


Enhanced Credit Card Fraud Detection Using Federated Learning, LSTM Models, and the SMOTE Technique

Weddou Mohamedhen¹, Maha Charfeddine²^a and Yessine Hadj Kacem¹^b

¹*CES Laboratory, National Engineering School of Sfax, University of Sfax, Sfax, Tunisia*

²*Research Groups in Intelligent Machines, National Engineering School of Sfax (ENIS), Sfax, Tunisia*
{weddou.mohamedhen, yessine.hadjkacem}@enis.tn, maha.charfeddine.tn@ieee.org

Keywords: Credit Card Fraud, Federated Learning, Class Imbalance, SMOTE, LSTM, CNN, ADAM, Data Privacy.

Abstract: In recent years, credit card transaction fraud has caused significant financial losses for both consumers and financial institutions. To effectively combat these losses, the development of a sophisticated fraud detection system is necessary. However, credit card fraud detection (CCFD) presents significant challenges, particularly in regards to data security and privacy, limiting financial institutions' ability to share transaction data for model training. This paper introduces the use of Federated Learning for CCFD, a technique that allows for decentralized learning while protecting data privacy. Federated Learning enables multiple institutions to collaborate on model training without having to share sensitive data, effectively addressing privacy concerns. To address the problem of class imbalance in fraud detection datasets, we apply the Synthetic Minority Oversampling Technique (SMOTE) to ensure a balanced dataset. Our study compares Long Short-Term Memory (LSTM) networks to Convolutional Neural Networks (CNN) within a Federated Learning framework. The experimental results demonstrate that combining SMOTE and LSTM in a Federated Learning setup produces superior performance. These findings highlight the strength of LSTM models in processing sequential transaction data and reveal that Federated Learning, when paired with resampling techniques, strengthens fraud detection accuracy.

1 INTRODUCTION

The prevalence of credit card transactions has surged due to the expansion of e-commerce, electronic banking, and mobile payments, leading to a significant increase in credit card fraud. This rise has resulted in global fraud losses reaching \$33 billion in 2022, projected to reach \$43 billion by 2026 (Report, 2023), (Payments, 2023), (Moneyzine, 2023).

Fraudulent transactions are often carried out through stolen or falsified credit card information, making fraud detection a critical issue. Despite the success of Machine Learning (ML) models, challenges such as dataset insufficiency and class imbalance persist.


Dataset Insufficiency. Privacy concerns limit the availability of public datasets, hindering the development of robust fraud detection systems. To address this, we use Federated Learning (FL), which enables collaborative model development across institutions


while preserving privacy (Salam et al., 2023), (Zhang and et al., 2021).

Skewed Class Distribution. Fraud detection datasets are typically imbalanced, with fraudulent transactions being rare. To overcome this, we apply the Synthetic Minority Oversampling Technique (SMOTE) to balance the dataset (G. Bejjanki and Narsimha, 2018).

This paper proposes a novel approach combining Federated Learning and SMOTE to tackle dataset insufficiency and class imbalance. We optimize key FL parameters, such as the fraction of participating institutions (F) and the number of local epochs (E), and compare LSTM and CNN models. Our results show that LSTM outperforms CNN in fraud detection. This work aims to improve the accuracy and reliability of fraud detection systems in financial institutions.

The paper is structured as follows: Section 2 reviews existing approaches, Section 3 presents our methodology, Section 4 outlines results, and Section 5 concludes with future work.

^a <https://orcid.org/0000-0003-2996-4113>

^b <https://orcid.org/0000-0002-5757-6516>

2 RELATED WORK

Fraud detection algorithms in credit card transactions employ Machine Learning techniques to effectively identify fraudulent activities. Traditional approaches predominantly use centralized learning models such as Decision Trees (DT), Random Forests (RF), Logistic Regression (LR), Support Vector Machines (SVM), Extreme Gradient Boosting (XGB), and unsupervised methods like Generative Adversarial Networks (GAN), Auto-Encoders (AE), Restricted Boltzmann Machines (RBM), and One-Class SVM (OCSVM) (G. Bejjanki and Narsimha, 2018), (X. Niu and Yang, 2019). While effective, these methods face scalability and data privacy challenges due to their centralized nature. For example, a study using the UCI Credit Card Dataset, comprising 284,807 transactions, highlighted the imbalanced nature of the data with only 492 fraudulent transactions, impacting model accuracy.

