future internet

MDPI

Article

Efficiency of Federated Learning and Blockchain in Preserving Privacy and Enhancing the Performance of Credit Card Fraud Detection (CCFD) Systems

Tahani Baabdullah [1,2,*], Amani Alzahrani [1,2], Danda B. Rawat [1,2,*] and Chunmei Liu [2]

[1] Data Science and Cybersecurity Center (DSC2), Department of Electrical Engineering and Computer Science, Howard University, Washington, DC 20059, USA; amani.alzahrani@bison.howard.edu
[2] Department of Electrical Engineering and Computer Science, Howard University, Washington, DC 20059, USA; chuliu@howard.edu
* Correspondence: tahani.baabdullah@bison.howard.edu (T.B.); danda.rawat@howard.edu (D.B.R.)

Abstract: Increasing global credit card usage has elevated it to a preferred payment method for daily transactions, underscoring its significance in global financial cybersecurity. This paper introduces a credit card fraud detection (CCFD) system that integrates federated learning (FL) with blockchain technology. The experiment employs FL to establish a global learning model on the cloud server, which transmits initial parameters to individual local learning models on fog nodes. With three banks (fog nodes) involved, each bank trains its learning model locally, ensuring data privacy, and subsequently sends back updated parameters to the global learning model. Through the integration of FL and blockchain, our system ensures privacy preservation and data protection. We utilize three machine learning and deep neural network learning algorithms, RF, CNN, and LSTM, alongside deep optimization techniques such as ADAM, SGD, and MSGD. The SMOTE oversampling technique is also employed to balance the dataset before model training. Our proposed framework has demonstrated efficiency and effectiveness in enhancing classification performance and prediction accuracy.

Keywords: credit card fraud; financial fraud; fraud detection; federated learning; blockchain; fog computing; cybersecurity; privacy preservation; data protection

Citation: Baabdullah, T.; Alzahrani, A.; Rawat, D.B.; Liu, C. Efficiency of Federated Learning and Blockchain in Preserving Privacy and Enhancing the Performance of Credit Card Fraud Detection (CCFD) SYSTEMS. *Future Internet* **2024**, *16*, 196. https://doi.org/10.3390/fi16060196

Academic Editor: Gianluigi Ferrari

Received: 18 April 2024
Revised: 20 May 2024
Accepted: 31 May 2024
Published: 2 June 2024

1. Introduction

1.1. Credit Card Fraud

In the contemporary era, technology permeates nearly every field of our lives, from education and healthcare to finance, economics, industry, trade, politics, and entertainment. The methods through which consumers conduct transactions have undergone profound transformation and expansion in recent years. As a result, the surge in electronic commerce (e-commerce) and online credit card transactions for purchases and payments can be directly linked to the evolution of modern lifestyles, technology advancements, and the ubiquitous presence of online applications.

According to industry research, merchants are projected to incur losses of USD 130 billion from fraudulent transactions between 2018 and 2023 [1]. Many financial institutions allocate a security budget ranging from 20% to 30%, known as extended detection and response (XDR), considered a top priority in their security investments [2]. Credit card fraud occurs when a fraudster unlawfully uses someone's credit card, either online or physically, by stealing credit card information for unauthorized transactions. Fraudsters continually innovate fraud methods to breach credit card systems and conduct unauthorized transactions. Credit card fraud, whether offline or online, can occur without the authorization or permission of the cardholder. Offline credit card fraud takes place when a fraudulent individual physically uses a credit card at a point-of-sale (POS) terminal. On the

other hand, online credit card fraud occurs when a fraudster steals credit card information and conducts unauthorized transactions or payments over the internet [3,4].

As widely acknowledged, AI and ML methodologies have showcased their efficacy and efficiency in detecting anomalies and fraud across diverse domains, including credit card transactions. Leveraging machine learning-based fraud detection presents numerous benefits, including automated fraud identification, real-time processing, minimized verification durations, and the capacity to uncover latent correlations within data. Conversely, conventional fraud detection methods entail the manual establishment of decision rules, consuming considerable time, necessitating multiple verification procedures (potentially inconveniencing users), and primarily detecting overt fraudulent activities [3,4].

1.2. Fraud Detection System Challenges and Limitations

Although many credit card fraud detection (CCFD) systems and frameworks have been proposed in the academic and industrial sectors, they face numerous challenges and limitations that disturb their efficiency and effectiveness. These issues demand heightened attention for resolution, encompassing concerns such as imbalanced data, adversarial attacks, feature engineering, real-time detection, the cost of false positives, and data privacy, as described below [5–13]:

- Unavailability of public datasets: The unavailability of public datasets, particularly real-world credit card datasets, is attributed to confidentiality concerns, making it challenging to access data due to privacy considerations for cardholders.
- Imbalanced data and skewed class distribution: The imbalance in class distribution is a critical issue that affects prediction accuracy.
- Changes of fraud methods and patterns: Fraudsters persistently devise novel attack and fraud tactics to outwit CCFD systems, allowing them to evade the detection of both new and previously unseen fraudulent transactions. They employ various adversarial techniques such as data poisoning, evasion attacks, and manipulation of input data to deceive the model.
- Changes in cardholder's behavior: The CCFD system encounters challenges in fraud detection due to the continuous changes in cardholders' behavior over time. These changes are not uniform across all cardholders' lives.
- Concept drift: This occurs when the ML classifier model, trained on historical data, becomes outdated or less effective when deployed in a dynamic environment due to changing data distribution. This can prevent its ability to detect all fraud patterns, as expected.
- Real-time detection system: Establishing a real-time detection system is essential for effective fraud prevention. However, it becomes increasingly challenging in environments with high transaction volumes and large datasets. Any delay in fraud detection can lead to financial losses for cardholders and financial institutions.
- False alarms: A rise in false alarms can detrimentally affect the accuracy and reliability of CCFD systems, potentially causing problems for individuals and financial institutions.
- Data privacy: Data privacy and protection are crucial aspects that demand heightened focus. However, the challenge lies in accessing publicly available data for conducting experiments, presenting a significant obstacle for CCFD system developers due to these concerns.
- Many ML techniques classify one class more accurately than the other: Each machine learning (ML) or deep learning (DL) algorithm possesses its own set of advantages and disadvantages, which allow it to detect certain patterns while potentially missing others effectively. Some algorithms may perform well in detecting one class of data but struggle with others, or they may excel in specific detection systems but falter in others.

Therefore, there is a pressing need to develop robust CCFD systems that integrate various techniques to enhance their strength and efficiency. These techniques may include cloud computing, fog computing, edge computing, IoT, blockchain, and more. Additionally,

integrating a combination of machine learning (ML) or deep learning (DL) algorithms is essential to bolster the strength of the CCFD system, leveraging the advantages of each algorithm to create a robust detection system.

1.3. Problem Statement and Motivation

The increase in credit card fraud is readily apparent in various aspects of modern life, both physical and electronic. This issue has garnered significant attention in research circles, being recognized as a trending topic due to its direct impact on individuals and financial institutions, leading to financial losses. Consequently, numerous credit card fraud (CCFD) systems have been proposed in academic and industrial domains. The primary focus of these systems has been to enhance fraud detection accuracy and overall performance. Numerous challenges and issues confront credit card fraud detection (CCFD) systems, presenting obstacles and security vulnerabilities. These include preserving privacy, preserving cardholders' data, ensuring the availability of online fraud detection systems, detecting previously unseen fraudulent transactions, adapting to evolving fraud methods, and understanding changes in cardholders' behavior.

While deploying these CCFD systems on cloud servers offers heightened performance and computational capabilities, it also raises data privacy and protection concerns. Therefore, our objective is to enhance the performance and accuracy of these CCFD systems while preserving privacy and data protection. To achieve this, our paper proposes a robust CCFD system that integrates various techniques, including cloud computing, fog computing, federated learning, and blockchain.

Specifically, we concentrate on integrating federated learning with blockchain to enhance the classification performance, prediction accuracy, and data protection capabilities of our CCFD system.

1.4. Proposed Work Contributions

We propose the blockchain-federated learning credit card fraud detection system, which includes the following key contributions:

- Addressing the issue of skewed classes in the datasets, we employ the Synthetic Minority Oversampling Technique (SMOTE) to balance the dataset before training our models.
- Recognizing the significance of preserving privacy and data protection for maintaining institutional reputations, we incorporate blockchain and federated learning (FL) into our credit card fraud detection system to ensure these aspects for our cardholders (clients).
- Utilizing three machine and deep learning algorithms (RF, CNN, LSTM) to develop multiple, self-improving, and maintainable fraud detection models enhances the system structure and learning module.
- Incorporating deep optimization techniques (ADAM, SGD, MSGD) into our detection system to adjust neural network attributes, such as weights and learning rates, reduces overall loss and improves prediction accuracy.

1.5. Paper Organization

The rest of this article is organized as follows: Section 2 explains the background, and Section 3 discusses the related research papers. Section 4 proposes the design for our predictive framework and the methodology used in our experiment. Section 5 describes the experimental results and performance evaluation. Finally, Section 6 presents the conclusion of this paper and summarizes this and future work.

2. Background

Federated learning (FL), as shown in Figure 1, is a machine learning technique that significantly contributes to preserving privacy and protecting data during the learning process. In federated learning, two types of learning models exist: local and global learning

models. The global learning model resides in the cloud server and distributes parameters to the local learning models. These local fog node models receive the parameters and conduct model learning locally. Subsequently, the updated parameters are sent back to the global model. These updated parameters iteratively proceed forward and backward until reaching the target with the minimum error, as explained in Algorithm 1 [14,15]. Integrating of FL and blockchain in credit card services ensure preserved privacy, data protection, decentralized storage, secure payment networks, and automated tasks [16,17].

