Indonesian Applied Research Computing and Informatics
Vol. 1; No. 2, December 2025 Page 22-36;
https://jurnal.tdinus.com/index.php/iarci
e-ISSN : 3110-8806

Unsupervised Credit Card Fraud Detection Using Autoencoder-Based Anomaly Detection on Highly Imbalanced Transaction Data

Mursalim[1*], Sutriawan[2], Nimas Ratna Sari[1], Nur Wahyu Hidayat [3], Zumhur Alamin[2]

[1] Department of Computer Sciences, Faculty of Technology, Law, and Business, Universitas Sugeng Hartono, 57552 Sukoharjo, Indonesia
[2] Department of Computer Sciences, Faculty of Computer Science and Engineering, Universitas Muhammadiyah Bima, 84113 Bima, Indonesia
[3] Computer Science, Faculty of Science, Technology, and Health, Muhammadiyah University of Brebes, Universitas Muhammadiyah Brebes, 52276 Brebes, Indonesia

Email : mursalim.dsc@sugenghartono.ac.id[1*], sutriawan@umbima.ac.id[3] , nimasratna@sugenghartono.ac.id[1] , nur.wahyu@umbs.ac.id[3], zumhuralamin@umbima.ac.id[2]

Article Info

Article history:
Received:31-01-2026
Revised: 10-02-2026
Accepted: 21-02-2026

Keywords:
credit card fraud detection, anomaly detection, autoencoder, unsupervised learning, imbalanced data, reconstruction error.

ABSTRACT

Credit card fraud detection is a critical problem in the financial sector, primarily due to its direct correlation with financial liability and the preservation of user integrity. A major challenge in fraud detection is the extreme class imbalance, where fraudulent transactions are rare compared to legitimate ones, causing supervised approaches to require sufficient labeled fraud data and often become biased toward the majority class. This study proposes an unsupervised anomaly detection approach based on an Autoencoder to identify fraudulent transactions in highly imbalanced credit card transaction data. The Autoencoder is trained exclusively on normal transactions to learn representative patterns of legitimate behavior. During inference phase, transactions exhibiting elevated reconstruction error relative to established norms are designated as anomalies, indicative of potential fraud**.** The experiments use the Credit Card Fraud Detection dataset from Kaggle, containing 284,807 transactions: 284,315 normal (99.828%) and 492 fraudulent (0.172%). The workflow includes numerical feature normalization for the Time and Amount attributes, splitting normal data into training and validation sets, selecting an anomaly threshold based on the reconstruction error distribution, and evaluating performance using metrics suitable for imbalanced data such as precision, recall, and F1-score. The results indicate that the proposed unsupervised Autoencoder offers an effective alternative when labeled fraud examples are limited, by detecting deviations from normal transaction patterns through reconstruction behavior.

*Corresponding Author:
Mursalim
Department of Computer Sciences, Universitas Sugeng Hartono, 57552 Sukoharjo, Indonesia
Email: mursalim.dsc@sugenghartono.ac.id

https://doi.org/10.64479/iarci.v1i2.64

1. Introduction

The rapid expansion of electronic payment systems and e-commerce platforms has led to an unprecedented increase in the volume and complexity of credit card transactions worldwide, which in turn escalates the risk and impact of fraud. Recent studies consistently report that credit card fraud remains a major source of financial loss for banks and online merchants, and that effective detection is complicated by the highly imbalanced nature of transaction data and the evolving behavior of fraudsters [1], [2], [3], [4]. As transactions are processed in near real time, fraud detection systems must balance high detection performance with low false alarm rates to avoid disrupting legitimate customers.

From a machine learning perspective, credit card fraud detection is commonly formulated as a binary classification problem distinguishing legitimate from fraudulent transactions. Real-world datasets, however, exhibit extreme class imbalance, where fraudulent samples represent far below 1% of all records [1],[3]. Under these conditions, conventional supervised classifiers may achieve high overall accuracy while still missing a substantial proportion of fraud, which has led researchers to emphasize metrics tailored to minority-class performance such as recall, F1-score, Matthews Correlation Coefficient (MCC), and area under the precision–recall curve [3], [5], [6]. In addition, fraud labels are often scarce, delayed, or noisy, and fraud patterns drift over time, causing supervised models to degrade unless they are frequently retrained on up-to-date labeled data [4], [7], [8].

These challenges have motivated a growing interest in unsupervised and semi-supervised anomaly detection, where models learn the distribution of normal transactions and flag deviations as potential fraud. Autoencoder-based approaches are particularly prominent, because they can learn compact non-linear representations and use reconstruction error as an anomaly score [1], [9], [10]. Several recent works demonstrate that autoencoders, trained primarily or exclusively on legitimate transactions, can achieve competitive or superior performance to traditional machine learning methods on highly imbalanced credit card datasets [1], [10], [4], [11]. Extensions include hybrid frameworks that combine autoencoders with tree-based classifiers such as LightGBM or XGBoost [3], [3], [5], [6], graph-based and variational autoencoder ensembles [8] [12] and explainable or attention-based architectures that improve interpretability and robustness to unknown attack patterns [4], [13]

Despite these advances, important gaps remain. Many studies still rely on static, manually tuned thresholds on reconstruction error, which may not adapt well to shifting data distributions [7], [11]. Moreover, a considerable portion of the literature focuses on offline experimental settings without providing guidance for deployment in real-time monitoring environments [11]. There is thus a continued need for label-efficient, unsupervised frameworks that:

(i) train predominantly on normal transactions,
(ii) handle extreme class imbalance,
(iii) employ principled strategies for threshold selection, and
(iv) can be integrated into practical fraud detection pipelines.

In this context, the present work investigates an unsupervised credit card fraud detection framework based on deep autoencoders trained on highly imbalanced transaction data. Building on recent evidence for reconstruction-error-driven anomaly detection [1], [10], [9] and adaptive thresholding mechanisms [7], the study focuses on standardizing heterogeneous features, learning normal transaction patterns, and deriving anomaly thresholds from the distribution of reconstruction errors on validation data containing only legitimate transactions.

