Smart Fraud Detection System for Online Payments Using AI and Real-Time Notifications

[1]Singam Aruna, [2]Kethavathu Srinivasa Naik, [3]Sarika Nair, [4]Magadapalli Vasavi, [4]Gorusu Sai Poornima, [4]Varri Kusuma, [4]Hemadri Sowmya, [4]Dondapati Santhi Sri

[1]Head of Department (M.Tech, Ph.D), [2]Professor, [4]BTech Student

[1,4]Department of Electronics and Communication Engineering, Andhra University College of Engineering for Women, Visakhapatnam 530 017, India
[3]Department of Electronics and Communication Engineering, Andhra University College of Engineering, Visakhapatnam 530 003, India
[2]Department of Electronics and Communication Engineering, Vignan's Institute of Information Technology, Duvvada, Visakhapatnam 530 046, India

[4]magadapallivasavi2003@gmail.com

***Abstract*—With increasing use of online transactions, there are increasing concerns for financial fraud, as many criminals and fraudsters are acting in new and better ways to affect internet users and businesses involved in electronics transactions. To put it a different way, where fraud protection methods have been slow to move forward with the comparison of then to now in the availability of sophisticated fraudsters, thus, while fraud detection methods themselves do move forward, fraudsters adapt to fraudulent detection methods by now responding to changes in transaction data that may be suspect, noticing trends in the recent transaction behaviours that may have changed, and being able to provide alerts when anomalies are detected. It is very interesting that machine learning methods can detect and push back reduce financial fraud that is so critical in society today, and so, in this paper, looks at applying machine learning techniques which are supervised methods at some of the most popular supervised learning methods with Logistic Regression, Random Forest and XGBoost, and looks at data that has been balanced using the SMOTE(Synthetic Minority Over Sampling Technique). In this development of the web application where certain methods of the machine learning can be made accessible in the app to demonstrate an application that is efficient to address a solution to detect online financial fraud via a web application built on the Flask framework, and with the added feature of email alerts automatically being sent to alert when instances occur. Experimental results demonstrate high accuracy and efficacy for this method which offers better security along with a reduced number of false positives.**

***Index Terms*—Fraud Detection, Machine Learning, AI, Real-Time Alerts, Online Transactions, SMOTE, Flask based web application, Supervised Learning.**

I. INTRODUCTION

The rise of digital transactions has increased the likelihood of financial fraud for both people and organizations alike. Cybercriminals are better than in the past, taking advantage of all the weaknesses in traditional rule-based fraud detection systems. Cybercriminals are now using more advanced techniques that traditional rule-based fraud detection systems do not consider, thus fuelling the major gaps in security.

This research work gives a proposal for a real-time fraud detection system using Artificial Intelligence (AI) and Machine Learning (ML). The model provides fraud detection through transaction features such as amount, location, and user behaviors. The model will approve over time leveraging continuous learning. The entire system is made of three components: a machine learning classification model, a Flask-based web dashboard built for transaction analysis in real-time, and an automated email alert system. Together, the three components provide on-demand fraud detection implementation and proactive response, which can secure online financial transactions.

II. LITERATURE REVIEW

Machine Learning (ML) and Artificial Intelligence (AI) have progressed fraud detection in financial transactions. Rule-based systems have proven effective in detecting fraud in transactions early in its inception. While these rule-based systems have proven effective, they have struggled to keep up with adaptive strategies of evolving fraud technologies [1]. ML-based frameworks in detecting fraud in financial transactions performs better an require a lot of labelled datasets with continuous retraining [2][3]. Unsupervised learning frameworks, such as Autoencoders and K-means, could detect unknown fraud patterns, but they also tended to return with more false positive outcomes [4]. Hybrid models, which include both supervised learning, have proven to be more effective in detecting fraud patterns, and mainly ANN, LSTM have shown promise in recognizing complex patterns. However, the required computational requirements for deep learning applications adversely affect incident detection, as it must upgrade machine learning to the next levels of ANN and LSTM [6]. Our AI fraud detection system meets these complexities through a hybrid model, continuous updates with real-time alerts, and enormous scalability [7][8][9][10].

