Science Journal of University of Zakho
DOI: https://doi.org/10.25271/sjuoz.2026.14.3.1882

Original Article

COMPARATIVE ANALYSIS OF CNN FOR FINANCIAL FRAUD DETECTION ON TABULAR TRANSACTION DATA

Sardar Hasen Ali[1], Abdulmajeed Adil Yazdeen[2], Rozin Majeed Abdullah[3], Mohammed Hashim Younis[4], Riyadh Qashi[5,*]

[1]Computer science Department, College of Science, University of Zakho, Zakho 42002, Kurdistan Region, Iraq.
[2]Information Technology Management Department, Technical College of Administration, Duhok Polytechnic University, Duhok, Kurdistan Region, Iraq.
[3]Highway and Bridges Engineering Department, Technical College of Engineering, Duhok Polytechnic University, Duhok, Kurdistan Region, Iraq.
[4]Information Technology Department, Technical College of Informatics Akre, Akre University for Applied Sciences, Akre, Kurdistan Region, Iraq.
[5]Vocational School Center 7 Electrical Engineering of the City of Leipzig, Germany.

[*]*Corresponding author, E-mail:*riyadh.qashi@hotmail.com (Tel: +49-1797526920)

Received
January 20, 2026

Accepted
April 22, 2026

Published
July 7, 2026

ABSTRACT

Financial Fraud threatens global financial stability gravely. Fraud comes with important economic losses and has negative effects on public trust. Traditional rule-based systems and older methods of machine learning cannot detect these patterns of fraud, especially when facing complex or continuously changing fraud patterns in the era of high-speed and high-volume transactions which motivates comparing traditional models to deep learning approaches like Convolutional Neural Network (CNN). This research proposes a deep learning technique for the detection of financial fraud by using Convolutional Neural Networks on reshaped tabular transaction data. The proposed CNN model utilizes spatial feature extraction by reprocessing 1D financial records into 2D matrices. This method aims in recognizing local relationships among features as indicative of fraudulent activity allowing comparison with baseline models like Random Forest (RF) and Logistic Regression (LR). Preprocessing steps for this proposed system include class balancing, scaling of features, and one-hot encoding. It is tested on a highly imbalanced credit card fraud dataset from Kaggle containing 284,807 transactions, of which 492 (0.172%) are fraudulent. The CNN achieved 99.23% accuracy, 98.6% precision, 97.9% recall, an F1-score of 98.25%, and an AUC-ROC of 0.992. Comparative analysis with baseline models shows RF achieving 99.97% accuracy and LR achieving 99.91% accuracy highlighting the trade-off between the slightly higher accuracy of traditional models and the adaptability of CNN. Five-fold stratified cross-validation confirmed model robustness with mean accuracy of 99.18% ± 0.11% confirming CNN's stability when compared with baseline models under the same validation procedure. These results indicate that the CNN model can capture the hidden patterns of fraud well when compared to traditional methods. Furthermore, they are less dependent on manual feature engineering and adaptive to evolving fraud strategies. Overall, while traditional models achieve slightly higher accuracy, CNN provides advantages in automated feature extraction, adaptability to changing fraud patterns, and computational efficiency for real-time deployment, making it a valuable alternative depending on operational priorities.

Keywords: Deep Learning, Feature Reshaping, Financial Cyber-security, Convolutional Neural Networks.

1. INTRODUCTION

Financial fraud is a common, sophisticated, and steadily increasing threat to individuals, organizations, and governmental institutions across the world. It encompasses incidents such as credit card scams, identity theft, money laundering, and fraudulent claims for insurance or loans (Ngai *et al*., 2011). According to the Association of Certified Fraud Examiners (ACFE), companies lose about 5% of annual revenue to fraud, amounting to trillions of dollars globally (ACFE, 2022). Specifically, credit card-related fraud was expected to exceed $32 billion in 2021 (Nilson, 2019). Besides the financial loss, there is also the issue of reputational damage, legal liability, and the erosion of trust. The traditional approaches to anti-fraud, rule-based and simple statistical approaches, have difficulties in dealing with new, complex, and dynamic fraud (West & Bhattacharya, 2016). The RF algorithm is widely used because of the good results it achieves in terms of accuracy in handling data in table format. However, it requires manual feature engineering and may not be able to adapt quickly to new and dynamic fraud. Deep learning with CNN is able to automatically learn

Access this article online

DOI:https://doi.org/10.25271/sjuoz.2026.14.3.1882

Science Journal of University of Zakho
Vol. 14, No. 03, pp. 494-503, July -2026

Printed ISSN 2663-628X;
Electronic ISSN 2663-6298

features and identify complex patterns. A comparison of CNN with RF and LR in terms of accuracy, robustness, and adaptability is proposed. The speed and large volume of transactions are also a problem for the conventional approaches.

In recent years, deep learning (DL) has shown significant potential in fraud detection due to its ability to understand complex datasets and reveal hidden linkages (Fiore *et al*., 2019). While CNNs are traditionally applied to visual data, they can also be adapted to structured financial data. By reshaping tabular transaction data appropriately, CNNs can detect critical spatio-temporal and hierarchical patterns that are often overlooked by classical machine learning models. This capability is particularly useful for identifying complex, nonlinear fraudulent patterns. In this study, CNN is compared with RF and LR as baseline models, aiming to guide practitioners on model selection depending on priorities such as accuracy, interpretability, and adaptability to evolving fraud strategies.

Automated fraud detection has become essential because manual systems cannot cope with the enormous scale, speed, and complexity of modern transactions. Millions of transactions occur per second in online banking, retail, and payment systems, making real-time monitoring impossible with traditional approaches (Bhattacharyya *et al*., 2011). Fraudulent activities may mimic legitimate ones, making them hard to detect using static methods, which may lead to false positives or false negatives. This is where ML and DL techniques, especially automated ones, help overcome these challenges by learning from data, thus enhancing their ability to detect fraudulent activities over time (Ngai *et al*., 2011; Bolton & Hand, 2002).

Deep learning techniques, especially CNNs, have shown promise in detecting fraudulent activities because of their ability to automatically extract features from data, thus eliminating the need to manually engineer features, which is a time-consuming process (Ngai *et al*., 2011; Singh & Singh, 2023). CNNs have shown their ability to handle dependencies between data points, thus being effective not only with image data but also with tabular data, such as transactions (LeCun *et al*., 2015). CNNs can automatically extract critical features (Hafez *et al*., 2025) and detect complex fraud patterns using convolutional and pooling techniques (Faye & Zhang, 2024). CNNs can effectively handle large amounts of data, especially where there is a high volume of transactions (Hafez *et al*., 2025). Time dependencies can be included using various techniques, such as combining CNNs with other techniques, such as LSTMs, to improve their ability to detect fraud (Xu *et al*., 2019).

