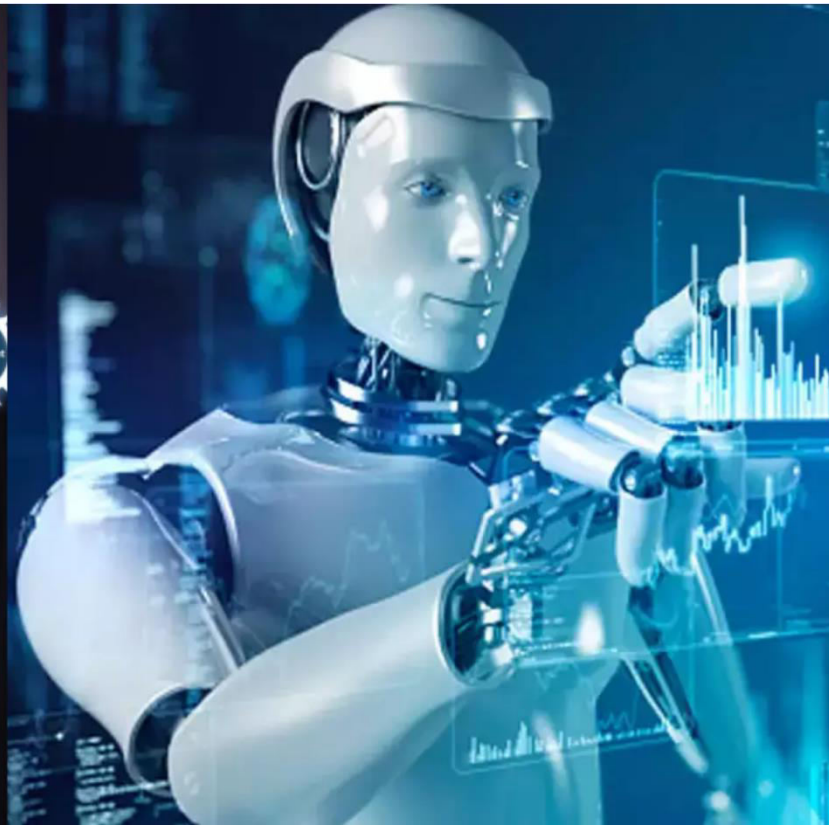
IJIRSET
International Journal of Innovative Research in Science | Engineering | Technology

ISSN(O): 2319-8753
ISSN(P): 2347-6710

International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.699

Volume 15, Issue 3, March 2026

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 3, March 2026

|DOI: 10.15680/IJIRSET.2026.1503297|

Online Payment Fraud Detection Using Machine Learning

Mr. Y. Siva Naga Raju M.Tech[1], Yeruva Nagarjuna Reddy[2], Sunkara Siva Sruthi[3], Gollakummari Sivagopiraju[4], Chennupalli Omkarmanikanta[5]

Assistant Professor, Kallam Haranadhareddy Institute of Technology, Guntur, Andhra Pradesh, India [1]

U.G Students, Department of CSE (Data Science), Kallam Haranadhareddy Institute of Technology, Guntur, Andhra Pradesh, India[2-5]

ABSTRACT: Online payment fraud has become one of the most critical threats in the digital financial ecosystem, resulting in billions of dollars in annual losses for businesses and consumers worldwide. Traditional rule-based fraud detection systems often generate excessive false positives and fail to adapt to evolving fraud patterns. This study presents a machine learning-based approach to detect fraudulent online payment transactions in real time. A comprehensive dataset of over 6.3 million transaction records sourced from Kaggle was analyzed using five classification algorithms — Random Forest, Decision Tree, ExtraTrees, Support Vector Machine (SVM), and XGBoost. After thorough data preprocessing, exploratory data analysis, and model evaluation, the best-performing model was deployed as a Flask-based web application that classifies transactions as FRAUD or NOT FRAUD through an intuitive user interface. The proposed system achieves up to 79% classification accuracy and demonstrates the effectiveness of machine learning in securing online payment platforms.

