Behavioural-Aware Hybrid Ensemble Model with Cost-Sensitive and Explainable Credit Card Fraud Detection Using Machine Learning

Jeet Bandyopadhyay[1], Disha Roy[2], Roshni Ghosh Tagore[3], Hrishav Hari[4], Mahendrani Chanda[5]
[1,2,3,4,5]*B. Tech, Department of Computer Science and Engineering, (Artificial Intelligence and Machine Learning), Institute of Engineering and Management, Kolkata*

***Abstract*—With a rise in digital transactions, credit card fraudulence is an issue of growing concern. Correctly detecting fraud transactions is hard due to data imbalance, the dynamic nature of fraud patterns, and the low interpretability of existing models. To increase the accuracy on fraud detection, the proposed model introduced a Behavioural-Aware Hybrid Ensemble Model that takes both supervised and unsupervised machine learning benefits into account. The proposed model is optimal in terms of high performance and practical value by extracting behavioural features including transaction time, frequency, and location and by employing cost-sensitive learning and model explainability with SHAP. Our approach makes use of Random Forest, XGBoost, and Logistic Regression as supervised classifiers and Autoencoder and Isolation Forest as an unsupervised anomaly detection technique. To deal with data imbalance, the model is trained using SMOTE, and it is assessed using real-world metrics such as cost, recall, and AUC-ROC.**

***Index Terms*—Credit Card Fraud Detection, Hybrid Ensemble model, Cost Sensitive-learning, SMOTE, Autoencoder, Explainable AI (XAI) using SHAP**

I. INTRODUCTION:

The rapid development of financial technology and the fast growth of E-commerce in recent years have accelerated the usage of online credit card transactions. With that increase, credit card fraud has become more common and sophisticated. Criminals of one type or another such as phishers and skimmers are using phishing, skimming, or hacking incidents to illicitly obtain cardholder information's. Credit card fraud imposes severe financial burdens on both consumers and merchants. The costlier reality is that the credit card fraud ecosystem resulted in global losses of $21.84 billion in 2015 and surged to $28.65 billion in 2019, according to industry estimates. It is an urgent problem and one that underscores the necessity for improved fraud detection systems. Many machine learning approaches have been employed to address this issue for both the online and offline fraudulent transaction detection by monitoring behaviours and detecting anomalies. But most current approaches fail, because they do not generalize datasets properly where fraudulent transactions are much lower than legitimate ones. They are not transparent and the experts can hardly tell why a transaction would be flagged. The disparate costs between false alarms and missed frauds are ignored.
In this paper, the proposed model presents an innovative large-scale credit card fraud detection model to fight these challenges.

The main contributions of the work:
A hybrid fraud detection strategy combining the strength of supervised learning (XGBoost) with the abilities of the unsupervised detection (Autoencoder and Isolation Forest) to determine the fraud patterns/techniques as well as the slight anomalies.
The employment of Synthetic Minority Oversampling Technique (SMOTE) to maintain the balance of the dataset, and certain transactional features, such as the ratio of transactions by hour and the volume metrics, are used to represent the different types of transaction better.
A cost-sensitive evaluation approach that emphasizes reducing the amount of more harmful false negatives (fraud cases that were not predicted) rather than merely minimizing false positives is in line with real world based financial risk.

Interpretation of model's decisions using SHAP (SHapley Additive exPlanations) to allow transparent and interpretable decisions by domain experts, boosting trust and supporting better decision making.
Validation of our method on a very imbalanced European credit card transaction dataset is available publicly. Our model performs well in terms of precision, F1-score, recall, AUC and MCC, indicating the improvement over traditional classifiers.

The rest of this paper is structured as follows:
In section II, we will point out the work related to the current one.
In Section III, we describe the methodology, machine learning models and preprocessing processes we applied.
Section IV discusses about the results, experimental setup and the evaluation metrics.
Section V, then concludes with future prospects.

II. LITERATURE REVIEW:

The discovery of credit card fraud has become more and more important since the explosive growth of card-based transactions. Burdened by the explosive increase in sophisticated and emerging fraud patterns, classic rule-based systems are arduous to maintain, thus motivating researchers to investigate numerous mainstream machine learning (ML) and deep learning (DL) methods. Many researches have focused on various methods, to solve some major problems such as class imbalance, real-time detection, and model performance, but there are still blank spots need to be filled.

