IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 18 June 2024, accepted 9 July 2024, date of publication 11 July 2024, date of current version 23 July 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3426955

SURVEY

Deep Learning for Credit Card Fraud Detection: A Review of Algorithms, Challenges, and Solutions

IBOMOIYE DOMOR MIENYE, (Member, IEEE), AND NOBERT JERE
Department of Information Technology, Walter Sisulu University, Buffalo City Campus, East London 5200, South Africa
Corresponding author: Ibomoiye Domor Mienye (imienye@wsu.ac.za)

ABSTRACT Deep learning (DL), a branch of machine learning (ML), is the core technology in today's technological advancements and innovations. Deep learning-based approaches are the state-of-the-art methods used to analyse and detect complex patterns in large datasets, such as credit card transactions. However, most credit card fraud models in the literature are based on traditional ML algorithms, and recently, there has been a rise in applications based on deep learning techniques. This study reviews the recent DL-based literature and presents a concise description and performance comparison of the widely used DL techniques, including convolutional neural network (CNN), simple recurrent neural network (RNN), long short-term memory (LSTM), and gated recurrent unit (GRU). Additionally, an attempt is made to discuss suitable performance metrics, common challenges encountered when training credit card fraud models using DL architectures and potential solutions, which are lacking in previous studies and would benefit deep learning researchers and practitioners. Meanwhile, the experimental results and analysis using a real-world dataset indicate the robustness of the deep learning architectures in credit card fraud detection.

INDEX TERMS Credit card, CNN, deep learning, fraud detection, GRU, LSTM, machine learning.

I. INTRODUCTION

Credit card transactions have grown due to the rapid technological advancements and convenience of electronic services [1], [2]. Consequently, there has been an increase in security issues, such as credit card fraud, which has become a significant concern for both financial institutions and customers [3], [4]. According to the Nielsen report, losses from credit card fraud in 2019, 2020, and 2021 were approximately 28.65, 28.50, and 32.34 billion dollars, respectively [5], [6], [7]. Additionally, losses due to credit card fraud globally have tripled in the last decade, from 9.84 billion dollars in 2011 to 32.34 billion dollars in 2021 [8].

Machine learning (ML) methods have been widely used for credit card fraud detection (CCFD), achieving state-of-the-art performances [9], [10], [11]. ML algorithms can be classified into supervised, unsupervised, semi-supervised, or reinforcement learning [12]. The most widely used ML method for identifying credit card fraud is the supervised learning (SL) method [13]. Supervised learning entails training an ML algorithm using a dataset where each data point has a label. The label indicates the specific class the data point belongs to, such as fraud or not fraud. SL techniques tend to learn the relationship between the input features (or independent variables) and the output labels (dependent variables).

Several studies have demonstrated the ability of neural networks to identify fraudulent transactions in complex credit card data [14], [15]. A neural network is a type of machine learning with a learning process that mimics the human brain and can be supervised or unsupervised [16]. Neural networks with multiple layers in the network also called deep learning (DL), can progressively extract higher-level features and analyse complex patterns with enhanced predictions. DL approaches have been used to identify fraudulent transactions in credit card data. For example, Mienye and Sun [17] developed an approach for credit card fraud detection using a stacked ensemble of

The associate editor coordinating the review of this manuscript and approving it for publication was Tony Thomas.

long short-term memory (LSTM) and gated recurrent unit (GRU) networks, with a multilayer perceptron (MLP) as the base learner. The DL-based ensemble performed excellently compared to other ML algorithms and the individual DL architectures. Similarly, Esenogho et al. [18] proposed a DL-based approach for credit card fraud detection using the LSTM neural network as the base learner in the adaptive boosting (AdaBoost) implementation, achieving excellent classification performance. Additionally, different studies have used convolutional neural networks [19], [20].

Meanwhile, recurrent neural networks (RNN) and their variants, such as LSTM and GRU, are the most widely used DL-based networks for modelling and analysing credit card transactions due to their ability to learn sequential data and detect temporal relationships [21], [22], [23]. The LSTM network is useful for learning long-term dependencies in a sequence. It is powerful because it can remember information from previous time steps and selectively forget or update that information as new inputs are processed. Like LSTM, GRU can selectively update or forget data from earlier time steps due to its gating mechanism, making it suitable for time series modelling.

However, despite the robustness of deep learning techniques, there are certain benefits and limitations in using them for credit card fraud detection. Therefore, this study aims to review the application and role of deep learning in credit card fraud detection. The significance of this study lies in its comprehensive review of the current state of deep learning applications in credit card fraud detection, highlighting the primary challenges and potential solutions. By systematically analyzing various deep learning techniques and their performance, this study provides valuable insights for researchers and practitioners. The main objectives and contributions of this review are as follows:

- A review of the most current research on credit card fraud detection, focusing on deep learning techniques.
- A concise but comprehensive overview of the main deep learning techniques used in CCFD and their performance comparison.
- A detailed evaluation of the widely used performance evaluation metrics, focusing on their suitability for CCFD.
- An in-depth analysis of existing challenges in using DL-based techniques for credit card fraud detection, potential solutions, and research directions.

Furthermore, to ensure a comprehensive and unbiased review, a systematic approach was employed for literature gathering. The literature selection began with a comprehensive search of multiple academic databases, including IEEE Xplore, SpringerLink, ScienceDirect, and Google Scholar. Keywords such as ''credit card fraud detection,'' ''deep learning,'' ''CNN,'' ''RNN,'' ''LSTM,'' and ''GRU'' were used to identify relevant studies. We included articles published between 2015 and 2024 to capture the most recent advancements and trends in the field. Studies were further filtered based on relevance, impact, and their contribution to the understanding of deep learning applications in credit card fraud detection.

The rest of the paper is structured as follows: section II presents a discussion of credit card fraud detection and related reviews. Section III discusses publicly available credit card datasets, and Section IV presents a comprehensive overview of deep learning and relevant architectures. Section V discusses widely used performance evaluation metrics and their suitability for credit card fraud detection. Section VI presents a review of recent studies that applied deep learning for credit card fraud detection. Section VII presents the experimental results and analysis of the various DL methods, and Section VIII discusses the challenges of using DL to detect credit card fraud and possible solutions. Section IX presents a general discussion and future research directions, and Section X concludes the study and highlights future research directions.

II. RELATED WORKS

A. CREDIT CARD FRAUD DETECTION

Credit card fraud occurs when an unauthorised user obtains access to someone's credit card details and performs transactions. It is an inclusive term for fraud committed via a bank card, including credit and debit cards [24]. Although the transactions are frequently carried out online, they can be carried out using the actual card when misplaced or stolen. Fraudsters use different methods to obtain the cardholder's information, including phishing, where a fraudster poses as a financial official to coerce a user into disclosing personal information, and skimmers use an interface to an automated teller machine (ATM) or point-of-sale device that can read a card directly [25], [26].

Detecting credit card fraud is essential in ensuring the security of consumers' finances and financial information. The two main approaches to detecting fraudulent activity are automated systems and manual investigation. While automated systems rely on algorithms and machine learning techniques to identify patterns of fraudulent behaviour, manual investigation involves human intervention to analyse suspicious activities and gather evidence. Automated systems are more popular due to their ability to process large volumes of data quickly and efficiently, and they utilize advanced statistical and machine learning models [27].

Furthermore, machine learning algorithms, including neural networks, are widely employed for detecting credit card fraud. For example, Mienye and Sun [28] proposed a CCFD method using hybrid feature selection based on genetic algorithm and information gain, and the learning algorithm was the extreme learning machine (ELM). The genetic algorithm's fitness function employed in the study was the geometric mean, which was used to tackle the class imbalance problem, leading to improved classification performance. Similarly, Karthik et al. [29] proposed a hybrid ensemble approach for credit card fraud detection to solve the imbalance class problem. The study combined boosting

and bagging methods, i.e., adaptive boosting (AdaBoost) and random forest, respectively, achieving superior performance compared to the individual classifiers.

Randhawa et al. [30] developed a hybrid ensemble based on majority voting and adaptive boosting. They compared the performance with some single classifiers, including decision tree, support vector machines (SVM), and naïve Bayes. The proposed hybrid method achieved the best Matthews Correlation Coefficient (MCC) score of 1. Other examples of ML algorithms in credit card fraud detection include random forest [31], XGBoost [32], convolutional neural network (CNN) [33], [34], RNN [35], LSTM [22], [36], [37], GRU [38], and bidirectional gated recurrent unit (BiGRU) [39].

Meanwhile, apprehending credit card scammers often relies on the availability and quality of data. Law enforcement agencies and financial institutions utilize transactional data, along with advanced machine learning algorithms, to identify suspicious patterns indicative of fraud [24]. Collaboration between banks and cybersecurity firms enables real-time monitoring and alerts, which are crucial in catching fraudsters. For instance, data-sharing agreements allow for the aggregation of data across different banks, providing a broader view of fraudulent activities. This collaborative effort enhances the detection and prevention of credit card fraud, thereby increasing the chances of apprehending scammers [40]. Furthermore, anonymized and synthetic data generation techniques can be used to augment training datasets, allowing models to better generalize and detect new types of fraud, ultimately aiding in the apprehension of scammers.

B. RELATED REVIEWS

Several research works have examined fraud detection in many reviews and surveys that have appeared in peer-reviewed articles. For instance, Modi and Dayma [41] presented reviews regarding the application of ML in detecting credit card fraud. Lucas and Jurgovsky [42] examined the difficulties in detecting credit card fraud. They concentrated on methods proposed to handle the concept drift and imbalance problems, which are two major challenges faced when analysing credit card transaction data. Concept drift occurs when the statistical properties of the data used to train an ML model change over time. As a result, the model may function differently than intended or produce less accurate predictions. The review provided a detailed discussion of concept drift, imbalance classification and approaches to handling them.

Al-Hashedi and Magalingam [43] provided a broad review of fraud detection, including insurance, credit card, and other financial fraud. The review described the ML methods used for the different fraud detection problems. Additionally, datasets and performance evaluation metrics were discussed. Also, the article lists the benefits and drawbacks of each ML method. Nevertheless, the review is limited to the following ML techniques: SVM, logistic regression, artificial neural network, k-nearest neighbor (KNN), GA, Bayesian network, decision tree, fuzzy logic, and hidden Markov model.

Popat and Chaudhary [44] examined several ML-based CCFD studies, focusing on the difficulties encountered by the ML models when detecting fraud. The methods studied include logistic regression, SVM, decision tree, ANN, and Bayesian Belief Network.. Ryman-Tubb et al. [45] conducted a review and analysed current techniques for detecting card fraud via transactional volumes. The methods reviewed include SVM, KNN, CNN, MLP, decision tree, and random forest. Pandey et al. [46] reviewed CCFD, focusing on the different types and statistics of credit card fraud in India.

Alamri and Ykhlef [47] presented a survey of credit card fraud detection studies that employed sampling techniques after identifying the imbalance class problem as the main challenge researchers face when building CCFD models. The study considered oversampling techniques, such as synthetic minority oversampling technique (SMOTE) and Borderline-SMOTE, undersampling methods, such as random undersampling (RUS) technique and Tomek links, and hybrid sampling methods, such as SMOTE-ENN and SMOTE-Tomek links. The study identified hybrid sampling methods as more efficient in handling the imbalance class problem in CCFD, while noting that oversampling techniques can lead to overfitting and undersampling can discard essential samples.

While several reviews examine credit card fraud detection systems, most of them have a very narrow scope, such as those focusing on sampling techniques [47], where the authors specifically reviewed studies that aimed to solve the imbalance problem in credit card fraud detection using resampling methods, showing the importance of effective data resampling. Meanwhile, some of the reviews have a broad scope, including [41], [43], and [44]. While they touched on vital areas of fraud detection, they have some limitations. For instance, Modi and Dayma [41] focused on performance evaluation of the machine learning algorithms without delving deep into the inner workings of the algorithms, Al-Hashedi and Magalingam [43] only surveyed the period 2009 to 2019, and Popat and Chaudhary [44] reviewed selected ML algorithms. Meanwhile, credit card fraud has increased considerably in recent years, and considering the robustness of deep learning methods in different areas, it has become imperative to explore their applicability and performance in credit card fraud detection. Therefore, this study aims to review deep learning methods and their performance in detecting credit card fraud. In addition, this review also covers specific gaps in related reviews, such as identifying and reviewing suitable evaluation metrics, challenges in building CCFD models, and potential solutions.

