Journal of Data Analytics and Intelligent Decision-making

Journal homepage: jdaid.qom.ac.ir

University of Qom

An Ensemble Learning Framework for Credit Card Fraud Detection Using Machine Learning and Deep Learning

Zahra Davoodian[a], Ehsan Hajizadeh*[b]

A. Department of Industrial Engineering and Management Systems, Amirkabir University of Technology (Tehran Polytechnics), Hafez St., 15875-4413, Tehran, Iran.
B. Department of Industrial Engineering and Management Systems, Amirkabir University of Technology (Tehran Polytechnics), Hafez St., 15875-4413, Tehran, Iran.

ARTICLEINFO

Keywords:
Fraud detection
Ensemble learning
Credit cards
Machine learning
Deep learning

ABSTRACT

The rapid growth of digital payment systems has heightened the need for accurate and scalable methods to detect credit card fraud. This study evaluates a range of machine learning and deep learning algorithms, including Logistic Regression, Decision Tree, Random Forest, K-Nearest Neighbors (KNN), XGBoost, Convolutional Neural Networks (CNN), Baseline MLP (Multi-Layer Perceptron), and Long Short-Term Memory (LSTM), to identify effective approaches for detecting fraudulent transactions. Based on comparative analysis, Random Forest and LSTM achieved the strongest individual performance, with accuracies exceeding 96%. Building on these findings, a stacking ensemble model was constructed by integrating Random Forest and LSTM as base learners and Logistic Regression as the meta-classifier. The framework incorporates Convolutional Autoencoder (CAE) for feature extraction and Random Undersampling (RUS) with three resampling ratios (1:1, 1:5, and 1:10) to address class imbalance. Experimental results indicated that the ensemble model provided improved predictive accuracy compared to individual algorithms, achieving an accuracy of 99.98%, a precision of 99.86%, and a recall of 99.89% under a 1:10 resampling ratio. Rather than proposing a new algorithmic architecture, this study contributes to a systematic and unified evaluation of widely used ML and DL approaches and demonstrates the effectiveness of integrating CAE, RUS, and a Random Forest–LSTM stacking ensemble in enhancing fraud detection performance.

* Corresponding author.
E-mail addresses: zahradavoodian79@gmail.com (Z. Davoodian), ehsanhajizadeh@aut.ac.ir (E. Hajizadeh)
Received 2 August 2025; Received in revised form 25 August 2025; Accepted 12 September 2025
Available online 16 September 2025
3115-8161

Cite this article: Davoodian, Z., Hajizadeh, E. (2025). An Ensemble Learning Framework for Credit Card Fraud Detection Using Machine Learning and Deep Learning *Journal of Data Analytics and Intelligent Decision-making*, 1(3), 31-55.
https://doi.org/10.22091/jdaid.2025.14411.1016

1. Introduction

Digital transformation has significantly reshaped financial transactions in recent years, enabling customers to conduct payments through credit cards, online banking, and mobile systems with unprecedented convenience (Madan et al., 2022). As digital payment infrastructures expand, fraudulent credit card activities have grown in both frequency and complexity, driven in part by remote authentication vulnerabilities and the ease with which card information can be misused without physical access (Van Vlasselaer et al., 2015). Credit card fraud now represents one of the most prevalent forms of identity theft, causing substantial financial losses for consumers and institutions worldwide (Afriyie et al., 2023). These challenges underscore the necessity for automated, data-driven fraud detection systems that are capable of adapting to evolving fraud patterns in near real time.

Machine learning (ML) and deep learning (DL) techniques have become integral to modern fraud detection due to their ability to identify hidden patterns in large, complex transactional datasets (Alarfaj et al., 2022; Błaszczyński et al., 2021; Charizanos et al., 2024; Forough & Momtazi, 2021, Roseline et al., 2022). A wide range of supervised learning models, such as logistic regression, random forest, K-nearest neighbors, XGBoost, support vector machines, and various neural network architectures, have been explored for classifying transactions as fraudulent or legitimate (Charizanos et al., 2024; Randhawa et al., 2018). More recent studies have incorporated deep learning models, such as CNNs, LSTMs, and autoencoders, to better capture nonlinear relationships and sequential patterns in transaction streams (Alarfaj et al., 2022; Benchaji et al., 2021).

Despite progress in this field, challenges persist, particularly regarding class imbalance and the integration of diverse model types within unified frameworks. Many real-world datasets contain only a small fraction of fraudulent transactions, which can lead to biased model performance. Furthermore, although ensemble learning, including stacking architectures, has shown promise, the literature varies widely in the choice of models, preprocessing methods, and evaluation procedures, making comprehensive comparisons difficult.

To address these gaps, this study conducts a systematic evaluation of widely used ML and DL algorithms under a consistent experimental framework. Based on the strongest-performing models identified in the benchmarking phase, a stacking ensemble integrating Long Short-Term Memory (LSTM) and Random Forest is implemented. The framework additionally incorporates Convolutional Autoencoder (CAE) for feature extraction and Random Undersampling (RUS) with multiple ratios (1:1, 1:5, 1:10) to examine how resampling intensity affects performance. Rather than claiming the introduction of a completely new approach, this study extends existing research by combining established methods within a unified pipeline and analyzing their comparative and integrated performance on a real-world dataset.

The contributions of this research are threefold:

(1) A comprehensive comparison of common machine learning and deep learning models for credit card fraud detection using consistent preprocessing and evaluation methods;

(2) The development of a stacking ensemble model that integrates LSTM and Random Forest, supported by CAE feature extraction and multiple RUS configurations;

(3) An empirical analysis of how different undersampling ratios influence ensemble performance, offering practical insights for fraud detection system design.

The remainder of this paper is structured as follows: Section 2 provides a review of the relevant literature on ML and DL-based fraud detection. Section 3 describes the dataset, preprocessing techniques, and model architectures. Section 4 presents the experimental results and discussion. Section 5 concludes the study and outlines directions for future research.

2. Related Research Studies

There is a need for an in-depth review of all prior studies in the domain of research and scientific inquiry to comprehend the current state of knowledge on the subject under examination. The literature review provides an overview of the research area, offering context for the study and its relevance to the existing body of knowledge.

Some studies introduced machine learning models for credit card fraud detection. Perols (2011) compared the effectiveness of six widely used machine learning as well as statistical and machine learning models in detecting financial statement fraud while accounting for different assumptions related to misclassification costs and the ratios of fraudulent to non-fraudulent firms. The findings reveal, somewhat unexpectedly, that logistic regression and support vector machines (SVM) demonstrate superior performance compared to bagging, artificial neural networks, and stacking (Adewumi & Akinyelu, 2017).

A systematic review on credit card fraud detection, using machine learning methodologies, highlights the integration of multiple algorithms to detect fraudulent transactions and emphasizes the challenges posed by imbalanced datasets, which are a major hurdle in improving detection accuracy (Choudhury et al., 2018). Additionally, a study evaluates data mining techniques for detecting credit card fraud, proposes improved metrics for false negatives, and finds that the support vector machine algorithm is most effective in handling imbalanced datasets (Banerjee et al., 2018).

A research paper investigates and utilizes machine learning algorithms, including hybrid methods with AdaBoost and majority voting, to detect credit card fraud, demonstrating that the majority voting method achieves high accuracy on both public and real-world datasets, even with added noise (Randhawa et al., 2018). Furthermore, recent research has proposed a big data analysis framework alongside various machine learning methods for fraud detection, with the random forest model emerging as the most effective (Patil et al., 2018). Additionally, a paper applies and compares supervised machine learning algorithms and ensemble learning methods to detect credit card fraud using a real-world dataset, identifying key variables for higher accuracy and demonstrating the effectiveness of a super classifier (Dhankhad et al., 2018).

Several studies have investigated new methods of detecting fraud in the financial services industry and analyzing credit card transactions. Among them is one that aimed to develop a new method for fraud detection in real-time transactions by classifying cardholders based on transaction amounts and, then, using the sliding window method to help determine behavior trends. The study discovered that logistic regression, decision tree, and random forest yielded superior results, particularly following the application of SMOTE to balance the dataset (Dornadula & Geetha, 2019). Concurrently, another research study examined eight machine learning methods for identifying credit card fraud. The results revealed that SVM and ANN exhibited the highest performance across three metrics: accuracy, sensitivity, and the area under precision-recall curve (AUPRC) (Makki et al., 2019). In another study, multiple machine

learning and deep learning methods, including KNN, random forest, SVM, autoencoders, CNN, restricted Boltzmann machines (RBM), and deep belief networks (DBN), were benchmarked against European, Australian, and German datasets to detect fraudulent transactions using AUC, MCC, and cost of failure as evaluation metrics (Raghavan & El Gayar, 2019). A study evaluated several machine learning algorithms, including logistic regression, random forest, and support vector machine, on a credit card fraud detection dataset enhanced with the SMOTE oversampling technique and incremental learning, showcasing high precision and recall for fraud classification (Puh & Brkić, 2019).