KEYWORDS: Online Payment Fraud Detection, Machine Learning, Random Forest, Support Vector Machine, Flask Web Application, Data Preprocessing, Classification Algorithms, Fraud Classification, Real-time Detection, Feature Engineering.

I. INTRODUCTION

Online payment fraud is one of the fastest-growing financial crimes in the digital era. As e-commerce platforms, digital wallets, and internet banking have expanded globally, fraudulent transactions have simultaneously increased in sophistication and frequency. Financial institutions, payment gateway operators, and individual consumers bear the consequences of fraudulent activities including direct financial losses, reputational damage, and erosion of consumer trust in digital commerce.

Traditional fraud detection systems rely primarily on rule-based filters and manual review processes. While these approaches provide a basic level of protection, they are inherently reactive — they respond to known fraud patterns rather than anticipating new ones. As fraudsters continuously evolve their techniques, rule-based systems struggle to keep pace, often generating excessive false positives that burden fraud review teams and create friction for legitimate customers.

Machine learning offers a fundamentally different approach. By training models on large historical transaction datasets, machine learning algorithms can identify complex, non-linear relationships between transaction features that are predictive of fraud. Unlike static rule engines, ML models can adapt to new data and improve over time. This study leverages a Kaggle dataset of 6.3 million online payment records to train and compare multiple classification algorithms and deploy the best-performing model as a real-time fraud detection web application.

II. PROBLEM STATEMENT

Online payment fraud causes significant financial harm to businesses, payment processors, and consumers. Existing fraud detection systems are often based on fixed rule sets that cannot adapt to new fraud patterns. These systems

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 3, March 2026

|DOI: 10.15680/IJIRSET.2026.1503297|

generate high false positive rates, causing legitimate transactions to be incorrectly flagged and disrupting the user experience for genuine customers.

The core challenge is developing an automated, adaptive system capable of analyzing multiple transaction features simultaneously — including transaction amount, type, and account balance changes — to accurately classify transactions as fraudulent or legitimate in real time. Without proper machine learning models, it becomes difficult to identify the subtle behavioral patterns that distinguish fraud from legitimate activity in large-scale transaction datasets. This study addresses this problem by applying machine learning classification algorithms to detect online payment fraud, comparing their performance, and deploying the best model through an accessible Flask web application that enables real-time fraud prediction.

III. LITERATURE SURVEY

Previous studies have extensively explored the application of machine learning and data analytics for fraud detection in financial systems. Researchers have found that ensemble methods, anomaly detection approaches, and supervised classification algorithms are particularly effective for identifying fraudulent transactions in large datasets.

A. Traditional Fraud Detection Approaches

Early fraud detection systems relied on rule-based engines that flagged transactions exceeding predefined thresholds. These systems, while simple to implement, could not handle the complexity of modern fraud patterns and generated high false positive rates that overwhelmed fraud review teams.

B. Machine Learning for Fraud Detection

Supervised learning approaches including Decision Trees, Random Forests, and Support Vector Machines have demonstrated strong performance in binary fraud classification tasks. These models learn from labeled historical transaction data and can generalize to new unseen transactions, providing a significant improvement over static rule engines.

C. Ensemble Methods in Financial Fraud Detection

Ensemble methods such as Random Forest and XGBoost aggregate predictions from multiple base learners to produce more robust and accurate classifications. These methods are particularly well-suited for imbalanced datasets — a common characteristic of fraud data where fraudulent transactions represent a small minority of total records.

D. Feature Engineering for Transaction Data

Research highlights the importance of feature engineering in improving fraud detection accuracy. Key features such as transaction amount, account balance differentials, transaction type, and behavioral patterns are critical predictors. Proper feature scaling and categorical encoding are essential preprocessing steps that significantly impact model performance.

