applied sciences

MDPI

Article

Achieving Excellence in Cyber Fraud Detection: A Hybrid ML+DL Ensemble Approach for Credit Cards

Eyad Btoush [1,*], Xujuan Zhou [1], Raj Gururajan [1,2], Ka Ching Chan [1] and Omar Alsodi [1]

[1] School of Business, University of Southern Queensland (UniSQ), Springfield, QLD 4300, Australia; xujuan.zhou@unisq.edu.au (X.Z.); raj.gururajan@unisq.edu.au (R.G.); kc.chan@unisq.edu.au (K.C.C.); omar.alsodi@unisq.edu.au (O.A.)
[2] School of Computing, SRM Institute of Science and Technology, Chennai 603203, India
* Correspondence: eyadabdellatif.a.q.marazqahbtoush@unisq.edu.au

Abstract: The rapid advancement of technology has increased the complexity of cyber fraud, presenting a growing challenge for the banking sector to efficiently detect fraudulent credit card transactions. Conventional detection approaches face challenges in adapting to the continuously evolving tactics of fraudsters. This study addresses these limitations by proposing an innovative hybrid model that integrates Machine Learning (ML) and Deep Learning (DL) techniques through a stacking ensemble and resampling strategies. The hybrid model leverages ML techniques including Decision Tree (DT), Random Forest (RF), Support Vector Machine (SVM), eXtreme Gradient Boosting (XGBoost), Categorical Boosting (CatBoost), and Logistic Regression (LR) alongside DL techniques such as Convolutional Neural Network (CNN) and Bidirectional Long Short-Term Memory Network (BiLSTM) with attention mechanisms. By utilising the stacking ensemble method, the model consolidates predictions from multiple base models, resulting in improved predictive accuracy compared to individual models. The methodology incorporates robust data pre-processing techniques. Experimental evaluations demonstrate the superior performance of the hybrid ML+DL model, particularly in handling class imbalances and achieving a high F1 score, achieving an F1 score of 94.63%. This result underscores the effectiveness of the proposed model in delivering reliable cyber fraud detection, highlighting its potential to enhance financial transaction security.

Keywords: credit card cyber fraud; fraud detection; artificial intelligence; machine learning; deep learning; ensemble techniques; resampling techniques

Academic Editors: Douglas O'Shaughnessy and Pedro Couto

Received: 25 November 2024
Revised: 25 December 2024
Accepted: 14 January 2025
Published: 22 January 2025

Citation: Btoush, E.; Zhou, X.; Gururajan, R.; Chan, K.C.; Alsodi, O. Achieving Excellence in Cyber Fraud Detection: A Hybrid ML+DL Ensemble Approach for Credit Cards. *Appl. Sci.* **2025**, *15*, 1081. https://doi.org/10.3390/app15031081

1. Introduction

The technological revolution is developing rapidly owing to several key enabling technologies, such as Artificial Intelligence (AI), the Internet of Things (IoT), and big data. Given the widespread adoption of ever-evolving Internet technology, banks are implementing new technology and digital platforms to increase both their client base and revenue [1]. However, the rapid increase in technology usage has exacerbated cyber fraud using credit cards.

The growing prevalence of credit card cyber fraud poses a significant threat globally, with fraudulent activities ranging from the illegal appropriation of credit cards to the replication of card information and account takeover [2]. The widespread use of credit cards—over 2.8 billion worldwide—has increased opportunities for fraudsters to exploit this trend. In the United States alone, credit card fraud accounts for 46% of global fraudulent activities, and projections indicate that the total global loss due to credit card fraud could

reach USD 43 billion by 2026. Australia experienced a substantial rise in credit card fraud, with an estimated loss of AUD 2.2 billion in 2023, according to the Australian Bureau of Statistics. These statistics underscore the urgent need for improved fraud detection systems to counter the escalating threat of credit card cyber fraud, particularly as the landscape of online transactions continues to evolve [3]. The incidence of card cyber fraud among Australians saw a substantial increase in 2023. According to a report from the Australian Bureau of Statistics (ABS), an estimated gross amount of $2.2 billion was lost due to card cyber fraud. The proportion of Australians affected by card cyber fraud has increased from 6.9 percent in 2020–2021 to 8.7 percent in 2022–2023. Figure 1 shows the change in Australians' affected by personal cyber fraud from 2020 to 2023 [4].

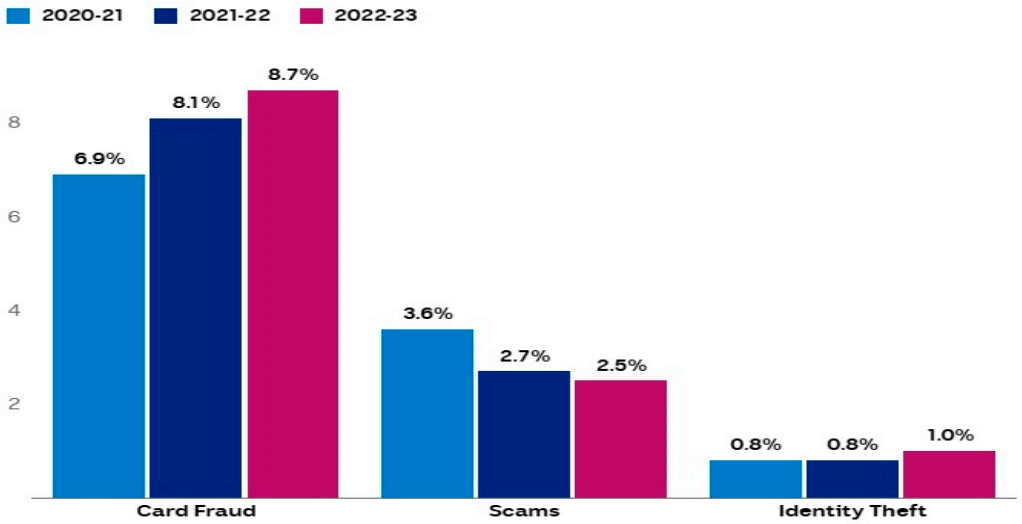

Figure 1. Change in Australians affected by personal fraud.

Detecting credit card cyber fraud is crucial for maintaining financial security, requiring continuous advancement to address evolving fraudulent tactics [5]. Traditional methods have limitations, prompting the exploration of Machine Learning (ML) and Deep Learning (DL) techniques [6,7] to improve accuracy, adaptability, and performance. However, ML and DL face challenges such as class imbalance, overfitting, and scalability issues [8], along with data-related challenges like unbalanced class distributions and evolving fraud patterns [9]. This paper develops and evaluates a Hybrid ML+DL model to improve fraud detection efficiency and accuracy.

The motivation for employing a hybrid ML+DL model in credit card fraud detection stems from the limitations of traditional models and the need to combine the strengths of both ML and DL techniques. ML models, such as Decision Tree (DT), Random Forest (RF), Support Vector Machine (SVM), eXtreme Gradient Boosting (XGBoost), Categorical Boosting (CatBoost), and Logistic Regression (LR) address challenges like imbalanced datasets and complex fraud patterns but struggle with sequential and temporal dependencies in transaction data. DL models like Convolutional Neural Network (CNN) and Bidirectional Long Short-Term Memory Network (BiLSTM) address these dependencies, uncover hidden relationships, and adapt to varying fraud patterns [10]. The inclusion of attention mechanisms enhances the model's focus on critical features, improving interpretability and accuracy [11]. By combining ML's efficiency with DL's depth, the hybrid model offers a robust, scalable, and highly accurate framework for tackling evolving credit card fraud.

The novelty of this approach lies in its integration of both ML and DL techniques into a hybrid ensemble model for cyber fraud detection in the banking sector. Unlike traditional ensemble methods, this model combines classical ML models (e.g., RF, SVM, LR)

and advanced DL techniques (e.g., CNN-BiLSTM-Attention). This hybridisation allows the model to detect complex fraud strategies, overcoming challenges such as imbalanced datasets, evolving attack patterns, and real-time detection needs. The combination of models provides flexibility, adaptability, and the ability to learn from new fraud patterns, while the meta-classifier enhances accuracy and minimises false positives.

The primary contribution of this research is the development of a hybrid ML+DL model using a stacking ensemble framework that integrates ML (DT, RF, SVM, LR, XGBoost, CatBoost) and DL (CNN-BiLSTM with attention). The model addresses challenges like imbalanced datasets, complex fraud patterns, and the need for precise fraud detection. Tested on a real-world dataset, it outperforms state-of-the-art models, significantly reducing false positives and undetected fraud cases. This work enhances accuracy, robustness, and provides insights for deploying scalable fraud prevention systems.

The remainder of this paper is organised as follows. Section 2 presents the related works, and Section 3 presents the material and methods used in this study. The proposed credit card fraud prediction approach is introduced in Section 4. Section 5 presents results and discussion, and Section 6 presents the conclusions.

2. Related Works

In this section, we examine the related literature on proposed systems and techniques for credit card fraud detection. The existing work in this field is categorised into two sections based on the technique used, ML algorithms and DL algorithms.

2.1. Machine Learning

ML, a field that enables computers to perform tasks without explicit programming, has the potential to achieve accurate predictions of risk and anomalous behaviour inside datasets, including instances of credit card theft [12]. The classification challenge in the field of ML pertains to the objective of accurately predicting the class label associated with certain data items. The objective of this scenario is to forecast whether a transaction is fraudulent or legitimate [13].

The SVM function discerns the optimal decision boundary that effectively distinguishes genuine transactions from fraudulent transactions. Ref. [14] implemented a model for credit card cyber fraud detection. The results indicate that the novel DT classifier achieves 94.86% accuracy, whereas the SVM predicts the same with 98.59% accuracy. In [15], an SVM classifier was implemented using the Multilayer Perceptron (MLP) technique. The research findings indicated that the SVM and MLP techniques achieved an accuracy of 94.59% and 91.21%, respectively.

DTs are hierarchical data structures that are commonly employed for classification or regression problems [16–18], employed the DT classifier to identify financial cyber fraud. The DT algorithm demonstrated the highest accuracy among the other classifiers. DT using the boosting technique was applied by [19]. The results show that the model achieved the highest accuracy of 98.3%.