Several research studies have investigated different methods to enhance credit card fraud detection. Among these, one study explored a range of techniques, including logistic regression, KNN, random forest, naive bayes, multilayer perceptron, AdaBoost, quadrant discriminative analysis, pipelining, and ensemble learning. The findings revealed that the pipelining method demonstrated superior performance across various metrics such as accuracy, precision, recall, F1 score, Matthew's correlation coefficient (MCC), and balanced classification rate (Bagga et al., 2020). Additionally, a spatial-temporal attention-based graph network (STAGN) for credit card fraud detection was proposed, which fuses temporal and location-based transaction graph features learned by a graph neural network, applies spatial-temporal attention on tensor representations, and achieves superior performance compared to existing methods, as demonstrated by empirical evaluations on real-world datasets and validation by domain experts (Cheng et al., 2020). In another study, an intelligent approach for detecting fraud in credit card transactions using an optimized light gradient boosting machine (OLightGBM) was proposed, achieving superior performance in accuracy (98.40%), area under ROC curve (AUC) (92.88%), precision (97.34%), and F1-score (56.95%) compared to other methods, as demonstrated through experiments on real-world datasets containing both fraudulent and legitimate transactions (Taha & Malebary, 2020).

A range of studies investigated the issue of fraud detection and prediction. One of them conducted a systematic review regarding intelligent methods for the detection of fraudulent financial statements to highlight the supremacy of supervised machine learning and data mining techniques, recognizing major issues and gaps and suggesting future research directions, including unsupervised and semi-supervised approaches, bio-inspired methods, and utilizing unstructured textual and audio data (Ashtiani & Raahemi, 2021). Another review implemented logistic regression, Naïve Bayes, and K-nearest neighbor algorithms to detect credit card fraud, finding logistic regression to be the most effective, especially when random under-sampling techniques were applied to the imbalanced datasets (Itoo, 2021). Additionally, another research emphasized the effectiveness of ANNs in credit card fraud detection (Asha & Kr, 2021). Additional studies have shown that incorporating AdaBoost into SVM, Logistic Regression, Random Forest, XGBoost, Decision Tree, and Extra Tree models improves their accuracy in predicting credit card fraud (Ileberi et al., 2021). A new credit card fraud detection system was designed using sequential modeling with attention mechanisms and LSTM deep recurrent neural networks, which improves accuracy by identifying the critical transactions in the sequence. Its effectiveness has been demonstrated through experiments (Benchaji et al., 2021). In property insurance fraud prediction, the random forest model excelled in various metrics, while deep neural network models showed strength in recall (Severino & Peng, 2021).

Subsequent studies further examined various aspects of credit card fraud detection using machine learning techniques. One study addressed class imbalance in datasets and called for further research to enhance predictive accuracy in fraud detection (Madhurya et al., 2022). Another study examined different machine learning models such as support vector machine, bayesian learning, rule-based learning, dempster–shafer theory, transaction aggregation technique, random forest, and XGBoost for fraud detection. The XGBoost model demonstrated

the most effective performance (Tehreem Ashfaq et al., 2022). Another study evaluated supervised methods using real data, highlighting that the best approach for preventing fraudulent credit card transactions during the COVID-19 pandemic was the AllKNN-CatBoost model, which outperformed previous models with an AUC of 97.94%, a recall of 95.91%, and an F1-Score of 87.40%. This was based on the evaluation of 66 machine learning models using a real-world European dataset and stratified K-fold cross-validation (Alfaiz, 2022). Additionally, a study proposed an efficient credit card fraud detection approach using an ensemble classifier based on LSTM neural networks, combined with AdaBoost and a hybrid data resampling method (SMOTE-ENN), demonstrating superior performance with sensitivities and specificities of 0.996 and 0.998, respectively, compared to other algorithms on real-world datasets (Esenogho et al., 2022). Furthermore, a systematic review provided a comprehensive overview of machine learning for financial fraud detection, addressing key points, weaknesses, and future research directions (Ali et al., 2022). Moreover, another research evaluates multiple machine learning models for credit card fraud detection, highlighting the ANN model's exceptional accuracy of 99.96% and the SVM model's AUC of 95.5%. However, it suggests that a hybrid approach, utilizing a federated learning framework with ANN and SVM, offers the best performance by enhancing accuracy while maintaining user data privacy (Bin Sulaiman et al., 2022). Lastly, deep learning models with additional layers for feature extraction were proposed for credit card fraud detection, resulting in improved accuracy, F1 score, and AUC curve compared to existing methods (Alarfaj et al., 2022).

Additionally, one study examined feature extraction and data sampling techniques, concluding that the RUS method, followed by the CAE method, produced the best performance (Salekshahrezaee et al., 2023). Another study presented a deep learning approach using LSTM, GRU, and an MLP neural network within a stacking classification framework, surpassing other machine learning methods with higher sensitivity and superior performance (Domor Mienye & Sun, 2023). Additionally, research on supervised machine learning algorithms indicated that the random forest algorithm is well-suited for credit card fraud detection and prediction, achieving high accuracy and AUC (Afriyie et al., 2023). Moreover, a study modeled credit card fraud detection using various machine learning classifiers and data balancing techniques, with XGBoost achieving the best performance (99% precision and accuracy) when paired with random oversampling on imbalanced datasets (Gupta et al., 2023). In another study, a hybrid feature-selection method combining filter and wrapper methods was proposed in detecting credit card fraud. This process uses information gain with a genetic algorithm wrapper, which is optimized through the geometric mean, thereby improving performance with sensitivity and specificity rates of 0.997 and 0.994, respectively (Mienye & Sun, 2023). Lastly, the ResNeXt-embedded Gated Recurrent Unit model, enhanced with the Jaya optimization algorithm and using SMOTE extensively for data imbalance mitigation, was proposed for real-time financial fraud detection. It has been shown that this approach leads to a 10% to 18% performance improvement over existing algorithms on three authentic financial transaction datasets while keeping high computational efficiency (Almazroi & Ayub, 2023).

One study addressed the growing challenge of fraud in credit card transactions amid increasing card usage and evaluated machine learning techniques for effective detection (Feng & Kim, 2024). In another research, statistical and machine learning models were comparatively analyzed for payment card fraud detection using both public and real transaction records, demonstrating the effectiveness of aggregated features identified by genetic algorithms (Manjeevan Seera et al., 2024). Another study also proposed a new approach by incorporating CNNs with traditional ML algorithms to improve the accuracy of fraud detection and demonstrated better performance compared to prior methods (Ming et al., 2024). Additionally, in another research, innovative techniques were introduced to handle imbalanced datasets in

healthcare fraud detection, highlighting the efficacy of hybrid resampling methods such as SMOTE-ENN (Bounab et al., 2024). Another study compared machine learning models, datasets, and training-testing splits for financial statement fraud detection; it found the ERT model with original-sampling and an 80:10 split to be most effective, offering practical insights for stakeholders (Riskiyadi, 2024). Another research focuses on enhancing credit card fraud detection using ensemble methods such as SVM, KNN, RF, Bagging, and Boosting, addressing data imbalance and real-time processing challenges. The ensemble outperforms traditional methods, emphasizing its potential in advancing fraud detection systems amid evolving fraud techniques (Khalid et al., 2024). The articles emphasize that, particularly for banking and cash handling, the need for speed in processing financial transactions is crucial and that, in future research, enhancements in predictive accuracy and applications to real-world challenges must be pursued. This research focuses on evaluating high-accuracy machine learning and deep learning algorithms, along with common data preprocessing methods, for developing a model with the highest accuracy based on the outcomes.

3. Research Methodology

This section provides information on the dataset, its features, an explanation of the studied algorithms, the proposed model, as well as the evaluation metrics for assessing algorithm performance.

3.1. Fraud Detection Dataset

The dataset used in this research was collected by a financial institution, and due to security concerns, its name remains undisclosed. This dataset is available at the Kaggle website and includes transactions performed by credit card holders, making it publicly accessible. The dataset comprises 1,000,000 transactions, with 87,400 transactions marked as fraudulent. As is evident, this dataset exhibits an imbalance between valid and fraudulent transaction classes, as only 8.74% of transactions are labeled as fraudulent. The dataset consists of seven features and an additional column named 'fraud,' resulting in a total of eight columns. Each of the seven columns has 1,000,000 rows of observations, and all columns contain quantitative variable data. Figure 1 shows the class distribution in this dataset, and Table 1 represents the explanation of the dataset's columns.

Figure 1

Class Distribution of the Dataset

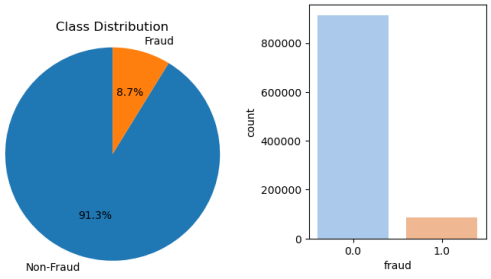


Table 1

Features of the Dataset

Name	Description	Unique data	Number	Number
distance_from_home	The distance from the location where the transaction took place.	1000000	1000000 non-null	float64
distance_from_last_transaction	The distance from the location of the last transaction that has occurred.	1000000	1000000 non-null	float64
ratio_to_median_purchase_price	The transaction price-to-average transaction price ratio.	1000000	1000000 non-null	float64
repeat_retailer	Has the transaction been previously conducted from the same location?	2	1000000 non-null	float64
used_chip	Was the transaction conducted using a credit card?	2	1000000 non-null	float64
used_pin_number	Was the transaction conducted using the card's PIN code?	2	1000000 non-null	float64
online_order	Is this transaction an online order?	2	1000000 non-null	float64
fraud	Is the transaction fraudulent?	2	1000000 non-null	float64

The seven input features in this dataset are engineered behavioral descriptors rather than raw transaction attributes. As summarized in Table 1, they capture (i) spatial information related to cardholder behavior (distance_from_home, distance_from_last_transaction), (ii) transaction pricing behavior (ratio_to_median_purchase_price), and (iii) contextual flags describing the transaction channel and authentication mechanism (repeat_retailer, used_chip, used_pin_number, online_order). The first three variables are continuous, whereas the latter four are binary indicators (0/1). Although the number of features is smaller than in some PCA-transformed benchmark datasets commonly used in the literature, these engineered variables are designed to summarize key aspects of cardholder behavior that are directly relevant to fraud risk, such as unusual location, price deviations, and atypical usage channels.