Ileberiet *etal.* [1] leveraged the AdaBoost algorithm combined with SMOTE to handle class-imbalanced data and obtained high recall results. But their method was based on feature-only statistics and lacked inputs from behavior and also easily overfitted in noisy conditions. C/s Rejwan Bin Sulaiman *et al.* [2] presented some ML classifiers like SVMs, DTs, and Neural Network followed by the importance of preprocessing and imbalance handling, but did not show the comparison, performance metrics or implementation details. Dal Pozzolo *et al.* [3] proposed an economical learning framework based on under sampling specifically designed to minimize fraud cost but did not include any unsupervised models or explainable approaches such as SHAP or LIME. Sahin *et al.* [4] managed to successfully apply unsupervised methods including Isolation Forest and clustering when there were sparse labels but had high false positives and failed to use prioritized cost. There is also a significant focus on deep learning models. Jurgovsky *et al.* [5] employed LSTM networks for the purpose of catching sequence-wise transaction anomalies, which appear to be good at capturing temporal dependencies, however, hard to interpret and require high computational load. Fawaz Khaled Alarfaj*et al.* [6] continued with more deep learning architectures like CNNs and LSTMs along with traditional machine learning algorithms KNN and SVM. DL showed better performance but the proposed solution required high computational resources and was not suitable for real-time systems without enough transparency issue system. Ruttala Sailusha *et al.* [7] worked on hyperparameter tuning that was beneficial for classifier performance, although they did not provide resampling techniques and model interpretability, that are required to deploy a model in a trustworthy manner.

Based on the above, Carcillo *et al.* [8] pointed out that it is important to develop ensemble models for fraud detection in streaming data and confirmed that ensemble and hybrid models are more effective compared to single classifiers. Bhattacharyya *et al.* [9] combined Random Forests with cost-sensitive matrices to address the financial impact of false negatives but lacked behavioral insights. Fiore *et al.* [10] proposed hybrid models using Neural Networks and SVMs, yet their solution was complex and lacked explainability.

Nguyen et al. [11] used Graph Neural Networks to capture user-merchant interactions, offering behavioral context but requiring heavy feature engineering and lacking interpretability. Bauder and Khoshgoftaar [12] reviewed deep learning models and found that most lacked transparency, scalability, and were not robust to class imbalance. Nonetheless, there are still some gaps. Several studies also raised other related methodological issues; for example, a considerable amount of studies limit themselves to a few types of classifiers, thus failing to investigate advanced ensemble methods such as XGBoost or Extreme Trees (ET). Class imbalance receives almost

no consideration in the existing learning algorithms most specifically boosting techniques and AdaBoost was not many times touched upon. Furthermore, the majority of work employed only basic performance measures like classification accuracy which sidelining some key aspect of fraud handling problem. Especially, significant explanation interpreted from a model which is necessary for regulatory transparency issue within financial institution was largely ignored in most studies investigated model published so far. Where these studies tend to fall short is in incorporation of behavior-aware features, which may provide useful context on what transactions users tend to make. Also, many do not proactively address the financial impact of false negatives and only a few models even put in pipeline explanation of the model. The gaps presented above are addressed in our proposed model by:

Combination of two approaches, i.e., Supervised and Unsupervised (Autoencoder + Isolation Forest + Logistic Regression + Random Forest + XGBoost) is used. Behavior-based engineering of such characteristics as inter-transaction time and variations in user location, Data balancing by SMOTE, Incorporation of SHAP in explainability have been implemented. The model not only compares using traditional measures (accuracy, recall) but also real-world (cost, interpretability). Such a complete, hybrid, and explainable framework not only guarantees high predictive performance but also includes operation reliability and regulatory compliance to deploy the solution in a real environment financial system.

Comparative Analysis of Credit Card Fraud Detection Models

Table 1: Existing Research Works vs this research works.

