International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 05 | May - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Fraud Detection in Financial Transactions Using Machine Learning Techniques

Sivasubramanyam Medasani
Prof. Dept. of Computer Science
& Engineering,
KSSEM Bengaluru, India
sivasubramanyam.m@kssem.edu.in

Kasturi Poornima
Dept. of Computer Science
& Engineering,
KSSEM Bengaluru, India
kasturipoornima@gmail.com

Nikitha L
Dept. of Computer Science
& Engineering,
KSSEM Bengaluru, India
nikitha.lakshmana47@gmail.com

Kushmitha T A
Dept. of Computer Science
& Engineering,
KSSEM Bengaluru, India
kushmitha8904@gmail.com

Kusuma B
Dept of Computer Science
& Engineering,
KSSEM Bengaluru, India
kusumashiva523@gmail.com

***Abstract*—Through integrating data warehousing, data visualization, information retrieval, and stream processing analytics, this project seeks to create a strong fraud detection framework for financial institutions. Significant volumes of transactional data are efficiently processed and stored, allowing for rapid retrieval and in-depth analysis. To detect trends and irregularities, advanced machine learning models such as Random Forest, Decision Tree, Logistic Regression & Naïve Bayes are used. To evaluate these models, we use precision, recall, F1-score, and AUC-ROC. To help ensure practical deployment in financial environments, the system places a strong emphasis on scalability and real-time processing. This improves fraud prevention and strengthens confidence in economic systems.**

Keywords: K-Means, Financial Transactions, Random Forests, Decision Trees, Ensemble Learning, Logistic Regression, Fraud Detection, and Classification.

I. INTRODUCTION

With the rise in online transactions and the intricacy of payment systems, financial fraud has become a more sophisticated and widespread threat in the digital age. Conventional rule-based detection systems frequently fail to identify new fraud patterns, which leads to a significant number of false positives and cases that go unnoticed [1], [2]. To help overcome these obstacles, recent research has highlighted the necessity of machine learning (ML)-powered systems with intelligence that can adaptively identify fraudulent activity [3], [4].

To help identify whether a transaction is fake or not, this study suggests a fraud framework for detection that utilizes both algorithms for supervised and unsupervised machine learning, such as Decision Trees, Random Forest, Logistic Regression, Naïve Bayes, and K-Means Clustering. Prior Studies have indicated the efficacy of these techniques when handling high-dimensional, unbalanced datasets and revealing obscure patterns in financial data [5]– [8]. The ability of the model to identify fraud instantly is improved by integrating stream processing, data warehousing, and real-time analytics, which also improves scalability and operational efficiency [9], [10].To increase the precision and resilience of fraud identification, a large body of research supports the use of hybrid models that combine statistical, machine learning, and anomaly detection techniques [11]–[14]. Furthermore, methods like feature selection, ensemble learning, and transaction aggregation are accustomed to addressing issues like unequal class distribution and changing fraud behaviors [15]– [18]. This project intends to contribute a scalable, accurate, and useful fraud detection solution appropriate for deployment in actual financial systems by expanding upon these tried-and-true methods [19], [20].

II. RELATED WORK

Finding evidence of financial fraud has been thoroughly investigated using data mining and machine learning techniques. By identifying temporal patterns in behavior, Whitrow et al. [1] showed how transaction aggregation could increase detection accuracy. In their survey of a range of classification and clustering algorithms, Phua et al. [2] and Ngai et al. [3] highlighted their functions in identifying fraud in a range of financial domains. These studies highlight the drawbacks of conventional rule-based systems and the advantages of data-driven strategies.

Liu and Fan [4] demonstrated that when it comes to detecting fraudulent activity, Random Forest and other group models outperform single classifiers. When comparing models such as logistic regression, neural networks, and Naïve Bayes, Bhattacharyya et al. [5] and Sahin and Duman [6] demonstrated that hybrid models increase accuracy and robustness, the usefulness of statistical anomaly detection in conjunction with contemporary. Further, machine learning techniques were emphasized by Bolton and Hand [9].