Before modeling, the dataset was examined for missing values, and none were detected. Continuous variables were then normalized to a common scale, while binary indicators were retained in their original 0/1 form. The data were randomly partitioned into training and test subsets using a stratified sampling strategy to preserve the original proportion of fraudulent and non-fraudulent transactions. On the training set, five-fold cross-validation was employed for model selection and hyperparameter tuning for all classifiers, as discussed in Section 4.

3.2. Machine Learning Algorithms in Credit Card Transaction Fraud Detection

This study evaluates several widely used machine learning models for fraud detection, including XGBoost, logistic regression, decision tree, random forest, and KNN. Each model is implemented using scikit-learn or XGBoost with hyperparameters tuned via cross-validation, as described below.

3.2.1. Decision Tree

Decision tree is a classification method in supervised learning. It takes the form of a tree structure by dividing data into groups based on key features. In this study, the decision tree

model is implemented using the decisiontreeclassifier from scikit-learn. The model was tuned using a grid search over maximum depth values (Alarfaj et al., 2022) and minimum samples per leaf (Alarfaj et al., 2022). The best-performing configuration (max depth = 10, minimum samples leaf = 5) was selected via five-fold cross-validation. The model uses the seven engineered features shown in Table 1 as predictors and the binary fraud indicator column as the target variable.

3.2.2. Logistic Regression

Logistic regression is a method of regression analysis commonly used when the dependent variable has two categories, such as fraud or non-fraud. Logistic regression is used as a baseline linear classifier. It is implemented with the logistic regression class from scikit-learn using the "liblinear" solver for binary classification. Hyperparameters, such as regularization strength C $\in \{0.1, 1, 10\}$, were tuned using five-fold cross-validation. All seven features were standardized before training to ensure numerical stability.

3.2.3. Random Forest

Random forest is a machine learning technique that falls under the category of supervised learning. It employs ensemble learning, using multiple decision tree models for classification and prediction tasks (Akazue et al., 2023). Decision trees in random forest are weak learners, especially when large datasets are used, where the algorithm tends to perform better (Flondor et al., 2024). In order to construct a random forest, a bagging approach is employed, where a set of decision trees is created using randomly selected subsets of the dataset. In this study, random forest is implemented using the random forest classifier from scikit-learn. A grid search was performed over the number of trees {100, 150, 200} and maximum depth {10, 15, None}. The selected configuration consists of 150 trees with a maximum depth of 10. Each tree is trained on a bootstrap sample and considers a random subset of features at each split. Class imbalance is addressed through external resampling (Section 3.5.1), so internal class weighting is not used.

3.2.4. K-Nearest Neighbors (KNN)

The K-nearest neighbors algorithm, on the other hand, constitutes a supervised learning method to observe fraud by the similarity within data points. This might be viewed as representing two transactions at each point using the K-nearest neighbor in feature space; thereafter, the algorithm labels transactions as fraudulent or valid based on the model created from these transactions (Khaled Alarfaj et al., 2022). The K-nearest neighbors classifier is implemented using the KNeighborsClassifier from scikit-learn. Since KNN is sensitive to feature scaling, all continuous variables were standardized before training. The number of neighbors k was tuned in the range {3, 5, 7, 9}, with k = 7 yielding the best validation performance. Euclidean distance is used as the distance metric.

3.2.5. XGBoost

XGBoost is one of the enhanced supervised learning algorithms that has a wide application in fraud detection because of its efficiency in handling imbalanced datasets to find complex patterns. Combination learning creates strong predictive models by iteratively building decision trees and optimizing a specific loss function using gradient boosting. In XGBoost, there are techniques that avoid overfitting and give weight to the unclassified samples, with most of the weight going towards fraud detection. XGBoost is implemented using the XGBClassifier from the XGBoost library. The model is tuned with a grid search over learning rates {0.01, 0.1}, max depths {3, 5, 7}, and number of estimators {100, 200}. The best model uses a learning rate of

0.1, maximum depth of 5, and 200 estimators. Early stopping with 20 rounds is applied to prevent overfitting.

3.3. Deep Learning Algorithms in Credit Card Transaction Fraud Detection

In addition to traditional machine learning algorithms, deep learning algorithms have gained prominence in fraud detection, representing a current trend. These algorithms have gained increased attention in recent years. In this study, the performance of Convolutional Neural Network (CNN), Baseline MLP (Multi-Layer Perceptron), and Long Short-Term Memory (LSTM) was examined on the dataset.

3.3.1. Baseline MLP (Dense Neural Network)

The baseline deep learning model is a simple feedforward neural network (MLP) implemented in Keras. The network consists of an input layer receiving the seven normalized features, followed by two dense layers with 64 and 32 neurons, respectively, each using ReLU activation. A dropout layer with a rate of 0.5 is applied between the dense layers. The output layer uses a sigmoid activation to produce the probability of fraud. The model is trained using the Adam optimizer and binary cross-entropy loss for 35 epochs.

3.3.2. Convolutional Neural Network (1D-CNN)

In this study, a one-dimensional Convolutional Neural Network (1D-CNN) is evaluated as an additional deep learning baseline. Although CNNs are traditionally used for image and text data, recent work has shown that 1D convolutions can be applied to tabular fraud-detection datasets by treating the feature vector as a short sequence. Following this approach, each transaction—originally represented as a 7-dimensional feature vector—is reshaped into an input of shape (7, 1), allowing 1D convolutional filters to learn local interactions among adjacent features.

The architecture used in this study consists of two Conv1D layers with 32 and 64 filters, each followed by batch normalization and dropout to reduce overfitting. After the convolutional block, a flatten layer converts the feature maps into a one-dimensional vector, which is passed through fully connected layers (64 and 32 units, ReLU activation). The final output layer uses a sigmoid activation function to produce a binary fraud probability. The model is trained using the Adam optimizer and binary cross-entropy loss for 100 epochs.

3.3.3. Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) is a unique form of artificial recurrent neural network (RNN), specifically designed for handling time series data in deep learning. It addresses the low accuracy issue of traditional RNNs. Unlike feedforward neural networks, LSTM includes feedback connections between hidden units that are important for individual time steps (Jan, 2021).

In this dataset, transactions are provided as individual records without temporal identifiers or customer IDs that would enable constructing true chronological sequences. To allow the use of an LSTM model for comparison purposes, each 7-dimensional feature vector is reshaped into a pseudo-sequence of length 7 with one feature per time step (shape: 7×1). This representation does not reflect real transaction time series but allows the LSTM to operate as a nonlinear sequence-processing baseline. We acknowledge this limitation and discuss it further in Section 4.

In this study, the LSTM model consists of a single LSTM layer with 32 hidden units followed by a dense layer with a sigmoid activation for binary classification. The model is trained using the Adam optimizer and binary cross-entropy loss for 50 epochs with a batch size of 256. As discussed above, the LSTM processes a pseudo-sequence representation of the

original feature vector, which allows us to include LSTM as a nonlinear deep-learning baseline while acknowledging its limitations for non-temporal data.

3.4. Ensemble Learning

Ensemble learning is a machine learning approach that combines multiple algorithms to enhance classification performance compared to individual models (Brown, 2010). Common weaknesses in machine learning models include high bias, high variance, and low accuracy. Ensemble methods address these issues effectively, especially in handling class imbalances and improving the detection of minority classes (Sun et al., 2021). Ensemble learning techniques are typically categorized into bagging, boosting, and stacking for organizational purposes (Abdul Rehman Khalid, 2024).

This study specifically focuses on stacking, a method where various base algorithms are employed to construct models known as level-0 models. Later, another algorithm, known as a meta-learner, or level-1 classifier, is trained on the task of combining these models' predictions. First, the level-0 models are trained on the resampled training data. Their out-of-fold predictions are then used to form the training set for the level-1 meta-learner. While bagging and boosting rely on combinatorial rules, stacking ensemble learning uses yet another machine learning algorithm, called the meta-learner, to aggregate the predictions of the level-0 models (Mienye & Sun, 2023).

3.5. Proposed Method for Credit Card Fraud Detection

Considering the exploration of various models in machine learning and deep learning, this research, with reference to the paper conducted by Domor Mienye and Sun (2023), proposes the use of two models, LSTM and Random Forest, in an ensemble learning approach such as stacking. This research implements and compares this ensemble learning model with other algorithms.

3.5.1. Data Preprocessing

In financial security, fraud detection in credit cards can block unauthorized transactions. It needs strong fraud detection with gaining momentum in electronic payment systems. Data preprocessing is a fundamental step in any fraud detection model that helps in enhancing the accuracy and involves cleaning, normalization, and dimensionality reduction of data. These steps enable machine learning models to identify meaningful patterns in fraudulent transactions.