RF is a flexible ensemble of DTs that is commonly used in the field of credit card cyber fraud detection. Ref. [20] applied RF for cyber fraud detection on skewed data. The results indicated that RF had the highest accuracy (95.19%) compared to KNN, LR, and DT. Furthermore, RF was applied with other techniques such as SVM, NB, and KNN in [21]. The results showed that the RF algorithm performed better than other techniques. A hybrid model was proposed by [22]. The results show that RF with KNN performed better than RF with a single classifier. In [23], LR, RF, and CatBoost were applied to detect cyber fraud. The results show that RF with CatBoost provides a high accuracy.

LR is a statistical strategy that models a binary dependent variable using a logistic function. The new system uses LR to build the classifier proposed by [24]. On a comparative analysis of the LR-based classifier with KNN and voting classifiers, the results indicate that the LR-based method yields the most precise conclusions.

Ref. [25] introduced an ensemble model that combines KNN, SVM, RF, Bagging, and Boosting classifiers. The performance of this ensemble was impressive and demonstrated the effectiveness of merging multiple classifiers to improve the accuracy. A compared different ensemble methods to predict cyber fraud in credit cards has been performed by [26]. The experiment shows that XGBoost performs better than other ensemble methods.

2.2. Deep Learning

DL is a subset of ML that focuses on analysing data through hierarchical feature extraction. The key advantage of DL is its ability to automatically learn features without the need for manual feature selection [27]. DL algorithms such as CNN and LSTM are associated with image processing and Natural Language Processing (NLP). Using these methods for credit card cyber fraud detection has yielded better performance than traditional algorithms [28].

CNNs have demonstrated their efficacy in successfully processing multi-channel data, therefore enabling a thorough analysis of transaction information. Ref. [29] used DL techniques such as CNN, BILSTM with an Attention Layer to classify illegitimate transactions. The CNN-Bi-LSTM-ATTENTION model is highly effective in identifying fraudulent classes. Analysis indicated that the model was adequate and yielded an accuracy of 95%. Ref. [30] introduced a credit card cyber fraud detector that was optimised for large-scale real-time datasets and utilised a CNN combined with a smart matrix algorithm. Compared to alternative ML approaches, the performance of the three-layered CNN model is superior. Ref. [31] suggested a hybrid CNN-SVM model for the detection of fraudulent credit card transactions. The experimental findings indicate that the hybrid CNN-SVM model achieved classification performances of 91.08%, 90.50%, 90.34%, and 90.41, respectively, in terms of accuracy, precision, recall, and F1-score.

LSTM is a helpful technique for predicting cyber fraud because of its historical knowledge and the link between prediction outputs and historical input. Ref. [32] developed a new model to improve both the present detection techniques and detection accuracy considering large amounts of data. The findings demonstrated that LSTM performed perfectly, achieving 99.95% accuracy. Ref. [33] recommended a model to record the previous purchasing behaviour of card holders. The results showed that the LSTM model achieved a high level of performance. Ref. [34] proposed a new model as a means of mitigating misclassification in cyber fraud detection systems. The application of an LSTM-RNN was implemented. A comparison of the obtained results to previous research revealed that this model achieved both a high rate of accurate classification and a low rate of false alarms. Ref. [35] introduced a novel hybrid model with the objective of identifying the occurrence of credit card cyber fraud. RNN-LSTM and an attention mechanism have been proposed. Comparing the performance of RNN-LSTM to that of ANN, XGBoost, RF, NB, and SVM classifiers reveals that the proposed model generates robust results with an accuracy of 99.4%.

BiLSTM models were employed to analyse data in both forward and backward directions. These models utilise LSTM cells equipped with memory units that enable them to record temporal patterns and relationships [36]. Ref. [37] introduced the Hybrid Sampling (HS)-Similarity Attention Layer (SAL)-BiLSTM architecture. By hybrid sampling of the minority class and undersampling of the majority class, the SMOTE-ENN decreases data discrepancy. SAL is implemented to quantify the similarity of a data sequence to assign

importance to distinctive features and mitigate the issue of overfitting in classification. The recall value of the proposed HS-SAL-BiLSTM was 99.2%, whereas that of the existing RF-SMOTE-SVM was 97.7%. A DL-based hybrid approach for detecting fraudulent transactions was applied by [38]. A Bi-LSTM autoencoder with an isolation forest is incorporated into the new model. This model suggests that fraudulent transactions can be detected at a rate of 87%.

The academic literature highlights various Machine Learning (ML) and Deep Learning (DL) methodologies applied to credit card cyber fraud detection [39]. In recent years, there has been a notable increase in the exploration of DL techniques, which offer flexibility in responding to complex data patterns and detecting new fraudulent behaviours. Despite advancements in ML and DL algorithms, current models still struggle with handling large-scale, real-world data and adapting to the evolving nature of cyber fraud. This research presents a novel hybrid ML+DL model that integrates multiple ML algorithms—such as Decision Trees (DT), Random Forest (RF), Support Vector Machines (SVM), XGBoost, CatBoost, and Logistic Regression (LR)—with Deep Learning techniques like CNN-BiLSTM. This hybrid approach optimises both precision and recall, while capturing complex fraud patterns that previous models often overlook, offering a more robust and adaptive solution to the challenges of credit card fraud detection.

3. Materials and Methods

This section explores the credit card dataset employed in the study and provides a comprehensive explanation of the various algorithms and strategies utilised in formulating the suggested credit card cyber fraud detection methodology.

3.1. Dataset

To assess the effectiveness of the suggested ML models, a widely recognised dataset was chosen for both training and testing. This dataset is available at https://www.kaggle.com/mlg-ulb/creditcardfraud, accessed on 3 May 2021. The dataset consisted of client transactions at a European bank in 2013. The real-world dataset consists of 284,807 credit card transactions.

3.2. Programming Language

Python, an easy, interpreted, object-oriented, and high-level language, is popular. The system utilises Numpy for linear algebra and multidimensional arrays. Pandas makes data manipulation fast and flexible. Python Scikit-learn was used for statistics and ML. The algorithms ran on 3.3 GHz Intel Core i7 PCs with 16 GB RAM. Python 3.11.4 models learn. Code runs in Jupyter Notebook.

3.3. Evaluation and Reflection

This study evaluates ML and DL algorithms for cyber fraud detection using 5-fold cross-validation to address dataset imbalance, focusing on F1 score for model evaluation and comparison. Metrics like confusion matrix, accuracy, recall, precision, and AUC-ROC were used to balance selectivity and specificity while minimising errors. Table 1 presents the performance indicators. Table 1 provides a thorough summary of the performance metrics.

Table 1. Performance indicators.

Metrics	Description	Equation	Range
Accuracy	Assess the number of TPs	$A = \frac{TN+TP}{TN+FN+TP+FP}$	[0–1]
Recall	The ratio of TP to a TP and FN	$R = \frac{TP}{TP+FN}$	[0–1]
Precision	The ratio of TP to a TP and FP	$P = \frac{TP}{TP+FP}$	[0–1]
F1-Score	Combines precision and recall	$F_1 = 2\frac{P*R}{P+R}$	[0–1]
AUC	The area between two points bounded by the function and the x axis.	$AUC = \int_a^b f(x)dx$	[0–1]

4. Method

This section describes the successful development of a hybrid ML+DL approach that integrates the ML techniques: DT, RF, SVM, XGBoost, CatBoost, and LR with the DL techniques CNN and BiLSTM. Figure 2 illustrates a block diagram of the proposed modelling framework that was developed in this study.

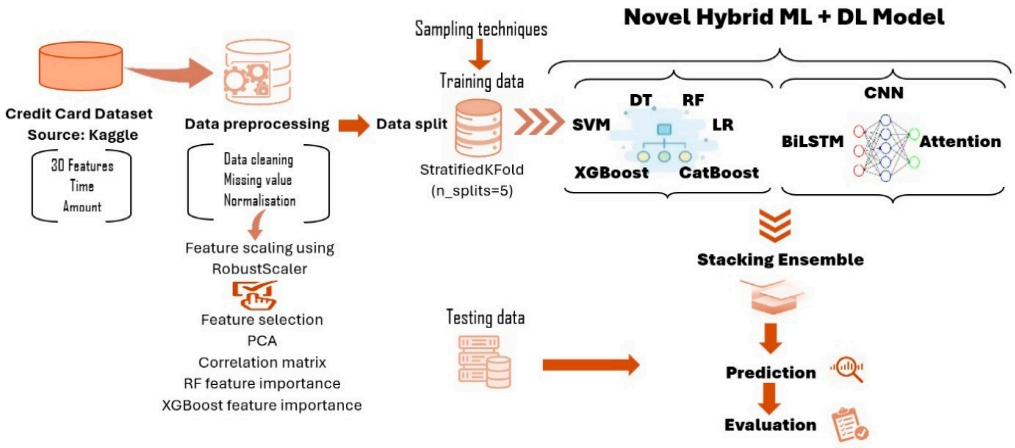

Figure 2. Hybrid ML+DL model.

The heart of the model lies in its architecture, which includes both ML classifiers and a DL model. The classical classifiers include RF, SVM, LR, DT, XGBoost, and CatBoost. These classifiers are trained individually on the training data and evaluated using standard metrics. Additionally, we introduce a CNN-BiLSTM with an attention mechanism. The CNN architecture consists of convolutional layers followed by batch normalisation, max-pooling layers, Bidirectional LSTM layers, Dropout layers for regularisation, and a custom AttentionLayer. This AttentionLayer helps the model focus on important features. The CNN is trained alongside the classical classifiers and is evaluated similarly. Moreover, we employ various callbacks during training, such as F1ScoreCallback, ModelCheckpoint, ReduceLROnPlateau, and EarlyStopping, to monitor the model's performance and prevent overfitting. Finally, we build a StackingClassifier that combines the predictions of all classifiers ML and DL, using an RF classifier as the final estimator. This StackingClassifier is trained on the training data and evaluated on the test set. The performance metrics, including accuracy, precision, recall, F1-score, ROC AUC score, and confusion matrix, are computed and visualised for comprehensive analysis. In summary, the novel hybrid stacking ML+DL model is a hybrid approach that combines the strengths of classical ML algorithms with the representation learning capabilities of a CNN-BiLSTM with an attention

mechanism, resulting in a robust and effective framework for detecting fraudulent transactions.