III. CREDIT CARD DATASETS

Due to privacy and security concerns, credit card datasets are not easily accessible. However, there are a few publicly available credit card datasets that are used for fraud detection,

and they are described in this section. Meanwhile, other publicly available credit card datasets are not considered in this section since they were not curated for fraud detection. For example, the Taiwan and Australian credit card datasets were designed for credit card default and risk prediction. The fraud detection datasets include the following:

- *European credit card dataset:* The European credit card dataset [48] has been widely used by researchers in building robust CCFD models. This dataset contains 284,807 transactions from European countries, which have been labeled as either normal or fraudulent. Each transaction includes 28 features, such as time of the transaction, amount, and various anonymized variables. The dataset has become a benchmark for evaluating the performance of fraud detection algorithms. Of the 284,807 transactions, only a tiny fraction (0.17%) belong to the positive class (i.e., fraud transactions), while the majority class (99.83%) represents the negative class or legitimate transactions. This imbalanced distribution poses a significant challenge for many machine learning algorithms and requires careful consideration during model development. The dataset was released in 2013, and while it is older, it remains relevant for current research due to its comprehensive nature.
- *Brazilian credit card dataset:* This dataset was obtained from a large Brazilian bank, and it contains 374,823 transactions [29]. The fraud samples comprise 3.74% of the records. Each record in the dataset has 17 numerical features, including merchant category code, post/zip code of current and previous transactions, current transaction amount, previous transaction amount, transaction type (card present), credit limit, card type (e.g., Mastercard, Visa, Diners), local/international transaction, previous transaction fraud score, and time since last transaction. Despite its age, this dataset provides valuable insights into transaction patterns and fraud detection.
- *IEEE-CIS Fraud Detection Dataset:* Released in 2019, the IEEE-CIS dataset is one of the more recent publicly available datasets [49]. It contains transaction data provided by Vesta Corporation and includes a mix of fraud and non-fraud transactions over a period. The dataset consists of two files: one with identity information and another with transaction details, comprising approximately 590,000 transactions. Features include device type, device information, card information, transaction amount, and timestamp. The dataset is highly imbalanced, with a small fraction representing fraudulent transactions.
- *PaySim Synthetic Dataset:* PaySim is a synthetic dataset generated using a simulation based on real transaction data [50]. Although it is not real-world data, it was created to mimic the transaction behaviors found in a real financial institution. Released in 2017, the dataset includes features such as transaction type, amount, balance, and origin and destination accounts. PaySim is valuable for its realistic simulation of fraud scenarios, and it contains over 6 million transactions.

IV. OVERVIEW OF DEEP LEARNING

In this Section, an overview of DL is presented, including a detailed description of the widely used DL architectures. Deep learning, a branch of ML, maps input data to new representations or generates predictions using neural networks [51]. Meanwhile, neural networks consist of interconnected neurons with weighted connections. The neuron converts its input into a single output by summing its weighted inputs using a non-linear activation function [52]. The network's weight parameters are modified using gradient descent optimisation, reducing the loss function, i.e., the discrepancy between the expected and actual outputs. A neural network can have one or more hidden layers. A neural network with one hidden layer is often referred to as a shallow network, while a network with many hidden layers is called a deep neural network (DNN). Figure 9 shows a general shallow neural network (or simple ANN) and deep neural network architectures, where the latter has multiple hidden layers.

Furthermore, deep learning is a broader term used to describe ML techniques that are based on neural networks with many layers (deep architectures). Deep learning can be unsupervised, semi-supervised, or supervised [54]. Deep learning methods perform better than shallow machine learning methods in most applications with big and high-dimensional data [55], [56]. Additionally, the ability of deep learning to achieve excellent performance when the data increases sets it apart from conventional machine learning. Because DL architectures can handle massive datasets to create an efficient data-driven model, it is beneficial when working with large volumes of data, such as credit card transaction data [56], [57].

Deep-learning architectures include DNN, RNN, CNN, transformers, and deep reinforcement learning. These DL architectures have produced results that are as good as human expert performance and sometimes outperforming the human experts in domains such as image recognition, natural language processing, computer vision, and speech recognition. Meanwhile, RNNs are well-suited for sequential data modelling, such as credit card transactions, and are a significant focus of this study. Though RNNs can model sequence data effectively, they are challenging to train due to issues with vanishing and exploding gradients [58], which led Hochreiter and Schmidhuber [59] to develop the LSTM to tackle the vanishing gradients problem effectively. The GRU, first presented in [60], manages the data and performs LSTM-like tasks without requiring an additional memory unit. The bidirectional variants of these networks are unique variations that enable the system to forecast the current state more accurately by utilising data from subsequent time steps in addition to earlier time steps. The following subsections present brief but concise overview of these deep learning methods.

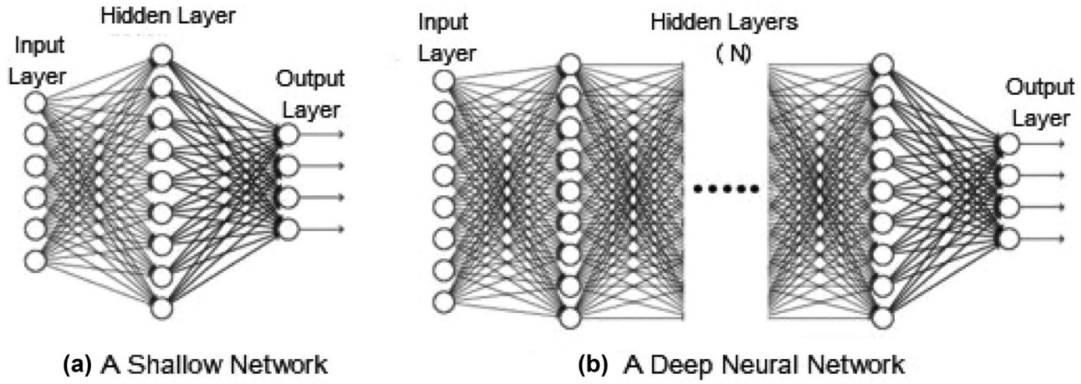

FIGURE 1. Simple ANN vs deep learning [53].

A. MULTILAYER PERCEPTRON

Multilayer Perceptron is a feedforward artificial neural network for supervised learning problems. MLP is considered the foundation network of deep learning or deep neural networks. It is a fully connected network comprising an input layer where input data is received, one or more hidden layers that serve as the neural network's computational engine, and an output layer that makes the prediction based on the given inputs [61]. Furthermore, backpropagation, a supervised learning algorithm, is widely utilised for training the MLP. The backpropagation is considered as the primary building block of network networks. Meanwhile, the widely used error function in the MLP is the mean squared error, represented as follows:

$$E = \frac{1}{2}\sum_{i=1}^{n} || p_i - t_i ||^2 \tag{1}$$

where n represents the sample size, while p_i and t_i represent the predicted and actual outputs for the $i - th$ sample. Meanwhile, the MLP network uses activation functions to determine its output, and examples of the activation functions include Softmax, rectified linear unit (ReLu), Sigmoid, and hyperbolic tangent (Tanh) [62]. In training the MLP, different optimisation techniques can be used, including stochastic gradient descent (SGD) and adaptive moment estimation (Adam). Lastly, the MLP hyperparameters mainly need to be tuned for optimal performance, and these hyperparameters include the number of neurons, hidden layers, and iterations.

B. CONVOLUTIONAL NEURAL NETWORK

The convolutional neural network is a well-known deep learning architecture with wide applications in image recognition [63], [64], [65], achieving state-of-the-art performances, and has recently been applied in several cCCFD models [66], [67]. It consists of neurons that have learnable weights and biases. Meanwhile, the CNN's hidden layers are made up of convolutional, pooling, and fully connected layers [68]. A CNN showing this multi-layer architecture is shown in Figure 2. The convolutional layer, CNN's core component, uses learnable filters to compute a convolution operation on the input. A set of feature maps is produced after the convolution operation. The pooling layer is utilized to reduce the feature maps' spatial dimensions [69]. After the feature extraction and downsampling by the convolutional and pooling layers, respectively, their output is mapped by the fully connected layers to the final output of the CNN [70]. For a classification problem like CCFD, this mapping returns the probability for each class (fraud or not fraud).

C. SIMPLE RNN

Conventional neural networks assume that each unit in the input vectors is independent. As a result, sequential data cannot be predicted by the typical neural network. However, recurrent neural networks are built to have time series memory, making them suitable for processing sequential data [72]. They can effectively model temporal dependencies in the data. Figure 3 shows a simple RNN architecture.

The autoregressive architecture of RNNs allows them to maintain a hidden state that can capture information from prior time steps. This feature is significant when working with sequential data like credit card transaction records. In the simple RNN implementation, the current hidden state h_t is computed according to:

$$h_t = \tanh(Ux_t + Wh_{t-1}) \tag{2}$$

where U is the matrix of trainable weights for the input x_t, W is the matrix of trainable weights for the previous hidden state h_{t-1}, and tanh is the activation function applied element-wise.

D. LONG SHORT-TERM MEMORY

The LSTM network is a well-known RNN variant that was developed primarily to solve the vanishing gradient issue associated with the simple RNN, which made it challenging to identify long-term dependencies in the data. LSTMs have unique gating mechanisms that enable them to store information better over longer sequences, as shown in Figure 4.

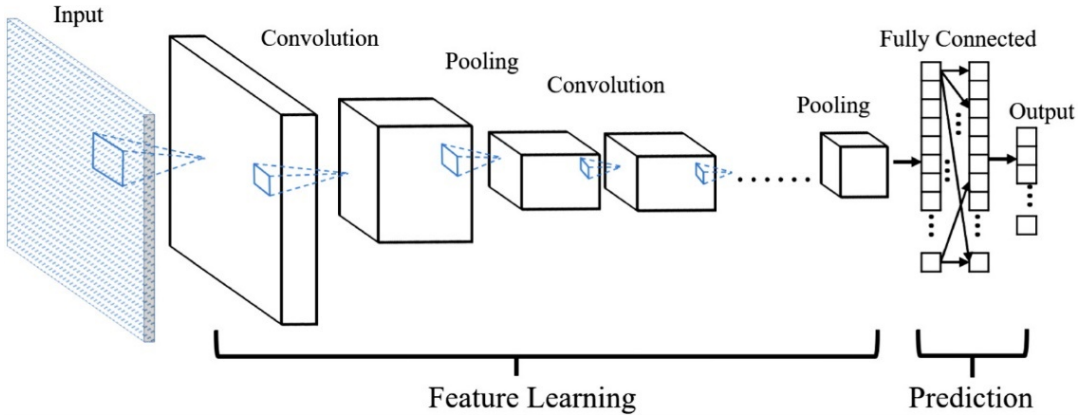

FIGURE 2. CNN Architecture [71].

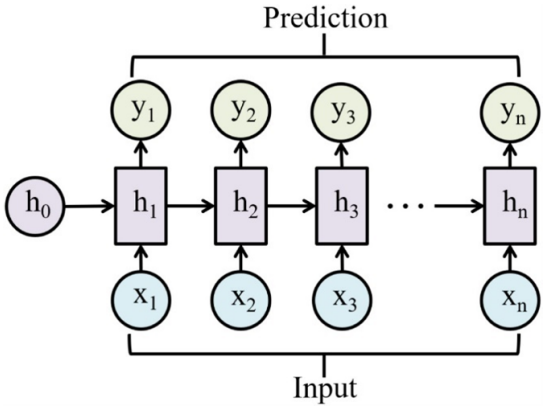

FIGURE 3. The architecture of Simple RNN.

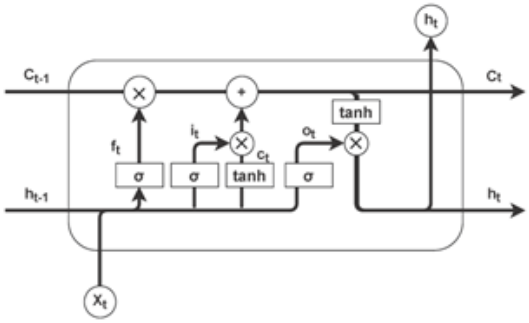

FIGURE 4. Architecture of the LSTM network [73].