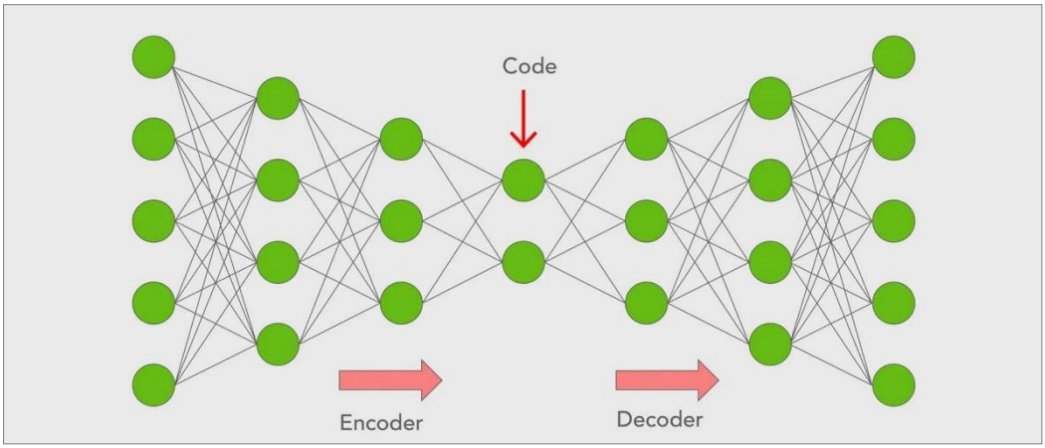
Feature Extraction: In the preprocessing stage, data were cleansed to enable analysis based on complete data. Feature extraction is crucial for fraud detection in transactions, involving the selection and creation of relevant features representing patterns in genuine and fraudulent transactions. In this research, CAE method was used.

CAE, an advanced technique for credit card fraud detection, employs deep learning principles and convolutional neural networks to automatically identify patterns in transactional data. It excels in learning underlying patterns, enhancing fraud detection accuracy, and strengthening financial security (Salekshahrezaee et al., 2023).

The CAE learns a compressed representation of the input features and produces a three-dimensional latent vector that captures nonlinear relationships among the original variables. Its structure includes an encoder compressing input data and a decoder reconstructing data. Nonlinear activation functions like ReLU are crucial for capturing intricate relationships in real-world data. Utilizing the Adam optimizer and mean squared error loss function, this method was trained for 100 epochs on the dataset. Figure 2 illustrates the architecture of autoencoder.

Figure 2

Autoencoder Architecture (Mygreatlearning, 2024)



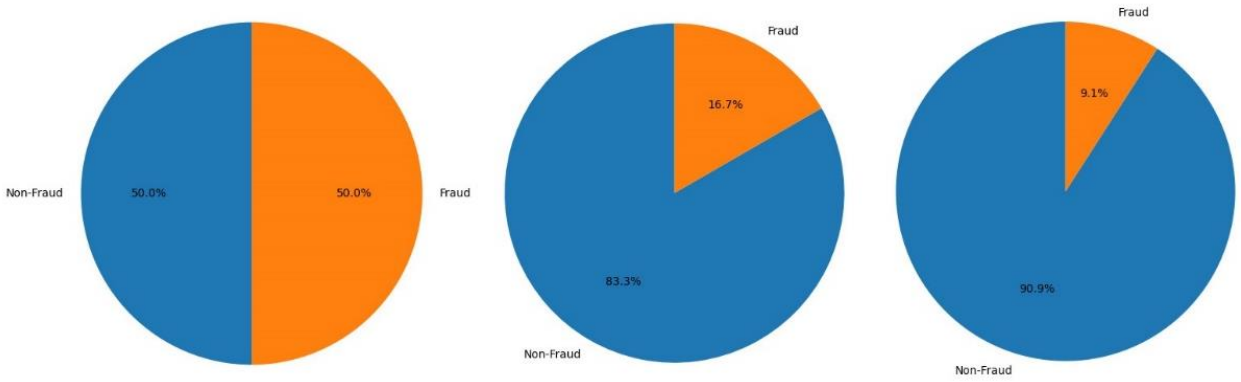
The CAE model consists of an encoder with layers of size [7 → 4 → 3] using ReLU activation, and a symmetric decoder [3 → 4 → 7]. The encoder's three-dimensional latent representation is used as the input feature vector for all ML and DL models. Training is performed for 100 epochs using Adam optimizer (learning rate = 0.001) and mean squared error loss. CAE is trained on the original dataset before resampling.

Data Resampling: Fraud detection is critical in complex financial transactions in order to protect consumers and institutions alike. Resampling techniques such as undersampling are important for any improvement in fraud detection models. Resampling handles the problem of class imbalance either by oversampling the minority class (fraudulent transactions) or by under-sampling the majority class (valid transactions). Specifically, random undersampling (RUS) focuses on randomly choosing a subset of the majority class, aiming for a more balanced distribution between the majority and minority classes.

Despite its ability to alleviate class imbalance issues, RUS has drawbacks, potentially leading to the loss of valuable information in the majority class and reduced model generalization. However, in credit card fraud detection, RUS helps mitigate class imbalance, improving model performance and enhancing the accurate identification of fraudulent transactions (Salekshahrezaee et al., 2023). Similar to the findings of the research conducted by Salekshahrezaee et al. (2023), utilizing RUS in resampling techniques for credit card fraud detection datasets has proven to enhance model accuracy. In this study, RUS was employed with three ratios (1:1, 1:5, and 1:10) on the dataset, and its results on the proposed model will be discussed further. Figure 3, shows the different ratios.

Figure 3

RUS Technique, Applied with Ratios of 1:1, 1:5, and 1:10, Respectively



3.5.2. Proposed Ensemble Model

Referring to the exploration of various machine learning and deep learning models, this study implements an ensemble learning model based on stacking. The ensemble model combines LSTM and random forest in a stacking architecture with two layers, level-0 and level-1. In level-0, base classifiers (LSTM and random forest) are trained and tested on out-of-sample data, forming a new dataset for meta-learner training in level-1.

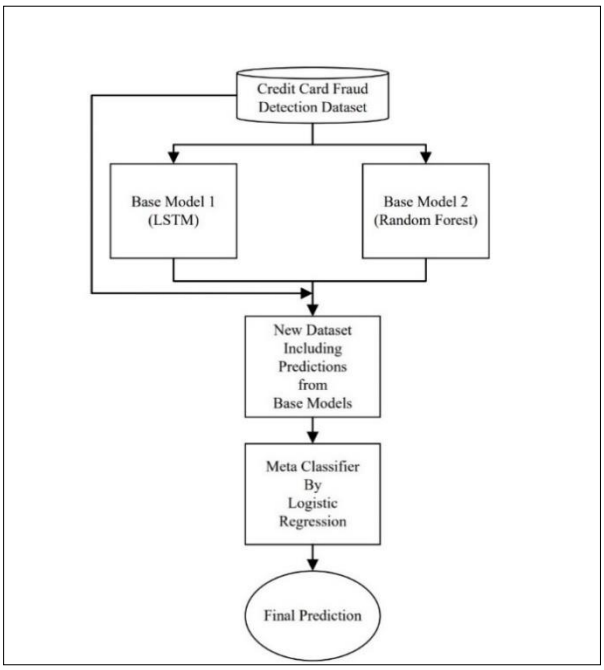
LSTM and random forest are chosen as Level-0 learners due to their strong capabilities in modeling sequential data and high predictive performance. The meta-classifier in Level-1, a logistic regression model, is trained based on the outputs of multiple base models to make predictions (Mienye & Sun, 2023). The use of a meta-classifier leverages the strengths of different base models for more accurate and robust predictions. The logistic regression model takes the predicted probabilities from LSTM and random forest as input features and learns to perform the final classification. Figure 4 shows the proposed model flowchart.

Algorithm of the Proposed Method

Input: Credit card transactions dataset
Procedure:
Step One: Train level-0 models
Step Two: Create a new dataset with probabilities from level-0 models and the original dataset
Step Three: Test the model using meta-classifier and the new dataset
Output: Prediction of data using meta-classifier

Figure 4

The Flowchart of the Proposed Model



In the stacking framework, random forest and LSTM serve as level-0 classifiers. Out-of-fold predicted probabilities are generated using five-fold cross-validation on the training set, ensuring no leakage between training and meta-learning stages. These probabilities form a two-dimensional feature vector for each transaction. A logistic regression meta-classifier is trained on this dataset to produce the final fraud prediction. During inference, the trained level-0 models first generate probabilities for each transaction, which are then passed into the meta-classifier to obtain the final output.

3.6. Evaluation Metrics

In recent decades, machine learning and artificial intelligence have advanced significantly, introducing transformative methods and deep neural networks. Evaluating algorithm performance, particularly in fraud detection, is essential for creating accurate and interpretable models (Afriyie et al., 2023). This section discusses evaluation methods such as confusion matrix and ROC, learning and loss curves.

3.6.1. Confusion Matrix

One common evaluation method is the confusion matrix, which summarizes the relationship between actual and predicted labels (Faraji, 2022). The inputs to this matrix, TP, TN, FP, and FN, represent true positive, true negative, false positive, and false negative, respectively. Table 2 corresponds to the confusion matrix.

