International Journal of Science and Technology (IJST)
ISSN: 3049-1118, Volume- 3, Issue- 1 (Jan – Mar 2026)

Enhanced Credit Card Fraud Detection: Behavioral Features, SMOTE-ENN Balancing, and LSTM-AdaBoost Ensembles

Subash Timalsina[1], Subarna Sapkota[2], Prajwal Rai[3], Dipendra Ghimire[4]
[1&2]Student, Nepal College of Information and Technology, Pokhara University
[3]Student, Student, Kantipur City College, Purbanchal University
[4]Lecturer, Ara Institute of Canterbury, Canterbury University

ABSTRACT

Credit card fraud causes substantial financial losses to both consumers and financial institutions globally because of the increasing volume of online transactions and the sophistication of fraud schemes necessitate advanced detection systems. Detection accuracy is enhanced in this research through the fusion of advanced feature engineering, behavioral analysis, and neural network ensemble methods. Preprocessing of data is done to standardize the structures, kernel behavioral features of spending patterns, locations of transaction, frequency are elicited, and class imbalance is mitigated by means of SMOTE-ENN, which is later evaluated by LSTM networks with the ADA Boost to create a robust ensemble model for credit card fraud detection with and without behavioral feature integration. The results demonstrated that ensemble learning methods, particularly AdaBoost with behavioral features, achieved the highest overall performance, yielding an F1-score of 0.9642, ROC-AUC of 0.9867, and strong precision and recall values are 0.9803, 0.9485, respectively. In contrast, LSTM models benefited significantly from behavioral features, improving their ROC-AUC from 0.7732 to 0.8632 and F1-score from 0.738 to 0.79. Interestingly, AdaBoost without SMOTE-ENN underperformed (F1 = 0.4923, ROC-AUC = 0.3512), highlighting the importance of handling class imbalance effectively. Overall, our results suggest that combining ensemble methods with behavioral insights and appropriate data balancing techniques leads to highly accurate and reliable fraud detection.

Keywords: Fraud detection; behavior analysis; behavior-based fraud detection; LSTM; Ada Boost.

1. Introduction

Over the last ten years, e-commerce has grown a lot, leading to more people using credit cards. This rise in credit card use has also resulted in more fraudulent transactions (Randhawa et al., 2018). Credit card fraud poses significant losses to both consumers and financial institutions. The increasing volume of online transactions and the complexity of fraud schemes necessitate the development of advanced detection systems. Traditional rule-based methods are often

*Corresponding Author Email: subash.222508@ncit.edu.np
Published: 24 March 2026
DOI: https://doi.org/10.70558/IJST.2026.v3.i1.241200

inadequate due to their inability to adapt to evolving fraud patterns. Therefore, integrating machine learning techniques, particularly neural networks, and behavioral techniques has gained prominence in enhancing fraud detection systems.

1.1 Background

A credit card is considered fraudulent when someone else uses it without your authorization(Sailusha et al., 2020). Some people use others' credit card PINs or account details to make unauthorized transactions without having the actual physical card. Credit card fraud detection helps identify fraudulent transactions.

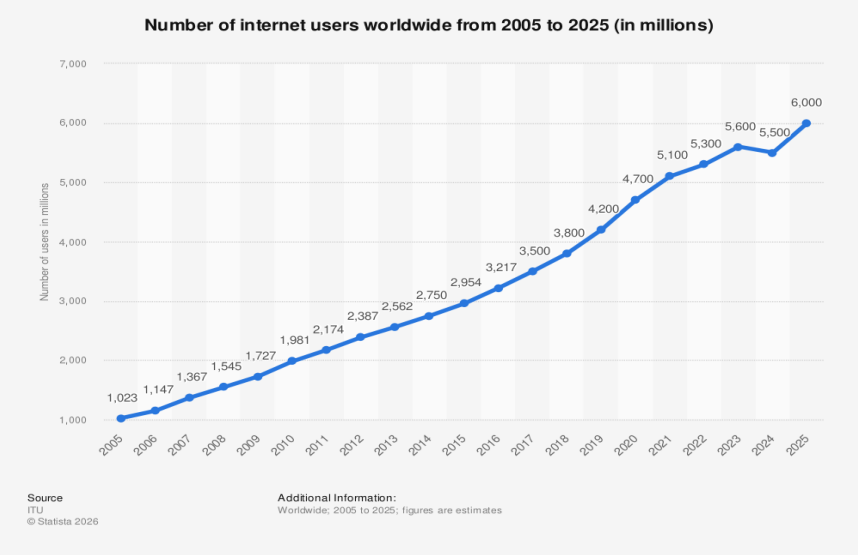

Figure 1: Global Internet Users (2005-2023, in Millions $)(Randhawa et al., 2018)

Figure 1 explains that the number of internet users is growing linearly, whereas it relates to revenue from Figure 2. In the year 2014, 2,750 million internet users were generating 1,336 billion US Dollars. By 2023, 5,400 million people using the internet were yielding 6151 billion US. Dollar's revenue.