Ref.	Authors	Methodology	Strengths	Limitations	Gaps Filled by Proposed Model
[1]	Ileberi *et al.* (2021)	AdaBoost + SMOTE	Improved recall for imbalanced data	Lacks behavioral features; sensitive to noise and overfitting	Adds behavioral features, combines unsupervised models, adds explainability via SHAP
[2]	Rejwan Bin Sulaiman *et al.* (2019)	Review of ML models (SVM, DT, NN)	Highlights preprocessing importance and class imbalance	No implementation; lacks performance comparison	Implements real models with detailed evaluation, adds hybrid/ensemble strategy
[3]	Dal Pozzolo *et al.* (2015)	Cost-sensitive learning + undersampling	Optimizes based on fraud cost	Lacks diverse models and interpretability tools	Introduces model explainability (SHAP), uses hybrid learning
[4]	Sahin *et al.* (2011)	Isolation Forest + clustering	Useful for unlabeled data; anomaly detection	High false positives; no cost-based prioritization	Adds cost evaluation metrics, uses supervised + unsupervised fusion
[5]	Jurgovsky *et al.* (2018)	LSTM (deep learning)	Captures sequential fraud patterns	Requires high computation; lacks interpretability	Keeps efficient models; adds explainability and cost-efficiency
[6]	Alarfaj *et al.* (2022)	CNNs, LSTMs vs SVM, KNN	DL shows high performance	DL models lack transparency, unsuitable for real-time	Uses interpretable models, more suitable for deployment

[7]	Sailusha *et al.* (2022)	Supervised ML + hyperparameter tuning	Parameter optimization improves results	No resampling or explainability	Uses SMOTE, SHAP, cost and performance evaluation
[8]	Carcillo *et al.* (2018)	Streaming data + ensemble	Emphasizes real-time hybrid models	No behavior-awareness or explainability	Adds behavior features, interpretability, and cost-driven metrics
[9]	Bhattacharyya *et al.* (2011)	Random Forest + Cost Matrix	Reduces financial cost of false negatives	Does not include behavioral features	Adds behavioral analytics and real-time cost prioritization
[10]	Fiore *et al.* (2019)	Hybrid of Neural Networks + SVM	Improved detection using hybrid model	Complex; lacks transparency and explainability	Uses simpler interpretable models with SHAP for transparency
[11]	Nguyen *et al.* (2021)	Graph Neural Networks (GNN)	Leverages user-merchant relationships (behavioral)	Requires high feature engineering; limited interpretability	Adds scalable behavioral features and interpretable ensemble models
[12]	Bauder & Khoshgoftaar (2020)	Review of Deep Learning for Fraud	Comprehensive DL review	DL methods lack transparency and class imbalance treatment	Focuses on explainability (SHAP), SMOTE for imbalance, and cost-driven hybrid modeling

Here, Table [1], shows the comparison between this research work and the existing research works.

III. METHODOLOGY

The block diagram given below describes the workflow of the proposed model used in this paper. It starts with data set collection where raw data is collected for analysis then data pre-processing is done and organized using SMOTE. Then the model is implemented by developing and setting up the analytical model. Henceforth, Model Aggregation is been done by combining multiple Machine Learning models for better results and model performance is evaluated by Evaluation Matrix and finally prediction of the model is done.

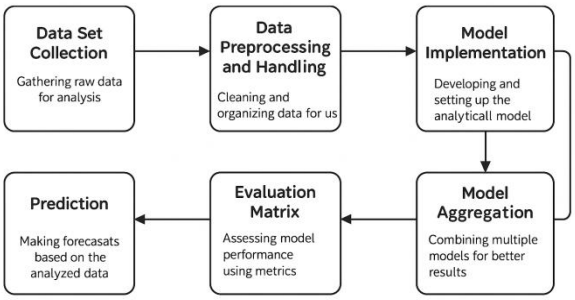

Fig 1: Block Diagram of Credit Card Fraud Detection Model

A.Dataset Information

We use Credit Card Fraud Detection Dataset from Kaggle, we obtain it from an open-source platform named Kaggle which is accessible to everyone. It consists of 284,807 transactions of European cardholders in the period of two days in September 2013, with 492 frauds, which represents only 0.17% of all transactions. Each transaction contains 30 features -anonymized components V1 to V28, 'Time', 'Amount' and the class label (whether it has been flagged as fraud or not). Such imbalanced high-quality information creates a particularly difficult task that

expands the fraud detection beyond mere accuracy in synthetic data and to meet the challenges of the real-world fraud data in the least positive-class sample size.