Table 2

Confusion Matrix

		Predict	
Actual		Positive (Fraud)	Negative (Normal)

	Positive (Fraud)	True Positive (TP)	False Negative (FN)
	Negative (Normal)	False Positive (FP)	True Negative (TN)

In Table 2, True Positive (TP) means the number of actual fraudulent transactions that are predicted correctly. True Negative (TN) is the count of legitimate transactions that were correctly predicted. False Positive (FP) represents the number of correct, nonfraudulent transactions predicted as fraudulent. While, on the other hand, False Negative (FN) stands for the number of fraudulent transactions predicted as normal.

Due to the data's imbalance, accuracy alone may not provide a comprehensive assessment of fraud data. Hence, precision, recall, and F1 score are utilized to evaluate algorithm performance, computed based on the confusion matrix. Precision quantifies the proportion of accurately detected fraudulent transactions (TP) relative to the total number of predicted fraudulent transactions (TP + FP). Equations 1 and 2 outline the formulations for accuracy and precision, respectively.

$$Accuracy = \frac{TN + TP}{TN + TP + FN + FP} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

Recall or sensitivity, measures the ratio of correctly detected fraudulent transactions (TP) to the total number of fraudulent transactions. The F1 score, an average weighted measure of precision and recall, ranges between 0 and 1, with values close to 1 indicating better performance (Khaled Alarfaj et al., 2022). Equations 3 and 4 define recall and F1 score.

$$Recall\ /\ Sensitivity = \frac{TP}{TP + FN} \tag{3}$$

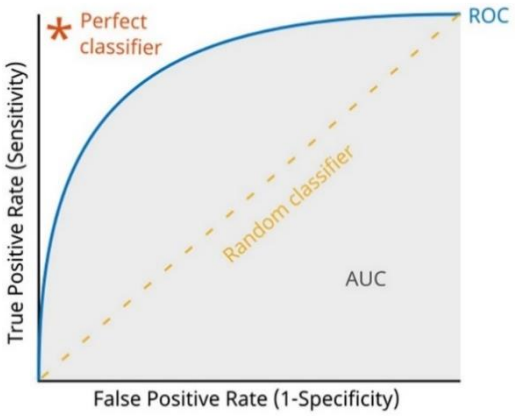
$$F1\ Score = \frac{2 \times precision \times recall}{precision + recall} = \frac{TP}{TP + \frac{1}{2}(FP + FN)} \tag{4}$$

3.6.2. ROC Curve

The ROC curve serves as a vital metric, particularly in fraud detection, illustrating the balance between true positive rate (TPR) and false positive rate (FPR) at different thresholds. As illustrated in Figure 5, the optimal scenario represents a curve reaching the top-left corner, where TPR equals 1, and FPR equals 0. The area under the curve (AUC) provides a comprehensive summary of overall performance, with a value of 1 indicating a flawless classifier and 0.5 suggesting random guessing. In fraud detection, a model with a higher AUC, as evidenced by a curve closer to the top-left corner, effectively discerns between genuine and fraudulent transactions across varying thresholds (Afriyie et al., 2023).

Figure 5

ROC Curve



4. Results and Discussion

After completing preprocessing steps, including feature scaling, CAE-based feature extraction, and random undersampling (RUS) to address class imbalance, the models are trained and evaluated on the dataset.

4.1. Results of Machine Learning Models

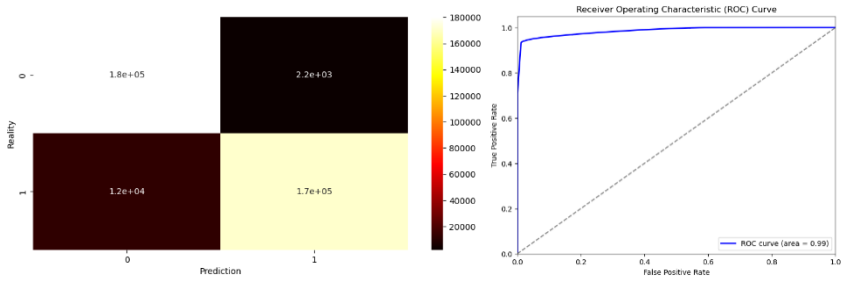
After dividing the dataset into training and testing sets, machine learning algorithms are implemented on the training set. In the decision tree algorithm, a model without specific hyperparameters is primarily created. Then, the model is examined by defining a dictionary with various upper limits for the depth of the decision tree, including values of 5, 10, 15, and None (indicating no depth restriction). Subsequently, a search using grid functions determines the optimal value for the decision tree depth, which is found to be 5. The model is subsequently trained on the training dataset using this optimal depth value.

In the random forest algorithm, the maximum depth of individual decision trees in the forest is set to 10, and the number of iterations for producing repeatable results is set to 5. Similar to the decision tree algorithm, the optimal number of decision trees for this algorithm is determined through grid functions, considering values of 50, 100, 150, and 200, with the optimal value being 150. Subsequently, the model is fitted to the training dataset with this specified number of decision trees.

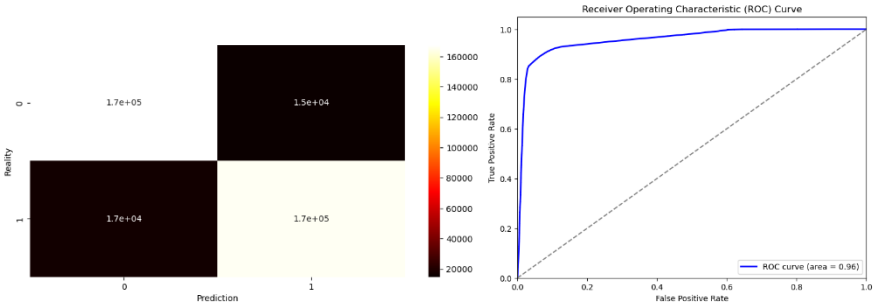
In the KNN algorithm, the optimal number of neighbors, with values ranging from 3 to 9, is determined through grid functions, and the optimal value is found to be 6. Subsequently, the model is fitted to the training dataset with this specified number of neighbors. Figure 6 shows the results for machine learning models of this study. In Figure 7, the accuracy and F1-score of algorithms are compared.

Figure 6

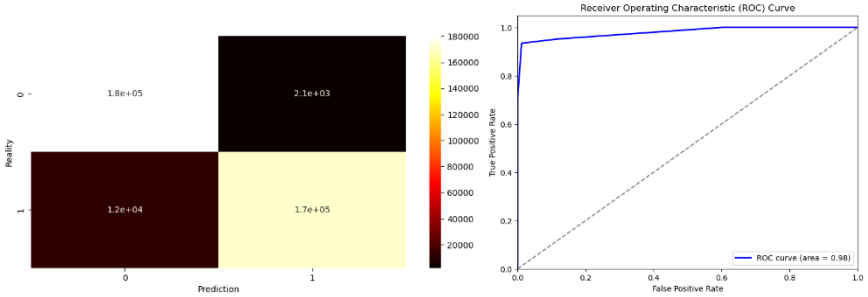
Confusion Matrix and ROC Curve of Studied Algorithms



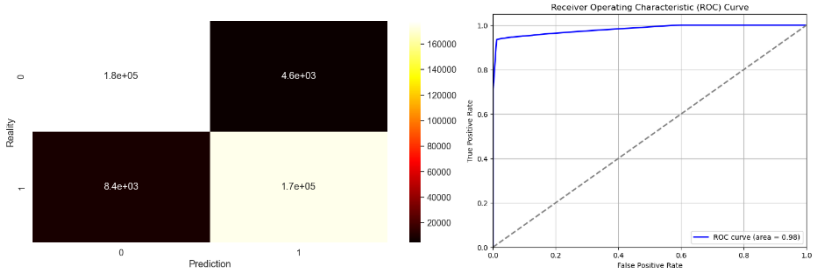
XGBoost



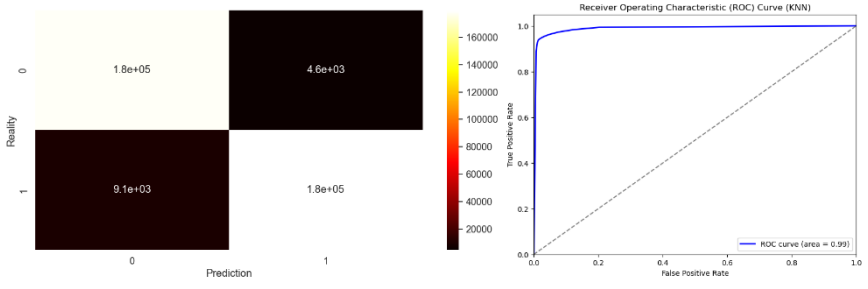
Logistic Regression



Decision Tree



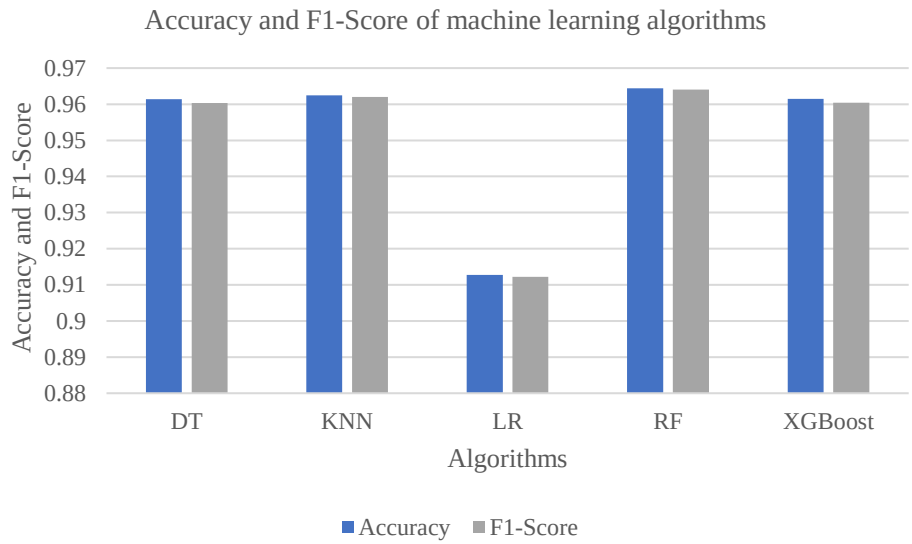
Random Forest



KNN

Figure 7

Comparison of Accuracy and F1-Score of Algorithms



In this study, various machine learning algorithms, including logistic regression, KNN, random forest, decision tree, and XGBoost, have been utilized to develop predictive models. To assess the performance of these models and help prevent overfitting and underfitting, a five-fold cross-validation has been implemented. As shown in Table 3, all machine learning models achieve average accuracies around 96% in five-fold cross-validation. However, given the class imbalance in the dataset, accuracy alone is not sufficient to evaluate fraud detection performance. Therefore, in the subsequent analysis and in the evaluation of the proposed ensemble model, we place greater emphasis on precision, recall, F1-score, and ROC-based metrics.