4.1. Machine Learning Techniques

Various ML techniques that have been applied including DT, RF, SVM, XGBoost, CatBoost, and LR, each tailored to address specific challenges in credit card cyber fraud detection. For instance, the DT model utilises recursive partitioning to create a tree-like structure, ensuring accurate classification by analysing features and decision points. On the other hand, RF constructs multiple DTs to improve predictive accuracy while preventing overfitting. SVM optimises the separation between distinct classes in the feature space, hence improving their effectiveness in binary classification problems. XGBoost employs gradient boosting to iteratively refine predictions, while CatBoost efficiently handles categorical features without pre-processing difficulties. Lastly, LR provides a fundamental yet powerful approach to binary classification, offering insights into the relationships between features and the target variable.

4.1.1. Decision Tree (DT)

DT is a supervised ML algorithm that recursively partitions the dataset into branches representing decisions based on input features. It uses hyperparameters such as max_depth (limits tree complexity), min_samples_split (minimum samples to split a node), min_samples_leaf (minimum samples in a leaf), and criterion (entropy to measure split quality). This configuration balances complexity and generalisation, enabling accurate classification.

4.1.2. Random Forest (RF)

RF is an ensemble method that creates multiple DTs and combines their predictions. Key hyperparameters include n_estimators (number of trees), max_depth (limits tree depth), min_samples_split and min_samples_leaf (control node splitting), and random_state (ensures reproducibility). These parameters help reduce overfitting while capturing complex patterns effectively.

4.1.3. Support Vector Machine (SVM)

SVM separates classes by maximising the margin between them using a kernel function. It uses an RBF kernel for non-linear separability, with hyperparameters like C (controls trade-off between margin size and classification error) and gamma (influences data point impact). SVM excels in binary classification by leveraging support vectors and optimising the decision boundary.

4.1.4. XGBoost

XGBoost is a gradient boosting algorithm that builds trees sequentially, correcting prior errors. Key hyperparameters include eta (learning rate), max_depth (tree depth), subsample and colsample_bytree (control training and feature subsampling), and random_state (ensures reproducibility). These configurations improve predictive performance while mitigating overfitting.

4.1.5. CatBoost

CatBoost efficiently handles categorical data without pre-processing. It uses hyperparameters like iterations (number of boosting rounds), learning_rate (step size for weight updates), depth (tree depth), and l2_leaf_reg (regularisation to prevent overfitting). This model simplifies workflows while delivering robust performance on mixed datasets.

4.1.6. Logistic Regression (LR)

LR is a linear classifier suited for binary tasks. It uses liblinear as the solver, allowing L1 (feature selection) and L2 (mitigates outliers) regularisation. With a fixed random_state, results are reproducible. Performance is evaluated through metrics like AUC-ROC, confusion matrix, and F1-score, ensuring reliable predictions. LR is a statistical strategy that models a binary dependent variable using a logistic function. LR determines the probability of a binary response using a functional approach and various features. It employs a nonlinear sigmoid function to determine the parameters that provide the best fit. The sigmoid function (sigma) and its corresponding input (x) are as follows:

$$\sigma(x) = \frac{1}{(1 + l^{-x})}$$

$$x = w_0 z_0 + w_1 z_1 + \cdots + w_n z_n$$

The optimal coefficients w and vector z, representing the input data, were obtained by multiplying each element individually. The result of adding these values is a numerical value that ultimately determines the classification score of the target class. If the sigmoid value is less than 0.5, it is considered to be zero; otherwise, it is 1.

4.2. Deep Learning Techniques

The model architecture consists of 18 layers, each contributing uniquely to the feature extraction and classification process:

1-Conv1D Layer: The architecture begins with a one-dimensional convolutional layer equipped with 32 filters and a kernel size of 3. This layer utilises the ReLU activation function to introduce non-linearity and efficiently capture local patterns in the input sequence. The input shape for this layer is defined as (X_train.shape [1], 1), where X_train.shape [1] represents the number of features in each input sample.

2-Batch Normalisation Layer: Immediately following the first Conv1D layer, Batch Normalisation is applied. This technique normalises the activations of the previous layer; by minimising internal covariate shift, the training process is substantially improved in terms of stability and efficiency. It helps in maintaining a consistent distribution of inputs across layers.

3-MaxPooling1D Layer: The MaxPooling layer is implemented to downscale the feature maps, with a pool size of 2. This layer reduces the spatial dimensions and retains the most significant features while mitigating the computational load and helping to prevent overfitting.

4-Conv1D Layer: The model then incorporates a second Conv1D layer with 64 filters and a kernel size of 3. This layer continues to capture more complex patterns, building on the representations learned by the first convolutional layer.

5-Batch Normalisation Layer: Like the first set of layers, Batch Normalisation is applied to further stabilise the learning process and reduce the dimensionality of the feature maps.

6-MaxPooling1D Layer: Another MaxPooling layer is applied to downsample the feature maps, retaining the most important features while reducing computational complexity and preventing overfitting.

7-Conv1D Layer: The filter size is increased to 128 in the third Conv1D layer, which also maintains a kernel size of 3. This layer allows the model to learn even more abstract and higher-level features from the input sequence, which are crucial for accurate classification.

8-Batch Normalisation Layer: Batch Normalisation is applied again to ensure stable and efficient learning by normalising the activations.

9-MaxPooling1D Layer: A third MaxPooling layer is applied to further reduce the spatial dimensions of the feature maps, retaining the most critical features.

10-Bidirectional LSTM Layer: Following the convolutional layers, the model employs a Bidirectional LSTM (BiLSTM) layer with 128 units. Unlike traditional LSTMs, The BiLSTM algorithm processes the input sequence in both the forward and backward orientations, allowing the model to incorporate dependencies from both past and future contexts. This bidirectional processing is particularly beneficial for understanding the sequential nature of the data and for improving the context-awareness of the model.

11-Dropout Layer: To prevent overfitting, a Dropout layer with a dropout rate of 0.5 is added after the first BiLSTM layer. Randomly, 50% of the neurons are removed during each training iteration using this technique.

12-Bidirectional LSTM Layer: Another Bidirectional LSTM layer with 64 units is then included, providing a more compact representation of the sequential data while still benefiting from bidirectional context.

13-Dropout Layer: An additional Dropout layer with the same dropout rate of 0.5 is applied after the second BiLSTM layer to further reduce overfitting and improve model robustness.

14-Attention Layer: The Attention Layer computes a context vector by focusing on the most relevant parts of the sequence. This layer works by assigning higher weights to the time steps that contribute more significantly to the output prediction. This mechanism enhances the model's interpretability and efficiency by allowing it to focus on the most informative segments.

15-First Dense Layer: An attention-driven fully linked layer with 64 neurons and ReLU activation. This layer incorporates features from preceding levels to teach the model complicated feature interactions.

16-Dropout Layer: Another Dropout layer with a rate of 0.5 is applied to prevent overfitting.

17-Second Dense Layer: A second Dense layer with 32 neurons with ReLU activation reduces dimensionality and prioritises classification criteria.

18-Output Layer: In the final Dense layer, a single neuron is equipped with a sigmoid activation function. This layer generates a probability score ranging from 0 to 1, which denotes the probability of the positive class. Table 2 shows the Deep Learning model structure.

The sequence of layers in this model is meticulously designed to maximise the strengths of different neural network components and to ensure efficient and effective learning. Starting with Conv1D layers is crucial for extracting local patterns. These layers can discover short-term relationships and important local features, which provide the basis for later processing. Introducing Batch Normalisation early stabilises the learning process. The computational intricacy is reduced by MaxPooling, which reduces the dimensionality of the feature maps while retaining essential features. Using deeper Conv1D layers, by progressively increasing the number of filters in subsequent Conv1D layers, the model can capture more complex and abstract patterns. This hierarchical feature extraction is essential for understanding the underlying structure of the data. Following the convolutional layers with BiLSTM layers allows the model to capture long-term dependencies and contextual information from both past and future states. The bidirectional nature of these layers enhances the model's ability to understand the sequential context of the data. Placing the Attention Layer after the BiLSTM layers allows the model to dynamically focus on the most relevant parts of the sequence, improving interpretability and performance. The attention mechanism ensures that the model gives more weight to important time steps, enhancing its predictive capabilities. Dense layers following the attention mechanism integrate and transform the features extracted by the previous layers. These layers, combined with Dropout, reduce the dimensionality and focus on the most relevant aspects for classification,

ensuring robust and accurate predictions. Finally, the sigmoid activation function in the output layer is well-suited for binary classification, as it effectively maps the input to a probability score between 0 and 1.

The model's exhaustive comprehension of the data is guaranteed by the combination of Conv1D and BiLSTM layers, which capture both local patterns and long-term dependencies. Batch Normalisation and MaxPooling layers enhance the stability and efficiency of the model, ensuring faster and more reliable training. The attention mechanism enhances the model's interpretability and efficiency by allowing it to focus on the most informative parts of the sequence, improving overall performance. The inclusion of Dropout layers mitigates overfitting, promoting better generalisation to unseen data. In summary, this hybrid model combines the strengths of CNN and BiLSTM architectures with an attention mechanism to effectively capture and prioritise important features in sequential data. This design leads to improved performance in binary classification tasks by leveraging local pattern recognition, long-term dependency modelling, and dynamic attention-based feature weighting. The careful selection and ordering of layers, along with robust algorithmic components, make this model a powerful tool for tackling complex binary classification challenges. The Adam optimiser is selected due to its efficient management of sparse gradients and adaptive learning rate capabilities. The binary cross-entropy loss function measures the difference between predicted probabilities and class labels, making it suited for binary classification applications. This loss function is particularly effective for models outputting probabilities, ensuring that the model's predictions are calibrated accurately. The ReLU activation function is employed in the hidden layers due to its ability to introduce non-linearity. ReLU is computationally efficient and helps in mitigating the vanishing gradient problem, allowing the model to learn more effectively.

