FRAUD DETECTION IN CREDIT CARD TRANSACTION USING MACHINE LEARNING

Gopinath.A ,Kamesh.S ,Vengatesan V.B
Mr. Antony suresh Assistant Professor
Department of Information Technology
St.Joseph College of Engineering Sriperumbudur, Chennai-602 117

ABSTRACT

The purpose of this project is to detect the fraudulent transactions made by credit cards by the use of machine learning techniques, to stop fraudsters from the unauthorized usage of customers' accounts. The increase of credit card fraud is growing rapidly worldwide, which is the reason actions should be taken to stop fraudsters. Putting a limit for those actions would have a positive impact on the customers as their money would be recovered and retrieved back into their accounts and they won't be charged for items or services that were not purchased by them which is the main goal of the project. In online transactions, we do not have to appear in person somewhere in the transaction, so we are vulnerable to fraudulent attacks. If the transaction is fraudulent, we can determine it byanalysing the previous transaction and comparing it to the current transaction. If the nature of the previous transaction and the current transaction vary considerably, the current transaction may be a fraudulent transaction. Banks and credit card companies use different methods to detect fraud, such as, Neural Networks and neighbourhood algorithms.

I.INTRODUCTION

Credit card fraud is a huge ranging term for theft and fraud committed using or involving at the time of payment by using this card. The purpose may be to purchase goods without paying, or to transfer unauthorized funds from an account. Credit card fraud is also an add on to identity theft. As per the information from the United States Federal Trade Commission, the theft rate of identity had been holding stable during the mid 2000s, but it was increased by 21 percent in 2008. Even though credit card fraud, that crime which most people associate with ID theft, decreased as a percentage of all ID theft complaints In 2000, out of 13 billion transactions made annually, approximately 10 million or one out of every 1300 transactions turned out to be fraudulent. Also, 0.05% (5 out of every 10,000) of all monthly active accounts was fraudulent. Today, fraud detection systems are introduced to control one-twelfth of one percent of all transactions processed which still translates into billions of dollars in losses. Credit Card Fraud is one of the biggest threats to business establishments today. However, to combat the fraud effectively, it is important to first understand the mechanisms of executing a fraud. Credit card fraudsters employ a large number of ways to commit fraud. In simple terms, Credit Card Fraud is defined as "when an individual uses another individuals' credit card for personal reasons while the owner of the card and the card issuer are not aware of the fact that the card is being used". Card fraud begins either with the theft of the physical card or with the important data associated with the account, including the card account number or other information that necessarily be available to a merchant during a permissible transaction. Card numbers generally the Primary Account Number (PAN) are often reprinted on the card, and a magnetic stripe on the back contains the data in machine-readable format. It contains the following Fields: Name of card holder

- Card number
- Expiration date
- Verification/CVV code
- Type of card
- There are more methods to commit credit card fraud.

Fraudsters are very talented and fast-moving people. In the Traditional approach, to be identified by this paper is Application Fraud, where a person will give the wrong information about himself to get a credit card. There is also the unauthorized use of Lost and Stolen Cards, which makes up a significant area of credit card fraud. There are more enlightened credit card fraudsters, starting with those who produce Fake and Doctored Cards; there are also those who use Skimming to commit fraud. They will get this information held on either the magnetic strip on the back of the credit card, or the data stored on the smart chip is copied from one card to another. Site Cloning and False Merchant Sites on the Internet are getting a popular method of fraud for many criminals with a skilled

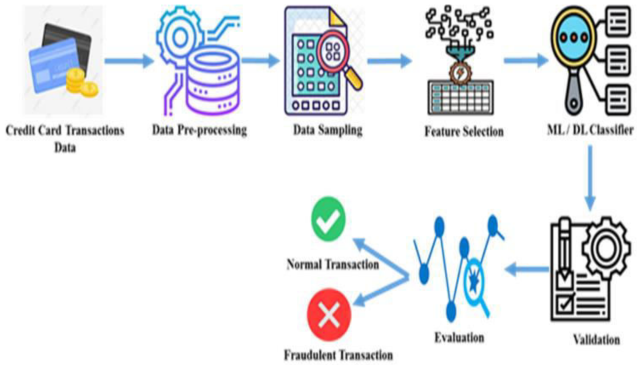
II. EXISTINGANDPROPOSEDSYSTEM

In traditional fraud detection systems, the methods primarily relied on rule-based techniques, where a set of predefined rules was used to detect fraudulent transactions. These rules were often based on statistical analysis and historical data. However, these methods had several limitations: The rules were static and could not adapt to new types of fraud, making the system less effective over time. The system often flagged legitimate transactions as fraudulent, leading to customer dissatisfaction. Manual Intervention Significant manual intervention was required to analyze flagged transactions, which was time-consuming and prone to human error.