LSTM models have performed excellently in many time series prediction applications, including credit card fraud detection. It comprises three types of gates: forget, output, and input [18]. An LSTM's forget gate decides what data from the previous hidden state should be kept or discarded. During training, the LSTM can focus on relevant inputs better and maintain a steady gradient by selectively ignoring certain information. Only necessary information is added to the cell state because the input gate regulates new data flow into the cell state. The output gate then uses the updated cell state and the input data to determine what information should be transmitted to the next block. The gating mechanism enables LSTM to extract the sequence's

FIGURE 5. A stacked LSTM.

long-term properties. Assuming x_t is the input, the following functions are computed by the LSTM cell:

$$i_t = \sigma(V_i x_t + W_i h_{t-1} + b_i) \tag{3}$$
$$f_t = \sigma(V_f x_t + W_f h_{(t-1)} + b_f) \tag{4}$$
$$\tilde{c}_t = tanh(V_c x_t + W_c h_{(t-1)} + b_c) \tag{5}$$
$$c_t = f_t \otimes c_{(t-1)} + i_t \otimes \tilde{c}_t \tag{6}$$
$$o_t = \sigma(V_o x_t + W_o h_{(t-1)} + b_o) \tag{7}$$
$$h_t = o_t \otimes tanh(c_t) \tag{8}$$

where i_t, f_t, c_t, and o_t denote the input, forget, cell, and output gates, respectively. Meanwhile, V_*, W_*, and b_* represent the learnable parameters, while h_* represents the hidden state. Furthermore, σ is the sigmoid activation function and $\otimes$ denotes the element-wise product [74].

The standard LSTM network is made up of one hidden LSTM layer and a feedforward output later, but it can be extended to have many hidden layers and every layer to have several memory cells, and this is called a stacked LSTM network. By stacking the LSTM hidden layers, the network becomes deeper, which is important because the success of deep learning models has been linked to how deep the network is [75]. A general block diagram of a stacked LSTM is shown in Figure 5.

E. GATED RECURRENT UNIT

The GRU was introduced by Cho et al. [60]. They also have gating mechanisms that aid in managing the information flow inside the network but without an output gate. A model can be trained using the GRU to keep previous information or remove irrelevant information. The GRU architecture is shown in Figure 6. In contrast to LSTMs, GRUs have a simpler architecture with just two gates: an update gate z_t and a reset gate r_t. The reset gate regulates how much the previous hidden state influences the current hidden state, whereas the update gate controls how much the prior hidden state is kept [76]. With a less complex and more computationally

efficient network than LSTMs, GRUs are able to capture long-term dependencies in sequential data more effectively because of this gating mechanism. The functions that a GRU cell computes are as follows:

$$r_t = \sigma(V_r x_t + W_r h_{(t-1)} + b_r) \tag{9}$$
$$z_t = \sigma(V_z x_t + W_z h_{(t-1)} + b_z) \tag{10}$$
$$\tilde{c}_t = tanh(V_c x_t + W_c(r_t \otimes h_{t-1}) + b_c) \tag{11}$$
$$c_t = (1 - z_t) \otimes c_{(t-1)} + z_t \otimes \tilde{c}_t \tag{12}$$
$$h_t = c_t \tag{13}$$

where W_r, W_z, V_r, and V_z are weight matrices while b_r and b_z represent the bias vectors [77].

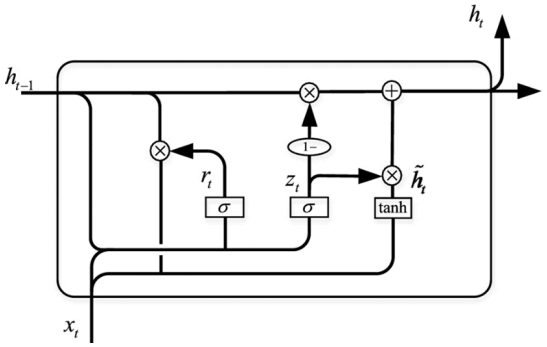

FIGURE 6. Architecture of the GRU network.

F. BIDIRECTIONAL LONG SHORT-TERM MEMORY

Unidirectional Long Short-Term Memory, or LSTM, only stores past data since its inputs are limited to the past and must be fed in the correct order. On the other hand, a bidirectional long short-term memory network (BiLSTM) combines two hidden states to process the inputs in both forward and backward directions, as shown in Figure 7. This feature allows the network to store information from incoming inputs, guaranteeing that information about previous and upcoming states is always accessible. In other words, a BiLSTM is precisely like an LSTM, except that it employs historical and future data to compute the weights [17]. The BiLSTM's network structure comprises two LSTMs with opposite information propagation directions. At each time unit, the current pre-hidden state output and post-hidden state output are derived and recorded. The BiLSTM's output value is then determined by connecting the two hidden states. The mathematical formulation of the BiLSTM network is as follows:

$$h_t^f = LSTM(x_t, h_{t-1}^f) \tag{14}$$
$$h_t^b = LSTM(x_t, h_{t-1}^b) \tag{15}$$
$$o_t = W_f.h_t^f + W_b.h_t^b + b \tag{16}$$

where $LSTM(\cdot)$ represent the mapping of the LSTM layers, while W_f and W_b are the weight matrix of the forward and backward LSTM layers, and b is the output layer's deviation vector. Furthermore, BiLSTM models are substantially more efficient in natural language processing and can outperform conventional unidirectional LSTMs in time series prediction [78]. Because of its dual model architecture, BiLSTMs have the drawback of requiring longer training times.

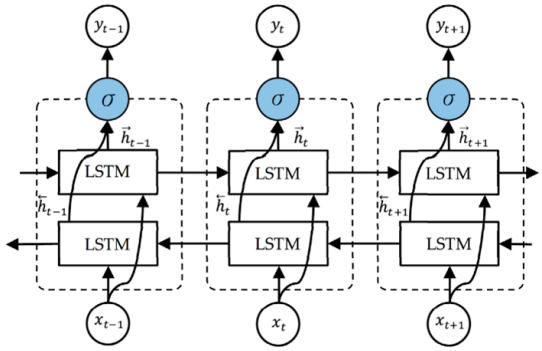

FIGURE 7. The architecture of the BiLSTM network [78].

G. BIDIRECTIONAL GATED RECURRENT UNIT

Bidirectional gated recurrent unit (Bi-GRU) is a variant of the popular GRU recurrent neural network designed to improve temporal modelling accuracy. It is an example of a bi-directional RNN, meaning it can process input sequences in both forward and backward directions. In recent years, Bi-GRU has become a popular choice for temporal modelling in deep learning applications. It is an efficient model that combines forward and backwards information propagation to improve accuracy when predicting future events or sequences. Its main strengths over GRU are its ability to capture bidirectional dependencies and its improved performance in tasks involving long-term dependencies. In the Bi-GRU implementation, the input sequences are computed in both directions using two sublayers, modelling forward and backward hidden sequences, which are combined to obtain the current hidden state h_t and output o_t of the network [79]. The mathematical formulation is represented by the following:

$$h_t^f = GRU(x_t, h_{t-1}^f) \tag{17}$$
$$h_t^b = GRU(x_t, h_{t-1}^b) \tag{18}$$
$$h_t = W_f.h_t^f + W_b.h_t^b) \tag{19}$$
$$o_t = \phi(W^o h_t) \tag{20}$$

where h_t^f and h_t^b represent the forward and backward hidden sequences, the GRU function denotes the nonlinear transformation of the input, W^o represents the weight coefficient in the network's hidden and output layers, and ϕ denotes the activation function applied to the output layer.

H. TRANSFORMER MODELS

While traditional deep learning architectures, such as CNNs, LSTM, and GRU, have been widely used in fraud detection and have achieved excellent performance, they have limitations, particularly in capturing long-range dependencies and processing large-scale datasets efficiently. Transformer models have recently gained attention in the field of credit card

fraud detection due to their robust performance in sequence modelling and anomaly detection tasks. Unlike traditional RNNs and CNNs, Transformers employs a self-attention mechanism that allows them to weigh the importance of different parts of an input sequence dynamically. The core component of a Transformer model is the self-attention mechanism, which computes attention scores for each pair of tokens in the input sequence [80]. The attention mechanism is defined as follows:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{21}$$

where Q (queries), K (keys), and V (values) are the input matrices, and d_k is the dimension of the keys. The softmax function ensures that the attention scores sum to one, highlighting the most relevant tokens [81]. Meanwhile, the Transformer model uses multiple self-attention heads to capture different aspects of the relationships within the input sequence:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \text{head}_2, \ldots, \text{head}_h)W^O \tag{22}$$

where each attention head head_i is computed as:

$$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \tag{23}$$

where W_i^Q, W_i^K, and W_i^V are learned projection matrices, and W^O is the output projection matrix. Furthermore, Transformers can be pre-trained on large datasets and fine-tuned on specific fraud detection tasks, leveraging transfer learning to improve performance. The pre-training phase typically involves learning general representations from a large corpus of transaction data, while the fine-tuning phase adapts these representations to the specific characteristics of fraudulent transactions.

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{pre-train}} + \lambda\mathcal{L}_{\text{fine-tune}} \tag{24}$$

where $\mathcal{L}_{\text{pre-train}}$ is the loss during the pre-training phase, $\mathcal{L}_{\text{fine-tune}}$ is the loss during the fine-tuning phase, and λ is a weighting factor.

V. PERFORMANCE EVALUATION METRICS

An essential step in ensuring effective credit card fraud detection is the performance metrics used to assess the model's prediction performance. This section discusses metrics that are widely applied and their suitability in credit card fraud detection. The confusion matrix provides a summary of the binary classification results, and it is shown in Table 1, indicating true Positive (TP), true negative (TN), false position (FP), and false negative (FN).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{25}$$

$$ErrorRate = 1 - Accuracy \tag{26}$$

When assessing the performance of ML models, the most commonly utilised measures are accuracy and error rate [82], [83], i.e., Equation 25 and Equation 26, respectively. However, when dealing with CCFD, which is mostly an imbalance classification task, neither is sufficient because the majority class, or the non-fraud class, dominates the final value. Hence, a naïve classifier can achieve 99% accuracy by labelling all samples as not fraud when given input data where the positive class distribution makes up only 1% of the data set. There would be no actual benefit to such a model. Other commonly used metrics are sensitivity, specificity, and precision, and their mathematical formulations are shown below:

TABLE 1. Confusion matrix.

	Actual Positive	Actual Negative
Predicted Positive	True Positive	False Positive
Predicted Negative	False Negative	True Negative

$$Precision = \frac{TP}{TP + FP} \tag{27}$$

$$Sensitivity = \frac{TP}{TP + FN} \tag{28}$$

$$Specificity = \frac{TN}{TN + FP} \tag{29}$$

Precision represents the fraction of positively predicted samples that are classified correctly. Because precision takes into account the number of negative instances that are wrongly predicted as positive, it is sensitive to class imbalance [84], [85]. However, precision on its own is inadequate as it does not reveal how many positive instances were mistakenly classified as negative. Sensitivity, also known as the true positive rate (TPR), quantifies the proportion of the positive instances that the classifier accurately predicted to be positive. The class imbalance has no effect on sensitivity since it solely depends on the positive class. Meanwhile, the number of negative instances that are incorrectly classified as positive is not taken into account by sensitivity. Specificity, also called true negative rate (TNR), quantifies the proportion of the negative instances that were classified correctly. Furthermore, there are other metrics that tend to combine earlier discussed metrics to obtain a more comprehensive evaluation of the model performance, including F-measure, G-mean, and balance accuracy. Their mathematical formulations are shown as follows:

$$F - Measure = \frac{2XPrecisionXSensitivity}{Precision + Sensitivity} \tag{30}$$

$$G - Mean = \sqrt{SensitivityXSpecificity} \tag{31}$$

$$BalancedAccuracy = \frac{Sensitivity + Specificity}{2} \tag{32}$$

F-measure is the harmonic mean of sensitivity and precision. It presents a way to combine sensitivity and precision into one metric that captures the properties of both metrics. The geometric mean (G-mean) combines specificity and sensitivity into one metric that considers a balance between both minority and majority class performances. Like G-mean, the balanced accuracy metric computes a metric sensitive to

the minority class instances by combining TPR and TNR. When evaluating the performance of classifiers trained with imbalanced data, F-measure, G-Mean, and balanced accuracy are better metrics compared to accuracy and error rate [86].