E. Deep Learning Approaches

Recent studies have explored deep learning architectures including LSTM networks and Autoencoders for fraud detection in sequential transaction data. While these approaches show promise for capturing temporal dependencies, they require significantly more computational resources and larger datasets compared to traditional ML classifiers.

F. Research Gap Identified

From the existing studies, the following research gaps are identified:

- Many studies focus on fraud detection model accuracy without providing an accessible deployment solution for real-time prediction.
- Limited research directly compares five or more classification algorithms on the same fraud detection dataset with comprehensive evaluation metrics.
- Few studies integrate model training, evaluation, and deployment into a complete end-to-end web application that non-technical users can operate.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 3, March 2026

|DOI: 10.15680/IJIRSET.2026.1503297|

The proposed study addresses these gaps by comparing five classification algorithms on a large-scale fraud dataset and deploying the best model as a Flask web application that enables real-time FRAUD or NOT FRAUD classification through a user-friendly interface.

Study	Technique Used	Focus Area	Limitation
Study A	Rule-Based Filtering	Threshold-based transaction flagging	Cannot adapt to new fraud patterns
Study B	Decision Tree Classifier	Binary fraud classification	Prone to overfitting on imbalanced data
Study C	Random Forest & XGBoost	Ensemble-based fraud detection	Requires large datasets for training
Study D	Support Vector Machine	High-dimensional transaction classification	Slow training on large-scale datasets
Study E	Deep Learning (LSTM)	Sequential fraud pattern detection	High computational cost, limited deployment

Table: Comparison of Existing Fraud Detection Studies

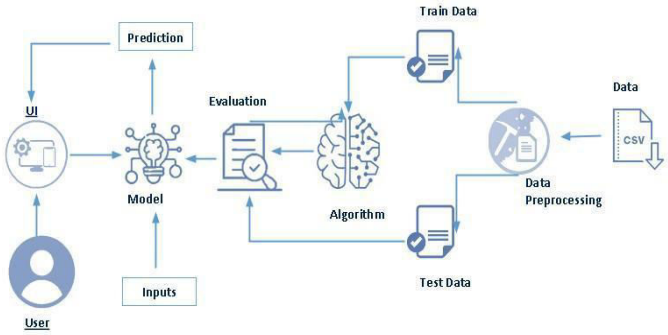


Fig: Proposed System Architecture

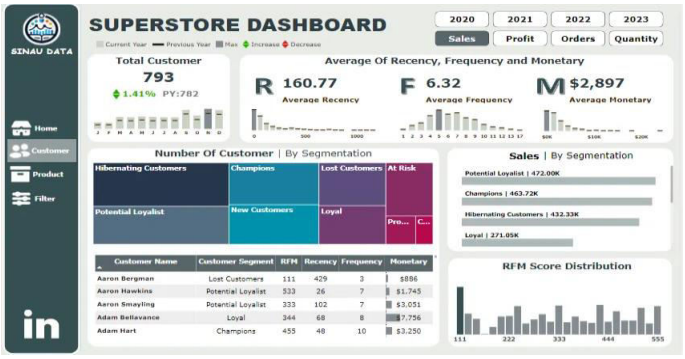


Fig: Proposed ML Workflow for Fraud Detection

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 3, March 2026

|DOI: 10.15680/IJIRSET.2026.1503297|

IV. PROPOSED METHODOLOGY

A. Data Collection

The dataset used in this study is the "PS_20174392719_1491204439457_logs.csv" sourced from Kaggle (https://www.kaggle.com/datasets/rupakroy/online-payments-fraud-detection-dataset). The dataset contains over 6.3 million online payment transaction records and includes the following key features:

- step — Unit of time (1 step = 1 hour of simulated time)
- type — Type of transaction: CASH-IN, CASH-OUT, DEBIT, PAYMENT, TRANSFER
- amount — Transaction amount in local currency
- nameOrig — Customer ID initiating the transaction
- oldbalanceOrg — Initial balance of sender before the transaction
- newbalanceOrig — Sender balance after the transaction
- nameDest — Recipient customer ID
- oldbalanceDest — Initial balance of recipient before transaction
- newbalanceDest — Recipient balance after transaction
- isFraud — Target variable: 1 = Fraudulent, 0 = Legitimate

