IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

UPI Fraud Detection Using Machine Learning

[1]Aishwarya Murkute, [2]Deepali Jadhav, [3]Yuvaraj Patil

[1]Student Department of Computer Science and Engineering (Data Science), [2]Professor Computer Science and Engineering (Data Science), [3]Professor Electronics and Telecommunication

[1]KIT's College of Engineering (Autonomous), Kolhapur, India

Abstract: This project proposes a UPI fraud detection model using machine learning algorithms—Random Forest (RF), Logistic Regression (LR), Decision Tree (DT), and Support Vector Machine (SVM)—to classify transactions as legitimate or fraudulent. To improve detection accuracy, Voting Classifiers are employed using different algorithm combinations, including RF+DT, RF+LR, RF+SVM, LR+DT, LR+SVM, and DT+SVM. The performance of these models is compared based on evaluation metrics such as accuracy, precision, recall, and F1-score to determine the most effective approach for fraud detection with high accuracy and minimal false positives.

Index Terms - Unified Payments Interface (UPI), Fraud Detection, Machine Learning, Random Forest (RF), Logistic Regression (LR), Decision Tree (DT), Support Vector Machine (SVM), Voting Classifiers, Financial Fraud, Digital Payments, Classification Algorithms, Ensemble Learning, Accuracy, Precision, Recall, F1-score

I. INTRODUCTION

The introduction of the Unified Payments Interface (UPI) has revolutionized digital payments in India, offering a quick, secure, and seamless way to transfer funds between accounts [8]. UPI has become the backbone of the digital payments ecosystem, facilitating transactions through smartphones and digital banking systems. By enabling direct bank-to-bank transfers and providing a single interface for various payment systems, UPI eliminates the need for multiple banking apps, making digital payments more accessible and efficient [8]. With its growing adoption by individuals and businesses, UPI has significantly contributed to the shift toward a cashless economy.

However, as the usage of UPI continues to rise, so does the risk of fraudulent activities [10]. Cybercriminals exploit the ease of transactions and the widespread use of mobile devices to commit various forms of fraud, including phishing, unauthorized transactions, and social engineering attacks [10]. Phishing schemes trick users into revealing sensitive information such as OTPs or login credentials, which are then used for unauthorized transfers. Malicious software and fraudulent apps can compromise user devices, allowing attackers to gain access to UPI credentials and execute transactions without consent. Additionally, fraudsters often employ social engineering techniques, manipulating users into disclosing confidential details or authorizing payments under false pretences.

These fraudulent activities pose significant risks to both individual users and the financial system at large. Detecting fraud in real-time is crucial to mitigating financial losses and ensuring the security of digital transactions. However, traditional fraud detection methods often rely on predefined rules, which may not be effective against evolving fraud tactics [1][3]. Given the dynamic nature of fraudulent activities and the vast number of transactions processed daily, there is a pressing need for advanced fraud detection models capable of identifying emerging fraud patterns with high accuracy.

Machine learning (ML) has emerged as a powerful tool for fraud detection, offering the ability to analyse large volumes of transaction data and detect patterns indicative of fraudulent behaviour [2][6][10]. Unlike

rule-based systems, ML algorithms can adapt to new fraud trends by learning from historical data, making them more effective in real-time fraud detection. The ability of ML models to identify hidden relationships between features enables them to recognize complex fraud patterns and improve classification accuracy [2][4].

In this project, we propose a UPI fraud detection model utilizing machine learning techniques. Specifically, we investigate four widely used classification algorithms—Random Forest (RF), Logistic Regression (LR), Decision Tree (DT), and Support Vector Machine (SVM)—to classify transactions as legitimate or fraudulent based on transaction attributes. To enhance detection performance, we further employ ensemble learning through Voting Classifiers, combining different algorithm pairs, including RF+DT, RF+LR, RF+SVM, LR+DT, LR+SVM, and DT+SVM. The performance of these models is evaluated based on key metrics such as accuracy, precision, recall, and F1-score to identify the most effective fraud detection approach with minimal false positives.

