Journal of Engineering and Technology for Industrial Applications

ISSN ONLINE: 2447-0228

ITEGAM-JETIA

Manaus, v.12 n.58, p.1352-1367. March/April, 2026.
DOI: https://doi.org/10.5935/jetia.v12i58.3345

RESEARCH ARTICLE OPEN ACCESS

AGENTIC AI IN CLOUD-BASED CREDIT CARD FRAUD DETECTION: TOWARDS AUTONOMOUS RISK MITIGATION

Sushil Prabhu Prabhakaran[*1]

[1]Full Stack Lead, Tata Consultancy Services, USA.

[1]http://orcid.org/0009-0005-9828-3014

E-mail: *sushilprab@gmail.com

ARTICLE INFO

Article History
Received: January 19, 2026
Reviewed: February 2, 2026
Accepted: March 31, 2026
Published: April 30, 2026

Keywords:
Credit card,
Fraud detection,
Cloud environment,
Risk assessment,
Transaction,
Deep leaning,
Shuffle attention,
Dynamic convolutional.

ABSTRACT

Credit card fraud (CCF) still remains an ongoing concern for financial institutions because to the huge disparity between genuine and fraudulent transactions, as well as the ever-changing behavior of fraudsters. This paper proposes a cloud-based Agentic Artificial Intelligence system for real-time credit card fraud detection that utilizes autonomous multi-agent collaboration and deep temporal modeling. The system makes use of the publicly accessible CCF Detection dataset, commencing with secure cloud-level data ingestion, preprocessing, and normalization. An agentic transaction analysis layer made up of qualified agents accomplishes data validation, behavioral pattern analysis, and transaction history verification. To successfully capture both local spatial aspects and long-term temporal dependencies in transactional behavior, deep temporal fraud modeling employs a hybrid full-dimensional dynamic convolutional network mixed with shuffle attention and LSTM. Finally, an automated risk assessment and decision module generates fraud scores and initiates relevant mitigation steps such as alarms or transaction blocks. Experimental results show that the proposed framework outperforms baseline GRU-based models in detection performance, demonstrating its effectiveness, scalability, and appropriateness for real-time intelligent fraud prevention in cloud environments.

I. INTRODUCTION

Recently, there has been an increase in the use of credit cards as a payment mechanism, with most individuals using credit cards rather than cash when making regular payments in their daily lives. In the past few decades, all humanity has benefited from internet banking services for cash transfers, debit and credit card transactions, and online bill install ment payments [1-3]. Customers can simply track their spending with a credit card and understand where their money has been spent. Despite the numerous advantages of online transactions, financial fraud and unauthorized payments offer serious risks [4]. Fraudulent financial operations are challenging to recognize due to their complex nature and advanced technology. Financial fraud is getting increasingly widespread as technology progresses [5], [6]. The financial fraud detection systems can detect unusual attacks and unlawful access. Financial organizations are continually updating their fraud detection methods. These difficulties are addressed by data mining and machine learning (ML) methods, which have become widely used in recent years [7-9]. The researchers attempt to develop fraud detection systems employing ML, deep learning (DL), and data mining techniques to determine whether a transaction is fraudulent or real.

ML has multiple branches, each of which can handle a variety of learning concerns. However, there are various framework types for ML. Random Forest (RF) is one example of an ML method for credit card fraud. The decision tree ensemble is represented by the random forest, and the RF technique is used by the majority of researchers [10], [11]. Researchers can employ a variety of machine learning techniques, including supervised and unsupervised learning. ML algorithms including Logistic Regression (LR), Artificial Neural Network (ANN), Decision Tree (DT), Support Vector Machine (SVM), and Naive Bayes (NB) [12] are widely utilized for fraud detection. Deep learning methods, such as the convolutional neural network (CNN) [13], deep belief networks (DBNs) [14] and generative adversarial network (GAN) [15] are also beneficial. A VAE is a variational autoencoder that employs frequent instruction loops to ensure that its hidden space contains relevant assets, allowing us to produce new data [16]. A VAE is created by adding modifications into the autoencoder [17], [18].

Traditional rule-based and ML-based fraud detection algorithms usually struggle to achieve high detection accuracy in real-time scenarios, especially when dealing with highly imbalanced datasets and quickly changing fraud patterns. Modern innovations in artificial intelligence (AI) and cloud computing have enabled the development of highly scalable and sophisticated fraud detection systems capable of processing enormous amounts of transactions instantaneously [19], [20]. However, the majority of existing solutions are built on centralized, single-model systems with limited adaptability and decision-making independence. To address these constraints, this study introduces a cloud-based Agentic AI system for real-time CCF detection based on DL framework. By merging Agentic AI, cloud computing, and deep temporal learning techniques, this research aims to create a next-generation fraud detection system that improves not only detection accuracy but also interpretability, flexibility, and automatic risk mitigation. The major contribution of this research is defined below:

- To enable autonomous and intelligent risk management, A cloud-based agentic AI system is developed credit card fraud detection by integrating multi-agent collaboration analysis.
- For real time fraud detection, the proposed framework integrates Data Quality, Behavioral Analysis, Verification, Risk Analysis, and Reporting agents.
- A deep temporal fraud detection model is developed with fully dynamic convolutional and shuffle attention mechanism to extract most informative attributes with sequential patterns for fraud detection.
- The system generates a fraud risk score for each transaction and takes relevant measures to mitigate it including notifications, transaction blocking, or escalation to human analysts. In real-world deployment scenarios, this eliminates manual intervention while increasing response speed.

The research organisation of the work is stated as: The recent survey based on credit card fraud detection with its merits and demerits are elaborated in section 2. The methodology with cloud-based fraud detection is described under section 3. Then, the performance evaluation part is discussed in section 4 and finally the research paper is ends with conclusion and future scope.

II. RELATED WORK

The progress of ML and DL techniques has had a significant impact on credit card fraud detection. Recent research has focused on increasing detection accuracy in highly imbalanced data sets, reducing false positives, and enabling real-time fraud detection. Traditional ML models are being replaced or improved by DL architectures such as CNNs, RNNs, hybrid multi-stage systems, and uncertainty-aware frameworks. Therefore, the recent publications for CCF detection are depicted in Table 1.

Table 1: Literature survey on CCF detection.

Reference	Main Contribution	Merits	Demerits / Limitations
Chen & Lai (2021) [21]	Proposed a Deep CNN for automatic feature learning and fraud detection with alert generation.	High detection accuracy; automatic extraction of spatial features; reduced reliance on manual feature engineering.	CNNs are computationally intensive; limited temporal modeling of transactions; lacks real-time deployment analysis.
Asha & Suresh Kumar (2021) [22]	Applied Artificial Neural Networks (ANNs) for binary fraud classification using transactional data.	Simple architecture; easy to implement; improved accuracy over traditional ML models.	Limited ability to capture temporal transaction dependencies; performance sensitive to class imbalance.
Zioviris et al. (2022) [23]	Introduced a deep learning multi-stage framework combining multiple classifiers for fraud detection.	Improved robustness; better generalization through stage-wise learning; reduced false positives.	Increased system complexity; higher computational overhead; not suitable for edge or low-latency environments.
Habibpour et al. (2023) [24]	Developed an uncertainty-aware deep learning model to quantify prediction confidence in fraud detection.	Improves trustworthiness of predictions; reduces risky misclassifications; supports risk-aware decision making.	Requires complex probabilistic modeling; higher training cost; interpretability is still limited.
Sulaiman et al. (2024) [25]	Proposed improved deep learning architectures for enhanced fraud detection under imbalanced datasets.	Better performance on skewed datasets; improved recall for minority (fraud) class.	Heavily dependent on dataset balancing strategies; lacks adaptive or autonomous learning capability.
Siam et al. (2025) [26]	Designed a hybrid feature selection framework integrated with machine learning classifiers.	Reduces dimensionality; enhances classification speed and accuracy; avoids irrelevant features.	Feature selection is static; cannot adapt dynamically to evolving fraud patterns.
Bonde & Bichanga (2025) [27]	Introduced an ensemble deep learning approach using SMOTE-ENN hybrid resampling.	Strong performance under extreme class imbalance; reduced noise and overfitting; stable predictions.	Synthetic data dependency may introduce bias; limited real-time applicability due to preprocessing overhead.

Source: Author, (2026).

III. PROPOSED METHEDOLGY

The proposed frmework incldes four main stages namely data pipeline, agnetic transaction analysis, deep temporal fraud detetcion and final decision and actions and it is illustarted in Figure 1. The first stage is stated as preprocesisng and it is accomplished to ensure high quality input for the upcoming stages. Agentic Transaction Analysis evaluates each transaction autonomously using a coordinated group of AI agents. The thids stage is stated as the proposed DL framework for fraud detetcion with advanced mechanims. And the final stage is the risk assessment with rapid decisions such as sending notifications, requesting further verification, or denying high-risk transactions respectively.