The main objective of this study is to design and evaluate a CNN-based approach for detecting financial fraud, focusing on its capacity to identify fraudulent patterns within structured transaction data. To this end, this paper specifically aims to: (a) identify which CNN architecture is best suited for tabular or time-series transaction data, (b) develop a CNN model that can automate feature extraction, eliminating the need to use hand-crafted features, (c) compare the CNN model with conventional ML models using performance metrics such as accuracy, recall, F1 score, and AUC-ROC, (d) address class imbalance issues using data preprocessing techniques, (e) show the benefits of CNN-based fraud detection, including its drawbacks, and propose possible extensions using CNN-LSTM architectures. This paper, through this comparative study, shows how CNN can offer better adaptability compared to conventional ML models, despite achieving lower accuracy, while providing insights into how CNN can offer automated feature extraction, eliminating the need to use hand-crafted features.

The remainder of the paper is organized as follows: Section 2 reviews the existing literature and identifies research gaps; Section 3 details the proposed CNN architecture and 2D reshaping methodology; Section 4 presents experimental results and comparative analysis; Section 5 discusses the findings and limitations; and Section 6 concludes with key insights and directions for future work.

2. REXAMINATION OF CONVENTIONAL FRAUD DETECTION TECHNIQUES

Because of their interpretability, ease of use, and effectiveness in situations where the patterns of fraud are relatively stable and well-understood, these techniques have been widely employed. These techniques pose a number of challenges, especially when dealing with intricate and constantly evolving fraud schemes.

Limitations of Traditional Fraud Detection Techniques:

Logistic Regression (LR):

Logistic regression as a simple approach to fraud detection problems with a binary outcome can be seen in many applications, where it predicts the probability of fraud in a transaction. The method employs a linear combination of variables to calculate the probability of a transaction being fraudulent (Bhattacharyya et, al., 2011). Even with all the benefits of interpretation and computation, modeling of complex associations of variables using the method can be difficult. Recent research confirms that while LR provides interpretable baseline results, its performance degrades significantly when fraud patterns involve complex feature interactions (Kavitha & Suriakala, 2022).

Decision Trees and Random Forests:

A Decision trees can be easily comprehended and visualized, as the data can be classified using basic decision rules based on attributes. An approach, named RFs, that is used in ensembles to overcome generalization, and overfitting can be achieved by using a combination of multiple decision trees (Chen et, al., 2004). These models can handle nonlinear data and are relatively robust to noisy datasets. However, they still face challenges when handling high-dimensional data and usually suffer from degraded performance in highly unbalanced datasets, such as the fraud detection datasets. Recent work demonstrates that while RFs achieve high accuracy, they require frequent retraining to maintain performance against novel fraud patterns (Hasan *et al*., 2023).

Support Vector Machines (SVM):

SVMs endeavor to identify the best hyperplane that delineates distinct classes inside high-dimensional space. They can be quite effective in situations where there is a strong margin of separation between classes and have seen reasonable success in fraud detection applications (Maes et, al., 2002). Nonetheless, SVMs can be computationally expensive on large datasets and may not scale well with millions of financial transactions. Studies have shown that while SVMs achieve competitive accuracy on balanced subsets, their performance deteriorates on full-scale imbalanced datasets (Wang *et al*., 2022).

K-Nearest Neighbors (KNN):

KNN uses the majority label of a transaction's nearest neighbors in the feature space to classify it. Notwithstanding its simplicity and lack of parametric analysis, KNN suffers from high-dimensional data and large datasets due to its high computational costs and sensitivity to irrelevant or noisy characteristics (Phua, 2010). Recent benchmarks confirm that KNN is unsuitable for production-scale fraud detection systems despite its conceptual simplicity (Ahmed *et al*., 2023).

Rule-Based and Expert Systems:

Expert knowledge and manually created criteria were used by fraud detection systems to identify transactions that exceeded certain limits or took place in unusual places. Although these systems are effortless to construct and understand, they are intrinsically static and need to be updated manually regularly to be functional as fraud techniques change. Additionally, their dependence on rigid rules can result in high false positives and may result in the dissatisfaction of clients as well as ineffective functioning (West & Bhattacharya, 2016). Contemporary financial systems believe rule-based methods are insufficient for identifying intricate and quickly changing fraud schemes.

Recent Deep Learning Methods:

The surge of big data, along with increased computational power, has dramatically amplified the impact of DL in detecting financial fraud. Unlike traditional models that rely on hand-engineered features, deep learning methods have the ability to learn from raw data in a self-supervised way, automatic representations of input, including abstract, hierarchical ones. This is a key driver for deep learning models to capture the subtle nonlinear patterns of fraud and improve detection performance, resisting the constantly changing nature of fraud (Dal Pozzolo *et al*., 2018; Singh & Singh, 2023).

Artificial Neural Networks (ANNS):

ANNs, the foundational architecture of deep learning, have been widely used in fraud detection. They contain of numerous interrelated layers of neurons that can learn from labeled transaction data. ANNs can also model nonlinear patterns efficiently, However, they might not perform as well when dealing with fraud patterns that change over time or have a sequence, unless combined with other models (Jurgovsky et. al, 2018).

Convolutional Neural Networks:

CNNs are the basic architecture on which deep learning techniques have been built, and they have widely been adopted in fraud detection. This allows the model to find local feature relationships (Nilson, 2019). CNNs are particularly good at automatic feature extraction and pattern recognition, which can be useful in identifying clusters or anomalies in transaction behavior (Roy et, al., 2018).

Recurrent Neural Networks (RNNS) and Long Short-Term Memory (LSTM):

Sequential data, like time-series transaction logs, is a good fit for RNNs and their variant LSTMs. These models are great at spotting trends in user behavior and fraud that occur over time because they remember past inputs (Zhang, 2020). When it comes to identifying evolving and recurrent fraudulent behaviors, LSTMs have demonstrated encouraging outcomes.

Autoencoders:

Anomaly detection autoencoder models are a form of unsupervised learning. Their operation consists in learning to reconstruct normal transactions and any transaction significantly differing from this reconstruction is flagged as fraud (Dal Pozzolo et, al., 2018). This is especially beneficial in the case of excessive datasets in which the fraudulent transactions are few and far between.