III. MATERIALS

Programming Language: Python 3.9, selected for its extensive support in machine learning and web development.

Machine Learning Framework: The detection system uses TensorFlow for model development and inference processes, Keras to develop neural networks, and Scikit-learn to process the data, feature selection, model training and evaluation.

Data Analysis Libraries: NumPy for efficient numerical computations. Pandas was used data manipulation, preprocessing and handling large datasets. Matplotlib was used visualization of data distributions, feature importance, and model performance.

Development Environment: Jupyter Notebook (for model development, testing, visualization, iterative experimentation).

Web Technologies: The web interface for fraud detection operates through Flask as well as HTML, CSS and JavaScript.

IV. Methodology

Dataset Overview

A credit card fraud detection dataset was developed with anonymized transaction data labelled to allow for classification in a typical experimental process. This dataset was collected based on real world transactions and contained a total of 284,807 samples. Of these, only 492 samples, or about 0.173%, were fraudulent transactions and considered class 1, while every other sample was a genuine transaction and considered class 0. Hence, the proportion of samples classified as class 0 much exceeds those of class 1(a severe class imbalance). The features that are used for fraud detection task contains 28 principal components (V1-V28) obtained from PCA fro the need of confidentiality (original features were removed), and 2 original features, 'Time' and 'Amount', that capture the timing of the transaction and the amount was transacted. To address the level of class imbalance and improve the performance, preprocessing work was done which consists of scaling the features and using SMOTE to oversample the fraudulent class. This dataset has been used as the training and testing source in developing machine learning models for fraud detection in real-time.

Performance Evaluation of the Four Algorithms used

The evaluation used Logistic Regression; Random Forest; and XGBoost as main machine learning algorithms for fraud detection. Each model was trained in a SMOTE-balanced dataset and tested in real transaction data. The models were compared using major metrics such ad accuracy, precision, recall and F1.

Different algorithms used in the project are compared through Table 1.

TABLE 1 Comparison of Different Algorithms

Model	**Accuracy**	**Precision**	**Recall**	**F1-Score**
Logistic Regression	0.9734	1.00	0.97	0.99
Random Forest	0.9990	1.00	1.00	1.00
XGBoost	0.9910	1.00	0.99	0.99

The Logistic Regression model had an accuracy of 97.34%, it was a clear and interpretable model with excellent precision, and although it has a slightly less recall when compared to XGBoost, it was a fairly simple model. Overall, for an accuracy of 99.10% XGBoost had fairly strong performance. It benefited from the gradient boosting mechanism that it was able to utilize to allow it to learn better while also achieving very high precision and recall. Random Forest performed better than any of the above models with the accuracy rating at 99.90%; it also achieved perfect precision, recall, and F1-score. Random Forest was able to learn from an ensemble of decision trees and was much stronger against overfitting, it was able to deal effectively with imbalanced data and was a more robust means of dealing with data than others. Random Forest has been finally decided as the final choice to deploy for real-time fraud detection.

V. Experimental Details

System Development & Deployment

A. Flask-Based Web Application

The implementation of fraud detection system occurs as a web application, based on Flask, that takes financial transaction data and process it in real-time. The fraudulent characteristics of the data are classified by machine learning models that operate in web application and give real-time feedback.

B. Security Enhancement

The models allow users to easily adapt to complexities of fraud because they may update themselves to changing characteristics in dataset. To that end, the web application allows users to receive real-time alerts when fraud is detected, granting users and companies a better security option when completing electronic transaction tasks.