Furthermore, the receiver operating characteristic (ROC) curve, the area under the ROC curve (AUC), and the precision-recall curve are other important metrics. The ROC curve is a plot of the true positive rate versus the false positive rate at various classification thresholds, and it demonstrates the ability of a classifier to distinguish between the positive and negative classes [65]. The AUC is a summary of the ROC curve. It has a value range of 0 to 1, with 1 indicating that all of the classifier's predictions are accurate and 0 indicating that all of the predictions are incorrect. The precision-recall (PR) curve demonstrates the tradeoff between precision and recall at various thresholds [87]. Since high precision indicates a low false positive rate, while high recall indicates a low false negative rate, an area under the precision-recall curve with a high value indicates high recall and precision values.

In imbalance classification tasks, such as credit card fraud detection, the ROC curve can be misleading because a small number of correct or wrong classifications can lead to a significant change in the ROC curve or AUC value. Meanwhile, the PR curve focuses on the minority class, making it a more suitable metric for imbalance classification [88]. Hence, the PR curve, together with other metrics previously discussed, is recommended for imbalanced credit card fraud detection.

VI. DEEP LEARNING APPLICATIONS IN CREDIT CARD FRAUD DETECTION

Benchaji et al. [23] developed a CCFD model via sequential modelling of the credit card data using deep LSTM neural networks and attention mechanisms. The proposed approach takes into account the sequential nature of the credit card data and enables the classifier to determine which transactions in the input sequence are the most significant. Specifically, in the proposed approach, the LSTM was used to ensure sequential modelling of the data, the attention mechanism was employed to improve the performance of the LSTM, and the uniform manifold approximation and projection (UMAP) was introduced to select the most significant attributes. The models yielded good performance with an accuracy of 96.7%.

Similarly, Femila et al. [89] developed a credit card fraud detection model with the aim of lowering losses caused by credit card fraud. This study aimed to identify credit card fraud using an LSTM model. An attention mechanism was also incorporated to boost the LSTM's performance since models with such a structure have shown to be effective in sequence modelling. Other classifiers like SVM, naive Bayes, and ANN were contrasted with the LSTM, and the experimental results showed that the LSTM yielded robust outcomes, including an accuracy of 100%.

Najadat et al. [90] developed a model based on BiLSTM and BiGRU with MaxPooling layers. Meanwhile, the dataset was preprocessed using three resampling techniques: random oversampling, random undersampling, and SMOTE. The study compared the performance of the deep learning-based classifier and other ML classifiers, including logistic regression, random forest, voting, naïve base, AdaBoost, and decision tree. When random oversampling was applied, the proposed BiLSTM-BiGRU obtained excellent performance, with an AUC of 91.4%.

Forough and Momtazi [91] developed a credit card fraud detection model that considers the sequential structure of credit card transactions. The method used LSTM models as base classifiers in an ensemble implementation, where a feed-forward neural network (FFNN) was used as the voting mechanism. The proposed LSTM ensemble outperformed other ML and DL techniques when experimented on two credit card datasets. Specifically, the proposed ensemble achieved an AUC of 0.879 and 0.88 on the European and Brazilian datasets.

Aurna et al. [92] proposed a federated learning (FL) based CCFD approach in order to protect the privacy of sensitive credit card data. This allows the model to be trained without exposing credit card information to third parties on the cloud. The study considered three deep learning models based on LSTM, MLP, and CNN. The influence on the conventional centralised and FL systems is then examined using four different sampling procedures to address the data imbalance problem. The proposed approach was compared with other well-performing methods in the literature, and the experimental results show that the proposed method obtains excellent performance with accuracies for CNN, MLP, and LSTM models being 99.51%, 98.77%, and 98.20%, respectively.

Xie et al. [93] developed a time-aware attention-based interactive LSTM (TAI-LSTM) method for credit card fraud detection. The method contains two time-aware gates, a time-aware attention module and an interaction module. The approach was built to capture the customer's long and short-term spending behaviour and detect behavioural changes over time. The time-aware attention model aims to extract behavioural information from the sequential credit card data, while the interactive module aims to acquire more thorough and logical representations. The findings demonstrate that the learned representation can accurately differentiate between fraudulent and genuine behaviours and that the suggested approach outperforms similar methods with a sensitivity of 99.6%.

Sehrawat and Singh [21] used an auto-encoder with LSTM and GRU neural networks to detect credit card fraud. In the proposed approach, the autoencoder performed representation learning from the data, which was achieved by excluding the class labels. The auto-encoder's output combined with the class labels were supplied as input to the LSTM and GRU models to detect fraud. The LSTM obtained a classification accuracy of 99.1%.

Ajitha et al. [94] compared the performance of a CNN model with other ML algorithms, including XGBoost, SVM, random forest, KNN, logistic regression, and decision. The

CNN model consists of one flattened layer, one fully connected layer, and two convolutional layers with the ReLu activation function. The experimental results indicated that the CNN obtained a classification accuracy of 97.2%, outperforming the other classifiers.

Yousuf Ali et al. [95] developed DL models combined with the SMOTE oversampling approach to forecast credit card fraud. The paper employed three widely used deep learning architectures: LSTM, CNN, and a DNN. The experimental results showed that the CNN model achieved a significant increase in accuracy after the SMOTE-based resampling, especially in detecting fraud instances. The CNN obtained an accuracy of 99.9%. The study concluded that the CNN architecture can aid in reducing financial losses due to credit card fraud. Gambo et al. [19] also employed CNN for credit card fraud detection, combining it with the adaptive synthetic (ADASYN) sampling method. After the resampling of the credit card dataset, the CNN model achieved an accuracy of 99.8%.

Mizher and Nassif [96] presented credit card fraud detection models based on the convolutional neural network technique and two machine learning algorithms: SVM and random forest. Using highly skewed real-world credit card data, the models were assessed and contrasted, and the random forest achieved the best performance with an accuracy of 99.7%. Meanwhile, the CNN obtained an accuracy of 93.5%.

Furthermore, recent advancements in deep learning have seen the rise of Transformer-based models, which have revolutionized various fields, including natural language processing and, more recently, fraud detection. Transformers, such as the Bidirectional Encoder Representations from Transformers (BERT) model and its variants, have demonstrated exceptional performance in sequence modelling and anomaly detection tasks due to their ability to capture long-range dependencies and contextual information effectively. The application of Transformer models in credit card fraud detection offers several advantages over traditional DL architectures like LSTM and CNN. For example, their self-attention mechanism allows them to focus on the most relevant parts of a transaction sequence, improving the accuracy of fraud detection. Studies such as those of Igbal and Amin [97] and Tang and Liu [98] have explored the application of Transformers in credit card fraud detection, showing promising results, with accuracy of 100% and 98.98%, respectively. Table 2 summarises the deep learning methods reviewed in this study, comparing their performances based on the accuracy metric.

VII. EXPERIMENTAL RESULTS AND ANALYSIS

In this section, we analyze the performance of the MLP and deep learning architectures trained with the European credit card dataset. The DL techniques include CNN, simple RNN, LSTM, GRU, BiLSTM, and BiGRU. To ensure a fair comparison, the models were trained using the parameters listed in Table 4. These parameters were chosen based on their widespread use in the literature and their effectiveness in similar tasks. To ensure robust evaluation, the models were trained and validated using k-fold cross-validation. Specifically, we used 5-fold cross-validation, where the dataset was randomly partitioned into five equal-sized subsets. Each subset was used as a validation set once, while the remaining four subsets were used for training. This process was repeated five times, and the performance metrics were averaged over the five folds to provide a reliable estimate of model performance.

Table 4 and Figure 8 show the performance of the various models in terms of accuracy, sensitivity, specificity, precision, and F-measure. Additionally, Figure 9 shows the ROC curves of the models. The results reveal interesting insights into the behaviour of the models, especially regarding the near-perfect accuracy and specificity values compared to the relatively lower sensitivity and F-measure.

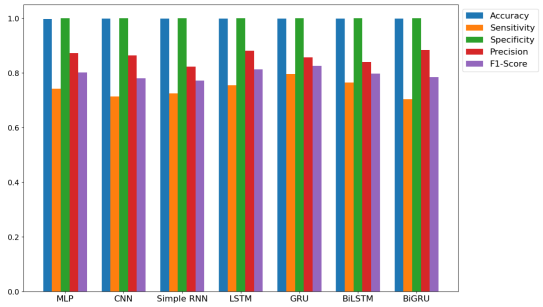

FIGURE 8. Performance comparison.

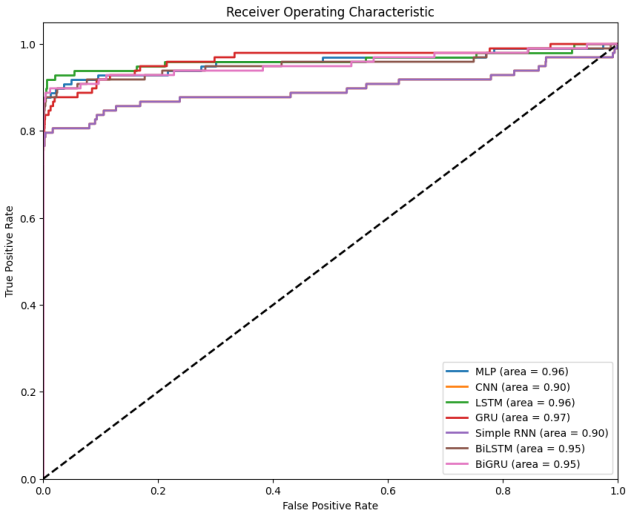

FIGURE 9. ROC curves of the various models.

Firstly, all the models performed well in classifying the majority class (non-fraud samples), indicated by the high accuracy and specificity scores. Specificity and accuracy metrics reflect the model's performance on the majority class samples. Precisely, specificity measures the correctly classified non-fraud samples or true negative rate, which was very high due to the high number of negative samples. The European credit card dataset is highly imbalanced,

TABLE 2. Summary of the DL-based credit card fraud studies.

Reference	Year	Method	Dataset	Accuracy (%)	Sensitivity (%)	AUC (%)
Najadat et al. [90]	2020	BiLSTM-BiGRU	IEEE-CIS Fraud Detection	-	80.06	83.9
Benchaji et al. [23]	2021	LSTM + UMAP	European Credit Card	96.7	91.9	-
Forough and Momtazi [91]	2021	LSTM-FFNN	European Credit Card	-	74.1	87.0
Femila et al. [89]	2022	LSTM + attention mechanism	European Credit Card	100	1.00	-
Esenogho et al. [18]	2022	Boosted LSTM ensemble	European Credit Card	-	96.2	95.0
Yousuf Ali et al. [95]	2022	CNN	European Credit Card	99.9	-	-
Yousuf Ali et al. [95]	2022	LSTM	European Credit Card	97.3	-	-
Gambo et al. [19]	2023	CNN+ADASYN	European Credit Card	99.8	-	-
Aurna et al. [92]	2023	FL with LSTM	European Credit Card	99.8	76.5	-
Aurna et al. [92]	2023	FL with CNN	European Credit Card	99.9	82.9	-
Xie et al. [93]	2023	TAI+LSTM	European Credit Card	-	99.6	51.0
Xie et al. [93]	2023	GRU	European Credit Card	-	60.8	50.0
Mienye and Sun [17]	2023	LSTM and GRU Stacked Ensemble	European Credit Card	-	90.5	92.0
Mienye and Sun [17]	2023	LSTM	European Credit Card	-	77.1	81.0
Sehrawat and Singh [21]	2023	Autoencoder with LSTM	European Credit Card	99.1	90.0	-
Ajitha et al. [94]	2023	CNN	European Credit Card	97.2	90.2	-
Mizher and Nassif [96]	2023	CNN	European Credit Card	93.5	-	-
Igbal and Amin [97]	2024	Transformer model	European Credit Card	100	100	100
Tang and Liu [98]	2024	Transformer model	”TipDM Cup” Financial dataset	98.9	95.5	96.73

TABLE 3. Parameters of the various deep learning models.