B. Data Preprocessing

The raw dataset underwent several preprocessing steps before model training:

- Removing irrelevant identifier columns (nameOrig, nameDest) that do not contribute to model learning
- Checking for null and missing values using isnull().sum() — the dataset was confirmed to have zero null values
- Encoding categorical variables: the "type" column was encoded using LabelEncoder from scikit-learn
- Identifying and treating outliers in the "amount" column using boxplots and transformation plots
- Splitting the dataset into 80% training and 20% testing sets using train_test_split()
- Standardizing feature values using Feature Scaling to ensure consistent model input

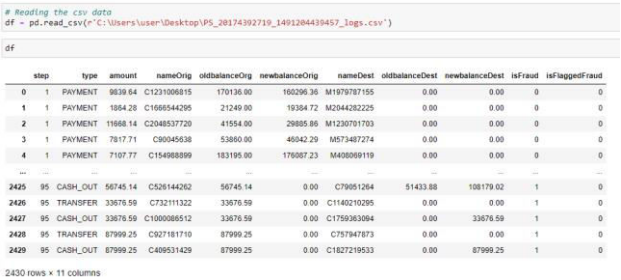
C. System Architecture

The system architecture of the proposed fraud detection system consists of three integrated layers that transform raw transaction data into real-time fraud predictions.

First, the transaction dataset is collected from Kaggle and stored as a CSV file. The data undergoes preprocessing including null value checks, irrelevant column removal, label encoding of categorical variables, and outlier treatment. The cleaned dataset is then used for model training.

Five machine learning classification algorithms — Random Forest, Decision Tree, ExtraTrees, SVM, and XGBoost — are trained and evaluated on the preprocessed dataset. Model performance is measured using accuracy, confusion matrix, and classification reports. The best-performing model is serialized using Pickle and saved as model.pkl for deployment.

The deployment layer consists of a Flask web application with three HTML pages (home.html, predict.html, submit.html). When a user submits transaction details through the web interface, the Flask backend loads the saved model, preprocesses the input, and returns a real-time FRAUD or NOT FRAUD prediction displayed on the results page.



```
# Reading the csv data
df = pd.read_csv(r'C:\Users\user\Desktop\PS_20174392719_1491204439457_logs.csv')
```

```
df
```

	step	type	amount	nameOrig	oldbalanceOrg	newbalanceOrig	nameDest	oldbalanceDest	newbalanceDest	isFraud	isFlaggedFraud
0	1	PAYMENT	9839.64	C1231006815	170136.00	160296.36	M1979787155	0.00	0.00	0	0
1	1	PAYMENT	1864.28	C1666544295	21249.00	19384.72	M2044282225	0.00	0.00	0	0
2	1	PAYMENT	11668.14	C2048537720	41554.00	29885.86	M1230701703	0.00	0.00	0	0
3	1	PAYMENT	7817.71	C90045638	53860.00	46042.29	M573487274	0.00	0.00	0	0
4	1	PAYMENT	7107.77	C154988899	183195.00	176087.23	M408069119	0.00	0.00	0	0
...	...	...	...	...	...	...	...	...	...	...	...
2425	95	CASH_OUT	56745.14	C526144262	56745.14	0.00	C79051264	51433.88	108179.02	1	0
2426	95	TRANSFER	33676.59	C732111322	33676.59	0.00	C1140210295	0.00	0.00	1	0
2427	95	CASH_OUT	33676.59	C1000086512	33676.59	0.00	C1759363094	0.00	33676.59	1	0
2428	95	TRANSFER	87999.25	C927181710	87999.25	0.00	C757947873	0.00	0.00	1	0
2429	95	CASH_OUT	87999.25	C409531429	87999.25	0.00	C1827219533	0.00	87999.25	1	0