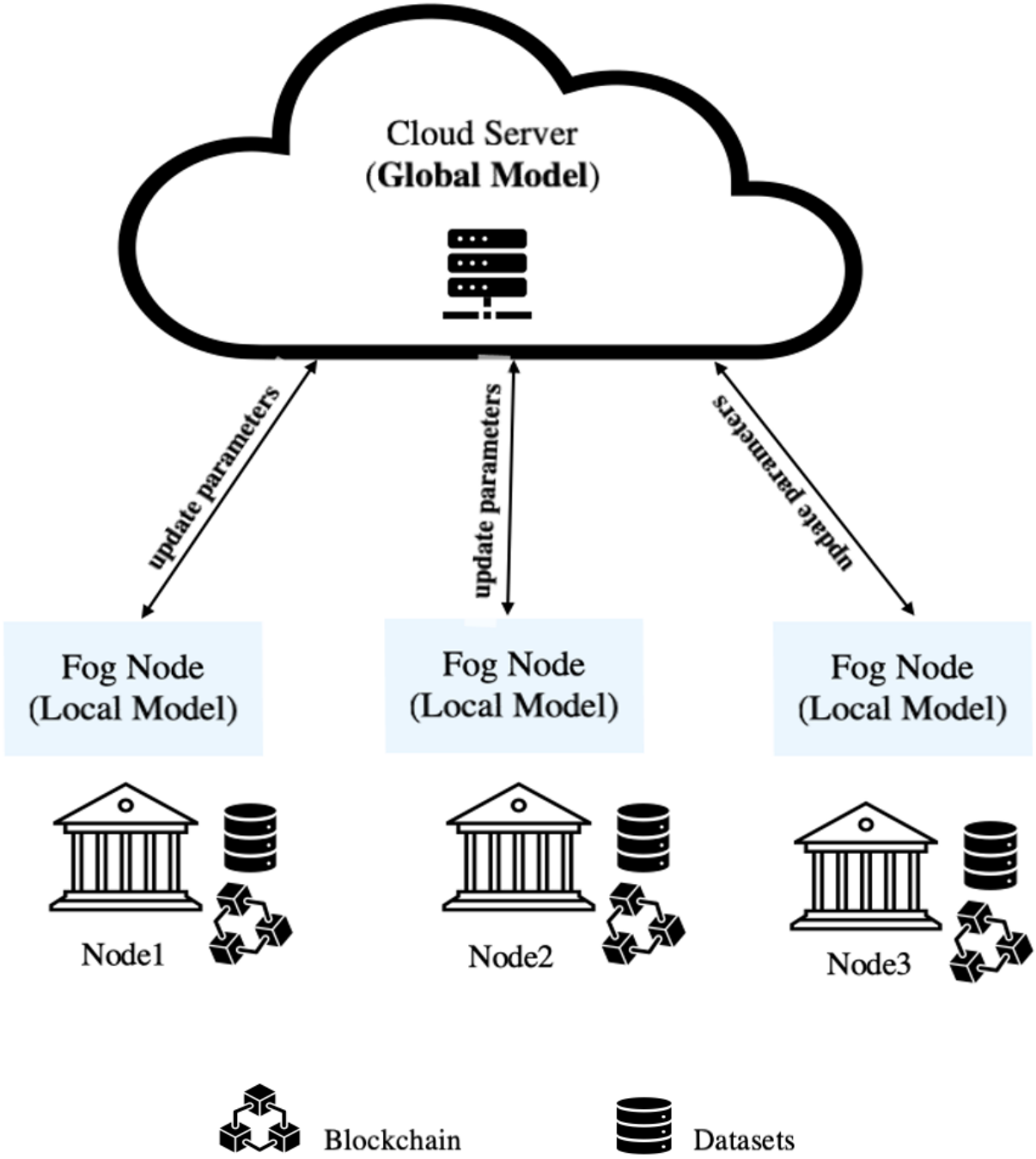

Figure 1. Federated learning.

Algorithm 1: Federated Learning Algorithm

Data: Node local datasets $\{D_1, D_2, ..., D_n\}$;
Global model parameters θ;
Learning rate η
Result: Updated global model parameters θ
Initialize global model parameters θ;
while *not reached convergence* **do**
  Randomly select a subset of nodes (local models) $N \subset \{1, 2, ..., n\}$;
  for *each node* $i \in N$ **do**
    Send current global model parameters θ to node i;
    Node i computes local updates using dataset D_i: $\theta_i \leftarrow \theta - \eta \nabla \ell_i(\theta)$;
    Send local updates θ_i back to server (global model);
  end
  Aggregate local updates to update global model parameters: $\theta \leftarrow \frac{1}{|N|} \sum_{i \in N} \theta_i$;
end

Federated learning is a recent ML technique that has gained popularity for its ability to train ML classifier models on local devices or servers without sharing private data with cloud servers, thus ensuring privacy preservation and data protection. Various applications leverage federated learning, including healthcare, financial services, smartphones and IoT devices, autonomous vehicles, telecommunications, manufacturing, retail, energy and utilities, agriculture, and government and public services. McMahan et al. proposed a distributed ML technique called federated learning (FL) in [14]. It presented a practical method for the federated learning of deep networks based on iterative model averaging, and it conducted a comprehensive empirical evaluation across five different models and four datasets.

Zhang et al. proposed "FedSI", a novel federated continual learning method that adapts the synaptic intelligence method to the federated learning scenario. This approach aims to enhance performance on non-IID data by incorporating knowledge from other local models. They developed CFedSI, a communication-efficient federated continual learning method that reduces communication overheads by employing the bidirectional compression and error compensation (BCEC) algorithm. This BCEC algorithm works on compressing transmitted data, both uplink and downlink, while ensuring training divergence through error compensation [18].

Cicceri et al. introduced a "DILoCC" framework to oversee various devices, including wearable devices, sensors, and applications. This architecture utilized a Distributed Incremental Learning (DIL) approach, enabling collaboration among the sensing devices. Additionally, it enhances system efficiency by mitigating the repercussions of "Catastrophic Forgetting". Hence, the paper elucidated how IoMT and wearable devices enhance healthcare through predictive ICT systems. Additionally, "DILoCC" oversees wearables using a Distributed Incremental Learning approach [19].

In their paper, Rauniyar et al. [20] explored the utilization of Federated Learning (FL) technology in medical applications. They provided an overview of current research trends and outcomes aimed at designing reliable and scalable FL models. Zhan et al. introduced a taxonomy of existing incentive mechanisms for federated learning, accompanied by comparison and contrast of various approaches to incentive mechanisms [21]. Many research papers have proposed surveys and conducted in-depth discussions about federated learning technology, as indicated in Table 1.

Blockchain is a new technology with decentralized storage, tamper resistance, and traceability. It operates as a distributed ledger consisting of multiple nodes. Therefore, each node stores all data independently of a central authority. A blockchain consists of a series of blocks that hold specific information connected to form a chain structure in the chronological order in which they are generated. It utilizes cryptographic knowledge to

ensure its immutability and unforgeability. Therefore, blockchain technology prevents data tampering, and forgery, and provides flexibility in tracing data [22–24].

Table 1. The most referenced survey papers in federated learning research.

Paper	Cited #	Year	Area	Contribution
[25]	1200	2020	FL, Mobile Edge Network	FL enables collaborative training and DL for mobile edge network optimization
[26]	423	2021	FL and IoT	Survey of FL applications in IoT networks
[27]	320	2021	FL and ML	Comparing different ML-based deployment architectures on FL
[28]	313	2020	FL and ML	Overview of FL enabling technologies, protocols, and applications
[29]	310	2021	FL and IoT	Recent advances of federated learning towards enabling federated learning-powered IoT applications
[30]	268	2023	FL and data mining	Comprehensive review on federated learning systems
[31]	267	2021	FL and ML	Survey desirable criteria and future directions in communication and networking systems
[32]	240	2023	FL and edge computing	Explores the domain of personalized FL and taxonomy of PFL techniques
[33]	237	2022	FL and IoT	Surveys problem statements and emerging challenges of applying FL within heterogeneous IoT
[34]	194	2020	FL and IoT	Surveys existing studies on FL and its use in wireless IoT
[35]	151	2020	FL and ML	Highlights the need for personalization and surveys recent research on this topic
[36]	145	2022	FL and ML	Discusses several approaches that address the performance issues associated with FL impact on the security and overall performance of the IoT
[37]	134	2020	FL	Data-driven learning model-based cooperative localization and location data processing with emerging machine learning and big data methods
[38]	119	2022	FL and cybersecurity	Extensive study on the ability of FL to provide better cybersecurity and prevent various cyberattacks in real times
[21]	88	2022	FL	Surveys the incentive mechanism design for federated learning
[39]	87	2021	FL and cybersecurity	Comprehensive survey of the unique security vulnerabilities exposed by the FL ecosystem
[40]	76	2022	FL and cybersecurity	Comprehensive survey on privacy and robustness in FL over the past five years
[41]	66	2022	FL and deep learning	Analyzes and presents the main ideas based on differential privacy (DP) to guarantee users' privacy in DL and FL
[42]	64	2022	FL and blockchain	Survey on the synergy of FL and blockchain to enable drone edge intelligence for green sustainable environments
[43]	51	2023	FL and IoT and cybersecurity	Survey on the existing intrusion detection solutions proposed for the IoT ecosystem including IoT devices, fog computing, and cloud computing layers
[44]	49	2021	FL and ML	Reviews existing contemporary works and Explains the challenges of each type of FL survey
[45]	46	2023	FL and IoMT	Presents privacy-related issues in IoMT
[46]	35	2022	FL and blockchain	Presents a solution taxonomy of BC-based FL in UAVs for B5G networks
[47]	33	2023	FL and IoT	Provides a novel federal classification between cloud, edge, and fog, and presents a comprehensive research roadmap on offloading for different federated scenarios
[48]	25	2022	FL and data privacy	Survey reviews the Privacy-Preserving Aggregation (PPAgg) protocols proposed to address privacy and security issues in FL systems
[49]	23	2022	FL and privacy	Discusses the current state of research on blockchain and FL
[50]	21	2020	FL	Reviews related studies of FL to base on the baseline a universal definition gives a guiding for the future work
[51]	21	2022	FL and blockchain	Proposes an FL-based layered healthcare informatics architecture along with the case study on FL-based electronic health records (FL-EHR)
[52]	20	2024	FL and adversarial attacks	Provides a comprehensive understanding of the attacks' effect by identifying FL attacks with low budgets, low visibility, and high impact.