To help identify both known and unknown forms of fraud, more recent methods combine supervised models with unsupervised learning. While Dal Pozzolo et al. [7] tackled practical issues like data imbalance and changing fraud patterns, Carcillo et al. [8] and Jurgovsky et al. [20] investigated such hybrid systems. The advancement of scalable and adaptable frameworks for detecting fraud is based on these studies.

III. Proposed Methodology

To successfully identify fraudulent financial transactions, the suggested system uses a hybrid methodology that blends unsupervised and supervised machine learning algorithms. The process starts with a thorough data preprocessing stage where the dataset is cleaned, missing value imputation is performed, categorical variable encoding is done, and numerical feature normalization is performed. To guarantee that the data is reliable, noise-free, and prepared for building strong models, these procedures are crucial [1].

Several algorithms are utilized in the modeling stage to recognize a range of fraud patterns. Because of its ease of use, interpretability, and efficiency in solving issues with logistic regression and binary classification is employed as a baseline classifier [3]. Transparent and understandable decision-making is made viable by the incorporation of Decision Trees, which create a flowchart-like model based on feature thresholds [4]. Random Forest is employed as an ensemble technique that aggregates the output of multiple decision trees, improving accuracy and robustness while reducing overfitting and improving generalization [2].

Based on the premise of feature independence, Naïve Bayes is used because of its efficiency and speed with high-dimensional data [5]. K-Means Clustering is an unsupervised technique that classifies similar transactions and flags outliers as possible fraud for identifying new fraud patterns [6].

Measures like F1-score, recall, accuracy, and precision, AUC-ROC are used to evaluate the model's performance, guaranteeing accurate fraud identification with few false positives[7], [8]. To ensure scalable and effective transaction classification in live systems, the best-performing models are saved using Pickle and then used for real-time predictions.

IV. Selected Algorithms

A collection of algorithms for machine learning that are both supervised and unsupervised was chosen to undertake this project based on their effectiveness in prior research and their applicability to fraud detection. To categorize activities as either fraudulent or non-fraudulent, these models were utilized in practice using Python and assessed on datasets of financial transactions. Algorithm selection strikes a balance between interpretability, scalability, detection accuracy, and data imbalance management—all of which are major obstacles in actual financial fraud detection systems [1], [3], and [4].

A. Decision Tree

Because of their simple, rule-based structure and interpretability, decision trees were used as the baseline supervised learning model in this project. To identify transactions as either fraudulent or not, the model was trained on transactional features like amount, transaction type, and account identifiers using Python's Scikit-learn library. By dividing data according to Gini impurity, the tree can learn conditional rules that highlight questionable activity. Because of this, it was particularly beneficial for spotting high-risk patterns and figuring out the importance of features early within the model development process. The Decision Tree model was a useful benchmark and provided high transparency for fraud explainability and auditing purposes, despite its propensity for overfitting [3, 4, 5].

B. Logistic Regression

A lightweight, interpretable binary classifier used to determine the likelihood that a transaction is fraudulent is called logistic regression. It is a good place to start when assessing model performance because it works especially well when input features show a linear relationship with the output class. Scikit-learn was used in this project's implementation, and it was trained on features like transaction type, amount, and user behavior metrics. Predicted numbers ranging from 0 to 1, which represent the likelihood of fraud, were mapped using the sigmoid activation function. To improve generalization and avoid overfitting, regularisation strategies such as L2 (Ridge) were used. Because of its speed, interpretability, and simplicity of use, logistic regression has demonstrated strong baseline performance in financial fraud detection tasks and is frequently used in industry benchmarks.[6],[9]. It functioned as a foundational model in this system to verify data preprocessing and to compare with more intricate classifiers, such as ensemble methods and Random Forest.

C. Random Forest

Because of its high accuracy, resilience, and capacity to lessen overfitting—a prevalent problem with individual decision trees—Random Forest, an ensemble of decision trees, is used. Using bootstrap samples and random feature selection, it builds several trees. Afterward, uses majority voting to aggregate predictions. Scikit-learn was used in this project's implementation, and it received training on a range of transaction traits, including account identifiers, transaction type, and amount. The dataset's inherent class imbalance was particularly well-handled by Random Forest, which also detected subtle fraud patterns that were missed by more straightforward models. For best results, hyperparameters like the maximum tree depth in addition to the number of estimators, were adjusted. Accuracy and interpretability were enhanced by the algorithm's feature importance rankings, which revealed the most important characteristics to fraud detection. Its application greatly improved classification performance, and as a result, it became an integral component of the final fraud detection pipeline [4, 5, 7].