Table 2. The Deep Learning structure.

Layer (Type)	Output Shape	Param #
conv1d (Conv1D)	(None, 15, 32)	128
batch_normalization (BatchNormalization)	(None, 15, 32)	128
max_pooling1d (MaxPooling1D)	(None, 7, 32)	0
conv1d_1 (Conv1D)	(None, 5, 64)	6208
batch_normalization_1 (BatchNormalization)	(None, 5, 64)	256
max_pooling1d_1 (MaxPooling1D)	(None, 2, 64)	0
conv1d_2 (Conv1D)	(None, 2, 128)	24,704
batch_normalization_2 (BatchNormalization)	(None, 2, 128)	512
max_pooling1d_2 (MaxPooling1D)	(None, 1, 128)	0
Bidirectional (Bidirectional)	(None, 1, 256)	263,168
Dropout (Dropout)	(None, 1, 256)	0
bidirectional_1 (Bidirectional)	(None, 1, 128)	164,352
dropout_1 (Dropout)	(None, 1, 128)	0
attention_layer (AttentionLayer)	(None, 128)	129
dense (Dense)	(None, 64)	8256
dropout_2 (Dropout)	(None, 64)	0
dense_1 (Dense)	(None, 32)	2080
dense_2 (Dense)	(None, 1)	33
Total params: 469,954 Trainable params: 469,506 Non-trainable params: 448		

The Algorithm 1 describes a novel hybrid stacking model that combines ML and DL techniques for classification. The process begins by pre-processing the dataset, which involves cleaning, encoding, and normalising the features. Afterward, the data are split into training and testing sets, with the features and target values separated. The training and testing sets are then normalised to ensure that all features contribute equally to the model. The core of the algorithm is the creation of a stacking classifier, where multiple base learners are employed. These learners include RF, SVM, LR, DT, XGBoost, CatBoost, and a hybrid CNN-BiLSTM-Attention DL model. The stacking model uses these base learners to make predictions, which are then combined by a meta-classifier, typically an RF. To evaluate the model, the algorithm applies cross-validation using StratifiedKFold to preserve class distribution in each fold. The model is trained and evaluated over multiple iterations, with performance metrics being recorded and printed after each iteration. The procedure repeats this process n times, ensuring robust evaluation of the stacking model's performance on the test data. This hybrid approach is designed to leverage the strengths of both ML and DL models for improved classification accuracy and generalisation.

Algorithm 1: Novel hybrid stacking ML+DL model

```
Procedure Stacking Hybrid ML+DL _model (X, y, n, cv_folds, test_size)
    Pre-process (X, y)
    Split data into (X, y)
    Normalize (X_train, X_test)
    Model ← Create stacking classifier with RandomForestClassifier
              as meta-classifier()
         ('Random Forest', rf_classifier)
         ('SVM', svm_classifier)
         ('Logistic Regression', lr_classifier)
         ('Decision Tree', dt_classifier)
         ('XGBoost', xgb_classifier)
         ('CatBoost', catboost_classifier)
         ('CNN-BiLSTM-Attention', CNN-BiLSTM-Attention _classifier)

      Cross-validation:  StratifiedKFold with cv_folds
for k ← 0 to n-1 do
    Train StackingClassifier (X_train, y_train)
    Predict y_pred on (X_test)
    Evaluate Model(X_test, y_test)
    Print Evaluation Metrics

end for
End Procedure
```

4.3. Data Processing

To facilitate the development of the cyber fraud detection model, a dataset containing credit card transaction information was obtained and loaded into the analysis environment. This step is essential for understanding the nature of the data. Credit card transactions totalling 284,807 were conducted in the real world over the course of 24 h. The dataset is highly imbalanced. Out of 284,315 records, only 492 are labelled as fraudulent transactions. There are 31 columns in this dataset, 30 columns representing attributes and one column representing the target class, which shows whether a transaction is fraudulent or genuine. The dataset comprises 28 variables, which have been converted from the original set of variables using Principal Component Analysis (PCA). The dataset file is formatted in the Comma-Separated Values (CSV) format. The CSV file can be read using the pandas module

in Python. Cyber fraud detection systems often encounter an amount of highly imbalanced data, where most credit card transactions are genuine but just a small fraction is fraudulent. Figure 3 shows the percentage of fraudulent vs non-fraudulent transactions.

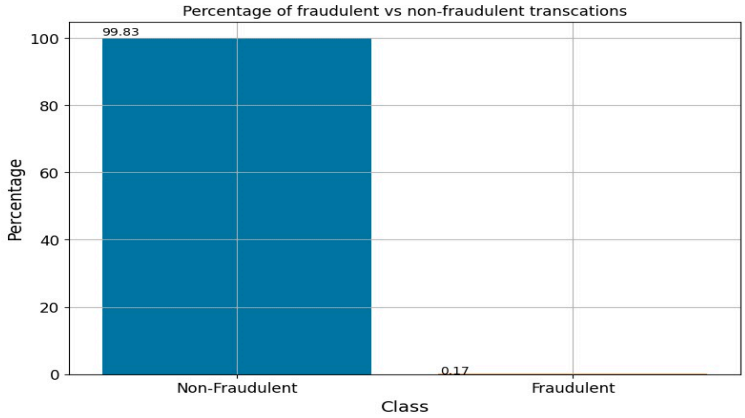

Figure 3. The percentage of fraudulent vs non-fraudulent transactions.

Because of the dataset's extreme imbalance, with real transactions accounting for 99.827% and cyber fraud transactions for just 0.173%, training the model using raw data is unlikely to yield the desired outcomes, despite potentially high evaluation metrics. Figure 4 shows the plot of the credit card dataset.

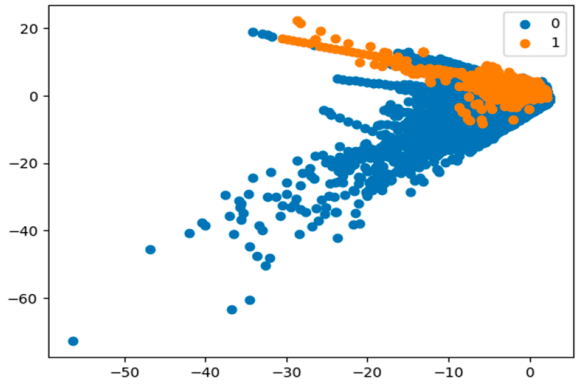

Figure 4. The plot of the credit card dataset.

It is imperative to pre-process the data before executing an ML algorithm. This is because the predictors are generated with unique requirements by various models, and the prediction output may be influenced by the data training. The dataset is composed of numerical values that are acquired through the PCA process. Nevertheless, the original characteristics have not been released because of the confidentiality concerns. A total of 30 features were created, with 28 of them being derived by PCA. PCA is a widely used method for reducing the dimensionality of data. 'Amount' and 'Time' are the only attributes that have not been converted into principal components. The pre-processing tasks have been accomplished by utilising the Python data manipulation package pandas and the ML module sci-kit learn. The sequential process is visually depicted in Figure 5.

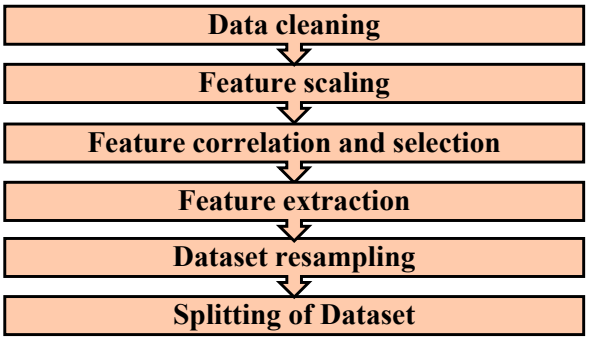

Figure 5. The data pre-processing steps.

4.3.1. Data Cleaning

In the course of data cleansing, two primary activities are frequently implemented. The initial task entails the elimination of null values and absent values from the dataset. The second responsibility is the management of outliers, which are data points that deviate considerably from the majority of the dataset. In total, the dataset contains 284,807 transactions. No null values were present in the dataset.

4.3.2. Feature Scaling

Data pre-processing includes this step to normalise a dataset's independent variables. It is centred around 0 or between 0 and 1, depending on the scaling mechanism. Feature scaling improves the performance of models by ensuring feature scales are similar. We developed feature scaling using RobustScaler. This procedure scales features in X_train and X_test using RobustScaler. The RobustScaler fitted to the training data (X_train) will scale its features.

4.3.3. Feature Correlation and Selection

Feature selection is a critical component of ML and data analysis, with the primary objective of identifying the most pertinent features to improve interpretability and improve model performance. Since RF, XGBoost, and permutation methods are often performed well for cyber fraud detection tasks, utilising four distinct feature importance techniques, including the correlation matrix, RF, XGBoost, and permutation analysis, we evaluate the significance of features. Following the analysis with each method, we compare the results and aggregate the top 17 important features. The features are V17, V14, V12, V10, V16, V3, V7, V11, V4, V18, V1, V9, V5, V2, V6, V21, and V19. These selected features are then designated for utilisation in subsequent stages of the process.

4.3.4. Feature Extraction

A technique like PCA is used for dimensionality reduction. Feature extraction helps reduce noise, prevent overfitting, and lower computational costs by selecting and transforming the most relevant data. Ultimately, this step enhances model accuracy and generalisation by simplifying the input data. We applied PCA as a dimensionality reduction algorithm on our dataset to produce robust and discriminative features for detecting fraudulent transactions.

4.3.5. Data Splitting

This method serves two crucial purposes: mitigating the risk of overfitting and verifying the performance of the model in real-world situations. The complete dataset is partitioned into a training set comprising 80% of the data and a test set including the remaining 20%.