Classifier	Parameters
MLP	Layers: 2 hidden layers; Neurons: [64; 32]; Activation function: ReLU; Optimizer: Adam; Learning rate: 0.001; Batch size: 32.
CNN	Layers: 2 conv. layers; 1 FC layer; Filters: [32; 64]; FC Units: [128]; Kernel Size: (3; 3); Activation function: ReLU; Pooling: MaxPooling (2; 2); Optimizer: Adam; Learning rate: 0.001; Batch size: 32.
Simple RNN	Layers: 1 hidden layer; Neurons: 50; Activation function: Tanh; Optimizer: Adam; Learning rate: 0.001; Batch size: 32.
LSTM	Layers: 1 hidden layer; Neurons: 50; Activation function: Tanh; Dropout: 0.2; Optimizer: Adam; Learning rate: 0.001; Batch size: 32.
GRU	Layers: 1 hidden layer; Neurons: 50; Activation function: Tanh; Dropout: 0.2; Optimizer: Adam; Learning rate: 0.001; Batch size: 32.
BiLSTM	Layers: 1 hidden layer; Neurons: 50 (each direction); Activation function: Tanh; Dropout: 0.2; Optimizer: Adam; Learning rate: 0.001; Batch size: 32.
BiGRU	Layers: 1 hidden layer; Neurons: 50 (each direction); Activation function: Tanh; Dropout: 0.2; Optimizer: Adam; Learning Rate: 0.001; Batch Size: 32.

TABLE 4. Performance evaluation of the various classifiers.

Classifier	Accuracy	Sensitivity	Specificity	Precision	F1-Score
MLP	0.9980	0.7415	0.9999	0.8724	0.8016
CNN	0.9993	0.7143	0.9998	0.8642	0.780
Simple RNN	0.9994	0.7253	0.9998	0.8227	0.7721
LSTM	0.9994	0.7551	0.9998	0.8810	0.8132
GRU	0.9994	0.7959	0.9998	0.8571	0.8254
BiLSTM	0.9992	0.7653	0.9996	0.8395	0.7972
BiGRU	0.9993	0.7041	0.9998	0.8846	0.7841

with most records being legitimate and only a few being labelled as fraud. Therefore, training the models on such an imbalanced dataset ensured they were efficient at classifying the majority class samples, contributing to their high accuracy and specificity values,

However, the GRU model achieved superior performance across the various metrics, including having the best sensitivity. The GRU is known for its effectiveness in capturing temporal dependencies with lesser parameters compared to the LSTM, a possible reason for the high scores. Meanwhile, the high sensitivity value demonstrates that the GRU is relatively better at identifying fraud instances, which is crucial in credit card fraud detection engines. Also, its balance between precision and sensitivity, as indicated in the F-measure of 0.8254, implies that the GRU efficiently manages the trade-off between detecting fraud instances and minimizing false positives.

Furthermore, the MLP model achieved good performance, especially with regard to accuracy and specificity. However, its sensitivity indicates a limitation in predicting fraud cases compared to the GRU. Meanwhile, Simple RNN and CNN seem to have the lowest performance compared to the other models. The RNN achieved the least F-measure of 0.772. RNNs are inefficient when faced with long-term dependencies due to the vanishing gradient issue, which could explain the poor performance. The CNN obtained an F-measure of 0.780, which is better than the RNNs but less than the remaining models. Lastly, the discrepancy between the high performance in the majority class samples and

the poor performance in the minority class (fraud) samples can be mainly attributed to the imbalanced credit card data. In order to enhance the performance of the minority class, researchers can explore methods such as oversampling, ensemble learning, and cost-sensitive learning that penalize wrong predictions in the minority class more than the other class.

VIII. CHALLENGES AND POTENTIAL SOLUTIONS

Researchers and practitioners usually encounter challenges when developing deep learning-based credit card fraud detection models. In this section, an attempt is made to discuss some of these challenges and potential solutions.

A. CLASS IMBALANCE

In binary classification problems, such as credit card fraud detection, a class imbalance occurs when one class (also called the majority class) significantly outnumbers the other class, known as the minority class. This results in a skewed dataset, making the learning process challenging for machine learning algorithms. Most credit card transaction data have a class imbalance, making it challenging to identify fraud. Meanwhile, the minority class (fraud samples) is more important than the majority class, and wrongly predicting a fraud case as legitimate has a higher cost than predicting a legitimate transaction as fraud [99]. Also, most machine learning algorithms used for classification tasks assume that there are equal amounts of samples in each class.

Furthermore, class imbalance can also lead to biased and poor results. Therefore, addressing this problem is essential when building ML and DL models. Some methods to address the imbalance class problem include data resampling techniques like oversampling and undersampling, ensemble methods that can handle imbalanced datasets, and cost-sensitive learning algorithms that assign different weights to minority and majority classes [100]. However, when building deep learning models for credit card fraud detection, the following approaches have attracted a lot of attention from the DL community:

- *Loss function adaptation:* This can be used to make deep learning methods learn effectively from imbalanced data. It involves changing the loss function of the DL model to make it insensitive to the skewed distribution. The loss function adaptation is an algorithm-level modification and has been successfully applied in several DL models [101], [102]. This approach is similar to cost-sensitive learning as it is based on the premise that instances should not be treated equally during training and that errors in minority class instances are costlier than those in the majority class and, hence, should be penalised more severely [103]. Based on this idea, Wang et al. [104] and Lin et al. [105] proposed mean false error and focal loss, two robust adapted loss functions for deep learning modelling. Other loss functions include generalised cross-entropy loss [106] and class-balanced loss [107], which were introduced more recently.
- *Hybrid Models:* Hybrid models that combine deep learning algorithms with traditional machine learning algorithms that are better suited for imbalance classification, such as decision trees and random forests, have been studied recently and have achieved excellent results. For example, Dar et al. [108] developed a hybrid model, combining DNN and XGBoost and Semwal et al. [109] combined CNN with LSTM and GRU. The performance of hybrid methods has been shown to be superior to standard DL classifiers [110].
- *Ensemble methods:* Ensemble learning can be employed in combining deep learning models. Additionally, the models can be trained on different resampled data. Ensemble techniques such as EasyEnsemble [111] and Balanced Bagging [112] are effective in creating ensemble models that are well-suited to handle imbalanced data.

B. LACK OF SUFFICIENT DATA

At the moment, there are not enough real-world credit card datasets to create reliable models for a variety of reasons, mostly pertaining to privacy concerns [113]. Also, most available data are unlabelled. Hence, it takes extra effort to label the data. Therefore, a frequently used technique to identify fraud is anomaly detection. However, anomaly detection depends on user behaviour, and any deviation can be interpreted as fraud. Systems that detect anomalies rely on users' past behaviour, which has limitations.

A potential approach used to solve this problem includes instance generation using DNNs: Generative models based on deep neural networks can be adapted to function similarly to oversampling methods, where artificial instances can be effectively introduced into a particular embedding space by an encoder/decoder pair. To learn the latent distribution of data, researchers have successfully used generative adversarial networks (GANs), variational autoencoders (VAEs), and Wasserstein autoencoders (WAEs) [114]. These methods can be expanded to generate more data for CCFD modelling.

C. INTERPRETABILITY

Deep learning models are often considered black-box models, making it challenging to interpret their results. Understanding why a transaction was classified as fraud or legitimate can be difficult. Meanwhile, achieving complete interpretability in DL models can be difficult. The following techniques can be employed to enhance the transparency and understandability of the deep learning model's decision-making process, making it more useful in practical applications, such as credit card fraud detection.

- *Regularization Techniques:* Applying regularization techniques, such as L1 regularization that encourages sparsity in the model's parameters, could result in simpler and more interpretable models [115].

- *SHapley Additive exPlanations (SHAP):* This method uses Shapley values from game theory in explaining a given prediction. The SHAP values assign contributions to every feature used in making the prediction [116]. It provides a method to understand the significance of each feature in the decision-making process of the deep learning model.
- *Model Distillation:* This technique involves transferring knowledge from a large model to a smaller model [117]. Examples of large models include deep learning and ensemble learning-based models. Using this approach, a simpler and interpretable model can be trained on the predictions of the DL model. Smaller models, such as decision trees and logistic regression, are easy to interpret.

D. DATA DRIFT

Many ML models are built on the assumption that the data distribution used in training and testing remains stationary. Data drift, also known as covariate shift, occurs when the distribution of the data used in training the model differs from the distribution of the data on which the model is being applied [118]. It is a common problem in many real-world systems, such as credit card fraud detection. Therefore, credit card data needs to be monitored regularly for changes in the statistical properties, and this can be achieved via visualisation, statistical tests, and observing key metrics. Models trained on older data may not effectively detect new fraud patterns. Some of the methods used in solving this problem include:

- *Incremental Learning:* Incremental learning is an approach used to update a model with the latest data while retaining the learned knowledge from the previous training. This ensures the entire model is not retrained, and methods such as transfer learning, online learning, and fine-tuning can be used to achieve such incremental learning [119].
- *Adaptive Learning Rate:* The learning rate of deep learning models can be adjusted during training to adapt to changes in the data distribution [119]. Specifically, lower learning rates can be used to ensure the model converges to a new distribution without forgetting the previous data.
- *Model retraining:* A well-known method for handling data drift is retraining the DL model with new data. Such retraining can be automated and set at regular intervals or manually triggered when sufficient drift is observed.
- *Ensemble modelling:* Ensemble models can be used to combine the predictions from multiple DL models, where one or more models can be used to determine data drift and modify the ensemble model's composition accordingly [120].

E. PRIVACY AND SECURITY CONCERNS

Using credit card transaction data containing personal and financial information raises concerns about data breaches and unauthorized access. To address these concerns, researchers and practitioners must implement robust security measures, such as data anonymization and encryption, to ensure the confidentiality and integrity of the data [121]. Data anonymization involves removing or obfuscating personally identifiable information from the data while maintaining the core patterns and characteristics necessary for training the deep learning models. It can be achieved using generalization, suppression, or perturbation techniques.

Generalization involves substituting specific values with more general categories or ranges [122]. For example, instead of using exact transaction amounts, the data can be grouped into ranges, such as <USD20, USD20-USD50, and USD50-USD100, etc., preserving the overall distribution of transaction amounts while protecting individual transaction details. Suppression entails removing sensitive attributes, such as credit card numbers and the cardholder's name, ensuring that no personal information is accessible. Another method for anonymizing sensitive financial data is perturbation. It entails adjusting the values of particular attributes or introducing random noise [123]. In perturbation, transaction amounts can be perturbed by adding a small random value to each amount, making it difficult to determine the exact values while maintaining the data's statistical properties.

In addition to data anonymization, encryption is crucial in ensuring the security of sensitive financial data used for training deep learning models. Encryption entails converting the data into a format only accessible with the correct decryption key [124]. It ensures that the data will remain unreadable and unusable even if it is intercepted or viewed without authorization. Asymmetric and symmetric encryption are two examples of the different encryption methods that can be used. Asymmetric encryption employs two keys: a public key for encryption and a private key for decryption. Symmetric encryption uses a single key for both encryption and decryption.

F. ETHICS AND FAIRNESS

When constructing credit card fraud detection models using deep learning techniques, it is crucial to ensure that these models do not exhibit bias towards particular individuals or groups based on factors such as ethnicity, gender, or socioeconomic status [125]. One challenge in achieving fairness is the potential for bias in the training data. If the training data is skewed towards specific groups, the resulting model may likewise demonstrate bias in its predictions. For instance, if the majority of the training data consists of fraudulent transactions from a particular demography, the model can unfairly identify transactions from that demographic as fraudulent, resulting in biased treatment.

To address this challenge, researchers and practitioners must carefully curate the training data to ensure the inclusion of a wide range of demographic groups. This can be accomplished by gathering data from diverse sources and ensuring that the data is evenly distributed among different

groups. In addition, methods such as data augmentation can be employed to artificially enhance the presence of under-represented groups in the training data. Another approach to addressing bias in deep learning models is the utilization of fairness metrics and algorithms. Fairness metrics measure the degree of fairness or bias in the predictions made by the model, whereas fairness algorithms aim to reduce any identified bias [126]. For example, one approach involves modifying the decision threshold of the model according to various demographic groupings to provide equivalent sensitivity to fraudulent transactions across all groups.

G. ADAPTABILITY AND SCALABILITY

Another challenge is the adaptability and scalability of deep learning-based credit card fraud detection models. Given the ever-changing nature of fraud, it is imperative for the models to be adaptable and have the ability to identify new and emerging patterns of fraudulent activity [127], [128]. One challenge in achieving adaptability is the need for continuous model updates and retraining. Conventional machine learning models sometimes necessitate manual feature engineering and the retraining of models, which can consume significant time and resources. However, deep learning models have the ability to automatically learn and adapt to new patterns without requiring user intervention. Meanwhile, this requires access to up-to-date and relevant data for training.