D. Naïve Bayes

When applied to large-scale, high-dimensional financial datasets, Naïve Bayes is a quick and effective probabilistic classifier. The assumption of conditional independence between features, which is based on Bayes' Theorem, simplifies the process and permits quick computation without significantly sacrificing accuracy. The distribution of continuous features, such as transaction amount and balance differences, was modeled in this project using the Gaussian variant of Naïve Bayes. The model produced good results in early-stage fraud screening, especially when characteristics like location or transaction type had a strong correlation with fraudulent activity. Because of its low latency and low requirements, it was also advantageous in situations that called for real-time predictions. Particularly in situations where computational efficiency was crucial, Naïve Bayes provided a lightweight substitute for quick detection [6, 9, 12].

E. K-Means Clustering

An unsupervised method for identifying unusual transactions without the need for labelled data is K-Means Clustering. It finds outliers—possibly fraudulent transactions that don't follow the usual behavioral patterns—by clustering transactions according to feature similarity [8], [13]. By identifying new or unidentified fraud types, this model enhances supervised techniques and adds to a hybrid detection framework.

V. IMPLEMENTATION

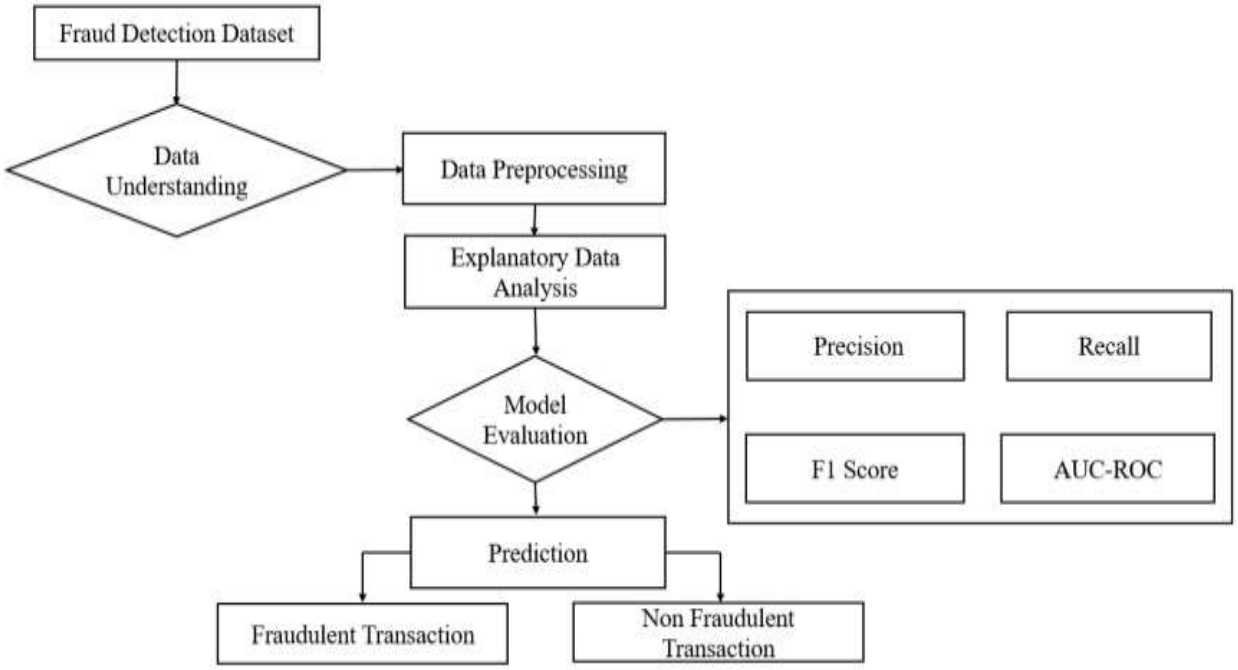

Figure 1: Work Flow Diagram

Steps in Implementation.