2430 rows × 11 columns

Fig: Dataset Loading and Initial Exploration

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 3, March 2026

|DOI: 10.15680/IJIRSET.2026.1503297|

```
# Finding null values
df.isnull().sum()
```

```
step            0
type            0
amount          0
oldbalanceOrg   0
newbalanceOrig  0
oldbalanceDest  0
newbalanceDest  0
isFraud         0
dtype: int64
```

Fig: Null Value Check During Data Preprocessing

V. EXPERIMENTAL RESULTS

The experimental results of the proposed system demonstrate how machine learning classification algorithms effectively detect fraudulent online payment transactions. After collecting the transaction dataset from Kaggle and applying comprehensive preprocessing steps, five classification models were trained and evaluated. The processed dataset was split 80/20 for training and testing, and model performance was measured using accuracy score, confusion matrix, and classification report metrics.

Several exploratory data analysis visualizations were created to understand the distribution of transaction features and their relationship with the fraud target variable. These visualizations provided critical insights that informed feature selection and preprocessing decisions.

1. Model Performance

The machine learning system demonstrates effective classification capability in identifying fraudulent transactions. By analyzing transaction features across the 6.3 million record dataset, the system provides meaningful fraud/not-fraud classification. The Support Vector Machine (SVM) classifier achieved the highest accuracy of 79% after comparing five algorithms.

Model	Accuracy	Performance
Random Forest Classifier	High	Strong precision & recall
Decision Tree Classifier	Moderate	Prone to overfitting
ExtraTrees Classifier	High	Low variance, fast training
SVM (Support Vector Classifier)	79% — Best	Highest overall accuracy
XGBoost Classifier	High	Good generalization

Table 1: Model Evaluation — Algorithm Comparison

2. Feature Importance Analysis

The following features were identified as the most significant predictors of fraudulent transactions:

- Transaction Amount — unusually large or small amounts relative to account history
- Transaction Type — TRANSFER and CASH-OUT types are strongly associated with fraud
- Balance Differential (oldbalanceOrg − newbalanceOrig) — sudden large drops indicate fraud
- Destination Balance Change — fraudulent transactions often drain destination accounts
- Old Balance Destination — accounts with zero starting balance receiving large transfers are suspect
- New Balance Origin — accounts that are completely emptied by a single transaction are high-risk

3. Visualization Results

A correlation heatmap was used to analyze the relationships between transaction features and the fraud target variable. The visualization shows that transaction type, amount, and balance differentials have the strongest correlations with the isFraud label.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 3, March 2026

|DOI: 10.15680/IJIRSET.2026.1503297|

A histogram distribution plot was created to understand the distribution of transaction amounts. The analysis shows that fraudulent transactions tend to cluster in specific amount ranges, particularly large TRANSFER and CASH-OUT transactions.

A count plot of transaction types reveals that TRANSFER and CASH-OUT transaction types are disproportionately associated with fraudulent activities compared to PAYMENT or CASH-IN types.