Federated Learning (FL) has emerged as a promising alternative, addressing concerns over data security and privacy inherent in centralized models (M. Fahmi and Nagati, 2016), (K. Chen and Zhang, 2019). FL enables collaborative model training on distributed data sources without sharing sensitive information, thereby enhancing privacy while improving model performance. Recent studies, such as one involving a federated dataset of over 1 million transactions from multiple banks, have demonstrated FL's effectiveness in real-time credit card fraud detection, showing significant improvements compared to traditional centralized models (K. Chen and Zhang, 2019).

In Deep Learning, unsupervised models like AE and RBM have shown high accuracy rates (88% to 94%) in detecting credit card fraud when integrated into Federated Learning frameworks (al., 2019), (Suvana and Kowshalya, 2020). For instance, using the IEEE-CIS Fraud Detection dataset, with over 590,000 records, these models required careful parameter tuning and computational resources but offered robust performance in identifying complex fraud patterns. Privacy-preserving strategies such as combining FL with differential privacy and homomorphic encryption further bolster the security of fraud detection systems (Albertio, 2019). (al., 2020).

Hybrid techniques integrating Decision Trees, clustering algorithms, pairwise matching, Neural Networks, and genetic algorithms are also being explored to predict fraud in various transactional datasets citeb15, (Dornadula and Geetha, 2019). For example, a hybrid approach on the European cardholders dataset, consisting of 284,807 records, leveraged local and global model characteristics to optimize perfor-

mance while minimizing communication overhead.

To address class imbalance challenges in fraud detection datasets, various methods including cost-sensitive Deep Learning approaches and resampling techniques like SMOTE, EUS-Bag, and PSOAANN have been developed (Kamaruddin and Ravi, 2016). These techniques aim to balance dataset distributions and enhance model robustness against rare fraud cases. A study using the Kaggle Credit Card Fraud Detection dataset, which includes 284,807 transactions, found that applying SMOTE improved the detection rate of fraudulent transactions significantly, though implementation requires careful consideration to avoid biases and maintain computational efficiency.

Recent works have further advanced fraud detection in the FL framework. Salam et al. (2023) proposed a Federated Learning model for credit card fraud detection, incorporating data balancing techniques to address privacy and class imbalance. Similarly, Li and Walsh (2024) introduced *FedGAT-DCNN*, combining Graph Attention Networks and dilated convolutions in FL to enhance fraud detection. Our work builds upon these studies by integrating SMOTE directly into FL and optimizing key FL parameters, such as the fraction of participating institutions (F) and the number of local epochs (E), to enhance model performance and scalability.

Compared to existing approaches, our work offers several significant value additions:

- **Privacy Protection Through Federated Learning.** Unlike traditional approaches that require sharing raw data, our method allows secure collaboration among financial institutions, addressing data privacy concerns.
- **Enhanced Fraud Detection with SMOTE.** By using SMOTE to balance the data, our approach overcomes the challenge of class imbalance, thereby improving the performance of predictive models.
- **Comparative Analysis of LSTM and CNN Models.** Our study provides a detailed comparative analysis of LSTM and CNN networks within a Federated Learning framework, offering valuable insights into the effectiveness of these models for credit card fraud detection.
- **Practical Applicability for Financial Institutions.** Experimental results demonstrate that the combination of SMOTE and LSTM within the Federated Learning system yields the best results, highlighting the superiority of the LSTM model in handling sequential transaction data.

In summary, the combination of centralized and Federated Learning models, along with advanced