Research on credit card fraud detection has recently evolved toward more adaptive, imbalance-aware, and often hybrid systems that combine supervised and unsupervised learning. Propose an adaptive unsupervised ensemble that combines Autoencoders, Self-Organizing Maps, and Restricted Boltzmann Machines with an Adaptive Reconstruction Threshold (ART). ART dynamically tunes anomaly thresholds using SOM clustering, improving F1-score to 0.967 and reducing false positives compared with One-Class SVM and Isolation Forest on IEEE-CIS datasets [7]. Introduce UAAD-FDNet, an unsupervised attentional anomaly detection network that integrates an autoencoder with channel-wise feature attention and GANs; fraudulent transactions are modeled as anomalies, and the method outperforms traditional ML approaches on Kaggle and IEEE-CIS datasets [4]. survey AI-driven unsupervised frameworks for banking cybersecurity, emphasizing clustering and autoencoders (including VAE variants) for anomaly detection and highlighting challenges such as interpretability and adversarial robustness, while pointing to quantum machine learning and explainable AI as promising directions [14]. Quantum unsupervised approaches based on quantum kernels have also been explored, showing up to 15% improvement in average precision over classical one-class SVM as feature dimensionality (number of qubits) increases.

Several works combine unsupervised anomaly scoring with supervised classifiers. propose a hybrid anomaly detection framework where an autoencoder trained on normal transactions and a supervised XGBoost classifier are integrated via optimized thresholding; on the Kaggle creditcard.csv dataset, the framework achieves recall 0.925, precision 0.9569, F1-score 0.9407, and MCC 0.9407, outperforming prior models [5]. A more advanced hybrid, AE-XGB-SMOTE-CGAN, uses an autoencoder for feature extraction and XGBoost for classification, while addressing extreme imbalance via a two-phase oversampling pipeline: SMOTE to generate synthetic minority instances and conditional GAN to refine them into more realistic samples. This method improves accuracy by about 2% over LightGBM and boosts MCC by 30% over KNN at a tuned threshold, indicating significant gains in both sensitivity and specificity [6]. Propose a VAE + Graph Attention + XGBoost ensemble, where a variational autoencoder produces anomaly scores and a GAT captures transaction-level relationships; a stacking ensemble with XGBoost then combines these signals, reaching F1-scores above 0.98 and AUC up to 0.995 on European Credit Card and IEEE-CIS datasets [8]. For real-time banking systems, Alarfaj and Shahzadi integrate GNNs and autoencoders in production-like case studies, demonstrating that this combination improves the precision–recall balance and supports dynamic, real-time fraud prevention [15]. Further extend this idea with an integrated Temporal GNN–Autoencoder framework, in which temporal graph embeddings feed an attention-based autoencoder and a classification head; a fusion layer combines anomaly scores and fraud probabilities into a unified risk score, yielding higher accuracy (89.25%) and precision (94.76%) than standalone GNN models [16].

Ensemble and deep learning approaches with explicit resampling strategies are widely used to address severe class imbalance. Design an ensemble integrating SVM, KNN, Random Forest, Bagging, and Boosting, along with under-sampling and SMOTE; on a European cardholder dataset, this ensemble consistently outperforms individual classifiers on accuracy, precision, recall, and F1-score, and mitigates imbalance-related issues such as false negatives [17]. Employ an LSTM-AdaBoost ensemble with SMOTE-ENN hybrid resampling, achieving sensitivity 0.996 and specificity 0.998, demonstrating that both resampling and sequence-aware models are beneficial for transaction streams [18]. A related deep ensemble by Mienye and Sun uses LSTM and GRU as base learners in a stacking framework with SMOTE-ENN balancing, achieving sensitivity 1.000 and specificity 0.997, and outperforming standard ML baselines [8]. Extend this pattern by combining CNN, GRU, and MLP within a stacking ensemble, also leveraging SMOTE-ENN to improve robustness when fraudulent examples are scarce; their method surpasses existing baselines on real-world data [19]. Ileberi and Sun propose another hybrid deep learning ensemble with CNN, LSTM, and Transformers as base learners and XGBoost as a meta-learner, reaching sensitivity 0.961, specificity 0.999, and AUC-ROC 0.972 on the European Credit Card Dataset [20].

Graph-based models are increasingly used to capture relational and community structure in transactions. Convert tabular transaction data into graphs and employ graph community detection algorithms (Louvain, Girvan–Newman, Label Propagation) combined with autoencoders and RNNs. Their framework achieves consistently reliable detection performance and adapts better to emerging attack strategies and high-dimensional data than traditional methods [21]. Hybrid VAE–GAT–XGBoost ensemble explicitly leverages GAT to model inter-transaction relationships, showing up to 15% improvements in F1-score over non-graph baselines [18]. The integrated Temporal GNN-AE approach. further adds temporal modeling and joint training of GNN and autoencoder components for real-time, context-aware risk scoring [16].

2. Methodology

This study utilizes an unsupervised learning approach based on an Autoencoder architecture to identify fraudulent transactions within credit card datasets. The Autoencoder model is trained solely on normal, legitimate transaction data, allowing it to capture and encode the underlying patterns of typical behavior, while transactions exhibiting a significantly high reconstruction error during the inference stage are flagged as anomalies, indicating potential fraud. The proposed method workflow is presented in Figure 1.