5. Results and Discussion

Initially, we assess the performance of ML algorithms individually and in the absence of employing ensemble techniques. In datasets with class imbalances, such as fraud detection, the F1 score provides a balanced evaluation by harmonising precision (minimising false positives) and recall (minimising false negatives). Unlike accuracy, which can be misleading in such cases, the F1 score focuses on the minority class while addressing both critical error types. This makes it a nuanced metric for assessing model effectiveness, especially when both detecting fraud and avoiding legitimate transaction misclassification are crucial. The outcomes derived from this evaluation are comprehensively depicted in Table 3, illustrating a comparative analysis of the algorithms. Figure 6 shows the performance of ML algorithms without ensemble techniques.

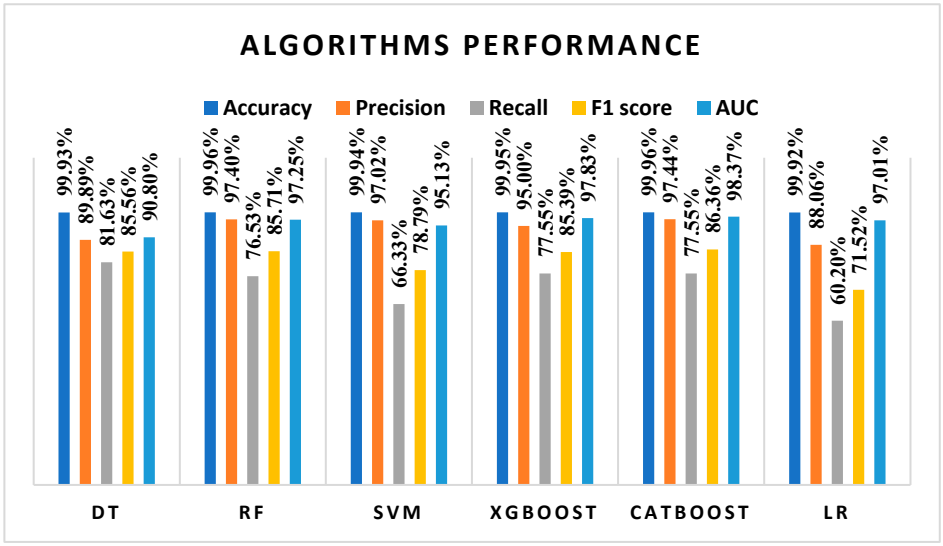

Figure 6. Performance of ML algorithms.

Table 3. Algorithms' performance.

ML	Accuracy	Precision	Recall	F1 Score	AUC
DT	99.93%	89.89%	81.63%	85.56%	90.80%
RF	99.96%	97.40%	76.53%	85.71%	97.25%
SVM	99.94%	97.02%	66.33%	78.79%	95.13%
XGBoost	99.95%	95.00%	77.55%	85.39%	97.83%
CatBoost	99.96%	97.44%	77.55%	**86.36%**	98.37%
LR	99.92%	88.06%	60.20%	71.52%	97.01%

The DT model achieves high accuracy (99.93%) and precision (89.89%) but has a lower recall (81.63%), resulting in an F1 score of 85.56%. RF delivers exceptional accuracy (99.96%) and precision (97.40%) but falls short on recall (76.53%), with an F1 score of 85.71%. Similarly, SVM shows strong accuracy (99.94%) and precision (97.02%) but lower recall (66.33%), yielding an F1 score of 78.79%. XGBoost balances performance with 99.95% accuracy, 95.00% precision, and an F1 score of 85.39%. CatBoost excels with 99.96% accuracy, 97.46% precision, and the highest F1 score of 87.01%, indicating an excellent balance between precision and recall. In contrast, LR achieves 99.92% accuracy but lower precision (88.06%) and recall (60.20%), resulting in a lower F1 score of 71.52%. Overall, CatBoost emerges as the top performer, followed by RF and XGBoost, while LR and SVM lag in capturing positive instances effectively.

Subsequently, we evaluate the performance of DL algorithms independently, without incorporating ensemble techniques. This evaluation focuses on understanding the effectiveness of standalone DL methods in detecting credit card cyber fraud, particularly in handling sequential data and uncovering intricate patterns. The results of this assessment are summarised in Table 4 for CNN and Table 5 for BiLSTM. Figures 7 and 8 visually represent the confusion matrix for CNN and BiLSTM.

The comparison between CNN and BiLSTM models highlights key differences in performance, with the F1 score as the primary focus. For the CNN model, an F1 score of 82.80% was achieved with an epoch size of 20 and a batch size of 64, improving slightly to 83.98% with larger epoch and batch sizes (50 and 128). The improvement was driven by increased precision (91.57%), though recall slightly decreased (77.55%), indicating strong identification of fraud but limited coverage of all cases. In contrast, the BiLSTM model excelled, particularly at larger configurations. Its F1 score increased from 82.16% (epoch size 20, batch size 64) to 85.86% (epoch size 50, batch size 128), with higher recall (83.67%) and solid precision (88.17%). This demonstrates BiLSTM's superior balance and effectiveness in detecting fraudulent transactions compared to CNN, particularly for imbalanced datasets.

Table 4. Results of CNN model using several epochs.

Matrix	Epoch Size 20, Batch Size 64	Epoch Size 50, Batch Size 128
Loss	0.002668	0.001816
TP	77	76
FP	11	7
TN	56,853	56,857
FN	21	22
Accuracy	99.94%	99.95%
Precision	87.50%	91.57%
Recall	78.57%	77.55%
Cross-Validation/Mean Accuracy	99.93%	99.94%
F1 score	82.80%	83.98%
AUC	89.28%	88.77%
PRC	68.79%	71.05%
Total fraudulent transaction	98	98

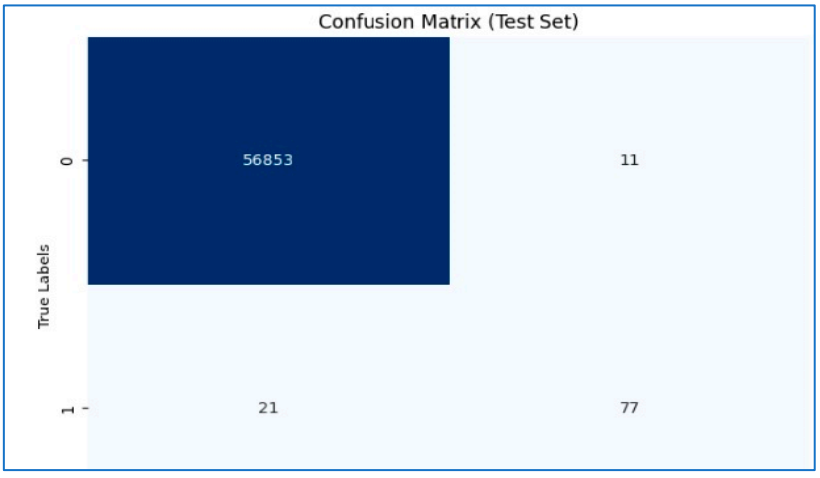

Figure 7. Confusion matrix of CNN model.

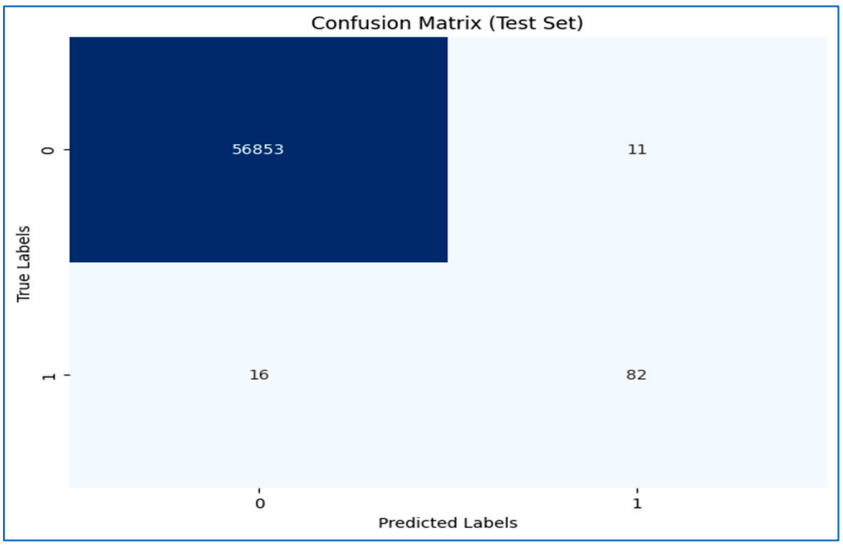

Figure 8. Confusion matrix of BiLSTM model.

Table 5. Results of BiLSTM model using several epochs.

Matrix	Epoch Size 20, Batch Size 64	Epoch Size 50, Batch Size 128
Loss	0.00295	0.002386
TP	76	82
FP	11	11
TN	56853	56853
FN	22	16
Accuracy	99.94%	99.95%
Precision	87.36%	88.17%
Recall	77.55%	83.67%
Cross-Validation/Mean Accuracy	99.94%	99.94%
F1 score	**82.16%**	**85.86%**
AUC	88.77%	91.83%
PRC	67.78%	73.80%
Total fraudulent transaction	98	98

5.1. The Hybrid ML+DL

Initially, we evaluate the performance of the novel hybrid stacking ML+DL model without utilising any sampling techniques. The results from this evaluation are thoroughly detailed in Table 6. Additionally, Figure 9 presents the confusion matrix.

Table 6. Hybrid ML+DL model performance.