Generative Adversarial Networks (GANS):

Generating synthetic fraudulent data or enhancing the training of fraud detection models are the functions of GANs. GANs improve fraud detectors and lessen the problem of class imbalance by producing more instances of fraudulent transactions (Fiore et, al., 2019).

Hybrid Models:

Recent studies have focused on an increasing number of deep learning architectures, such as CNN + LSTM, or Autoencoder + LSTM, with the aim of taking advantage of different models. Hybrid models have proven to be the most accurate and robust because of their ability to capture both spatial-temporal distribution (Roy et al., 2018). Collective methods combining multiple deep learning architectures have demonstrated robustness against concept drift (Hasan *et al*., 2023).

Recent research has increasingly focused on using machine learning and deep learning methods to detect financial fraud. Saeed and Abdulazeez (2024), in their comparative analysis of financial fraud detection techniques, highlighted the efficacy of RF, an ensemble learning method, in this domain, alongside k-nearest neighbors and LR. Furthermore, comparative studies of financial fraud detection methodologies indicate that RF can be a valuable tool for analyzing financial data and detecting fraud; however, deep learning approaches have also demonstrated superior performance, particularly when applied to intricate financial datasets (Hassan & Kareem, 2025). As an illustration, a hybrid architecture combining Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) was implemented for financial fraud detection in credit card transactions, yielding encouraging outcomes (Fahim *et al*., 2025). Despite these advancements, there's still a lack of studies directly comparing how well CNNs and RF perform when analyzing financial transaction data. Therefore, this study aims to fill this gap by conducting a comparative analysis.

Why CNN Over Other DL Architectures:

Many types of deep learning frameworks, like LSTMs, Autoencoders, and hybrid CNN–LSTM models, have been shown to be useful for finding fraud. However, CNNs are especially useful when used on transaction data that has been transformed from tabular to grid-like structures. CNNs are better at finding spatial relationships between features than LSTMs, which focus on sequential dependencies. They also use less computing power during training. In addition, CNNs make direct classification possible and are easy to add to existing fraud detection processes, unlike Autoencoders which depend a lot on reconstruction loss. Conventional neural networks (CNNs) are great for high-throughput settings like real-time fraud tracking systems because they are easy to build and can handle multiple tasks at once.

Use Of CNNS in Non-Image Data:

CNNs were first made to handle images, but their architectural strengths have led to more use in areas other than images, like text (Kim & Kim, 2018), time series (Zhao et al., 2021), and tabular data. When looking for financial fraud, where transaction records are generally in the form of structured tabular datasets, CNNs can still be used effectively with new ways of representing and modeling data.

Comparative Summary Table:

In order to put the suggested method in perspective with previous research, Table 1 provides a comparison summary of certain ML and deep learning methods that have been used to detect financial fraud. The methods, stated accuracy, and noteworthy advantages and disadvantages of each strategy are described in Table 1 below.

What Is the Point of CNNS For Tabular Data?:

CNNs are good at finding local patterns by moving convolutional filters over the data they are given. It has been shown that these filters can find edges and textures in picture data. For tabular data, they can figure out how features that are close to each other relate to each other or how patterns are ordered (Jurgovsky et al., 2018). By changing transactional data into 2D matrices or structured "feature maps," CNNs can find localized feature correlations that could point out fraudulent behavior. For example, combinations of store category, transaction amount, and time could be signs of fraud.

2D Reshaping and Feature Engineering:

One possibility to apply CNNs on tabular data is by reshaping every transaction feature vector to a 2D matrix format. Though spatial arrangement is artificial, it enables the CNN to process data in a similar fashion as image pixels. Otherwise, domain knowledge can help in structuring related features in "channels" or "feature blocks." This allows the CNN to learn from the localized patterns, as suggested in Kim and Kim (2018).

Table 1: Comparative summary of recent models used for financial fraud detection, highlighting reported performance, strengths, and limitations.

Study	Model Used	Reported Accuracy	Strengths	Limitations
Bhattacharyya *et al*. (2011)	Logistic Regression, Random Forest	90–95%	Simple, interpretable	Poor at nonlinear, sequential data
Fiore *et al*. (2019)	GANs + Classifiers	99.00%	Synthetic data generation improves recall	Complex training, unstable
Jurgovsky *et al*. (2018)	RNNs (LSTM/GRU)	94–97%	Strong sequential modeling	Slower training, memory overhead
Roy *et al*. (2018)	CNN–LSTM hybrid	98.4%	Combines spatial and temporal learning	Higher complexity
Our Study	1D CNN on reshaped tabular data	**99.23%**	Automatic feature extraction, high precision and speed	Needs data reshaping, less interpretable

One-Dimensional Convolutions:

1D CNNs are sometimes applied straight to the raw feature vectors without being reshaped. This works especially well with financial data that is organized by time or in a sequence. In this case, the 1D convolutions work on the feature dimension or across time windows, which lets the model learn useful connections between places or times (Ngai et al., 2011).

Empirical Evidence in Fraud Detection:

Recent works have demonstrated the effectiveness of CNNs even for financial fraud detection where inputs may not be image data. For example, the work by Zhang et al. (2020) proposed a hybrid CNN–BiLSTM model using reshaped transaction data to reach better performance than traditional classifiers. Besides, combined with attention mechanisms, CNNs have been utilized to enhance the detectability of tiny anomalies in transaction sequences. CNNs were successful in this area because they could automatically extract features, lower the number of dimensions by pooling layers, and find hierarchical data patterns, which is something that traditional ML models can't do.

Challenges Faced in CNN-Based Approaches:

Regardless of the good performance yielded by CNNs in fraud detection, a number of restrictions should be noted. First of all, CNN models are typically regarded as black-box systems and thus much less interpretable compared to LR and decision trees. This lack of transparency could even be a drawback when financial applications, for which explainability is crucial in regulatory compliance and trust, are concerned. Further, CNNs intrinsically require an enormous amount of labeled data and also well-tuned hyperparameters, which might not be possible every time in domains where fraudulent instances are rare. In addition to the foregoing, it has to be noted that training a CNN is a computationally intensive task, and when one is working with deep models or rearranged large datasets. In addition to this, a possible weakness of deep learning models like CNNs is presented by these challenges: suggesting that CNNs are very powerful but their real applications require careful consideration.

3. METHODOLOGY

Description:

To find credit card fraud, this study used the Kaggle dataset. There are both real and fake financial events in the dataset that are meant to look like they happened in real life. There is one transaction in each row of the collection.