Table 3

Result of Five-Fold CV

No.	Algorithm Name	Average Accuracy	Standard Deviation of Accuracy
1	Random Forest	0.96142	0.0004
2	KNN	0.96024	0.0003
3	XGBoost	0.96146	0.0004
4	Decision Tree	0.96136	0.0004
5	Logistic Regression	0.91287	0.0008

4.2. Results of Deep Learning Models

Each deep learning model is implemented with a specific architecture designed based on the literature review. As shown in Table 4, the convolutional neural network (CNN) model, following the design outlined in Alarfaj et al. (2022), is implemented with 12 layers and trained for 100 epochs on the dataset. Figure 8 illustrates the learning curve and the loss curve of the CNN model, providing insights into the model's fitness.

In the baseline algorithm, the previous model is extended to accommodate larger batch sizes. The total number of parameters is 2,433, with 2,433 trainable parameters and 0 non-trainable parameters. This algorithm features four layers, utilizes binary cross-entropy as the loss function, and employs the Adam optimizer. As shown in Table 5, this model was

implemented for 35 epochs on the dataset. Figure 9 illustrates the learning curve of the baseline model.

As shown in Table 6, the LSTM algorithm was implemented with three layers: one layer with 15 units for input values, followed by a dropout layer to prevent overfitting, and a dense layer with two units for classifying data as fraudulent or valid. In this algorithm, the total number of parameters is 1,232, the number of trainable parameters is 1,232, and the number of non-trainable parameters is 0. This model has a sparse categorical cross-entropy loss function and uses the Adam optimizer with a learning rate of 0.1. Figure 10 indicates the loss curve of LSTM model. Table 7 compares the accuracy of the three deep learning models. As with the machine learning models, these accuracies are interpreted together with F1-score and other metrics reported in the ensemble evaluation.

Table 4

CNN Architecture

Layer	Output Shape	Parameter Count	Details
Conv1D	(None, 4, 32)	128	Kernel size of 32 × 3 and ReLU activation function
BatchNormalization	(None, 4, 32)	128	
Dropout	(None, 4, 32)	0	Dropout rate of 0.2
Conv1D	(None, 4, 64)	6208	Kernel size of 64 × 3 and ReLU activation function
BatchNormalization	(None, 4, 64)	256	
Dropout	(None, 4, 64)	0	Dropout rate of 0.5
Flatten	(None, 256)	0	ReLU activation function
Dense	(None, 64)	16448	
Dropout	(None, 64)	0	Dropout rate of 0.5
Dense	(None, 32)	2080	ReLU activation function (100)
Dense	(None, 25)	825	ReLU activation function (50)
Dense	(None, 1)	26	ReLU activation function (25)

Table 5

Baseline MLP Architecture

Layer	Output Shape	Parameter Count	Details
Dense	(None, 64)	320	ReLU activation function
Dropout	(None, 64)	0	Dropout rate of 0.5
Dense	(None, 32)	2080	ReLU activation function
Dense	(None, 1)	33	Sigmoid activation function

Table 6

LSTM Architecture

Layer	Output Shape	Parameter Count	Details
LSTM	(None, 15)	1380	
Dropout	(None, 15)	0	Dropout rate of 0.2
Dense	(None, 2)	32	Softmax activation function

Figure 8

CNN Model's Learning Curve and Loss Curve

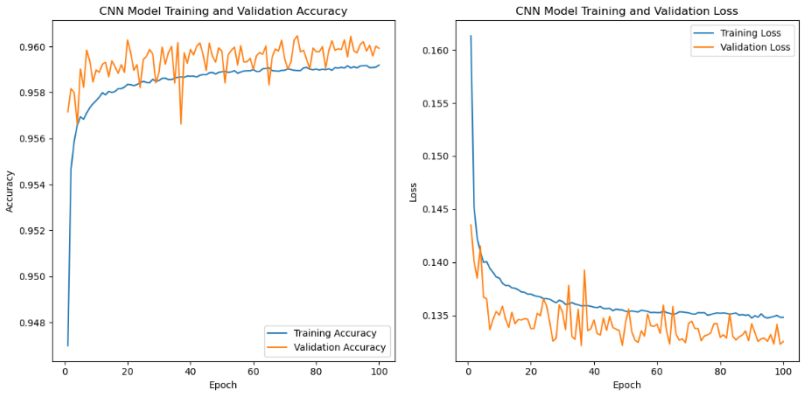


Figure 9
Baseline MLP's Learning Curve

Figure 10
LSTM Model's Loss Curve

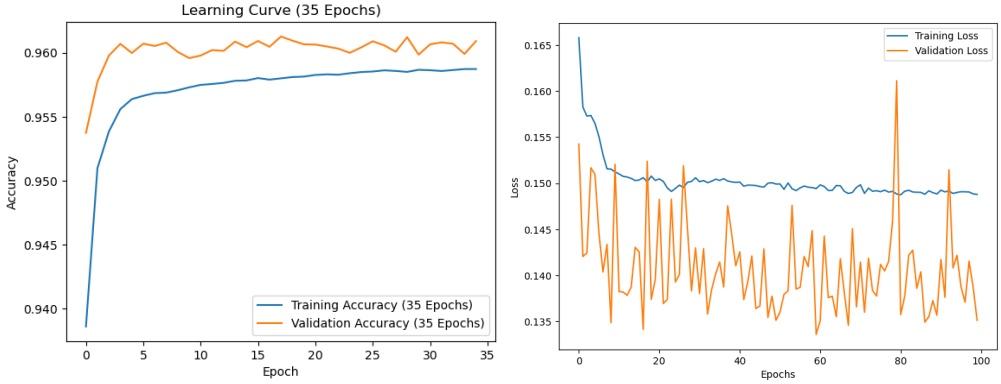


Table 7

Comparison of Models' Accuracy

No.	Algorithm	Accuracy
1	LSTM	0.9603
2	BL	0.9602
3	CNN	0.9599

4.3. Results of the Proposed Model

Based on study results, LSTM and random forest, known for their high accuracy, were chosen as baseline models. RUS and CAE methods were employed for data preprocessing, enhancing model performance. The RUS method was implemented with three different ratios (1:10, 1:5, 1:1), and the CAE method was used to derive a three-dimensional latent representation of the original seven features for model training. The proposed model aims to improve fraud detection by combining LSTM and random forest algorithms. The models were trained on the dataset, and the predicted class results, along with the actual classes, were used as input for training the logistic regression algorithm, and this algorithm was utilized to predict and present the final output. In this section, Figures 11, 12, and 13 display the curves and confusion matrix for each ratio, while Tables 8, 9, and 10 present the corresponding results.

Figure 11

Loss and Learning Curve of LSTM Algorithm and Confusion Matrix of Proposed Model with a 1:10 Ratio in Resampling

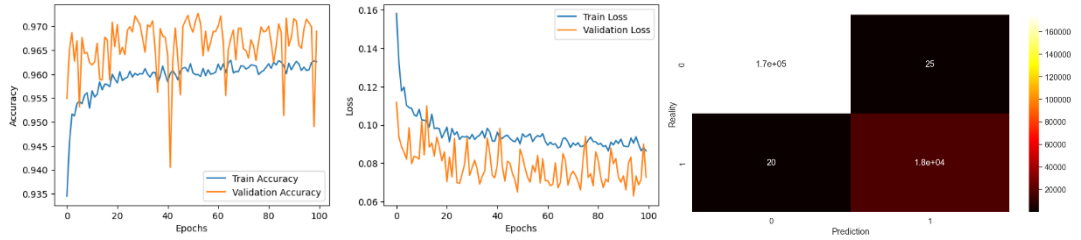


Figure 12

Loss and Learning curve of LSTM algorithm and Confusion Matrix of Proposed Model with a 1:5 Ratio in Resampling