Model	Accuracy	Precision	Recall	F1 Score	AUC
Hybrid stacking ML+DL	99.97%	97.62%	83.67%	**90.11%**	91.83%

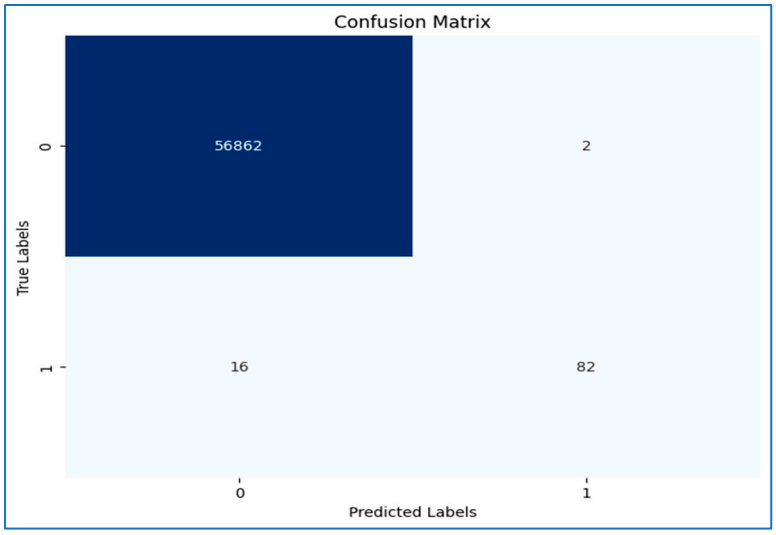

Figure 9. Confusion matrix hybrid ML+DL model.

The novel hybrid ML+DL model exhibits outstanding performance across several assessment measures. With an accuracy of 99.97%, the model showcases its ability to accurately classify instances, which is crucial in credit card cyber fraud detection where even minor errors can have significant consequences. Additionally, the model achieves a high precision of 97.62%, indicating a low rate of false positives. This is particularly advantageous in cyber fraud detection, where correctly identifying fraudulent transactions is paramount to minimising financial losses for both customers and financial institutions.

Moreover, the model exhibits a commendable recall of 83.67%, highlighting its capability to capture a high proportion of actual positive cases. In cyber fraud detection scenarios, where the number of fraudulent transactions is typically much lower than legitimate ones, a high recall ensures that the model effectively identifies fraudulent activities, thereby enhancing cyber fraud detection efficiency. The F1 score of 90.11% further emphasises the model's balanced performance between precision and recall, demonstrating its ability to manage false positives and false negatives. The AUC value of 91.83% underscores the model's ability to distinguish between fraudulent and legitimate transactions effectively. A high AUC suggests that the model performs well across various threshold values, further reinforcing its reliability in making accurate predictions.

The novel hybrid ML+DL model excels in its intricate design, which effectively integrates the advantages of both classic ML and DL techniques. The ML techniques incorporated, such as DT, RF, SVM, XGBoost, CatBoost, and LR, offer a diverse set of tools tailored to address specific challenges in credit card cyber fraud detection. These techniques leverage various algorithms and strategies to effectively capture patterns and make accurate predictions. Furthermore, the DL techniques utilised, including CNN and BiLSTM networks, are well-suited for handling sequential data such as transaction sequences. The integration of attention mechanisms further enhances the model's ability to focus on relevant segments of the input sequence, improving interpretability and performance. Overall, the novel hybrid ML+DL model benefits from comprehensive feature extraction, stability, efficiency, dynamic attention mechanisms, robustness, and generalisation capabilities. It is an effective tool for addressing intricate binary classification challenges, particularly in credit card cyber fraud detection scenarios, due to its high-level architecture, meticulous layer sequencing, and robust algorithmic components.

5.2. Comparison Between Hybrid ML+DL and Individual Techniques:

The comparison of performance across ML techniques, DL techniques, and the hybrid ML+DL model highlights significant advancements in detecting credit card fraud, particularly when focusing on the F1 score as a key measure of effectiveness. Each approach demonstrates unique strengths, but the hybrid model clearly outperforms standalone methods by leveraging their complementary capabilities.

Among the evaluated ML algorithms, CatBoost demonstrated the best performance, achieving an F1 score of 86.36%, closely followed by RF with 85.71%. These models also excelled in precision, with CatBoost and RF achieving 97.44% and 97.40%, respectively. However, both struggled with lower recall rates (77.55% and 76.53%), indicating limitations in identifying all fraudulent transactions. Other ML algorithms, such as DT and LR, performed less effectively, with F1 scores of 85.56% and 71.52%, respectively. These results show that, while ML models are highly precise, they may fail to capture a significant portion of fraudulent cases, which is critical in cyber fraud detection.

DL techniques offered a competitive edge, particularly in handling sequential data and capturing intricate patterns. Among the DL methods, BiLSTM outperformed CNN, achieving an F1 score of 85.86% at larger epoch and batch sizes (50 and 128), compared to CNN's 83.98%. BiLSTM's superior recall of 83.67% highlights its strength in detecting a broader range of fraudulent transactions, making it more reliable in minimising false negatives. On the other hand, CNN demonstrated slightly higher precision (91.57% vs. BiLSTM's 88.17%), indicating better accuracy in correctly identifying fraud but with a slight compromise in recall. Overall, while DL methods show promise, they are most effective when optimised for the characteristics of fraud detection datasets.

The hybrid ML+DL model achieved the highest overall performance, combining the strengths of both ML and DL techniques. With an F1 score of 90.11%, it surpassed all standalone models, balancing precision (97.62%) and recall (83.67%) more effectively. By integrating ML algorithms, such as RF and CatBoost, with DL architectures like CNN and BiLSTM using a stacking ensemble framework, the hybrid model leveraged their complementary strengths. Its superior accuracy (99.97%) and robust handling of imbalanced datasets demonstrate its ability to detect fraudulent transactions more reliably than individual methods. The hybrid approach's success highlights its ability to mitigate the weaknesses of standalone models while capitalising on their unique advantages.

The analysis demonstrates that, while ML and DL techniques independently offer strong capabilities, they are limited in their ability to address all aspects of credit card fraud detection. The hybrid ML+DL model, by combining the precision of ML with the sequential data processing capabilities of DL, delivers the most balanced and effective solution. This underscores the potential of hybrid approaches to set a new benchmark in fraud detection, ensuring higher accuracy, better generalisation, and improved robustness in real-world applications.

The novel hybrid stacking ML+DL model demonstrates superior performance compared to the most advanced models in terms of accuracy and F1 score on the European dataset. While some models achieve high accuracy individually, the novel hybrid approach, which combines ML and DL through stacking, achieves the highest accuracy of 99.97% and a competitive F1 score of 90.11%. This suggests that the integration of both ML and DL techniques in a stacking framework enhances the model's predictive capabilities, offering promising results for card cyber fraud detection. Table 7 shows a comparison of performance with existing models.

Table 7. Comparison of performance with existing models.

Study Ref.	Model	Accuracy	F1 Score	Dataset
[32]	LSTM	99.95%	---	European cards
[40]	BiLSTM	91.37%	---	European cards
[34]	LSTM-RNN	99.58%	88.76%	European cards
[29]	CNN-BiLSTM	95%	---	European cards
[28]	CNN	99.72%	---	European cards
[41]	CNN-ELM	98.7%	---	European cards
[42]	CNN	99.81	83.72%	European cards
[35]	RNN-LSTM-Attention	99.4%	---	European cards
[43]	Hybrid ML+BCBSMOTE	---	85.20%	European cards
NovelHybrid ML+DL model	Stacking ML+DL	99.97%	90.11%	European cards

5.3. The Hybrid ML+DL with Sampling:

The novel hybrid stacking ML+DL model employs a variety of resampling techniques to address the issue of class imbalance in our dataset. Class imbalance is a common issue in real-world applications such as cyber fraud detection [44], where the minority class (e.g., fraudulent transactions) is significantly underrepresented compared to the majority class [45]. To address this, we employ Borderline-SMOTE (Borderline Synthetic Minority Oversampling Technique), ROS (RandomOverSampler), Tomek Link, ENN, and SMOTEENN (SMOTE + ENN) as strategies to rebalance our dataset and enhance the performance of our hybrid stacking ML+DL model.

A. Borderline-SMOTE

In the novel hybrid stacking ML+DL model, Borderline-SMOTE addresses class imbalance by generating synthetic samples near the decision boundary, where misclassifications often occur. It focuses on difficult-to-classify minority class instances, enhancing the classifier's ability to learn clear and accurate decision boundaries [46]. This method is applied to the training data to balance classes before model training.

B. RandomOverSampler (ROS)

RandomOverSampler (ROS) balances the dataset by randomly duplicating minority class instances until both classes are equally represented. By providing a balanced training dataset, ROS helps the ML and DL components of the hybrid model learn more effectively.

C. Tomek Links

Tomek Links improves class separability by removing ambiguous instances near the decision boundary. Identifying and eliminating pairs of nearest neighbours from different classes reduces noise, helping the classifier focus on more distinct class patterns.

D. Edited Nearest Neighbours (ENN)

ENN cleans the dataset by removing noisy or ambiguous instances, especially near decision boundaries. It iteratively eliminates misclassified instances based on their k nearest neighbours, ensuring the dataset emphasises clear class boundaries. This refinement enhances the model's ability to detect minority class instances accurately.

E. SMOTEENN

SMOTEENN combines SMOTE and ENN to handle class imbalance. SMOTE generates synthetic minority class samples to balance the dataset, while ENN removes noisy or borderline instances, improving class distinction. This two-step process ensures the model is trained on a refined and balanced dataset, enhancing its robustness and predictive accuracy.

We employed several resampling techniques, oversampling, undersampling, and a combination of oversampling and undersampling, to address the significant class imbalance present in the dataset. Results from our hybrid stacking ML+DL model integrated with the several resampling techniques are shown in Table 8. Figure 10 shows the f1 score for the sampling techniques with the novel hybrid stacking ML+DL model. Figure 11 shows the performance of the sampling techniques with the novel hybrid stacking ML+DL model.

Table 8. Performance of resampling techniques.

