-scores and PR-AUCs. The Fuzzy-GAT (79.3 ± 2.9%), the crisp GCN (79.9 ± 3.0%) and Logistic Regression (79.4 ± 1.8%) were statistically indistinguishable (paired t-test p = 0.29 against the GCN and p = 0.98 against Logistic Regression). The Fuzzy-GAT scored below Random Forest and below XGBoost in all five seeds (mean differences of −2.8 and −3.5 points; p = 0.014 and p = 0.002). On seed 0, McNemar's test was significant against XGBoost (p = 0.039) but not against Random Forest (p = 0.092). The operating point matters. The tree ensembles performed best at the default threshold of 0.5 (F1 84.0% and 84.3%), whereas the class-weighted graph models, whose scores the positive-class weight pushes upward, flagged thousands of legitimate transactions at 0.5 (F1 8–10%, accuracy 96.6–97.4%, below the 99.83% no-skill level). Figure 7 shows the seed-0 confusion matrices at the tuned threshold: the Fuzzy-GAT caught 75 of 95 frauds with 11 false alarms, against 74 with 2 for XGBoost.

Table 1: Test Performance over Five Seeds (Mean ± SD; 56,746 Transactions and 95 Frauds per Test Partition) At the Validation-Optimised Threshold, With the F1-Score at the Default Threshold for Reference

Model	Precision (%)	Recall (%)	F1 (%)	ROC-AUC	PR-AUC	MCC (mean)	F1 at 0.5 (%)
Logistic Regression	87.1 ± 2.2	73.1 ± 4.1	79.4 ± 1.8	0.974 ± 0.010	0.758 ± 0.020	0.797	55.0 ± 3.1
Random Forest	92.4 ± 7.2	74.1 ± 2.8	82.1 ± 3.3	0.954 ± 0.014	0.810 ± 0.040	0.827	84.0 ± 3.6
XGBoost	94.8 ± 4.8	73.7 ± 3.1	82.8 ± 2.4	0.974 ± 0.008	0.814 ± 0.036	0.835	84.3 ± 2.0
Standard GCN (crisp edges)	89.0 ± 1.6	72.6 ± 5.8	79.9 ± 3.0	0.976 ± 0.011	0.711 ± 0.031	0.803	10.3 ± 1.2
Fuzzy-GAT (proposed)	88.3 ± 2.4	72.2 ± 4.9	79.3 ± 2.9	0.969 ± 0.012	0.696 ± 0.027	0.798	8.1 ± 1.1

Library-default hyperparameters for the tabular models. Accuracy at the tuned threshold is 99.93–99.95% for every model. Graph-model PR-AUCs are slightly conservative: float32 scores saturated at 1.0 for the most confident transactions, and recomputing seed 0 from logits raises the Fuzzy-GAT's PR-AUC from 0.705 to 0.746, still below Random Forest (0.876) and XGBoost (0.868) on the same split.

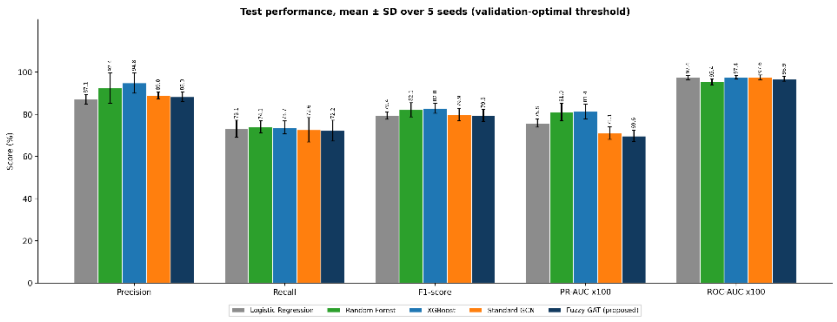

Figure 6: Test performance of all models, mean ± SD over five seeds at the validation-optimised threshold (graph-model PR-AUC is slightly conservative; see the note to Table 1)

