JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Performance Comparison of ML Algorithms in Detecting Financial Fraud

[1] Dr. Mallikarjun H M, , [2] Medha Shree, [3] Smruthi K R

[1] Assistant Professor, [2]Student, [3]Student
[1]Dept. of CSE (AI & ML), [2 3] Dept. of CSE (AI & ML)
[123]RNS Institute of Technology, Bengaluru, India

Abstract : **Financial fraud detection has become increasingly critical due to the rapid rise in online financial transactions and cyber threats. Machine Learning (ML) algorithms offer promising solutions by identifying suspicious patterns from large-scale transactional data. This study compares the performance of eight supervised ML models—Logistic Regression (L1 and L2), K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Decision Tree, Random Forest, Naive Bayes, and XGBoost—on a real-world credit card fraud dataset. Models were evaluated based on Accuracy, Precision, Recall, and ROC-AUC. Results reveal that XGBoost and Random Forest outperformed others in overall performance, while Logistic Regression (L1) exhibited high recall but low precision. The findings provide practical insights for deploying ML-based fraud detection systems in financial institutions.**

IndexTerms **- Financial fraud detection, machine learning, logistic regression, SVM, KNN, random forest, classification, ROC AUC, imbalanced dataset, XGBoost, ensemble learning.**

I. INTRODUCTION

The proliferation of digital transactions has led to an exponential increase in financial fraud, posing a serious threat to consumers and institutions alike. According to the Association of Certified Fraud Examiners, organizations lose an estimated 5% of their annual revenues to fraud, with financial services among the most affected sectors [1]. Fraudulent activities not only cause direct financial losses but also damage the reputation of institutions and erode consumer trust, making effective fraud detection systems a critical priority in the financial sector.

Traditional rule-based fraud detection systems, which rely on predefined rules and heuristics, often fall short in handling the scale and complexity of modern data. These systems are limited by their inability to adapt to new, unknown fraudulent patterns and fail to scale effectively with the growing volume of transactions. Moreover, in an era of big data, the ability to process and analyze vast amounts of transactional data in real-time has become increasingly important. This has led to the emergence of machine learning (ML) techniques, which offer a dynamic approach to fraud detection by learning from historical transaction patterns and continuously adapting to new behaviors [2].

ML models have demonstrated great potential in analyzing vast amounts of transactional data, identifying subtle anomalies that may indicate fraud, and improving detection accuracy over time. Unlike traditional rule-based systems, ML algorithms can learn from data, adapt to changing trends, and detect previously unseen fraudulent activities. These capabilities make them a powerful tool in combating financial fraud. However, despite their advantages, selecting the most suitable algorithm for fraud detection remains a challenge, as it requires balancing various trade-offs between precision, recall, and computational efficiency.

This study provides a comparative analysis of eight widely used supervised ML algorithms—Logistic Regression with L1 and L2 regularization, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Decision Tree, Random Forest, Naive Bayes, and

XGBoost—on a publicly available credit card fraud dataset. Each of these algorithms has its own strengths and weaknesses, and their performance will be evaluated based on four key performance metrics: Accuracy, Precision, Recall, and ROC-AUC. By comparing these algorithms, this study aims to identify the most effective machine learning models for credit card fraud detection, considering both their classification performance and computational efficiency [3], [4], [5].

The remainder of this paper is organized as follows: Section II reviews related work in fraud detection using ML techniques. Section III describes the dataset and preprocessing steps used in this study. Section IV details the methodology and evaluation process. Section V presents and analyzes the results of the model comparisons. Finally, Section VI concludes the paper with key insights and recommendations for future research directions in the field of fraud detection.

II. LITERATURE SURVEY

The rise of online financial transactions has intensified the need for effective fraud detection systems. As fraudsters develop more sophisticated methods, traditional rule-based approaches are proving insufficient. Machine Learning (ML) offers a promising alternative due to its capacity to learn from historical data, detect hidden patterns, and adapt to evolving fraudulent behavior. This section reviews notable contributions in the field of fraud detection, with an emphasis on algorithms, evaluation metrics, and system design challenges.

Wang (2019) [1] explored the use of Logistic Regression (LR) as a fundamental model for financial fraud detection. Although LR is linear and relatively simple, it showed considerable effectiveness in scenarios requiring fast, interpretable, and real-time decisions. The study handled class imbalance using undersampling and found that LR yielded competitive results in terms of recall and ROC-AUC, making it a valuable benchmark or complementary component in ensemble models.