B.Data Preprocessing

To suit the dataset for training models, it is worthy to note the following steps are imperative when preparing the data:
Feature Scaling: The features of 'Time' and 'Amount' are quite different in scales which can have an adverse effect on machine learning model performance. We used StandardScaler to standardize these two features with making them something alike. As it is only less than 1% of cases are fraud, we used SMOTE (Synthetic Minority Oversampling Technique) to generate artificial samples for the minority class. This helps in preventing classifier being biased towards majority class(non-fraud). Next, the SMOTE-balanced dataset was trained/tested at 80% -20% split to accurately assess model generalization performance.

Table 2: Class Distribution

Class	Count	Percentage
Non-Fraud	284,315	99.83%
Fraud	492	0.17%
Total	284,807	100%

Here, Table [2], shows the class distribution of the dataset and the percentage of its classification

C. Feature Engineering

Behavior analytics play an important role in detection of fraudulent actions because of that. From other way to detail the changes in user behavior, the following specific factors were engineered by us: Transaction velocity: Helps in identifying bursty activities by determining the number of transactions a user makes within the last day or hour.
Average Amount per User: Here one can judge the value of transactions, so the pulses can be found. Temporal and behavioral information provided by these engineered features do not only enhance the dataset they also make it more sensitive to various patterns in user behavior that may indicate fraud.

D. Hybrid Ensemble Architecture

This study proposes an ensemble model hybrid that combines the unsupervised anomaly detection and supervised classification, for the purpose of giving a balanced approach to identifying the known, as well as unknown patterns of fraud.

Unsupervised Models:
I. Autoencoder:
These are trained especially on normal transactions to learn the typical patterns in the domains, any transaction with high reconstruction error is flagged as an anomaly.

II. Isolation Forest:
It is based on the feature selection by randomly separating values of features to isolate anomalies which are faster to detect.

Supervised Models:
I. Random Forest
II. XGBoost
III. Logistic Regression
The learning algorithms use the SMOTE-balanced dataset to train these models, and they also assigned class labels in detecting fraud and non-fraud transactions.

E.Prediction Workflow of the Ensemble

The ensemble system processes predictions in a pipeline:
This practical guide shows you the

Step 1: If either the Autoencoder or Isolation Forest label any transaction as an outstanding one, it is classified as a “high risk” product.

Step 2: After categorizing the “high risk” deal, then all the three types of supervised models are made to predict, and their predictions are gathered.

In the Step 3: The concluding class is determined by a weighted majority voting mechanism. XGBoost was given a higher weight as it showed better precision in catching fraud cases during validation test. This complex model guarantees that even in the case of rarer unknown frauds, there are methods to detect them using unsupervised paradigm, while supervised models do the job with high precision.

F.Cost-Sensitive Evaluation

In contrast to the usual methods that focus on a pure accuracy, we have developed a cost-sensitive

evaluation, method that takes into account the cost of the prediction error in real life.

Let:

- FN = False Negatives (fraud transactions predicted as normal)
- FP = False Positives (normal transactions predicted as fraud)
- TP = True Positives
- TN = True Negatives
- C_{FN} = Cost of missing a fraud (e.g., ₹1000)
- C_{FP} = Cost of investigating a normal transaction (e.g., ₹10)

Then, the Total Cost is calculated as:

Total Cost=$(FN \times C_{FN}) + (FP \times C_{FP})$ [1]

Cost Ratio (CR)=Total Cost / Total Transactions [2]

G. Evaluation Metrics

The combination of classic and cost-sensitive criteria was applied to evaluate the performance of the models:

Accuracy: Predicting it correct in general.

Precision: How many frauds that were flagged up, were indeed fraud.

Recall: How many frauds have been successfully nabbed.

F1-Score: The Harmonic average of precision and recall. The formula for F1 score is [1]:

F1=2*(Recall*Precision)/Recall+Precision [3]

AUC-ROC: The area under the ROC curve of overall model discrimination power.

Confusion Matrix:

Graphical representation of distribution of prediction.

Custom Cost Score:

Total cost incurred with our penalty weights which are based on reality.

Such measures make sure that we pick models that will not only be statistically sound but also minimize true losses and labor expended in the field.