A structured pipeline that blends supervised and unsupervised learning techniques is employed to carry out the suggested fraud detection framework. The first step in the process is gathering transactional data from realistic datasets, like PaySim1, which mimic fraud scenarios and mobile money transactions [7], [13].

To enhance input quality and model performance, preprocessing steps are employed to clean the data, normalize features, address class imbalance, and create novel features that capture behavioral patterns [3], [5]. To guarantee strong validation of model generalizability, the dataset is then divided into training (70%) and testing (30%) sets [5, 13].

Five models are created and contrasted:

Decision trees and logistic regression models for interpretable baseline classification [5, 9];

Random Forest for accuracy based on ensembles and resistance to overfitting [2], [6];

Naïve Bayes for probabilistic learning with the assumption of feature independence [9];

Unsupervised identification of emerging trends in fraud using K-Means Clustering [8], [15].

Accuracy, F1-score, recall, precision, and AUC-ROC are the main metrics utilized to assess the model. These offer a thorough understanding of fraud detection performance, particularly when applied to unbalanced datasets [3, 6, 7]. Following the evaluation, the top-performing models are incorporated into a Flask-based user interface and serialised using the .pkl format. Real-time transaction input, secure login, user registration, and the immediate classification of new transactions, as "Fraudulent" or "Non-Fraudulent," are all supported by this interface.

The setup is put through a thorough testing procedure that consists of unit, integration, white box, black box, and testing for user acceptance to guarantee reliability [3]. The architectural design within the system is appropriate for implementation in actual financial environments since it guarantees scalability, modularity, and maintainability.

Real-time fraud detection through stream processing technologies and the incorporation of deep learning models (e.g., CNNs, RNNs) for sequential and spatial pattern recognition are among the planned improvements [14], [17].

DATASET:

The PaySim dataset, a synthetic simulation of mobile money exchanges that are founded on actual financial behaviour, is employed to build the identification of a fraud system. Because of its realistic structure and transaction diversity, it is openly available on Kaggle and has been extensively utilised in research on the detection of fraud [7], [13].

The primary benefit of utilizing this dataset is:

The PaySim dataset is ideal for research on fraud detection because it provides the benefit of realism without sacrificing privacy. Machine learning models can be trained on data that reflects real user behaviour, such as transaction types, frequency, and volume, through mimicking mobile

money exchanges that are based on actual transactional patterns and distributions.

The dataset's size and extreme imbalance make it a perfect testbed for creating and evaluating fraud detection algorithms in scenarios similar to those that financial institutions encounter. This enables the researchers to assess the model's functionality in identifying infrequent fraudulent occurrences, which is a significant problem in practical settings [3], [6]. Furthermore, the dataset's synthetic nature facilitates open experimentation, benchmarking, and reproducibility across studies while removing regulatory barriers, thereby supporting industry-driven and academic advancements in fraud detection systems [15].

Software used for Implementation.

The primary platform used to write the programs for every machine learning tool is Python 3.7, which makes use of well-known libraries like Numpy and Pandas Python modules. Additionally, the user interface and trained models for real-time fraud prediction are created using Flask, a lightweight Python web framework.

Evaluation Metrics.

Accuracy Score: Accuracy Score: A common metric within the machine learning field for assessing a model's correctness is the accuracy score. It calculates the proportion of accurately anticipated data points out of all data points. The accuracy score shows how close one value is to another.

$$accuracy(y, \hat{y}) = \frac{1}{n_{samples}} \sum_{i=0}^{n_{samples}-1} 1(\hat{y} = y_i). \quad (1)$$

F1 Scores: Another metric used in machine learning is the F1 score. Precision (P) and Recall (R) are weighted averages. Simply put, precision represents the proportion of identified positives that were true. Conversely, recall speaks of the proportion of actual proportions that were accurately identified. The greatest amount of this F1 score, also known as the Dice similarity coefficient, is 1.