The integration of federated learning (FL) and blockchain introduces intricate computational and operational challenges, potentially amplifying the costs associated with deployment and maintenance. This complexity arises due to various factors, including the following [53–56]:

- Computational Overhead: Integrating these two technologies may impose significant computational demands, particularly in handling large-scale data. This arises from the necessity of coordinating multiple devices or servers for model training, the cryptographic operations required for transaction validation and the consensus mechanisms inherent in blockchain applications.
- Data Synchronization: Employing federated learning involves utilizing distributed data sources, with each node training a local model using its respective data. However, this introduces an additional layer of complexity, necessitating the maintenance of data consistency and synchronization across multiple nodes within a blockchain network.
- Scalability Challenges: While federated learning (FL) and blockchain systems are scalable to handle large volumes of data and network devices and nodes, their scalability can lead to increased communication overhead and consensus latency in FL and blockchain networks. These factors can result in performance bottlenecks and elevated operational costs.
- Security and Privacy Concerns: Federated learning (FL) relies on sharing model parameter updates and aggregating information among nodes, raising concerns about data privacy and confidentiality. Integrating FL with blockchain introduces additional privacy challenges due to the inherent transparency of blockchain technology. Furthermore, blockchain presents complex cryptographic challenges in safeguarding sensitive information while ensuring integrity.
- Regulatory Compliance: The intersection of federated learning (FL) and blockchain introduces challenges regarding regulatory compliance. Ensuring compliance with data protection regulations, privacy laws, and industry standards adds complexity and may incur additional compliance costs. Compliance with these regulations and standards is essential but may necessitate additional resources and considerations.

3. Related Work

Many researchers are interested in developing robust credit card fraud detection systems due to the increasing number of fraudulent transactions worldwide and the substantial financial losses incurred, negatively impacting financial institutions, communities, and individuals.

Wang et al. [57] presented a fraud detection system (AFLCS) employing FL models based on the CNN algorithm with Approx-SMOTE. This system not only enhanced the existing credit card fraud detection system but also significantly reduced processing time by up to 30 times without compromising performance. Integrating salt and interference items strengthened security measures, preventing external intruders and internal pretenders from accessing the original text, thereby improving the privacy and security of data.

Zheng et al. [58] proposed a novel federated meta-learning framework for fraud detection, enabling banks to learn a fraud detection model from distributed local models stored in their respective local databases. A centralized global learning model aggregates parameter updates from locally computed updates of the fraud detection model. This approach aims to safeguard privacy and protect the sensitive information of cardholders. Additionally, the authors formulated an enhanced triplet-like metric learning, designed a novel meta-learning-based classifier, and employed joint comparison with K negative samples in each mini-batch.

Abdul Salam et al. [59] introduced a credit card fraud detection (CCFD) system based on federated learning, implemented using TensorFlow Federated and PyTorch frameworks. This experiment involved a comparative analysis of different individual and hybrid resampling techniques to tackle skewed datasets. This paper clarified that hybrid resampling methods outperformed deep learning classification models, yielding superior results for machine learning classification models.

Yang et al. [60] presented a framework that utilizes federated learning to train fraud detection models locally. FFD allows banks to preserve privacy and protect data by training data locally without sharing datasets, thereby enabling accurate fraud detection. The global learning model is based on aggregating local learning models. Additionally, oversampling methods were applied to address the issue of skewed datasets.

Bian and Zheng [61] employed federated learning to enhance the accuracy of credit card fraud detection. It utilized real-world credit card transaction datasets and employed a Dirichlet distribution to randomly allocate sample data to 10 simulated banks. This paper introduces a novel algorithm that considers the weight of the positive class (class 1), and it introduces an innovative approach to model the distance within the aggregation strategy.

Our experiment was based on a comparison with the frameworks presented in two research papers [57,58], which evaluated their performance and results against a list of state-of-the-art related works. These state-of-the-art research papers have proposed credit card fraud detection methods using the same dataset, namely the European credit card dataset, including BMR [62], APATE [63], PD-FDS [64], SPD [65], CMAB [66], RawLR [67], RMNLS [68], FlowScope [69], FD-META [58], and APPROX-SMOTE [57], as compared later in our paper and listed in Table 2.

Table 2. List of related works and state of the art.

Paper	Year	Framework	Methods	Limitations
[62]	2014	BMR	Detecting credit card fraud using the Bayes minimum risk approach to achieve improved results, quantified by monetary savings	Probability estimations are not consistently accurate across all classification algorithms. Additionally, utilizing the full dataset tends to yield better results compared to under-sampling, which is less effective.
[68]	2015	RMNLS	Utilizing undersampling to address a class imbalance. Incorporating investigators' feedback to enhance the accuracy of alerts	Non-stationary data distribution. Highly unbalanced class distributions.
[63]	2016	APATE	Preventing a transaction by employing a graph-based automated fraud detection system before approving it	Utilizing a redesigned APATE tailored to the e-commerce sector's realities by incorporating a limited set of confirmed fraudulent transactions
[64]	2017	PD-FDS	A fraud detection system based on purchase density does not necessitate the presence of pre-existing fraudulent transactions	Various fraud detection systems are employed by payment service providers, as well as the potential risk of personal information exposure in credit card transactions
[65]	2017	SPD	Representing transactions in a bipartite graph enables the identification of suspicious patterns from known compromised cards. This involves defining new attributes to capture suspiciousness and employing a non-linear classifier for evaluation	Not mentioned
[66]	2018	CMAB	A modified version of the Contextual Multi-Armed Bandit algorithm outperforms commonly employed offline models in terms of cumulative rewards	Balancing between exploration and exploitation. Demonstrates superior performance over offline models, particularly in scenarios involving concept drift
[67]	2018	RawLR	Grouping cardholders according to similar transaction behaviors Aggregating transactions, extracting behavioral patterns, training classifiers, and implementing a feedback mechanism	Not Mentioned
[69]	2020	FlowScope	FlowScope surpasses state-of-the-art baselines in accurately identifying accounts engaged in money laundering, across both injected and real-world data scenarios	Current methods concentrate on detecting dense subgraphs. Decreased accuracy in detecting high-volume flows of funds.
[58]	2021	FD-META	Federated meta-learning for enhancing fraud detection. Enhanced triplet-like metric learning and classifier based on meta-learning.	Imbalanced dataset with limited instances of fraud. Data security prevents sharing of datasets between banks,
[57]	2023	APPROX-SMOTE	Enhancing the performance and accuracy of traditional fraud detection models poses a challenge	The bank credit dataset exhibits a severe imbalance between positive and negative samples. Data privacy and security concerns prevent dataset sharing among users

4. Design and Methodology

4.1. Proposed Framework

In this paper, we propose our blockchain-federated learning credit card fraud detection system based on the integration of FL with blockchain technology, as illustrated in Figures 2 and 3.

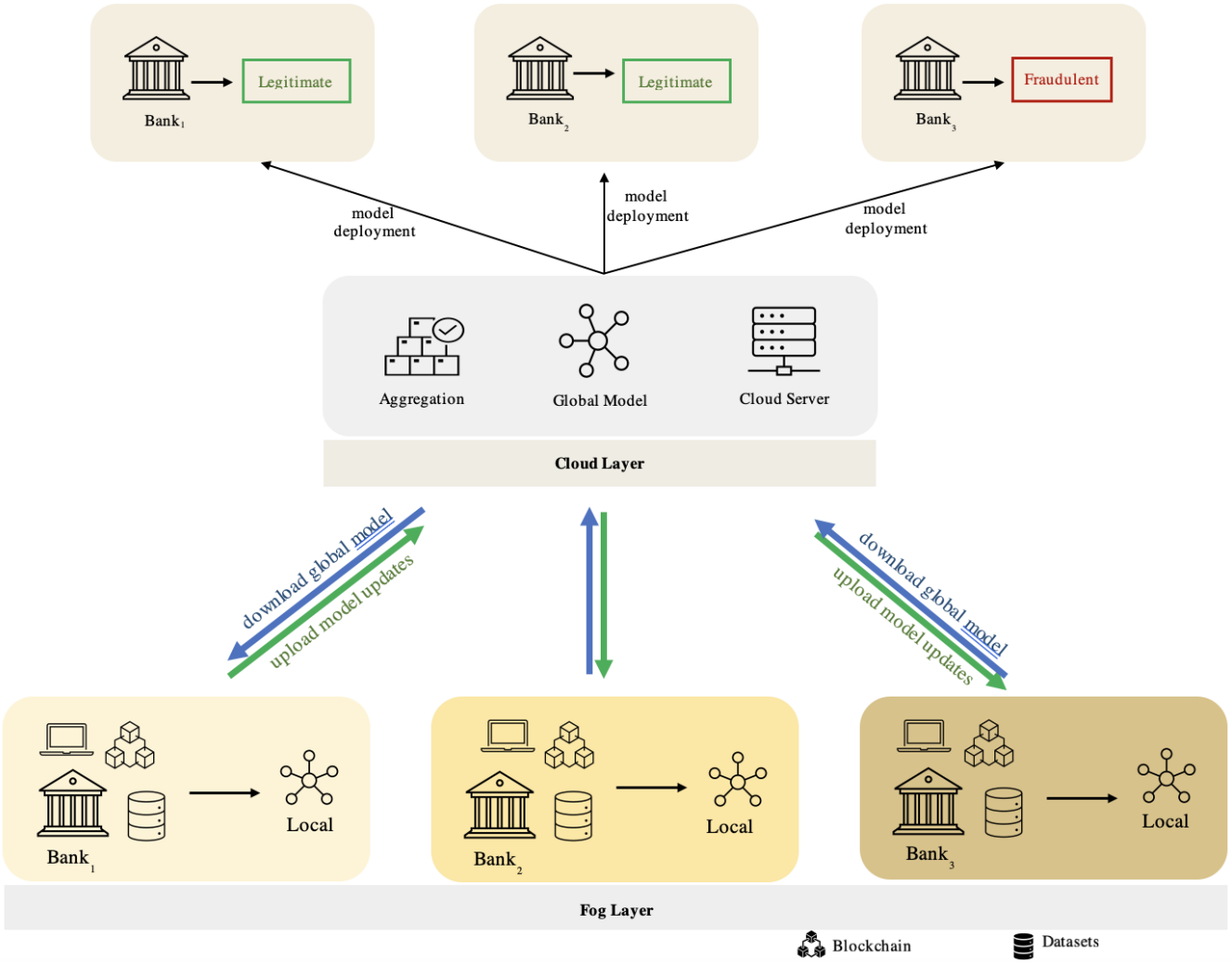

Figure 2. Proposed fraud detection framework.

Integrating FL and blockchain techniques in our detection system ensures the preservation of privacy and data protection. Federated learning ensures the accurate training of our models by sending the initial model parameters from the global learning model (cloud server) to the local learning model in each fog node (bank) individually. Each fog node represents a different bank with its local learning model. Thus, updates will be sent and received between the local and global learning models until reaching the target with the minimum loss values. Blockchain is a distributed ledger (database) shared among a computer network's nodes, used to store datasets in blocks linked together via cryptographic hashes. Blockchain is a new technology that features decentralized storage, tamper resistance, and traceability [16,17,22–24].

