

Identifying Fraudulent Credit Card Transactions Using Ensemble Learning

K. Bhagavan¹, Kappaganthu Venkata Nagasatyanarayana²

¹ Professor, Department of Computer Science and Engineering, K L University, Vaddeswaram, Guntur, Andhra Pradesh, India, bhagavan@kluniversity.in, bhagavankonduri1981@gmail.com

² Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Andhra Pradesh, Guntur(D) 2401050187cse@gmail.com

Abstract

One major challenge for banks and financial institutions is identifying fraudulent credit card transactions, as criminals can also masquerade as legitimate cardholders. To address the inherent class imbalance problem in a fraud detection problem, resampling techniques such as oversampling, undersampling, and SMOTE are applied to the datasets that include European Data and Sparkov Data. To improve the classification, ensemble learning is employed to select different algorithms to make it more accurate and reliable. The study indicates the use of an ensemble-based method with advanced resampling techniques to improve the training and forecasting of the models. The comparison of several classification models is done in great detail and the best model is found to be the Stacking Classifier. It combines multiple base models to enhance the accuracy, precision, recall and F1 score of all techniques. This can prove to be a major step in enhancing fraud detection systems, providing accurate identification of fraudulent transactions and fewer false positives. The proposed solution highlights the importance of ensemble methods and data balancing techniques for tackling the complexity in financial fraud detection.

Keywords: Fintech, credit card fraud detection, ensemble learning, machine learning, simulated dataset, real-world data set."

1. Introduction

In the digital payments world, detecting CCF is a significant problem that is related to identifying irregular payments made via credit cards. The problem is quite massive, as in 2020, more than 2.183 million cardholders had Visa and MasterCard cards [1]. Reports say that there were almost 1.4 million occurrences of identity theft that year, with 393,207 of these being linked to credit card fraud [2]. In 2020, fraudulent credit card use cost \$28.6 billion, up from \$23.97 billion in 2017, a 19.3% rise. These losses will grow to \$408 billion by the end of the next decade, according to the Nilson Report of 2022 [3]. While the banks and other financial institutions are taking steps to slow the increase, it continues to rise. This is why robust credit card fraud detection systems are crucial.

There are many approaches that have been discovered to prevent consumers from abusing their credit cards. They can be classified into two categories: Statistical Methods and ML Methods [2]. From a statistical point of view, finding fraudulent transactions means finding outliers in a dataset. Various statistical techniques such as box-and-whisker plots, normal distribution analysis and cluster-based techniques have been employed in the process. The complexity of credit card transaction data, however, has increased the need for credit card ML solutions, which offer more flexibility and precision.

ML classifiers are designed to examine historical data and make predictions on what transactions are likely fraudulent. Some of the most common methods to identify CCFs are given by decision trees, support vector machines, neural networks and regression models [3] [4] [5]. While these methods have been moderately successful to varying extents, their success usually depends on the quality and attributes of the data sets that are used to train them.

The ensemble learning methods have been added to enhance the detection capability. Ensemble learning attempts to create a better and more reliable meta-classifier that is an aggregation of a set of basic classifiers. One way to do this is called "bagging," which is making numerous samples from the training data with replacement to train classifiers separately. Then, aggregation methods like voting are used to come up with final predictions. Another method to obtain an ensemble of classifiers is boosting. It learns them sequentially and passes the prediction errors from one model to another to improve the overall performance. AdaBoost, Gradient Boosting, and XGBoost are other examples [6].

The third type of ensemble, called stacked generalization or stacking, involves creating a strong ensemble of classifiers that are then used to form a powerful meta-classifier. Some researchers recommend stacking as a possible solution to reduce the complexity of credit card fraud detection by merging different credit card fraud detection models together [7].

2. Related work

Credit card fraud detection has been a challenge for financial institutions for a long time, and their troubles have only increased over time with the number of transactions, along with the skill of the thieves. There is a lot of research that has been done on how ML and statistical methods can be implemented to reduce the risks associated with fraudulent transactions. The aim of these strategies is to improve the accuracy and reliability of fraud detection processes and to minimize the false negative and false positive rates.

Varmedja et al. [10] explored the application of ML algorithm for credit card fraud detection in this study. Their study underlined the need of tackling unbalanced data, a typical problem in fraud detection jobs, through strategies like oversampling, undersampling, and the usage of SMOTE. The findings indicated ensemble learning methods such as Random Forest and Gradient Boosting performed better than individual models due to their ability to better generalize to unseen data. Raj and Portia [11] also investigated other methods to detect fraud and concluded that ML classifiers are the most effective to detect unusual patterns in transaction data. They stated that ML techniques are suitable for fraud pattern changes as they're adaptive.