Possible solutions to this problem include the use of techniques such as transfer learning and online learning. Transfer learning is utilising pre-trained deep learning models trained with large-scale datasets and fine-tuning them for the specific objective of credit card fraud detection. It enables the model to leverage the information and patterns acquired from the large datasets while adjusting to the particular fraud detection objective. Online learning enables the model to consistently update and acquire knowledge from newly accessible data. Online learning allows for incremental modifications based on new data rather than retraining the entire model from scratch [129]. This makes it more adaptable and scalable in detecting new fraud patterns. Furthermore, a vital aspect of the adaptability and scalability challenge pertains to the computational resources necessary for the training and deployment of deep learning models. Organisations with limited resources or infrastructure often face challenges while training deep learning models due to the substantial data and processing power requirements.

Lastly, to tackle this issue, academics can investigate methodologies like distributed computing and cloud computing. Distributed computing entails the distribution of computational tasks among numerous machines or nodes, enabling accelerated and more efficient training of deep learning models. Distributed computing can be achieved by utilising parallel processing and distributed training frameworks [130]. Cloud computing enables users to access flexible and readily available computing resources via the internet. Organisations can optimise the training and deployment of DL models by utilising cloud computing systems, which enable them to flexibly adjust their computational resources according to their requirements.

IX. DISCUSSIONS AND FUTURE RESEARCH DIRECTIONS

Deep learning models have significantly transformed numerous domains, including fraud detection. This research concisely describes the main deep learning-based architectures used for credit card fraud detection, including simple RNN, LSTM, GRU, BiLSTM, BiGRU, and CNN. The effectiveness of these models in real-world situations, particularly in the dynamic credit card fraud detection field, varies. Firstly, MLP has gained extensive usage in diverse applications due to its ability to learn complex patterns and make accurate predictions. However, its effectiveness in credit card fraud detection needs has been examined and found to be limited. The fundamental reason for this is that MLP does not possess the ability to capture temporal dependencies and sequential patterns, which are essential in detecting fraudulent activities.

On the other hand, initially designed for image analysis, CNN has demonstrated encouraging results in detecting credit card fraud, as shown in Table 2. By considering the transaction data as a two-dimensional image, CNN can effectively extract relevant features and identify fraudulent patterns. However, it is also limited with regard to credit card fraud detection. Furthermore, simple RNN, LSTM, GRU, BiLSTM, and BiGRU have been explored for credit card fraud detection. Simple RNN, although capable of capturing temporal dependencies, has struggled with long-term dependencies, limiting its effectiveness in this domain. Conversely, LSTM has demonstrated exceptional performance due to its ability to retain information over long sequences, making it well-suited for credit card fraud detection [18]. GRU, a variant of LSTM, has also shown promising results, combining the ability to retain information with a simplified architecture. BiLSTM and BiGRU, which incorporate bidirectional processing, have been found to further improve the accuracy of fraud detection models by considering both past and future contexts.

Several key conclusions can be drawn from this research. Firstly, deep learning architectures play a crucial role in efficiently detecting credit card fraud. The performance of the models differs with changes in the distribution of the samples. For example, models trained with balanced datasets achieve more robust performance than those trained with imbalanced data. Therefore, effective data resampling and engineering should be considered before model training. Also, optimizing deep learning models to consider the imbalanced nature of the data is beneficial.

Secondly, though several deep learning-based architectures have been employed for detecting credit card fraud, the following architectures have been widely utilized: CNN, LSTM, GRU, and other RNN variants. Even though they can be computationally expensive compared to traditional ML algorithms, they usually achieve higher performance. Thirdly, different research works have used single DL classifiers,

achieving excellent classification performance. However, some researchers have explored hybrid deep learning models, which perform significantly better than single deep learning models. Also, the ensemble of deep learning models has led to superior performance compared to single deep learning models. However, it increases the computational complexity of the model.

Furthermore, future research in credit card fraud detection using deep learning can explore several promising avenues to enhance the robustness, accuracy, and applicability of detection systems. One critical area is the development and implementation of hybrid and ensemble deep learning architectures. Combining different models, such as LSTM with CNN or GRU with Transformer models, can leverage the strengths of each architecture to improve overall performance. These hybrid models can potentially provide more accurate detection by capturing both temporal dependencies and spatial features of transaction data. Additionally, ensemble methods, which integrate multiple models' predictions, can enhance the system's robustness by reducing the variance and bias associated with individual models.

Moreover, while hybrid and ensemble models hold great promise, their practical deployment often faces challenges related to computational complexity and resource requirements. Future research can focus on optimizing these models to make them more efficient and scalable for real-world applications. Techniques such as model pruning, quantization, and the use of efficient neural network architectures can significantly reduce computational overhead without sacrificing accuracy. Investigating the trade-offs between model complexity and performance and developing adaptive models that can dynamically adjust their complexity based on the available computational resources will be crucial for deploying these advanced systems in operational environments.

Another important direction for future research is enhancing the interpretability and explainability of deep learning models in fraud detection. As these models become more complex, understanding their decision-making processes becomes more challenging, yet it is essential for gaining trust from users and meeting regulatory requirements. Research should focus on developing methods that can provide clear and actionable insights into how models make predictions. Techniques like attention mechanisms, SHapley Additive exPlanations, and layer-wise relevance propagation can help explain the inner workings of deep learning models. Additionally, integrating these interpretability methods with real-time fraud detection systems will ensure that financial institutions can respond quickly and transparently to fraudulent activities, thereby improving the overall security and trustworthiness of credit card transaction systems.

X. CONCLUSION

Deep learning methods have been widely applied in different fields due to their robustness and performance. Recently, deep learning architectures have produced exceptional performance in credit card fraud detection. This paper presents a comprehensive review of the current state of deep learning applications in credit card fraud detection, highlighting the primary challenges and potential solutions. The study provides valuable insights for researchers and practitioners and can guide the development of more robust and efficient fraud detection models, ultimately contributing to more secure financial transactions and reducing the economic impact of fraud.

REFERENCES

[1] B. Lebichot, G. M. Paldino, W. Siblini, L. He-Guelton, F. Oblé, and G. Bontempi, "Incremental learning strategies for credit cards fraud detection," *Int. J. Data Sci. Anal.*, vol. 12, no. 2, pp. 165–174, Aug. 2021.

[2] X. Zhang, Y. Han, W. Xu, and Q. Wang, "HOBA: A novel feature engineering methodology for credit card fraud detection with a deep learning architecture," *Inf. Sci.*, vol. 557, pp. 302–316, May 2021.

[3] S. Bakhtiari, Z. Nasiri, and J. Vahidi, "Credit card fraud detection using ensemble data mining methods," *Multimedia Tools Appl.*, vol. 82, no. 19, pp. 29057–29075, Aug. 2023.

[4] M.-H. Yang, J.-N. Luo, M. Vijayalakshmi, and S. M. Shalinie, "Contactless credit cards payment fraud protection by ambient authentication," *Sensors*, vol. 22, no. 5, p. 1989, Mar. 2022.

[5] J. Wang, W. Liu, Y. Kou, D. Xiao, X. Wang, and X. Tang, "Approx-SMOTE federated learning credit card fraud detection system," in *Proc. IEEE 47th Annu. Comput., Softw., Appl. Conf. (COMPSAC)*, Jun. 2023, pp. 1370–1375.

[6] A. A. El-Naby, E. E.-D. Hemdan, and A. El-Sayed, "An efficient fraud detection framework with credit card imbalanced data in financial services," *Multimedia Tools Appl.*, vol. 82, no. 3, pp. 4139–4160, Jan. 2023.

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

[8] M. A. Islam, M. A. Uddin, S. Aryal, and G. Stea, "An ensemble learning approach for anomaly detection in credit card data with imbalanced and overlapped classes," *J. Inf. Secur. Appl.*, vol. 78, Nov. 2023, Art. no. 103618.

[9] T. K. Dang, T. C. Tran, L. M. Tuan, and M. V. Tiep, "Machine learning based on resampling approaches and deep reinforcement learning for credit card fraud detection systems," *Appl. Sci.*, vol. 11, no. 21, p. 10004, Oct. 2021.

[10] N. S. Alfaiz and S. M. Fati, "Enhanced credit card fraud detection model using machine learning," *Electronics*, vol. 11, no. 4, p. 662, Feb. 2022.

[11] I. D. Mienye and N. Jere, "A survey of decision trees: Concepts, algorithms, and applications," *IEEE Access*, vol. 12, pp. 86716–86727, 2024.

[12] S. Dong, Y. Xia, and T. Peng, "Network abnormal traffic detection model based on semi-supervised deep reinforcement learning," *IEEE Trans. Netw. Service Manag.*, vol. 18, no. 4, pp. 4197–4212, Dec. 2021.

[13] E. A. L. M. Btoush, X. Zhou, R. Gururajan, K. C. Chan, R. Genrich, and P. Sankaran, "A systematic review of literature on credit card cyber fraud detection using machine and deep learning," *PeerJ Comput. Sci.*, vol. 9, p. e1278, Apr. 2023.

[14] R. Bin Sulaiman, V. Schetinin, and P. Sant, "Review of machine learning approach on credit card fraud detection," *Hum.-Centric Intell. Syst.*, vol. 2, no. 1, pp. 55–68, 2022.

[15] E. N. Osegi and E. F. Jumbo, "Comparative analysis of credit card fraud detection in simulated annealing trained artificial neural network and hierarchical temporal memory," *Mach. Learn. Appl.*, vol. 6, Dec. 2021, Art. no. 100080.

[16] P. Wang, E. Fan, and P. Wang, "Comparative analysis of image classification algorithms based on traditional machine learning and deep learning," *Pattern Recognit. Lett.*, vol. 141, pp. 61–67, Jan. 2021.

[17] I. D. Mienye and Y. Sun, "A deep learning ensemble with data resampling for credit card fraud detection," *IEEE Access*, vol. 11, pp. 30628–30638, 2023.

[18] E. Esenogho, I. D. Mienye, T. G. Swart, K. Aruleba, and G. Obaido, "A neural network ensemble with feature engineering for improved credit card fraud detection," *IEEE Access*, vol. 10, pp. 16400–16407, 2022.

[19] M. L. Gambo, A. Zainal, and M. N. Kassim, "A convolutional neural network model for credit card fraud detection," in *Proc. Int. Conf. Data Sci. Appl. (ICoDSA)*, Jul. 2022, pp. 198–202.

[20] T. Berhane, T. Melese, A. Walelign, and A. Mohammed, "A hybrid convolutional neural network and support vector machine-based credit card fraud detection model," *Math. Problems Eng.*, vol. 2023, pp. 1–10, Jun. 2023.

[21] D. Sehrawat and Y. Singh, "Auto-encoder and LSTM-based credit card fraud detection," *Social Netw. Comput. Sci.*, vol. 4, no. 5, p. 557, Jul. 2023.

[22] J. Raval, P. Bhattacharya, N. K. Jadav, S. Tanwar, G. Sharma, P. N. Bokoro, M. Elmorsy, A. Tolba, and M. S. Raboaca, "RaKShA: A trusted explainable LSTM model to classify fraud patterns on credit card transactions," *Mathematics*, vol. 11, no. 8, p. 1901, Apr. 2023.

[23] I. Benchaji, S. Douzi, B. El Ouahidi, and J. Jaafari, "Enhanced credit card fraud detection based on attention mechanism and LSTM deep model," *J. Big Data*, vol. 8, no. 1, pp. 1–21, Dec. 2021.

[24] S. Gold, "The evolution of payment card fraud," *Comput. Fraud Secur.*, vol. 2014, no. 3, pp. 12–17, Mar. 2014.

[25] K. L. Ambashtha and P. Kumar, "Online fraud," in *Financial Crimes: A Guide to Financial Exploitation in a Digital Age*. Berlin, Germany: Springer, 2023, pp. 97–108.

[26] K. Guers, M. M. Chowdhury, and N. Rifat, "Card skimming: A cybercrime by hackers," in *Proc. IEEE Int. Conf. Electro Inf. Technol. (eIT)*, May 2022, pp. 575–579.