$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} = \frac{\text{TP}}{\text{TP} + \frac{1}{2}(\text{FP} + \text{FN})} \quad (2)$$

Receiver Operating characteristic curve: The operating characteristic curve (ROC) of the receiver represents a graph or curve that demonstrates the relationship between the rates of false positives and true positives. This is a tool for assessing the productivity of classification models. TPR is on the Y axis of the ROC curve, whereas FPR is on the X axis. When the FPR is zero and the TPR is one, the classifiers are not making any incorrect predictions about any data points. The larger the region in the curve, the better the classifier performs.

VI. OUTCOMES

Making application of models for machine learning (Decision trees, Random Forest, Logistic Regression, Naïve Bayes, and k-means clustering), the project successfully developed an advanced fraud detection framework that improved detection accuracy, decreased false positives, and increased scalability. Metrics like F1-score, recall, precision, and AUC-ROC validated the system's performance, providing a reliable, automated solution to financial fraud detection challenges.

Furthermore, the algorithms were compared and assessed in connection with F1-score and classification accuracy, as indicated in Tables 1 and 2.

Algorithms	Accuracy Score
Random Forest	99.35
Decision Tree	99.12
Gaussian Naive	66.83
Logistic regression	89.98

Table 1. Accuracy Score

Model	Metric	Class 0	Class 1	Accuracy	Macro Average	Weighted Average
Random Forest	Precision	1.00	0.99	0.99	0.99	0.99
	Recall	0.99	1.00	0.99	0.99	0.99
	F1-Score	0.99	0.99	0.99	0.99	0.99
	AUC-ROC	2479	2449	4928	4928	4928
Decision Tree	Precision	0.99	0.99	0.99	0.99	0.99
	Recall	0.99	0.99	0.99	0.99	0.99
	F1-Score	0.99	0.99	0.99	0.99	0.99
	AUC-ROC	2479	2449	4928	4928	4928
Gaussian Naïve	Precision	0.61	0.90	0.66	0.75	0.75
	Recall	0.96	0.37	0.66	0.66	0.66
	F1-Score	0.74	0.52	0.66	0.63	0.63
	AUC-ROC	2479	2449	4928	4928	4928
LogisticRegression	Precision	0.87	0.95	0.90	0.91	0.91
	Recall	0.96	0.85	0.90	0.90	0.90
	F1-Score	0.91	0.90	0.90	0.90	0.90
	AUC-ROC	2479	2449	4928	4928	4928

Table 2. Classification metrics analysis

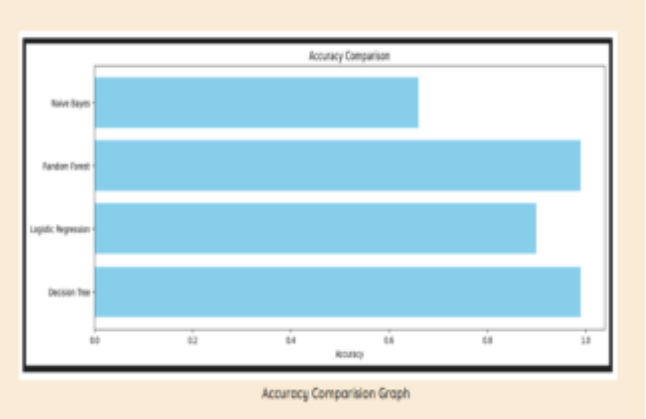

Figure 2. Accuracy Comparision

VII. SUMMARY

This research successfully produced and assessed a machine learning-based system for detecting fraud that can reliably identify whether financial transactions are fraudulent or not. The system showed high accuracy and robustness by utilizing algorithms such as logistic regression, decision trees, Random Forest, Naïve Bayes, and K-Means Clustering. This was particularly apparent when handling the class imbalance that is common in financial fraud data. The system's scalability and practicality were improved by integrating preprocessing methods, performance metrics, and real-time deployment through Flask. For model training and validation, the PaySim dataset offered a safe and realistic setting. All things considered, the project tackles significant drawbacks of conventional rule-based systems and provides a dependable, flexible, and scalable answer to contemporary financial fraud detection problems.