II. RESEARCH METHODOLOGY

2.1 System Overview

The growing popularity of the Unified Payments Interface (UPI) in India has revolutionized the digital payments landscape, offering a fast, convenient, and secure platform for transferring money across banks. However, with increased adoption comes a corresponding rise in fraudulent activities, making the need for robust fraud detection systems more urgent than ever [10][6]. UPI frauds take various forms, including phishing, unauthorized transactions, and the exploitation of malicious software to access user accounts. These fraudulent activities threaten the integrity and security of the platform, undermining user confidence and risking significant financial losses.

To address these challenges, this project proposes a **machine learning-based fraud detection system** that classifies UPI transactions as legitimate or fraudulent based on patterns identified in transaction data. By analysing historical transaction records, the system can detect anomalies and flag potentially fraudulent transactions in real time. Machine learning provides the advantage of continuous learning, allowing the system to adapt to emerging fraud patterns and improve detection accuracy over time [1][2][6].

2.2 Key Components of the Fraud Detection System

2.2.1. Data Collection and Feature Extraction

A high-quality dataset is essential for training an effective fraud detection system [2][4]. Transaction data will be collected from UPI platforms, including both legitimate and fraudulent transactions. The dataset will include features such as:

- **Transaction UPID:** A unique identifier for each transaction.
- **Transaction Amount:** The monetary value of the transaction.
- **Transaction Time:** The timestamp when the transaction was made.

These features will be analysed to understand patterns that differentiate normal transactions from fraudulent ones.

2.2.2. Data Preprocessing

Before feeding the data into the machine learning models, preprocessing is required to enhance data quality and ensure effective model training. Key preprocessing steps include:

- **Data Cleaning:** Handling missing values, removing duplicate records, and eliminating outliers.
- **Data Transformation:** Converting categorical variables (e.g., transaction type) into numerical representations for model compatibility.
- **Normalization:** Scaling numerical features (e.g., transaction amounts) to ensure consistency, particularly for models like Logistic Regression and SVM, which are sensitive to feature scaling [4].
- **Feature Selection:** Identifying the most relevant features for fraud detection to improve model efficiency and accuracy [4].

After preprocessing, the dataset will be split into training and testing sets to evaluate model performance. Training dataset will contribute 80% of the data and remaining 20% will be used for testing.

2.2.3. Machine Learning Model Development

The core of the fraud detection system lies in machine learning algorithms that classify transactions. Four widely used classification algorithms will be employed:

2.2.3.1 Random Forest (RF)

An ensemble learning method that builds multiple decision trees and outputs the majority class prediction. Known for its high accuracy, robustness, and ability to handle complex datasets, Random Forest also provides insights into feature importance, helping identify key fraud indicators [6][9].

2.2.3.1 Logistic Regression (LR)

A linear model used for binary classification that predicts the probability of a transaction being fraudulent. Despite its simplicity, Logistic Regression is effective when relationships between features and fraud likelihood are relatively linear [2].

2.2.3.1 Decision Tree (DT)

A tree-based algorithm that recursively splits data based on feature values. It is intuitive and interpretable but prone to overfitting, which can be mitigated with pruning techniques [6].

2.2.3.1 Support Vector Machine (SVM)

A supervised learning algorithm that finds the optimal hyperplane to separate legitimate and fraudulent transactions. SVM performs well in high-dimensional spaces and is effective for complex fraud patterns [5].

2.2.3.1 Ensemble Learning (Voting Classifiers)

To enhance performance, the system also employs **Voting Classifiers**, combining different models to improve fraud detection [9]. The following model combinations will be tested:

1. **RF + DT**
2. **RF + LR**
3. **RF + SVM**
4. **LR + DT**
5. **LR + SVM**
6. **DT + SVM**

Each model will be evaluated using key performance metrics such as **accuracy, precision, recall, F1-score** with the goal of achieving a balance between detecting fraudulent transactions and minimizing false positives.