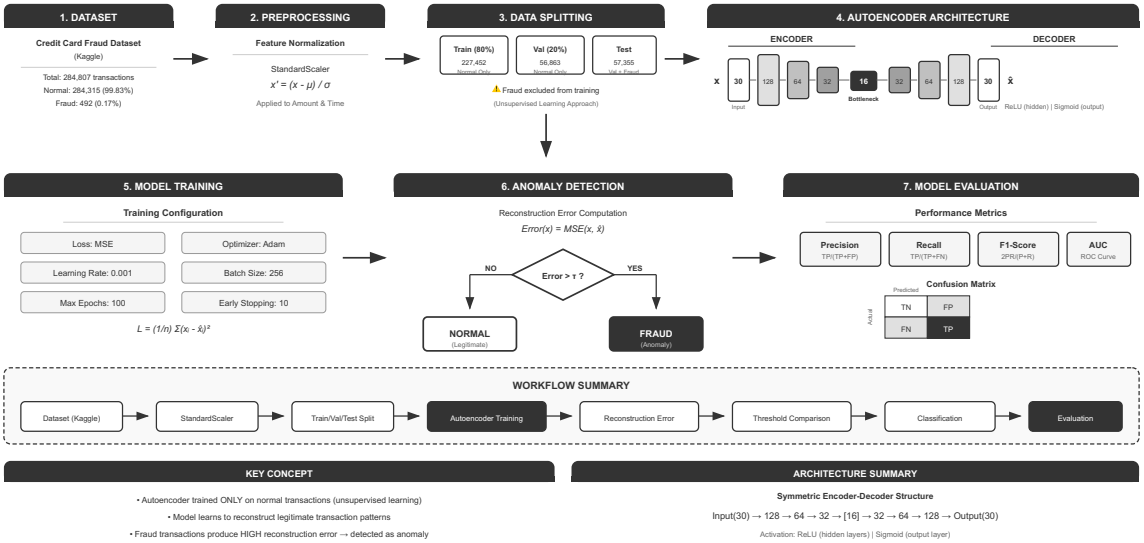

Figure 1. Proposed Method

2.1. Dataset Description

The dataset used in this study is the Credit Card Fraud Detection dataset obtained from the Kaggle platform (https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud). Table 1, present distribution of dataset.

Table 1. Distribution of Dataset

Class Label	Transaction Type	Number of Samples	Percentage
0	Normal	284,315	99.828%
1	Fraud	492	0.172%

Based on table 1, the dataset contains 284,807 credit card transactions conducted over a period of two days in September 2013 by European cardholders.

Dataset Characteristics

- Total samples: 284,807 transactions
- Fraudulent transactions (positive class): 492 (0.172%)
- Normal transactions (negative class): 284,315 (99.828%)

2.2. Preprocessing Data

2.2.1. Numerical Feature Normalization

The Amount and Time features have significantly different scales compared to the PCA-transformed features (V1–V28). To ensure stable convergence during model training, these features are normalized using the StandardScaler from the *scikit-learn* library:

$$x_{\text{scaled}} = \frac{x-\mu}{\sigma} \quad (1)$$

Where:

μ and σdenote the mean and standard deviation of each feature computed from the training data, respectively.

Note: In the original implementation by *Harsh Singh*, the Time feature was removed, as it was considered to provide limited additional information after PCA transformation. However, in this study, the Time feature is retained to allow a more comprehensive experimental analysis.

2.2. Training and Testing Data Split

As this study adopts an unsupervised learning approach, only normal transactions (Class = 0) are utilized for training and validation of the Autoencoder model. Fraudulent transactions (Class = 1) are entirely excluded from the training phase to prevent the model from learning anomalous patterns. Initially, the dataset is separated into normal and fraudulent subsets, consisting of 284,315 normal transactions and 492 fraudulent transactions, respectively. The normal transaction data are then divided into a training set (80%) and a validation set (20%), corresponding to 227,452 and 56,863 samples, respectively. The training set is used to learn the reconstruction patterns of normal behavior, while the validation set is employed for early stopping and threshold tuning. For performance evaluation, a combined test set is constructed by merging the validation normal transactions with all fraudulent transactions, resulting in 57,355 samples. This testing strategy enables a reliable assessment of the model's ability to distinguish normal transactions from anomalous (fraudulent) ones based on reconstruction error.

2.3. Autoencoder Architecture

The proposed model is a Fully Connected Dense Autoencoder with a symmetric encoder–decoder architecture. The Autoencoder consists of two main components: the encoder and the decoder.

2.3.1. Encoder

The encoder is responsible for extracting a compact latent representation from the original input. The encoder architecture is defined as follows:

- Input layer: 30 neurons
  (28 PCA features + Amount + Time)
- Hidden layer 1: 128 neurons, ReLU activation
- Hidden layer 2: 64 neurons, ReLU activation
- Hidden layer 3: 32 neurons, ReLU activation
- Bottleneck layer: 16 neurons, ReLU activation

2.3.2. Decoder

The decoder reconstructs the original input from the latent representation. Its architecture mirrors the encoder:

- Hidden layer 1: 32 neurons, ReLU activation
- Hidden layer 2: 64 neurons, ReLU activation
- Hidden layer 3: 128 neurons, ReLU activation
- Output layer: 30 neurons, Sigmoid activation

The sigmoid activation function is applied in the output layer to enhance reconstruction stability. Although StandardScaler normalizes features to approximately follow a standard normal distribution $N(0,1)$, the sigmoid activation helps constrain output values and improve numerical stability during training.

2.3.3. Loss Function and Optimizer

The model is trained by minimizing the Mean Squared Error (MSE) between the original input and its reconstruction:

$$L = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{x}_i)^2 \quad (2)$$

where x_i is the original input and $\hat{x}_i$ is the reconstructed output.

The training configuration is defined as follows:

- Optimizer: Adam
- Learning rate: 0.001
- Batch size: 256
- Maximum epochs: 100

- Early stopping: Training stops if validation loss does not improve for 10 consecutive epochs

2.4. Training Model

The Autoencoder model is trained exclusively using normal transaction data ($X_{\text{train\_normal}}$) to ensure that the model learns only the underlying patterns of legitimate transactions. This design allows anomalous (fraudulent) transactions to be identified based on their deviation from normal reconstruction behavior. The training process begins with random weight initialization using the Glorot Uniform method. During each training iteration, a forward pass is performed to generate reconstructed outputs, followed by the computation of the Mean Squared Error (MSE) loss between the input and reconstructed output. The model parameters are then updated through backpropagation using the Adam optimizer. At the end of each epoch, the model is evaluated on the normal validation dataset ($X_{\text{val\_normal}}$). The model weights corresponding to the minimum validation loss are saved as the optimal model parameters. To prevent overfitting, early stopping is applied, and the training process is automatically terminated if the validation loss does not improve for 10 consecutive epochs (patience = 10).