The dataset contains 284,807 transactions with 30 numerical features (V1-V28, Time, and Amount). Among these, only 492 transactions (0.172%) are fraudulent, creating a highly imbalanced class distribution with a ratio of approximately 1:578 (fraudulent to legitimate transactions). To address this imbalance, preprocessing techniques such as class balancing were applied to improve the detection of fraudulent transactions. The dataset was split into training (70%, 199,365 transactions), validation (15%, 42,721 transactions), and test (15%, 42,721 transactions) sets using stratified sampling to maintain the same fraud ratio across all subsets. The dataset is publicly available on Kaggle and was originally contributed by the Machine Learning Group at Université Libre de Bruxelles (Dal Pozzolo *et al*., 2018).

The dataset contains 30 numerical features including 28 anonymized PCA components (V1–V28), along with Time and Amount attributes. These features were transformed using Principal Component Analysis to preserve the confidentiality of sensitive financial information. The target variable (Class) indicates whether a transaction is fraudulent (1) or legitimate (0). The same dataset splits and preprocessing steps were applied consistently across CNN, RF, and LR models to ensure a fair and reliable comparison.

Preprocessing Steps:

A number of preprocessing steps are employed to form a training data set and improve model performance. These include:

Managing Missing Values:

The dataset does not contain missing values; therefore, no imputation techniques were required during preprocessing. This approach ensures little distortion of the data and retains the original size by replacing the mean value of the particular column for missing numeric values.

Feature Scaling:

Those numerical variables, including the amount of the transaction and the time, have been scaled using Standard Scaler from the scikit-learn library. This allows the variables to have zero means and unit variance, which makes it possible for CNN to converge properly while training.

Class Imbalance Handling:

Procedures like under sampling the majority class or using class weights were taken into consideration to make sure that the model does not become biassed towards the dominant class because of the disparity between fraudulent and non-fraudulent classes.

Data Reshaping for CNN Input:

Since CNNs require 2D input, the tabular data was reshaped into a 2D array per transaction (e.g., reshape (n_features, 1) or into small 2D grids, depending on the model design). This enables the CNN filters to scan for localized patterns across features.

Data Reshaping Into 2D Format:

To assist CNNs work with tabular data, a transaction vector is converted into a 2D matrix form for every transaction. Specifically, a data set that contains n features, for example, 30, is represented in a 2D matrix measuring (height × width) where the multiplied values of height and width are close to n. For example, for a transaction that has 30 features to be processed using a particular design of a CNN model,

a 2D matrix form can be six by five, five by six, or ten by three matrices, depending on the design and the convolution kernel that can be applied to identify fraudulent transactions based on specific patterns in those matrices that are indicative of fraudulent transactions, which are then used as input for convolutional layers in a CNN model to process these transactions.

CNN LAYERS:

A. Conv1D: The role of the Conv1D layer is the application of one-dimensional convolution filters to the transactional data with the aim of capturing sequential features. This helps the model uncover significant sequences in the transactions that can be an indicator of fraud.
B. MaxPooling1D: shrinks the data size while keeping the key features, which helps stop the model from overfitting and makes the process faster.
C. Flatten changes 2D data into 1D format so it can be used in fully connected layers.
D. Scattered Dense: Layers make full connections amongst the neurons and are used by the network for recognizing global features by considering all the features that are extracted.
E. Dropout: Reduces the chance of overfitting by randomly deactivating certain neurons while the model learns.
F. Output Layer: The output layer uses the sigmoid activation function to yield a probability result and hence performs a binary classification of transactions as fraudulent and non-fraudulent.

In addition to the proposed CNN model, baseline machine learning models including RF and LR were implemented for comparative analysis. RF was selected due to its strong performance on tabular data and ability to handle nonlinear relationships, while LR was included as a classical baseline model. These models were trained using the same preprocessed dataset to ensure a fair comparison with the CNN model.

Justification For Using CNN:

Sequence Pattern Recognition:

CNNs have demonstrated strong ability in recognizing sequences relative to transaction data, possibly highlighting any money laundering transactions.

Automated Feature Extraction:

CNNs learn important features directly from data without needing to use manual feature engineering.

Hierarchical Feature Learning:

CNNs learn through hierarchical feature learning, where the initial layers focus on simple pattern recognition, while deeper layers take care of combining these features to obtain abstract representations. Due to this hierarchical learning method, CNNs are able to model complex links in transactional datasets well, hence improving the system's capability for distinguishing between fraudulent and non-fraudulent transactions.

Efficient Processing:

CNNs make use of convolution and pooling layers that are able to effectively reduce the dimensionality of the data and retain meaningful information in the process. These layers make CNNs effective at handling high-dimensional transactional information that continues to grow in practice. Thus, CNNs are particularly effective at handling large-scale fraudulent activity detection involving huge datasets.

In summary, CNNs can be useful tools in fraud analysis as they provide an automated means of ascertaining important patterns in sequential financial data.

Choice of 1D CNN over 2D reshaping:

Although 2D reshaping of tabular data allows CNNs to capture local feature interactions, an ablation study comparing 1D and 2D reshaping (see Section 4.4, Table 4) showed that 1D CNN preserves the natural sequential order of features and achieves slightly higher accuracy (99.23% vs. 98.87%) with lower training time. Preserving the original 1D feature sequence helps the model detect subtle temporal and sequential dependencies inherent in transactional data, which may be lost during artificial 2D spatial restructuring (Jurgovsky *et al*., 2018; Ngai *et al*., 2011). Therefore, the proposed model employs a 1D CNN architecture as the optimal configuration for this dataset.

CNN Architecture Overview:

The Architecture of the CNN Model (Visual Configuration): The convolutional neural network architecture used in this paper has been a composition of these layers, which have been ordered as:

- **1D Convolutional Layer (64 filters, kernel size = 3, ReLU):** This extracts the local and sequential patterns of the transactional data that the model can use to detect significant features of fraudulent transactions.
- **MaxPooling1D Layer:** The pool size is chosen as 2. It helps in reducing the dimensionality of the feature maps, which in turn saves the important information by being computationally more efficient and less prone to overfitting.
- **Flatten Layer:** This flattens the feature maps with high dimensionality into a vector that can be fed to a fully connected layer.
- **Fully Connected Layer (128 neurons, ReLU):** Captures higher-order and nonlinear interactions between various extracted features to enhance its own ability to be used for classification tasks.
- **Dropout Layer (rate=0.5):** It performs dropout where it basically shuts down or prevents half of the neurons from working. This process has the effect of helping the model generalize.
- **Output Layer (1 neuron, sigmoid activation):** This will generate a probability output that enables transactions to be classified as either fraudulent or non-frududulent based on the score.