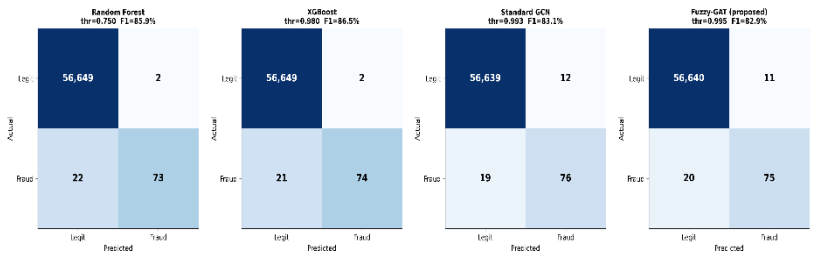

Figure 7: Seed-0 test confusion matrices at the validation-optimised threshold for Random Forest (TN 56,649, FP 2, FN 22, 0 TP 73), XGBoost (56,649 / 2 / 21 / 74), the standard GCN (56,639 / 12 / 19 / 76) and the Fuzzy-GAT (56,640 / 11 / 20 / 75)

Comparison with Recent State-of-the-Art Methods

Table 2 places these results beside recently published detectors, using the values those studies report. The comparison is indicative only, because datasets, de-duplication, splits and resampling differ, and none of the cited models was re-run here. Four of the cited studies use the same ULB benchmark (Shi et al., 2025; Akakoh & Edegbe, 2026; Ohwomado et al., 2026; Ojo et al., 2026), and those reporting F1 (93.0–98.8%) exceed the 79.3% measured here. Part of the gap may reflect protocol differences such as duplicate removal and averaging over splits. Either way, the comparison gives no support to a claim that fuzzy graph attention outperforms well-configured non-relational models on this benchmark.

Table 2: Comparison with Recent Fraud-Detection Studies (Values As Reported by the Cited Authors)

Study	Method	Dataset	Acc. (%)	F1 (%)	ROC-AUC	Fuzzy	Graph
Kirkos et al. (2007)	Decision tree / neural network / Bayesian belief network	Greek firms' financial statements	73.6–90.3	N/R	N/R	No	No
Dou et al. (2020)	CARE-GNN	YelpChi / Amazon	N/R	N/R	0.757/0.897	No	Yes
Tian et al. (2023)	ASA-GNN	Bank transactions (PR01)	N/R	90.4	0.922	No	Yes
Charizanos et al. (2024)	Online fuzzy logistic regression	Credit-card transactions	>99.0	N/R	N/R	Yes	No
Xiang et al. (2025)	RGTAN (risk-aware gated temporal attention)	YelpChi / Amazon / S-FFSD	N/R	84.9/92.0/ 75.1	0.950/0.975 /0.846	No	Yes
Shi et al. (2025)	RMGACNet	ULB European credit card	97.72	97.70	N/R	No	Yes
Devi et al. (2025)	RL-GNN (GAT + reinforcement learning)	IEEE-CIS	N/R	N/R	0.872	No	Yes
Huo et al. (2025)	RGCN	IBM credit-card transactions	99.88	N/R	N/R	No	Yes
Sha et al. (2025)	Heterogeneous GNN + graph attention	IEEE-CIS	N/R	N/R	N/R	No	Yes
Akakoh & Edegbe (2026)	Stacked RF + XGBoost ensemble with SMOTE-Tomek	ULB European credit card	N/R	93.0	N/R	No	No
Ojo et al. (2026)	LinearSVC / LR / decision tree / RF with SMOTE	ULB European credit card	99.99	98.81	N/R	No	No
Ohwomado et al. (2026)	Ablation-optimised CNN (5-fold CV)	ULB European credit card	99.88	N/R	0.966	No	No
This study	Fuzzy-GAT (5 seeds)	ULB European credit card (de-duplicated; k-NN transaction graph)	99.94	79.3	0.969	Yes	Yes

N/R = not reported in the cited source. Multiple values indicate results on multiple datasets in the order listed. Values for this study are means over five seeds at the validation-optimised threshold.

Ablation Study

Table 3 and Figure 8 compare the four graph variants over the same three seeds. F1-scores lay between 77.1% and 79.7%, and every difference was smaller than the seed-to-seed standard deviation (2.8–3.7 points). Crisp binary edges scored highest (79.7 ± 2.8%) and uniform weights on all edges lowest (77.1 ± 3.7%). The full Fuzzy-GAT (79.2 ± 3.3%) scored 0.1 points below the same model with weak-signal edges removed (79.4 ± 3.1%). PR-AUC and ROC-AUC show the same pattern. Neither fuzzy weighting nor weak-signal preservation produced a measurable improvement.