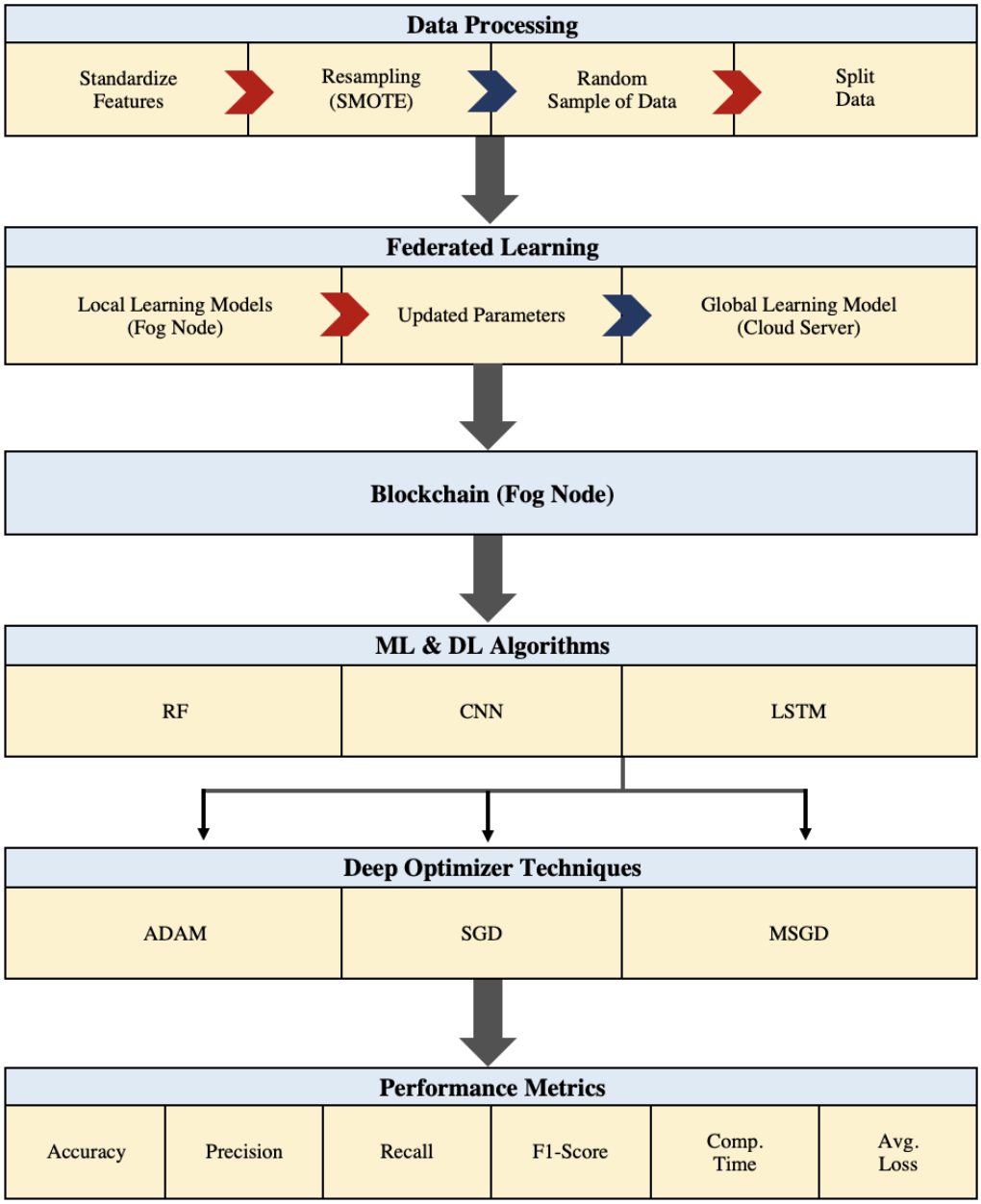

Figure 3. Framework techniques.

In addition, our framework employs three machine learning and deep neural network algorithms: Random Forest (RF), Convolutional Neural Networks (CNNs), and long short-term memory (LSTM). As demonstrated in our previous work [70], certain algorithms, such as SVM, NB, and LR, did not perform optimally in anomaly detection tasks like ours. Therefore, for this experiment, we constructed our credit card fraud detection (CCFD) system using algorithms known for their high capability and performance in accurately detecting fraud instances, namely RF, CNN, and LSTM. Leveraging their strong performance, these algorithms are poised to yield significant improvements when integrated with other techniques such as federated learning, blockchain, and fog computing. Consequently, we selected RF as the optimal classifier model among traditional machine learning algorithms, while CNN and LSTM were chosen as the deep neural network algorithms.

A deep learning optimizer is a mathematical function used to improve the weights of the network based on the gradients and other information, depending on the formulation of the optimizer. Hence, we utilize these three optimizers: ADAM, SGD, and MSGD.

In our previous work [70], we evaluated classifier models using the baseline dataset, and oversampling and undersampling techniques. The dataset exhibits class imbalance, where the number of fraudulent transactions is significantly lower than that of legitimate transactions, leading to reduced prediction accuracy. Resampling the dataset before training the classifier models is crucial to ensure accurate fraud detection. Undersampling techniques, such as NearMiss, aim to address the skewed dataset by retaining transactions close to the minority class and discarding those further away. However, this approach reduces the dataset size, potentially negatively impacting accuracy.

Moreover, we utilized the Synthetic Minority Oversampling Technique (SMOTE) to balance the dataset and alleviate class imbalance before model training. SMOTE augments the number of minority class instances, effectively addressing the imbalance. We have applied SMOTE in various experiments across different research papers, and its efficiency and effectiveness have been validated when compared to both the baseline dataset and the NearMiss Undersampling technique [70].

We utilize performance metrics to evaluate our predictive classifier models, including accuracy, precision, recall, F1-score, computation time, and average loss.

Our proposed framework distinguishes itself by integrating blockchain technology with federated learning, ensuring high privacy preservation and data protection. Additionally, we employ several deep learning optimization techniques to enhance algorithm performance. Collectively, these techniques set our proposed framework apart.

4.2. Dataset Overview

The Europe Credit Card (ECC) dataset is utilized in this experiment, representing an imbalanced real-world dataset by ULB (Université Libre de Bruxelles) on big data mining and fraud detection. It encompasses credit card transactions made by European cardholders in September 2013. The dataset comprises all transactions occurring over two days, with 492 instances of fraud identified out of 284,807 transactions, as explained in Table 3.

Table 3. Overview of the Kaggle dataset: ECC

	Fraud	**Non-Fraud**
Transaction	492	284,315
Class	1	0

As evident, the dataset exhibits highly imbalanced data, with fraudulent transactions accounting for approximately 0.172% of all transactions. As illustrated in Figure 4, ECC exhibits a highly imbalanced dataset, resulting in imbalanced class distribution and skewed data that impact the training of classifier models and prediction accuracy. The dataset exhibits class imbalance, where the number of fraudulent transactions is significantly lower than that of legitimate transactions, leading to reduced prediction accuracy. Resampling the dataset before training the classifier models is crucial to ensure accurate fraud detection. The substantial disparity between the two classes, with class 0 representing original transactions and class 1 representing fraudulent transactions, can hinder the ability of trained models to recognize fraud patterns and identify fraudulent transactions. Therefore, resampling techniques must be applied in the skewed dataset to process the data before training the model.

Resampling techniques encompass oversampling and undersampling methods. Undersampling techniques, such as NearMiss, aim to address the skewed dataset by retaining transactions close to the minority class and discarding those further away. However, this approach reduces the dataset size, potentially impacting accuracy negatively. Moreover, we utilized the Synthetic Minority Oversampling Technique (SMOTE) to balance the dataset and alleviate class imbalance before model training. SMOTE augments the number of minority class instances, effectively addressing the imbalance. We have applied SMOTE in various experiments across different research papers, and its efficiency and effective-

ness have been validated when compared to both the baseline dataset and the NearMiss Undersampling technique [70].

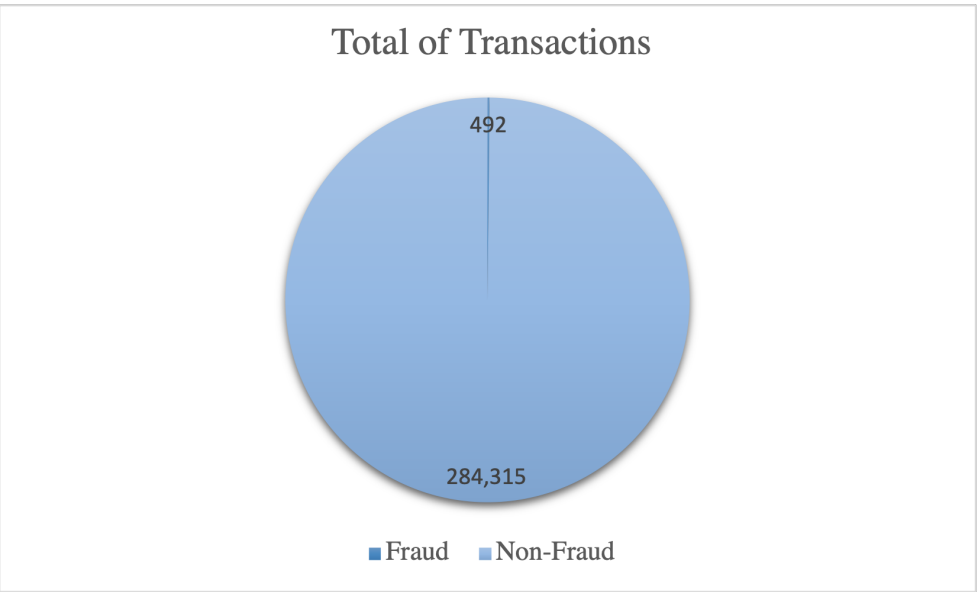

Figure 4. Imbalanced dataset ECC.

The dataset consists of numerical input variables and has undergone PCA transformation due to confidentiality concerns. Consequently, providing the original features or further background information is not feasible. The features include 30 principal components derived from PCA (V1 through V28), with "Time" and "Amount" being the only features not subjected to PCA transformation. Additionally, the "Class" feature serves as the output variable, taking a value of 1 in the case of fraudulent transactions and 0 for legitimate transactions (https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud (accessed on 1 April 2024)).