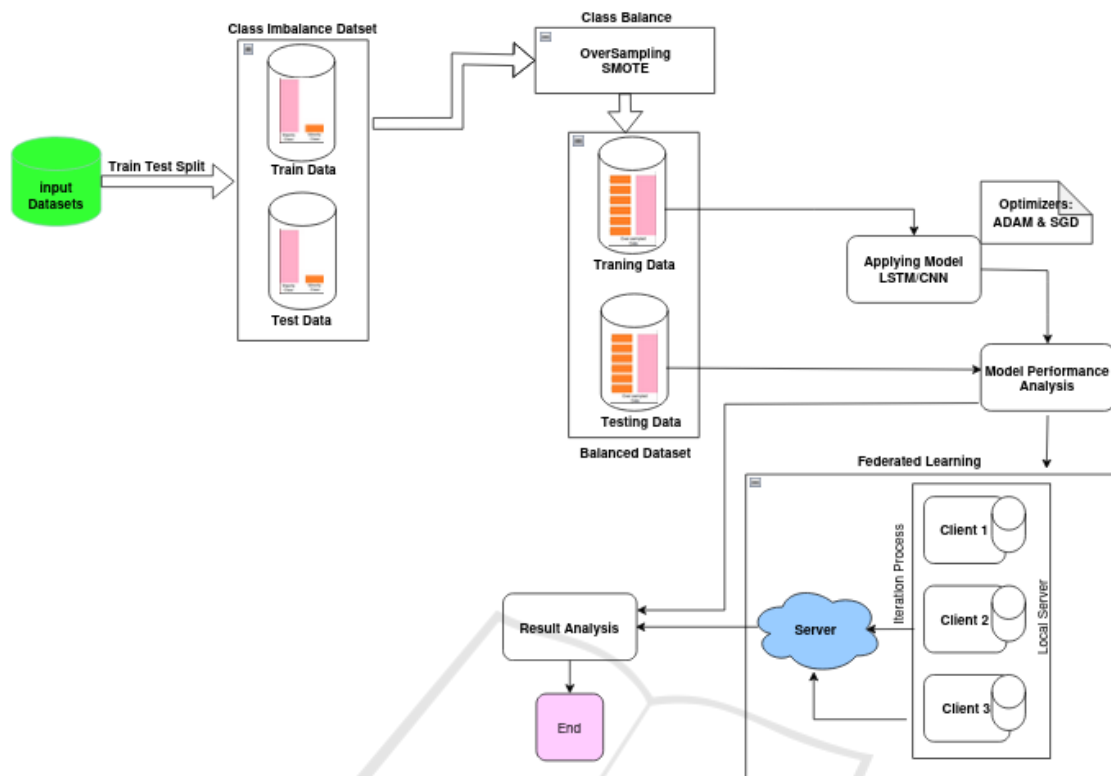


Figure 1: Federated Learning architecture-based intelligent Credit Card Fraud Detection (CCFD) system.

Deep Learning techniques, privacy-preserving strategies, and effective class imbalance management, holds great promise for improving the reliability and efficiency of credit card fraud detection systems. Further research should look into these methodologies, particularly their adaptability to changing fraud patterns, in order to improve system security and performance.

3 METHODOLOGY

Figure 1 depicts our proposed Intelligent Credit Card Fraud Detection (CCFD) system built on a secure Federated Learning architecture. The system begins with input transaction and label data, followed by preprocessing steps like cleaning, normalization, and feature engineering (Ali et al., 2024a). Through our Federated Learning framework, local models train on distributed datasets without sharing raw data, and an aggregation server combines them into a global model. The model is evaluated using metrics like precision, recall, accuracy, and F1-score, and predicts the likelihood of new transactions being fraudulent or legitimate. The following sub-sections provide details about our suggested architecture:

3.1 Dataset Description

The dataset used in this research was obtained from Kaggle (Group—ULB, 2018). It consists of 284,807 anonymous credit card transactions from European cardholders in September 2013, of which 492 are fraudulent, resulting in a heavily skewed dataset. The dataset includes 30 features that describe each transaction, including transaction amount and time. Clients are fully aware of all features, ensuring effective alignment of their data for federated learning. There are no missing data points, maintaining the integrity of the analysis.

Table 1: Overview of the Dataset Obtained from (Group—ULB, 2018).

	Total Dataset	#Fraud	#Not Fraud	Label Not F	Label F
Dataset	284,807	492	284,315	0	1

3.2 Resampling Technique

Synthetic minority oversampling technique (SMOTE) SMOTE is an effective oversampling method to address class imbalance by generating synthetic examples of the minority class rather than duplicating existing ones. SMOTE selects similar instances from

the minority class and interpolates between them to create new, synthetic data points, resulting in a more balanced dataset and improved model performance as depicted in Figure 2. Several studies have demonstrated the efficacy of SMOTE in improving classifier performance in imbalanced datasets, concerning cyberattack detection challenges (Ali et al., 2024b). Thus, to handle class imbalance, SMOTE is used to create synthetic fraudulent examples, applied only to the training set (XTrain).