Table 3: Ablation of Fuzzy Weighting Components (Mean ± SD over Seeds 0–2, Validation-Optimised Threshold)

Model variant	Precision (%)	Recall (%)	F1 (%)	ROC-AUC	PR-AUC
Crisp binary edges ($\mu \geq 0.5$)	88.5 ± 1.1	72.6 ± 5.5	79.7 ± 2.8	0.972 ± 0.017	0.688 ± 0.039
All edges, uniform weight	85.2 ± 3.3	70.5 ± 4.8	77.1 ± 3.7	0.963 ± 0.022	0.676 ± 0.040
Fuzzy weights, weak-signal edges removed	87.7 ± 1.3	72.6 ± 5.5	79.4 ± 3.1	0.972 ± 0.017	0.689 ± 0.039
Fuzzy-GAT (full)	87.4 ± 2.9	72.6 ± 5.5	79.2 ± 3.3	0.967 ± 0.016	0.687 ± 0.033

Crisp and no-weak-signal variants keep only edges with $\mu \geq 0.5$; the uniform and full variants keep all edges.

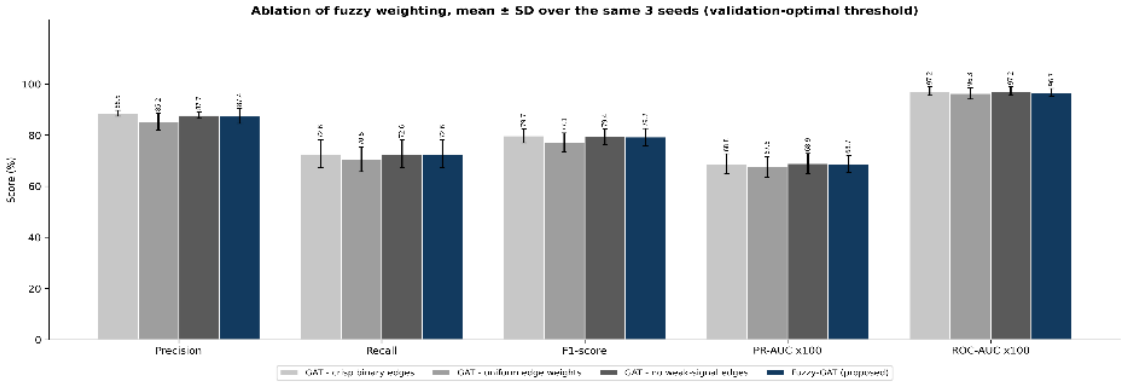

Figure 8: Ablation of fuzzy weighting: test performance of the four graph variants, mean ± SD over three seeds

Sensitivity Analysis and Weak-Signal Contribution

Removing edges with μ below τ, from τ = 0.05 (1.85 million edges kept) to τ = 0.80 (0.53 million), left the seed-0 test F1-score at exactly 82.9% in every case, with identical precision (87.2%) and recall (78.9%): each pruned model reached the same confusion counts at its tuned threshold. PR-AUC varied only between 0.708 and 0.723, and detection of perturbed frauds at λ = 0.5 between 42.3% and 47.4%, without a trend (Figure 9). Cutting the graph to about a quarter of its edges therefore had no measurable effect, consistent with the ablation.

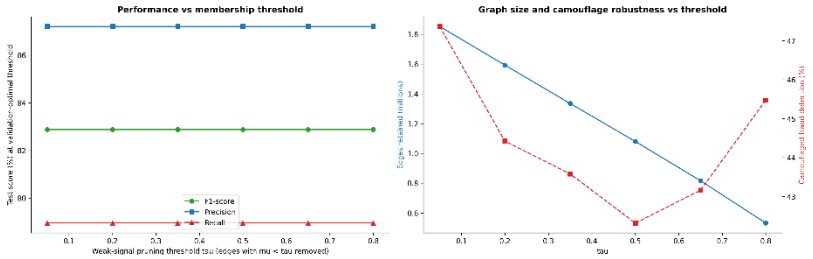

Figure 9: Sensitivity of the seed-0 Fuzzy-GAT to the weak-signal pruning threshold τ: test precision, recall and F1-score (left), and edges retained with detection of perturbed frauds at λ = 0.5 (right)

Robustness to Fraudster Camouflage

Table 4 and Figure 10 report detection of the perturbed frauds. Without camouflage (λ = 0), the models detected 76–82% of cases, and the crisp GCN detected the most. Detection fell steeply for every model as λ increased. At λ = 0.25 the Fuzzy-GAT was marginally best (78.9%, against 76.8% for Random Forest). At λ = 0.50 it detected 47.2%, behind Logistic Regression (56.4%) and XGBoost (51.8%), and at λ = 0.75 every model detected between 14.5% and 22.9%. No model, including the Fuzzy-GAT, was robust to this form of camouflage.