	Model	Accuracy	Precision	Recall	F1 Score
With sampling	Hybrid ML+DL+Bordersmote	94.90%	99.95%	89.85%	94.63%
	Hybrid ML+DL+ROS	93.36%	99.96%	86.75%	92.89%
	Hybrid ML+DL+Tomek	99.95%	1.0	71.43%	83.33%
	Hybrid ML+DL+ENN	99.96%	98.70%	77.55%	86.86%
	Hybrid ML+DL+SMOTEEEN	94.24%	99.92%	88.56%	93.90%
Without sampling	Hybrid ML+DL	99.97%	97.62%	83.67%	90.11%

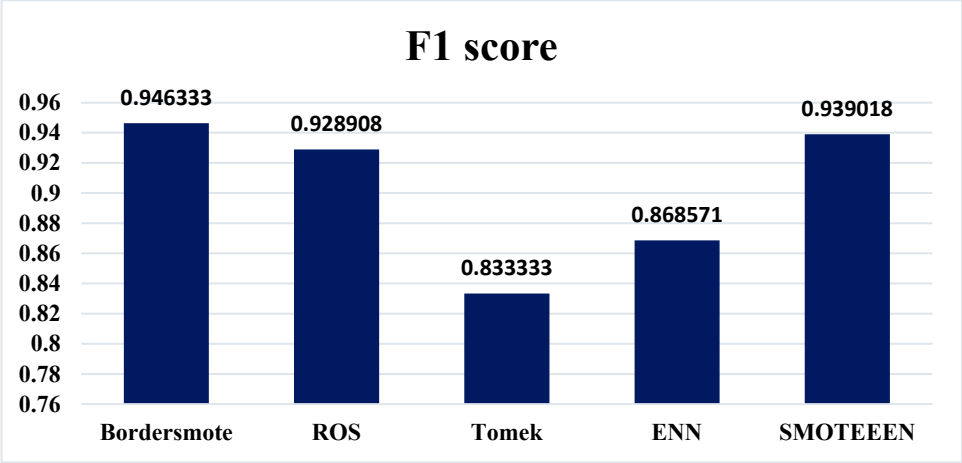

Figure 10. F1 score of sampling with the novel hybrid ML+DL model.

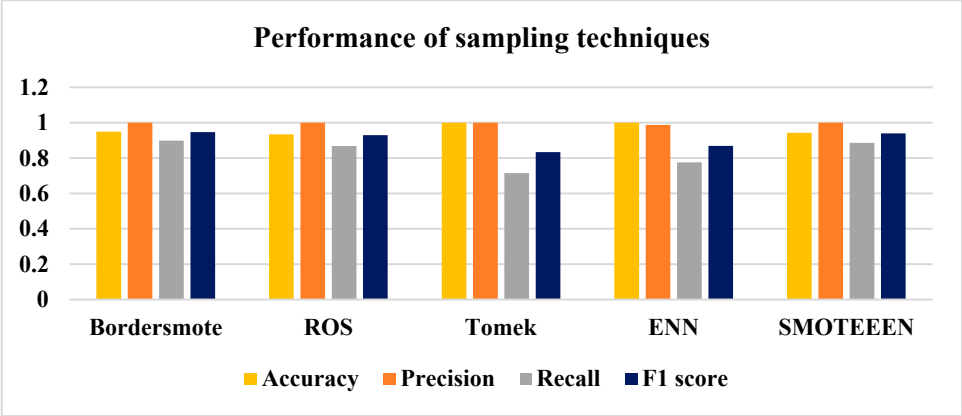

Figure 11. Performance of sampling with the hybrid ML+DL model.

The hybrid stacking ML+DL models integrated with resampling techniques show significant advancements in cyber fraud detection by addressing class imbalance and false

positives while achieving strong accuracy and F1 scores. Each model balances strengths and trade-offs to suit different scenarios. The Borderline-SMOTE model achieved an F1 score of 94.63% with high precision (99.95%) and recall (89.85%), balancing fraud detection and minimising false positives. ROS yielded an F1 score of 92.89%, precision of 99.96%, and recall of 86.75%, providing robust performance with slightly lower recall than Borderline-SMOTE. Tomek Links ensured no false positives with 100% precision and 99.95% accuracy but had a lower recall (71.43%), leading to an F1 score of 83.33%. ENN delivered an F1 score of 86.86% with 98.70% precision and 77.55% recall, reflecting reliable performance but with room to improve recall. SMOTEENN achieved an F1 score of 93.90% with precision of 99.93% and recall of 88.56%, offering a strong balance between precision and recall. Overall, Borderline-SMOTE and SMOTEENN excel, highlighting the effectiveness of advanced resampling techniques in enhancing detection accuracy and robustness.

5.4. Comparison of Performance Based on F1 Score with and Without Sampling

In our evaluation, the F1 score served as a critical measure of our hybrid stacking ML+DL model 's performance with and without various sampling techniques. The results indicate a notable variation in the model's performance depending on the sampling method applied.

The comparison between the hybrid ML+DL model with and without sampling highlights the significant impact of sampling techniques on performance metrics. Without sampling, the hybrid model achieves an F1 score of 90.11%, with high accuracy (99.97%) and precision (97.62%) but a lower recall (83.67%) compared to models with sampling. In contrast, applying sampling techniques like Borderline-SMOTE and SMOTEENN enhances the F1 score to 94.63% and 93.90%, respectively, by improving recall to 89.85% and 88.56%, while maintaining exceptional precision (over 99.90%). Other sampling methods, such as ROS, also improve the F1 score (92.89%) and recall (86.75%), though techniques like Tomek Links and ENN prioritise precision, achieving 100% and 98.70%, respectively, but at the cost of lower recall (71.43% and 77.55%). Overall, sampling techniques significantly enhance the model's ability to detect fraudulent transactions (recall), with Borderline-SMOTE and SMOTEENN offering the best balance between precision and recall, resulting in superior F1 scores. While the novel hybrid stacking ML+DL model performs well on its own, the inclusion of appropriate sampling methods, such as Bordersmote and SMOTEENN, can further optimise its performance, making it more adept at identifying fraudulent activities while minimising false positives.

In summary, the strength of the novel hybrid stacking ML+DL model lies in its sophisticated architecture that leverages both traditional ML techniques and advanced DL approaches. Incorporating a variety of ML techniques such as DT, RF, SVM, XGBoost, CatBoost, and LR, the model effectively captures patterns and makes accurate predictions. Additionally, the integration of CNNs and BiLSTM networks, equipped with attention mechanisms, enables the model manage transaction sequences. This comprehensive approach ensures dynamic feature extraction, stability, efficiency, and interpretability, making the novel hybrid stacking ML+DL model a potent instrument for addressing intricate binary classification challenges in the detection of credit card cyber fraud.

6. Conclusions

Detecting credit card cyber fraud is vital for financial security as fraud tactics grow more sophisticated. This research tackles challenges like class imbalance and evolving fraud patterns by developing a hybrid stacking ML+DL model that integrates ML, DL, and advanced sampling techniques. Evaluated primarily on the F1 score, the model achieves an impressive 91.11 without sampling and 94.63% with sampling, demonstrating an excellent

balance between precision and recall, making it highly effective in detecting true cases while minimising false positives and negatives. The model's robust performance and scalability provide financial institutions with a powerful tool to automate fraud detection, enhance security, and reduce costs. It not only outperforms baseline models but also addresses the critical challenge of imbalanced datasets, making it suitable for real-world applications. Additionally, its ability to adapt to evolving fraud patterns ensures long-term relevance in dynamic environments.

While limited by a single dataset, future research should prioritise validating models on multiple datasets to ensure consistent performance across various data sources, enhancing the generalisability of the proposed methodologies to different conditions and types of fraudulent activities. Additionally, advanced DL techniques like GANs and autoencoders, along with real-time implementation, should be explored to improve adaptability and scalability. The model's performance is also sensitive to pre-processing methods and hyperparameter tuning, and performance varies with decision threshold adjustments, requiring a balance between precision and recall. This study highlights the hybrid ML+DL model as a transformative solution for cyber fraud detection, paving the way for stronger, more secure financial systems to protect businesses and consumers from evolving threats.

Author Contributions: Conceptualisation, E.B. and X.Z.; methodology, E.B., X.Z. and K.C.C.; software, E.B.; validation, E.B., X.Z. and K.C.C.; formal analysis, E.B. and R.G.; investigation, E.B., X.Z. and R.G.; resources, E.B. and K.C.C.; data curation, E.B. and O.A.; writing—original draft preparation, E.B.; writing—review and editing, E.B., X.Z. and O.A.; visualisation, E.B. and X.Z.; supervision, X.Z., R.G. and K.C.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The dataset is available at https://www.kaggle.com/mlg-ulb/creditcardfraud (accessed on 1 January 2025). The dataset consisted of client transactions at a European bank in 2013. The real-world dataset consists of 284,807 credit card transactions.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Carbo-Valverde, S.; Cuadros-Solas, P.; Rodríguez-Fernández, F. A machine learning approach to the digitalization of bank customers: Evidence from random and causal forests. *PLoS ONE* **2020**, *15*, e0240362. [CrossRef] [PubMed]
2. Bagga, S.; Goyal, A.; Gupta, N.; Goyal, A. Credit card fraud detection using pipeling and ensemble learning. *Procedia Comput. Sci.* **2020**, *173*, 104–112. [CrossRef]
3. Merchant Cost Consulting. Credit Card Fraud Statistics 21-Merchantcostconsulting. 2024. Available online: https://merchantcostconsulting.com/lower-credit-card-processing-fees/credit-card-fraud-statistics (accessed on 12 December 2023).
4. Australian Bureau of Statistics 2022-23-Financial-Year, Personal Fraud, ABS, Viewed 15 May 2024, Australian Bureau of Statistics Reveal Details of 'Sizeable' Increase in Card Fraud as Australians Lose $2.2 Billion in 2023—ABC News. Available online: https://www.abc.net.au/news/2024-03-20/abs-card-fraud-scam-data/103609822 (accessed on 1 January 2025).
5. Chatterjee, P.; Das, D.; Rawat, D.B. Digital twin for credit card fraud detection: Opportunities, challenges, and fraud detection advancements. *Future Gener. Comput. Syst.* **2024**, *158*, 410–426. [CrossRef]
6. Btoush, E.A.L.M.; Zhou, X.; Gururajan, R.; Chan, K.C.; Genrich, R.; Sankaran, P. A systematic review of literature on credit card cyber fraud detection using machine and deep learning. *PeerJ Comput. Sci.* **2023**, *9*, e1278. [CrossRef] [PubMed]
7. Alhowaide, A.; Alsmadi, I.; Tang, J. PCA, Random-forest and pearson correlation for dimensionality reduction in IoT IDS. In Proceedings of the 2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), Vancouver, BC, Canada, 9–12 September 2020.
8. Btoush, E.; Zhou, X.; Gururaian, R.; Chan, K.C.; Tao, X. A survey on credit card fraud detection techniques in banking industry for cyber security. In Proceedings of the 2021 8th International Conference on Behavioral and Social Computing (BESC), Doha, Qatar, 29–31 October 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1–7.