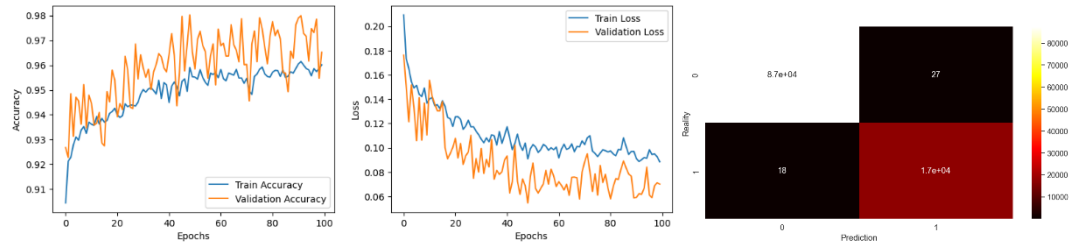


Figure 13

Loss and Learning Curve of LSTM Algorithm and Confusion Matrix of Proposed Model with a 1:1 Ratio in Resampling

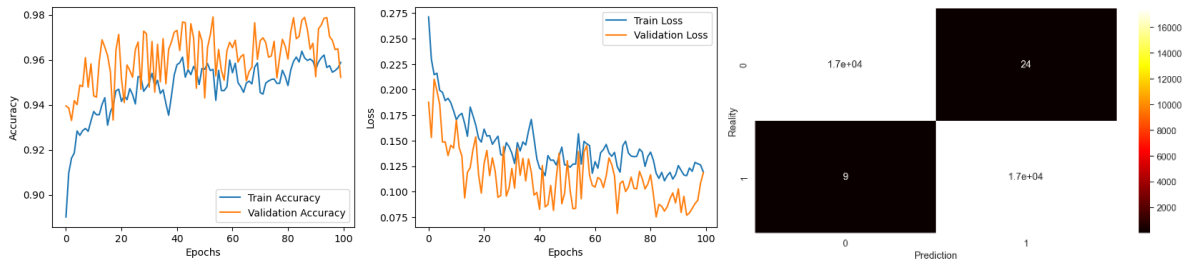


Table 8

Results of the Proposed Model for 1:10 Resampling Ratio

Evaluation Metric/Model Name	LSTM	Random Forest	Proposed Model
Accuracy	0.9696	0.9970	0.9998
Precision	0.9565	0.9951	0.9986
Recall	0.6991	0.9942	0.9989
F1-Score	0.8078	0.9956	0.9987

Table 9

Results of the Proposed Model for 1:5 Resampling Ratio

Evaluation Metric/Model Name	LSTM	Random Forest	Proposed Model
Accuracy	0.9641	0.9965	0.9996
Precision	0.8885	0.9945	0.9984
Recall	0.8964	0.9950	0.9990
F1-Score	0.8924	0.9947	0.9987

Table 10

Results of the Proposed Model for 1:1 Resampling Ratio

Evaluation Metric/Model Name	LSTM	Random Forest	Proposed Model
Accuracy	0.9507	0.9960	0.9991
Precision	0.9466	0.9940	0.9986
Recall	0.9553	0.9945	0.9995
F1-Score	0.9509	0.9963	0.9991

Given the very high performance values obtained, particularly for the proposed ensemble model (e.g., an accuracy of 99.98% and an F1-score above 0.998 for the 1:10 ratio), we performed additional checks to rule out common sources of overestimation, such as data leakage or improper train–test splitting. The dataset was first split into training and test sets using a stratified procedure, with all resampling operations (RUS) and model training performing exclusively on the training set. The test set was kept completely separate and used only once for final evaluation. Furthermore, the dataset does not include explicit cardholder identifiers, so transactions cannot be grouped by user; however, we verified that no duplicate rows appear across the training and test partitions. These steps help ensure that the reported performance reflects generalization to unseen data rather than information leakage.

The aim of the proposed stacking model is to improve fraud detection performance over individual classifiers by combining LSTM and random forest through a logistic regression meta-learner. As shown in Tables 8 to 10, the ensemble model not only increases overall accuracy but also, more importantly, achieves very high precision, recall, and F1-score across all three RUS ratios. This indicates that the model is able to correctly identify the vast majority of fraudulent transactions while keeping the rate of false alarms low.

In the stacking framework, the meta-classifier effectively assigns more influence to the random forest component, which exhibits the strongest individual performance among the base learners. Table 11 further reports five-fold cross-validation results for the proposed model and random forest, showing consistently high performance with very low standard deviations, suggesting stable behavior across different data splits.

4.4. Discussions

The proposed model, combining random forest and LSTM, achieved strong fraud detection performance across all evaluation metrics. Among the three RUS ratios, the 1:10 configuration yields the highest overall performance, with particularly strong recall and F1-score for the fraud class, while maintaining a high precision. The low standard deviations reported in Table 11 indicate that the ensemble's behavior is stable across different cross-validation folds.

Table 11

Result of Five-Fold CV for the Proposed Model

Algorithm Name	Random Forest	Proposed Model

Resampling Ratio	Average Accuracy	Accuracy Standard Deviation	Average Accuracy	Accuracy Standard Deviation
1:10	0.999619	0.000140	0.999765	0.000053
1:5	0.999461	0.000196	0.999570	0.000134
1:1	0.998670	0.000248	0.999056	0.000145

5. Conclusion

The increasing threat of credit card fraud poses a persistent challenge for financial institutions, particularly in environments where transaction volumes are high and fraudulent behavior evolves rapidly. This study evaluated a range of machine learning and deep learning methods for credit card transaction fraud detection on an imbalanced real-world dataset, and proposed an ensemble learning framework designed to improve detection performance on the minority (fraud) class.

Across the benchmark models, random forest emerged as the strongest machine learning classifier, achieving an average accuracy above 96%, with a ROC-AUC of 0.98. Among the deep learning models, the LSTM-based network showed competitive performance, with an accuracy of 96.03% on the test set. Building on these results, the proposed stacking ensemble, combining LSTM and random forest as level-0 learners with a logistic regression meta-classifier, achieved very high performance under different random undersampling (RUS) ratios. In particular, for the 1:10 resampling configuration, the ensemble obtained an accuracy of 0.9998, a precision of 0.9986, a recall of 0.9989, and an F1-score of 0.9987, indicating that it can correctly identify the vast majority of fraudulent transactions while maintaining a low false-alarm rate.

Despite these promising results, the study has several limitations. The experiments are conducted on a single publicly available dataset with seven engineered behavioral features, and the lack of cardholder identifiers prevents true sequence modeling of user-level transaction histories. The LSTM model, therefore, operates on a pseudo-sequential representation rather than genuine temporal data, which should be considered when interpreting its contribution. Moreover, the very high performance metrics reported for the ensemble model warrant further validation on additional datasets and in real-world deployment settings to fully confirm its robustness and generalizability.

Future research can extend this work in several directions. First, combining more diverse base learners, such as support vector machines, gradient boosting variants, or attention-based deep architectures, may further enhance ensemble diversity and performance. Second, a more detailed exploration of feature engineering, including the integration of temporal and customer-level information when available, could provide richer representations for fraud detection models. Finally, investigating risk-sensitive decision thresholds and cost-aware evaluation, with a particular focus on reducing false positives while maintaining high recall, remains a critical avenue for improving the practical utility of fraud detection systems in operational environments.

REFERENCES

Abdul Rehman Khalid, N. O., Omair Uthmani, M., Ashawa, M., Osamor, J., & Adejoh, J. (2024). Enhancing credit card fraud detection: An ensemble machine learning approach. *Big Data and Cognitive Computing, 8*(1), 27. https://doi.org/10.3390/bdcc8010006

Adewumi, A. O., & Akinyelu, A. A. (2017). A survey of machine-learning and nature-inspired based credit card fraud detection techniques. *International Journal of System Assurance Engineering and Management, 8*, 937–953.

Afriyie, J. K., Tawiah, K., Pels, W. A., Addai-Henne, S., Dwamena, H. A., Owired, E. O., Ayeh, S. A., & Eshun, J. (2023). A supervised machine learning algorithm for detecting and predicting fraud in credit card transactions. *Decision Analytics Journal*, *6*, 100163.

Akazue, M. I., Debekeme, I. A., Edje, A. E., Asuai, C., & Osame, U. J. (2023). Unmasking fraudsters: ensemble features selection to enhance random forest fraud detection. *Journal of Computing Theories and Applications, 1*(2), 201-211.

Alarfaj, F. K., Malik, I., Khan, H. U., Almusallam, N., Ramzan, M., & Ahmed, M. (2022). Credit card fraud detection using state-of-the-art machine learning and deep learning algorithms. *IEEE Access, 10*, 39700–39715.

Alfaiz, N. S., & Fati, S. M. (2022). Enhanced credit card fraud detection model using machine learning. *Electronics, 11*(4), 662.

Ali, A., Razak, S. A., Othman, S. H., Abdalla Elfadil Eisa, T., Al-Dhaqm, A., Nasser, M., Elhassan, T., Elshafie, H., & Saif, A. (2022). Financial fraud detection based on machine learning: A systematic literature review. *Applied Sciences, 12*, 9637.

Almazroi, A. A., & Ayub, N. (2023). Online payment fraud detection model using machine learning techniques. Ieee Access, 11, 137188-137203.

Asha, R., & Kr, S. K. (2021). Credit card fraud detection using artificial neural network. *Global Transitions Proceedings, 2*, 35–41.

Ashtiani, M. N., & Raahemi, B. (2021). Intelligent fraud detection in financial statements using machine learning and data mining: a systematic literature review. *Ieee Access, 10*, 72504-72525.