2.2.4. System Deployment as a Web-Based Application

Once the machine learning model is trained and optimized, it will be **deployed as a web-based application using Flask** [7]. The application will include two main components:

- **User Interface (UI):** A simple, intuitive web interface that allows users and financial institutions to interact with the fraud detection system. The UI will provide real-time feedback on transactions, indicating whether they are flagged as fraudulent.
- **Backend:** A Flask-powered backend that processes incoming transaction data, applies the trained fraud detection model, and returns the results to the UI.

This web-based implementation ensures that the fraud detection system is accessible and scalable, allowing for seamless integration into existing financial platforms.

III. RESULTS AND DISCUSSION

The performance of the proposed UPI fraud detection system was evaluated using multiple machine learning models [6][10], including Random Forest (RF), Logistic Regression (LR), Decision Tree (DT), Support Vector Machine (SVM), and various Voting Classifiers. The models were assessed using standard evaluation metrics such as Accuracy, Precision, Recall, and F1-Score to determine their effectiveness in detecting fraudulent transactions. The results are summarized in table 3.1.

3.1. Performance Evaluation of Individual Models

Among the individual models, **Random Forest (RF) achieved the highest accuracy of 0.92**, outperforming all other classifiers. It also demonstrated a strong balance between precision (0.91) and recall (0.96), leading to the highest F1-score (0.93). These results indicate that RF is the most effective model for fraud detection, as it efficiently learns patterns and generalizes well to new data. The confusion matrix for random forest is as in fig. 31

Decision Tree (DT) also exhibited strong performance, achieving an accuracy of 0.91, with precision and recall values of 0.92 and 0.93, respectively. However, DT models are prone to overfitting, which could impact their ability to handle real-world transaction variations. The confusion matrix for decision tree is as in fig. 3.2

Logistic Regression (LR) performed the worst among all models, with an accuracy of 0.78. While its precision was relatively high at 0.89, its recall value of 0.71 indicates that it failed to correctly classify a significant number of fraudulent transactions. This result suggests that LR is not well-suited for fraud detection in high-dimensional and complex datasets [2][3], where non-linear patterns dominate. The confusion matrix for logistic regression is as in fig. 3.3

SVM, which is often used for classification problems, showed moderate performance with an accuracy of 0.84. While it had a precision of 0.87 and recall of 0.86, its overall performance was lower compared to ensemble-based models. This suggests that SVM alone may not be optimal for real-time fraud detection, especially when handling large-scale transaction data. The confusion matrix for support vector machine forest is as in fig.3.4

The performance model for individual models is as shown in fig 3.5

3.2. Performance of Voting Classifiers

To further improve classification accuracy, Voting Classifiers combining multiple models were tested. The highest accuracy among these combinations was 0.91, observed in RF+DT (fig.3.6), RF+LR (fig.3.7), and DT+SVM (fig.3.11). These classifiers demonstrated strong precision and recall, with an F1-score of 0.93, matching the performance of the standalone Decision Tree model [9].

The RF+SVM (fig.3.8) classifier, while maintaining a high recall (0.95), had a slightly lower precision (0.89), leading to an accuracy of 0.90. This indicates that the combination of RF and SVM tends to favor recall over precision, meaning it is more likely to detect fraudulent transactions but may also generate more false positives. The LR+DT (fig 3.9) classifier, has the lowest accuracy.

The LR+SVM (fig.3.10) classifier recorded the lowest accuracy among Voting Classifiers at 0.82, confirming that combining two relatively weaker models does not necessarily result in improved fraud detection performance. The performance comparison for voting classifiers is as displayed in fig 3.12

3.3. Best Model Selection and Practical Implications

Based on the accuracy, precision, recall, and F1-score, Random Forest (RF) was identified as the best-performing model for UPI fraud detection, see fig 3.5. While some Voting Classifiers (RF+DT, RF+LR, DT+SVM) performed competitively (accuracy: 0.91), they did not exceed RF's performance, see fig 3.12. Additionally, ensemble methods introduce additional computational complexity [9], which could impact real-time fraud detection capabilities.