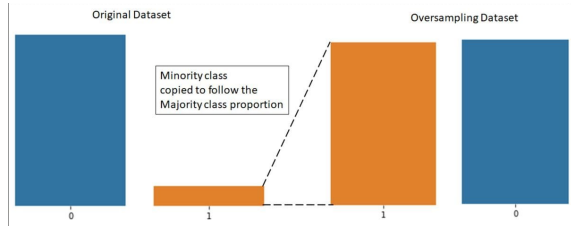


Figure 2: Visualization of the Synthetic Minority Over-sampling Technique (SMOTE) applied to balance the dataset.

To reduce the content while maintaining the key information, here's a more concise version of the section. This version retains the core explanations about LSTM, CNN, and optimizers, focusing on the essentials to fit within an 8-page limit:

3.3 Modeling

We developed two Deep Learning (DL) models, Long Short-Term Memory (LSTM) and Convolutional Neural Networks (CNN), to detect credit card fraud. While Federated Learning ensures data privacy, it does not fully address the need for advanced models. Integrating LSTM and CNN enhances the ability to handle complex patterns in the data.

Long Short-Term Memory Networks (LSTMs). (Li and Zhao, 2021) are designed to capture temporal dependencies in sequential data, enabling the detection of fraud by modeling transaction sequences over time.

Convolutional Neural Networks (CNNs). (Zhang and Liu, 2022) are used here to capture local feature patterns in transaction data, even though CNNs are typically used for grid-like data. While GRU or RNNs may be better suited for sequential data, CNNs are included to explore their potential in fraud detection.

Optimizers play a key role in model performance: **ADAM (Adaptive Moment Estimation).** (V. Felbab and Horváth, 2021) adapts learning rates and speeds up convergence, even with noisy data.

SGD (Stochastic Gradient Descent). (Akbar and Chowanda, 2022) updates parameters using small

batches, helping to avoid local minima and train efficiently on large datasets.

By combining LSTM and CNN models within Federated Learning and using advanced optimizers, we improve fraud detection accuracy and reliability.

3.4 Federated Fraud Detection Framework

In our Federated Learning-based fraud detection system, C banks each hold a private dataset, and SMOTE is used to address class imbalance by generating synthetic minority class examples. The banks collaborate to build a fraud detection model while maintaining privacy, agreeing on a common model architecture, including activation and loss functions, before training begins.

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^n L_c(x_i, y_i; w) \quad (1)$$

where

$$L_c(x, y; w) = \frac{1}{|D_c|} \sum_{i \in D_c} L_c(x_{ci}, y_{ic}; w) \quad (2)$$

The server initializes the fraud detection model parameters. During each communication round, a random fraction of banks is selected to participate. These banks download the current global model, compute gradients on their private datasets, update their local models, and send the updates back to the server. The server then aggregates the updates to improve the global model.

$$w_{t+1} \leftarrow w_t - \eta \sum_{c=1}^C \frac{1}{n} \sum_{i \in D_c} \nabla L_c(x_{ci}, y_{ic}; w) \quad (3)$$

Considering the impact of skewed data on model performance, we use the combination of data size and detection model performance α_{t+1} on each bank as the weight of the parameter vector. It can be written as:

$$w_{t+1} \leftarrow w_t - \sum_{c=1}^C \frac{n_c}{\sum_{c=1}^C n_c} \alpha_{t+1} f_c \quad (4)$$

In Federated Learning (FL), each bank performs a gradient descent step using its own data, and the server aggregates these models via a weighted average, updating the global model over T iterations. FL addresses privacy concerns by allowing collaborative training on distributed data while maintaining confidentiality. However, FL faces challenges like communication costs, which depend on factors such as the fraction of participating banks, minibatch size, and the number of local epochs. These parameters must be optimized to balance parallelism and efficiency. The training process is shown in Algorithm 1 and Figure 3.

Algorithm 1. FFD Framework.

Require: The private dataset of banks and financial institutions

Ensure: A credit card fraud detection model with Federated Learning

- 1: **Server Update:**
- 2: Initialize the detection classifier and its parameters w_0
- 3: **for** each round $t = 1, 2, \dots, T$ **do**
- 4: Randomly choose $\max(F \cdot C, 1)$ banks as N_t
- 5: **for** each bank $c \in N_t$ **in parallel do**
- 6: $w_{t+1}, \alpha_{t+1} \leftarrow \text{BankUpdate}(n, w_t)$
- 7: **end for**
- 8: $w_{t+1} \leftarrow \sum_{i=1}^T \frac{C_i}{C} \alpha_t + 1$
- 9: **end for**
- 10:
- 11: **BankUpdate**(n, w):
- 12: **Data Processing:** Rebalance raw dataset with SMOTE (applied only to the training set) and split it into two parts: 80% training data and 20% testing data
- 13: **Training:**
- 14: $B \leftarrow \text{Split } D_n$ into batches of size B
- 15: **for** each local epoch i from 1 to E **do**
- 16: **for** each batch $b \in B$ **do**
- 17: $w \leftarrow w - \eta \nabla L(x, y; w)$
- 18: **end for**
- 19: **end for**
- 20: **Testing:**
- 21: Return w and validation accuracy α to server