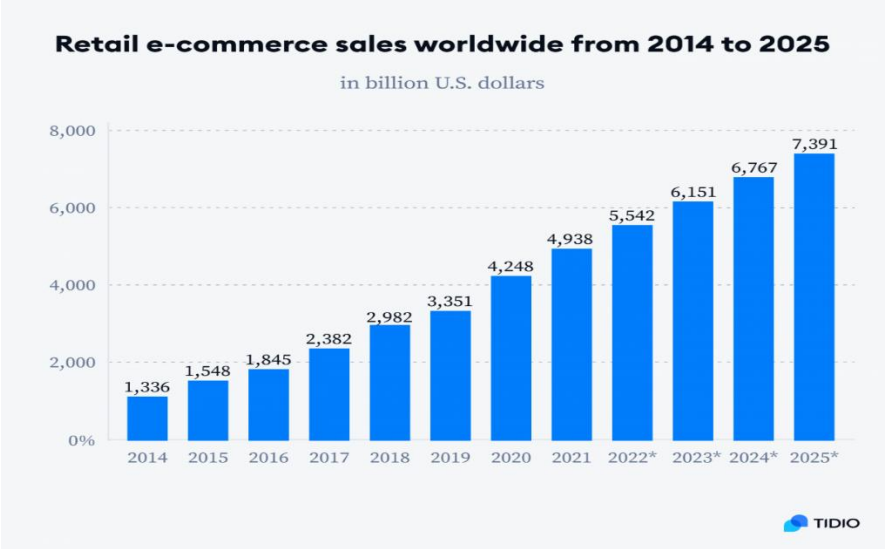

Figure 2: Global E-commerce sales (2014 to 2025) (Tingfei et al., 2020).

In 2018, losses from credit card fraud reached $27.85 billion, a 16.2% increase from $23.97 billion in 2017, and by 2023, the losses were $35 billion (Tingfei et al., 2020). Fraud is increasing every year, but losses can be minimized with effective fraud monitoring and prevention. Machine learning has been used to create various credit card fraud detection systems (Oo & Thein, 2022).

1.1 Problem Statement

Credit card fraud results in massive financial loss globally as virtual and traditional card transactions are on the rise. Conventional systems that rely on transaction time and transaction value are failing to detect dynamic behavioral patterns, such as location, frequency, spending patterns, and device information, which prompts high false positives, poor user experiences, and low fraud detection (Randhawa et al., 2018). In this study, these problems are addressed through behavioral feature engineering, SMOTE-ENN resampling of class imbalance, and hybrid LSTM-AdaBoost ensembles, which deliver the best results in the identification of fraud through adaptive methods.

1.2 Research Objectives

The main objective to this research is “To improve credit card fraud detection by integrating behavioral features with LSTM-AdaBoost ensembles and SMOTE-ENN balancing” method.

1.3 Significance/Rationale of the Study

The study aims to develop better methods for detecting and preventing credit card fraud, which can save millions of dollars for consumers, businesses, and financial institutions. The impact of the research on fraud detection will be felt in a person's daily activities. By reducing the incidence of fraud, financial institutions can maintain greater economic stability, which is beneficial for the broader economy.

1. Literature Review

The research from the Department of Electrical and Electronic Engineering Science at the University of Johannesburg, utilized sampling techniques such as SMOTE-ENN to balance a highly imbalanced dataset (Esenogho et al., 2022). They applied an LSTM Ensemble model with the training dataset derived from the results of SMOTE-ENN. Credit card fraud is very low in number, so the paper was focused on using the SMOTE-ENN approach for preparing balanced data. The dataset includes credit card transactions from European clients over two days in September 2013. The dataset is imbalanced, with 492 fraudulent transactions out of 284,807. All attributes, except "Time" and "Amount," are numerical due to transformations and coded as V1 to V28 for confidentiality. The "Class" attribute indicates fraud (1) or legitimacy (0). This study uses the AdaBoost algorithm to create a strong ensemble model with an LSTM network as the base. Future research will explore more resampling methods and better feature selection to improve classification performance. In the context of the research gap, though the research was conducted in the year 2022, people’s shopping trends are changing day by day, and detecting fraud based on transactional behavior would be a good challenge. Sailusha et al., (2020), also performed similar research in the year 2020, using the same data set mentioned before the research. The paper primarily focused on classifying fraud