Given these observations, as per table 3.1, RF is the most suitable model for real-world deployment due to its high accuracy, strong recall, and robustness in identifying fraudulent transactions with minimal false positives. The scalability of RF further strengthens its applicability in financial systems processing millions of transactions daily.

Future improvements may include incorporating deep learning-based anomaly detection, integrating real-time feature engineering techniques, and adapting the fraud detection model using continuous learning methods to counter evolving fraud tactics.

Mean Squared Error (MSE) is a common metric used to evaluate the performance of a regression model, but it can also be used to assess classification models when probabilities or confidence scores are available. The formula for MSE is:

$$\text{MSE}=\frac{1}{n}\sum_{i=1}^{n}(y_i-\hat{y}_i)^2$$

where:

y_i= Actual (true) value of the transaction class (fraud or legitimate, typically represented as 0 or 1)

$\hat{y}_i$= Predicted probability or output of the model

n= Total number of data points (transactions)

∑= Summation over all observations

MSE can be estimated as below and is displayed for each model in table 3.1:
MSE=1−Accuracy

Model	Accuracy	Precision	Recall	F1-Score	MSE
Random Forest (RF)	**0.92**	0.91	**0.96**	**0.93**	**0.08**
Logistic Regression (LR)	0.78	0.89	0.71	0.79	0.22
Decision Tree (DT)	0.91	**0.92**	0.93	**0.93**	0.09
Support Vector Machine (SVM)	0.84	0.87	0.86	0.87	0.16
Voting Classifier (RF + DT)	0.91	**0.93**	0.91	0.92	0.09
Voting Classifier (RF + LR)	0.91	**0.93**	0.91	0.92	0.09
Voting Classifier (RF + SVM)	0.9	0.89	**0.95**	0.92	0.1
Voting Classifier (LR + DT)	0.91	0.92	0.93	**0.93**	0.09
Voting Classifier (LR + SVM)	0.82	0.89	0.8	0.84	0.18
Voting Classifier (DT + SVM)	0.91	0.92	0.93	**0.93**	0.09