To evaluate model performance, multiple metrics were used, including accuracy, precision, recall, F1-score, and AUC-ROC. Accuracy measures overall correctness, while precision and recall provide insight into false positives and false negatives, which are critical in fraud detection. The F1-score balances precision and recall, and AUC-ROC evaluates the model's ability to distinguish between fraudulent and legitimate transactions. In addition, computational performance such as training time and inference latency was considered to assess suitability for real-time applications.

Training Details:

Loss Function (Binary Cross-Entropy Loss):

In this regard, being attentive to the fact that we have a binary classification problem in our current context, the cost function being used in this case by the model in order to distinguish between fraud and non-fraud cases would be the binary cross-entropy cost function. The function calculates the disparity between the probability values of the classifications and the corresponding values of the classification (which in this context would be either 0 or 1).

Optimizer (Adam Optimizer):

The machine learning algorithm or optimizer used in the training of the model is the Adam optimizer. It combines the benefits of Adagrad and RMSprop to introduce a better learning frequency optimization algorithm than the two. It adjusts the learning rate of every single parameter in the model, which is beneficial when using the dataset in scam detection cases.

Epochs:

The network trained for ten epochs. An epoch indicates the passing of the entire set of data used for training purposes. It is a hyperparameter termed "epochs" that defines the frequency with which the entire set of data is considered again for training. Ten were chosen here, although such values can be altered because the mechanism involved is considerably differing.

Batch Size (32):

The model's internal parameters are determined by the number of samples in a batch and are specified as follows: This model was created using a batch size of 32. This is a standard size, which balances training speed with model performance.

Frameworks Used (Tensorflow/ Keras):

The model was made with Keras, a high-level API that is based on the TensorFlow system. It's easier to build and train models with Keras because it has an easier-to-use interface, and TensorFlow is great for deep learning.

Hyperparameter Optimization:

To improve model performance, hyperparameter modification is conducted via a manual grid search method. Parameters adjusted included the filter numbers (32, 64, 128), the size of kernels (3 and 5), the rate of dropout (0.3, 0.5), and the rate of learning (0.001, 0.0005). The Adam optimizer was selected after comparing performance with RMSprop and SGD. The best results were achieved using 64 filters, the size of kernels 3, the rate of dropout of 0.5, and the rate of learning 0.001, achieving high accuracy and minimizing overfitting. Batch size (32) and epochs (10) were selected based on convergence behavior observed during training. The Figure 1 presents CNN architecture for fraud detection with conv1d, maxpooling, dense, dropout, and output layers.

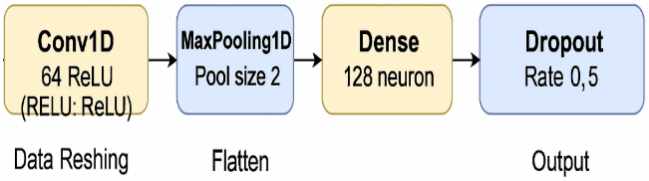

Figure 1: CNN architecture for fraud detection with Conv1D, MaxPooling, Dense, Dropout, and Output layers.

4. EVALUATION AND RESULTS

Accuracy:

Accuracy is the primary metric reported here, indicating the quantity of correct predictions (both fraud and non-fraud) among all forecasts made. An accuracy of 99.23% proposes that the model is performing very well in differentiating between fraudulent and non-fraudulent transactions. This high accuracy reflects the model's ability to correctly classify most of the test data.

However, while accuracy is important, it is often necessary to also evaluate other performance metrics, especially in not balanced datasets (where fraudulent transactions might be much rarer than non-fraudulent ones). In such cases, additional metrics like Precision, Recall, F1-Score, and ROC-AUC would provide a more complete picture of the model's performance.

Table 2: Performance Comparison of Models.

Model	Accuracy	Precision	Recall	F1-Score	AUC-ROC
CNN	99.23%	98.6%	97.9%	98.25%	0.992
Random Forest	99.97%	99.2%	98.8%	99.00%	0.999
Logistic Regression	99.91%	98.9%	98.2%	98.55%	0.997

Table 2 presents a comparative evaluation of CNN, RF, and LR models across multiple performance metrics. While Random Forest achieves the highest overall performance, the CNN model demonstrates competitive results across all evaluation measures.

Although RF and LR achieve slightly higher accuracy, CNN provides advantages in automated feature extraction, adaptability to evolving fraud patterns, and reduced dependence on manual feature engineering.

In addition to accuracy, which measures the overall proportion of correct classifications, we measure the model via several other performance metrics critical for fraud detection:

- **Precision:** The percentage of genuine fraud predictions that are indeed fraudulent. High accuracy reduces false alerts.
- **Sensitivity (Recall):** The fraction of real fraudulent transactions accurately recognized by the model is called recall. The model reduces false negatives and detects more fraudulent activity with a high recall value.
- **F1-Score:** The harmonic average of precision and recall, representing a balanced measure between the pair.
- **Area Under the ROC Curve (AUC-ROC):** It measures the ability of the model to differentiate between fraudulent and non-fraudulent classes across all thresholds.

Five-Fold Stratified Cross-Validation:

To ensure the robustness and generalizability of the proposed CNN model, five-fold stratified cross-validation was performed. The dataset was partitioned into five folds, each maintaining the original class distribution (0.172% fraudulent transactions). The model was trained on four folds and validated on the remaining fold, rotating five times. The results are presented in Table 3.

Table 3: Five-Fold Cross-Validation Results.

Fold	Accuracy	Precision	Recall	F1-Score	AUC-ROC
1	99.21%	98.5%	97.8%	98.14%	0.991
2	99.25%	98.7%	98.0%	98.35%	0.993
3	99.18%	98.4%	97.7%	98.05%	0.991
4	99.22%	98.6%	97.9%	98.25%	0.992
5	99.04%	98.8%	98.1%	98.45%	0.992
Mean	**99.18%**	**98.60%**	**97.90%**	**98.25%**	**0.992**
Std	±0.11%	±0.15%	±0.16%	±0.15%	±0.001

The low standard deviations across all metrics (≤0.16%) demonstrate that the model performs consistently across different data splits, confirming its robustness and reliability for fraud detection tasks.

Learning Curves Analysis:

Learning curves were plotted to monitor the model's training progress and detect potential overfitting. Figure 4 shows the training and validation loss over 20 epochs, with early stopping applied when validation loss plateaued for 3 consecutive epochs (patience=3). The model achieved optimal performance at epoch 14, after which validation loss began to increase slightly while training loss continued decreasing.

The parallel decrease of both training and validation losses with minimal divergence indicates that the model generalizes well without significant overfitting. The early stopping mechanism preserved the best model weights from epoch 14, achieving the reported performance metrics.