VIII. REFERENCES

[1]. Chen, Y., Wang, Y., & Jiang, Y. (2020). A survey on fraud detection approaches in the financial domain. IEEE Access, 8, 37373-37390.
[2]. Liu, C., & Fan, J. (2020). Financial fraud detection model: Based on random forest. Journal of Computational and Applied Mathematics, 371, 112668.
[3]. Phua, C., Lee, V., Smith, K., & Gayler, R. (2010). A comprehensive survey of data mining-based fraud detection research. Artificial Intelligence Review, 34(1), 1-14.
[4]. 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.
[5]. Ngai, E. W. T., Hu, Y., Wong, Y. H., Chen, Y., & Sun, X. (2011). The application of data mining techniques in financial fraud detection: A classification framework and an academic review of the literature. Decision Support Systems, 50(3), 559-569.
[6].Kou, Y., Lu, C. T., Sirwongwattana, S., & Huang, Y. P. (2004). Survey of fraud detection techniques. IEEE International Conference on Networking, Sensing and Control, 2004, 749-754.
[7]. Dal Pozzolo, A., Boracchi, G., Caelen, O., Alippi, C., & Bontempi, G. (2018). Credit card fraud detection: A realistic modeling and a novel learning strategy. IEEE Transactions on Neural Networks and Learning Systems, 29(8), 3784-3797.
[8]. Carcillo, F., Borgne, Y. Oblé, F., Caelen, O., Kessaci, Y., Bontempi, G., & Termier, A. (2019). Integrating supervised and unsupervised learning in detection. Information Sciences, 557, 317-331.
[9]. Bolton, R. J., & D. J. Hand. (2002). Statistical Detecting fraud: A review. Statistical Science, 17(3), 235-249.
[10]. Van Vlasselaer, V., Eliassi-Rad, T., Akoglu, L., Snoeck, M., & Baesens, B. (2017). AFRAID: Fraud detection via active feature space augmentation. International Information Conference and Knowledge Management, 2017, 1961-1964.
[11]. Juszczak, P., Whitrow, C., Hand, D. J., Weston, D., & Adams, N. M. (2009One technique for identifying fraudulent credit card transactions is transaction aggregation. finding knowledge and mining data, 18(1), 30-55.
[12]. Ahmed, M., Mahmood, A. N., & Hu, J. (2016). An analysis of network anomaly detection methods. The Journal of Network and Computer Applications, 60, 19-31.
[13]. Wei, W., Wang, L., Zhu, M., & Yang, S. (2013). A comprehensive survey on hybrid data mining techniques in detecting credit card fraud. The International Conference on Machine Learning and Cybernetics, 2013, 1127-1131.
[14]. West, J., & Bhattacharya, M. (2016). A thorough analysis of intelligent financial fraud detection. Computers & Security, 57, 47-66.
[15]. Pourhabibi, T., Ong, S. H., Ismail, R., & Lai, K. K. (2020). Fraud detection: A systematic literature review of graph-based anomaly detection approaches. Decision Support Systems, 133, 113303.
[16]. Chen, C. H., & Hu, Y. H. (2013). The detection and prevention of financial statement fraud: A study of banks. Journal of Forecasting, 32(4), 368-382.
[17]. Bauder, R. A., & Khoshgoftaar, T. M. (2018). The detection of fraud involving credit cards using transaction aggregation strategy as well as machine learning methods. 2018 IEEE Worldwide Conference on Information Reuse and Integration (IRI), 191-198.
[18]. Iyer, R. M., & Parthiban, L. (2020). Machine learning algorithms are used to detect credit card fraud. International Journal of Engineering and Advanced Technology (IJEAT), 9(4), 1065-1068.
[19]. Patil, R. B., & Joshi, M. A. (2014). Survey on methods for detecting fraud in healthcare. 2014 International Gathering on Computational Intelligence and Computing Research, 1-5.
[20]. Granitzer, G., Ziegler, K., Calabretto, S., Portier, P. E., He-Guelton, L., Jurgovsky, J., & Caelen, O. (2018). To detect credit card fraud, use sequence classification. Applications of Expert Systems, 100.