The possibilities of using DL to detect credit card fraud have also been investigated. Vimal and Vimal [12] put forward a population-based optimized and condensed fuzzy deep belief network to make detection more accurate. They were robust to very unbalanced datasets, by applying a combination of feature selection and dimensionality reduction techniques. In order to detect fraud, Razooqi et al. [13] proposed a hybrid system using fuzzy logic methods along with neural networks with better accuracy and better interpretability than the conventional ML methods. The above studies are examples of the increasing research efforts using DL and hybrid approaches to solve the issues of fraudulent transaction detection.

Another aspect of fraud recognition that has been aided by visualization techniques is the process of identifying the fraud. Lokanan [14] examined the patterns of fraud in credit card transactions and money laundering as they can be identified using visualization tools. It has been useful for scientists to grasp the complicated data linkages and outlier data, bringing ease to viewing large amounts of data. Of course, visualization can't be the only method to locate something, but it is effective when combined with other means.

Another new method for identifying credit card fraud is with graph-based methods. Lebichot et al. [15] created a semi-supervised system based on models to detect fraud using graphs. The system was designed as a graph of nodes representing transactions and edges representing the interactions among the transactions with the abnormal transaction pattern being used as a way of identifying fraud. One of the challenges in real-world fraud detection is the lack of labeled data, which was addressed in this method by using semi-supervised learning.

ML algorithms are still the most effective way to identify credit card fraud, as the algorithms can be trained and adapted. Awoyemi et al. [16] conducted a comparative study of a few ML classifiers including DT, SVM and k-NN. They compared and concluded that ensemble methods, particularly bagging and boosting algorithms, performed better. The ensemble learning is a method of learning by aggregating predictions from various classifiers to achieve better detection accuracy. This will reduce the risk of the model overfitting and increase the chances of generalization.

The hybrid models have been used to provide solutions to these problems. Statistical approaches and ML models are used to provide the entire approach to fraud detection. In the first phase, suspicious transactions can be detected using some typical outlier detection methods like clustering or density based algorithms. These transactions are then passed on to ML classifiers to further analyze. This multi-layered approach addresses all fraudulent tendencies, combining the strength of each approach.

While machine learning and statistical techniques have come a long way in detecting fraud, there is an increasing

demand for XAI solutions in this domain. Regulatory needs and the need to provide transparency in decision making will call for development of models that can offer comprehensible explanation of their predictions. This is especially important in financial applications because false positives can cause consumer unhappiness and false negatives can result in huge financial losses.

3. Materials and methods

The method described aims at improving the detection of credit card fraud using a combination of enhanced data sampling, feature selection, and ML algorithms. The dataset contains European Data and Sparkov Data that will be fixed with class imbalance using data sampling techniques such as OverSampling, UnderSampling, and SMOTE. We will reduce the dimensionality and improve the model's performance using PCA decomposition. To discover different patterns of the data, we will construct and test various MLmodels including RF [19], NB [18] and XGB [20]. Also, ensemble approaches will be used to make things more accurate and stable. These include a Soft Voting Classifier with combination of RF, Naive Bayes, and XGBoost, LightGBM, and a Stacking Classifier with Bagging Classifier using RF and DT. The aim of using these methods is to create a highly effective and fast fraud detection system.

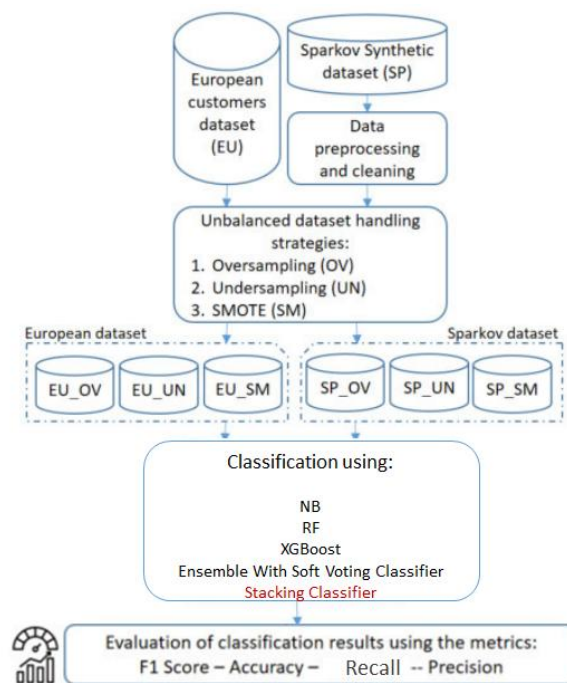


Fig.1 Proposed Architecture

European and Sparkov and identifies fake transactions. The data is first cleaned and preprocessed. To overcome the issue of imbalance in datasets, oversampling, undersampling, and SMOTE are applied. For the classification, NB, RF, XGB, Ensemble with Soft Voting Classifier, Stacking Classifier are used. F1 Score, Accuracy, Recall and Precision are used to test the model performance.

i) Dataset Collection:

The European and Sparkov dataset are both transaction datasets with attributes such as time, amount and anonymized information for fraud detection using ML and to study whether there are fraud signs.