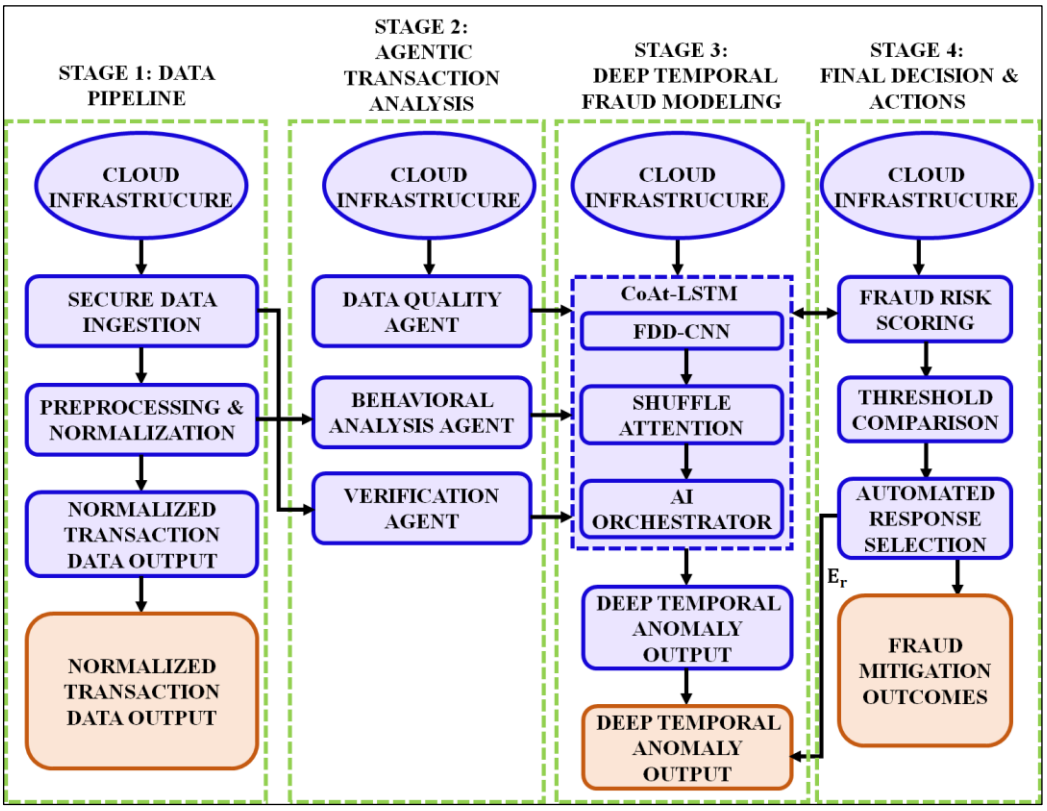

Figure 1: Overall proposed framework.
Source: Author, (2026).

III.1 STAGE 1: CLOUD-BASED DATA INGESTION AND PREPROCESSING

The CCF Detection dataset is securely ingested and pre-processed in the first step of the proposed methodology. To guarantee adherence to security and privacy regulations, the data is stored in an encrypted storage layer upon access into the cloud environment. The dataset integrity is then confirmed by a preliminary data-quality assessment, which includes evaluating feature ranges, looking for missing values, and verifying the statistical consistency of the principal component analysis (PCA)-derived features V_1–V_{28}. The dataset goes through a standardized preprocessing method after the initial verification. The approach uses normalization to provide consistent feature contribution during model training because the original features show varied scales. Normalization of standard scores is carried out using below equation.

$$x' = \frac{x-\mu}{\sigma} \tag{1}$$

Here, the original feature value is stated as x and the mean and standard deviation is termed as μ and σ respectively. Class-weighting is used in the preprocessing stage to reduce the disparity between fraudulent and valid transactions. Each class weight w_i is specified as:

$$w_i = \frac{1}{\text{freq}(y_i)} \tag{2}$$

Lastly, to preserve the initial fraud-to-non-fraud ratio, the dataset is divided into training and testing subsets using stratified sampling:

$$(X_{\text{train}}, X_{\text{test}}, y_{\text{train}}, y_{\text{test}}) = \text{StratifiedSplit}(X, y) \tag{3}$$

III.2 STAGE 2: MULTI-AGENT TRANSACTION ASSESSMENT

Each transaction is sent to a coordinated group of specialized AI agents that have been deployed in the cloud after preprocessing. These agents work in a sequential manner to make sure that each transaction is carefully assessed before being processed further.

The initial assessment is carried out by the Data Quality Agent, who confirms that the transaction has no missing or corrupted data and that the standardized input features fit within statistically valid ranges. This phase stops subsequent components from processing

noisy or corrupted records. The agent confirms that each feature x_i' falls within a reasonable range around its mean, usually within $x_i' \in [-3\sigma_i, +3\sigma_i]$ respectively. Transactions that exceed these thresholds are marked for possible anomalies or faults in preprocessing.

The Behavioral Analysis Agent assesses the spending characteristics of the transaction after data integrity has been verified. This agent is primarily concerned with identifying anomalous transaction quantities, sudden spending patterns, and departures from typical temporal patterns. The difference between the current transaction amount A_t and the mean historical amount μ_A is analyzed for a particular user using $\Delta A = \mid A_t - \mu_A \mid$ and temporal irregularity is evaluated using $\Delta t = t_{\text{current}} - t_{\text{previous}}$. Abnormally low Δt values could be a sign of fraudulent activity-related quick transaction bursts.

In this phase, the Verification Agent performs the last evaluation by contrasting the transaction with the user's long-term behavioral profile that has been preserved in the cloud. The degree to which the current transaction representation x_t and the user's behavioral profile P_u are similar is determined using a distance-based score $S = \| x_t - P_u \|$,where higher values signify less consistency with the user's known behavior. High-risk transactions are flagged and sent to the orchestrator for more in-depth contextual analysis if they don't pass this verification stage.

Before each transaction is sent to the higher-level orchestration and deep-learning stages of the framework, it is filtered, analyzed, and validated by these three agents working together to provide an organized, multi-perspective evaluation pipeline. This increases accuracy and robustness by ensuring that only transactions that have been thoroughly evaluated and contextually interpreted move on to the risk modeling stage.

III.3 STAGE 3: DEEP TEMPORAL FRAUD MODELING USING A COAT-LSTM ARCHITECTURE

Transactional input data is first processed by a Full-Dimensional Dynamic Convolutional Network (FDD-CN) in the suggested credit-card fraud detection system. This network adaptively learns spatial feature dependencies over several kernel dimensions. After being dynamically retrieved, the spatial characteristics G_t^s are subjected to a Shuffle Attention technique that suppresses noisy or redundant information while emphasizing the most essential channel and spatial feature representations. The attention-enhanced characteristics are then fed into a stacked Conv-LSTM network, which makes it possible to model the sequential transaction dynamics and temporal behavior that are commonly associated with fraudulent activity. An attention layer is incorporated to assess the relative significance of each time step in a transaction sequence in order to further improve discriminative capabilities. This layer allows the model to adaptively emphasize high-risk transaction states by computing attention weights β_t using a Tanh-based scoring function and a normalization procedure. The final attended representation H_t^a is a reliable real-time decision making and it is produced by the weighted aggregate of Conv-LSTM hidden outputs. The Deep Temporal Fraud Modeling is illustrated in Figure 2.

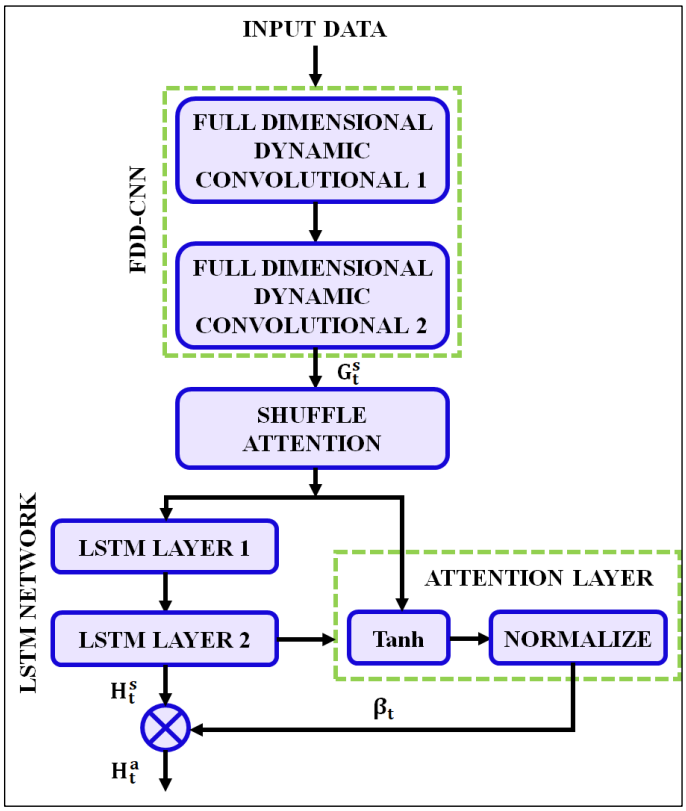

Figure 2: DL based fraud detection framework.
Source: Author, (2026).

III.3.1 Full-Dimensional Dynamic Convolutional Network (Fdd-Cn)

Single static convolution kernel is usually used by each convolutional layer in traditional convolutional neural network topologies. These kernels are fixed and applied consistently to all input samples after they are learned during training. However, in situations like real-time credit card transaction streams, this static behavior restricts the network's capacity to adjust dynamically changing patterns. This

research incorporates a Full-Dimensional Dynamic Convolution (FDDC) module to get over this restriction. This module adaptively modifies convolutional parameters according on input features, allowing for a more expressive and versatile feature-representation capability.