Table 3: Model Performance Comparison

Model	Accuracy	Precision	Recall	F1 Score	ROC AUC Score
Logistic Regression	97.80	6.72	90.54	12.51	94.17
Autoencoder	97.83	0.29	3.37	0.53	50.69
XGBoost	99.94	88.80	80.40	84.39	90.19
Random Forest	99.95	93.54	78.37	85.29	89.18

Table [3] shows the comparison between the different supervised and unsupervised models used in this paper.

H. Criteria of Selection of Models

The last model is selected on the grounds of a balance between AUC-ROC and Cost Ratio (CR): Models which maximize the AUC, but have low CR are preferred. This plan does not engage in over-aggressive approach to detecting fraud (high in number of false-positives) and it does not under-detect it (high in the amount of loss). All in all, the Efficient Fraud Detection = High Recall+ Low-Cost Ratio.

IV. RESULT AND DISCUSSIONS

The confusion Matrices of the Different Machine Learning Models are

A. Random Forest Evaluation:

Here in Fig 2, the confusion matrix for Random Forest shows the Accuracy of 99.95 %, Precision of 93.54 %, Recall Value of 78.37%, F1 Score of 85.29% and finally ROC AUC Score of 89.18%. obtained by Random Forest Supervised Model.

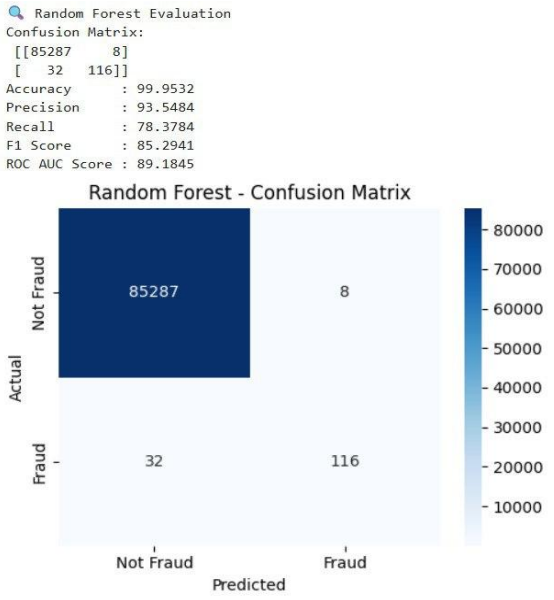

Fig 2: Confusion Matrix for Random Forest

B. XG Boost Evaluation:

Here in Fig 3, the confusion matrix for XG Boost shows 99.94% accuracy and 88.80 %, precision Recall Value of 80.40%, F1 Score of 84.39% and finally ROC AUC Score of 90.19%obtained by XG Boost Supervised Model.

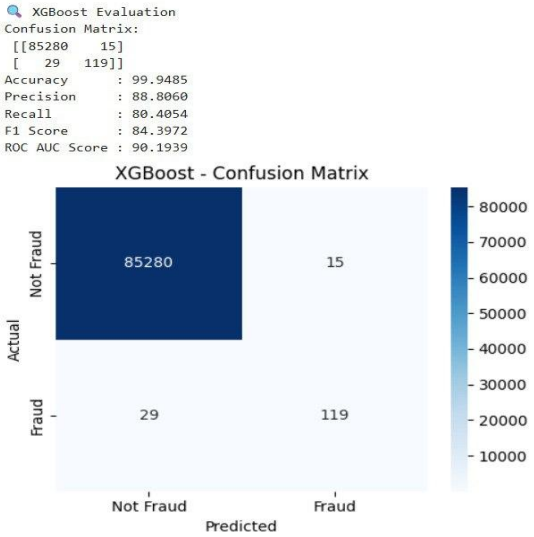

Fig 3: Confusion Matrix for XG Boost

C. Autoencoder Evaluation:

Here in Fig 4, the confusion matrix for Autoencoder shows the Accuracy of 97.83%, Precision of 0.29%, Recall Value of 3.37%, F1 Score of 0.53% and finally ROC AUC Score of 50.69% obtained by XG BoostSupervised Model.