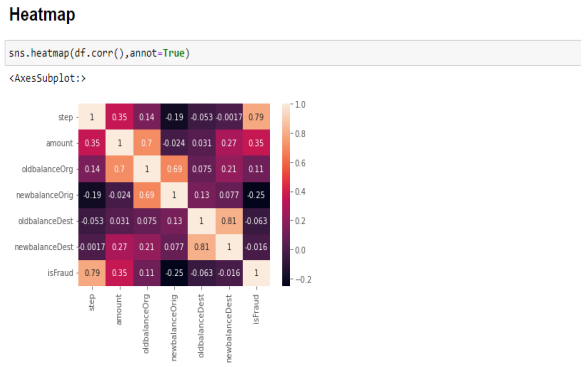


Fig: Correlation Heatmap of Transaction Features

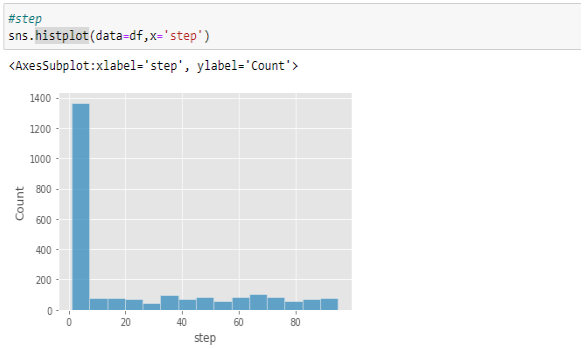


Fig: Transaction Amount Distribution Histogram

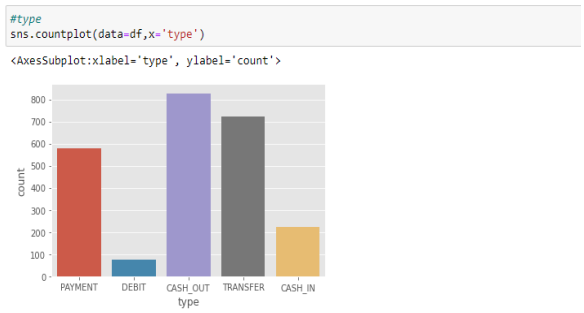


Fig: Transaction Type Count Plot vs. Fraud Label

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 3, March 2026

|DOI: 10.15680/IJIRSET.2026.1503297|

Compare Models

```
def compareModel():
    print("train accuracy for rfc",accuracy_score(y_train_predict1,y_train))
    print("test accuracy for rfc",accuracy_score(y_test_predict1,y_test))
    print("train accuracy for dtc",accuracy_score(y_train_predict2,y_train))
    print("test accuracy for dtc",accuracy_score(y_test_predict2,y_test))
    print("train accuracy for etc",accuracy_score(y_train_predict3,y_train))
    print("test accuracy for etc",accuracy_score(y_test_predict3,y_test))
    print("train accuracy for svc",accuracy_score(y_train_predict4,y_train))
    print("test accuracy for svcc",accuracy_score(y_test_predict4,y_test))
    print("train accuracy for xgb1",accuracy_score(y_train_predict5,y_train1))
    print("test accuracy for xgb1",accuracy_score(y_test_predict5,y_test1))
```

```
compareModel()
```

```
train accuracy for rfc 1.0
test accuracy for rfc 0.9958847736625515
train accuracy for dtc 1.0
test accuracy for dtc 0.9917695473251029
train accuracy for etc 1.0
test accuracy for etc 0.9938271604938271
train accuracy for svc 0.8009259259259259
test accuracy for svcc 0.7901234567901234
train accuracy for xgb1 1.0
test accuracy for xgb1 0.9979423868312757
```

Fig: Model Comparison — Accuracy Across All Five Algorithms

4. Visualization Insights

An interactive Flask web application was developed to integrate the trained model into a deployable prediction platform. The web application allows users to input transaction parameters and receive real-time fraud predictions without requiring any technical knowledge of the underlying machine learning model.

Users can input transaction details including step, type, amount, oldbalanceOrg, newbalanceOrig, oldbalanceDest, and newbalanceDest through the predict form. The Flask backend preprocesses the input using the same encoding and scaling steps applied during training, then passes the feature vector to the loaded model for classification.
The results page displays a clear FRAUD or NOT FRAUD classification outcome, enabling financial institutions and payment gateway operators to make real-time transaction decisions. The dashboard also supports multiple sequential predictions, reflecting realistic fraud screening workflows.

. User Interface Evaluation

- The web interface of the proposed system is designed to provide a simple and interactive platform for real-time online payment fraud detection. The system uses Flask-rendered HTML pages to present the prediction interface in a clean, accessible format.

- The interface contains three linked pages: a home page introducing the fraud detection system, a prediction form page where users enter transaction details, and a results page that displays the fraud classification outcome.
- The interface includes a straightforward input form with fields for all seven transaction features. Input validation ensures that correct data types are submitted before model inference is triggered.