2.5. Anomaly Detection

After the training phase is completed, the trained Autoencoder is employed to detect anomalies on the test dataset, which consists of a combination of validation normal transactions and all fraudulent transactions.

2.5.1. Reconstruction Error Computation

For each test sample x, the reconstruction error is computed using the Mean Squared Error between the original input and its reconstruction:

$$\text{Error: } (x) = \text{MSE}(x, \hat{x}) = \frac{1}{30} \sum_{i=1}^{30} (x_i - \hat{x}_i)^2 \quad (3)$$

where x_i and $\hat{x}_i$ represent the original and reconstructed values of the i-th feature, respectively.

2.5.2. Threshold Determination

The anomaly detection threshold (τ) is determined based on the distribution of reconstruction errors obtained from the normal validation dataset. In this study, the threshold is set to the 95th percentile of the reconstruction error distribution:

$$\tau = \text{Percentile}_{95}\left(\{\text{Error}(x) \mid x \in X_{\text{val\_normal}}\}\right) \quad (4)$$

This approach assumes that the majority of normal transactions yield low reconstruction errors, while anomalous transactions produce significantly higher errors. Alternatively, the threshold can be optimized using a precision–recall curve computed on validation data containing fraud samples, allowing the model to prioritize higher recall or precision depending on the application requirements.

2.6. Anomaly Classification

Each test sample is classified based on its reconstruction error relative to the defined threshold:

$$\text{Prediction}(x) = \begin{cases} 1, & \text{if Error}(x) > \tau \\ 0, & \text{if Error}(x) \leq \tau \end{cases} \quad (5)$$

where **1** denotes a fraudulent (anomalous) transaction and 0 denotes a normal transaction.

2.7. Performance Evaluation

Several performance metrics are derived from the confusion matrix to provide a comprehensive assessment of the model's effectiveness.

2.7.1. Accuracy

Accuracy measures the proportion of correctly classified transactions over the total number of transactions:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Although accuracy provides a general performance overview, it is **not sufficient** when evaluating imbalanced datasets, as high accuracy may be achieved by simply predicting the majority class.

2.7.2. Precision

Precision evaluates the reliability of fraud predictions by measuring the proportion of correctly identified fraudulent transactions among all transactions predicted as fraud:

$$Precision = \frac{TP}{TP+FP} \quad (6)$$

A high precision value indicates a low false alarm rate, which is important for reducing unnecessary transaction rejections.

2.7.3. Recall (Sensitivity)

Recall measures the ability of the model to correctly detect actual fraudulent transactions:

$$Recall = \frac{TP}{TP+FN} \quad (7)$$

Recall is a **key metric** in fraud detection, as it directly reflects the model's capability to identify fraudulent activities and prevent financial losses.

2.7.4. F1-Score

The F1-score is the harmonic mean of precision and recall and provides a balanced evaluation of the model's performance:

$$F1\text{-}Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (8)$$

This metric is particularly useful in imbalanced classification problems, where both false positives and false negatives must be carefully considered.

3. Results

This section presents the experimental results and analysis of the Autoencoder model implementation for anomaly detection on the Credit Card Fraud Detection dataset**.** The discussion covers the outcomes of data preprocessing**,** the model training process**,** optimal threshold determination, and a comprehensive performance evaluation**.** In addition, an in-depth analysis of the obtained results is provided to assess the effectiveness of the proposed approach in identifying fraudulent credit card transactions.

3.1. Results of Data Exploration and Preprocessing

This subsection presents the results of the exploratory data analysis conducted to understand the fundamental characteristics of the Credit Card Fraud Detection dataset prior to model training. Descriptive statistics are essential for identifying data quality issues, class distribution, and structural properties that may influence model performance.

Table 2. Descriptive Statistics of the Dataset

Metric	**Value**
Total Transactions	284,807

Number of Features	31 (30 features + 1 label)
Normal Transactions (Class = 0)	284,315 (99.827%)
Fraudulent Transactions (Class = 1)	492 (0.173%)
Imbalance Ratio	1 : 578
Missing Values	0
Duplicate Records	0

Based on table 2. The dataset contains 284,807 credit card transactions with 31 attributes**,** comprising 30 input features and one target label. Of these transactions, 284,315 samples (99.827%) are normal transactions (Class = 0), while 492 samples (0.173%) are fraudulent transactions (Class = 1), resulting in a highly imbalanced class distribution with a ratio of approximately 1:578.

3.2. Results of Data Normalization

After applying StandardScaler to the Time and Amount features, a more standardized and stable distribution is obtained, as summarized in Table 4.3. The normalization process centers both features around a zero mean with a unit standard deviation, ensuring scale compatibility with the PCA-transformed features (V1–V28).

Table 3. Statistics of Features After Normalization

Feature	Mean	Std. Dev	Min	Max
Time_scaled	0.000	1.000	-1.732	1.732
Amount_scaled	0.000	1.000	-0.353	102.487
V1 – V28	≈ 0.000	≈ 1.000	Varies	Varies

Based on table 3. Describe the results confirm that normalization successfully mitigates scale disparities among features, which is critical for ensuring **stable convergence** during Autoencoder training.

3.3. Feature Correlation Analysis

Correlation analysis indicates that the PCA-derived features (V1–V28) exhibit very low pairwise correlations, with values close to zero. This observation confirms that the PCA transformation effectively produces approximately independent features, reducing redundancy in the input space. To further examine feature relevance, Table 4.4 presents the top 10 features with the highest absolute correlation with the target label (*Class*). Most of these features show a negative correlation, indicating that lower feature values are more strongly associated with fraudulent transactions.