and non-fraud transactions using algorithms such as Random Forest and AdaBoost. The results were analyzed using confusion matrices and ROC curves for different algorithms. For future research, it has been suggested to utilize deep learning algorithms to assess the accuracy of the algorithms. The author Chen & Lai, (2021), conducted joint research in 2021 and published in the Article Journal of Artificial Intelligence and Capsule Networks regarding "Deep Convolution Neural Network Model for Credit-Card Fraud Detection and Alert". Several models and algorithms have been utilized in credit-card fraud detection, including fuzzy Darwinian systems, fuzzy neural networks, hidden Markov models, artificial immune systems, k-nearest neighbors, genetic algorithms, support vector machines, Bayesian networks, and decision trees. Despite their utility, these models face challenges in training with highly skewed financial fraud datasets and in addressing the varying fraudulent behaviors. The paper suggested that the integration of real-time datasets and the continuous updating of fraud detection mechanisms are crucial for improving the efficiency of financial fraud detection systems. Fraud location, timing calculation, and various other features may be incorporated into a single algorithm in future work.

Rb & Kr, (2021) proposed "Credit Card Fraud Detection Using Artificial Neural Network" to identify fraudulent transactions. This study measures performance and accuracy, comparing the ANN with SVM and k-NN. It also explores methods like LightGBM, decision trees, and ensemble learning. According to the findings, the ANN-based system shows greater accuracy in fraud detection compared to traditional machine learning algorithms. The study discusses the potential of deep learning techniques to improve the reliability and efficiency of fraud detection systems .The author Sadgali et al., (2021), studied fraud detection systems by looking at how credit card users behave. They proposed a smart machine learning system that uses rough set theory to choose the most relevant features, fuzzy logic, and association rules to create rules that assess the risk of transactions. This system assigns a score to each transaction, showing how likely it is to be fraudulent, and this helps to improve overall fraud detection. The approach adds a layer of behavioral analysis to the fraud detection process, offering a new angle to traditional methods that only rely on transaction data. The paper focuses on analyzing behavioral trends for credit card fraud deetection (Sondinti & Yasmeen, 2022). The authors conducted a logical analysis to identify hidden patterns and trends, leveraging game-theoretical models to illustrate potential strategies of both attackers and defenders . The federated learning approach allows access to decentralized information and patterns, representing activities across various locations, times, devices, and behaviors.

2. Methodology

3.1 Research Methodology

The proposed research methodology given in Figure 3 for credit card fraud detection system.

The methodology starts with data collection from Kaggle , preprocesses data along with behavior feature extraction and then uses ensemble method to detect the fraud transactions.

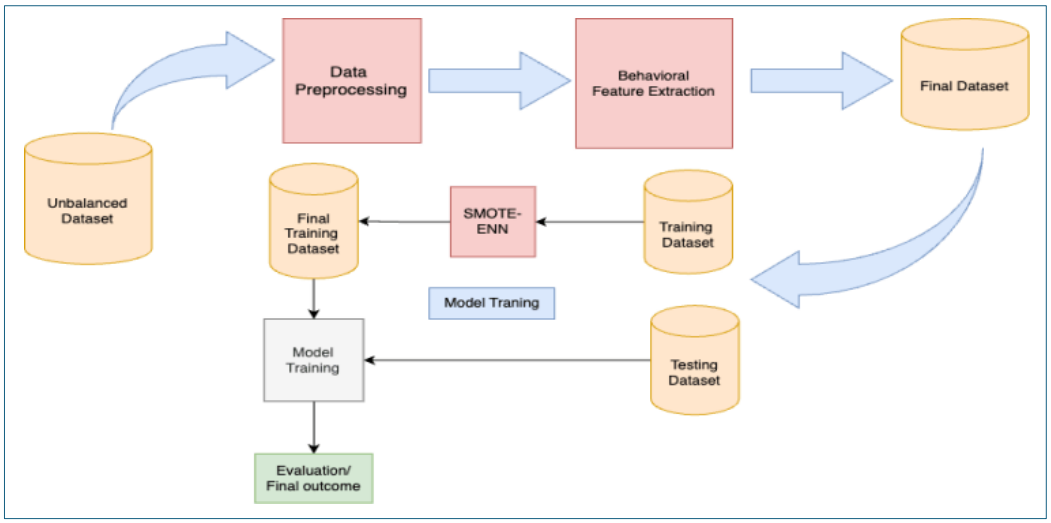

Figure 3: Proposed block diagram for credit card fraud detection system

3.2 SMOTE-ENN

The credit card dataset in the study is highly imbalanced, resulting in poor ML model performance, so SMOTE is used to address these issues . It is an oversampling approach that balances the dataset's class distribution by introducing synthetic samples into the minority class. Under-sampling strategies, such as ENN, provide a balanced dataset by removing part of the majority class samples. Meanwhile, under-sampling strategies can eliminate potentially important samples that may be critical in the learning process. As a result, the suggested credit card fraud detection model leverages the SMOTE-ENN approach to generate a balanced dataset.