4.3. Data Preprocessing and Feature Extraction

Data preprocessing is the initial step in training our models. The dataset comprises many transactions, totaling 284,807, prompting us to randomly select 40% of the data for preprocessing. Subsequently, the data are divided into a 70% portion for the training dataset and a 30% portion for the testing dataset. Furthermore 70% is further partitioned within the training data, with 60% allocated to the training dataset and 10% to the validation dataset. The dataset undergoes cleaning and PCA transformation, after which we scale/normalize the data to standardize features and mitigate outlier instances. Following this, we execute the subsequent steps: converting NumPy arrays to PyTorch tensors, establishing data loaders, and segregating features (input) from labels (output). Notably, the features consist of 30 principal components derived from PCA (V1 through V28), with "Time" and "Amount" remaining unaltered by PCA transformation. Additionally, the "Class" feature functions as the output variable, taking 1 for fraudulent transactions and 0 for legitimate ones. All features except the "Class" feature serve as inputs.

4.4. Methodology

The proposed detection involves applying FL integrated with blockchain on a fog node. Each fog node represents a different bank with its local learning model. Federated learning ensures the accurate training of our models by sending the initial model parameters from the global learning model (cloud server) to the local learning model in each fog node (bank) individually. The FL algorithm is usually based on the following steps [53–56]:

1. Initialization: A global learning model is initialized on a cloud server with random parameters generated to create and initialize it.
2. Model Distribution: Each local model (bank) downloads the global model from the cloud server. The parameter initialization is distributed to fog nodes (banks), which serve as local learning models.
3. Local Training: Each bank trains its local learning model using its local data, starting from the downloaded global model. During training, the local model updates its parameters based on its local dataset.
4. Model Aggregation: Upon completing local training, each local model sends its updated parameters back to the cloud server (global model). The global model aggregates these updates from all local models to prepare a new global model with the recent parameter updates.
5. Global Model (Updated Parameters): The updated model parameters are frequently communicated back to the cloud server after training at each bank.
6. Iteration: Steps 3 to 5 are iteratively repeated until the target with the minimum loss is achieved. Each round involves local training on individual banks, followed by model aggregation at the cloud server.
7. Convergence: After multiple rounds of training, the global model converges to a state where it has captured knowledge from all local models (fog nodes) while preserving privacy and data protection.
8. Deployment: Once training is complete, the final global model is deployed, while individual banks retain their local data without sharing them with the cloud server.

We assumed three banks for experimental purposes, but the framework can accommodate any number of banks. Each bank will train its local learning model using the parameters received from the global model. Using FL helps keep the data on their local server more confidential and preserves privacy. Additionally, our system involves blockchain technology, which provides an immutable ledger to facilitate faster information reception, more accurate data processing, risk reduction, transaction recording, asset tracking in a business network, and more [16,17].

5. Experiment Results and Performance Evaluation

5.1. Predictive Models and Performance Metrics

The architectures and hyperparameters of the CNN and LSTM models used in our experiment are summarized in Table 4. The global model parameters consist of configurations for two neural network architectures: Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM). For the CNN, the input dimension is set to 30, indicating the size of input features, while the hidden dimension is also 30, determining the number of filters in the convolutional layers. No specific number of layers is specified for the CNN. The learning rate for optimization is set at 0.001, determining the step size during parameter updates, and the training is conducted for 10 epochs with a batch size of 64 samples. The log interval parameter defines the frequency at which the training progress is logged, set to every 10 batches. For the LSTM model, the input and hidden dimensions are both set to 30, specifying the size of input features and the numbers of hidden units in the LSTM layers, respectively. One layer of LSTM is employed in this configuration. Similarly, the learning rate, number of epochs, batch size, log interval, and splitting ratio remain consistent with the CNN settings. Additionally, the data splitting ratio is set at 70:30, indicating the proportion of data allocated for training and testing/validation, respectively. The model operates across three separate banks, possibly denoting different data sources or subsets utilized in the federated learning framework.

Table 4. Architecture of the CNN and LSTM models.

	Model	
Parameter	**CNN**	**LSTM**
Input_dim	30	30
Hidden_dim	30	30
Num_layers	0	1
Learning_rate	0.001	0.01
Num_epochs	10	10
Batch_size	64	64
Log_interval	10	10
Splitting_ratio	70:30	70:30
Num_banks	3	3

ADAM, SGD, and MSGD deep learning optimizers are utilized to improve network weights based on gradients and other models' information during the training process, to ensure our predictions are as accurate and optimized as possible. Adaptive Moment Estimation (ADAM) is a popular deep optimizer algorithm for training deep learning models. It combines the ideas of RMSProp and Momentum, computing adaptive learning rates for each parameter [71] as follows:

$$\begin{aligned} m_t &= \beta_1 \cdot m_{t-1} + (1-\beta_1) \cdot g_t, \\ v_t &= \beta_2 \cdot v_{t-1} + (1-\beta_2) \cdot g_t^2, \\ \hat{m}_t &= \frac{m_t}{1-\beta_1^t}, \\ \hat{v}_t &= \frac{v_t}{1-\beta_2^t}, \\ \theta_{t+1} &= \theta_t - \frac{\eta}{\sqrt{\hat{v}_t} + \epsilon} \cdot \hat{m}_t, \end{aligned} \tag{1}$$

where

- m_t and v_t: first and second moment estimates.
- g_t: gradient at time step t.
- β_1 and β_2: decay rates for first and second moment estimates
- η: learning rate.
- ϵ: small constant added to the denominator to prevent division by zero.

Stochastic Gradient Descent (SGD) uses a randomly selected single sample for each iteration [72].

$$\theta_{t+1} = \theta_t - \eta \cdot \nabla J(\theta_t), \tag{2}$$

where

- θ_t: parameter vector at time step t.
- η: learning rate.
- $\nabla J(\theta_t)$: gradient of loss function J with respect to parameters θ_t.

Mini-batch Stochastic Gradient Descent (MSGD) updates the parameters using mini-batches of data, providing a balance between the efficiency of SGD and the stability of batch gradient descent [73].

$$\theta_{t+1} = \theta_t - \eta \cdot \frac{1}{m} \sum_{i=1}^{m} \nabla J(\theta_t; x_i, y_i), \tag{3}$$

where

- θ_t: parameter vector at time step t.

- η:learning rate.
- m: mini-batch size.
- $\nabla J(\theta_t; x_i, y_i)$: gradient of loss function J with respect to parameters θ_t by mini-batch of m samples (x_i, y_i).

We evaluate our predictive classifier models using the following performance metrics: accuracy, precision, recall, F1-score, computation time, and average loss. These metrics are evaluated based on the following parameters: true positive (TP), true negative (TN), false positive (FP), and false negative (FN). A true positive occurs when the predicted output is true and indeed true. Conversely, if the predicted output is false and indeed false, it is referred to as a true negative. A false positive arises when the predicted output is true, but in reality, it is false. Conversely, if the predicted output is false yet true, it is termed a false negative. The definitions and equations for each metric are as follows (https://www.javatpoint.com/performance-metrics-in-machine-learning (accessed on 1 April 2024)):

- Accuracy is the number of correct predictions to the total number of predictions.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

- Precision is the ratio of true positive to the total positive predictions (true positive and false positive).

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

- Recall (sensitivity) provides the accuracy for the positive instances (class 1) as fraudulent transactions.

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

- F1-Score is the ratio of true positives to the total number of positives (true Positive and false negative).

$$F1\text{-}score = 2 \times \frac{Presession \times Recall}{Presession + Recall} \tag{7}$$

The experiment is conducted using Python3 along with several open-source machine learning tools, including Scikit Learn 0.24.2, Pandas 1.1.5, Numpy 1.26.4, Matplotlib 3.3.4, Imblearn 0.8.1, Pytorch 1.13.1, and Syft 0.1.29a1 (federated). The specifications of the desktop computer used in our experiment are as follows: CPU Ryzen 5 3600x, 16 GB RAM, and Windows 11 64-bit.

5.2. Performance Evaluation

The experiment conducted in this paper involved comparing the performance of the proposed framework with other fraud detection systems introduced in previous research papers. Table 5 illustrates the performance evaluation for two deep neural network learning algorithms, CNN and LSTM with ADAM, SGD, and MSGD deep optimizer techniques.

Table 5. Models performance evaluation (our work).

	ADAM		**SGD**		**MSGD**	
Metrics	**CNN**	**LSTM**	**CNN**	**LSTM**	**CNN**	**LSTM**
Accuracy	0.94	0.95	**0.97**	0.93	0.96	0.95
Precision	0.93	0.99	**0.97**	0.97	0.98	0.99
Recall	0.95	0.90	**0.96**	0.88	0.94	0.92
F1-Score	0.94	0.95	**0.97**	0.93	0.96	0.95
Comp.Time	0.04	0.27	0.04	0.25	0.04	0.29
Avg.Loss	0.18	0.25	**0.10**	0.21	0.12	0.17

The table showcases various performance metrics, such as accuracy, precision, recall, F1-score, computation time, and average loss, as shown in Figures 5–7. CNN, coupled

with the SGD, demonstrated high performance, achieving an accuracy, precision, recall, and F1-score of 0.97, 0.97, 0.96, and 0.97, respectively. Additionally, this model exhibited a computation time of 0.04 and an average loss of 0.10.

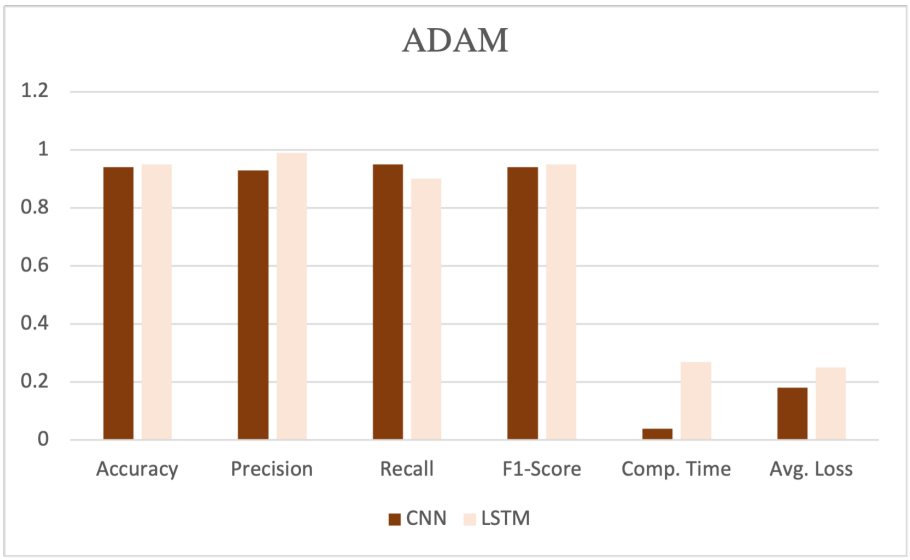

Figure 5. Performance evaluation with ADAM optimizer.