Table 3.1 Performance metrics of machine learning models

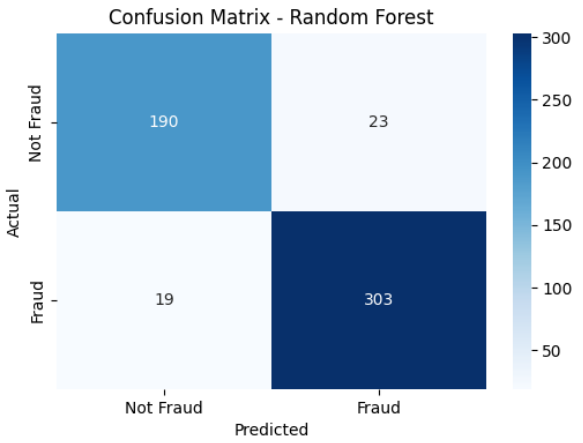

Fig. 3.1 Confusion Matrix- Random Forest

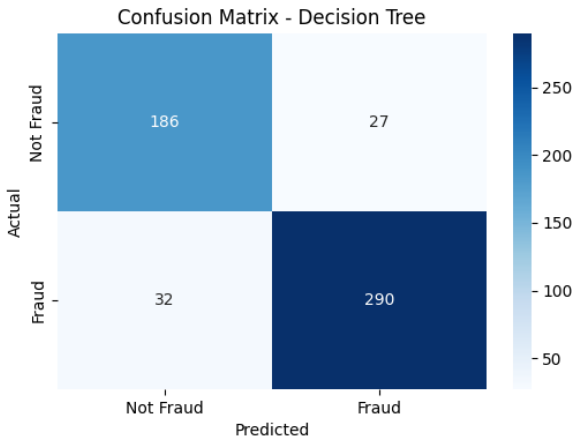

Fig. 3.2 Confusion Matrix- Decision Tree

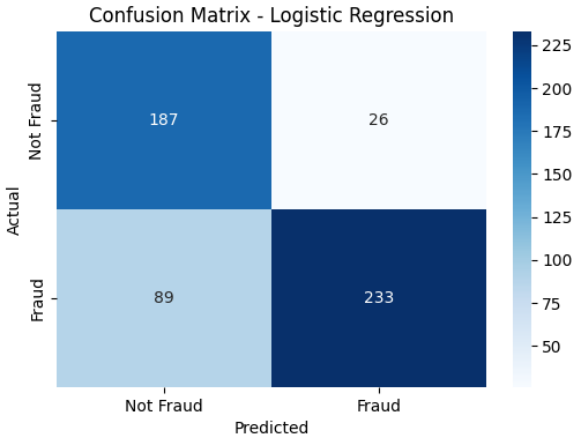

Fig. 3.3 Confusion Matrix- Logistic Regression

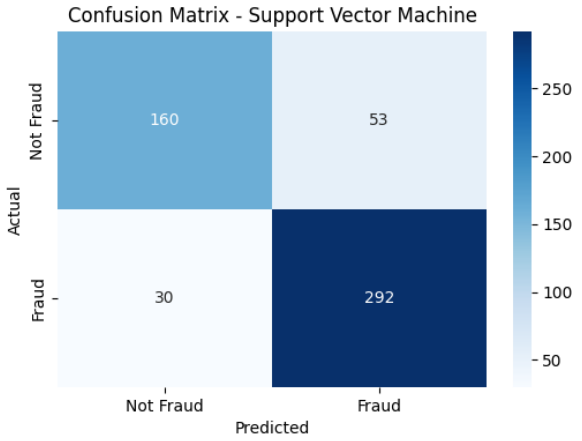

Fig. 3.4 Confusion Matrix- Support Vector Machine

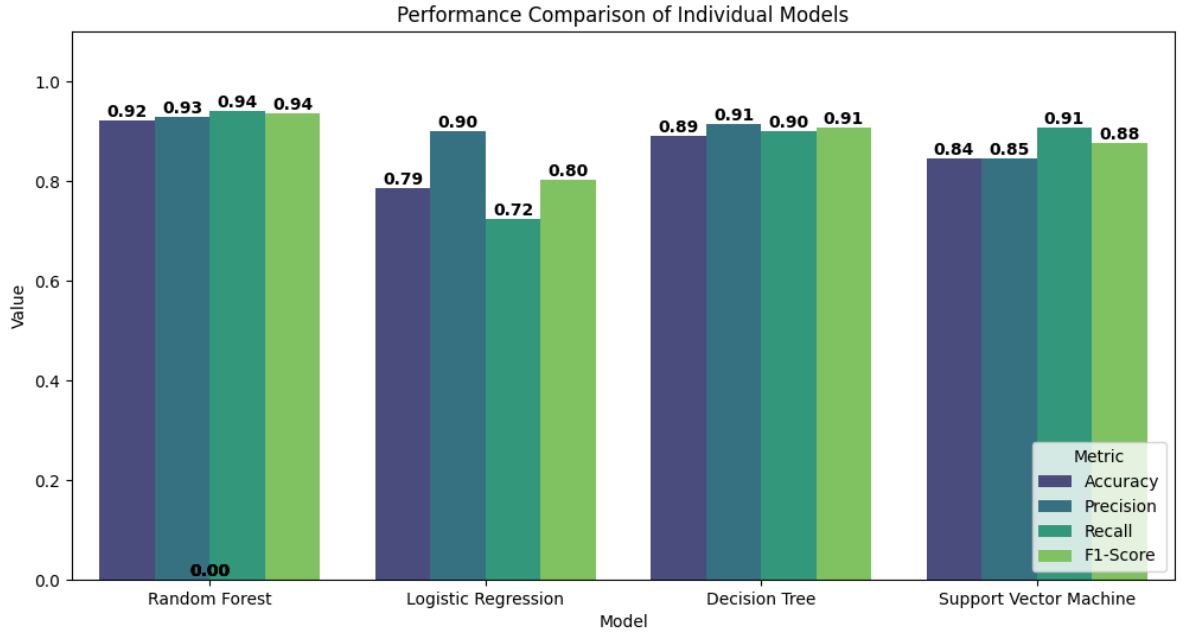

Fig. 3.5 Performance Comparison of Individual Models

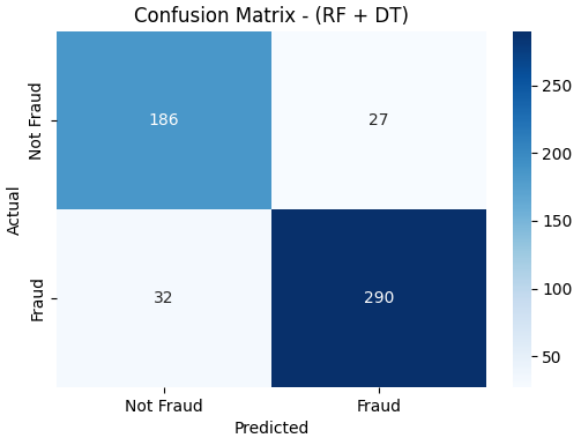

Fig. 3.6 Confusion Matrix- RF + DT

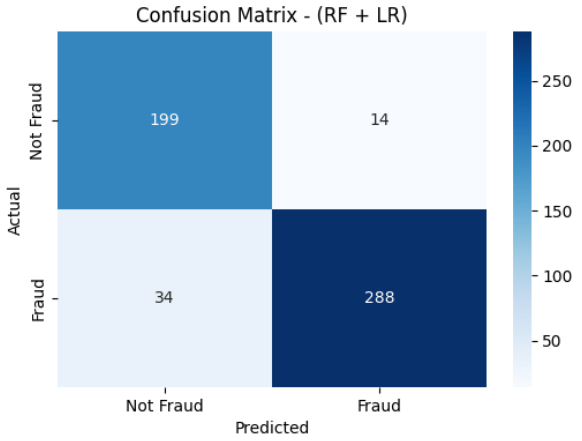

Fig. 3.7 Confusion Matrix- RF + LR

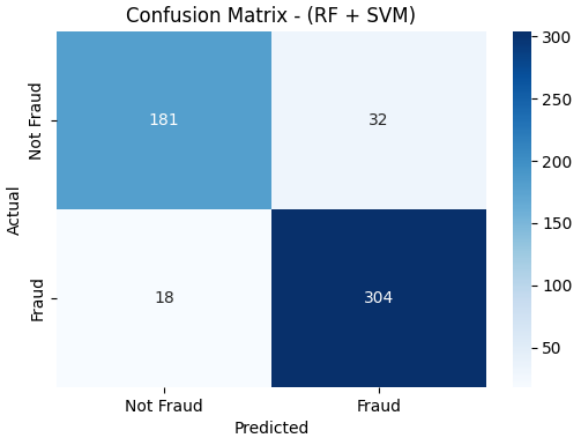

Fig. 3.8 Confusion Matrix- RF + SVM

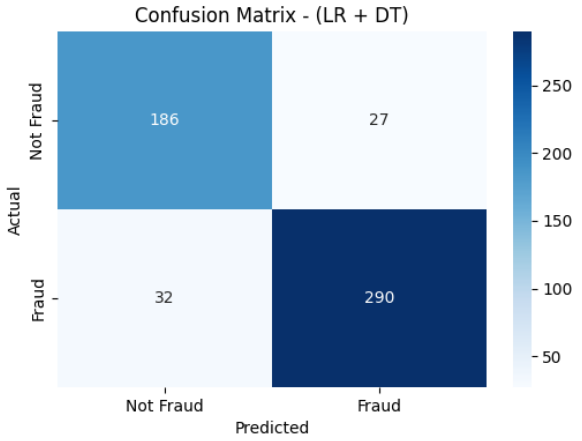

Fig.3.9 Confusion Matrix- LR + DT

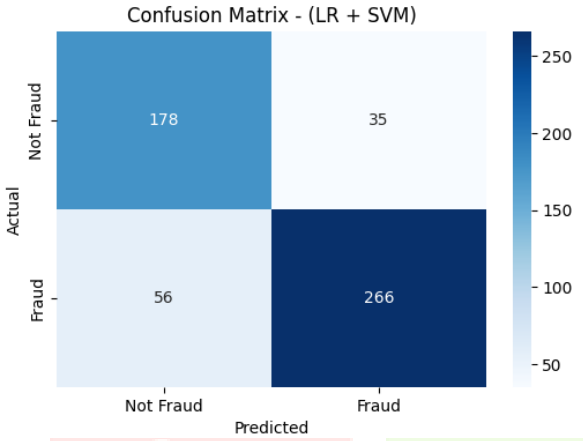

Fig. 3.10 Confusion Matrix- LR + SVM

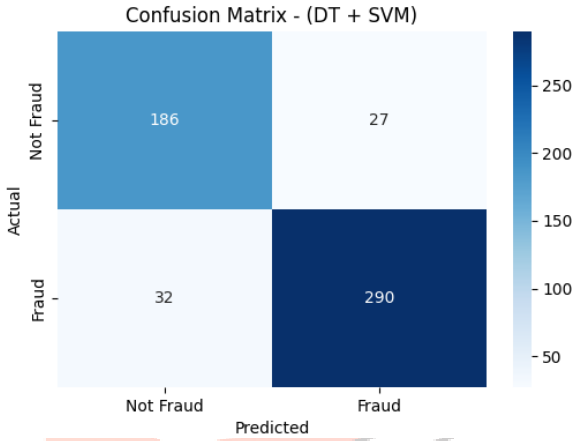

Fig. 3.11 Confusion Matrix- DT +SVM

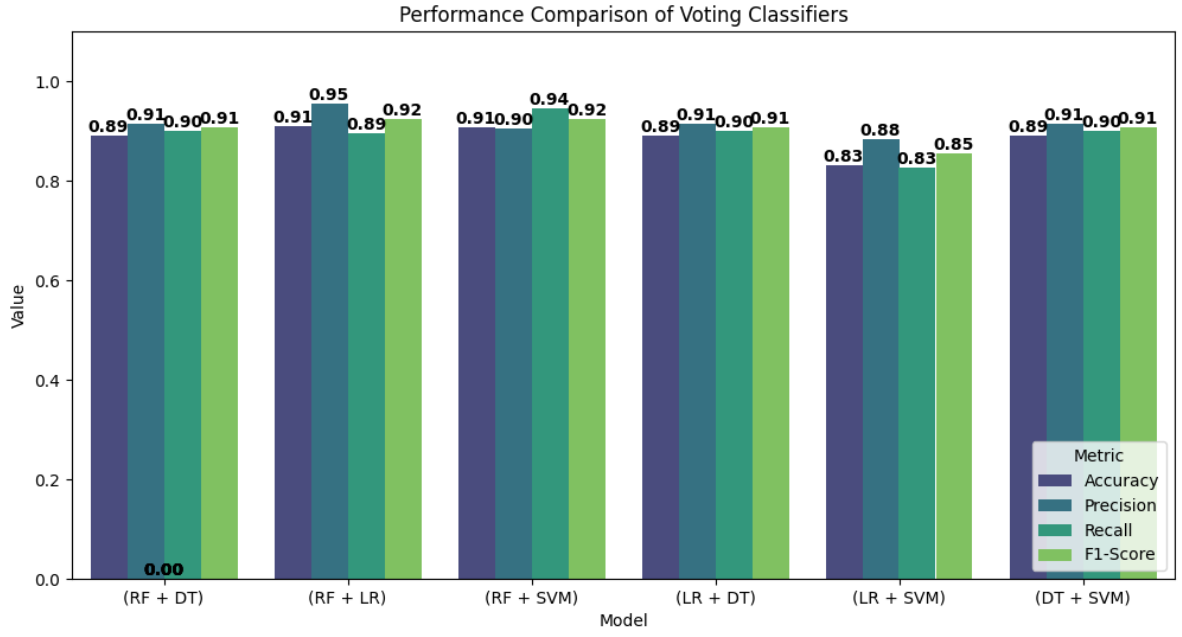

Fig. 3.12 Performance Comparison of Voting Classifiers

VI. CONCLUSION