[27] R. Van Belle, B. Baesens, and J. De Weerdt, "CATCHM: A novel network-based credit card fraud detection method using node representation learning," *Decis. Support Syst.*, vol. 164, Jan. 2023, Art. no. 113866.

[28] I. D. Mienye and Y. Sun, "A machine learning method with hybrid feature selection for improved credit card fraud detection," *Appl. Sci.*, vol. 13, no. 12, p. 7254, Jun. 2023.

[29] V. S. S. Karthik, A. Mishra, and U. S. Reddy, "Credit card fraud detection by modelling behaviour pattern using hybrid ensemble model," *Arabian J. Sci. Eng.*, vol. 47, no. 2, pp. 1987–1997, Feb. 2022.

[30] K. Randhawa, C. K. Loo, M. Seera, C. P. Lim, and A. K. Nandi, "Credit card fraud detection using AdaBoost and majority voting," *IEEE Access*, vol. 6, pp. 14277–14284, 2018.

[31] A. M. Aburbeian and H. I. Ashqar, "Credit card fraud detection using enhanced random forest classifier for imbalanced data," in *Proc. Int. Conf. Adv. Comput. Res.* Cham, Switzerland: Springer, 2023, pp. 605–616.

[32] S. E. Kafhali and M. Tayebi, "XGBoost based solutions for detecting fraudulent credit card transactions," in *Proc. Int. Conf. Adv. Creative Netw. Intell. Syst. (ICACNIS)*, Nov. 2022, pp. 1–6.

[33] K. Illanko, R. Soleymanzadeh, and X. Fernando, "A big data deep learning approach for credit card fraud detection," in *Computer Networks, Big Data and IoT*. Cham, Switzerland: Springer, 2022, pp. 633–641.

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

[35] H. Fanai and H. Abbasimehr, "A novel combined approach based on deep autoencoder and deep classifiers for credit card fraud detection," *Exp. Syst. Appl.*, vol. 217, May 2023, Art. no. 119562.

[36] Y. Xie, G. Liu, C. Yan, C. Jiang, M. Zhou, and M. Li, "Learning transactional behavioral representations for credit card fraud detection," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 4, pp. 5735–5748, Apr. 2024.

[37] Z. Wang, S. Kim, and I. Joe, "An improved LSTM-based failure classification model for financial companies using natural language processing," *Appl. Sci.*, vol. 13, no. 13, p. 7884, Jul. 2023.

[38] J. Karthika and A. Senthilselvi, "An integration of deep learning model with navo minority over-sampling technique to detect the frauds in credit cards," *Multimedia Tools Appl.*, vol. 82, no. 14, pp. 21757–21774, Jun. 2023.

[39] N. Prabhakaran and R. Nedunchelian, "Oppositional cat swarm optimization-based feature selection approach for credit card fraud detection," *Comput. Intell. Neurosci.*, vol. 2023, pp. 1–13, Jan. 2023.

[40] V. Bach Nguyen, K. G. Dastidar, M. Granitzer, and W. Siblini, "The importance of future information in credit card fraud detection," in *Proc. 25th Int. Conf. Artif. Intell. Statist.*, vol. 151, G. Camps-Valls, F. J. R. Ruiz, and I. Valera, Eds., Mar. 2022, pp. 10067–10077.

[41] K. Modi and R. Dayma, "Review on fraud detection methods in credit card transactions," in *Proc. Int. Conf. Intell. Comput. Control (IC)*, Jun. 2017, pp. 1–5.

[42] Y. Lucas and J. Jurgovsky, "Credit card fraud detection using machine learning: A survey," 2020, *arXiv:2010.06479*.

[43] K. G. Al-Hashedi and P. Magalingam, "Financial fraud detection applying data mining techniques: A comprehensive review from 2009 to 2019," *Comput. Sci. Rev.*, vol. 40, May 2021, Art. no. 100402.

[44] R. R. Popat and J. Chaudhary, "A survey on credit card fraud detection using machine learning," in *Proc. 2nd Int. Conf. Trends Electron. Informat. (ICOEI)*, May 2018, pp. 1120–1125.

[45] N. F. Ryman-Tubb, P. Krause, and W. Garn, "How artificial intelligence and machine learning research impacts payment card fraud detection: A survey and industry benchmark," *Eng. Appl. Artif. Intell.*, vol. 76, pp. 130–157, Nov. 2018.

[46] K. Pandey, P. Sachan, and N. G. Ganpatrao, "A review of credit card fraud detection techniques," in *Proc. 5th Int. Conf. Comput. Methodologies Commun. (ICCMC)*, Apr. 2021, pp. 1645–1653.

[47] M. Alamri and M. Ykhlef, "Survey of credit card anomaly and fraud detection using sampling techniques," *Electronics*, vol. 11, no. 23, p. 4003, Dec. 2022.

[48] (2018). *Credit Card Fraud Detection*. Accessed: Oct. 17, 2023. [Online]. Available: https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud

[49] (2019). *IEEE-CIS Fraud Detection*. Accessed: Oct. 17, 2023. [Online]. Available: https://www.kaggle.com/c/ieee-fraud-detection

[50] E. A. Lopez-Rojas, A. Elmir, and S. Axelsson, "PaySim: A financial mobile money simulator for fraud detection," in *Proc. 28th Eur. Modeling Simulation Symp. (EMSS)*, Sep. 2016, pp. 249–255.

[51] G. F. Montufar, R. Pascanu, K. Cho, and Y. Bengio, "On the number of linear regions of deep neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 1–9.

[52] B. Yuen, M. T. Hoang, X. Dong, and T. Lu, "Universal activation function for machine learning," *Sci. Rep.*, vol. 11, no. 1, p. 18757, Sep. 2021.

[53] I. H. Sarker, "AI-based modeling: Techniques, applications and research issues towards automation, intelligent and smart systems," *Social Netw. Comput. Sci.*, vol. 3, no. 2, p. 158, Mar. 2022.

[54] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 7553.

[55] I. D. Mienye and Y. Sun, "Effective feature selection for improved prediction of heart disease," in *Proc. Pan-African Artif. Intell. Smart Systems Conf.* Cham, Switzerland: Springer, 2021, pp. 94–107.

[56] C. Guo, B. Zhao, and Y. Bai, "Deepcore: A comprehensive library for coreset selection in deep learning," in *Proc. Int. Conf. Database Expert Syst. Appl.* Cham, Switzerland: Springer, 2022, pp. 181–195.

[57] W.-F. Zeng, X.-X. Zhou, S. Willems, C. Ammar, M. Wahle, I. Bludau, E. Voytik, M. T. Strauss, and M. Mann, "AlphaPeptDeep: A modular deep learning framework to predict peptide properties for proteomics," *Nature Commun.*, vol. 13, no. 1, p. 7238, Nov. 2022.

[58] A. Mehrish, N. Majumder, R. Bharadwaj, R. Mihalcea, and S. Poria, "A review of deep learning techniques for speech processing," *Inf. Fusion*, vol. 99, Nov. 2023, Art. no. 101869.

[59] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[60] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder–decoder for statistical machine translation," 2014, *arXiv:1406.1078*.

[61] Y. Moodi, M. Ghasemi, and S. R. Mousavi, "Estimating the compressive strength of rectangular fiber reinforced polymer–confined columns using multilayer perceptron, radial basis function, and support vector regression methods," *J. Reinforced Plastics Compos.*, vol. 41, nos. 3–4, pp. 130–146, Feb. 2022.

[62] S. R. Dubey, S. K. Singh, and B. B. Chaudhuri, "Activation functions in deep learning: A comprehensive survey and benchmark," *Neurocomputing*, vol. 503, pp. 92–108, Sep. 2022.

[63] S. S. Yadav and S. M. Jadhav, "Deep convolutional neural network based medical image classification for disease diagnosis," *J. Big Data*, vol. 6, no. 1, pp. 1–18, Dec. 2019.

[64] J. Naranjo-Torres, M. Mora, R. Hernández-García, R. J. Barrientos, C. Fredes, and A. Valenzuela, "A review of convolutional neural network applied to fruit image processing," *Appl. Sci.*, vol. 10, no. 10, p. 3443, May 2020.

[65] I. D. Mienye, P. Kenneth Ainah, I. D. Emmanuel, and E. Esenogho, "Sparse noise minimization in image classification using genetic algorithm and DenseNet," in *Proc. Conf. Inf. Commun. Technol. Soc. (ICTAS)*, Mar. 2021, pp. 103–108.

[66] R. San Miguel Carrasco and M.-Á. Sicilia-Urbán, "Evaluation of deep neural networks for reduction of credit card fraud alerts," *IEEE Access*, vol. 8, pp. 186421–186432, 2020.

[67] K. Fu, D. Cheng, Y. Tu, and L. Zhang, "Credit card fraud detection using convolutional neural networks," in *Neural Information Processing*, Kyoto, Japan. Cham, Switzerland: Springer, 2016, pp. 483–490.

[68] A. Dhillon and G. K. Verma, "Convolutional neural network: A review of models, methodologies and applications to object detection," *Prog. Artif. Intell.*, vol. 9, no. 2, pp. 85–112, Jun. 2020.

[69] M. Krichen, "Convolutional neural networks: A survey," *Computers*, vol. 12, no. 8, p. 151, Jul. 2023.

[70] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: An overview and application in radiology," *Insights into Imag.*, vol. 9, no. 4, pp. 611–629, Aug. 2018.

[71] L. González-Rodríguez and A. Plasencia-Salgueiro, "Uncertainty-aware autonomous mobile robot navigation with deep reinforcement learning," in *Deep Learning for Unmanned Systems*, Cham, Switzerland, 2021, pp. 225–257.

[72] A. Tsantekidis, N. Passalis, and A. Tefas, "Recurrent neural networks," in *Deep Learning for Robot Perception and Cognition*. Amsterdam, The Netherlands: Elsevier, 2022, pp. 101–115.

[73] P. Oliveira, B. Fernandes, C. Analide, and P. Novais, "Forecasting energy consumption of wastewater treatment plants with a transfer learning approach for sustainable cities," *Electronics*, vol. 10, no. 10, p. 1149, May 2021.

[74] J. Yang, J. Qu, Q. Mi, and Q. Li, "A CNN-LSTM model for tailings dam risk prediction," *IEEE Access*, vol. 8, pp. 206491–206502, 2020.

[75] M. Ma, C. Liu, R. Wei, B. Liang, and J. Dai, "Predicting machine's performance record using the stacked long short-term memory (LSTM) neural networks," *J. Appl. Clin. Med. Phys.*, vol. 23, no. 3, 2022, Art. no. e13558.

[76] H. Xie, M. Randall, and K.-W. Chau, "Green roof hydrological modelling with GRU and LSTM networks," *Water Resour. Manag.*, vol. 36, no. 3, pp. 1107–1122, Feb. 2022.

[77] S. Gao, Y. Huang, S. Zhang, J. Han, G. Wang, M. Zhang, and Q. Lin, "Short-term runoff prediction with GRU and LSTM networks without requiring time step optimization during sample generation," *J. Hydrol.*, vol. 589, Oct. 2020, Art. no. 125188.

[78] C. Cui, P. Wang, Y. Li, and Y. Zhang, "McVCsB: A new hybrid deep learning network for stock index prediction," *Exp. Syst. Appl.*, vol. 232, Dec. 2023, Art. no. 120902.

[79] Y.-H. Li, L. N. Harfiya, K. Purwandari, and Y.-D. Lin, "Real-time cuffless continuous blood pressure estimation using deep learning model," *Sensors*, vol. 20, no. 19, p. 5606, Sep. 2020.

[80] Y. Hao, L. Dong, F. Wei, and K. Xu, "Self-attention attribution: Interpreting information interactions inside transformer," in *Proc. AAAI Conf. Artif. Intell.*, 2021, vol. 35, no. 14, pp. 12963–12971.

[81] D. A. Tarzanagh, Y. Li, X. Zhang, and S. Oymak, "Max-margin token selection in attention mechanism," in *Proc. 37th Conf. Neural Inf. Process. Syst.*, 2023, pp. 48314–48362.

[82] G. Obaido, B. Ogbuokiri, C. W. Chukwu, F. J. Osaye, O. F. Egbelowo, M. I. Uzochukwu, I. D. Mienye, K. Aruleba, M. Primus, and O. Achilonu, "An improved ensemble method for predicting hyperchloremia in adults with diabetic ketoacidosis," *IEEE Access*, vol. 12, pp. 9536–9549, 2024.