Table 4: Detection Rate (%) of Perturbed Test Frauds by Camouflage Strength λ (seed 0; n = 475 per λ)

Model	λ = 0	λ = 0.25	λ = 0.50	λ = 0.75
Logistic Regression	78.9	69.9	56.4	14.5
Random Forest	76.4	76.8	44.8	16.2
XGBoost	78.1	75.8	51.8	22.9
Standard GCN (crisp edges)	82.1	74.9	41.3	14.7
Fuzzy-GAT (proposed)	78.9	78.9	47.2	16.8

λ is the fraction of the distance moved toward the legitimate-class mean; detection is recall at each model's validation-optimised threshold.

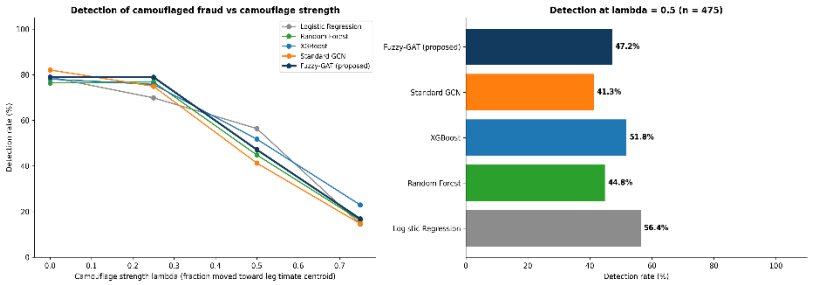

Figure 10: Detection of perturbed frauds as camouflage strength increases (left) and at λ = 0.5 (right)

Discussion

Three findings emerge. First, on this benchmark the Fuzzy-GAT did not outperform simple baselines: it matched Logistic Regression and a crisp GCN, and Random Forest and XGBoost were significantly better. Second, the mechanism the architecture was built around had no measurable effect. Fuzzy weights, uniform weights and crisp edges gave F1-scores within one standard deviation of each other, and pruning weak-signal edges at any threshold left F1 unchanged. A likely explanation is that the graph adds little information here. Its edges derive from the same features the nodes already carry, and fraud is so homophilous in that feature space (72.7% of fraud neighbours are fraud) that even a crisp graph connects most frauds to each other. The strength of each edge then matters little, and attention can learn to discount uninformative neighbours on its own. The arguments for fuzzy relational modelling (Pourhabibi et al., 2020) concern relations that carry information beyond the node features, such as shared devices or accounts, and the present data cannot test them.

Third, camouflage defeated every model. Because the graph is built from features, a fraud moved toward the legitimate profile also acquires legitimate neighbours, so the relational view is corrupted along with the features and offers no independent line of defence (Dou et al., 2020). A graph could add robustness only through relational evidence that fraudsters cannot easily alter, such as device or account links. The approach also has a practical cost. On the test machine, scoring one new transaction end to end took about 190 ms for the Fuzzy-GAT, mostly for neighbour search and subgraph

extraction, against about 90 ms for XGBoost; in batches of 1,000, the Fuzzy-GAT scored about 105 transactions per second and XGBoost about 7,600. Finally, the multi-seed protocol proved essential. Individual splits gave Fuzzy-GAT F1-scores between 76.3% and 82.9%, a range wide enough for a single favourable split to suggest an advantage that the average does not support.

Limitations and Threats to Validity

Several limitations qualify these conclusions. (i) The benchmark carries no account, device or IP identifiers, so the graph encodes feature similarity rather than verified relationships. The results show that fuzzy weighting of similarity edges does not help; they do not show that fuzzy relational modelling cannot help where genuine relational metadata exist. (ii) The setting is transductive: test transactions are present as unlabelled nodes during training, which is standard for node classification but not for streaming deployment. (iii) No model received a hyperparameter search, and tuning could change the ranking, although reporting two thresholds removes the largest source of asymmetry. (iv) The ablation used three seeds, and the sensitivity and camouflage analyses used one, so small effects could go undetected. (v) The camouflage test is a synthetic, feature-space perturbation of 95 frauds with noisy replicates, not a sample of real camouflaged fraud. (vi) The evaluation uses one benchmark, collected from European cardholders over two days in 2013; performance on the mobile-money-dominated patterns of sub-Saharan African markets, where financial-crime exposure is acute (Deloitte, n.d.; FATF et al., 2023; Fatokun et al., 2025; Okikiola et al., 2026), is untested. (vii) Only a Gaussian membership function with a median-distance bandwidth was studied. (viii) Graph-model scores were stored as float32 probabilities, which saturated for the most confident transactions and understate graph-model PR-AUC by up to 0.04; the F1-scores and every conclusion are unaffected.