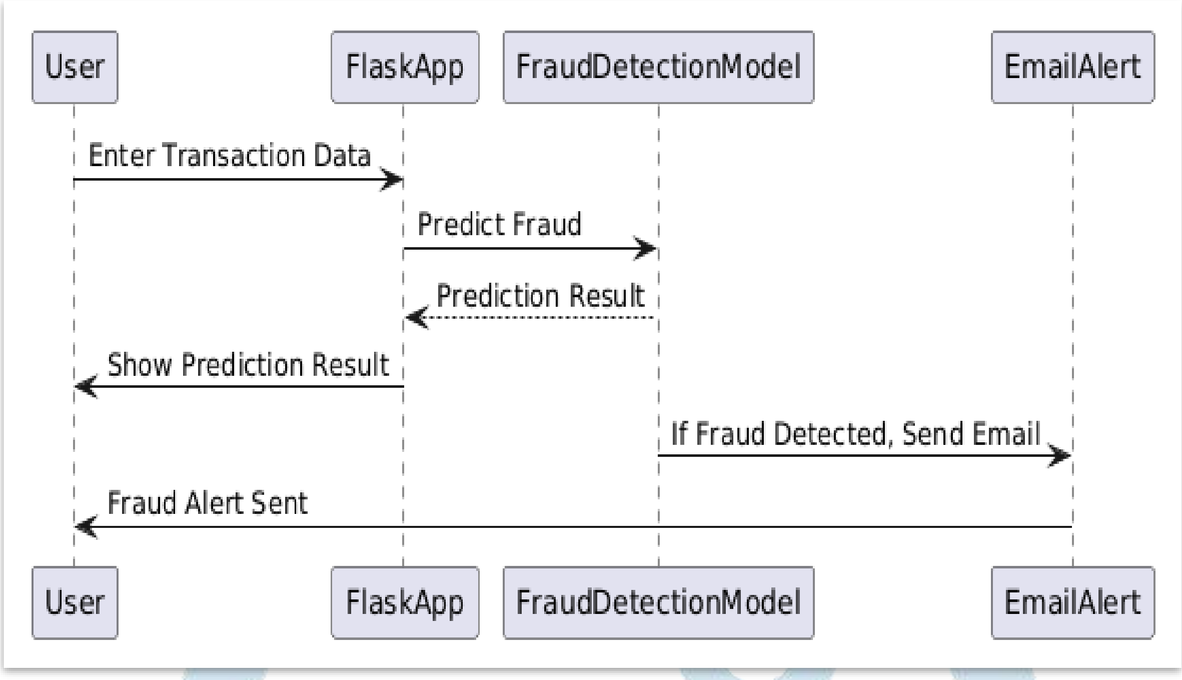


Fig. 1 — Sequence Diagram

VI. EXPERIMENTAL EVALUATION

The AI-based fraud detection system went through various rounds of testing to not only verify that it was functional, but also to ensure it was efficient, safe, and user-friendly.

A. Unit Testing

To verify that the various components of the system were working properly in isolation we tested them individually. For example, testing was conducted for email alerts as well as for machine learning supervised and unsupervised models.

B. Integration Testing

This test is conducted to test that input, fraud detection, and alerts components of system were working together, and were not creating any type of issues in the process.

C. Functional Testing

To verify that the actual fraud detection logic was properly handling legitimate and fraudulent transactions we needed to test it.

D. Performance Testing

The system tested in such a way that predictions were fast (1 second or less) and consistent, even with several transactions submitted at the same time.

E. Security Testing

The test was conducted for security vulnerabilities, such as SQL injection and XSS, and confirmed there were no vulnerabilities in the system.

E. Usability Testing

The test was conducted for responsiveness and clarity of the user interface, to make sure the system is intuitive and easy to use.

This testing did demonstrate that although the system is not only correct at identifying fraud, but also consistency by whoever is using it an ease of use is most important.

VII. Results and Discussion

A. Model Performance Comparison

Random Forest, XGBoost and Logistic Regression models were assessed in terms of metrics based on accuracy, precsion, recall, and F1-score. Random Forest and XGBoost both produced the highest accuracy at 99%, and Logistic Regression scored 97%. For precision, Random Forest had the highest score and lowered false positives, followed by XGBoost which had slightly lowered precision. Logistic Regression produced the lowest precision, resulting in increased number of false positives.