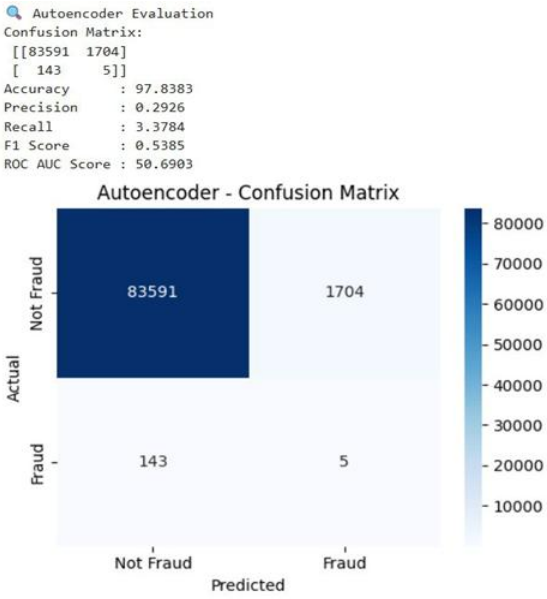

Fig 4: Confusion Matrix for Autoencoder

D. Logistic Regression Evaluation:

Here in Fig 5 the confusion matrix for Logistic Regression shows the Accuracy of 97.80%, Precision of 6.72%, Recall Value of 90.54%, F1 Score of 12.51% and finally ROC AUC Score of 94.17% obtained by Logistic Regression Supervised Model.

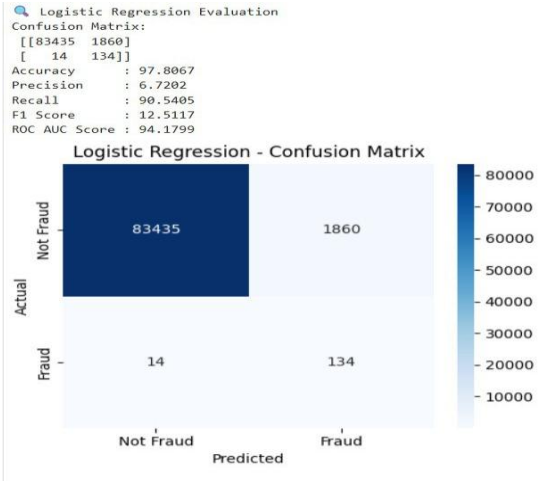

Fig 5: Confusion Matrix for Logistic Regression

E. Isolation Forest Evaluation:

Here Fig 5is a scatter plot that shows transaction behavior, with red points indicating anomalies (likely fraud) andblue points showing normal transactions. The model detects outliers based on transaction time and amount, helping to flag unusual or suspicious patterns.

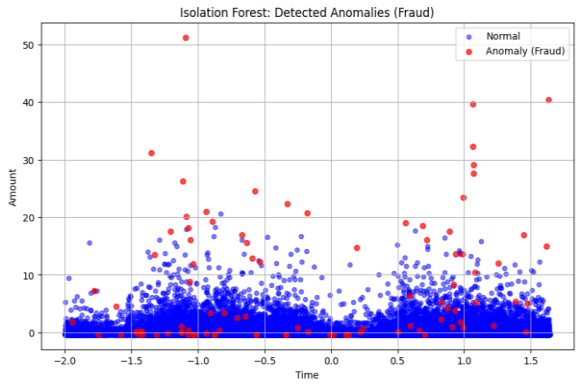

Fig 6: Scatter Plot for Isolation Forest Evaluation

F. Autoencoder Reconstruction Error:

Here in Fig 7 displays the distribution of reconstruction errors (MSE) from an autoencoder model for both normal and fraudulent transactions.

Fraudulent transactions usually have higher reconstruction errors, making them easier to identify as anomalies in the dataset.

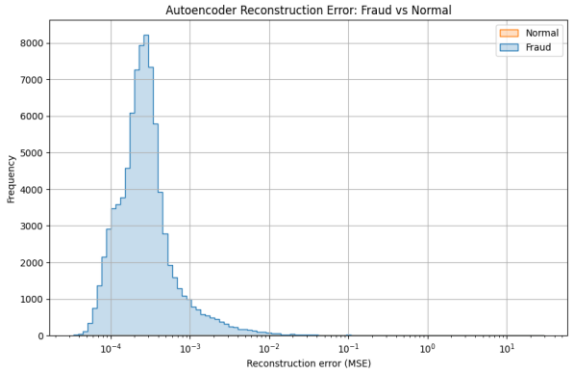

Fig 7: Histogram for Reconstruction Error

G. SHAP Feature Importance for XGBoost Model:

Here in Fig 8, displays the average absolute SHAP values, representing the impact of each feature on the model's predictions on average. Features like V4, V14, and V12 have the highest effect on the model's output, indicating their critical role in distinguishing between fraudulent and legitimate transactions.