CONCLUSION

This study developed a Fuzzy Graph Attention Network for credit card fraud detection and tested it on the European Credit Card Fraud benchmark over five stratified splits. The model reached an F1-score of 79.3 ± 2.9% and a ROC-AUC of 0.969 ± 0.012. It matched Logistic Regression and a crisp Graph Convolutional Network, and Random Forest and XGBoost significantly outperformed it. Fuzzy edge weighting and weak-signal edge preservation brought no measurable benefit, pruning weak edges left performance unchanged, and no model was robust to feature-space camouflage. Where the only relations are derived from transaction features, uncertainty-aware edge weighting therefore adds computational cost without improving detection. The findings do not rule out a benefit for graphs that carry independent relational evidence, and they show that architectural gains on this benchmark should be claimed only after multi-seed evaluation.

Future work should (i) test fuzzy graph attention on datasets with genuine account, device or merchant relationships, where edges carry information beyond the node features; (ii) tune all models, including the graph models, under a common search budget; (iii) evaluate inductively, scoring streaming transactions against a graph built only from past data; (iv) extend evaluation to regional and mobile-money datasets, particularly from sub-Saharan Africa; (v) compare Gaussian, trapezoidal, sigmoid and learnable membership functions; and (vi) assess robustness against realistic and generative-AI-crafted camouflage.

REFERENCES

Akakoh, U. I., & Edegbe, G. N. (2026). Mitigating financial fraud: A hybrid SMOTE-Tomek and stacked ensemble model approach. FUDMA Journal of Sciences, 10(14), 210–217. https://doi.org/10.33003/fjs-2026-1014-5398

Bahnsen, A. C., Aouada, D., Stojanovic, A., & Ottersten, B. (2016). Feature engineering strategies for credit card fraud detection. Expert Systems with Applications, 51, 134–142. https://doi.org/10.1016/j.eswa.2015.12.030

Bhattacharyya, S., Jha, S., Tharakunnel, K., & Westland, J. C. (2011). Data mining for credit card fraud: A comparative study. Decision Support Systems, 50(3), 602–613. https://doi.org/10.1016/j.dss.2010.08.008

Bolton, R. J., & Hand, D. J. (2002). Statistical fraud detection: A review. Statistical Science, 17(3), 235–255. https://doi.org/10.1214/ss/1042727940

Branco, P., Torgo, L., & Ribeiro, R. P. (2016). A survey of predictive modelling under imbalanced distributions. ACM Computing Surveys, 49(2), Article 31. https://doi.org/10.1145/2907070

Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5–32. https://doi.org/10.1023/A:1010933404324

Carneiro, N., Figueira, G., & Costa, M. (2017). A data mining based system for credit-card fraud detection in e-tail. Decision Support Systems, 95, 91–101. https://doi.org/10.1016/j.dss.2017.01.002

Charizanos, G., Demirhan, H., & İçen, D. (2024). An online fuzzy fraud detection framework for credit card transactions. Expert Systems with Applications, 252, Article 124127. https://doi.org/10.1016/j.eswa.2024.124127

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. Journal of Artificial Intelligence Research, 16, 321–357. https://doi.org/10.1613/jair.953

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 785–794). ACM. https://doi.org/10.1145/2939672.2939785

Dal Pozzolo, A., Caelen, O., Johnson, R. A., & Bontempi, G. (2015). Calibrating probability with undersampling for unbalanced classification. In Proceedings of the 2015 IEEE Symposium Series on Computational Intelligence (pp. 159–166). IEEE. https://doi.org/10.1109/SSCI.2015.33

Dawaki, M., Mohammed, A., & Ismail, M. (2022). Fraudulent text detection system using hybrid machine learning and natural language processing approaches. International Journal of Innovative Science and Research Technology, 7(10), 1110–1118. https://doi.org/10.5281/zenodo.7313563