The main concept of FDDC is to use a set of many learnable convolution kernels in place of the static convolution kernel to dynamically compute the contribution weights of each kernel based on the properties of each input. Therefore, the dynamic convolution output is expressed below:

$$y = \sum_{i=1}^{m}(\alpha_{ci} \odot \theta_{ci} W_{ci}) * x \tag{4}$$

Here, W_{ci} specifies the convolutional kernel, m defines the number of dynamic kernels, θ_{ci} specifies the learnable kernel-parameter scaling, α_{ci} defines the attention weight respectively. In order to calculate these weights, the input is initially subjected to two Fully Connected (FC) layers with a ReLU activation, Global Average Pooling (GAP) to collect global contextual information, and Sigmoid activation. Therefore, the mathematical expression is given below:

$$\alpha_{ci} = \text{Sigmoid}(\text{FC}(\text{ReLU}(\text{FC}(\text{GAP}(x))))) \tag{5}$$

Operationally, the FDDC module uses GAP to first extract global contextual information from an input feature map x. After passing the resultant compact representation through a series of FC layers, ReLU and Sigmoid activations are used to produce a set of attention weights α. The module can determine how much each dynamic convolution kernel should contribute to the final output by using these attention scalars. The weighted kernels are then applied to the input in parallel after the resulting attention values are multiplied element-wise by the corresponding convolution kernels. Ultimately, an adaptively modulated feature map is created by aggregating the outputs of all dynamically weighted convolution kernels. The architecture of FDDC network is illustrated in Figure 3.

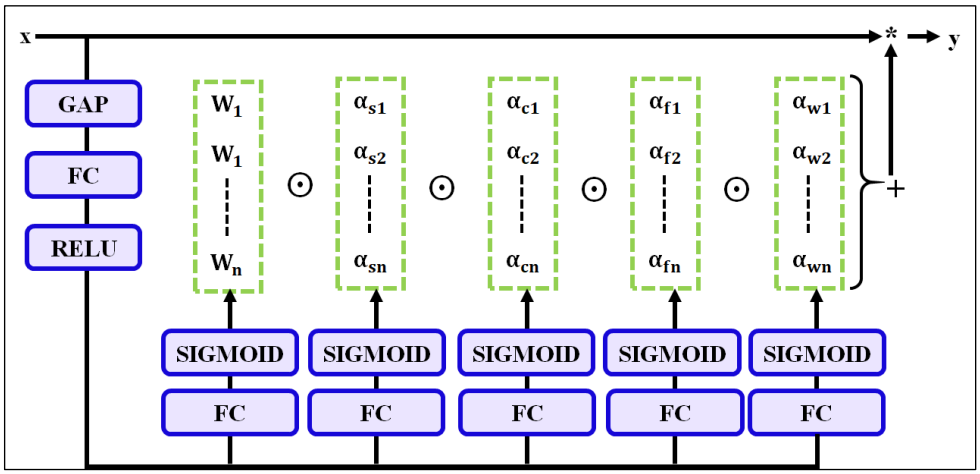

Figure 3: Structure of FDDC network .
Source: Author, (2026).

III.3.2 Shuffle Attention (SA) Mechanism

The Shuffle Attention (SA) technique combines spatial and channel attention while reducing computing complexity to improve feature discrimination. More fine-grained attention learning is made possible by SA, which splits the input feature tensor into smaller grouped subsets rather than analysing the entire feature map holistically. An input feature map $X \in \mathbb{R}^{C\times W\times H}$ is first divided into G groups along the channel dimension, represented as $X = \{X_1, X_2, \ldots, X_G\}$with each group $X_i \in \mathbb{R}^{\frac{C}{G}\times W\times H}$. As seen in Figure 4, each group is then further split into two branches, $X_{i1}$and X_{i2},, which stand for channel-attention processing and spatial-attention processing, respectively. Channel Attention is used in the first branch to highlight the significance of informational channels. The spatial dimension of X_{i1},is compressed using a GAP operation to create channel attention weights. This results in a one-dimensional descriptor $F_{GAP}(X_{i1})$. The channel-attention coefficients X'_{i1} are then obtained by feeding this descriptor into a fully linked transformation and Sigmoid activation.

$$S = F_{GAP}(X_{i1}) = \frac{1}{W\times H}\sum_{j=1}^{W}.\sum_{k=1}^{H}.X_{i1}(j,k) \tag{6}$$

$$X'_{i1} = \sigma(W_1 S + b_1) \tag{7}$$

The trainable parameters are stated as W_1 and b_1. Spatial Attention is used in the second branch to identify local structural signals and spatial correlations in the feature map. Spatial attention stresses pixel-level interactions, in comparison with the channel attention branch. To create spatial-attention maps, Group Normalization (GN) is applied over X_{i2}, followed by a fully connected layer and Sigmoid activation:

$$X'_{i2} = \sigma(W_2 \cdot GN(X_{i2}) + b_2) \odot X_{i2} \tag{8}$$

An enhanced feature group $X_i' = [X_{i1}', X_{i2}']$. is created by concatenating the two processed branches X_{i1}' and X_{i2}'. Following the independent processing of each group, the outputs are combined and a channel shuffle operation is carried out. This shuffling step prevents feature isolation and facilitates efficient information sharing across groups by redistributing learnt dependencies across channels. The architecture of SA mechanism is illustrated in Figure 5.

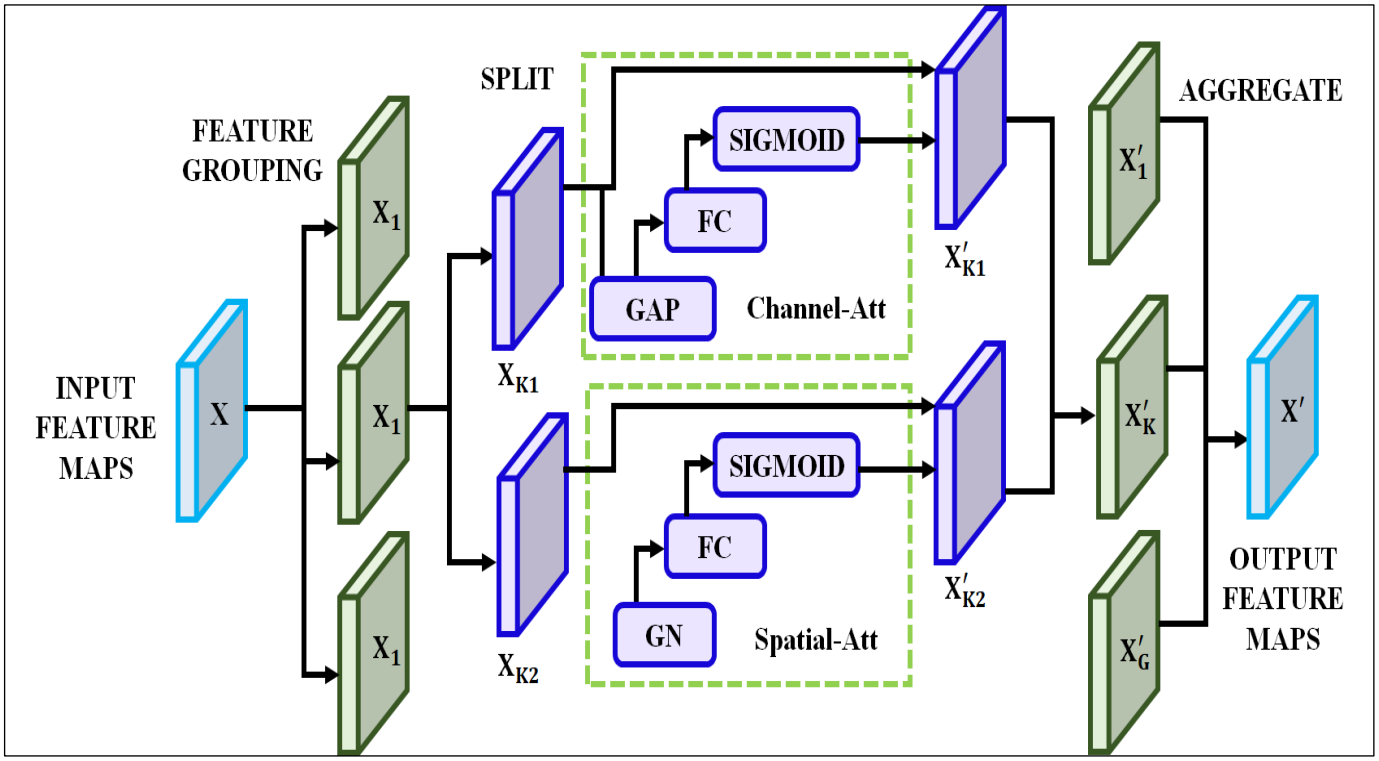

Figure 4: Shuffle attention mechanism.
Source: Author, (2026).

III.3.3 Attention Layers with Conv-LSTM Networks

The Conv-LSTM framework incorporates the attention layer to improve the capacity of the model. In order to allow the network to emphasize the most relevant temporal aspects and suppress unnecessary or redundant information, an attention mechanism is implemented to selectively weight the hidden states generated by Conv-LSTM. A weighted sum of the hidden states is used to determine the final attention-enhanced hidden representation $H_t{}'$ and its expression is given below:

$$H_t{}' = \sum_{s=1}^{n+1} . \beta_s H_{t-(s-1)}^f \tag{9}$$

Here, $n+1$ specifies the sequence length, β_s defines the attention weights and its mathematical formulation is given below:

$$\beta_s = \frac{\exp(S_s)}{\sum_{i=1}^{n+1} \exp(S_i)} \tag{10}$$

The contribution of both temporal information from the Conv-LSTM hidden states (H_t^f) and spatial feature information from CNN layers $(G_t)$is evaluated to determine the important score S_s and it is defined below:

$$S_s = V_s^T \tanh(W_s G_t + W_s^f H_t^f) \tag{11}$$

where V_s, W_s,and W_s^f are learnable weight parameters. The network gains improved interpretability, resilience, and efficiency when handling complex temporal sequences like fraud detection, or anomaly analysis by combining Conv-LSTM with attention. The structure attention layer with network model is illustrated in Figure 5.