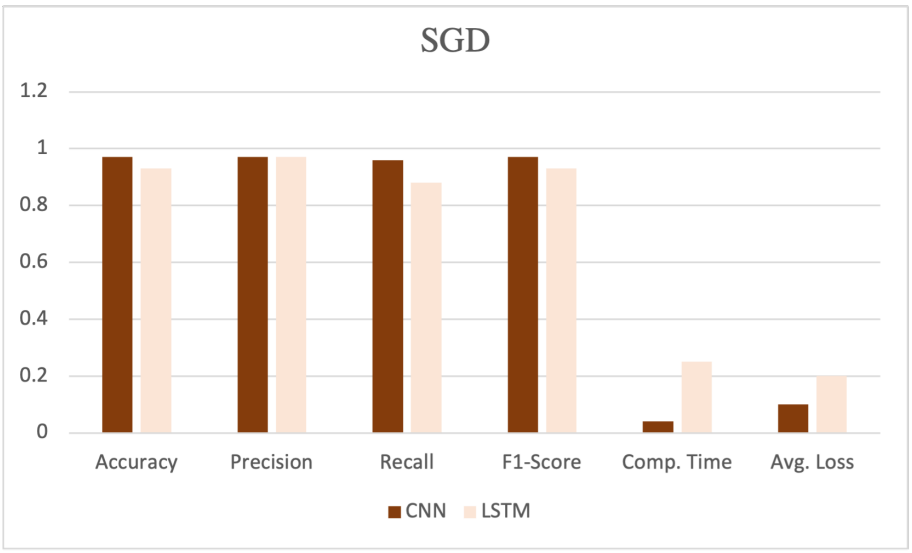

Figure 6. Performance evaluation with SGD optimizer.

Additionally, we implemented the RF algorithm integrated with the FL-blockchain (FL-blockchain-RF) framework, as shown in Table 6. We compared the performance of RF in our proposed detection system with our previous works [70,74]. FL-blockchain-RF exhibited high performance, achieving an accuracy, precision, recall, and F1-score of 0.99, 0.99, 1, and 0.99, respectively, as illustrated in Figure 8.

Table 7 presents performance comparison with other works. Based on the performance results, we observed that our proposed models exhibited good performance, achieving accuracy, precision, recall, and F1-score of 0.97, 0.97, 0.96, and 0.97, respectively.

Our framework has demonstrated its efficiency; however, our goal is to increase the accuracy, recall, and F1-score to achieve a high probability of around 0.99. Due to the size of our dataset sample and resource limitations, we encountered some constraints during this experiment. Therefore, it is feasible to further enhance its classification performance and prediction accuracy through several factors, including online fraud detection systems,

real-world datasets, dataset size, the high availability of resources, network depth, width, and cardinality. Adjusting weights, learning rate, number of hidden layers, epochs, and batch size will significantly improves the model's performance.

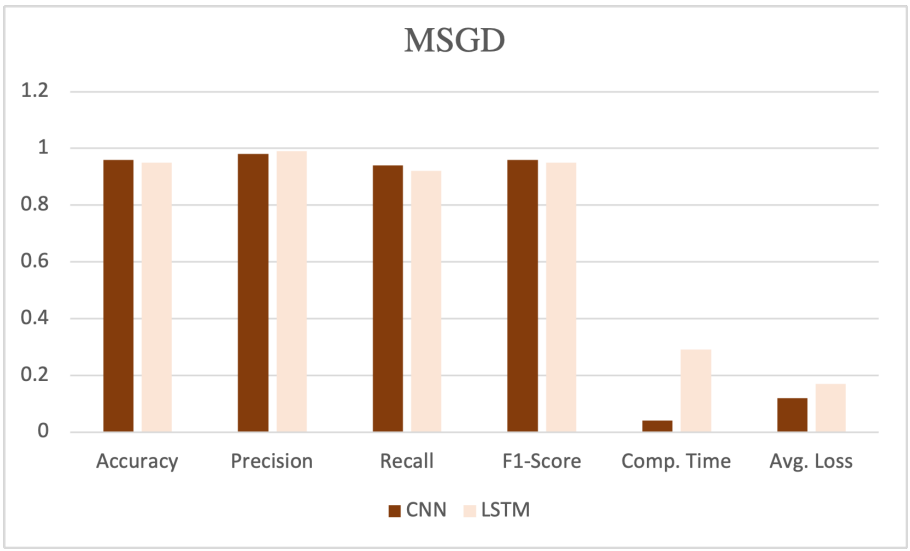

Figure 7. Performance evaluation with MSGD optimizer.

Table 6. RF performance comparison.

	Model		
Metrics	**RF [70]**	**RF [74]**	**RF (Our Work)**
Accuracy	0.99	0.96	**0.99**
Precision	0.89	0.97	**0.99**
Recall	0.85	0.92	**1**
F1-Score	0.87	0.94	**0.99**

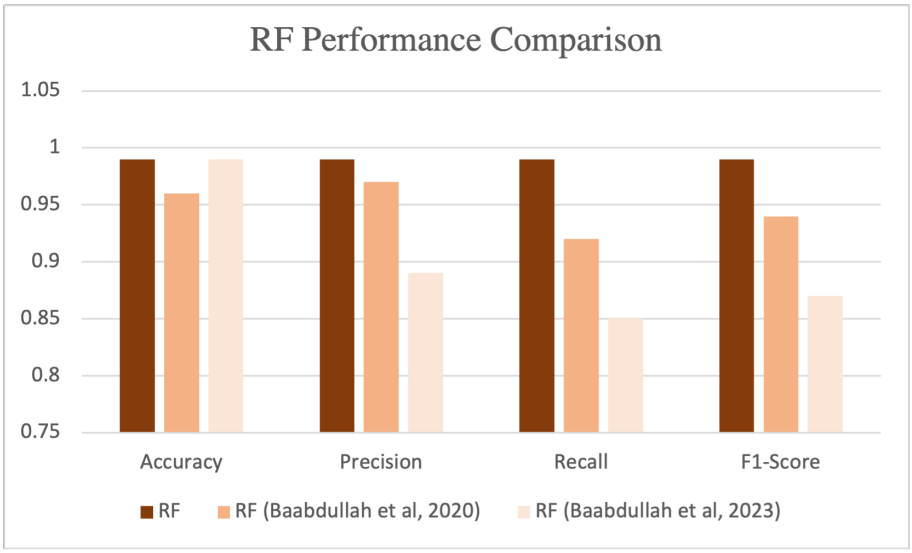

Figure 8. RF performance evaluation [70,74].

Table 7. Performance comparison with other methods.

	Metrics			
Method	**Accuracy**	**Precision**	**Recall**	**F1-Score**
BMR [62]	0.69	0.66	0.71	0.68
APATE [63]	0.70	0.66	0.71	0.69
PD-FDS [64]	0.70	0.68	0.74	0.71
SPD [65]	0.74	0.69	0.82	0.75
CMAB [66]	0.81	0.71	0.86	0.78
RawLR [67]	0.81	0.82	0.91	0.86
RMNLS [68]	0.83	0.89	0.92	0.90
Flow-Scope [69]	0.87	0.89	0.93	0.91
FD-Meta [58]	0.99	0.98	0.99	0.99
Approx-SMOTE [57]	0.98	0.98	0.97	0.98
Our work	**0.97**	**0.97**	**0.96**	**0.97**

6. Conclusions and Future Research Directions

The rising global adoption of credit cards has made them a preferred and commonly used payment option for daily transactions, exerting a significant influence on global financial cybersecurity. This study introduces a credit card fraud detection (CCFD) system employing blockchain-federated learning, which integrates federated learning (FL) with blockchain technology. Through the integration of FL and blockchain techniques, our system guarantees improved privacy, enhanced data protection, and minimized risk of data breaches. Additionally, the integration of FL and blockchain in credit card services ensures preserved privacy, data protection, decentralized storage, secure payment networks, and automated tasks. Three machine learning and deep neural network algorithms, RF, CNN, and LSTM, are utilized, alongside three optimization techniques: ADAM, SGD, and MSGD. Furthermore, the SMOTE oversampling technique is employed to balance the dataset before model training. The proposed framework has proven effective in enhancing classification performance and prediction accuracy.

Despite the vast number of credit card fraud detection (CCFD) systems and frameworks proposed in academic and industrial fields, numerous challenges and limitations adversely affect their efficiency and effectiveness. These limitations require greater attention for resolution, including issues such as imbalanced data, adversarial attacks, feature engineering, real-time detection, cost of false positives, and data privacy.

The imbalance of data is a critical issue affecting prediction accuracy due to class distribution disparities. Fraud continuously evolves, employing new attacks and fraud methods that can deceive CCFD systems and evade the detection of new and unseen fraudulent transactions. Fraudsters utilize adversarial attacks, including data poisoning, evasion attacks, and input data manipulation, to deceive the model.

Developing a real-time detection system is crucial for fraud detection, but it becomes challenging in high-volume environments and with big data. Any delay in fraud detection can result in financial losses for both cardholders and financial institutions. Moreover, an increase in false alarms can negatively impact the accuracy and integrity of CCFD systems, leading to potential issues for individuals and financial institutions.

Finally, data privacy and protection are critical aspects that warrant increased attention. However, due to these concerns, finding publicly available data for conducting experiments presents a significant obstacle for CCFD system developers.

In future work, additional efforts will be directed toward maintaining privacy and protecting data by implementing defensive measures against potential threats. Our objective is to further improve privacy and data protection by deploying a defensive system capable of detecting and preventing potential attacks or instances of fraud in real-time. Our next project involves implementing an online credit card fraud detection (CCFD) system that simulates various attacks and instances of fraud, followed by an evaluation of the system's performance. We will assess its ability to prevent, detect, and mitigate fraudulent transactions by identifying attack patterns.