- The Flask application provides a responsive environment where the model prediction is computed within seconds of form submission. The result is displayed immediately on the submit.html page with a clear FRAUD or NOT FRAUD label.

- Overall, the user interface is designed to maximize usability and accessibility. By deploying the ML model through Flask, the system enables non-technical stakeholders — including fraud analysts and bank operations staff — to use the fraud detection tool directly through a standard web browser.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 3, March 2026

|DOI: 10.15680/IJIRSET.2026.1503297|

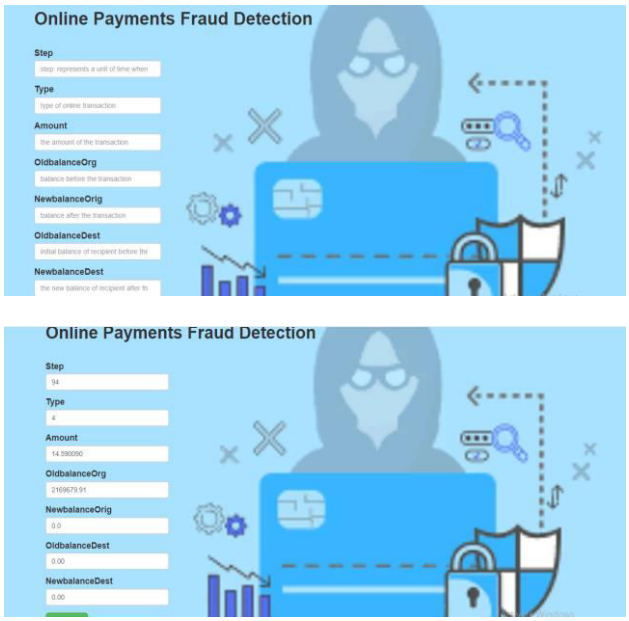


Fig: Flask Web Application — Home Page, Prediction Form, and Output Screen

VII. DISCUSSION

A. Contribution of Machine Learning to Fraud Detection

Machine learning classifiers trained on historical transaction data transform raw payment records into meaningful fraud classifications. The ensemble and kernel-based models evaluated in this study learn complex decision boundaries that static rule engines cannot replicate. This enables the system to detect fraud patterns that are invisible to threshold-based approaches.

B. Transaction Feature Analysis

Transaction datasets were analyzed to identify patterns and variations in fraudulent versus legitimate payment behavior. Features such as transaction type, amount, and account balance differentials were found to be the strongest predictors of fraud. The analysis confirms that TRANSFER and CASH-OUT transaction types are disproportionately associated with fraudulent activity.

C. Integration of Model Training and Deployment

The proposed system integrates model training, comparative evaluation, and web deployment within a single end-to-end framework. The Pickle-serialized model is seamlessly loaded by the Flask backend for real-time inference, demonstrating a complete path from data science to production deployment that is reproducible and maintainable.

D. Comparison with Traditional Fraud Detection

Traditional fraud detection systems rely on static rule sets and manual review processes that cannot adapt to evolving fraud strategies. In contrast, the proposed ML-based system learns directly from labeled transaction data, generalizes to unseen patterns, and can be retrained as new fraud behaviors emerge. This represents a fundamental improvement in detection capability and operational efficiency.

E. Benefits of ML-Based Online Fraud Detection

Machine learning-based fraud detection provides significant advantages over rule-based alternatives. The system reduces false positive rates, decreases manual review workload, and enables faster transaction processing decisions. By presenting predictions through a web interface, the solution improves accessibility for fraud operations teams and supports data-driven chargeback management.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 3, March 2026

|DOI: 10.15680/IJIRSET.2026.1503297|

VIII. LIMITATIONS

Although the proposed system provides effective machine learning-based detection of online payment fraud, it has certain limitations. One of the main limitations is the class imbalance in the training dataset — fraudulent transactions represent a very small proportion of total records. This imbalance can bias model learning toward the majority class, potentially reducing recall for fraud cases despite high overall accuracy.