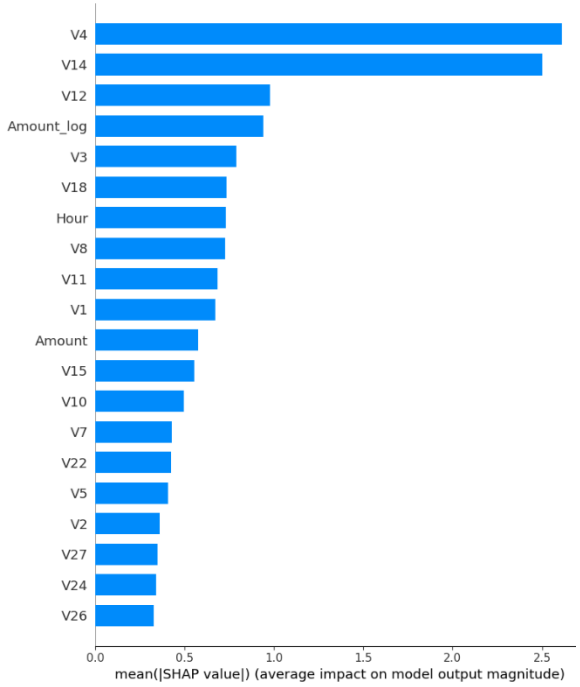

Fig 8: Bar Plot for SHAP Feature Importance

H. SHAP Beeswarm Plot:

Feature-Wise Impact and Value Distribution: SHAP beeswarm plot is showing the impact of the top features on the model output. Each point represents a SHAP value for an individual instance and color indicating the feature value (red = high, blue = low). Features like V4 and V14 have the maximum influence on predictions.

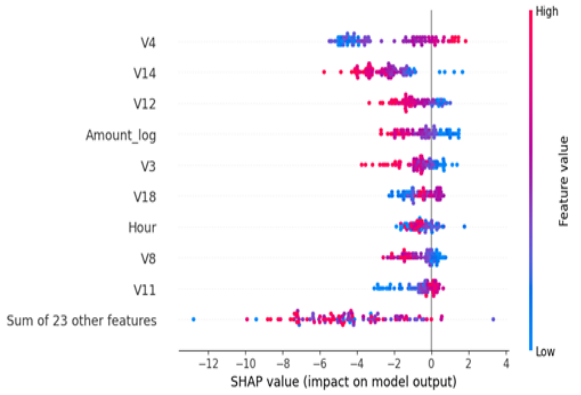

Fig 9: SHAP Beeswarm Plot

Table 4: Comparison of the Proposed Fraud Detection Model with Ileberi et al. (2021)

	Proposed Model	Ileberi *et al.* (2021) [1]
Model Type	Hybrid: Autoencoder + Isolation Forest + LR + RF + XGBoost + SHAP	AdaBoost + SMOTE
Feature Scope	Statistical + Behavioral (inter-transaction time, location variation)	Statistical features only (transaction attributes)
Accuracy (%)	99.95	99.67
Precision (%)	93.54	79.83

Table [4], compares our proposed hybrid model with Ileberi *et al.* (2021), showing improved performance due to the use of both statistical and behavioral features. The proposed model achieves higher accuracy than the benchmark model

V. CONCLUSION

This research introduced a Behavioral-Aware Hybrid Ensemble Model to improve credit card fraud detection by combining the strengths of both supervised and unsupervised machine learning techniques. By integrating behavior-based features with models like Autoencoder, Isolation Forest, XGBoost, Random Forest, and Logistic Regression, the system effectively captures both known and subtle fraud patterns. The use of cost-sensitive learning and explainable AI through SHAP further enhances the

model's practicality and transparency for real-world financial applications.

The model was evaluated on a widely used, highly imbalanced credit card transaction dataset and showed strong performance across key metrics such as recall, AUC-ROC, and cost ratio. These results highlight its ability to reduce false negatives while keeping the number of false positives manageable striking a balance that's essential for financial institutions.