Deloitte. (n.d.). Future of financial crime. Deloitte Africa. Retrieved September 3, 2026, from https://www.deloitte.com/za/en/services/consulting/perspectives/future-of-financial-crime.html

Devi, R. R., Raja, J. E., & Chin, Y. B. (2025). Reinforcement learning with graph neural network (RL-GNN) fusion for real-time financial fraud detection: A context-aware community mining approach. Scientific Reports, 15, Article 42953. https://doi.org/10.1038/s41598-025-25200-3

Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. Neural Computation, 10(7), 1895–1923. https://doi.org/10.1162/089976698300017197

Dou, Y., Liu, Z., Sun, L., Deng, Y., Peng, H., & Yu, P. S. (2020). Enhancing graph neural network-based fraud detectors against camouflaged fraudsters. In Proceedings of the 29th ACM International Conference on Information & Knowledge Management (pp. 315–324). ACM. https://doi.org/10.1145/3340531.3411903

FATF, INTERPOL, & Egmont Group. (2023). Illicit financial flows from cyber-enabled fraud. Financial Action Task Force. https://www.fatf-gafi.org/content/dam/fatf-gafi/reports/Illicit-financial-flows-cyber-enabled-fraud.pdf.coredownload.inline.pdf

Fatokun, J. O., Mustapha, S., Balogun, F., & Okorie, D. D. (2025). Fraud detection in Nigerian investment advisory sector using machine learning algorithms. FUDMA Journal of Sciences, 9(10), 5–11. https://doi.org/10.33003/fjs-2025-0910-4004

Fawcett, T. (2006). An introduction to ROC analysis. Pattern Recognition Letters, 27(8), 861–874. https://doi.org/10.1016/j.patrec.2005.10.010

Fu, K., Cheng, D., Tu, Y., & Zhang, L. (2016). Credit card fraud detection using convolutional neural networks. In A. Hirose, S. Ozawa, K. Doya, K. Ikeda, M. Lee, & D. Liu (Eds.), Neural Information Processing: ICONIP 2016 (pp. 483–490). Springer. https://doi.org/10.1007/978-3-319-46675-0_53

Huo, M., Lu, K., Zhu, Q., & Chen, Z. (2025). Enhancing customer contact efficiency with graph neural networks in credit card fraud detection workflow. In Proceedings of the 2025 IEEE 7th International Conference on Communications, Information System and Computer Engineering (CISCE) (pp. 320–324). IEEE. https://arxiv.org/abs/2504.02275

Jurgovsky, J., Granitzer, M., Ziegler, K., Calabretto, S., Portier, P.-E., He-Guelton, L., & Caelen, O. (2018). Sequence classification for credit-card fraud detection. Expert Systems with Applications, 100, 234–245. https://doi.org/10.1016/j.eswa.2018.01.037

Juszczak, P., Adams, N. M., Hand, D. J., Whitrow, C., & Weston, D. J. (2008). Off-the-peg and bespoke classifiers for fraud detection. Computational Statistics & Data Analysis, 52(9), 4521–4532. https://doi.org/10.1016/j.csda.2008.03.014

Kipf, T. N., & Welling, M. (2017). Semi-supervised classification with graph convolutional networks. In Proceedings of the 5th International Conference on Learning Representations (ICLR). https://openreview.net/forum?id=SJU4ayYgl

Kirkos, E., Spathis, C., & Manolopoulos, Y. (2007). Data mining techniques for the detection of fraudulent financial statements. Expert Systems with Applications, 32(4), 995–1003. https://doi.org/10.1016/j.eswa.2006.02.016

Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). Isolation forest. In Proceedings of the 2008 Eighth IEEE International Conference on Data Mining (pp. 413–422). IEEE. https://doi.org/10.1109/ICDM.2008.17

McNemar, Q. (1947). Note on the sampling error of the difference between correlated proportions or percentages. Psychometrika, 12(2), 153–157. https://doi.org/10.1007/BF02295996

Mordeson, J. N., & Nair, P. S. (2000). Fuzzy graphs and fuzzy hypergraphs. Physica-Verlag. https://doi.org/10.1007/978-3-7908-1854-3

Ngai, E. W. T., Hu, Y., Wong, Y. H., Chen, Y., & Sun, X. (2011). The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature. Decision Support Systems, 50(3), 559–569. https://doi.org/10.1016/j.dss.2010.08.006