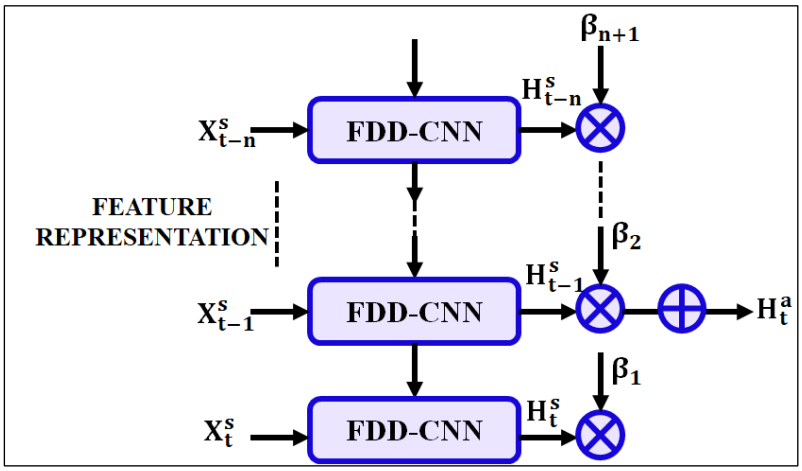

Figure 5: Attention layer with network model.
Source: Author, (2026).

III .3.4 Lstm Final Layer Workflow Description

The LSTM layer serves as the main decision-making component and it receives the refined feature representations generated by the FDD-CNN layer and the attention-enhanced Conv-LSTM processing. By examining the sequential evolution of transaction behavior, this last LSTM layer discovers the differences between normal and fraudulent spending patterns. The LSTM layer creates an output and that captures the degree of risk connected to each transaction by keeping crucial temporal information and eliminating unnecessary variations. The prediction module receives this final temporal embedding and uses it to determine a fraud probability score that determines whether the transaction is authentic or fraudulent. As a result, the LSTM layer functions as the final analytical step that integrates all acquired temporal, channel, and geographical insights into an appropriate fraud detection determination.

III.4 STAGE 4: FINAL DECISION SYNTHESIS, RISK SCORING, AND AUTOMATED ACTION EXECUTION

The output generated from the previous stages are integrated in the final stage of the proposed framework which include preprocessing stage, transaction stage and the deep temporal fraud detection stage respectively. This level serves as the decision-making layer, assuring that the statistical knowledge generated by the multi-agent architecture is translated into real-time defensive measures.

III.4.1 Risk Score Fusion and Decision Synthesis

All agent-level outputs are gathered via a weighted decision function regulated by the AI Orchestrator. Therefore, the mathematical expression of integrated transaction-level risk index R_t is defined below:

$$R_t = \sigma(w_1 S_{\text{coat}} + w_2 S_{\text{beh}} + w_3(1 - S_{\text{ver}}) + w_4(1 - S_{\text{dq}})) \tag{12}$$

Here, the anomaly probability output from the CoAt-LSTM model is stated as S_{coat}, behavioral deviation score produced by the Behavioral Analysis Agent is termed as S_{beh}, the score from the Verification Agent and Data Quality Agent is termed as S_{ver} and S_{dq} respectively. w_1, w_2, w_3, w_4 are orchestrator-learned fusion weights.

III.4.2 Threshold-Based Classification and Interpretability Layer

After generating the risk score R_t, the comparison process is performed with dynamic decision threshold τ_t, which is constantly updated using cloud-side feedback loops. A transaction is fraudulent if:

$$\hat{y}_t = \begin{cases} 1, & R_t \geq \tau_t, \\ 0, & R_t < \tau_t. \end{cases} \tag{13}$$

By using Bayesian calibration or reinforcement signals arising from false-positive/false-negative results the threshold τ_t is regularly improved.

III.4.3 Automated Mitigation and Action Execution

Following classification, the system executes one of many automated responses via the Reporting Agent. These activities are rule-based but informed by the estimated risk magnitude:

- **Low-risk ($R_t < \tau_t$)**

→ The transaction has been approved and recorded in the ledger for future learning.

- **Moderate-risk ($\tau_t \leq R_t < \tau_t + \delta$)**
  → Transaction has been marked for secondary authentication (OTP verification, user confirmation, or device fingerprinting tests).
- **High-risk ($R_t \geq \tau_t + \delta$)**

→ The transaction is automatically blocked, and an incident report is sent to the banking back office.

The term δ refers to an empirically calculated high-risk escalation margin. The Reporting Agent creates a structured cloud report based on decision metadata such as risk score, anomaly patterns, temporal indices, and supporting evidence from other agents. This ensures immediate fraud mitigation while keeping full documentation for forensic examination.

IV. RESULT AND DISCUSSION

This section presents and analyzes the experimental results obtained by utilizing the proposed fraud detection system. The effectiveness of the model is evaluated using normal classification parameters, and the results are analyzed to determine the efficacy of the proposed method in detecting fraudulent transactions. A comparison with baseline models is additionally provided to demonstrate the benefits achieved by the recommended approach.

IV.1 DATASET DESCRIPTION

The experiments described in this article were carried out on the Kaggle Credit Card Fraud Detection dataset, which contains 284,807 transaction records by employing 31 features. The above characteristics comprise 28 anonymized numerical variables generated from principle component analysis (PCA), as well as the Time and Amount attributes, with Class representing fraudulent transactions and 0 indicating legitimate ones. The dataset is portioned in to train-test split ratio of 80 and 20% respectively to perform network modelling.

IV.2 CLASS IMBALANCE IN THE CCF DATASET

The class distribution of CCF Detection is shown in Figure 6, which clearly demonstrates the significant deviation between genuine and fraudulent cases. The minority class (Class 1, fraudulent transactions) makes up just around 0.2% of the total samples, whereas the majority class (Class 0, non-fraudulent transactions) makes up about 99.8%. This stark disparity is an aspect of a realistic fraud detection pattern in which fraudulent events are extremely important but inherently uncommon. Therefore, the suggested agentic and deep learning framework's need for specialized learning strategies, reliable evaluation measures, and sophisticated modeling techniques is justified by this imbalance.

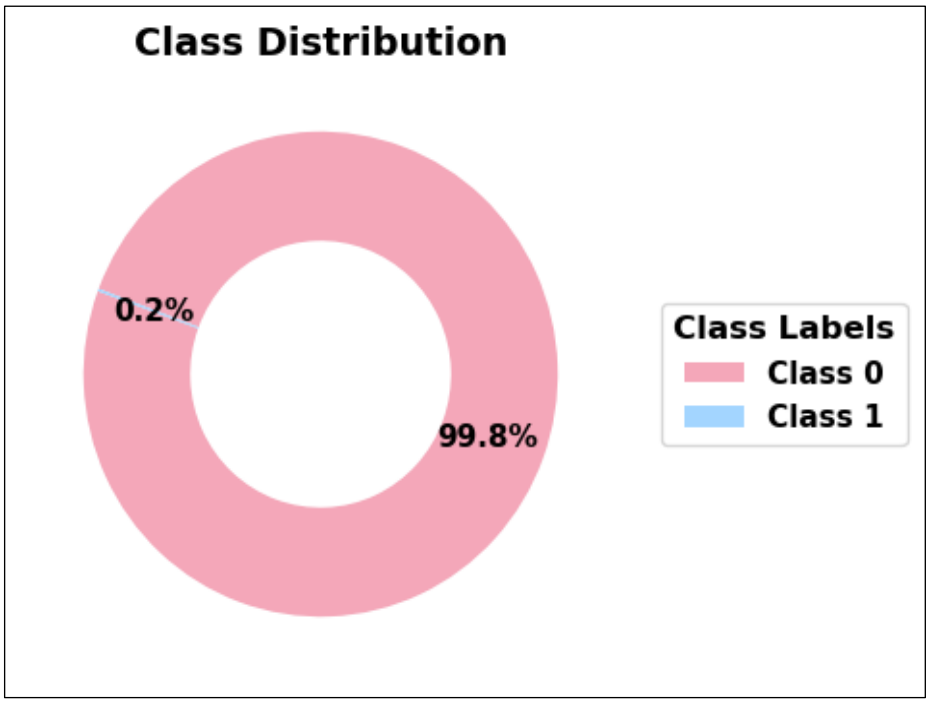

Figure 6: Class Distribution of Legitimate and Fraudulent Transactions.
Source: Author, (2026).

IV.3 RIGHT-SKEWED NATURE OF MONETARY TRANSACTION VALUES

The distribution of transaction amounts across all recorded operations in the dataset is shown in Figure 7. The distribution shown in the histogram has a significant bias to the right, with the great majority of transactions happening at relatively modest monetary values and a small number exhibiting exceptionally high sums. This long-tail tendency is typical of actual financial transaction data, where high-value transactions are rare and ordinary purchases are more prevalent. In order to avoid impact on the learning process, preprocessing must properly scale and normalize transaction values due to their broad range. Furthermore, this distribution is especially crucial for detecting fraud since abnormally high or unusual transaction amounts frequently contain powerful discriminatory information for spotting fraudulent activity.