Author Contributions: Conceptualization, T.B. and A.A.; methodology, T.B. and A.A.; software, T.B. and A.A.; validation, T.B., A.A. and D.B.R.; formal analysis, T.B. and A.A.; investigation, T.B. and C.L.; resources, T.B.; data curation, T.B.; writing—original draft preparation, T.B.; writing—review and editing, T.B. and A.A.; visualization, T.B.; supervision, D.B.R. and C.L.; project administration, T.B.; funding acquisition, D.B.R. and C.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded partly by the Mastercard Research Funds, Meta Gift Funds and NSF grants DMS2022448 and CCF-0939370 and Data Science and Cybersecurity Center (DSC2).

Data Availability Statement: The data is publicly available on Kaggle dataset website.

Acknowledgments: However, any opinion, finding, conclusions, or recommendations expressed in this document are those of the authors and should not be interpreted as necessarily representing the official policies expressed/implied by the funding agencies.

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

References

1. King, S.T.; Scaife, N.; Traynor, P.; Din, Z.A.; Peeters, C.; Venugopala, H. Credit Card Fraud Is a Computer Security Problem. *IEEE Secur. Priv.* **2021**, *19*, 65–69. [CrossRef]
2. Bajracharya, A.; Harvey, B.; Rawat, D.B. Recent Advances in Cybersecurity and Fraud Detection in Financial Services: A Survey. In Proceedings of the 2023 IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 8–13 March 2023; pp. 0368–0374.
3. Al Smadi, B.; Min, M. A critical review of credit card fraud detection techniques. In Proceedings of the 2020 11th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), New York, NY, USA, 28–31 October 2020; pp. 0732–0736.
4. Mead, A.; Lewris, T.; Prasanth, S.; Adams, S.; Alonzi, P.; Beling, P. Detecting fraud in adversarial environments: A reinforcement learning approach. In Proceedings of the 2018 Systems and Information Engineering Design Symposium (SIEDS), Charlottesville, VA, USA, 27 April 2018; pp. 118–122.
5. Abdallah, A.; Maarof, M.A.; Zainal, A. Fraud detection system: A survey. *J. Netw. Comput. Appl.* **2016**, *68*, 90–113. [CrossRef]
6. Zojaji, Z.; Atani, R.E.; Monadjemi, A.H. A survey of credit card fraud detection techniques: data and technique oriented perspective. *arXiv* **2016**, arXiv:1611.06439.
7. Mittal, S.; Tyagi, S. Computational techniques for real-time credit card fraud detection. In *Handbook of Computer Networks and Cyber Security: Principles and Paradigms*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 653–681.
8. Chaudhary, K.; Yadav, J.; Mallick, B. A review of fraud detection techniques: Credit card. *Int. J. Comput. Appl.* **2012**, *45*, 39–44.
9. Sadgali, I.; Sael, N.; Benabbou, F. Detection of credit card fraud: State of art. *Int. J. Comput. Sci. Netw. Secur.* **2018**, *18*, 76–83.
10. Adewumi, A.O.; Akinyelu, A.A. A survey of machine-learning and nature-inspired based credit card fraud detection techniques. *Int. J. Syst. Assur. Eng. Manag.* **2017**, *8*, 937–953. [CrossRef]
11. Delamaire, L.; Abdou, H.; Pointon, J. Credit card fraud and detection techniques: a review. *Banks Bank Syst.* **2009**, *4*, 57–68.
12. Zareapoor, M.; Shamsolmoali, P. Application of credit card fraud detection: Based on bagging ensemble classifier. *Procedia Comput. Sci.* **2015**, *48*, 679–685. [CrossRef]
13. Makki, S.; Assaghir, Z.; Taher, Y.; Haque, R.; Hacid, M.S.; Zeineddine, H. An experimental study with imbalanced classification approaches for credit card fraud detection. *IEEE Access* **2019**, *7*, 93010–93022. [CrossRef]
14. McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; y Arcas, B.A. Communication-efficient learning of deep networks from decentralized data. In Proceedings of the Artificial Intelligence and Statistics, PMLR, Fort Lauderdale, FL, USA, 20–22 April 2017; pp. 1273–1282.
15. Ntizikira, E.; Lei, W.; Alblehai, F.; Saleem, K.; Lodhi, M.A. Secure and privacy-preserving intrusion detection and prevention in the internet of unmanned aerial vehicles. *Sensors* **2023**, *23*, 8077. [CrossRef]
16. Chatterjee, P.; Das, D.; Rawat, D.B. Next Generation Financial Services: Role of Blockchain enabled Federated Learning and Metaverse. In Proceedings of the 2023 IEEE/ACM 23rd International Symposium on Cluster, Cloud and Internet Computing Workshops (CCGridW), Bangalore, India, 1–4 May 2023; pp. 69–74.
17. Bao, G.; Guo, P. Federated learning in cloud-edge collaborative architecture: key technologies, applications and challenges. *J. Cloud Comput.* **2022**, *11*, 94. [CrossRef]
18. Zhang, Z.; Zhang, Y.; Guo, D.; Zhao, S.; Zhu, X. Communication-efficient federated continual learning for distributed learning system with Non-IID data. *Sci. China Inf. Sci.* **2023**, *66*, 122102. [CrossRef]
19. Cicceri, G.; Tricomi, G.; Benomar, Z.; Longo, F.; Puliafito, A.; Merlino, G. DILoCC: An approach for Distributed Incremental Learning across the Computing Continuum. In Proceedings of the 2021 IEEE International Conference on Smart Computing (SMARTCOMP), Irvine, CA, USA, 23–27 August 2021; pp. 113–120.
20. Rauniyar, A.; Hagos, D.H.; Jha, D.; Håkegård, J.E.; Bagci, U.; Rawat, D.B.; Vlassov, V. Federated learning for medical applications: A taxonomy, current trends, challenges, and future research directions. *IEEE Internet Things J.* **2023**, *11*, 7374–7398. [CrossRef]

21. Zhan, Y.; Zhang, J.; Hong, Z.; Wu, L.; Li, P.; Guo, S. A Survey of Incentive Mechanism Design for Federated Learning. *IEEE Trans. Emerg. Top. Comput.* **2022**, *10*, 1035–1044. [CrossRef]
22. Li, W.; Yang, B.; Song, Y. Secure Multi-Party Computing for Financial Sector Based on Blockchain. In Proceedings of the 2023 IEEE 14th International Conference on Software Engineering and Service Science (ICSESS), Beijing/Guiyang, China, 17–18 October 2023; pp. 145–151.
23. Santin, G.; Skarbovsky, I.; Fournier, F.; Lepri, B. A Framework for Verifiable and Auditable Collaborative Anomaly Detection. *IEEE Access* **2022**, *10*, 82896–82909. [CrossRef]
24. Hassan, M.U.; Rehmani, M.H.; Chen, J. Anomaly detection in blockchain networks: A comprehensive survey. *IEEE Commun. Surv. Tutor.* **2022**, *25*, 289–318. [CrossRef]
25. Lim, W.Y.B.; Luong, N.C.; Hoang, D.T.; Jiao, Y.; Liang, Y.C.; Yang, Q.; Niyato, D.; Miao, C. Federated Learning in Mobile Edge Networks: A Comprehensive Survey. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 2031–2063. [CrossRef]
26. Nguyen, D.C.; Ding, M.; Pathirana, P.N.; Seneviratne, A.; Li, J.; Vincent Poor, H. Federated Learning for Internet of Things: A Comprehensive Survey. *IEEE Commun. Surv. Tutor.* **2021**, *23*, 1622–1658. [CrossRef]
27. Abdulrahman, S.; Tout, H.; Ould-Slimane, H.; Mourad, A.; Talhi, C.; Guizani, M. A Survey on Federated Learning: The Journey From Centralized to Distributed On-Site Learning and Beyond. *IEEE Internet Things J.* **2021**, *8*, 5476–5497. [CrossRef]
28. Aledhari, M.; Razzak, R.; Parizi, R.M.; Saeed, F. Federated Learning: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Access* **2020**, *8*, 140699–140725. [CrossRef]
29. Khan, L.U.; Saad, W.; Han, Z.; Hossain, E.; Hong, C.S. Federated Learning for Internet of Things: Recent Advances, Taxonomy, and Open Challenges. *IEEE Commun. Surv. Tutor.* **2021**, *23*, 1759–1799. [CrossRef]
30. Li, Q.; Wen, Z.; Wu, Z.; Hu, S.; Wang, N.; Li, Y.; Liu, X.; He, B. A Survey on Federated Learning Systems: Vision, Hype and Reality for Data Privacy and Protection. *IEEE Trans. Knowl. Data Eng.* **2023**, *35*, 3347–3366. [CrossRef]
31. Wahab, O.A.; Mourad, A.; Otrok, H.; Taleb, T. Federated Machine Learning: Survey, Multi-Level Classification, Desirable Criteria and Future Directions in Communication and Networking Systems. *IEEE Commun. Surv. Tutor.* **2021**, *23*, 1342–1397. [CrossRef]
32. Tan, A.Z.; Yu, H.; Cui, L.; Yang, Q. Towards Personalized Federated Learning. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, *34*, 9587–9603. [CrossRef]
33. Imteaj, A.; Thakker, U.; Wang, S.; Li, J.; Amini, M.H. A Survey on Federated Learning for Resource-Constrained IoT Devices. *IEEE Internet Things J.* **2022**, *9*, 1–24. [CrossRef]
34. Du, Z.; Wu, C.; Yoshinaga, T.; Yau, K.L.A.; Ji, Y.; Li, J. Federated Learning for Vehicular Internet of Things: Recent Advances and Open Issues. *IEEE Open J. Comput. Soc.* **2020**, *1*, 45–61. [CrossRef]
35. Kulkarni, V.; Kulkarni, M.; Pant, A. Survey of Personalization Techniques for Federated Learning. In Proceedings of the 2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4), London, UK, 27–28 July 2020; pp. 794–797.
36. Ghimire, B.; Rawat, D.B. Recent Advances on Federated Learning for Cybersecurity and Cybersecurity for Federated Learning for Internet of Things. *IEEE Internet Things J.* **2022**, *9*, 8229–8249. [CrossRef]
37. Yin, F.; Lin, Z.; Kong, Q.; Xu, Y.; Li, D.; Theodoridis, S.; Cui, S.R. FedLoc: Federated Learning Framework for Data-Driven Cooperative Localization and Location Data Processing. *IEEE Open J. Signal Process.* **2020**, *1*, 187–215. [CrossRef]
38. Alazab, M.; RM, S.P.; M, P.; Maddikunta, P.K.R.; Gadekallu, T.R.; Pham, Q.V. Federated Learning for Cybersecurity: Concepts, Challenges, and Future Directions. *IEEE Trans. Ind. Inform.* **2022**, *18*, 3501–3509. [CrossRef]
39. Bouacida, N.; Mohapatra, P. Vulnerabilities in Federated Learning. *IEEE Access* **2021**, *9*, 63229–63249. [CrossRef]
40. Lyu, L.; Yu, H.; Ma, X.; Chen, C.; Sun, L.; Zhao, J.; Yang, Q.; Yu, P.S. Privacy and Robustness in Federated Learning: Attacks and Defenses. In *IEEE Transactions on Neural Networks and Learning Systems*; IEEE: New York, NY, USA, 2022; pp. 1–21.
41. Ouadrhiri, A.E.; Abdelhadi, A. Differential Privacy for Deep and Federated Learning: A Survey. *IEEE Access* **2022**, *10*, 22359–22380. [CrossRef]
42. Alsamhi, S.H.; Almalki, F.A.; Afghah, F.; Hawbani, A.; Shvetsov, A.V.; Lee, B.; Song, H. Drones' Edge Intelligence Over Smart Environments in B5G: Blockchain and Federated Learning Synergy. *IEEE Trans. Green Commun. Netw.* **2022**, *6*, 295–312. [CrossRef]
43. Arisdakessian, S.; Wahab, O.A.; Mourad, A.; Otrok, H.; Guizani, M. A Survey on IoT Intrusion Detection: Federated Learning, Game Theory, Social Psychology, and Explainable AI as Future Directions. *IEEE Internet Things J.* **2023**, *10*, 4059–4092. [CrossRef]
44. Rahman, K.M.J.; Ahmed, F.; Akhter, N.; Hasan, M.; Amin, R.; Aziz, K.E.; Islam, A.K.M.M.; Mukta, M.S.H.; Islam, A.K.M.N. Challenges, Applications and Design Aspects of Federated Learning: A Survey. *IEEE Access* **2021**, *9*, 124682–124700. [CrossRef]
45. Ali, M.; Naeem, F.; Tariq, M.; Kaddoum, G. Federated Learning for Privacy Preservation in Smart Healthcare Systems: A Comprehensive Survey. *IEEE J. Biomed. Health Inform.* **2023**, *27*, 778–789. [CrossRef]
46. Saraswat, D.; Verma, A.; Bhattacharya, P.; Tanwar, S.; Sharma, G.; Bokoro, P.N.; Sharma, R. Blockchain-Based Federated Learning in UAVs Beyond 5G Networks: A Solution Taxonomy and Future Directions. *IEEE Access* **2022**, *10*, 33154–33182. [CrossRef]
47. Kar, B.; Yahya, W.; Lin, Y.D.; Ali, A. Offloading Using Traditional Optimization and Machine Learning in Federated Cloud–Edge–Fog Systems: A Survey. *IEEE Commun. Surv. Tutor.* **2023**, *25*, 1199–1226. [CrossRef]
48. Liu, Z.; Guo, J.; Yang, W.; Fan, J.; Lam, K.Y.; Zhao, J. Privacy-Preserving Aggregation in Federated Learning: A Survey. *IEEE Trans. Big Data* **2022**, 1–20. [CrossRef]