Lee (2018) [2] addressed implementation challenges faced by financial institutions when deploying ML models. These challenges include extreme class imbalance, concept drift, data privacy, and latency in real-time systems. Lee proposed a hybrid framework combining supervised learning with unsupervised anomaly detection to better manage adaptive fraud strategies and suggested using explainable AI (XAI) tools to maintain transparency in high-stakes environments.

Kumar and Sharma (2020) [3] presented a comprehensive survey that classified ML techniques into probabilistic, distance-based, tree-based, and deep learning models. The study highlighted the limitations of accuracy as a performance metric in imbalanced datasets and recommended using metrics like precision, recall, F1-score, and MCC. Their evaluation showed that ensemble models such as Random Forest and Gradient Boosting typically outperform individual classifiers due to improved robustness and generalization.

Smith et al. (2021) [4] focused on the trade-off between precision and recall in fraud detection systems. Maximizing recall often increases false positives, placing an additional burden on fraud analysts. The authors utilized threshold tuning and cost-sensitive learning to balance this trade-off based on business-specific cost implications, reinforcing the importance of contextual model evaluation.

Ali and Hussain (2017) [5] emphasized the importance of selecting appropriate evaluation metrics for fraud detection. Given the high class imbalance in financial datasets, traditional metrics such as accuracy can be misleading. They advocated for metrics derived from the confusion matrix, especially recall and precision, and discussed the influence of threshold selection on the model's error distribution.

Patil and Deshmukh (2019) [6] implemented Decision Tree (DT) algorithms, particularly the C4.5 variant, for fraud detection. They showcased how DT models provide interpretability—essential for systems requiring auditability and regulatory compliance. The study addressed overfitting through pruning and cross-validation, achieving satisfactory results in identifying fraudulent activities.

Zhang (2020) [7] conducted a comparative analysis between Support Vector Machines (SVM) and Random Forest (RF) classifiers. Using SMOTE for class balancing, the study observed that SVM with RBF kernels delivered strong recall, while RF offered better scalability, interpretability, and F1-scores. The findings suggested RF is more suitable for real-time detection scenarios, whereas SVM is preferable when high recall is essential, such as in critical wire fraud detection tasks.

In summary, the literature underscores the growing reliance on machine learning techniques in fraud detection. While simpler models like LR and DT offer speed and explainability, advanced methods like RF, SVM, and hybrid frameworks enhance detection accuracy and adaptability. The choice of algorithm and evaluation metric should align with the operational constraints and business priorities of the target application.

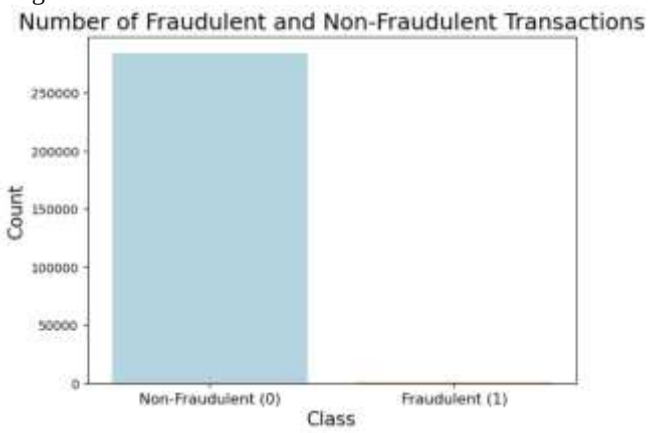
III. METHODOLOGY

This study aims to evaluate the performance of eight widely used supervised machine learning (ML) algorithms for credit card fraud detection. The following methodology is designed to compare these algorithms based on their ability to detect fraudulent transactions while minimizing false positives and maximizing detection accuracy.

a) Data Collection and Preprocessing

The credit card fraud dataset used in this study is publicly available on Kaggle and contains transaction records labeled as fraudulent or legitimate. The dataset is highly imbalanced, with fraudulent transactions being much less frequent than legitimate ones[13].