The implementation of a UPI fraud detection system using machine learning algorithms demonstrates the effectiveness of automated fraud classification in digital financial transactions [6][10]. The study evaluated Random Forest, Logistic Regression, Decision Tree, Support Vector Machine (SVM), and various Voting Classifier combinations based on key performance metrics such as accuracy, precision, recall, and F1-score.

Among all models tested, Random Forest emerged as the best-performing algorithm with the highest accuracy of 0.92. It also achieved a precision of 0.91, recall of 0.96, and an F1-score of 0.93, making it a highly reliable choice for fraud detection. While the Decision Tree (accuracy: 0.91) and Voting Classifiers (RF+DT, RF+LR, DT+SVM at 0.91) also showed strong performance, Random Forest's ability to balance detection accuracy and false positive reduction makes it the most effective model.

The integration of the fraud detection model into a Flask-based web application ensures scalability and real-time monitoring of transactions. This enables users and financial institutions to detect fraudulent activity proactively, reducing financial risks while maintaining user trust [10].

Overall, this project highlights the crucial role of machine learning in securing digital payment systems. As fraud techniques evolve, future enhancements could involve deep learning models, real-time anomaly detection, and adaptive learning systems [1][6] to further improve fraud detection accuracy and responsiveness.

V. REFERENCES

1. Ahmed, M., Mahmood, A. N., & Hu, J. (2016). A survey of network anomaly detection techniques. Journal of Network and Computer Applications, 60, 19–31. https://doi.org/10.1016/j.jnca.2015.11.016

2. Bhattacharyya, S., Jha, S., Tharakunnel, K., & Westland, J. C. (2011). Data mining for credit card fraud: A comparative study. Decision Support Systems, 50(3), 602–613. https://doi.org/10.1016/j.dss.2010.08.008

3. Buczak, A. L., & Guven, E. (2015). A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Communications Surveys & Tutorials, 18(2), 1153–1176. https://doi.org/10.1109/COMST.2015.2494502

4. Chandrashekar, G., & Sahin, F. (2014). A survey on feature selection methods. Computers & Electrical Engineering, 40(1), 16–28. https://doi.org/10.1016/j.compeleceng.2013.11.024

5. Cortes, C., & Vapnik, V. (1995). Support-vector networks. Machine Learning, 20(3), 273–297. https://doi.org/10.1007/BF00994018

6. Dal Pozzolo, A., Boracchi, G., Caelen, O., Alippi, C., & Bontempi, G. (2017). Credit card fraud detection: A realistic modeling and a novel learning strategy. IEEE Transactions on Neural Networks and Learning Systems, 29(8), 3784–3797. https://doi.org/10.1109/TNNLS.2017.2736643

7. Flask Documentation. (2024). Flask: Web development, one drop at a time. Flask Project. https://flask.palletsprojects.com/

8. National Payments Corporation of India (NPCI). (2023). Unified Payments Interface (UPI) – Product Overview. https://www.npci.org.in/what-we-do/upi/product-overview

9. Rokach, L. (2010). Ensemble-based classifiers. Artificial Intelligence Review, 33(1), 1–39. https://doi.org/10.1007/s10462-009-9124-7

10. Ryman-Tubb, N. F., Krause, P., & Garn, W. (2018). How artificial intelligence and machine learning research impacts payment card fraud detection n: A survey and industry benchmark. Computers & Security, 81, 963–981. https://doi.org/10.1016/j.cose.2018.09.010