49. Zhu, C.; Zhu, X.; Ren, J.; Qin, T. Blockchain-Enabled Federated Learning for UAV Edge Computing Network: Issues and Solutions. *IEEE Access* **2022**, *10*, 56591–56610. [CrossRef]
50. Li, L.; Fan, Y.; Lin, K.Y. A Survey on federated learning. In Proceedings of the 2020 IEEE 16th International Conference on Control & Automation (ICCA), Singapore, 9–11 Octobers 2020; pp. 791–796.
51. Patel, V.A.; Bhattacharya, P.; Tanwar, S.; Gupta, R.; Sharma, G.; Bokoro, P.N.; Sharma, R. Adoption of Federated Learning for Healthcare Informatics: Emerging Applications and Future Directions. *IEEE Access* **2022**, *10*, 90792–90826. [CrossRef]
52. Kumar, K.N.; Mohan, C.K.; Cenkeramaddi, L.R. The Impact of Adversarial Attacks on Federated Learning: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2024**, *46*, 2672–2691. [CrossRef]
53. Staňo, M.; Hluchỳ, L.; Bobák, M.; Krammer, P.; Tran, V. Federated learning methods for analytics of big and sensitive distributed data and survey. In Proceedings of the 2023 IEEE 17th International Symposium on Applied Computational Intelligence and Informatics (SACI), Timisoara, Romania, 23–26 May 2023; pp. 000705–000710.
54. Qu, Y.; Uddin, M.P.; Gan, C.; Xiang, Y.; Gao, L.; Yearwood, J. Blockchain-enabled federated learning: A survey. *ACM Comput. Surv.* **2022**, *55*, 1–35. [CrossRef]
55. Bhatia, L.; Samet, S. Decentralized federated learning: A comprehensive survey and a new blockchain-based data evaluation scheme. In Proceedings of the 2022 Fourth International Conference on Blockchain Computing and Applications (BCCA), San Antonio, TX, USA, 5–7 September 2022; pp. 289–296.
56. Neto, H.N.C.; Hribar, J.; Dusparic, I.; Mattos, D.M.F.; Fernandes, N.C. A Survey on Securing Federated Learning: Analysis of Applications, Attacks, Challenges, and Trends. *IEEE Access* **2023**, *11*, 41928–41953. [CrossRef]
57. Wang, J.; Liu, W.; Kou, Y.; Xiao, D.; Wang, X.; Tang, X. Approx-SMOTE Federated Learning Credit Card Fraud Detection System. In Proceedings of the 2023 IEEE 47th Annual Computers, Software, and Applications Conference (COMPSAC), Torino, Italy, 26–30 June 2023; pp. 1370–1375.
58. Zheng, W.; Yan, L.; Gou, C.; Wang, F.Y. Federated meta-learning for fraudulent credit card detection. In Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence, Yokohama, Japan, 11–17 July 2021; pp. 4654–4660.
59. Abdul Salam, M.; Fouad, K.M.; Elbably, D.L.; Elsayed, S.M. Federated learning model for credit card fraud detection with data balancing techniques. *Neural Comput. Appl.* **2024**, 1–26. [CrossRef]
60. Yang, W.; Zhang, Y.; Ye, K.; Li, L.; Xu, C.Z. Ffd: A federated learning based method for credit card fraud detection. In *Proceedings of the Big Data–BigData 2019: 8th International Congress, Held as Part of the Services Conference Federation, SCF 2019, San Diego, CA, USA, 25–30 June 2019*; Proceedings 8; Springer: Berlin/Heidelberg, Germany, 2019; pp. 18–32.
61. Bian, K.; Zheng, H. FedAvg-DWA: A Novel Algorithm for Enhanced Fraud Detection in Federated Learning Environment. In Proceedings of the 2023 4th International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE), Shenzhen, China, 25–27 October 2023; pp. 13–17.
62. Bahnsen, A.C.; Stojanovic, A.; Aouada, D.; Ottersten, B. Improving credit card fraud detection with calibrated probabilities. In Proceedings of the 2014 SIAM International Conference on Data Mining, SIAM, Philadelphia, PA, USA, 24–26 April 2014; pp. 677–685.
63. Lebichot, B.; Braun, F.; Caelen, O.; Saerens, M. A graph-based, semi-supervised, credit card fraud detection system. In *International Workshop on Complex Networks and their Applications*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 721–733.
64. Ki, Y.; Yoon, J.W. Pd-fds: Purchase density based online credit card fraud detection system. In Proceedings of the KDD 2017 Workshop on aNomaly Detection in Finance, PMLR, Halifax, NS, Canada, 14 August 2018; pp. 76–84.
65. Braun, F.; Caelen, O.; Smirnov, E.N.; Kelk, S.; Lebichot, B. Improving card fraud detection through suspicious pattern discovery. In *Proceedings of the Advances in Artificial Intelligence: From Theory to Practice: 30th International Conference on Industrial Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2017, Arras, France, 27–30 June 2017*; Proceedings, Part II 30; Springer: Berlin/Heidelberg, Germany, 2017; pp. 181–190.
66. Soemers, D.; Brys, T.; Driessens, K.; Winands, M.; Nowé, A. Adapting to concept drift in credit card transaction data streams using contextual bandits and decision trees. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32.
67. Jiang, C.; Song, J.; Liu, G.; Zheng, L.; Luan, W. Credit card fraud detection: A novel approach using aggregation strategy and feedback mechanism. *IEEE Internet Things J.* **2018**, *5*, 3637–3647. [CrossRef]
68. Dal Pozzolo, A. Adaptive Machine Learning for Credit Card Fraud Detection. Ph.D. Thesis, Université Libre de Bruxelles, Brussels, Belgium, 2015.
69. Li, X.; Liu, S.; Li, Z.; Han, X.; Shi, C.; Hooi, B.; Huang, H.; Cheng, X. Flowscope: Spotting money laundering based on graphs. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 4731–4738.
70. Baabdullah, T.; Alzahrani, A.; Rawat, D.B. On the Comparative Study of Prediction Accuracy for Credit Card Fraud Detection with Imbalanced Classifications. In Proceedings of the 2020 Spring Simulation Conference (SpringSim), Fairfax, VA, USA, 18–21 May 2020; IEEE: New York, NY, USA, 2020; pp. 1–12.
71. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
72. Jin, R.; Xing, Y.; He, X. On the convergence of mSGD and AdaGrad for stochastic optimization. *arXiv* **2022**, arXiv:2201.11204.

73. Danner, G.; Jelasity, M. Fully distributed privacy preserving mini-batch gradient descent learning. In *Proceedings of the Distributed Applications and Interoperable Systems: 15th IFIP WG 6.1 International Conference, DAIS 2015, Held as Part of the 10th International Federated Conference on Distributed Computing Techniques, DisCoTec 2015, Grenoble, France, 2–4 June 2015*; Proceedings 15; Springer: Berlin/Heidelberg, Germany, 2015; pp. 30–44.
74. Baabdullah, T.; Rawat, D.B.; Liu, C.; Alzahrani, A.; Almotairi, A. Analysis of Cardholder Spending Behavior and Transaction Authentication to Enhance Credit Card Fraud Detection. In Proceedings of the 2023 International Conference on Machine Learning and Applications (ICMLA), Jacksonville, FL, USA, 15–17 December 2023; pp. 1144–1149.