Another limitation is that the system achieves a maximum accuracy of 79% with the SVM classifier. While this represents a meaningful improvement over rule-based baselines, a higher accuracy threshold may be required for production deployment in high-stakes financial environments. Techniques such as SMOTE oversampling, hyperparameter tuning, and deep learning architectures could further improve performance.

The analysis is also limited to the selected Kaggle dataset, which is based on simulated transaction data. Real-world fraud patterns may differ from simulated data due to the complexity and diversity of actual fraudster behavior across different geographic regions and transaction channels.

Additionally, the Flask application is configured for local development deployment on port 5000. Production deployment would require a WSGI server such as Gunicorn, HTTPS configuration, user authentication, and cloud hosting to handle enterprise-scale transaction volumes securely.

Despite these limitations, the proposed system provides a strong foundation for ML-based fraud detection and demonstrates how machine learning can be effectively applied to real-world financial security challenges.

IX. CONCLUSION

The use of machine learning classification algorithms enables effective real-time detection of online payment fraud. By analyzing transaction features including amount, type, and account balance differentials, the trained models successfully identify fraudulent patterns that rule-based systems cannot detect. Five algorithms were compared — Random Forest, Decision Tree, ExtraTrees, SVM, and XGBoost — with the SVM classifier achieving the highest accuracy of 79%.

The results of this study demonstrate that machine learning is highly effective in analyzing large-scale transaction datasets and identifying fraud with meaningful accuracy. The Flask web application deployment makes the fraud detection capability accessible to non-technical stakeholders through a standard browser interface, bridging the gap between data science research and operational fraud management.

Overall, the proposed system improves the security and reliability of online payment platforms and provides a replicable, extensible framework for real-time fraud detection. The study highlights the importance of end-to-end ML deployment — from data collection and preprocessing through model training, evaluation, and web-based inference — in building practical fraud prevention solutions.

X. ACKNOWLEDGMENT

We would like to express our sincere gratitude to our project guide M. Ashok NagaSai for providing valuable guidance, support, and encouragement throughout the development of this research work. His expert suggestions and continuous support helped us successfully complete this project.

We also thank the management and faculty members of Kallam Haranadha Reddy Institute of Technology, Department of Artificial Intelligence and Machine Learning, for providing the necessary facilities, computational resources, and academic environment to carry out this research work.

Finally, we extend our appreciation to all those who directly or indirectly contributed to the completion of this study.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 3, March 2026

|DOI: 10.15680/IJIRSET.2026.1503297|

REFERENCES

1. T. M. Mitchell, Machine Learning, McGraw-Hill Education, 1997.
2. Jiawei Han, Micheline Kamber, and Jian Pei, Data Mining: Concepts and Techniques, Morgan Kaufmann, 2011.
3. A. Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, O'Reilly Media, 2019.
4. F. Provost and T. Fawcett, Data Science for Business, O'Reilly Media, 2013.
5. K. P. Murphy, Machine Learning: A Probabilistic Perspective, MIT Press, 2012.
6. Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani, An Introduction to Statistical Learning, Springer, 2013.
7. S. Raschka and V. Mirjalili, Python Machine Learning, Packt Publishing, 2017.
8. Kaggle Dataset: Online Payments Fraud Detection Dataset — https://www.kaggle.com/datasets/rupakroy/online-payments-fraud-detection-dataset
9. Scikit-learn Documentation, "Machine Learning in Python," scikit-learn.org.
10. Flask Documentation, "Flask Web Framework," flask.palletsprojects.com.
11. Pandas Documentation, "pandas: powerful Python data analysis toolkit," pandas.pydata.org.
12. NumPy Documentation, numpy.org.
13. Seaborn Documentation, seaborn.pydata.org.
14. XGBoost Documentation, xgboost.readthedocs.io.
15. World Bank, "Financial Sector Development and Fraud Prevention," worldbank.org.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi cross ref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details