Nilson Report. (2019, November). Payment card fraud losses reach $27.85 billion [Press release]. HSN Consultants. https://www.prnewswire.com/news-releases/payment-card-fraud-losses-reach-27-85-billion-300963232.html

Nilson Report. (2020, December). Card fraud losses reach $28.65 billion (Issue 1187). HSN Consultants. https://nilsonreport.com/articles/card-fraud-losses-reach-28-65-billion/

Ohwomado, K. O., Akazue, M. I., & Ojugo, A. A. (2026). A systematic ablation-based optimisation protocol for convolutional neural networks in electronic banking fraud detection. FUDMA Journal of Sciences, 10(12), 79–84. https://doi.org/10.33003/fjs-2026-1012-5285

Ojo, O. F., Otaru, P. O., Job, O. E., & Onoghojobi, B. (2026). Comparative performance of machine learning models for credit card fraud detection on imbalanced data: A study using SMOTE and the Kaggle European dataset. FUDMA Journal of Sciences, 10(3), 102–108. https://doi.org/10.33003/fjs-2026-1003-4745

Okikiola, F. M., Arogundade, T. S., Onadokun, I., Oladiboye, O. E., & Sodiq, A. K. (2026). Strengthening financial integrity: The role of cybersecurity in mitigating financial fraud in Nigeria's banking sector. FUDMA Journal of Sciences, 10(2), 175–183. https://doi.org/10.33003/fjs-2026-1002-4491

Oono, K., & Suzuki, T. (2020). Graph neural networks exponentially lose expressive power for node classification. In Proceedings of the 8th International Conference on Learning Representations (ICLR). https://openreview.net/forum?id=S1ldO2EFPr

Phua, C., Lee, V., Smith, K., & Gayler, R. (2010). A comprehensive survey of data mining-based fraud detection research. arXiv. https://arxiv.org/abs/1009.6119

Pourhabibi, T., Ong, K.-L., Kam, B. H., & Boo, Y. L. (2020). Fraud detection: A systematic literature review of graph-

based anomaly detection approaches. Decision Support Systems, 133, Article 113303. https://doi.org/10.1016/j.dss.2020.113303

Rosenfeld, A. (1975). Fuzzy graphs. In L. A. Zadeh, K. S. Fu, & M. Shimura (Eds.), Fuzzy sets and their applications to cognitive and decision processes (pp. 77–95). Academic Press. https://doi.org/10.1016/B978-0-12-775260-0.50008-6

Saito, T., & Rehmsmeier, M. (2015). The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. PLOS ONE, 10(3), Article e0118432. https://doi.org/10.1371/journal.pone.0118432

Sha, Q., Tang, T., Du, X., Liu, J., Wang, Y., & Sheng, Y. (2025). Detecting credit card fraud via heterogeneous graph neural networks with graph attention. arXiv. https://arxiv.org/abs/2504.08183

Shi, X., Wang, X., Zhang, Y., Zhang, X., Yu, M., & Zhang, L. (2025). Innovative novel regularized memory graph attention capsule network for financial fraud detection. PLOS ONE, 20(5), Article e0317893. https://doi.org/10.1371/journal.pone.0317893

Tian, Y., Liu, G., Wang, J., & Zhou, M. (2023). Transaction fraud detection via an adaptive graph neural network. arXiv. https://arxiv.org/abs/2307.05633

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., & Bengio, Y. (2018). Graph attention networks. In Proceedings of the 6th International Conference on Learning Representations (ICLR). https://openreview.net/forum?id=rJXMpikCZ

Wang, Q., Shen, Y., & Dong, H. (2025). Hypergraph-based contrastive learning for enhanced fraud detection. Frontiers in Artificial Intelligence, 8, Article 1703135. https://doi.org/10.3389/frai.2025.1703135

West, J., & Bhattacharya, M. (2016). Intelligent financial fraud detection: A comprehensive review. Computers & Security, 57, 47–66. https://doi.org/10.1016/j.cose.2015.09.005

Xiang, S., Zhang, G., Cheng, D., & Zhang, Y. (2025). Enhancing attribute-driven fraud detection with risk-aware graph representation. IEEE Transactions on Knowledge and Data Engineering, 37(5), 2501–2512. https://doi.org/10.1109/TKDE.2025.3543887

Zadeh, L. A. (1965). Fuzzy sets. Information and Control, 8(3), 338–353. https://doi.org/10.1016/S0019-9958(65)90241-X