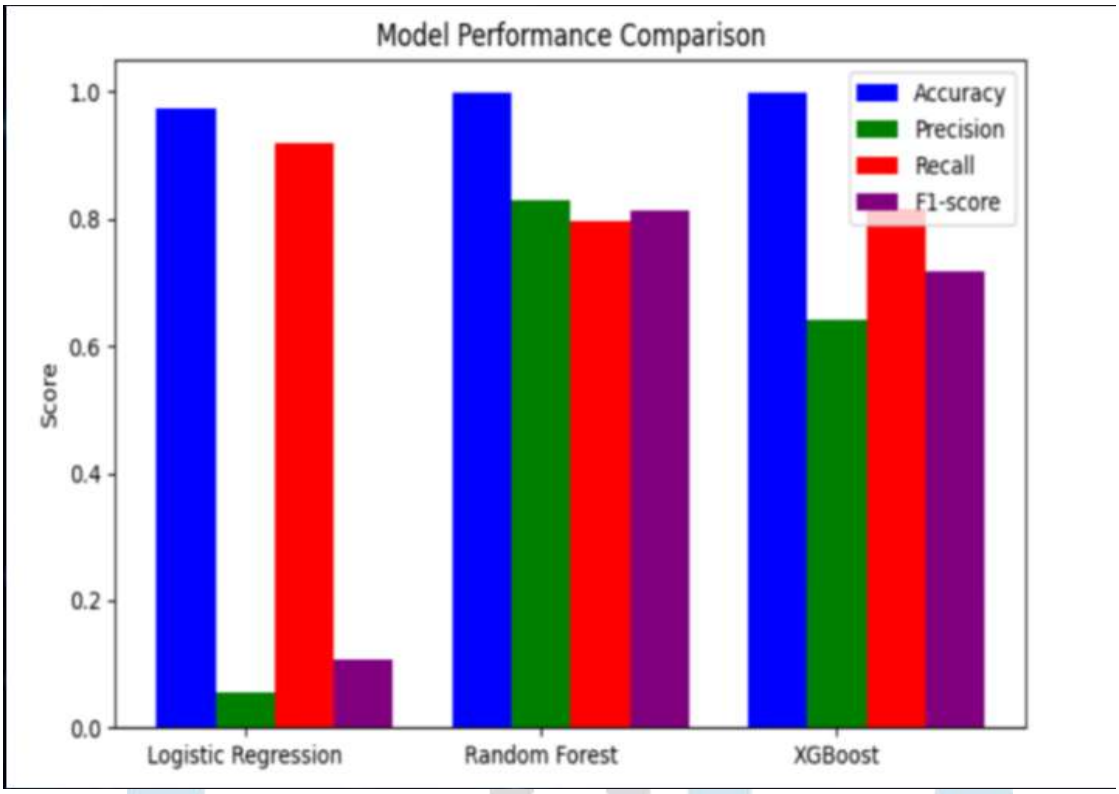
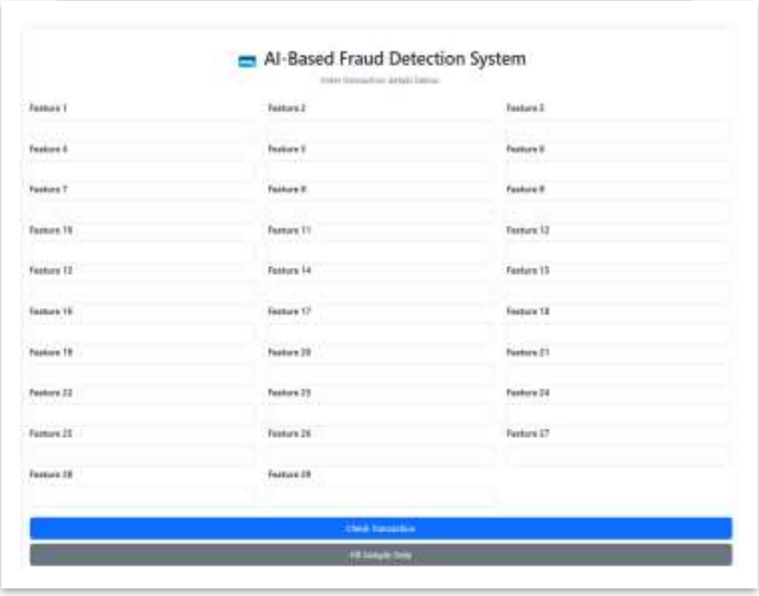