Step 1: Oversampling:

1: Select a random sample x_i from the minority class.

2: Identify the K nearest neighbors of xi.

3: Create a synthetic sample p by randomly choosing one of the K nearest

neighbors q and connecting p and q to form a line segment in the feature space.

4: Assign the minority class label to the new synthetic sample.

5: Produce additional synthetic samples as a convex combination of the selected samples.

Step 2: Under sampling:

6: Choose a sample xi ∈ S, where S represents the total number of xi samples

from the minority class.

7: Find the K nearest neighbors of xi.

8: If xi has a larger number of neighbors from the other class, remove xi from

the sample set.

9: Repeat steps 6—8 for all examples in the dataset.

3.3 LSTM Ensemble (AdaBoost)

This study utilizes the AdaBoost algorithm to construct a robust ensemble model where the base model is an LSTM network. Given the credit card dataset containing U training instances, U = {(x1, y1), . . ., (xn, yn)}, where x* denotes the independent variable and y* represents the dependent variable (i.e., fraud or legitimate transaction). Let Dm denote the weight distribution of the training samples at the mth boosting iteration, which is initially assigned a value of 1/n at the first iteration. The total classification error of the current base model can then be calculated using:

$$\epsilon_m = \sum_{i=1}^{n} D_m(i) \ , L_m(x_i) \neq y \qquad (1)$$

Here, x_i is the input sample, and y_i is the label. Lm denotes the trained LSTM model at iteration m. The weight distribution of the input data is updated based on the previous classifier's performance, giving higher weights to misclassified instances and lower weights to correctly classified ones. The weight update follows:

$$D_{m+1}(i) = \frac{D_m(i)}{Z_m} \cdot e^{-\partial_m y_i Lm(xi)} \qquad (2)$$

where Z_m denotes a normalization parameter and ∂m represents the voting weight of the base learner Lm. The normalization parameter ensures the weight D_m+1(i) have a suitable distribution. Meanwhile, Zm and ∂m can be mathematically represented as:

$$Z_m = \sum_{i=1}^{n} D_m(i) \cdot e^{-\partial_m y_i Lm(xi)} \qquad (3)$$

$$\partial_m = \frac{1}{2} \ln\left(\frac{1-\epsilon_m}{\epsilon_m}\right)$$

After M iterations, the ensemble classifier consists of M base learners. Therefore, the final AdaBoost prediction is the combined predictions weighted by ∂_m:

$$F(x) = \text{sgn}\left(\sum_{m=1}^{M} \partial_m \cdot Lm(x)\right) \qquad (4)$$

where the sign function sgn(x) is computed using:

$$\text{sgn}(x) = 1 \text{ if } x > 0,\ 0 \text{ if } x = 0,\ -1 \text{ if } x < 0 \qquad (5)$$

LSTM models, trained with resampled data taken from SMOTE-ENN, are combined with AdaBoost to form an ensemble. The final predictions are made using weighted voting of the LSTM network results.