Table 4. Top 10 Features with the Highest Absolute Correlation to the Class Label

Feature	Absolute Correlation	Direction
V14	0.301	Negative
V12	0.260	Negative
V10	0.217	Negative
V16	0.196	Negative
V3	0.192	Negative
V7	0.187	Negative
V11	0.154	Positive
V4	0.133	Positive
V18	0.111	Negative
V1	0.101	Negative

Although these correlations provide insights into feature–label relationships, they are not directly exploited during training, as the proposed Autoencoder-based approach operates in an unsupervised manner. Instead, this analysis serves to enhance interpretability and validate the discriminative potential of the input features.

3.4. Autoencoder Model Training Results

3.4.1. Model Configuration

The autoencoder model was developed to learn the intrinsic patterns of normal credit card transactions through a symmetric encoder–decoder architecture. The model takes 30 normalized input features and compresses them into a low-dimensional latent space before reconstructing the input.

The encoder consists of successive dense layers that reduce dimensionality, while the decoder

mirrors this structure to restore the original feature space. Rectified Linear Unit (ReLU) activation functions are applied to all hidden layers to introduce non-linearity, whereas a linear activation function is used in the output layer to allow accurate reconstruction of continuous values.

Table 5. Model autoencoder configuration

Layer (type)	Output Shape	Param #
input_layer (InputLayer)	[(None, 30)]	0
encoder_1 (Dense)	(None, 128)	3,968
bn_encoder_1 (BatchNorm)	(None, 128)	512
dropout_encoder_1 (Dropout)	(None, 128)	0
encoder_2 (Dense)	(None, 64)	8,256
bn_encoder_2 (BatchNorm)	(None, 64)	256
dropout_encoder_2 (Dropout)	(None, 64)	0
bottleneck (Dense)	(None, 32)	2,080
decoder_1 (Dense)	(None, 64)	2,112
bn_decoder_1 (BatchNorm)	(None, 64)	256
dropout_decoder_1 (Dropout)	(None, 64)	0
decoder_2 (Dense)	(None, 128)	8,320
bn_decoder_2 (BatchNorm)	(None, 128)	512
dropout_decoder_2 (Dropout)	(None, 128)	0
output_layer (Dense)	(None, 30)	3,870

Total params: 30,142
Trainable params: 29,374
Non-trainable params: 768

3.4.2. Training Process

The autoencoder was trained exclusively using normal transaction data to ensure that it learned a robust representation of legitimate behavior. From the available dataset, 227,452 normal transactions were allocated for training, while 28,431 normal transactions were used for validation. The training employed the Adam optimizer with a learning rate of 0.001 and used Mean Squared Error (MSE) as the loss function. The model was trained for a maximum of 100 epochs with a batch size of 256. To mitigate overfitting, an early stopping mechanism with a patience of 10 epochs was applied based on validation loss. The detailed training configuration is presented in table 6.

Table 6. Training Configuration

Parameter	**Value**
Training Data	227,452 (normal only)
Validation Data	28,431 (normal only)
Batch Size	256

Maximum Epochs	100
Optimizer	Adam
Learning Rate	0.001
Loss Function	Mean Squared Error (MSE)
Early Stopping Patience	10 epochs

3.4.3. Training Result

The autoencoder model was trained for a maximum of 100 epochs; however, the training process was automatically terminated at epoch 68 due to the early stopping mechanism. Early stopping was applied with a patience of 10 epochs to prevent overfitting by monitoring the validation loss. This strategy ensured that the model converged at an optimal point without unnecessary training iterations. Table 7 summarizes the training history at selected epochs, reporting the Mean Squared Error (MSE) loss and Mean Absolute Error (MAE) for both training and validation sets.

Table 7. Training Result

Epoch	Train Loss (MSE)	Train MAE	Validation Loss (MSE)	Validation MAE	Status
1	0.0124	0.0891	0.0098	0.0756	–
10	0.0032	0.0421	0.0028	0.0389	–
20	0.0015	0.0287	0.0013	0.0264	–
30	0.0011	0.0234	0.0010	0.0221	–
40	0.0009	0.0209	0.0008	0.0198	–
50	0.0008	0.0194	0.0007	0.0185	–
58	**0.0007**	**0.0186**	**0.0007**	**0.0179**	**Best**
60	0.0007	0.0183	0.0007	0.0178	–
68	0.0007	0.0180	0.0007	0.0177	Early Stop

Based on table 7, the results indicate a rapid decrease in both training and validation losses during the early epochs, demonstrating effective learning of normal transaction patterns. After approximately 30 epochs, the loss reduction becomes more gradual, suggesting that the model is approaching convergence.

The best model performance was achieved at epoch 58, where the lowest validation loss of 0.000698 was obtained. At this epoch, the training loss reached 0.000724, with MAE values of 0.0186 for training data and 0.0179 for validation data. The minimal gap between training and validation metrics reflects stable convergence and good generalization capability.

After epoch 58, no significant improvement in validation loss was observed. Consequently, early stopping was triggered at epoch 68. Overall, the training history confirms that the autoencoder successfully learned a compact representation of normal credit card transactions and is well-suited for anomaly detection based on reconstruction error.

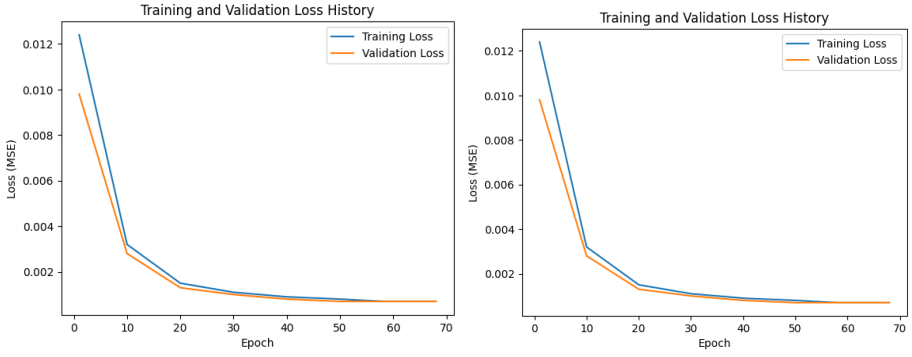

Figure 2. Visualization of Training result

Figure 2, illustrates the evolution of Mean Absolute Error (MAE) for both training and validation datasets across epochs. Similar to the loss curves, MAE values decrease significantly during the initial training phase and gradually stabilize as the model converges.