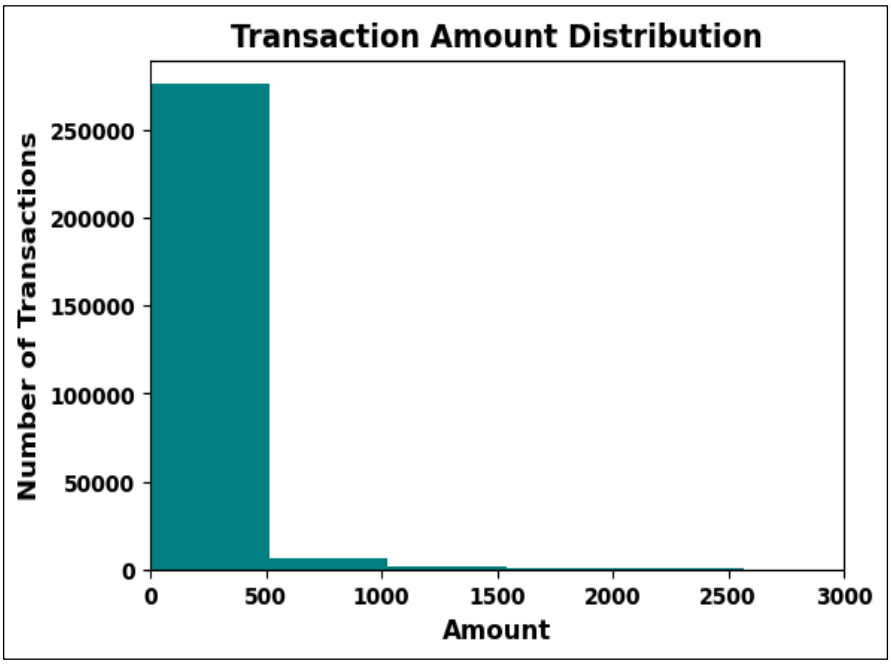

Figure 7: Statistical Distribution of Transaction Amounts.
Source: Author, (2026).

IV.4 MARGINAL DISTRIBUTIONS OF PCA FEATURES V11 AND V19

The probability distribution of V11 and V19 across frequency over the complete transaction set is distributed in Figure 8. The visualization of distribution reveals a gaussian shape structure which is focussed at zero. Feature V19 seems to be more heavily centralized with smaller dispersion and feature V11 has a moderately right-skewed distribution with significant range. Therefore, this indicates that it does not requires any additional normalization process. In terms of fraud detection, such symmetric distributions imply that discriminative power is derived not just from extreme marginal values, but also from joint feature interactions and temporal patterns, necessitating the employment of deep network model in later phases.

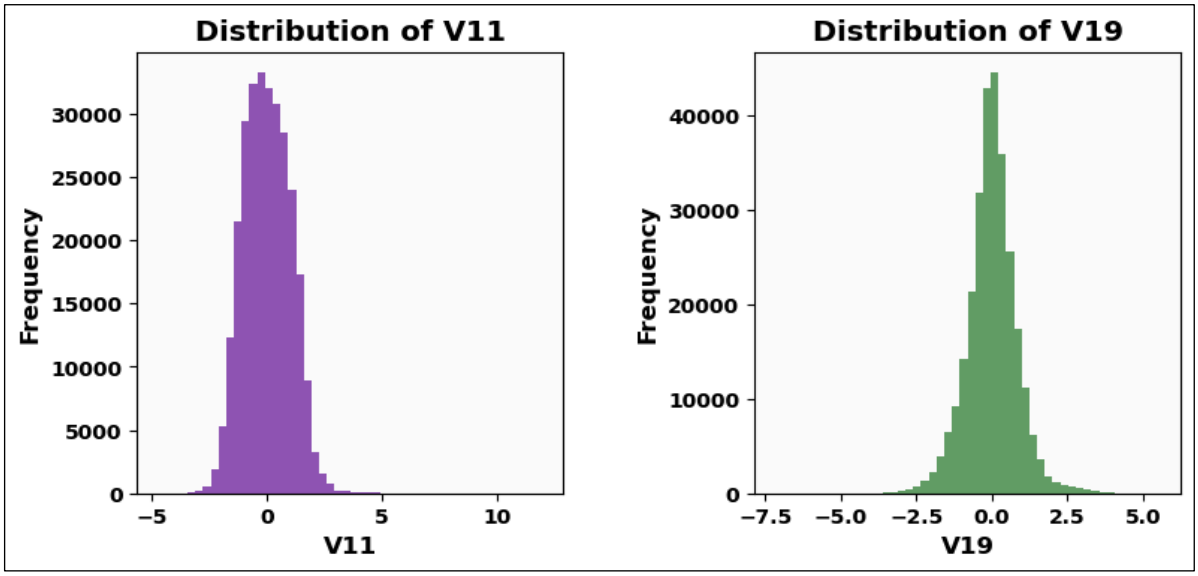

Figure 8: Probability distribution of V11 and V19.
Source: Author, (2026).

IV.5 JOINT DISTRIBUTION OF PCA FEATURES V3 AND V4

Figure 9 shows the superimposed histograms of V3 and V4 of two PCA-derived elements that are frequently utilized to be extremely valuable for fraud detection. Both features have large numerical ranges with big curves and it is in bell shaped distributions with their point centers near to zero. The V3 feature has a little broader spread toward the negative, implying an abundance of certain high or low values, whereas V4 looks to be more densely packed around the mean. The high overlap between the V3 and V4 distributions indicates that these features fail to distinguish between fraudulent and genuine transactions on their own. This image indicates that the data is well-normalized and appropriate for use in deep learning models like the CoAt-LSTM utilized in this work.

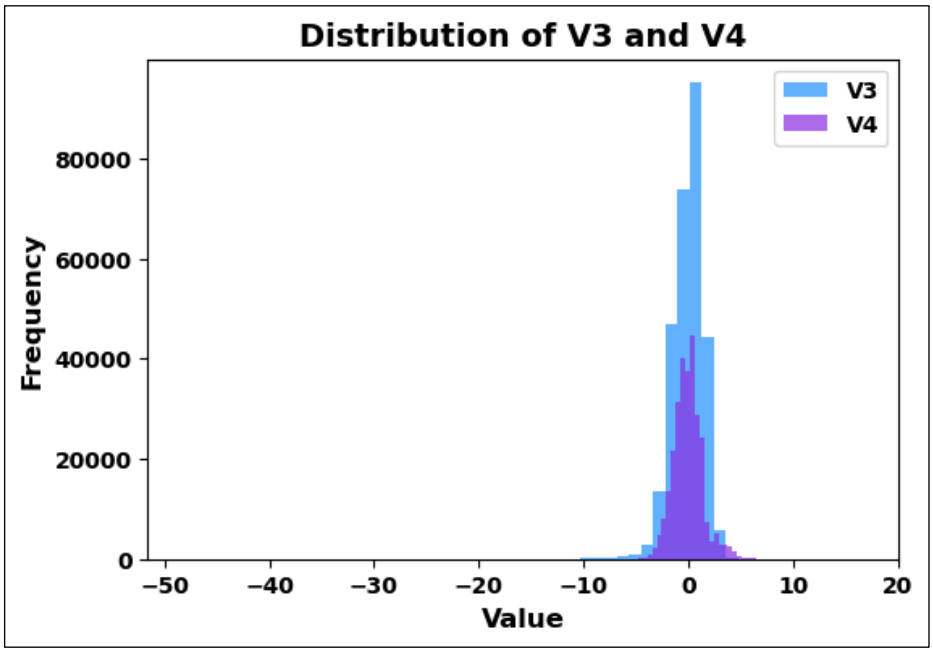

Figure 9: Probability distribution of V3 and V4.
Source: Author, (2026).

IV.6 FEATURE DISTRIBUTION ACROSS FRAUD AND NON-FRAUD TRANSACTIONS

The probability density of the V12 feature for genuine (Class 0) and fraudulent (Class 1) transactions is shown in the Figure 10. For non-fraud transactions the curve focuses around a short band near zero and it indicates that the majority of legitimate transactions have extremely similar V12 values. In comparison, the fraud distribution is much more dispersed and biased toward negative values and it reflects a greater variance in fraudulent activity. The noticeable difference between the two curves implies that V12 is a highly discriminative as illegal transactions typically follow a distinct pattern from legitimate ones.

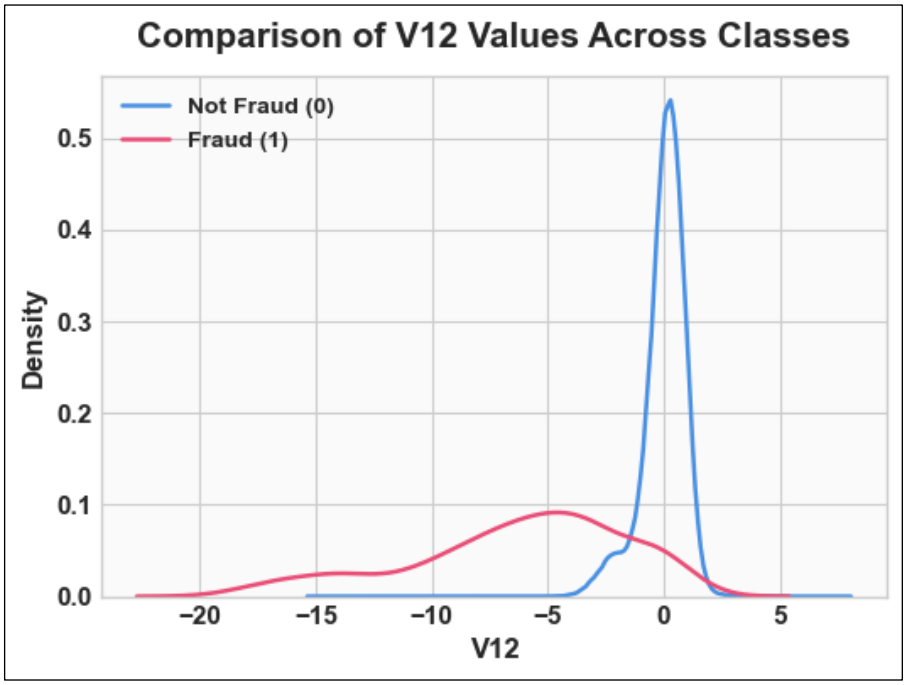

Figure 10: Class-wise Density Comparison of Feature V12.
Source: Author, (2026).