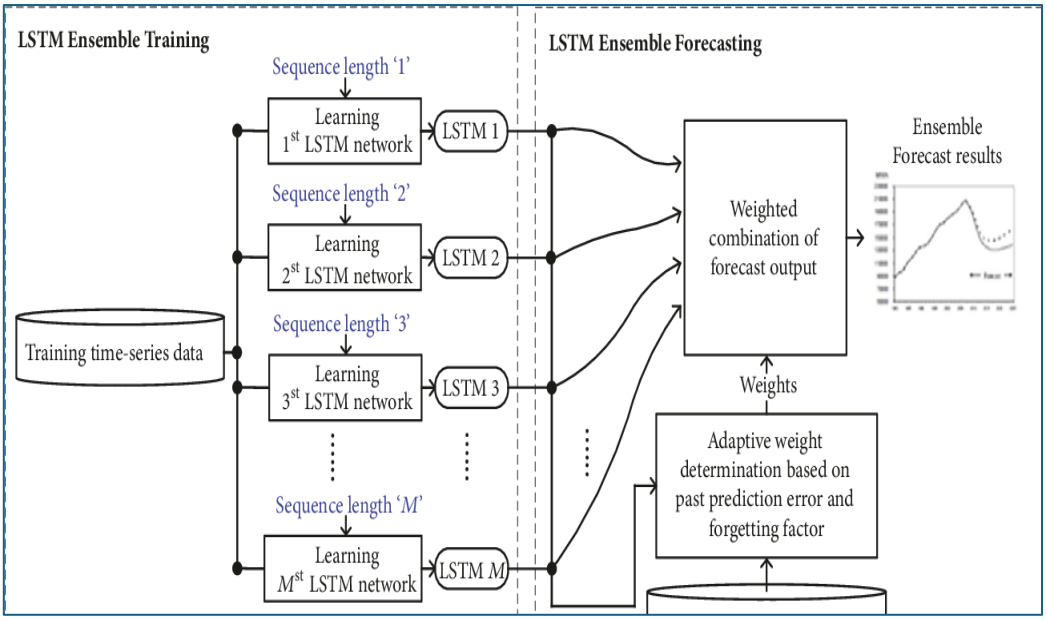

Figure 4: LSTM Ensemble (Ada Boosting M=50) (Randhawa et al., 2018)

3. Result and Discussion

4.1 Dataset

The dataset is taken from kaggle which includes over 20 million transactions from IBM's multi-agent virtual world simulation. It covers 2000 synthetic US consumers who travel globally, spanning decades of purchases and multiple cards per consumer.

Figure 5: Fraud vs Valid Data on the Data Set

Figure 5 shows that the number of instances or the value of "Not Fraud" is substantially greater than that of "Fraud." About a few thousand of the records are fraudulent out of 20 million records. SMOTE-ENN is used to balance the dataset on training data, and has a balanced dataset with non-fraud count data 19503276 and Fraud Count Data 18317511. From the dataset, features such as User, Card, Use Chip, Merchant Name, Merchant City, Merchant State, Zip, and MCC are used for computing behavioral features.

4.2 Data Preprocessing and Behavioral Feature Extraction

The first step is to preprocess the data, validate the data type, and generalize the data type to be uniform for the field. Missing values were filled by the mean method approach and eliminated duplicate records from the dataset and added the user and card information on the transaction row to provide more human-friendly features.

Several behavioral features such as temporal, spending, geographic features and fraud error patterns have been extracted from the existing by focusing on patterns and trends in fraudulent activities and these behavioral features are leveraged using advanced techniques like federated learning, privacy-preserving AI frameworks, and anomaly detection models to improve fraud detection while maintaining data privacy.

4.3 Dataset Splitting

The training set comprises 85% of the total transactions, with 80% of this segment dedicated to training and 20% for validation. The model testing set, which forms 5% of the entire dataset, includes 1,012,014 instances from the years 2019 and 2020. From the descending order by the transaction date. K-Fold Cross-Validation with 5 splits is applied to evaluate the performance of the model. The dataset was randomly divided into 5 equal subsets (folds). In each iteration, 4 folds were used for training and 1 fold for validation. This process was repeated 5 times, with each fold used once as the validation set. The final performance metric was obtained by averaging the results across all folds.

4.4 Model Development and Training

4.4.1 Long Short-Term Memory (LSTM) without Behavior Feature Extraction

The model uses a small L2 regularization value (0.000013) to prevent overfitting. The final dense layer applies a sigmoid activation, making the model suitable for binary classification tasks. Binary cross-entropy is selected as the loss function, aligning to distinguish between two classes. These settings help the model accurately predict outcomes in the subsequent results. Figure 6, shows the loss curve of an LSTM model trained without additional features. Both training and validation losses steadily decrease and converge, indicating that the model is learning effectively and generalizing well with minimal overfitting.

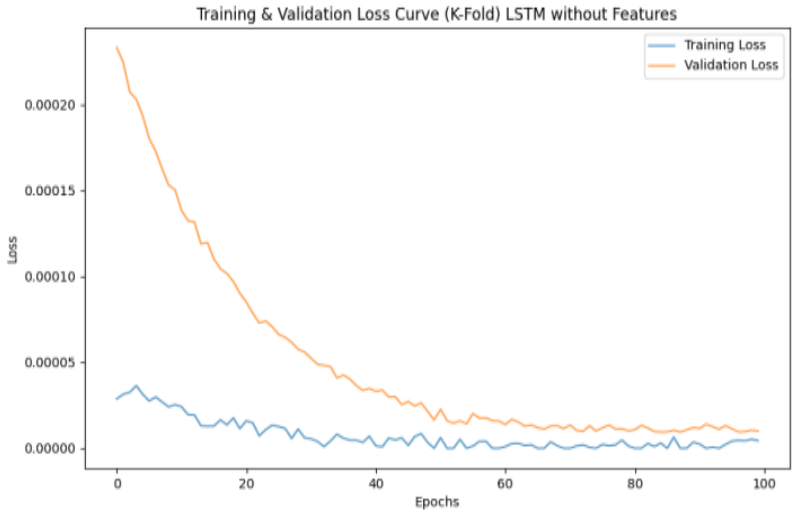

Figure 6: Loss Curve of LSTM without Behavior Feature extraction

Table 1: Confusion Matrix for LSTM without Behavior Features

		Predicted values	
	P + N	Positive	Negative
Actual values	Positive	5936 (TP)	2107 (FP)
	Negative	1078 (FN)	1002893 (TN)

Table 1 shows the performance of an LSTM model without behavioral features in predicting positive and negative cases. It correctly identified 5936 positive cases and 1002893 negative cases, but also made 2107 false positive and 1078 false negative errors. This model gained F1-score, ROC-AUC Score, precision and recall value 0.7380, 0.7732, 0.7784 and 0.8464 respectively.