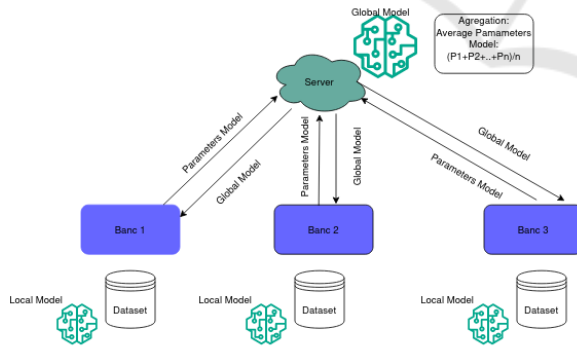


Figure 3: Federated Learning model for FDS.

4 EXPERIMENTAL RESULTS

This section compares the performance of LSTM and CNN models in a secure Federated Learning framework enhanced with SMOTE for credit card fraud detection. We evaluated models based on precision, recall, F1-score, and accuracy, with hyperparameters including a 30-input size, 128 hidden units across three

layers, and a learning rate of 0.001. ADAM and SGD optimizers were tested over 20 epochs with varying batch sizes. The setup involved 12 clients, with 10 clients per round over 100 communication rounds using TensorFlow Federated. The results are compared to previous related works.

4.1 Performance Metrics

To evaluate credit card fraud detection systems, we use several metrics beyond accuracy, particularly in imbalanced datasets. While accuracy measures overall correctness, it can be misleading if fraudulent transactions are misclassified. Therefore, we consider precision, which reflects the system's reliability in identifying fraud, recall, which measures its efficiency in detecting all fraudulent transactions, and the F1 score, which is the harmonic mean of precision and recall. These metrics provide a more comprehensive evaluation of the system's performance (Javatpoint, 2024).

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \quad (5)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (6)$$

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (7)$$

$$\text{Precision} = \frac{TP}{TP + FP} \quad (8)$$

4.2 Results and Discussions

In this section, performance metrics, including precision, recall, accuracy, and F1-score, have been discussed to ensure the effectiveness of all classifiers used in conjunction with the chosen resampling technique, SMOTE. For model evaluation, the LSTM and CNN algorithms were adopted for comparison. The experiments were conducted using an 80:20 training-to-testing ratio, which showed that LSTM outperforms CNN in federated learning for fraud detection due to its ability to capture temporal dependencies, whereas CNN is better suited for detecting local patterns, which are less relevant for the sequential nature of fraud data.

The Figure 4' diagrams depict the performance of the developed LSTM model with federated data learning (SMOTE+ LSTM+ FDL), employing two optimization techniques, notably ADAM and SGD, across different batch size configurations and client counts (numclients). Evaluation metrics include accuracy, precision, recall, and F1 score. The results

illustrate how these metrics vary based on model configuration parameters and the platform used. Each experimental configuration was conducted with 12 clients and 10 selected for training, ensuring a fair and representative comparison of performance. We observe significant advancements in model performance across these various metrics, highlighting the efficacy of tailored approaches like (SMOTE + LSTM + FDL) in achieving high accuracy, precision, recall, and overall balanced performance. The findings show ADAM's accuracy consistently of 0.999 value, while SGD slightly around 0.998. Precision shows significant variability, reaching up to 0.887 with ADAM and 0.646 with SGD, depending on the batch size. Recall remains steady, ranging from 0.907 to 0.926, and F1 Score fluctuates with values between 0.842 and 0.879. Regarding batch sizes, 256 seems to provide an optimal balance between precision and recall for the (SMOTE + LSTM + FDL) model, especially when using 100 iterations with the Adam optimizer and a learning rate of 0.001. These findings underscore the importance of optimizer selection in developing robust Deep Learning-based fraud detection systems, directly influencing model precision and stability of predictions. The analysis also highlights batch size's crucial role in model performance, particularly. In conclusion, our study advocates for ADAM as the preferred optimizer to maximize efficiency and reliability of the combination (SMOTE + LSTM + FDL)-based fraud detection systems.