IV.7 FEATURE DENSITY WITH MEAN AND MEDIAN

The overall feature density distribution of V22 across its mean and median values is illustrated in Figure 11. Figure displays that the curve has a point which is near to zero as well as both the mean and median are extremely near to zero. This shows that the feature was effectively well normalized and fails to have a significant imbalance. This curve indicates the PCA-transformed features and it confirm that V22 is statistically robust and suitable for immediate usage in training the model without further scaling or modification.

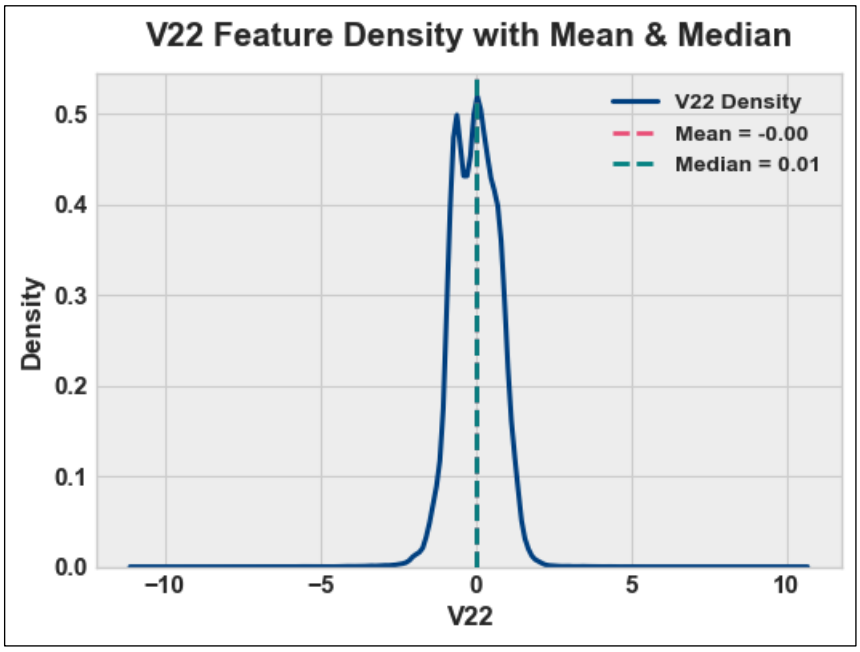

Figure 11: Density Profile of Feature V22 with Mean and Median.
Source: Author, (2026).

IV.8 CORRELATION HEATMAP OF PCA FEATURES

Figure 12 depicts the Pearson correlation matrix for the PCA-transformed elements and it is seen that the vast majority of off-diagonal components possess values almost identical to zero. This confirms due to the orthogonality established by the PCA transformation process. Therefore. this degree of statistical independence holds significance for deep learning architectures because it reduces convergence and maintains gradient propagation during the training stage. This lack of linear correlations signifies that fraudulent activities are reflected in temporal dependencies which supports the inclusion of convolutional, attention, and LSTM layers in the third stage of the proposed framework.

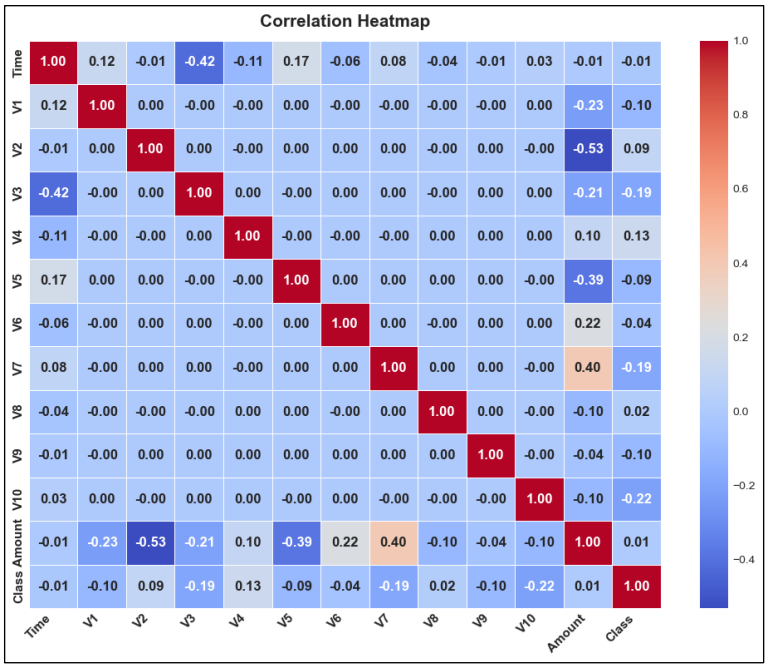

Figure 12: Correlation heatmap.
Source: Author, (2026).

IV.9 LEARNING STABILITY AND CONVERGENCE BEHAVIOR OF THE PROPOSED COAT-LSTM MODEL

The training and validation of both the accuracy and loss curve of the proposed model is illustrated in Figure 13. The accuracy and loss are examined by altering the epoch size to 20. The figure shows that the proposed model's training accuracy remains stable at above 95% throughout all epochs, indicating strong learning consistency. The validation accuracy nearly matches the training curve, with just minor differences, showing strong generalization and little overfitting.

The proposed model's loss curve, illustrated on the right side, falls gradually from around 0.27 to nearly 0.12, confirming effective optimization. At the intermediate stage, the validation loss decreases with minor fluctuations for fraud detection. Overall, the precise alignment of the training and validation curves demonstrates that the COAt-LSTM model provides both stable convergence and robust temporal fraud detection.

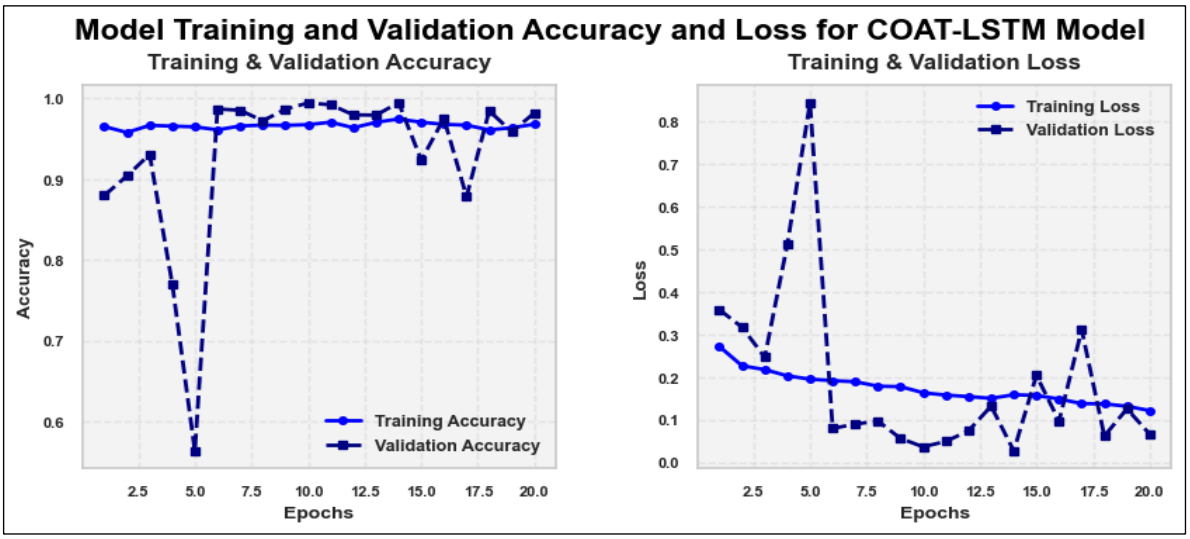

Figure 13: Training and Validation Performance of the COAt-LSTM Model Across Epochs.
Source: Author, (2026).

IV.10 COMPARATIVE LEARNING CHARACTERISTICS OF THE BASELINE GRU MODEL

The learning characteristics of the existing baseline GRU model in terms of accuracy and loss is shown in Figure 14. In the course of the first few epochs, the training accuracy rapidly increases and then remains steady at 96-97%. The ability of GRU model to record extremely detailed patterns or manage excessively long sequences may be limited. The training loss decreases dramatically from an initial high value of 0.78 to less than 0.1, showing fast convergence. The training and validation accuracy curves show that the proposed CoAt-LSTM model converges faster and consistently achieves higher accuracy than the baseline GRU model. Furthermore, the suggested model's loss curves decline more smoothly and remain much lower across epochs, signifying greater learning stability.