4.4.2 LSTM results with Behavior Features:

Figure 7 shows a gradual decrease in both Validation and training loss with increasing epoch and it performed reduced validation loss when compared to TCN model. This prediction model learns slowly and stops learning after 20 epochs.

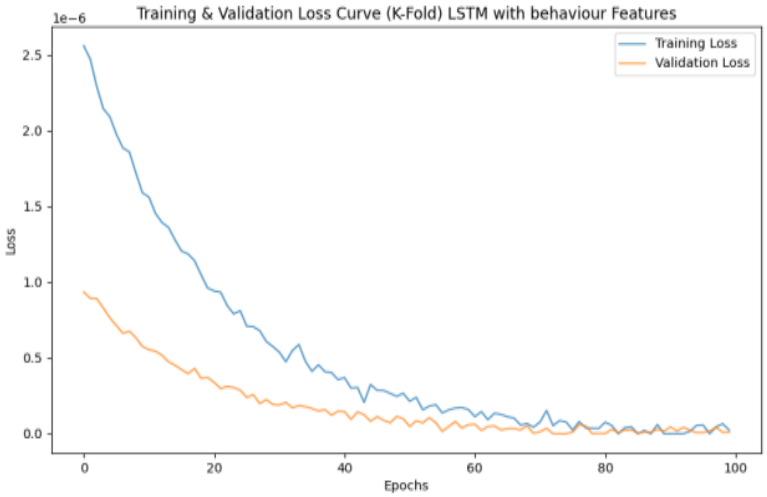

Figure 7: Loss Curve LSTM with Behavior Features

Table 2 shows the confusion matrix for the LSTM model with behavior features. It reveals that the model accurately predicted 6497 positives and 1003594 negatives. However, it also had 1406 false positives and 517 false negatives. This indicates that while the model is quite effective, there is still a margin for error. This method gave better output with the value for F1-score, precision and recall are 0.7900,8.8632 and 0.7837 respectively.

Table 2: Confusion Matrix for LSTM with Behavior Features

		Predicted values	
	P + N	Positive	Negative
Actual values	Positive	6497 (TP)	1406 (FP)
	Negative	517 (FN)	1003594 (TN)

4.4.3 Ada Boost (Ensemble Learning) without Behavior Features

Figure 8 shows the training and validation loss curves of an AdaBoost model trained without behavioral features using K-fold cross-validation. Both losses steadily decline and converge after around 60 epochs, indicating stable training and strong generalization. The low and closely aligned loss values suggest the model is performing well without signs of overfitting.