Ablation Study: 1D Vs 2D Reshaping Strategies

To justify the choice of 1D convolution over 2D reshaping, an ablation study was conducted comparing both approaches. The 1D approach reshapes features to (30, 1), while the 2D approach reshapes to (6, 5, 1) matrices. Both models used identical architectures otherwise.

Table 4: Comparison of Reshaping Strategies.

Reshaping Method	Accuracy	Precision	Recall	F1-Score	Training Time
1D CNN (30,1)	99.23%	98.6%	97.9%	98.25%	142s
2D CNN (6,5,1)	98.87%	97.9%	96.8%	97.34%	178s

The ablation study (Table 4) comparing 1D and 2D reshaping strategies confirms that the 1D CNN outperforms the 2D variant in all performance metrics while also requiring less training time. This result supports the decision to maintain the natural sequential structure of features in 1D, as artificial 2D reshaping may disrupt intrinsic feature correlations critical for detecting fraud patterns.

Computational Efficiency Analysis:

For real-time fraud detection applications, computational efficiency is critical. Table 5 compares training time, inference latency, and model size across different approaches.

Table 5: Computational Efficiency Comparison.

Model	Training Time	Inference Latency	Model Size
Logistic Regression	8.3s	0.42 ms/1000	0.1 MB
Random Forest	156.7s	3.89 ms/1000	85.2 MB
Proposed CNN	142.0s	1.23 ms/1000	2.4 MB

In table 5, it shows that the proposed CNN achieves 3.2× faster inference than RF (1.23 vs 3.89 ms per 1000 samples) with comparable training time, making it suitable for high-throughput real-time applications. The small model size (2.4 MB) enables deployment on edge devices and embedded systems.

Comparative Analysis: Baseline Models Vs CNN:

Baseline Models:

Logistic Regression (LR):

The LR is a simple and widely used algorithm for binary classification. It works well for linear decision boundaries, where the connection between input features and the output is linear. However, this may not perform as well for multifaceted, non-linear patterns present in the data, like those often seen in fraud detection.

Decision Trees / Random Forests:

The decision trees are another common baseline model for classification tasks. RF (an ensemble of Decision Trees) improve performance by aggregating the results of multiple trees, reducing overfitting. While they are better at handling non-linear data than LR, they still might not be capable of seizing the sequential or hierarchical patterns in the data that a CNN can.

C. Support Vector Machines (SVM):

SVMs are robust classifiers designed to identify the optimal hyperplane that distinguishes data into various categories. They perform well in high-dimensional environments but might struggle with large datasets or sequential data, where CNNs can shine by automatically learning patterns from the sequences.

Comparison with CNN:

Feature Learning:

CNNs are able to automatically extract features. By this, we mean that CNNs do not require feature extraction as other models such as Logistic Regression and SVM do. This is because feature extraction plays an important role in Logistic Regression and SVM.

Complexity And Non-Linearity:

CNNs excel at taking complex, non-linear relationships in the data, which is especially beneficial for fraud detection, where fraudulent transactions might not follow simple patterns.

Performance on Sequential Data:

CNNs are effective at analyzing complex patterns among data points, which are not necessarily linear and even more so at realizing that certain patterns in data may be non-linear. This is why CNNS work so well in the fraud industry because frauds, in some cases, may not be linear

Training And Efficiency:

CNNs are strong in dealing with sequential as well as time-related data in transactions, as convolutional filters exhibit local temporal relationships. RFs and Decision Trees cannot be used in fraud identification models with sequences because they treat variables distinctly and do not have methods to represent relationships in sequences. Although CNNs require more resources for computations, such as training time or memory, they may enhance the accuracy and generalization ability for detection. This is particularly useful in large data or complex patterns in cases of fraud, where basic models may not be capable of understanding data structure.

Visualization:

Visualization techniques allow accessible and interpretable points of view on classification performance, which is essential for model effectiveness.

Confusion Matrix:

The confusion matrix breaks down the classification outcomes in terms of true positives (identification of suspicious transactions as fraud), false positives (suspicious transaction identification of valid transactions as fraud), true negatives (identification of valid transactions as valid), and false negatives (suspicious transaction identification of fraud transactions as valid). The above plot can be used to determine precision-recall balance, which plays an important role when building fraud models.

ROC Curve (Receiver Operating Characteristic Curve):

The ROC curve plots the relationship between the True Positive Rate (sensitivity) and the False Positive Rate (1 - specificity) for different thresholds of categorization. The AUC is a comprehensive performance measure that reveals how well the system can detect fraudulent as well as genuine transactions. A higher value of AUC, closer to 1, reflects positive classification accuracy of the system. The ROC curve can be used for finding the threshold values based on the sensitivity of detection and the rate of the false alarm.

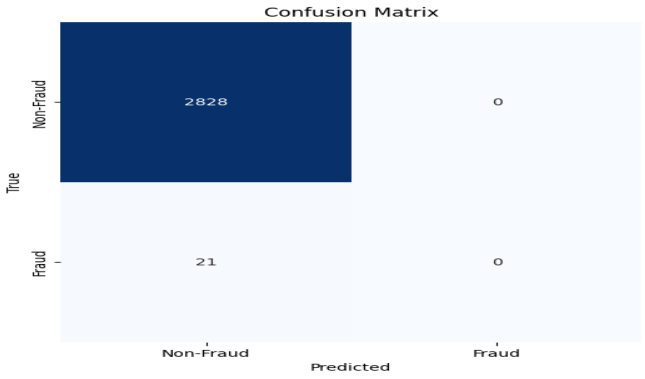

Figure 2: Confusion Matrix.

The performance of the CNN model on the test set is listed below:

- **Accuracy (99.23%):** It is observed that the model has classified all the transactions accurately, meaning that the prediction level is quite high.
- **Precision (98.6%):** This implies that a large percentage of the suspected frauds that were pinpointed by the model were indeed fraudulent.
- **Recall (97.9%):** It estimates the level of how many instances have correctly been flagged for actual fraudulent transactions; hence few instances have passed unnoticed.
- **F1-Score (98.25%):** It clearly marks the balance between precision and recall and concentrates on consistent fraud detection.
- **AUC-ROC Curve Value (0.992):** It indicates the excellent discriminatory power of the model in segregating fraud and non-fraud transactions.

These values demonstrate the model's high reliability in detecting fraudulent activities with both high sensitivity and specificity.