Moreover, Figure 5 illustrates the performance metrics of the developed CNN model using the two different optimizers, SGD and ADAM, across varying batch sizes (32, 64, 128, and 256) and consistent parameters of num clients = 12 and num selected = 10. The results reveal that Accuracy remains consistently high and stable, ranging from 0.998 to 0.999 across all batch sizes and optimizers. Precision exhibits noticeable variability, ranging from 0.470 to 0.880 for SGD and 0.830 to 0.887 for ADAM, reflecting the impact of optimizer choice on model precision. Recall shows relatively stable performance, hovering between 0.870 and 0.926, with slight fluctuations across different batch sizes and optimizers. Similarly, the F1 Score varies, demonstrating values from 0.635 to 0.879, emphasizing the trade-offs between precision and recall.

When comparing Figures 4 and 5, the results show that the (SMOTE + LSTM + FDL) model offers excellent overall performance, surpassing the results of the (SMOTE+ CNN + FDL) model, in terms of recall and F1 score. This makes it a recommended choice for achieving robust and balanced performance.

Table 2: Performance Metrics for LSTM with Adam Optimizer Across Different Batch Sizes.

Batch Size	Accuracy (ACC)	Precision	Recall	F1 Score
32	0.999	0.810	0.907	0.842
64	0.999	0.887	0.870	0.879
128	0.999	0.825	0.889	0.885
256	0.999	0.887	0.889	0.879

Table 3: Performance Metrics for CNN with Adam Optimizer Across Different Batch Sizes.

Batch Size	Accuracy	Precision	Recall	F1 Score
32	0.999	0.830	0.889	0.844
64	0.998	0.470	0.926	0.610
128	0.998	0.500	0.870	0.635
256	0.999	0.880	0.815	0.846

The comparison between LSTM and CNN models with Adam optimizer and SMOTE technique reveals distinct performance characteristics across various batch sizes. The LSTM model consistently achieves high accuracy, maintaining values at 0.999 across different batch sizes as observed in Table I. For precision, LSTM ranges from 0.810 to 0.887, indicating reliable classification of positive cases. The recall values for LSTM range from 0.870 to 0.907, showing consistent ability to identify true positive cases. Consequently, the F1 scores for LSTM vary between 0.842 and 0.885, reflecting a balanced performance in terms of both precision and recall with Adam optimizer.

In contrast, as displayed in Table II, the CNN model shows variability in performance metrics across batch sizes with Adam optimizer. While CNN achieves high accuracy, ranging from 0.998 to 0.999, its precision varies significantly, from 0.470 to 0.880. This variability suggests different levels of effectiveness in correctly identifying positive cases. CNN also shows fluctuating recall values, ranging from 0.815 to 0.926, indicating varying success in capturing true positive cases across different batch sizes. Correspondingly, the F1 scores for CNN range widely from 0.610 to 0.846, highlighting its sensitivity to batch size variations and the consequent impact on overall model performance with Adam optimizer.

Overall, these results underscore the LSTM model's stability and balanced performance with Adam optimizer and SMOTE, whereas the CNN model's effectiveness appears to be more sensitive to batch size adjustments, potentially influencing its precision and recall outcomes significantly.

In addition, the comparison between LSTM and CNN models using the SGD optimizer shows distinct performance profiles across batch sizes. As seen in Table IV, LSTM achieves high accuracy (0.998–0.999) with variable precision (0.520–0.646) and stable recall (0.889–0.926), resulting in F1

Table 4: Performance Metrics for LSTM with SGD Optimizer Across Different Batch Sizes.

Batch Size	Accuracy	Precision	Recall	F1 Score
32	0.998	0.530	0.926	0.736
64	0.999	0.610	0.907	0.826
128	0.999	0.520	0.889	0.644
256	0.999	0.646	0.926	0.762

Table 5: Performance Metrics for CNN with SGD Optimizer Across Different Batch Sizes.