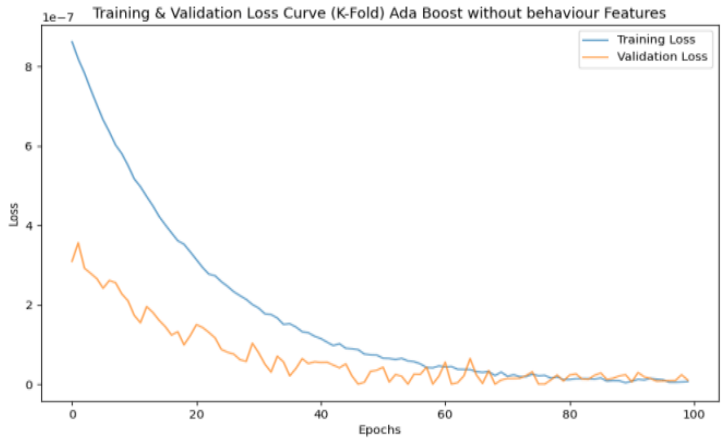

Figure 8: Ada Boost Loss curve without behavior features

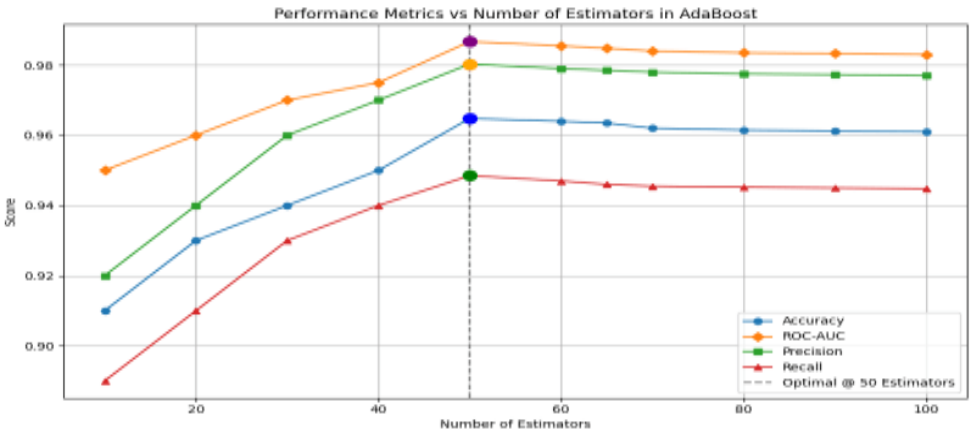

Figure 9: Performance Metrics vs Number of Estimators in AdaBoost

Table 3: Confusion Matrix Ada Boost without behavioral features

		Predicted values	
	P + N	Positive	Negative
Actual values	Positive	5647 (TP)	2051 (FP)
	Negative	1367 (FN)	1002949 (TN)

Table 3 presents the confusion matrix for the AdaBoost model without behavioral features. It shows that out of the actual positive cases, 5647 were correctly predicted as positive (true positives) while 2051 were incorrectly predicted as negative (false negatives). For the actual negative cases, 1367 were wrongly predicted as positive (false negatives), and 1,002,949 were correctly predicted as negative (true negatives).

4.4.4 Ada Boost result with Behavioral Features

Figure 10 displays the training result with respect to validation loss after we ran the model with 50 epochs and a validation split.

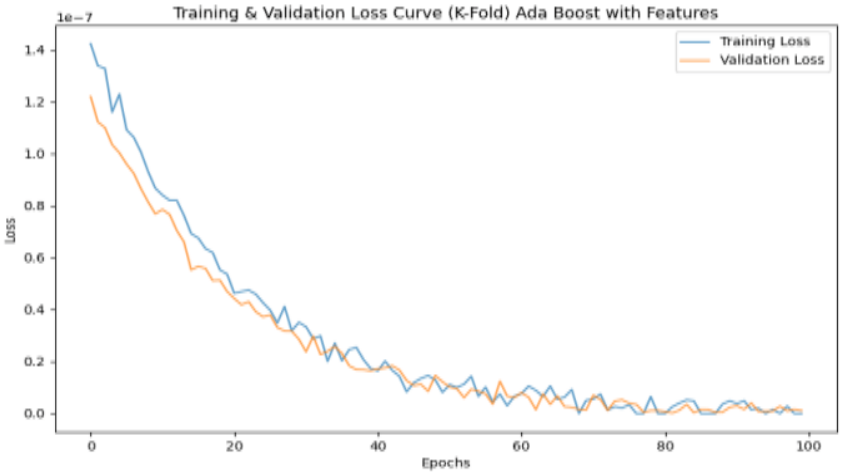

Figure 10: Ada Boost loss curve with Behavioral Feature

Table 4 presents the confusion matrix for the AdaBoost classifier, using behavioral features. It shows the number of true positives (6653), true negatives (1004867), false positives (133), and false negatives (361). This illustrates the model's ability to correctly identify both fraudulent and non-fraudulent transactions while maintaining a low rate of misclassification. This model gives the F1-score: 0.9642, Precision: 0.9803 and Recall: 0.9485.

Table 4 Confusion Matrix for Ada Boost with Behavior Features

		Predicted values	
	P + N	Positive	Negative

Actual values	Positive	6653 (TP)	133 (FP)
	Negative	361 (FN)	1004867 (TN)

Further investigation revealed that many of the 133 FP and 361 FN cases can be attributed to noisy or missing data, especially fields such as latitude, merchant state, merchant city, and others. These missing values likely led to misclassification and highlight the importance of thorough data preprocessing. Addressing these data quality issues by properly handling missing and noisy inputs during model training and validation could further reduce misclassifications, thereby improving overall model accuracy and reliability.

4.4.5 Comparision of Results with another Baseline Model

The given Table 5 gives an overall summary of accuracy of all baseline models with the value of F1-score, Precision, Recall and ROC-AUC score. It compares various classifiers with and without behavioral features. Across models, adding features consistently improves F1-score, ROC-AUC, precision, and recall, indicating enhanced performance. The AdaBoost model with behavioral features performs the best overall, achieving 98.67% ROC-AUC, along with high precision, recall, and F1-score. Notably, LSTM also performs well, especially without features, showing strong recall.