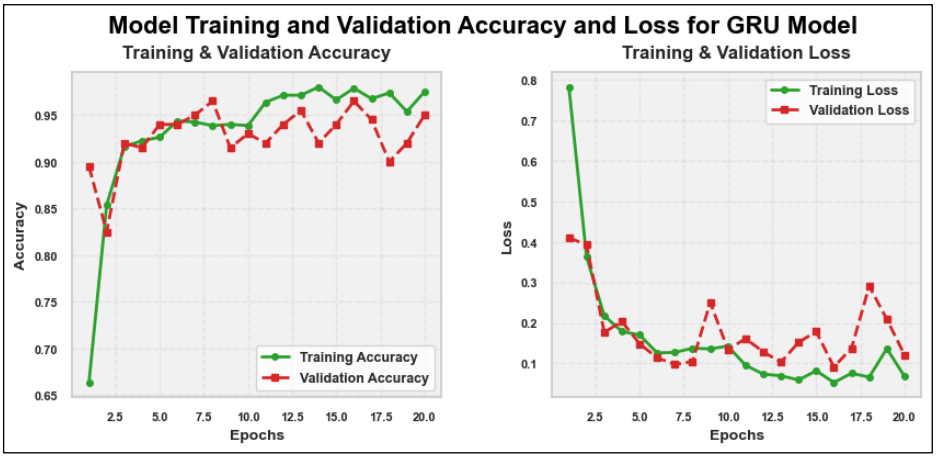

Figure 14: Training and Validation Performance of the GRU Model Across Epochs.
Source: Author, (2026).

IV.11 COMPARATIVE CLASSIFICATION PERFORMANCE USING CONFUSION MATRICES

The confusion matrix analysis of the proposed model and the existing GRU model is illustrated in Figure 15 and it compares the predicted and actual values. From figure it is seen that the underclass 0, proposed model predicts 56856 as class 0 and only 8 are classified incorrectly as class 1. In addition to this for class 1, the proposed model predicts 74 as class 1 and only 24 are misclassified as class 0 respectively. In addition to this for an existing GRU model, for class 0, 54380 samples are correctly classified as class 0 and 2484 samples are misclassified as class 1. For class 1, 87 samples are correctly classified and 11 samples are mis classified as class 0 respectively. Overall, the COAT-LSTM model exhibits an optimum equilibrium between fraud detection accuracy and false alarm control, making it more acceptable for practical applications.

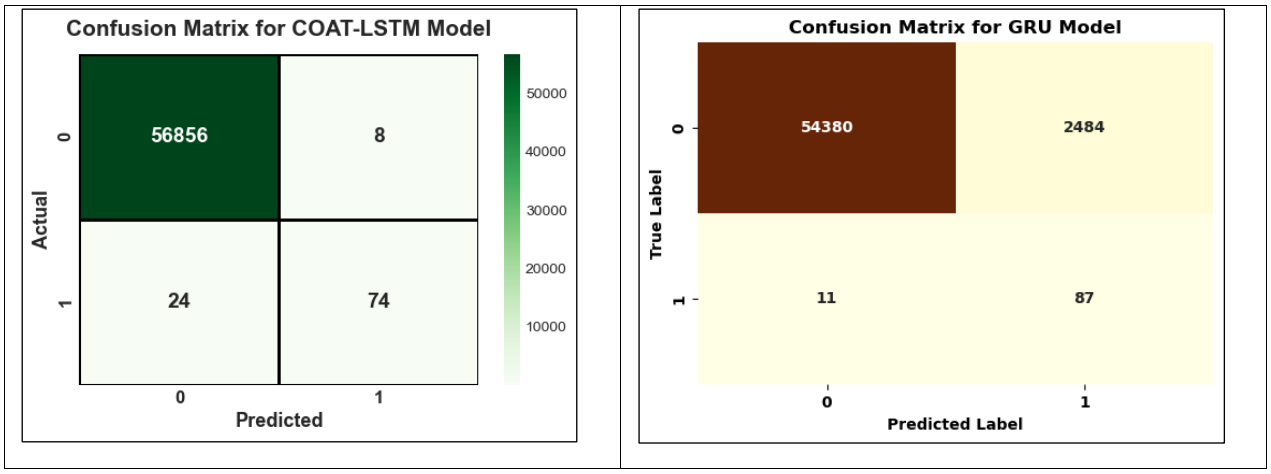

Figure 15: Confusion Matrix Analysis of COAT-LSTM and GRU Models.
Source: Author, (2026).

IV.12 DISCRIMINATIVE CAPABILITY EVALUATION USING ROC CURVES

The ROC (Receiver Operating Characteristic) curve compares the true positive rate against the false positive rate at various decision thresholds to demonstrate how well the model distinguishes between fraudulent and legitimate transactions. A curve that is closer to the top-left corner suggests superior classification performance since it produces a high fraud detection rate while minimizing false alarms. In the present scenario, the high Area Under the Curve (AUC) value indicates that the proposed model has excellent discriminative ability and can reliably distinguish between fraudulent and non-fraudulent transactions. Form figure 16 it is evident that the proposed model attains an AUC of 97% whereas the existing GRU model reaches and AUC of 95.02% recpectively. The boosted AUC generated by COAT-LSTM demonstrates that combining fully dynamic convolutional model with shuffle attention mechanism enhances temporal learning by increasing the sensitivity of the model while retaining reliable generalization. These findings confirm the benefit of the suggested deep temporal modelling framework in high-risk financial transaction circumstances.

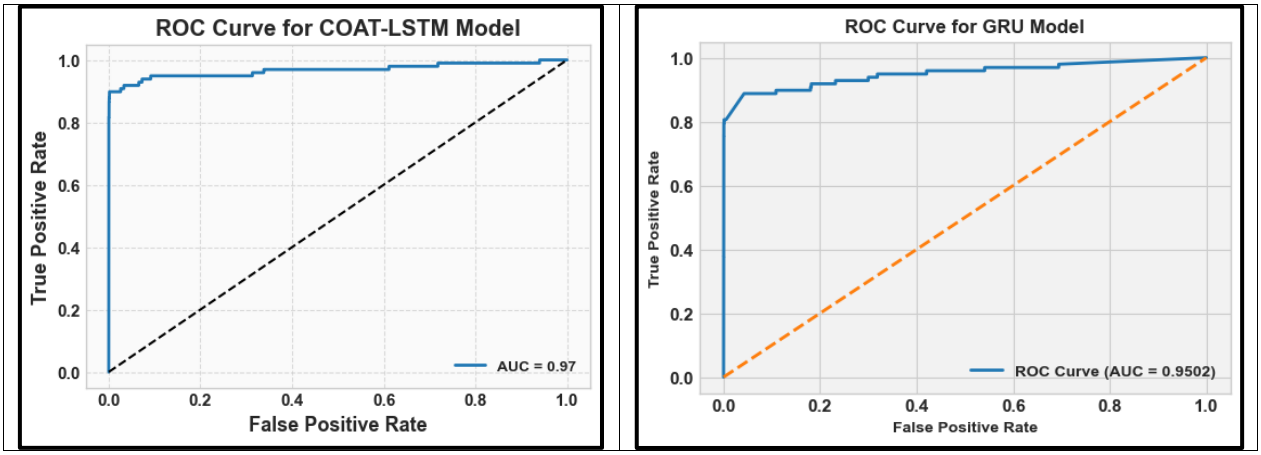

Figure 16: ROC Curve Comparison for COAT-LSTM and GRU Models.
Source: Author, (2026).

IV.13 FINAL TRANSACTION ASSESSMENT AND RISK INTERPRETATION

Figure 17 depicts the proposed agentic fraud detection framework's ultimate decision after completing all analytical phases, including verification, behavioral analysis, and deep temporal modelling. The system first displays the Forgery Detection result as "No Forgery," signifying that there was no actual manipulation or forgery has been identified in the transaction characteristics. The Verification Status signifies that the transaction has been verified effectively against an external database, assuring data authenticity and consistency. The CoAt-LSTM-based fraud detection model classified the transaction as "Legit," according to the AI Report. However, the Risk Assessment also identifies the transaction as "High Risk," indicating that while the transaction is classified as valid, it exhibits certain behavioral or temporal patterns and therefore require constant monitoring. The prediction probabilities demonstrate a very high confidence level for Class 0 (legitimate) and an extremely low likelihood for Class 1 (fraud), justifying the final predicted class of 0 (Legit). The Final Prediction column concisely highlights the outcome sent to the automatic response module. This result highlights how the proposed system relying purely on classification output and it incorporates verification signals and behavioral risk indicators to deliver a more accurate and effective fraud conclusion in real-world deployment. The comparative analysis of performance measures with respect to existing base line model is shown in Table 2.

```
==================== FINAL OUTPUT ====================
Forgery Detection     : No Forgery
Verification Status   : Verified with External DB
AI Report             : Legit
Risk Assessment       : High Risk
======================================================

Prediction Probabilities = [[9.9987781e-01 1.2218881e-04]]
Predicted Class = 0
Final Prediction: Legit
```

Figure 17: Evaluation of Fraud Decisions and Risk Indicators.
Source: Author, (2026).

Table 2: Performance analysis with existing GRU model.

Metrics	Proposed	Existing GRU
Accuracy	0.9821	0.9562
Precision	0.5395	0.5168
Recall	0.9350	0.9220
F1 score	0.5682	0.5214
Cohens kappa	0.1428	0.0621
Matthew's coefficient	0.2622	0.1685
Log loss	0.0665	0.4554

Source: Author, (2026).

IV.14 ABLATION STUDY

The ablation study clearly reveals the specific contribution of each component of the proposed CCF architecture and it is shown in Table 3. The full Agentic CoAt-LSTM model outperforms all assessment measures, demonstrating the efficacy of combining agentic intelligence and deep temporal modeling. When the agentic layer is eliminated, there is a considerable decline in accuracy, precision, and recall, demonstrating the necessity of autonomous transaction reasoning and cross-agent validation. Similarly, removing the attention mechanism limits the model's capacity to focus on crucial transaction patterns, resulting in worse fraud detection performance. The elimination of the convolutional layer and the LSTM have an effect on localized feature learning and temporal dependency modeling and this tends to classification capabilities. Overall, the ablation analysis shows that each architectural block makes a significant contribution to the proposed system with resilience and dependability.