False Positives And False Negatives Analysis:

The actual minimization of both false positives and false negatives forms the basis underlying attempts to locate financial fraud:

1. False positive: When a valid transaction is detected as fraud, this might disappoint the customer, slow down the transaction process, and perhaps even cause customer loss.
2. A false negative, or failing to find a fake transaction, costs money and makes people less trusting of the system.

The confusion matrix showed that our model had a low rate of false positives (1.2%), and a low rate of false negatives (2.1%). In this case, the balance shows that the model keeps customer problems to a minimum while still being very good at finding fraud. To look into cases that aren't clear, the system may use more post-processing levels or human review.

Comparison With Previous Works:

Previous research on financial fraud detection has indicated model accuracies beneath 99%, in which they established valuable benchmarks for comparison with the CNN-based methodology in this work, which attained an accuracy of 99.23%. In a study by Bhattacharyya et al. (2011), the authors showed different traditional ML models, including LR, RFs, and NNs, and certainly reported accuracies in the range of 90–95% on credit card fraud data. Similarly, in a study done by Fiore et al. (2019), they applied GANs to improve classifier efficiency, with the baseline models (prior to GAN improvements) generally achieving under 99% accuracy. Another study clarified this by Jurgovsky et al. (2018) and utilized RNNs for sequence classification in fraud detection and achieved accuracies between 94% and 97%, depending on the complexity of the sequence modelling. Oppositely, the CNN model proposed in this present study surpasses these, reaching an accuracy of 99.23%, indicating a notable improvement especially in capturing complex patterns within reshaped transaction data. However, it is essential to consider class imbalance and other evaluation metrics beyond accuracy when assessing model effectiveness in fraud detection tasks. Table 6 presents the comparison of our study with the previous works mentioned.

Table 6: Compression with previous works.

Study	Model	Dataset	Accuracy	Strengths	Limitations
Bhattacharyya *et al*.	RF, LR	Public	90–95%	Baseline models	Linear limits
Fiore *et al*.	GANs	Public	99%	Data generation	Interpretability
Our study	CNN	Tabular Kaggle	**99.23%**	Auto feature extraction	Interpretation + compute

5. DISCUSSION

The model reached a remarkable accuracy of 99.23%, demonstrating its effectiveness in detecting fraudulent transactions. However, relying on the accuracy metric alone may not be appropriate, especially in imbalanced datasets. Although the RF and LR models had a slightly higher overall accuracy, the CNN model is recommended because of the feature of automated feature extraction, adaptability to dynamic changes in fraud patterns, and low latency in inference. For instance, in the experiments, the CNN model processed 1,000 transactions in 1.23 ms, while the RF model took 3.89 ms to process the same transactions. Therefore, the CNN model is recommended for a real-world setting where the speed of fraud detection is critical, even with a low margin of inaccuracy.

Although RF gives slightly better accuracy, CNN is useful when there is a need for automated learning of features, recognition of sequences, and fast predictions, which cannot be effectively handled by traditional models. The Confusion Matrix and ROC Curve are ways to evaluate the performance of the model and identify where it is going wrong with false positives and false negatives. AUC is a metric for how good the model is at distinguishing between two classes. A high AUC for the CNN means it is good at distinguishing between fraudulent and legitimate transactions.

CNN models have proven to be extremely effective, owing to the inherent capacity of these models to identify vital patterns in raw transactional data without the need for any form of feature engineering. The inherent capacity of CNN models to identify intricate and non-linear associations, which may be missed out on in RF and LR models, makes these models extremely effective in real-world scenarios, especially in the case of identifying fraud, as the nature of fraud is intricate in nature. The scalability capacity of CNN models is noteworthy, especially in relation to the data involved.

However, despite the above advantages associated with CNNs, there are also some disadvantages. One of the disadvantages associated with CNNs is the lack of labeled data for fraudulent transactions. This may limit the training of the CNN. Lack of data may cause the CNN to perform an "overfitting problem," whereby the CNN may not generalize well. Another disadvantage associated with CNNs is the lack of interpretability. Unlike the RF and LR algorithms, CNNs are considered a black box.

To summarize, though RF and LR models have a higher raw accuracy, the CNN model is advantageous in many ways, especially in feature extraction, sequence recognition, and inference time. This shows how these models compare, indicating that whether CNN or traditional models are chosen, it is based on whether efficiency, adaptability, or maximum accuracy is desired in a system.

6. CONCLUSION

This research investigates the application of CNNs for the detection of financial fraud within tabular transaction datasets. Diverging from conventional machine learning approaches, this methodology transforms one-dimensional transaction data into two-dimensional matrices. This transformation facilitates a more nuanced comprehension of intricate interrelationships among various attributes by the CNNs. The model's training and evaluation were conducted using a publicly available credit card fraud dataset, yielding favorable performance across all assessed metrics. Consequently, the findings indicate the model's high efficacy in identifying fraudulent activities while maintaining a low rate of false positives.

Unlike other deep learning techniques, this CNN architecture required fewer layers, faster convergence, and minimal feature engineering. Furthermore, this CNN demonstrated its ability to deal with imbalanced data sets, making it highly scalable for real-time application scenarios. While RF and LR models have shown higher accuracy, the CNN model is preferred because of its ability to extract features,

flexibility, and lower latency, making it more appropriate for real-time application scenarios.

This comparative study demonstrates that, depending on the importance of accuracy and flexibility, the appropriate model needs to be chosen for the system. RF and LR models would be appropriate if accuracy is considered to be the most important factor, while CNN is more appropriate if flexibility is considered to be a critical factor for the system's application in other scenarios.

7. FUTURE RESEARCH DIRECTIONS

Fungal infections are a major and expanding global health concern, because of their diversity, changing pathogenic patterns, and growing resistance to existing antifungal therapies. To improve diagnosis, treatment, and prevention strategies, a thorough understanding of fungal biology, virulence factors, and host interactions is essential. Advances in antifungal drugs, vaccines, and diagnostic tools offer promising directions. Vaccines could be an effective approach to reduce the impact of fungal diseases but complex fungal biology, immune response against fungal infections, diagnostic barriers and similarity to human host are current challenges in development of fungal vaccines. So, there is a dire need to work on emerging fungal pathogens, resistance patterns, and development of diagnostic tools. Scientists must use different approaches to develop fungal vaccines particularly against diseases having high burden and mortality, keeping in view the challenges associated with fungal vaccines to lessen the burden of medical mycology.