Bagga, S., Goyal, A., Gupta, N., & Goyal, A. (2020). Credit card fraud detection using pipelining and ensemble learning. *Procedia Computer Science, 173*, 104–112.

Banerjee, R., Chen, S., Kashyap, M., & Purohit, S. (2018). Comparative analysis of machine learning algorithms through credit card fraud detection. In *2018 IEEE MIT Undergraduate Research Technology Conference (URTC)* (pp. 1–5). Cambridge, MA: IEEE.

Benchaji, I., Douzi, S., & El Ouahidi, B. (2021). Credit card fraud detection model based on LSTM recurrent neural networks. *Journal of Advances in Information Technology, 12*.

Benchaji, I., Douzi, S., El Ouahidi, B., & Jaafari, J. (2021). Enhanced credit card fraud detection based on attention mechanism and LSTM deep model. *Journal of Big Data*, *8*(1), 151.

Bin Sulaiman, R., Schetinin, V., & Sant, P. (2022). Review of machine learning approach on credit card fraud detection. *Human-Centric Intelligent Systems*, *2*(1), 55-68.

Błaszczyński, J., De Almeida Filho, A. T., Matuszyk, A., Szeląg, M., & Słowiński, R. (2021). Auto loan fraud detection using dominance-based rough set approach versus machine learning methods. *Expert Systems with Applications, 163*, 113740.

Bounab, R., Zarour, K., Guelib, B., & Khlifa, N. (2024). Enhancing medicare fraud detection through machine learning: Addressing class imbalance with SMOTE-ENN. *IEEE Access, 12*, 54382–54396.

Brown, G. (2010). Ensemble learning. In C. Sammut & G. I. Webb (Eds.), *Encyclopedia of Machine Learning* (pp. 1-5). Springer.

Charizanos, G., Demirhan, H., & İcen, D. (2024). An online fuzzy fraud detection framework for credit card transactions. *Expert Systems with Applications, 252*, 124127.

Cheng, D., Wang, X., Zhang, Y., & Zhang, L. (2020). Graph neural network for fraud detection via spatial-temporal attention. *IEEE Transactions on Knowledge and Data Engineering, 34*, 3800–3813.

Choudhury, T., Dangi, G., Singh, T. P., Chauhan, A., & Aggarwal, A. (2018). An efficient way to detect credit card fraud using machine learning methodologies. In *2018 Second International Conference on Green Computing and Internet of Things (ICGCIoT)* (pp. 591–597). IEEE.

Dhankhad, S., Mohammed, E., & Far, B. (2018, July). Supervised machine learning algorithms for credit card fraudulent transaction detection: a comparative study. In *2018 IEEE international conference on information reuse and integration (IRI)* (pp. 122-125). IEEE.

Domor Mienye, I., & Sun, Y. (2023). A deep learning ensemble with data resampling for credit card fraud detection. *IEEE Access, 11*, 30628–30638.

Dornadula, V. N., & Geetha, S. (2019). Credit card fraud detection using machine learning algorithms. Procedia computer science, 165, 631-641.

Esenogho, I. D. M., Swart, T. G., Aruleba, K., & Obaido, G. (2022). A neural network ensemble with feature engineering for improved credit card fraud detection. *IEEE Access, 10*, 16400–16407.

Faraji, Z. (2022). A review of machine learning applications for credit card fraud detection with a case study. *SEISENSE Journal of Management, 5*, 49–59.

Feng, X., & Kim, S. K. (2024). Novel machine learning based credit card fraud detection systems. *Mathematics, 12*(12), 1869.

Flondor, E., Donath, L., & Neamtu, M. (2024). Automatic card fraud detection based on decision tree algorithm. *Applied Artificial Intelligence, 38*, 2385249.

Forough, J., & Momtazi, S. (2021). Ensemble of deep sequential models for credit card fraud detection. *Applied Soft Computing, 99*, 106883.

Great Learning Editorial Team (2024, December 3). Introduction to autoencoders? What are autoencoders applications and types? *Mygreatlearning*. https://www.mygreatlearning.com/blog/autoencoder/

Gupta, P., Varshney, A., Khan, M. R., Ahmed, R., Shuaib, M., & Alam, S. (2023). Unbalanced credit card fraud detection data: A machine learning-oriented comparative study of balancing techniques. *Procedia Computer Science*, *218*, 2575-2584.

Ileberi, E., Sun, Y., & Wang, Z. (2021). Performance evaluation of machine learning methods for credit card fraud detection using SMOTE and AdaBoost. *IEEE Access*, *9*, 165286–165294.

Itoo, F., Meenakshi, & Singh, S. (2021). Comparison and analysis of logistic regression, Naïve Bayes and KNN machine learning algorithms for credit card fraud detection. *International Journal of Information Technology*, *13*, 1503–1511.

Jan, C.-L. (2021). Detection of financial statement fraud using deep learning for sustainable development of capital markets under information asymmetry. *Sustainability*, *13*, 20.

Khaled Alarfaj, F., Malik, I., Khan, H. U., Almusallam, N., Ramzan, M., & Ahmed, M. (2022). Credit card fraud detection using state-of-the-art machine learning and deep learning algorithms. *IEEE Access*, *10*, 39700–39715.

Khalid, A. R., Owoh, N., Uthmani, O., Ashawa, M., Osamor, J., & Adejoh, J. (2024). Enhancing credit card fraud detection: An ensemble machine learning approach. *Big Data and Cognitive Computing*, *8*, 6.

Madan, S., Sofat, S., & Bansal, D. (2022). Tools and techniques for collection and analysis of Internet-of-Things malware: A systematic state-of-art review. *Journal of King Saud University-Computer and Information Sciences*, *34*, 9867–9888.

Madhurya, M. J., Gururaj, H. L., Soundarya, B. C., Vidyashree, K. P., & Rajendra, A. B. (2022). Exploratory analysis of credit card fraud detection using machine learning techniques. *Global Transitions Proceedings*, *3*, 31–37.

Makki, S., Assaghir, Z., Taher, Y., Haque, R., Hacid, M.-S., & Zeineddine, H. (2019). An experimental study with imbalanced classification approaches for credit card fraud detection. *IEEE Access*, *7*, 93010–93022.

Manjeevan Seera, C. P. L., Ajay Kumar, Lalitha Dhamotharan, & Kim Hua Tan. (2024). An intelligent payment card fraud detection system. *Annals of Operations Research*, *334*, 445–467.

Mienye, I. D., & Sun, Y. (2023). A machine learning method with hybrid feature selection for improved credit card fraud detection. *Applied Sciences*, *13*(12), 7254.

Ming, R., Abdelrahman, O., Innab, N., & Ibrahim, M. H. K. (2024). Enhancing fraud detection in auto insurance and credit card transactions: A novel approach integrating CNNs and machine learning algorithms. *PeerJ Computer Science*, *10*, e2088.

Patil, S., Nemade, V., & Soni, P. K. (2018). Predictive modelling for credit card fraud detection using data analytics. *Procedia computer science*, *132*, 385-395.

Perols, J. (2011). Financial statement fraud detection: An analysis of statistical and machine learning algorithms. *Auditing: A Journal of Practice & Theory*, *30*, 19–50.

Puh, M., & Brkić, L. (2019). Detecting credit card fraud using selected machine learning algorithms. In *2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)* (pp. 1250–1255). IEEE.

Raghavan, P., & El Gayar, N. (2019). Fraud detection using machine learning and deep learning. In *2019 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE)* (pp. 334–339). IEEE.

Randhawa, K., Loo, C. K., Seera, M., Lim, C. P., & Nandi, A. K. (2018). Credit card fraud detection using AdaBoost and majority voting. *IEEE Access*, *6*, 14277–14284.

Riskiyadi, M. (2024). Detecting future financial statement fraud using a machine learning model in Indonesia: a comparative study. *Asian Review of Accounting*, *32*(3), 394-422.

Roseline, J. F., Naidu, G., Pandi, V. S., Alias Rajasree, S. A., & Mageswari, N. (2022). Autonomous credit card fraud detection using machine learning approach. *Computers and Electrical Engineering*, *102*, 108132.

Salekshahrezaee, Z., Leevy, J. L., & Khoshgoftaar, T. M. (2023). The effect of feature extraction and data sampling on credit card fraud detection. *Journal of Big Data*, *10*(1), 6.

Severino, M. K., & Peng, Y. (2021). Machine learning algorithms for fraud prediction in property insurance: Empirical evidence using real-world microdata. *Machine Learning with Applications*, *5*, 100074.

Sun, Y., Li, Z., Li, X., & Zhang, J. (2021). Classifier selection and ensemble model for multi-class imbalance learning in education grants prediction. *Applied Artificial Intelligence*, *35*, 290–303.

Taha, A. A., & Malebary, S. J. (2020). An intelligent approach to credit card fraud detection using an optimized light gradient boosting machine. *IEEE Access*, *8*, 25579–25587.

Tehreem Ashfaq, R. K., Yahaya, A. S., Aslam, S., Azar, A. T., Alsafari, S., & Hameed, I. A. (2022). A machine learning and blockchain based efficient fraud detection mechanism. *Sensors*, *22*, 7162.

Van Vlasselaer, V., Bravo, C., Caelen, O., Eliassi-Rad, T., Akoglu, L., Snoeck, M., & Baesens, B. (2015). APATE: A novel approach for automated credit card transaction fraud detection using network-based extensions. *Decision Support Systems, 75*, 38–48