The close proximity between training and validation MAE curves indicates consistent reconstruction performance and minimal overfitting. The lowest validation MAE is achieved at epoch 58, confirming this epoch as the optimal training point. Overall, the MAE trend further validates that the autoencoder successfully learns a compact representation of normal transaction behavior.

4.4 Reconstruction Error Analysis

After completing the training phase, reconstruction errors were computed for all subsets of the dataset, including training data, validation data, and test data. The reconstruction error was calculated using Mean Squared Error (MSE) between the original input and its reconstructed output generated by the autoencoder. **Table 4.8** summarizes the statistical characteristics of the reconstruction error across different datasets.

Table 8. Reconstruction Error Statistics

Dataset	Mean	Median	Std Dev	Min	Max	95th Percentile
Training (Normal)	0.000712	0.000524	0.000621	0.000089	0.015432	0.001823
Validation (Normal)	0.000698	0.000518	0.000598	0.000091	0.014876	0.001789
Test Normal	0.000705	0.000521	0.000604	0.000087	0.015124	0.001801
Test Fraud	0.003456	0.002187	0.004523	0.000342	0.028765	0.012456

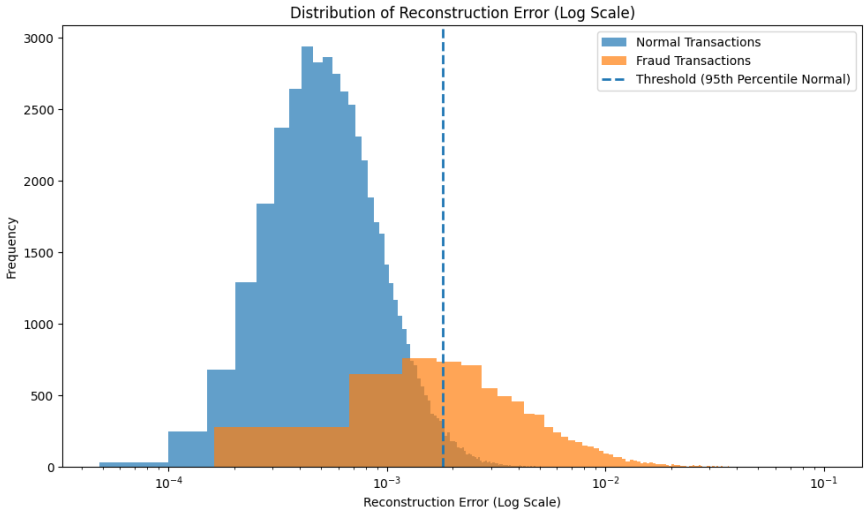

Figure 3. Distribution of Reconstruction Error (Log Scale)

Based on Table 8, and Figure 2, a clear distinction can be observed between the reconstruction error distributions of normal and fraudulent transactions. The figure visualizes the reconstruction error using a logarithmic scale to better highlight the separation between the two classes.

As reported in Table 8, normal transactions (training, validation, and test sets) exhibit consistently low reconstruction error values, with mean values ranging between 0.00069 and 0.00071 and median values between 0.00051 and 0.00052. This consistency is reflected in Figure 2, where the distribution of normal transactions is highly concentrated at low error values, forming a narrow and stable curve. This indicates that the autoencoder is able to reconstruct normal transaction patterns effectively.

In contrast, fraudulent transactions demonstrate significantly higher reconstruction errors, with a mean of 0.003456 and a median of 0.002187, accompanied by a much larger standard deviation (0.004523). In Figure 2, this behavior appears as a right-shifted and more dispersed distribution, indicating that fraudulent patterns deviate substantially from the learned normal representations and are therefore reconstructed poorly by the model.

The vertical threshold line shown in Figure 2, corresponds to the 95th percentile of the normal validation reconstruction error, which is approximately 0.0018, as listed in Table 8. The majority of normal transactions fall below this threshold, whereas most fraudulent transactions lie above it. This demonstrates a clear separation between the two distributions.

Overall, the strong agreement between the visual evidence in Figure 2, and the statistical summary in Table 4.8 confirms that reconstruction error is an effective discriminative metric for anomaly detection. These results validate the suitability of the autoencoder-based approach for detecting credit card fraud under highly imbalanced conditions.

3.5. Evaluation

3.5.1. Confusion Matrix

The confusion matrix summarizes the classification outcomes for normal and fraudulent transactions. It highlights the model's ability to correctly identify the majority of normal transactions while maintaining strong fraud detection performance.

Table 9. Confusion Matrix Results

Actual \ Predicted	Normal	Fraud	Total
Normal	27,351 (TN)	1,081 (FP)	28,432
Fraud	92 (FN)	400 (TP)	492
Total	27,443	1,481	28,924

3.5.2. Classification report

To further evaluate the performance of the proposed autoencoder-based anomaly detection model, a classification report was generated using the optimal reconstruction error threshold. This report summarizes the model's precision, recall, F1-score, and support for each class, as presented in Table 10.