Limitations: Like any data-driven system, this approach has some limitations. The behavioral features used are based on specific transaction patterns and may not generalize across different banks, regions, or user behaviors. Also, while unsupervised models like Autoencoder and Isolation Forest are good at detecting anomalies, they can sometimes generate false alerts when faced with entirely new patterns. And though SHAP values improve model interpretability, they can add extra computational overhead, especially in real-time settings.

There's still room for growth. Future work might focus on deploying this model in real-time fraud detection systems and exploring adaptive learning, so the model keeps up with constantly evolving fraud tactics. Adding more contextual features—like location, device ID, or transaction history—might help refine accuracy further. Finally, looking into privacy-preserving techniques like federated learning could make it possible to train fraud detection models across multiple institutions without sharing sensitive data.

REFERENCES:

[1] E. Ileberi, Y. Sun, and Z. Wang, "Performance evaluation of machine learning methods for credit card fraud detection using SMOTE and AdaBoost," *IEEE Access*, vol. 9, pp. 114716–114731, 2021, doi: 10.1109/ACCESS.2021.3134330.

[2] R. B. Sulaiman, A. Mustapha, and F. Saeed, "A review on the application of machine learning for credit card fraud detection," *International Journal of Emerging Trends in Engineering Research*, vol. 7, no. 11, pp. 580–589, 2019, doi: 10.30534/ijeter/2019/047112019.

[3] A. Dal Pozzolo, O. Caelen, R. A. Johnson, and G. Bontempi, "Calibrating probability with undersampling for unbalanced classification," in *Proc. IEEE Symposium Series on Computational Intelligence (SSCI)*, 2015, pp. 159–166, doi: 10.1109/SSCI.2015.33.

[4] Y. Şahin and E. Duman, "Detecting credit card fraud by decision trees and support vector machines," in *Proc. International MultiConference of Engineers and Computer Scientists (IMECS)*, vol. 1, 2011, pp. 442–447.

[5] J. Jurgovsky *et al.*, "Sequence classification for credit-card fraud detection," *Expert Systems with Applications*, vol. 100, pp. 234–245, 2018, doi: 10.1016/j.eswa.2018.01.037.

[6] F. K. Alarfaj, A. G. Alharbi, and A. A. Alzahrani, "A comparative study of machine learning and deep learning algorithms for credit card fraud detection," *Mathematics*, vol. 10, no. 19, p. 3662, 2022, doi: 10.3390/math10193662.

[7] R. Sailusha and B. Ramesh, "Credit card fraud detection using supervised learning algorithms," in *Proc. International Conference on Intelligent Systems and Computer Vision (ISCV)*, 2022, pp. 1–6, doi: 10.1109/ISCV55326.2022.9951456.

[8] F. Carcillo *et al.*, "SCARFF: A scalable framework for streaming credit card fraud detection with Spark," *Information Fusion*, vol. 41, pp. 182–194, 2018, doi: 10.1016/j.inffus.2017.08.003.

[9] S. Bhattacharyya, S. Jha, K. Tharakunnel, and J. C. Westland, "Data mining for credit card fraud: A comparative study," *Decision Support Systems*, vol. 50, no. 3, pp. 602–613, Feb. 2011, doi: 10.1016/j.dss.2010.08.008.

[10] U. Fiore, F. Palmieri, A. Castiglione, and A. De Santis, "Using generative adversarial networks for improving classification effectiveness in credit card fraud detection," *Information Sciences*, vol. 479, pp. 448–455, May 2019, doi: 10.1016/j.ins.2019.03.062.

[11] T. T. Nguyen, N. M. Nguyen, T. D. Nguyen, and E. H. Hwang, "Credit card fraud detection using graph neural network," in *Proc. International Conference on Big Data and Smart Computing (BigComp)*, Jeju Island, Korea, Jan. 2021, pp. 1–4, doi: 10.1109/BigComp50185.2021.00012.

[12] R. A. Bauder and T. M. Khoshgoftaar, "The effects of class imbalance techniques on the performance of deep learning models for credit card fraud detection," *Big Data*, vol. 8, no. 1, pp. 33–52, Mar. 2020, doi: 10.1089/big.2019.0129.