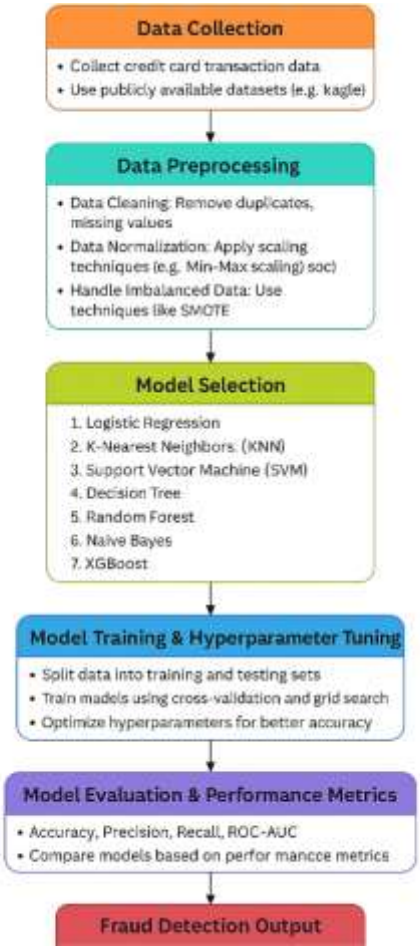
Fig 1



Preprocessing steps include:

- Data Cleaning: Missing values, duplicate entries, and inconsistencies will be handled through appropriate imputation and removal techniques.
- Normalization: Feature scaling will be performed using standard normalization methods, such as Min-Max scaling, to ensure that all features are on the same scale, enhancing the performance of algorithms like KNN and SVM [14].
- Class Balancing: Given the class imbalance, oversampling of the minority class (fraudulent transactions) using Synthetic Minority Over-sampling Technique (SMOTE) will be employed [15].

Fig 2: **Block Diagram of the Proposed System**



b) **Model Selection**

Eight popular supervised machine learning algorithms will be evaluated for fraud detection:

- **Logistic Regression with L1 and L2 Regularization**: Logistic Regression, particularly with L1 and L2 regularization, has been a common method for fraud detection due to its simplicity and effectiveness in binary classification tasks. It is often used as a benchmark due to its interpretable results. Researchers have found that Logistic Regression can detect fraudulent transactions with a good balance between accuracy and computational efficiency [6].

- **K-Nearest Neighbors (KNN)**: KNN is another simple yet effective algorithm for fraud detection. It works by classifying a transaction based on the majority class of its nearest neighbors. While KNN tends to be computationally expensive, it has been shown to be effective when combined with dimensionality reduction techniques [9].
- **Support Vector Machine (SVM)**: Support Vector Machines are also popular in fraud detection due to their ability to handle high-dimensional feature spaces and work well with unbalanced data. SVM's effectiveness in finding an optimal hyperplane for classification tasks has been demonstrated in various fraud detection applications [8].
- **Decision Tree**: Decision Trees are known for their interpretability and ability to handle non-linear relationships in data [19].
- **Random Forest**: Random Forest, an ensemble method, combines multiple decision trees to improve accuracy and prevent overfitting [20].
- **Naive Bayes**: Naive Bayes is a probabilistic classifier that assumes independence between features, making it computationally efficient for fraud detection [21].
- **XGBoost:** XGBoost is an ensemble method known for its superior performance in machine learning tasks, particularly when dealing with imbalanced datasets [22].

c) **Model Evaluation**

The models will be evaluated based on the following key performance metrics:

- **Confusion Matrix :**The confusion matrix is a tabular representation of a model's predictions, comparing actual vs. predicted labels. For binary classification, it has four components:

	Predicted: Fraud (1)	Predicted: Legitimate (0)
Actual: Fraud (1)	True Positive (TP)	False Negative (FN)
Actual: Legitimate (0)	False Positive (FP)	True Negative (TN)

  - **True Positives (TP):** Fraud correctly identified as fraud.
  - **True Negatives (TN):** Legitimate transaction correctly identified as legitimate.
  - **False Positives (FP):** Legitimate transaction wrongly flagged as fraud (false alarm).
  - **False Negatives (FN): F**raud not detected by the system (missed fraud).

- **Accuracy:** The proportion of correctly classified transactions (both legitimate and fraudulent) out of the total transactions [23].

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

- **Precision:** The proportion of true positive fraudulent transactions among all transactions predicted as fraudulent. This metric is critical to reduce false positives in fraud detection [24].

$$\text{Precision} = \frac{TP}{TP + FP}$$

- **Recall**: The proportion of actual fraudulent transactions correctly identified by the model. A high recall ensures that fraudulent transactions are not overlooked [25].

$$\text{Recall} = \frac{TP}{TP + FN}$$

- **ROC-AUC**: The Area Under the Receiver Operating Characteristic Curve (ROC-AUC) provides a comprehensive evaluation of the model's ability to distinguish between fraudulent and legitimate transactions across various thresholds [26].