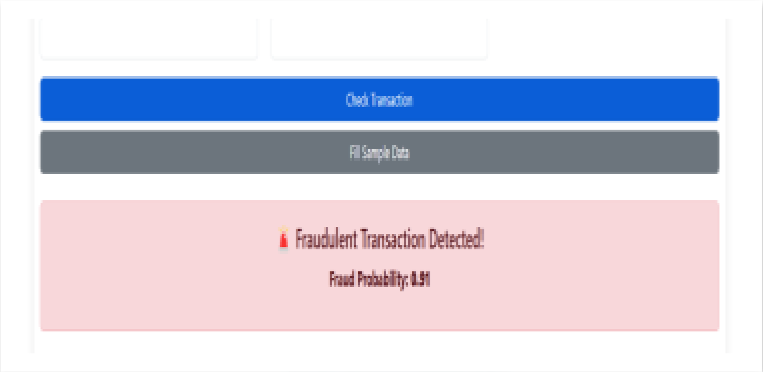
Fig. 2 — Comparing all the model performances

For recall, Logistic Regression had the highest recall even though it was a bit less precise, as it identified most of the true positives, but not as precise. Random Forest and XGBoost has lower recall but better precision than recall. For F1-svore, Random Forest and XGBoost had perfectly balanced f1 scores, while logistic regression had much lower f1 score than other two models. All three models produced similar baseline metrics; however Random Forest provided the best trade-off between accuracy, precision, recall, and F1-score.

B. System Output

With fraud detection system, which utilizes Flask framework, users can submit transaction data using a simple web interface. The web interface allows users to submit transaction data, and you, the administrator, can use the machine learning model to analyze the transaction and display the results. The system consists of three simple steps: processing data, matching to flagged transactions, and displaying the results, as detailed in Fig.3(a). If transaction is flagged, the user will receive an email notification. The system will also deliver the e-mail immediately to authorities and this is shown in Fig.4.





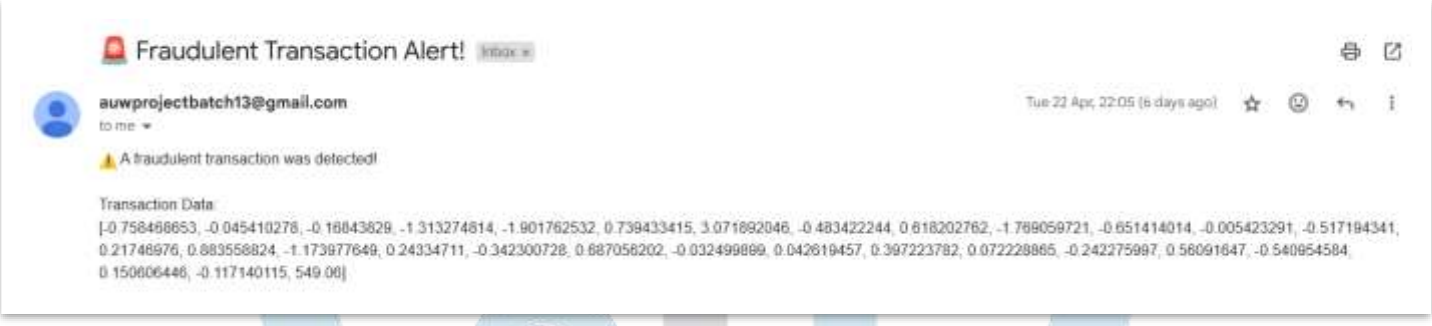
🚨 Fraudulent Transaction Detected!

Fraud Probability: 0.91

Fig. 3 — Webpage output showing the analysis of fraud detected

Figure 3(a): Web application Interface. Figure 3(b): Detected a fraud

In Fig.4, an email alert is sent to the receiver's email address when a fraud is detected.



🚨 Fraudulent Transaction Alert!

auwprojectbatch13@gmail.com

to me

Tue 22 Apr, 22:05 (6 days ago)

⚠ A fraudulent transaction was detected!