III. SYSTEMSTUDY

A study system for fraud detection in credit card transactions using machine learning is designed to accurately identify fraudulent activities within large small portion of the dataset. This imbalance is addressed using techniques such as Synthetic Minority Over-sampling Technique (SMOTE) or random undersampling.Data preprocessing also includes normalization of numeric features and handling any missing or noisy data. The dataset is split into training and testing sets, often using stratified sampling to preserve the proportion of fraud cases.Various machine learning models are implemented and compared, including Logistic Regression, Decision Trees, Random Forest, Gradient Boosting (e.g., XGBoost), and Neural Networks. Performance is evaluated using metrics suitable for imbalanced datasets, such as precision, recall, F1-score, and the area under the ROC curve (AUC-ROC). Cross-validation and hyperparameter tuning are applied to optimize model performance.The system may also incorporate anomaly detection algorithms like Isolation Forest or One-Class SVM, which are particularly useful when labeled data is scarce. Feature importance analysis helps in understanding which variables contribute most to fraud prediction, improving transparency and trust in the model.For practical deployment, the trained model can be integrated into a real-time processing system via APIs. Continuous model monitoring and retraining are essential to adapt to evolving fraud tactics. This study system offers a scalable and efficient approach to combat credit card fraud using machine learning techniques**.**

IV.ARCHITECTURE DIAGRAM



The architecture for a fraud detection system using machine learning is designed to support real-time decision-making while continuously improving model accuracy through feedback and retraining. It consists of the following layers:

This is the entry point for data. It includes real-time transaction streams from credit card networks, payment gateways, mobile applications, and ATM systems. Historical transaction logs from databases are also used for training purposes.Real-time data is captured using streaming platforms like Apache Kafka or AWS Kinesis, while batch data is extracted through ETL processes. All data is stored in scalable storage solutions such as Amazon S3, HDFS, or relational databases for easy access and retrieval.Incoming data is cleaned, transformed, and engineered into relevant features. This includes removing duplicates, handling missing values, normalizing transaction amounts, and generating behavioral features like transaction frequency. SMOTE or undersampling techniques are applied to address class imbalance.

Multiple models—such as Random Forest, XGBoost, or Neural Networks—are trained using labeled data. These models learn to distinguish between fraudulent and legitimate transactions. Anomaly detection models like Isolation Forest can also be used for unsupervised learning. Models are validated and stored for deployment.Trained models are deployed using Flask, FastAPI, or TensorFlow Serving. As new transactions arrive, the system scores them in real-time, assigning a fraud probability and flagging suspicious ones for further action.Model performance is monitored over time using dashboards (e.g., Grafana). Analyst feedback on flagged transactions is fed back into the system to improve future performance, enabling adaptive learning.

V.MODULES

- Data collection
- Data preprocessing
- Feature extraction
- Model selection

DATA COLLECTION

In fraud detection for credit transactions, data collection modules play a crucial role in capturing and processing relevant information. These modules gather real-time and historical data from various sources such as transaction records, customer profiles, merchant information, and device fingerprints. Key data points include transaction amount, time, location, device ID, IP address, and spending patterns. Integration with third-party databases enables verification of user identity and detection of anomalies. Advanced modules also use behavioral biometrics, such as typing speed or mobile swipe patterns, to enhance accuracy. Collected data is structured and transmitted to machine learning models and rule-based engines for analysis. The goal is to detect deviations from normal behavior that may indicate fraud. High-quality, timely data collection ensures effective detection, reduces false positives, and enables proactive fraud prevention. These modules must comply with data privacy laws and maintain secure handling to protect sensitive financial and personal information.

DATA PREPROCESSING

In credit card fraud detection, data preprocessing modules play a crucial role in preparing raw data for analysis and machine learning. These modules begin by cleaning the data—removing duplicates, handling missing values, and correcting inconsistencies. Next, they standardize and normalize numerical features like transaction amount and frequency to ensure uniformity. Categorical variables, such as transaction type or merchant category, are encoded using techniques like one-hot or label encoding. Since fraud data is highly imbalanced, preprocessing includes methods like SMOTE or undersampling to balance the dataset. Outlier detection is also performed to identify unusual patterns that might indicate fraud. Additionally, feature engineering creates new, more informative variables—such as transaction velocity or time between transactions—to improve detection accuracy. Finally, all data is transformed into a format suitable for machine learning models. Effective preprocessing enhances the system's ability to detect fraudulent behavior while reducing false positives and ensuring efficient, real-time analysis.