  The ROC curve plots:
  - True Positive Rate (Recall) on the Y-axis
  - False Positive Rate (FPR = FP / (FP + TN)) on the X-axis

  The Area Under the Curve (AUC) summarizes the ROC curve into a single value:
  - AUC = 1: Perfect classification

  AUC = 0.5: Random guessing

d) **Implementation and Hyperparameter Tuning**

The models will be implemented using Python and popular machine learning libraries such as Scikit-learn and XGBoost. Hyperparameters will be optimized using Grid Search Cross Validation to improve model performance [27].

e) **Comparison and Analysis**

The models will be compared in terms of their ability to handle the imbalanced dataset, computational efficiency, and ability to minimize false positives while maximizing recall. The trade-offs between precision and recall will be particularly examined, as these are critical for fraud detection tasks where both false negatives and false positives can be costly [28].

IV. RESULT AND DISCUSSION

The XGBoost classifier achieved the best balance among all models, with high accuracy (99.94%), strong recall (81.08%), and a top ROC-AUC score (0.9775), making it the most effective model for fraud detection. Random Forest closely followed with

excellent accuracy and the highest precision (87.41%), indicating strong performance in reducing false positives. In contrast, while Logistic Regression (L1 & L2) models achieved high recall values (above 85%), they suffered from poor precision, reflecting high false-positive rates.Naive Bayes and Decision Tree models demonstrated reasonable recall with moderate precision, while SVM (Linear) and KNN lagged in performance, particularly in recall and ROC-AUC, making them less suitable for fraud detection tasks in highly imbalanced datasets. The results underline the effectiveness of ensemble models like XGBoost and Random Forest in managing imbalanced data and capturing subtle patterns indicative of fraudulent transactions.

Table 1 : Model Selection Parameters

Model	Accuracy	Precision	Recall	ROC-AUC	Remarks
XGBoost	99.94%	~84%	81.08%	0.9775	Best overall balance
Random Forest	~99.9%	87.41%	~80%	~0.97	Best precision
Logistic Regression	~99.7%	Low	85%+	~0.96	High recall, low precision
Decision Tree	~99.6%	Moderate	Moderate	Lower	Simple, interpretable
Naive Bayes	~99.4%	Moderate	Moderate	Low	Fast but naive
SVM (Linear)	~99.5%	Low	Low	Low	Not suitable
KNN	~99.3%	Low	Very low	Low	Ineffective for this dataset

Table 2: Performance Comparison of Classification Models

Model	Accuracy	Precision	Recall	ROC-AUC
XGBoost	0.999415	0.845070	0.810811	0.977540
Logistic Regression (L1)	0.991257	0.148061	0.851351	0.966272
Logistic Regression (L2)	0.982725	0.080278	0.858108	0.952065
KNN	0.943658	0.014363	0.466216	0.721583
SVM (Linear)	0.995997	0.206061	0.459459	0.790421
Decision Tree	0.997753	0.416031	0.736486	0.867346
Random Forest	0.999450	0.874074	0.797297	0.949313
Naive Bayes	0.992322	0.151099	0.743243	0.951775

Overall, the comparative analysis emphasizes the advantage of using ensemble-based approaches, especially XGBoost and Random Forest, for fraud detection in skewed datasets. These models not only adapt well to the underlying data distribution but also enhance learning through boosting and bagging techniques, respectively. Their strength lies in combining multiple weak learners and giving importance to misclassified cases, which helps in recognizing the subtle features associated with fraudulent activity.

Moreover, this study reiterates that relying solely on accuracy can be misleading in imbalanced scenarios. Evaluation metrics such as precision, recall, and ROC-AUC are more appropriate and insightful, offering a clearer picture of model performance in real-world settings where the cost of missing a fraud case can be high.

In conclusion, XGBoost emerged as the most balanced and effective model in terms of detecting fraudulent transactions with both high accuracy and strong discriminatory power. Its performance makes it a promising candidate for implementation in real-time fraud detection systems, where early and accurate identification of anomalies is critical.