Transaction Data:
[-0.758468653, -0.045410278, -0.16843829, -1.313274814, -1.901762532, 0.739433415, 3.071892046, -0.483422244, 0.618202762, -1.769059721, -0.651414014, -0.005423291, -0.517194341, 0.21746976, 0.883558824, -1.173977649, 0.24334711, -0.342300728, 0.687056202, -0.032499899, 0.042619457, 0.397223782, 0.072228865, -0.242275997, 0.56091647, -0.540954584, 0.150606446, -0.117140115, 549.06]

Fig. 4— Email interface displaying alert message for fraud detection

The accuracy of the system is reliant on good training data to develop accuracy. Furthermore, the experimental results indicate that all machine learning methods used can detect fraud and provide reliable predictions when processing the transactions in real time.

VIII. Conclusion and Future Work

The research develops an AI fraud detection system for live e-commerce transaction checks. The system is trained to deliver low false positives while maintaining accuracy, and it learns in real-time about changing patterns of fraud with machine learning. When implemented into Flask interface, Behavior analysis is capable of conducting fast analysis and deliver real-time alerts; detecting rules violations before automatically alerting the relevant staff.

Further enhancements with deep learning and blockchain and we intend to add real-time streaming for security. It will also include support for big data sources Federated Learning and enriched with enterprise-grade mobile security features that you associate like MFA or Biometrics. This provides fraud prevention solutions that are affordable for business of any magnitude.

IX. Acknowledgement

This research work received guidance from Dr. K. Srinivasa Naik at Vignan's Institute of Information Technology. The authors thank Dr. Aruna, the Department Head of Electronics and Communication Engineering, and Dr. Rama Devi from AUCE for Women for their instrumental support and motivation.

References

[1] Rai A. K., & Dwivedi R. K. (2020). "Fraud detection in credit card data using unsupervised machine learning based scheme." 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), pp. 421-426. IEEE.

[2] Sadgali I., Nawal S. A. E. L., & Benabbou F. (2019). "Fraud detection in credit card transaction using machine learning techniques." 2019 1st International Conference on Smart Systems and Data Science (ICSSD), pp. 1-4. IEEE.

[3] Sahu A., Harshavardhan G. M., & Gourisaria M. K. (2020). "A dual approach for credit card fraud detection using neural networks and data mining techniques." 2020 IEEE 17th India Council International Conference (INDIACON), pp. 1-7. IEEE

[4] Strelcenia E., & Prakoonwit S. (2024). "Improving classification performance in credit card fraud detection by using new data augmentation." AI, 4(1), 8. doi:10.3390/ai4010008.

[5] Kumar A., & Poojitha M. (2024). "Credit card fraud detection." Proceedings of the 2024 International Conference on Intelligent System and Computing (ICISC), pp. 1-6. doi:10.1109/ICISC62624.2024.00020.

[6] Jonnalagadda V., Gupta P., & Sen E. (2019). "Detection of credit card fraud through random forest algorithm." International Journal for Research in Applied Science and Engineering Technology (IJRASET), 7(4), 1501-1506. doi:10.22214/ijraset.2019.3215.

[7] Mittal S., & Tyagi S. (2019). “Performance evaluation of machine learning algorithms of credit card fraud detection.” Performance of the 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence), pp. 320-325. doi:10.1109/CONFLUENCE.2019.8776925.

[8] Thongthawonsuwan P., & Ganokratanaa T. (2023). “Real-Time credit card fraud detection surveillance system.” Proceedings of the 2023 International Conference on Computational Intelligence (ICCI), pp. 1-6. doi:10.1109/ICCI57424.2023.10112320.

[9] Sabareesh R., Pathak D. N., Ranjan R., Prasanna R. D., Shalini P., & Bellary E. K. (2024). “AI-driven fraud detection in banking: enhancing transaction security.” Journal of Informatics Education and Research, Jan. 2024.

[10] Kousika N., Vishali G., Sunandhana S., & Vijay M. A. (2021). “Machine learning based fraud analysis and detection system.” J. Phys., Conf., 1916(1), 012115. doi:10.1088/1742-6596/1916/1/012115.