9. Mienye, I.D.; Jere, N. Deep learning for credit card fraud detection: A review of algorithms, challenges, and solutions. *IEEE Access* **2024**, *12*, 96893–96910. [CrossRef]
10. Reddy, N.M.; Sharada, K.A.; Pilli, D.; Paranthaman, R.N.; Reddy, K.S.; Chauhan, A. CNN-Bidirectional LSTM based Approach for Financial Fraud Detection and Prevention System. In Proceedings of the 2023 International Conference on Sustainable Computing and Smart Systems (ICSCSS), Coimbatore, India, 14–16 June 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 541–546.
11. Jainish, G.R.; Alwin Infant, P. Attention layer integrated BiLSTM for financial fraud prediction. Multimedia Tools and Applications. *Multimedia Tools Appl.* **2024**, *83*, 80613–80629.
12. Sharifani, K.; Amini, M. Machine learning and deep learning: A review of methods and applications. *World Inf. Technol. Eng. J.* **2023**, *10*, 3897–3904.
13. Afriyie, J.K.; Tawiah, K.; Pels, W.A.; Addai-Henne, S.; Dwamena, H.A.; Owiredu, E.O.; Ayeh, S.A.; Eshun, J. A supervised machine learning algorithm for detecting and predicting fraud in credit card transactions. *Decis. Anal. J.* **2023**, *6*, 100163. [CrossRef]
14. Reddy, S.T.S.; Sriramya, P. Comparison of the Support Vector Classifier algorithm with the Decision Tree algorithm for Credit Card Fraud Detection with the Goal of Improving Accuracy. *J. Surv. Fish. Sci.* **2023**, *10*, 2304–2313. [CrossRef]
15. Nama, F.A.; Obaid, A.J.; Alrammahi, A.A.H. *Credit Card Fraud Detection and Classification Using Deep Learning with Support Vector Machine Techniques*; Swaroop, A., Polkowski, Z., Correia, S.D., Virdee, B., Eds.; Proceedings of Data Analytics and Managemen; Springer: Singapore, 2023.
16. Kırelli, Y.; Arslankaya, S.; Zeren, M.T. Detection of credit card fraud in e-commerce using data mining. *Avrupa Bilim Ve Teknol. Derg.* **2020**, *20*, 522–529. [CrossRef]
17. Lim, K.S.; Lee, L.H.; Sim, Y.-W. A review of machine learning algorithms for fraud detection in credit card transaction. *International J. Comput. Sci. Netw. Secur.* **2021**, *21*, 31–40. [CrossRef]
18. Bandyopadhyay, S.; Thakkar, V.; Mukherjee, U.; Dutta, S. Emerging approach for detection of financial frauds using machine learning. *Asian J. Res. Comput. Sci.* **2021**, *11*, 9–22. [CrossRef]
19. Barahim, A.; Alhajri, A.; Alasaibia, N.; Altamimi, N.; Aslam, N.; Khan, I.U. Enhancing the credit card fraud detection through ensemble techniques. *J. Comput. Theor. Nanosci.* **2019**, *16*, 4461–4468. [CrossRef]
20. Amusan, E.; Alade, O.; Fenwa, O.; Emuoyibofarhe, J. Credit card fraud detection on skewed data using machine learning techniques. *Lautech J. Comput. Inform.* **2021**, *2*, 49–56.
21. Ata, O.; Hazim, L. Comparative analysis of different distributions dataset by using data mining techniques on credit card fraud detection. *Teh. Vjesn.* **2020**, *27*, 618–626. [CrossRef]
22. Choubey, R.; Gautam, P. Combined technique of supervised classifier for the credit card fraud detection. *Shodah Sarita* **2020**, *7*, 27–32.
23. Hema, A.; Muttipati, A. Machine learning methods for discovering credit card fraud. *Int. Res. J. Comput. Sci.* **2020**, *8*, 1–6.
24. Alenzi, H.Z.; Aljehane, N.O. Fraud detection in credit cards using logistic regression. *Int. J. Adv. Comput. Sci. Appl.* **2020**, *11*. [CrossRef]
25. Khalid, A.R.; Owoh, N.; Uthmani, O.; Ashawa, M.; Osamor, J.; Adejoh, J. Enhancing credit card fraud detection: An ensemble machine learning approach. *Big Data Cogn. Comput.* **2024**, *8*, 6. [CrossRef]
26. Faraj, A.A.; Mahmud, D.A.; Rashid, B.N. Comparison of different ensemble methods in credit card default prediction. *UHD J. Sci. Technol.* **2021**, *5*, 20–25. [CrossRef]
27. Xin, Y.; Kong, L.; Liu, Z.; Chen, Y.; Li, Y.; Zhu, H.; Gao, M.; Hou, H.; Wang, C. Machine learning and deep learning methods for cybersecurity. *IEEE Access* **2018**, *6*, 35365–35381. [CrossRef]
28. Alarfaj, F.K.; Malik, I.; Khan, H.U.; Almusallam, N.; Ramzan, M.; Ahmed, M. Credit card fraud detection using state-of-the-art machine learning and deep learning algorithms. *IEEE Access* **2022**, *10*, 39700–39715. [CrossRef]
29. Agarwal, A.; Iqbal, M.; Mitra, B.; Kumar, V.; Lal, N. Hybrid CNN-BILSTM-attention based identification and prevention system for banking transactions. *Nveo-Nat. Volatiles Essent. Oils J.* **2021**, *8*, 2552–2560.
30. Nalayini, C.; Katiravan, J.; Sathyabama, A.; Rajasuganya, P.; Abirami, K. Identification and Detection of Credit Card Frauds Using CNN. In Proceedings of the International Conference on Computers, Management & Mathematical Sciences, Nirjuli, India, 29–30 July 2022.
31. Berhane, T.; Melese, T.; Walelign, A.; Mohammed, A. A Hybrid Convolutional Neural Network and Support Vector Machine-Based Credit Card Fraud Detection Model. *Math. Probl. Eng.* **2023**, *2023*, 8134627. [CrossRef]
32. Alghofaili, Y.; Albattah, A.; Rassam, M.A. A financial fraud detection model based on LSTM deep learning technique. *J. Appl. Secur. Res.* **2020**, *15*, 498–516. [CrossRef]
33. Benchaji, I.; Douzi, S.; El Ouahidi, B. Credit card fraud detection model based on LSTM recurrent neural networks. *J. Adv. Inf. Technol.* **2021**, *12*, 113–118. [CrossRef]
34. Owolafe, O.; Ogunrinde, O.B.; Thompson, A.F.-B. A long short term memory model for credit card fraud detection. In *Artificial Intelligence for Cyber Security: Methods, Issues and Possible Horizons or Opportunities*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 369–391. [CrossRef]

35. Maheshwari, V.C.; Osman, N.A.; Aziz, N. A Hybrid Approach Adopted for Credit Card Fraud Detection Based on Deep Neural Networks and Attention Mechanism. *J. Adv. Res. Appl. Sci. Eng. Technol.* **2023**, *32*, 315–331. [CrossRef]
36. Alsodi, O.; Zhou, X.; Gururajan, R.; Shrestha, A. A Survey on Detection of cybersecurity threats on Twitter using deep learning. In Proceedings of the 2021 8th International Conference on Behavioral and Social Computing (BESC), Doha, Qatar, 29–31 October 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1–5.
37. Narayan, V.; Ganapathisamy, S. Hybrid Sampling and Similarity Attention Layer in Bidirectional Long Short Term Memory in Credit Card Fraud Detection. *Int. J. Intell. Eng. Syst.* **2022**, *15*, 35–44. [CrossRef]
38. Cheon M-j Lee, D.; Joo, H.S.; Lee, O. Deep learning based hybrid approach of detecting fraudulent transactions. *J. Theor. Appl. Inf. Technol.* **2021**, *99*, 4044–4054.
39. Muaz, A.; Jayabalan, M.; Thiruchelvam, V. A comparison of data sampling techniques for credit card fraud detection. *Int. J. Adv. Comput. Sci. Appl.* **2020**, *11*. [CrossRef]
40. Najadat, H.; Altiti, O.; Aqouleh, A.A.; Younes, M. Credit card fraud detection based on machine and deep learning. In Proceedings of the 2020 11th International Conference on Information and Communication Systems (ICICS), Irbid, Jordan, 7–9 April 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 204–208.
41. Yamini, K.; Anitha, V.; Polepaka, S.; Chauhan, R.; Varshney, Y.; Singh, M. An Intelligent Method for Credit Card Fraud Detection using Improved CNN and Extreme Learning Machine. In Proceedings of the 2023 8th International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, 1–3 June 2023.
42. Al Balawi, S.; Aljohani, N. Credit-card fraud detection system using neural networks. *Int. Arab J. Inf. Technol.* **2023**, *20*, 234–241. [CrossRef]
43. Alamri, M.; Ykhlef, M. Hybrid Undersampling and Oversampling for Handling Imbalanced Credit Card Data. *IEEE Access* **2024**, *12*, 14050–14060. [CrossRef]
44. Fang, W.; Li, X.; Zhou, P.; Yan, J.; Jiang, D.; Zhou, T. Deep learning anti-fraud model for internet loan: Where we are going. *IEEE Access* **2021**, *9*, 9777–9784. [CrossRef]
45. Fakiha, B. Forensic Credit Card Fraud Detection Using Deep Neural Network. *J. Southwest Jiaotong Univ.* **2023**, *58*. [CrossRef]
46. Veigas, K.C.; Regulagadda, D.S.; Kokatnoor, S.A. Optimized stacking ensemble (OSE) for credit card fraud detection using synthetic minority oversampling model. *Indian J. Sci. Technol.* **2021**, *14*, 2607–2615. [CrossRef]