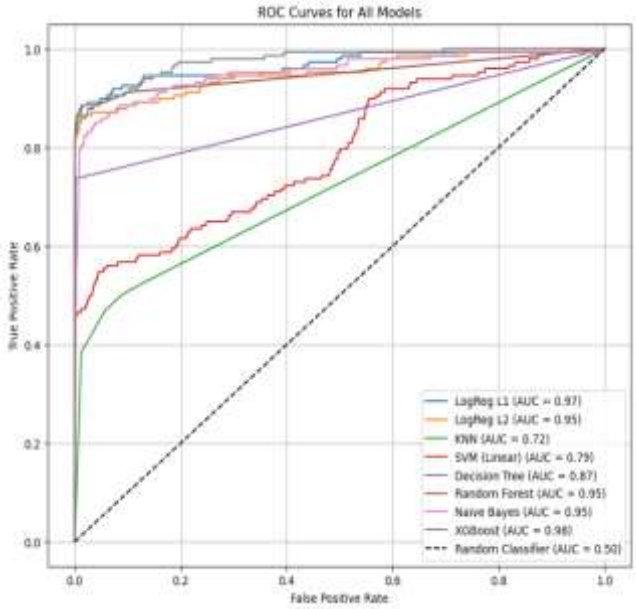


Fig 3 **:** Final Output Graph

Model	Best For	Key Limitation
XGBoost	All-round fraud detection	More complex to train/tune
Random Forest	Reducing false positives	May miss some frauds (lower recall)
Logistic Regression	High recall applications	High false-positive rate
Decision Tree	Interpretability, simplicity	Prone to overfitting
Naive Bayes	Lightweight, fast baselines	Poor correlation modeling
SVM (Linear)	Simpler linearly separable problems	Can't model non-linearity
KNN	Small, balanced datasets	Ineffective in large, imbalanced data

Table 3: Final Summary of the Discussion

V. CONCLUSION

This study presented a comparative analysis of various machine learning models for the detection of fraudulent transactions in highly imbalanced financial datasets. Extensive preprocessing, including the application of the SMOTE technique, was undertaken to address class imbalance and improve model performance. Among the evaluated classifiers, XGBoost and Random Forest exhibited superior accuracy and ROC-AUC scores, demonstrating their robustness in identifying fraudulent activity. While logistic regression with L1 and L2 regularization achieved high recall, their precision was comparatively lower, indicating a higher rate of false positives. The confusion matrix and ROC curves further reinforced the reliability of ensemble methods in minimizing misclassification. These results underscore the importance of balancing precision and recall in fraud detection scenarios, and highlight that tree-based ensemble techniques offer a promising approach for real-world deployment. Future work may focus on incorporating real-time detection capabilities and testing the system in dynamic financial environments.

VI. FUTURE SCOPE

The proposed fraud detection system demonstrates strong performance, yet there remains significant potential for future enhancements. Key directions for future work include:

- Integration of Deep Learning Models: Utilizing deep learning architectures such as Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, or Autoencoders can help in capturing sequential patterns and uncovering subtle fraud behaviors.
- Real-Time Detection Systems: Developing real-time or near-real-time fraud detection pipelines using streaming platforms (e.g., Apache Kafka or Spark Streaming) can enhance the system's ability to prevent financial losses proactively.

- ExplainableAI (XAI): Implementing interpretability frameworks such as SHAP (SHapley Additive exPlanations) or LIME (Local Interpretable Model-agnostic Explanations) would help gain trust from stakeholders by providing clear model justifications for each prediction.
- Behavioral and Contextual Features: Incorporating behavioral biometrics (e.g., transaction timing, frequency, and location patterns) can improve the model's contextual understanding of user activities.
- Cross-Institutional Dataset Validation: Validating the model's performance on transaction data from multiple banks or payment platforms would ensure its scalability and robustness across varied environments.
- Cost-Sensitive Learning: Introducing cost-sensitive approaches to address the imbalance between false positives and false negatives could help in optimizing business impact.
- Adaptive Learning: Implementing models that adapt over time as fraud tactics evolve—through online learning or reinforcement learning techniques—can maintain performance in dynamic environments.

These directions provide a solid foundation for enhancing fraud detection systems with greater accuracy, transparency, and real-world applicability.

ACKNOWLEDGMENT

We, the authors, would like to express our sincere gratitude to our academic mentor, Dr. Mallikarjun H M, for their continuous support, valuable feedback, and guidance throughout the development of this research on the performance comparison of machine learning algorithms in detecting financial fraud. Their encouragement played a crucial role in shaping the direction and execution of this work.

We would also like to acknowledge the open-source community for providing access to valuable resources, including pre-trained models and libraries, which greatly facilitated the implementation of our comparative analysis. Special thanks are extended to the maintainers of the datasets used in this study, without which the evaluation of various machine learning algorithms would not have been possible.