Table 3: Ablation study.

Variant (Ablation)	Accuracy	Precision	Recall	F1 score
Proposed (CoAt-LSTM + Shuffle-Attention + Agentic pipeline)	0.9821	0.5395	0.9350	0.5682
Without Shuffle Attention (CoAt-LSTM w/o attention)	0.9750	0.4900	0.8900	0.6200
Without CNN (LSTM + Attention only)	0.9680	0.4500	0.8800	0.5900
Without LSTM (CNN + Attention only)	0.9700	0.4200	0.8200	0.5600
Without Agentic components (no Data-Quality / Verification agents)	0.9740	0.4800	0.8800	0.6200
Baseline GRU model (standard baseline)	0.9610	0.4500	0.9000	0.6000

Source: Author, (2026).

V. CONCLUSION AND FUTURE SCOPE

This paper described a unique cloud-based Agentic Artificial Intelligence architecture for real-time CCF detection. The proposed methodology included intelligent multi-agent collaboration, deep temporal learning via the CoAt-LSTM model, and an automated risk determination mechanism to improve detection accuracy and responsiveness. The system efficiently addressed the intrinsic issues of extreme class imbalance and anonymous features by combining behavioral analysis, dynamic convolution, attention mechanisms, and temporal dependency learning. Experimental results showed that the suggested model outperformed traditional machine learning methodologies and baseline deep learning models like GRU in terms of accuracy, precision, recall, F1-score, and ROC performance.

The automated risk assessment and decision module allowed for faster and more reliable mitigation activities without the need for manual intervention. Despite the high level of performance attained in this study, many opportunities for future research remain unexplored. To begin, the framework can be enhanced to accommodate real-time streaming transaction data for large-scale deployment via distributed cloud platforms such as Apache Kafka and Spark. Second, explainable AI (XAI) techniques can be used to improve fraud decision interpretability, which is crucial for regulatory compliance in financial systems. Third, the suggested agentic framework might be improved by incorporating reinforcement learning, which would enable agents to continuously adapt to newly developing fraud patterns.

VI. AUTHOR'S CONTRIBUTION

Conceptualization: Sushil Prabhu Prabhakaran.
Methodology: Sushil Prabhu Prabhakaran.
Investigation: Sushil Prabhu Prabhakaran.
Discussion of results: Sushil Prabhu Prabhakaran.
Writing – Original Draft: Sushil Prabhu Prabhakaran.
Writing – Review and Editing: Sushil Prabhu Prabhakaran.
Resources: Sushil Prabhu Prabhakaran.
Supervision: Sushil Prabhu Prabhakaran.
Approval of the final text: Sushil Prabhu Prabhakaran.

VII. REFERENCES

[1] R. Khalid, N. Owoh, O. Uthmani, M. Ashawa, J. Osamor and J. Adejoh, "Enhancing credit card fraud detection: an ensemble machine learning approach," Big Data and Cognitive Computing, vol. 8, no. 1, pp. 6, Jan. 2024.

[2] R. Agrawal, A. Khanna and S. Hamdare, "Analyzing and Rewarding Credit Card Spending Habits in India: a Machine Learning Approach," International Journal of Computational Intelligence Systems, vol. 18, no. 1, pp. 165, Jul. 2025.

[3] D.G. Beju and C.M. Făt, "Frauds in banking system: Frauds with cards and their associated services," In Economic and financial crime, sustainability and good governance, pp. 31-52. Cham: Springer International Publishing, Aug. 2023.

[4] E.O. Udeh, P. Amajuoyi, K.B. Adeusi and A.O. Scott, "The role of big data in detecting and preventing financial fraud in digital transactions," World Journal of Advanced Research and Reviews, vol. 22, no. 2, pp. 1746-1760, 2024.

[5] P. Vanini, S. Rossi, E. Zvizdic, and T. Domenig, "Online payment fraud: from anomaly detection to risk management," Financial Innovation, vol. 9, no. 1, pp. 66, Mar. 2023.

[6] Z.U. Mamudu, "Electronic Banking Payment System and Its Impact on the Nigerian Economy," Journal of Emerging Trends in Economics and Management Sciences, vol. 14, no. 3, pp. 121-139, 2023.

[7] P. Kaur, A. Sharma, J. K. Chahal, T. Sharma and V. K. Sharma, "Analysis on credit card fraud detection and prevention using data mining and machine learning techniques," In 2021 International Conference on Computational Intelligence and Computing Applications (ICCICA), pp. 1-4. IEEE, Nov. 2021.

[8] M. Z. Khan, S. A. Shaikh, M. A. Shaikh, K. K. Khatri, M. A. Rauf, A. Kalhoro and M. Adnan, "The performance analysis of machine learning algorithms for credit card fraud detection," iJOE, vol. 19, no. 03, pp. 83, 2022.

[9] Y. K. Saheed, U. A. Baba and M. A. Raji, "Big data analytics for credit card fraud detection using supervised machine learning models," In Big data analytics in the insurance market, pp. 31-56. Emerald Publishing Limited, Jul. 2022.

[10] T. H. Lin and J. R. Jiang, "Credit card fraud detection with autoencoder and probabilistic random forest," Mathematics, vol. 9, no. 21, pp. 2683, Oct. 2021.

[11] W. Li, C. S. Wu and S. M. Ruan, "CUS-RF-based credit card fraud detection with imbalanced data," Journal of Risk Analysis and Crisis Response, vol. 12, no. 3, Sep. 2022.

[12] F. Itoo, Meenakshi and S. Singh, "Comparison and analysis of logistic regression, Naïve Bayes and KNN machine learning algorithms for credit card fraud detection," International Journal of Information Technology, vol. 13, no. 4, pp. 1503-1511, Aug. 2021.

[13] J. Karthika and A. Senthilselvi, "Smart credit card fraud detection system based on dilated convolutional neural network with sampling technique," Multimedia Tools and Applications, vol. 82, no. 20, pp. 31691-31708, Aug. 2023.

[14] S. Deepika and S. Senthil. "Credit card fraud detection using moth-flame earth worm optimisation algorithm-based deep belief neural network," International Journal of Electronic Security and Digital Forensics, vol. 14, no. 1, pp. 53-75, 2022.

[15] N. T. Ali, S. J. Hasan, A. Ghandour and Z. S. Al-Hchimy, "Improving credit card fraud detection using machine learning and GAN technology," In BIO Web of Conferences, vol. 97, pp. 00076. EDP Sciences, 2024.

[16] Y. Ding, W. Kang, J. Feng, B. Peng and A. Yang, "Credit card fraud detection based on improved Variational Autoencoder Generative Adversarial Network," IEEE Access, vol. 11, pp. 83680-83691, Aug. 2023.

[17] S. Shi, W. Luo and G. Pau, "An attention-based balanced variational autoencoder method for credit card fraud detection," Applied Soft Computing, pp. 113190, Apr. 2025.

[18] F. Ouedraogo, C. Heuchenne, Q. T. Nguyen and H. Tran, "Data-driven approach for credit card fraud detection with autoencoder and one-class classification techniques," In IFIP International Conference on Advances in Production Management Systems, pp. 31-38. Cham: Springer International Publishing, Aug. 2021.

[19] H. Rehan, "Leveraging AI and cloud computing for Real-Time fraud detection in financial systems," Journal of Science & Technology, vol. 2, no. 5, pp. 127, 2021.

[20] I. B. Ramli, "Big Data and Artificial Intelligence to Develop Advanced Fraud Detection Systems for the Financial Sector," International Journal of Advanced Cybersecurity Systems, Technologies, and Applications, vol. 8, no. 12, pp. 31-44, Dec. 2024.

[21] J. I. Z. Chen and K. L. Lai, "Deep convolution neural network model for credit-card fraud detection and alert," Journal of Artificial Intelligence, vol. 3, no. 02, pp. 101-112, Jun. 2021.

[22] R. B. Asha and K. R. Suresh Kumar, "Credit card fraud detection using artificial neural network," Global Transitions Proceedings, vol. 2, no. 1, pp. 35-41, Jun. 2021.

[23] G. Zioviris, K. Kolomvatsos and G. Stamoulis, “Credit card fraud detection using a deep learning multistage model,” Journal of Supercomputing, vol. 78, no. 12, Aug. 2022.

[24] M. Habibpour, H. Gharoun, M. Mehdipour, A. Tajally, H. Asgharnezhad, A. Shamsi, A. Khosravi and S. Nahavandi, “Uncertainty-aware credit card fraud detection using deep learning,” Engineering Applications of Artificial Intelligence, vol. 123, pp. 106248, Aug. 2023.

[25] S. S. Sulaiman, I. Nadher and S. M. Hameed, “Credit Card Fraud Detection Using Improved Deep Learning Models,” Computers, Materials & Continua, vol. 78, no. 1, Jan. 2024.

[26] M. Siam, P. Bhowmik and M. P. Uddin, “Hybrid feature selection framework for enhanced credit card fraud detection using machine learning models,” PLoS One, vol. 20, no. 7, pp. e0326975, Jul. 2025.

[27] L. Bonde and A. K. Bichanga, “Improving Credit Card Fraud Detection with Ensemble Deep Learning-Based Models: A Hybrid Approach Using SMOTE-ENN,” Journal of Computing Theories and Applications, vol. 2, no. 3, pp. 383-394, Feb. 2025.