FEATURE EXTRACTION

Feature extraction modules in credit card fraud detection are critical for identifying meaningful patterns and enhancing model accuracy. These modules transform raw transaction data into relevant features that help distinguish between legitimate and fraudulent behavior. Key extracted features include transaction amount, time of day, location, merchant type, and device ID. Behavioral features such as transaction frequency, average spend, and spending deviation are also derived to capture user habits. Time-based features like time since the last transaction or transactions within a short time frame can indicate suspicious activity. Advanced modules may also extract features based on IP address consistency, geo-location mismatches, or sudden changes in spending patterns. Dimensionality reduction techniques like PCA (Principal Component Analysis) are sometimes applied to simplify complex datasets while retaining important information. Effective feature extraction improves the performance of machine learning models by providing them with high-quality, informative data that enhances the detection of subtle fraud signals.

MODEL SELECTION

Model selection modules in credit card fraud detection are essential for choosing the most effective algorithm to accurately identify fraudulent transactions. These modules evaluate various machine learning models such as Logistic Regression, Decision Trees, Random Forests, Gradient Boosting, and Neural Networks. Each model is tested on historical transaction data using cross-validation techniques to assess performance metrics like accuracy, precision, recall, F1-score, and Area Under the Curve (AUC). Since fraud datasets are highly imbalanced, emphasis is placed on recall and precision to minimize false negatives and false positives. The selection process also considers model complexity, interpretability, and computational efficiency, especially for real-time fraud detection systems. Ensemble methods, which combine multiple models, are often favored for their robustness. Some systems use AutoML or grid search to automate model tuning and selection. Ultimately, the chosen model must balance speed, accuracy, and adaptability to detect evolving fraud patterns effectively in dynamic environments.

VI.SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in anunacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

TYPES OF TESTS

1. UNIT TESTING
2. INTEGRATION TESTING
3. FUNCTIONAL TEST

1.UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledgeof its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

2.INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfullyunit testing, the combination of components is correct and consistent. Integrationtesting is specifically aimed at exposing the problems that arise fromthe combination ofcomponents.

3.FUNCTIONAL TEST

Functional tests provide systematic demonstrations thatfunctions tested areavailable as specified by the business and technical requirements, system documentation, and user manuals.Functional testing is centred on the following items:Valid Input: identified classes of valid input must be accepted.Invalid Input: identified classes of invalid input must be rejected.Functions: identified functions must be exercised.

Output: identified classes of application outputs must be exercised.Systems/Procedures: interfacing systems or procedures must be invoked. Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

SYSTEM TEST

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known andpredictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

WHITE BOX TESTING

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black boxlevel.

BLACK BOX TESTING

Functions: identified functions must be exercised.

Output: identified classes of application outputs must be exercised.Systems/Procedures: interfacing systems or procedures must be invoked.Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

SYSTEM TEST

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points

.

WHITE BOX TESTING

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

BLACK BOX TESTING

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box. you cannot “see” into it. The test provides inputs and responds to outputs withoutconsidering how the software works.

UNIT TESTING:

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

VII.CONCLUSION

In this project, Machine learning technique like Logistic regression, svm, CNN, xg boost and Random Forest were used to detect the fraud in credit card system. Sensitivity, Specificity, accuracy and error rate are used to evaluate the performance for the proposed system. The accuracy for logistic regression, svm and random forest classifier are 90.0, 94.3, and 95.5 respectively. By comparing all the three methods, found that random forest classifier is better than the logistic regression and decision tree.Credit card fraud detection using machine learning has become a vital tool in combating the increasing volume and sophistication of fraudulent activities in digital transactions. Machine learning models offer a dynamic and data-driven approach, enabling systems to learn from historical transaction data and identify hidden patterns that signify fraud. Unlike traditional rule-based systems, machine learning techniques can adapt to new fraud strategies over time, making them more effective in real-time detection.A complete fraud detection system involves several key stages, including data collection, preprocessing, feature extraction, and model selection. Each stage contributes to the overall accuracy and efficiency of the system. Data preprocessing ensures the quality and balance of the dataset, while feature extraction creates informative variables that improve model performance. Careful model selection—often involving algorithms like Random Forests, XGBoost, or Neural Networks—ensures that the system can detect even subtle fraudulent behavior with high accuracy.Despite the advantages, challenges such as data imbalance, false positives, evolving fraud tactics, and the need for explainability persist. Addressing these issues requires continuous model retraining, integration of real-time analytics, and the use of advanced techniques like anomaly detection and ensemble learning.In conclusion, machine learning significantly enhances the ability to detect and prevent credit card fraud by leveraging complex data patterns and providing scalable, automated solutions. As fraudsters become more sophisticated, ongoing innovation and adaptation in machine learning methods are essential to stay ahead and ensure secure financial transactions for users and institutions alike.