Batch Size	Accuracy	Precision	Recall	F1 Score
32	0.998	0.495	0.907	0.626
64	0.999	0.610	0.907	0.718
128	0.999	0.783	0.889	0.825
256	0.998	0.527	0.907	0.644

scores between 0.644 and 0.826. In contrast, CNN achieves similar accuracy (0.998–0.999) but exhibits more variability in precision (0.495–0.783) and recall (0.889–0.907), with F1 scores ranging from 0.626 to 0.825, as shown in Table V. While LSTM shows consistent performance, CNN’s variability highlights the need for more tuning based on the application.

Table 6: Comparison between Previous Work and Our Best System (Smote+LSTM+ADAM+FDL).

Model	Accuracy	Precision	Recall	F1 Score
GRU (Forough and Momtazi, 2022)	0.997	0.8626	0.7208	0.7792
LSTM (Forough and Momtazi, 2022)	0.997	0.8575	0.7408	0.7866
Ensemble model (Forough and Momtazi, 2022)	0.998	0.9569	0.6674	0.7813
Smote+CNN (H. Ghafoor and Khan, 2022)	0.998	0.8263	0.8095	0.8178
Smote+LSTM+ADAM+FDL	0.999	0.887	0.889	0.879

4.3 Comparison with Previous Work

As exhibited in Table VI, our proposed (SMOTE+LSTM+ ADAM+ FDL) system achieves exceptional accuracy (0.999), precision (0.887), recall (0.889), and F1 Score (0.879), showcasing its effectiveness in credit card fraud detection. In contrast, previous works such as the GRU (Forough and Momtazi, 2022) and LSTM (Forough and Momtazi, 2022) models reported precisions of 0.8626 and 0.8575 respectively, with recall values of 0.7208 and 0.7408. The ensemble model (Forough and Momtazi, 2022) achieved a high precision of 0.9569 but lower recall (0.6674) and comparable F1 Score (0.7813). Moreover, the SMOTE + CNN approach (H. Ghafoor and Khan, 2022) demonstrated precision (0.8263), recall (0.8095), and F1 Score (0.8178), highlighting different trade-offs in performance compared to our method. These comparisons underscore the advancements and efficacy of our secure and intelligent proposed approach in addressing the challenges of credit card fraud detection.

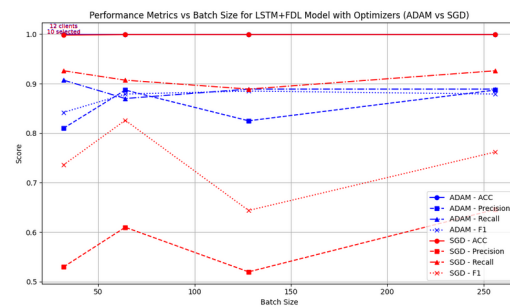


Figure 4: Performance Metrics vs Batch Size for (SMOTE+LSTM+FDL) Model with Optimizers (ADAM vs SGD).

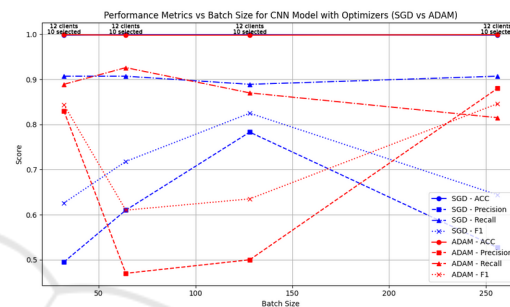


Figure 5: Performance Metrics vs Batch Size for (SMOTE+CNN+FDL) Model with Optimizers (ADAM vs SGD).

5 CONCLUSION

This study compared LSTM and CNN models for credit card fraud detection within a Federated Learning framework, using SMOTE to address class imbalance. LSTM outperformed CNN in precision, recall, and F1-score, achieving a precision of 0.887, recall of 0.889, and F1-score of 0.879, compared to CNN’s 0.880, 0.815, and 0.846. These results highlight LSTM’s effectiveness with sequential data. The combination of LSTM, Federated Learning, Adam optimizer, and SMOTE provided the best performance, suggesting it as an optimal approach for fraud detection systems. Future work should focus on improving Federated Learning protocols and exploring advanced models.

REFERENCES

A. Singh, R. R. and Tiwari, A. (2021). Credit card fraud detection under extreme imbalanced data: A comparative study of data-level algorithms. *J Exp Theor Artif Intell*, 34:1–28.