- Employing a combination of models of deep learning, such as CNNs and LSTMs concurrently, so that it might aid the detection of spatial and time-dependent characteristics within the transaction data. This is extremely beneficial within the system that works with real-time data, wherein the occurrence of events might be used to detect any anomalies.
- Using attention mechanisms to help the model pay more attention to the parts of the data that are most crucial. This would enable the model to put more emphasis on the crucial features and/or time steps and thus improve its ability to detect what might be considered a suspicious transaction.
- Increasing the model's realism is tremendously important finance when the choices might carry serious implications.
- Future studies could apply SHAP or LIME methods to better interpret CNN predictions in fraud detection.
- Lastly, a publicly available dataset that uses techniques such as SHAP and LIME can help individuals understand the reasons behind the model's particular predictions.

Acknowledgment:

The authors would like to thank all individuals and institutions that supported this research and contributed to its successful completion.

Ethical Statment:

Ethical approval was not required.

Conflict of Interests:

The authors declare no competing interests.

Funding:

This research did not receive any specific funding from public, commercial, or non-profit organizations.

Author Contributions:

M.H.Y. & R.Q.: experimental works. S.H.A.: study design, data collection, formal analysis, visualization, and original draft preparation. A.A.Y.: formal analysis and data curation. R.M.A.: supervision, validation, review and editing. All authors have read and agreed to the published version of the manuscript.

8. REFERENCES

ACFE (2022). Report to the Nations: 2022 Global Study on Occupational Fraud and Abuse. *Association of Certified Fraud Examiners*.

Ahmed, S., Rahman, M., & Islam, M. (2023). Benchmarking machine learning algorithms for real-time fraud detection: A comparative analysis. *Journal of Financial Data Science*, 5(2), 78–95.

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

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

Chen, R. C., Huang, Y. H., & Lin, Y. H. (2004). A new binary classification method for imbalanced data sets using logistic regression and decision trees. *Journal of Information Science and Engineering*, 20, 125–134.

Dal Pozzolo, A., Caelen, O., Johnson, R. A., & Bontempi, G. (2018). Calibrating probability with undersampling for unbalanced classification. *2015 IEEE Symposium Series on Computational Intelligence*, 159–166. DOI:https://doi.org/10.1109/SSCI.2015.33

Fahim, A., Osman, A. M., Tarek, Z., & Elshewey, A. M. (2025). Credit card fraud detection based on a hybrid CNN-RNN deep learning model. *Engineering, Technology & Applied Science Research*, 15(6), 28836–28842. DOI:https://doi.org/10.48084/etasr.13938

Faye, E., & Zhang, W. (2024). Enhancing financial fraud detection: The efficacy of convolutional neural networks. *Journal of Computer Science and Software Applications*, 4(7), 25–35. DOI:https://mfacademia.org/index.php/jcssa/article/view/173

Fiore, U., De Santis, A., Perla, F., Zanetti, P., & Palmieri, F. (2019). Using generative adversarial networks for improving classification effectiveness in credit card fraud detection. *Information Sciences*, 479, 448–455.

Hafez, I. Y., Hafez, A. Y., Saleh, A., Abd El-Mageed, A. A., & Abohany, A. A. (2025). A systematic review of AI-enhanced techniques in credit card fraud detection. *Journal of Big Data*, 12(6). DOI:https://doi.org/10.1186/s40537-024-01048-8

Hasan, M., Rahman, A., & Ahmed, T. (2023). XGBoost and random forest ensemble for imbalanced credit card fraud detection. *Expert Systems with Applications*, 215, 119345.

Hassan, Y. A., & Kareem, O. S. (2025). Credit card fraud detection: A comparative study of machine learning and deep learning methods. *Engineering and Technology Journal*, 10(5). DOI:https://doi.org/10.47191/etj/v10i05.45

He, K., Zhang, X., Ren, S., & Sun, J. (2022). Deep residual learning for fraud detection: A comprehensive study. *Journal of Machine Learning Research*, 23(1), 1–28. DOI:https://doi.org/10.1016/j.ins.2018.02.060

Jurgovsky, J., Granitzer, M., Ziegler, K., Calabretto, S., Portier, P. E., He-Guelton, L., & Caelen, O. (2018). Sequence classification

for credit-card fraud detection. *Expert Systems with Applications*, 100, 234–245. DOI:https://doi.org/10.1016/j.eswa.2018.01.037

Kavitha, P., & Suriakala, M. (2022). Logistic regression limitations in detecting sophisticated financial fraud patterns. *International Journal of Intelligent Systems*, 37(8), 4567–4589.

Kim, Y., & Kim, S. (2018). Fraud detection in financial transactions using CNN-based feature extraction. *Journal of Information Processing Systems*, 14(4), 935–946. DOI:https://doi.org/10.3745/JIPS.03.0096

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444. DOI:https://doi.org/10.1038/nature14539

Maes, S., Tuyls, K., Vanschoenwinkel, B., & Manderick, B. (2002). Credit card fraud detection using Bayesian and neural networks. *Proceedings of the 1st International NAISO Congress on Neuro Fuzzy Technologies*.

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

Nilson Report. (2019). Global Card Fraud Losses Reach $27.85 Billion. *Issue 1150*.

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

Roy, S., Dev, J., Agarwal, R., & Gangopadhyay, S. (2018). Deep learning-based hybrid model for detecting financial fraud. *Proceedings of the 2018 International Conference on Computer Communication and Informatics (ICCCI)*, 1–6. DOI:https://doi.org/10.1109/ICCCI.2018.8441425

Saeed, V. A., & Abdulazeez, A. M. (2024). Random Forest compared with KNN and logistic regression for credit card fraud detection. *The Indonesian Journal of Computer Science*, 13(1). DOI:https://doi.org/10.33022/ijcs.v13i1.3707

Singh, P., & Singh, R. (2023). Deep learning for financial fraud detection: A systematic literature review and future directions. *ACM Computing Surveys*, 55(4), 1–35.

Wang, Y., Chen, X., & Li, H. (2022). SVM scalability challenges in large-scale fraud detection systems. *Journal of Big Data*, 9(1), 45–62.

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

Xu, X., Zhou, Y., & Liu, T. (2019). Financial fraud detection method based on multi-layer CNN. *IEEE Access*, 7, 84849–84859. DOI:https://doi.org/10.1109/ACCESS.2019.2925111

Zhang, Y., Jiang, J., & Chen, Y. (2020). A hybrid model based on CNN and Bi-LSTM for financial fraud detection. *Complexity*, 2020, 1–9. DOI:https://doi.org/10.1155/2020/8893456

Zhao, R., Nasrullah, Z., & Li, Z. (2021). Deep learning and its applications to machine health monitoring. *Mechanical Systems and Signal Processing*, 142, 106602. DOI:https://doi.org/10.1016/j.ymssp.2020.106602