VIII. SCOPE OF FUTURE ENHANCEMENT

The scope for future enhancement in credit card fraud detection using machine learning is vast, driven by the rapid evolution of fraud techniques and advancements in technology. One key area is the integration of deep learning models, such as recurrent neural networks (RNNs) and transformers, which can better capture sequential transaction patterns and user behavior over time. Additionally, real-time fraud detection systems can be further optimized using streaming data analytics and edge computing to ensure instant response with minimal latency.Another promising direction is the incorporation of explainable AI (XAI) to improve transparency and trust in automated decisions, particularly for financial institutions and regulators. Enhancing multi-modal data fusion, which combines transaction data with biometric, geolocation, and device data, can significantly improve detection accuracy.The use of federated learning can also enhance data privacy by enabling institutions to train models collaboratively without sharing sensitive customer data. Moreover, the application of blockchain for secure and tamper-proof transaction logs can provide an additional layer of security.Finally, continuous model updating and adaptive learning systems will help keep pace with evolving fraud patterns. These enhancements promise to make fraud detection systems smarter, faster, and more resilient in the future**.**

VIII. REFERENCES

1. Singh Y, Hussain I, Mishra S, Singh B, “Adaptive neuron detection-based control of single-phase SPV grid integrated system with active filtering”, IET Power Electron., Vol. 10 Is. 6, pp. 657-666, 2017.
2. Y. Abakarim, M. Lahby, and A. Attioui, ``An efficient real time model for credit card fraud detection based on deep learning," in Proc. 12th Int. Conf. Intell. Systems: Theories Appl., Oct. 2018, pp. 1_7, Doi: 10.1145/3289402.3289530.
3. H. Abdi and L. J. Williams, ``Principal component analysis," Wiley Inter-discipl. Rev., Computed. Statist., vol. 2, no. 4, pp. 433_459, Jul. 2010, doi:10.1002/wics.101.
4. V. Arora, R. S. Leekha, K. Lee, and A. Kataria, ``facilitating user authorization from imbalanced data logs of credit cards using artificial intelligence," Mobile Inf. Syst., vol. 2020, pp. 1_13, Oct. 2020, Doi: 10.1155/2020/8885269.
5. A. O. Balogun, S. Basri, S. J. Abdulkadir, and A. S. Hashim, ``Performance analysis of feature selection methods in software defect prediction: A search method approach," Appl. Sci., vol. 9, no. 13, p. 2764, Jul. 2019, Doi: 10.3390/app9132764.
6. B. Bandaranayake, ``Fraud and corruption control at education system level: A case study of the Victorian department of education and early childhood development in Australia," J. Cases Educ. Leadership, vol. 17,
7. no. 4, pp. 34_53, Dec. 2014, Doi: 10.1177/1555458914549669.
8. J. Baszczyski, A. T. de Almeida Filho, A. Matuszyk, M. Szelg_, and R. Sowiski, ``Auto loan fraud detection using dominance-based rough set approach versus machine learning methods," Expert Syst. Appl., vol. 163,
9. Jan. 2021, Art. no. 113740, doi: 10.1016/j.eswa.2020.113740.
10. B. Branco, P. Abreu, A. S. Gomes, M. S. C. Almeida, J. T. Ascensão, and P. Bizarro, ``Interleaved sequence RNNs for fraud detection," in Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2020, pp. 3101_3109, doi: 10.1145/3394486.3403361.
11. F. Cartella, O. Anunciacao, Y. Funabiki, D. Yamaguchi, T. Akishita, and O. Elshocht, ``Adversarial attacks for tabular data: Application to fraud detection and imbalanced data," 2021, arXiv:2101.08030.
12. S. S. Lad, I. Dept. of CSE Rajaram Bapu

Institute of Technology Rajaramnagar Sangli Maharashtra, and A. C. Adamuthe, ``Malware classi_cation with improved convolutional neural network model,'' Int.J. Computed. Netw. Inf. Secure., vol. 12, no. 6, pp. 30_43, Dec. 2021, doi: 10.5815/ijcnis.2020.06.03.

13. V. N. Dorna Dula and S. Geetha, ``Credit card fraud detection using machine learning algorithms,'' Proc. Computed. Sci., vol. 165, pp. 631_641, Jan. 2019, doi: 10.1016/j.procs.2020.01.057.

14. I. Benchaji, S. Douzi, and B. E. Ouahidi, ``Credit card fraud detection model based on LSTM recurrent neural networks,'' J. Adv. Inf. Technol., vol. 12, no. 2, pp. 113_118, 2021, doi: 10.12720/jait.12.2.113-118.

15. Y. Fang, Y. Zhang, and C. Huang, ``Credit card fraud detection based on machine learning,'' Computed., Mater. Continua, vol. 61, no. 1, pp. 185_195, 2019, doi: 10.32604/cmc.2019.06144.