Akbar, M. and Chowanda, A. (2022). Sgd optimizer to reduce cost value in deep learning for customer churn

- prediction. *Journal of Theoretical and Applied Information Technology*, 100(11):8.
- al., W. L. (2020). Federated learning in mobile edge networks: A comprehensive survey. *IEEE Commun Surv Tutor*, 22(3).
- al., X. L. (2017). Can decentralized algorithms outperform centralized algorithms? a case study for decentralized parallel stochastic gradient descent. In *Adv Neural Inf Process Syst*, volume 30, page 6255.
- al., Y. W. (2019). Ffd: A federated learning-based method for credit card fraud detection. *J Big Data*, LNCS 11514:18–22.
- Albertio, C. (2019). Towards efficient and privacy-preserving federated deep learning. In *International Conference on Science and Technology on Communication Security Laboratory*. IEEE.
- Ali, A. H., Ammar, B., Charfeddine, M., and Hamed, B. B. (2024a). Enhanced intrusion detection based hybrid meta-heuristic feature selection.
- Ali, A. H., Charfeddine, M., Ammar, B., and Hamed, B. B. (2024b). Intrusion detection schemes based on synthetic minority oversampling technique and machine learning models. In *2024 IEEE 27th International Symposium on Real-Time Distributed Computing (ISORC)*, pages 1–8.
- Dornadula, V. and Geetha, S. (2019). Credit card fraud detection using machine learning algorithms. In *Procedia Comput Sci*, volume 165, pages 631–641.
- Forough, J. and Momtazi, S. (2022). Ensemble of deep sequential models and machine learning techniques for credit card fraud detection. *Expert Systems with Applications*, 197.
- G. Bejjanki, G. J. and Narsimha, G. (2018). *Class Processing and Systems*. Springer.
- Group—ULB, M. L. (2018). Credit card fraud detection anonymized credit card transactions labeled as fraudulent or genuine.
- H. Ghafoor, H. K. and Khan, M. (2022). Credit card fraud detection with machine learning algorithms: A systematic literature review. *Sustainability*, 11.
- Javatpoint (2024). Performance metrics in machine learning.
- K. Chen, S. S. and Zhang, L. (2019). Big data—bigdata 2019: 8th international congress, held as part of the services conference federation, scf 2019, san diego, ca, usa, june 25–30, proceedings. volume 11514. Springer.
- Kamaruddin, S. and Ravi, V. (2016). Credit card fraud detection using big data analytics: Use of psoaann-based one-class classification. In *Proceedings of the International Conference on Informatics and Analytics*, pages 1–8, Pondicherry, India.
- Li, X. and Zhao, L. (2021). Bank fraud detection using long short-term memory networks with attention mechanism. *Expert Systems with Applications*.
- M. Fahmi, A. H. and Nagati, K. (2016). Data mining techniques for credit card fraud detection: Empirical study. In *Sustain Vital Technol Eng Inf*, pages 1–9.
- Moneyzine (2023). Credit card fraud statistics & facts for 2023.
- Payments, C. (2023). Credit card fraud statistics 2023. Report, N. (2023). Global fraud losses 2023 projections. ResearchGate (2022). The effect of imbalanced data on credit card fraud detection.
- Salam, M. A., Fouad, K. M., Elbably, D. L., and Elsayed, S. M. (2023). Federated learning model for credit card fraud detection with data balancing techniques. *Neural Computing and Applications*.
- Stout, N. Undersampling and oversampling statistics visual example. Pinterest.
- Suvarna, R. and Kowshalya, A. M. (2020). Credit card fraud detection using federated learning techniques. *J Web Eng Technol*, 7(3):356–367.
- V. Felbab, P. K. and Horváth, T. (2021). Optimization in federated learning.
- X. Niu, L. W. and Yang, X. (2019). A comparison study of credit card fraud detection: Supervised versus unsupervised. *arXiv preprint arXiv:1904.10604*.
- X. Yao, T. Huang, C. W. R. Z. and Sun, L. (2019). Towards faster and better federated learning: A feature fusion approach. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 175–195, Taipei, Taiwan. IEEE.
- Zhang, W. and et al., T. W. (2021). Dynamic fusion-based federated learning for covid-19 detection. In *IEEE Internet Things J*, volume 8, pages 15884–15891.
- Zhang, Y. and Liu, H. (2022). Convolutional neural networks for credit card fraud detection with imbalanced datasets. *IEEE Transactions on Neural Networks and Learning Systems*.