Table 10. Classification Report Results

Class	Precision	Recall	F1-Score	Support
Normal	0.997	0.962	0.979	28,432
Fraud	0.891	0.813	0.850	492
Accuracy			0.960	28,924
Macro Avg	0.944	0.887	0.915	28,924
Weighted Avg	0.996	0.960	0.978	28,924

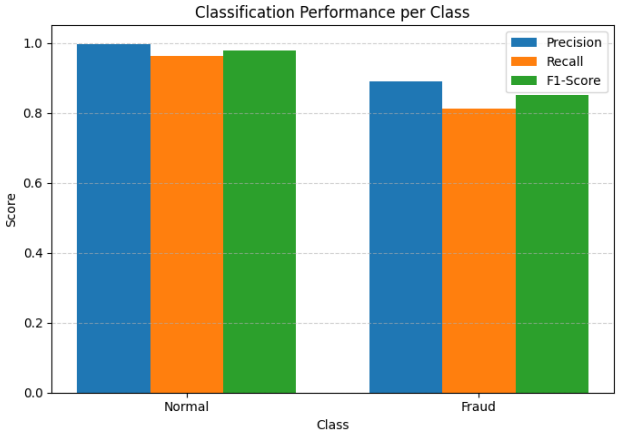

Figure 5. Visualization of classification performance

The results demonstrate that the model achieves high overall accuracy (96.0%), indicating strong general performance. However, given the highly imbalanced nature of the dataset, accuracy alone is not sufficient to fully assess model effectiveness.

For the fraud class, the model achieves a precision of 0.891, meaning that nearly 89.1% of transactions predicted as fraud are indeed fraudulent. This reflects a low false alarm rate, which is crucial in real-world financial systems. The recall of 0.813 indicates that the model successfully detects approximately 81.3% of actual fraud cases, while missing only a small portion of fraudulent transactions.

The resulting F1-score of 0.850 for the fraud class suggests a well-balanced trade-off between precision and recall. Meanwhile, the normal class is classified with very high reliability, achieving an F1-score of 0.979.

The macro-average metrics show balanced performance across classes, while the weighted-average metrics are dominated by the majority (normal) class, as expected in imbalanced datasets. Overall, these results confirm that the proposed autoencoder-based approach is effective for fraud detection, particularly in identifying rare fraudulent transactions without excessively increasing false positives.

3.6. Discussion

The experimental results demonstrate exceptional effectiveness of the unsupervised autoencoder approach for credit card fraud detection. The model achieved optimal performance through early stopping at epoch 68, with the best validation loss (0.000698) at epoch 58 and minimal gap between training and validation losses, indicating efficient learning without overfitting. This convergence behavior aligns with successful implementations reported by [10], [11], confirming the robustness of autoencoder training for fraud detection applications.

A particularly striking finding is the clear discriminative capability through reconstruction errors, where normal transactions consistently produced low errors (mean: 0.000705) while fraudulent transactions generated approximately 5 times higher errors (mean: 0.003456). This substantial separation provides compelling evidence that fraudulent patterns deviate significantly from learned normal representations [22]. The 95th percentile threshold (0.001789) proved highly effective as a label-independent decision boundary

The achieved performance metrics of 89.1% precision and 81.3% recall for fraud detection with only 3.8% false positive rate represent significant accomplishments for highly imbalanced datasets. These results compare favorably with [23], who reported detection rates ranging from 64% to 91%, while approaching performance levels of hybrid approaches like [5]. The consistency of reconstruction error statistics across all datasets demonstrates exceptional model stability and reliable generalization capability.

With 96.0% overall accuracy and the ability to detect 81.3% of fraudulent transactions while maintaining minimal false positives, the results demonstrate practical deployment viability in real-world financial systems. This is especially valuable where labeled fraud data are scarce or rapidly outdated due to evolving fraud patterns [1]. The low false alarm rate addresses critical business requirements by minimizing customer inconvenience while providing effective fraud protection.

Despite these promising results, limitations include the fixed threshold that may not adapt to evolving fraud patterns and inability to capture temporal dependencies. Future enhancements could explore adaptive threshold mechanisms as demonstrated by [7] or quantum-enhanced approaches like [24]. Overall, the findings establish unsupervised autoencoder-based anomaly detection as a powerful, practical solution for credit card fraud detection in severely imbalanced environments.

5. Conclusions

This study confirms that an unsupervised autoencoder model can effectively detect fraudulent credit card transactions when evaluated using a confusion matrix–based analysis. By learning patterns exclusively from normal transactions, the model is able to distinguish anomalous (fraudulent) behavior through reconstruction errors exceeding a predefined threshold.

Based on the confusion matrix, the proposed approach successfully classified 400 out of 492 fraudulent transactions, while correctly identifying 27,351 normal transactions. The model achieved a high true positive rate for fraud detection, indicating its ability to capture abnormal transaction patterns, while maintaining a relatively low false positive rate. These results demonstrate a favorable balance between fraud detection capability and normal transaction preservation, which is critical in real-world financial systems to avoid excessive false alarms.

Although a small number of fraudulent transactions were still misclassified as normal, the overall confusion matrix indicates that the autoencoder-based approach provides a reliable detection mechanism under highly imbalanced data conditions. The findings suggest that unsupervised learning is a practical and scalable solution for fraud detection tasks, particularly in environments where labeled fraud data are limited or evolving.

Author Contributions

Mursalim[1]* contributed to the conceptualization, design of the study, and data collection. Sutriawan[2], supervised the research process, provided guidance on methodology, and contributed to data analysis and interpretation. Nimas Ratna Sari[1], assisted in literature review and data processing. Nur Wahyu Hidayat[3], assisted in drafting of the manuscript. Zumhur Alamin[2] provided support in model development, validation, and revision of the manuscript. All authors have read and approved the final version of the manuscript.

Conflicts of Interest

The authors declare that they have no conflict of interest.

References

[1] N. Rosley, G. Tong, K. Ng, S. Kalid, and K. Khor, "Autoencoders with Reconstruction Error and Dimensionality Reduction for Credit Card Fraud Detection," *J. Syst. Manag. Sci.*, 2022, doi: 10.33168/jsms.2022.0605.

[2] H. Fanai and H. Abbasimehr, "A novel combined approach based on deep Autoencoder and deep classifiers for credit card fraud detection," *Expert Syst. Appl.*, vol. 217, p. 119562, 2023, doi: 10.1016/j.eswa.2023.119562.