Table 5 :Result Comparison with Various Models

Classifier	**Behavioral Features**	**F1-Score**	**ROC-AUC Score**	**Precision**	**Recall**
Long Short-Term Memory (LSTM)	No	0.738	0.7732	0.7384	0.8464
	Yes	0.7900	0.8632	0.7963	0.7837
AdaBoost with Ensemble Learning (Proposed model)	No	0.7676	0.8215	0.7336	0.8051
	Yes	**0.9642**	**0.9867**	**0.9803**	**0.9485**

4. Conclusion and Future Recommendations

5.1 Conclusion

The study's results demonstrated significant improvements in the detection of fraudulent transactions, effectively reducing the occurrences of false positives and false negatives. This was achieved through a robust model that was rigorously evaluated using a variety of performance metrics, including accuracy, precision, recall, and the F1-score inclusion of

behavioral attributes trapping user patterns and transaction attributes were critical, and improved the performance of the various classifiers. From the experiment, Long Short-Term Memory (LSTM) networks went up in a F1-score of 0.738 (no features) to 0.790 (with features), and ROC-AUC increased by 0.7732 to 0.8632. Even more significant improvements were realized with AdaBoost whose F1-score improved to 0.9642, ROC-AUC improved to 0.9867, precision improved to 0.9803 and recall was improved to 0.9485 after SMOTE ENN resampling and feature integration. These results demonstrate that the integration of powerful models with situational elements can give you better fraud detection.

5.2 Future Recommendations

In the evolving science of data science and machine learning, fraud detection requires a novel feature selection method to identify variables with high impact on the results and resampling algorithms such as SMOTE ENN to ensure class imbalance. Real-time datasets have additional potential (e.g., an opportunity to continuously retrain the model to meet the changing fraud techniques and facilitate autonomous learning based on the data stream). Future directions would include hybrid systems that combine LSTM, AdaBoost and explainable AI (i.e. SHAP values) to gain interpretable real-time systems and edge computing with low-latency deployment. All these measures will have a proactive and resilient framework against emerging threats.

REFERENCE

Chen, J. I.-Z., & Lai, K.-L. (2021). Deep Convolution Neural Network Model for Credit-Card Fraud Detection and Alert. *Journal of Artificial Intelligence and Capsule Networks*, *3*(2), 101–112. https://doi.org/10.36548/jaicn.2021.2.003

Esenogho, E., Mienye, I. D., Swart, T. G., Aruleba, K., & Obaido, G. (2022). A Neural Network Ensemble With Feature Engineering for Improved Credit Card Fraud Detection. *IEEE Access*, *10*, 16400–16407. https://doi.org/10.1109/ACCESS.2022.3148298

Oo, M. C. M., & Thein, T. (2022). An efficient predictive analytics system for high dimensional big data. *Journal of King Saud University - Computer and Information Sciences*, *34*(1), 1521–1532. https://doi.org/10.1016/j.jksuci.2019.09.001

Randhawa, K., Loo, C. K., Seera, M., Lim, C. P., & Nandi, A. K. (2018). Credit Card Fraud Detection Using AdaBoost and Majority Voting. *IEEE Access*, *6*, 14277–14284. https://doi.org/10.1109/ACCESS.2018.2806420

Rb, A., & Kr, S. K. (2021). Credit card fraud detection using artificial neural network. *Global Transitions Proceedings*, *2*(1), 35–41. https://doi.org/10.1016/j.gltp.2021.01.006

Sadgali, I., Sael, N., & Benabbou, F. (2021). Human behavior scoring in credit card fraud detection. *IAES International Journal of Artificial Intelligence (IJ-AI)*, *10*(3), 698. https://doi.org/10.11591/ijai.v10.i3.pp698-706

Sailusha, R., Gnaneswar, V., Ramesh, R., & Rao, G. R. (2020). Credit Card Fraud Detection Using Machine Learning. *2020 4th International Conference on Intelligent Computing*

and Control Systems (ICICCS), 1264–1270. https://doi.org/10.1109/ICICCS48265.2020.9121114

Sondinti, L. R. K., & Yasmeen, Z. (2022). Analyzing Behavioral Trends in Credit Card Fraud Patterns: Leveraging Federated Learning and Privacy-Preserving Artificial Intelligence Frameworks. *Universal Journal of Business and Management*, *2*(1), 38–49. https://doi.org/10.31586/ujbm.2022.1224

Tingfei, H., Guangquan, C., & Kuihua, H. (2020). Using Variational Auto Encoding in Credit Card Fraud Detection. *IEEE Access*, *8*, 149841–149853. https://doi.org/10.1109/ACCESS.2020.3015600