[83] T. O'Halloran, G. Obaido, B. Otegbade, and I. D. Mienye, "A deep learning approach for maize lethal necrosis and maize streak virus disease detection," *Mach. Learn. Appl.*, vol. 16, Jun. 2024, Art. no. 100556.

[84] R. Trevethan, "Sensitivity, specificity, and predictive values: Foundations, pliabilities, and pitfalls in research and practice," *Frontiers Public Health*, vol. 5, Nov. 2017, Art. no. 308890.

[85] I. D. Mienye, G. Obaido, K. Aruleba, and O. A. Dada, "Enhanced prediction of chronic kidney disease using feature selection and boosted classifiers," in *Proc. Int. Conf. Intell. Syst. Design Appl.* Cham, Switzerland: Springer, 2021, pp. 527–537.

[86] M. Rizwan, A. Nadeem, and M. A. Sindhu, "Analyses of classifier's performance measures used in software fault prediction studies," *IEEE Access*, vol. 7, pp. 82764–82775, 2019.

[87] G. Obaido, O. Achilonu, B. Ogbuokiri, C. S. Amadi, L. Habeebullahi, T. Ohalloran, C. W. Chukwu, E. D. Mienye, M. Aliyu, O. Fasawe, I. A. Modupe, E. J. Omietimi, and K. Aruleba, "An improved framework for detecting thyroid disease using filter-based feature selection and stacking ensemble," *IEEE Access*, vol. 12, pp. 89098–89112, 2024.

[88] B. Ozenne, F. Subtil, and D. Maucort-Boulch, "The precision–recall curve overcame the optimism of the receiver operating characteristic curve in rare diseases," *J. Clin. Epidemiol.*, vol. 68, no. 8, pp. 855–859, Aug. 2015.

[89] J. F. Roseline, G. Naidu, V. S. Pandi, S. A. A. Rajasree, and N. Mageswari, "Autonomous credit card fraud detection using machine learning approach?" *Comput. Electr. Eng.*, vol. 102, Sep. 2022, Art. no. 108132.

[90] H. Najadat, O. Altiti, A. A. Aqouleh, and M. Younes, "Credit card fraud detection based on machine and deep learning," in *Proc. 11th Int. Conf. Inf. Commun. Syst. (ICICS)*, Apr. 2020, pp. 204–208.

[91] J. Forough and S. Momtazi, "Ensemble of deep sequential models for credit card fraud detection," *Appl. Soft Comput.*, vol. 99, Feb. 2021, Art. no. 106883.

[92] N. F. Aurna, M. D. Hossain, Y. Taenaka, and Y. Kadobayashi, "Federated learning-based credit card fraud detection: Performance analysis with sampling methods and deep learning algorithms," in *Proc. IEEE Int. Conf. Cyber Secur. Resilience (CSR)*, Jul. 2023, pp. 180–186.

[93] Y. Xie, G. Liu, C. Yan, C. Jiang, and M. Zhou, "Time-aware attention-based gated network for credit card fraud detection by extracting transactional behaviors," *IEEE Trans. Computat. Social Syst.*, vol. 10, no. 3, pp. 1004–1016, Jun. 2023.

[94] E. Ajitha, S. Sneha, S. Makesh, and K. Jaspin, "A comparative analysis of credit card fraud detection with machine learning algorithms and convolutional neural network," in *Proc. Int. Conf. Adv. Comput., Commun. Appl. Informat. (ACCAI)*, May 2023, pp. 1–8.

[95] M. N. Y. Ali, T. Kabir, N. L. Raka, S. S. Toma, M. L. Rahman, and J. Ferdaus, "SMOTE based credit card fraud detection using convolutional neural network," in *Proc. 25th Int. Conf. Comput. Inf. Technol. (ICCIT)*, Dec. 2022, pp. 55–60.

[96] M. Z. Mizher and A. B. Nassif, "Deep CNN approach for unbalanced credit card fraud detection data," in *Proc. Adv. Sci. Eng. Technol. Int. Conf. (ASET)*, Feb. 2023, pp. 1–7.

[97] A. Iqbal and R. Amin, "Time series forecasting and anomaly detection using deep learning," *Comput. Chem. Eng.*, vol. 182, Mar. 2024, Art. no. 108560.

[98] Y. Tang and Z. Liu, "A distributed knowledge distillation framework for financial fraud detection based on transformer," *IEEE Access*, vol. 12, pp. 62899–62911, 2024.

[99] I. D. Mienye and Y. Sun, "Performance analysis of cost-sensitive learning methods with application to imbalanced medical data," *Informat. Med. Unlocked*, vol. 25, Jan. 2021, Art. no. 100690.

[100] D. Dablain, B. Krawczyk, and N. V. Chawla, "DeepSMOTE: Fusing deep learning and SMOTE for imbalanced data," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 9, pp. 6390–6404, Sep. 2023.

[101] K. Cao, C. Wei, A. Gaidon, N. Arechiga, and T. Ma, "Learning imbalanced datasets with label-distribution-aware margin loss," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 1–12.

[102] Q. Dong, S. Gong, and X. Zhu, "Imbalanced deep learning by minority class incremental rectification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 6, pp. 1367–1381, Jun. 2019.

[103] C. Zhang, K. C. Tan, H. Li, and G. S. Hong, "A cost-sensitive deep belief network for imbalanced classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 1, pp. 109–122, Jan. 2019.

[104] S. Wang, W. Liu, J. Wu, L. Cao, Q. Meng, and P. J. Kennedy, "Training deep neural networks on imbalanced data sets," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2016, pp. 4368–4374.

[105] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2999–3007.

[106] Z. Zhang and M. Sabuncu, "Generalized cross entropy loss for training deep neural networks with noisy labels," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, 2018, pp. 1–11.

[107] Y. Cui, M. Jia, T.-Y. Lin, Y. Song, and S. Belongie, "Class-balanced loss based on effective number of samples," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9260–9269.

[108] J. A. Dar, K. K. Srivastava, and S. Ahmed Lone, "Design and development of hybrid optimization enabled deep learning model for COVID-19 detection with comparative analysis with DCNN, BIAT-GRU, XGBoost," *Comput. Biol. Med.*, vol. 150, Nov. 2022, Art. no. 106123.

[109] V. B. Semwal, A. Gupta, and P. Lalwani, "An optimized hybrid deep learning model using ensemble learning approach for human walking activities recognition," *J. Supercomput.*, vol. 77, no. 11, pp. 12256–12279, Nov. 2021.
[110] M. U. Salur and I. Aydin, "A novel hybrid deep learning model for sentiment classification," *IEEE Access*, vol. 8, pp. 58080–58093, 2020.
[111] T.-Y. Liu, "EasyEnsemble and feature selection for imbalance data sets," in *Proc. Int. Joint Conf. Bioinf., Syst. Biol. Intell. Comput.*, 2009, pp. 517–520.
[112] J. Blaszczyński and J. Stefanowski, "Actively balanced bagging for imbalanced data," in *Foundations of Intelligent Systems*, Warsaw, Poland. Cham, Switzerland: Springer, 2017, pp. 271–281.
[113] E. Altman, "Synthesizing credit card transactions," in *Proc. 2nd ACM Int. Conf. AI Finance*, Nov. 2021, pp. 1–9.
[114] A. Chakrabarty and S. Das, "Statistical regeneration guarantees of the Wasserstein autoencoder with latent space consistency," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 17098–17110.
[115] D. Urda, J. Montes-Torres, F. Moreno, L. Franco, and J. M. Jerez, "Deep learning to analyze RNA-seq gene expression data," in *Advances in Computational Intelligence*, Cadiz, Spain. Cham, Switzerland: Springer, 2017, pp. 50–59.
[116] L. Antwarg, R. M. Miller, B. Shapira, and L. Rokach, "Explaining anomalies detected by autoencoders using Shapley additive explanations," *Exp. Syst. Appl.*, vol. 186, Dec. 2021, Art. no. 115736.
[117] Y. Yang, J. Qiu, M. Song, D. Tao, and X. Wang, "Distilling knowledge from graph convolutional networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 7072–7081.
[118] H. Raza, G. Prasad, and Y. Li, "EWMA model based shift-detection methods for detecting covariate shifts in non-stationary environments," *Pattern Recognit.*, vol. 48, no. 3, pp. 659–669, Mar. 2015.
[119] S.-S. Zhang, J.-W. Liu, and X. Zuo, "Adaptive online incremental learning for evolving data streams," *Appl. Soft Comput.*, vol. 105, Jul. 2021, Art. no. 107255.
[120] K. Rahmani, R. Thapa, P. Tsou, S. Casie Chetty, G. Barnes, C. Lam, and C. Foon Tso, "Assessing the effects of data drift on the performance of machine learning models used in clinical sepsis prediction," *Int. J. Med. Informat.*, vol. 173, May 2023, Art. no. 104930.
[121] S. Savvides, D. Khandelwal, and P. Eugster, "Efficient confidentiality-preserving data analytics over symmetrically encrypted datasets," *Proc. VLDB Endowment*, vol. 13, no. 8, pp. 1290–1303, 2020.
[122] S. Murthy, A. Abu Bakar, F. Abdul Rahim, and R. Ramli, "A comparative study of data anonymization techniques," in *Proc. IEEE 5th Int. Conf. Big Data Secur. Cloud (BigDataSecurity), Int. Conf. High Perform. Smart Comput., (HPSC), IEEE Int. Conf. Intell. Data Secur. (IDS)*, May 2019, pp. 306–309.
[123] H. Lee and Y. D. Chung, "Differentially private release of medical microdata: An efficient and practical approach for preserving informative attribute values," *BMC Med. Informat. Decis. Making*, vol. 20, no. 1, pp. 1–15, Dec. 2020.
[124] A. Ali, M. F. Pasha, J. Ali, O. H. Fang, M. Masud, A. D. Jurcut, and M. A. Alzain, "Deep learning based homomorphic secure search-able encryption for keyword search in blockchain healthcare system: A novel approach to cryptography," *Sensors*, vol. 22, no. 2, p. 528, Jan. 2022.
[125] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan, "A survey on bias and fairness in machine learning," *ACM Comput. Surv.*, vol. 54, no. 6, pp. 1–35, Jul. 2022.
[126] J. Mary, C. Calauzenes, and N. El Karoui, "Fairness-aware learning for continuous attributes and treatments," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 4382–4391.
[127] A. Alharbi, M. Alshammari, O. D. Okon, A. Alabrah, H. T. Rauf, H. Alyami, and T. Meraj, "A novel text2IMG mechanism of credit card fraud detection: A deep learning approach," *Electronics*, vol. 11, no. 5, p. 756, Mar. 2022.
[128] I. D. Mienye and N. Jere, "Optimized ensemble learning approach with explainable AI for improved heart disease prediction," *Information*, vol. 15, no. 7, p. 394, Jul. 2024.
[129] M. N. Fekri, H. Patel, K. Grolinger, and V. Sharma, "Deep learning for load forecasting with smart meter data: Online adaptive recurrent neural network," *Appl. Energy*, vol. 282, Jan. 2021, Art. no. 116177.
[130] A. Arunarani, D. Manjula, and V. Sugumaran, "Task scheduling techniques in cloud computing: A literature survey," *Future Gener. Comput. Syst.*, vol. 91, pp. 407–415, Feb. 2019.

IBOMOIYE DOMOR MIENYE (Member, IEEE) received the B.Eng. degree in electrical and electronic engineering and the M.Sc. degree (cum laude) in computer systems engineering from the University of East London, in 2012 and 2014, respectively, and the Ph.D. degree in electrical and electronic engineering from the University of Johannesburg, South Africa. His research interests include machine learning and deep learning for finance and healthcare applications.

NOBERT JERE received the M.Sc. and Ph.D. degrees in computer science from the University of Fort Hare, South Africa, in 2009 and 2013, respectively. He is currently an Associate Professor with the Department of Information Technology, Walter Sisulu University, South Africa. He has authored or co-authored numerous peer-reviewed journal articles and conference proceedings, chaired/co-chaired international conferences, and serves as a reviewer for numerous reputable journals. His main research interest includes ICT for sustainable development.