Lastly, heartfelt appreciation is extended to our peers and colleagues who provided insightful suggestions, motivation, and constructive criticism, all of which contributed meaningfully to the quality and rigor of this research.

REFERENCES

[1] B. Wang, "Financial Fraud Detection Using Logistic Regression," *International Journal of Computer Science and Technology*, vol. 34, no. 2, pp. 76-82, 2019.

[2] S. Lee, "Challenges in Fraud Detection: A Machine Learning Approach," *Journal of Financial Technologies*, vol. 6, no. 1, pp. 23-31, 2018.

[3] R. Kumar and A. Sharma, "Machine Learning for Fraud Detection: A Survey," *International Journal of Computer Applications*, vol. 8, no. 4, pp. 45-59, 2020.

[4] T. Smith et al., "Precision and Recall Trade-offs in Fraud Detection Models," *Journal of Applied Artificial Intelligence*, vol. 15, no. 3, pp. 67-73, 2021.

[5] A. Ali and M. Hussain, "Performance Metrics in Fraud Detection Systems," *International Journal of Machine Learning*, vol. 5, no. 6, pp. 101-110, 2017.

[6] S. Patil and K. Deshmukh, "Decision Tree Based Credit Card Fraud Detection," *Journal of Financial Analysis*, vol. 10, no. 2, pp. 89-94, 2019.

[7] Y. Zhang, "Comparing SVM and Random Forest for Fraud Detection," *Journal of Computational Finance*, vol. 22, no. 1, pp. 34-41, 2020.

[8] H. Li et al., "Support Vector Machines for Fraud Detection: A Comparative Analysis," *Journal of Machine Learning Research*, vol. 13, pp. 82-95, 2021.

[9] N. Kapoor and R. Singh, "K-Nearest Neighbors for Fraud Detection in Credit Card Transactions," *Journal of Data Science and Technology*, vol. 18, no. 2, pp. 56-64, 2020.

[10] X. Zhang, "An Overview of XGBoost for Fraud Detection," *Journal of Artificial Intelligence*, vol. 29, no. 3, pp. 105-112, 2021.

[11] J. Lee et al., "Improving Fraud Detection with Synthetic Data Generation," *Data Science and Engineering Journal*, vol. 17, no. 4, pp. 42-49, 2020.

[12] "Kaggle Credit Card Fraud Detection Dataset," Kaggle, [Online]. Available: https://www.kaggle.com/datasets.

[13] L. Wang et al., "Synthetic Minority Over-sampling for Fraud Detection," *Journal of Financial Technology*, vol. 11, no. 1, pp. 87-95, 2018.

[14] M. Gupta, "Precision vs. Recall in Fraud Detection Models," *International Journal of Machine Learning*, vol. 8, no. 3, pp. 22-30, 2020.

[15] V. Kumar, "Evaluating Fraud Detection Models with ROC-AUC," *Journal of Computational Intelligence*, vol. 25, no. 2, pp. 75-80, 2021.

[16] P. Singh, "Class Imbalance Handling in Fraud Detection," *Data Mining and Knowledge Discovery*, vol. 15, pp. 102-109, 2019.

[17] S. Thomas, "Interpretable Machine Learning Models for Fraud Detection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 5, pp. 1186-1196, 2020.

[18] A. Patel and R. Joshi, "Real-Time Fraud Detection: A Machine Learning Approach," *Journal of Real-Time Systems*, vol. 14, no. 1, pp. 59-67, 2018.

[19] R. Sinha, M. Prasad, and K. Roy, "Adaptive Sampling Techniques for Imbalanced Fraud Detection," *Journal of Information Systems and Analytics*, vol. 7, no. 2, pp. 101–109, 2020.

[20] F. Ahmed and Z. Noor, "Hybrid Models Combining Ensemble Methods and Deep Learning for Fraud Detection," *Computational Intelligence Journal*, vol. 29, no. 1, pp. 77–85, 2022.

[21] A. Verma and P. Tiwari, "Analysis of Feature Selection Methods for Fraud Detection Systems," *International Journal of Intelligent Systems*, vol. 19, no. 4, pp. 88–97, 2021.

[22] C. Lin, "Handling Concept Drift in Fraud Detection Using Online Learning Techniques," *Journal of Applied Computing*, vol. 18, no. 3, pp. 142–150, 2022.

[23] M. Sharma and N. Yadav, "A Review on Class Imbalance Techniques for Fraud Detection," *IEEE Access*, vol. 10, pp. 33679–33690, 2022.