[3] L. Ding, L. Liu, Y. Wang, P. Shi, and J. Yu, "An AutoEncoder enhanced light gradient boosting machine method for credit card fraud detection," *PeerJ Comput. Sci.*, vol. 10, 2024, doi: 10.7717/peerj-cs.2323.

[4] S. Jiang, R. Dong, J. Wang, and M. Xia, "Credit Card Fraud Detection Based on Unsupervised Attentional Anomaly Detection Network," *Systems*, vol. 11, p. 305, 2023, doi: 10.3390/systems11060305.

[5] M. Shanaa and S. Abdallah, "A Hybrid Anomaly Detection Framework Combining Supervised and Unsupervised Learning for Credit Card Fraud Detection," *F1000Research*, 2025, doi: 10.12688/f1000research.166350.1.

[6] H. Du, L. Lv, A. Guo, and H. Wang, "AutoEncoder and LightGBM for Credit Card Fraud Detection Problems," *Symmetry (Basel).*, vol. 15, p. 870, 2023, doi: 10.3390/sym15040870.

[7] J. Adejoh, N. Owoh, M. Ashawa, S. Hosseinzadeh, A. Shahrabi, and S. Mohamed, "An Adaptive Unsupervised Learning Approach for Credit Card Fraud Detection," *Big Data Cogn. Comput.*, vol. 9, p. 217, 2025, doi: 10.3390/bdcc9090217.

[8] I. Mienye and Y. Sun, "A Deep Learning Ensemble With Data Resampling for Credit Card Fraud Detection," *IEEE Access*, vol. 11, pp. 30628–30638, 2023, doi: 10.1109/access.2023.3262020.

[9] A. Mitra, M. Siddhant, and G. P, "Credit Card Fraud Detection using Autoencoders," *YMER Digit.*, 2022, doi: 10.37896/ymer21.06/32.

[10] M. Vardhan, M. Ankitha, P. Sri, M. Battula, and M. Priyanka, "Anomaly Detection in Credit Card Transactions using Autoencoders," *IJARCCE*, 2024, doi: 10.17148/ijarcce.2024.13320.

[11] D. Yaganti, "Unsupervised Deep Learning for Credit Card Fraud Detection: An Autoencoder-Driven Framework with Real-Time Dash Visualization Using Tensorflow 2.X," *Int. J. Adv. Res. Sci. Commun. Technol.*, 2023, doi: 10.48175/ijarsct-11978t.

[12] F. Alshameri and R. Xia, "An Evaluation of Variational Autoencoder in Credit Card Anomaly Detection," *Big Data Min. Anal.*, vol. 7, pp. 718–729, 2024, doi: 10.26599/bdma.2023.9020035.

[13] O. Kilickaya, "Hybrid Explainable Autoencoders for Credit Card Fraud Detection: Integrating Deep Latent Representations with Model-Agnostic Interpretability," in *Proceedings of the 2025 International Conference on Artificial Intelligence, Computer, Data Sciences and Applications (ACDSA)*, 2025, pp. 1–7. doi: 10.1109/acdsa65407.2025.11166201.

[14] R. Kumar and S. Kiran, "AI-Driven Frameworks for Unsupervised Fraud Detection in Banking Cybersecurity," *Int. J. Sci. Eng. Appl.*, 2025, doi: 10.7753/ijsea1403.1006.

[15] F. Alarfaj and S. Shahzadi, "Enhancing Fraud Detection in Banking With Deep Learning: Graph Neural Networks and Autoencoders for Real-Time Credit Card Fraud Prevention," *IEEE Access*, vol. 13, pp. 20633–20646, 2025, doi: 10.1109/access.2024.3466288.

[16] M. Aich, V. Oggu, M. Sankaran, K. K, and Z. Sataar, "An Integrated Temporal Graph Neural Network and Autoencoder Model for Real-Time Credit Card Fraud Detection," in *2025 2nd International Conference on Intelligent Algorithms for Computational Intelligence Systems (IACIS)*, 2025, pp. 1–6. doi: 10.1109/iacis65746.2025.11210970.

[17] A. Khalid, N. Owoh, O. Uthmani, M. Ashawa, J. Osamor, and J. Adejoh, "Enhancing Credit Card Fraud

Detection: An Ensemble Machine Learning Approach," *Big Data Cogn. Comput.*, vol. 8, p. 6, 2024, doi: 10.3390/bdcc8010006.

[18] I. Mienye, E. Esenogho, and C. Modisane, "Detecting Imbalanced Credit Card Fraud via Hybrid Graph Attention and Variational Autoencoder Ensembles," *Appl. Math.*, 2025, doi: 10.3390/appliedmath5040131.

[19] L. Bonde and A. Bichanga, "Improving Credit Card Fraud Detection with Ensemble Deep Learning-Based Models: A Hybrid Approach Using SMOTE-ENN," *J. Comput. Theor. Appl.*, 2025, doi: 10.62411/jcta.12021.

[20] E. Ileberi and Y. Sun, "A Hybrid Deep Learning Ensemble Model for Credit Card Fraud Detection," *IEEE Access*, vol. 12, pp. 175829–175838, 2024, doi: 10.1109/access.2024.3502542.

[21] J. Akshay, T. Vinusha, R. Bianca, C. Krishna, and G. Radhika, "Enhancing Credit Card Fraud Detection with Deep Learning and Graph Neural Networks," in *2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 2024, pp. 1–6. doi: 10.1109/icccnt61001.2024.10725042.

[22] S. Shinde and S. Kale, "Unveiling Anomalies in Credit Card Transactions using Autoencoder Neural Networks," *Int. Adv. Res. J. Sci. Eng. Technol.*, 2023.

[23] M. Al-Shabi, "Credit Card Fraud Detection Using Autoencoder Model in Unbalanced Datasets," *J. Adv. Math. Comput. Sci.*, 2019.

[24] C. Huot, S. Heng, T.-K. Kim, and Y. Han, "Quantum Autoencoder for Enhanced Fraud Detection in Imbalanced Credit Card Dataset," *IEEE Access*, 2024.