The European Data [8] set is used to find credit card fraud. It has 284,807 transactions and 31 attributes. They include details about the transaction, such as the time of the transaction, the amount and 28 anonymized numerical features (V1 to V28). There is also a target variable called "Class" which indicates if the transaction was fraudulent (1) or not (0). The data set has no missing values and all of the characteristics except the "Class" column are continuous. This data can be used to detect patterns and oddities in financial transactions.

	Time	V1	V2	V3	V4	V5	V6	V7	V8
0	0.0	-1.359807	-0.072781	2.536347	1.378155	-0.338321	0.462388	0.239599	0.098698
1	0.0	1.191857	0.266151	0.166480	0.448154	0.060018	-0.082361	-0.078803	0.085102
2	1.0	-1.358354	-1.340163	1.773209	0.379780	-0.503198	1.800499	0.791461	0.247676
3	1.0	-0.966272	-0.185226	1.792993	-0.863291	-0.010309	1.247203	0.237609	0.377436
4	2.0	-1.158233	0.877737	1.548718	0.403034	-0.407193	0.095921	0.592941	-0.270533

Fig.2 Dataset Collection Table – European

The Sparkov dataset has 1,296,675 entries, with 7 columns. The columns contain information about transactions, including trans_date_trans_time, which indicates the date and time the transaction occurs, and amt, which indicates the amount of the transaction; zip indicates the location of the transaction, and city_pop indicates the population size of the city in which the transaction occurs. The dataset also has information on customers, like their date of birth ("dob"), and the target variable "is_fraud," which shows if a transaction is fraudulent (1) or lawful (0). All the columns have numerical and categorical data types and cannot be null. The data set is good for fraud analysis.

	Unnamed: 0	trans_date_trans_time	cc_num	merchant	category	amt
0	0	2019-01-01 00:00:18	2703186189652095	fraud_Rippin, Kub and Mann	misc_net	4.97
1	1	2019-01-01 00:00:44	630423337322	fraud_Heller, Gutmann and Zieme	grocery_pos	107.23
2	2	2019-01-01 00:00:51	38859492057661	fraud_Lind-Buckridge	entertainment	220.11
3	3	2019-01-01 00:01:16	3534093764340240	fraud_Kutch, Hermiston and Farrell	gas_transport	45.00
4	4	2019-01-01 00:03:06	375534208663984	fraud_Keeling-Crist	misc_pos	41.96

Fig.3 Dataset Collection Table – Sparkov

ii) Pre-Processing:

During this pre-processing step, the various important steps such as data processing, data visualization, feature selection through PCA decomposition and data sampling techniques will be explored.

- Data Processing:** The data processing step is an important step in preparing the raw data for the ML model. It is the data cleaning process of missing value, change data type and remove duplicates. In the search of a credit card fraud, data from transactions should be standardized or normalized, particularly if applying algorithms that are sensitive to scale like neural networks. Also, categorical features could require encoding and unnecessary features should be deleted. Processed data enables models to learn patterns well and recognize fraudulent activity.
- Data Visualization:** To understand distribution and relationship between features which are important for preprocessing, data visualization is important. Some tools which can help you identify pattern, correlation and outlier within a dataset are histogram, scatter plot and heat map. It is helpful to imagine a 'map' of genuine transactions, fraudulent transactions and the correlation between characteristics of the transactions, like amount, time, etc. to identify trends and outliers. Visualization also helps to assess class imbalance, and guide additional preprocessing procedures such as resampling, or feature engineering.
- Feature Selection:** PCA is a dimensionality reduction technique that can be used to discover the essential features of the data. Transforms the initial features into a set of features which are linearly independent and sorted by variance. PCA is effective for decreasing the complexity of data, whilst retaining most of the information. For fraud detection, PCA can help identify the most unique features of the data and remove any redundant or irrelevant ones, which can boost the model's accuracy and efficiency.

d) **Data Sampling:** Data Sampling Methods are applied to handle the issue of class imbalance in the dataset for Fraud Detection. Oversampling: This is an approach that results in having more instances of the minority class. Undersampling: This is done by reducing the number of instances in the majority class. The SMOTE algorithm creates synthetic samples from the minority class using interpolation between the minority class samples. These strategies help the model in better detecting fraudulent transactions by providing balanced datasets, thus, assuring that the ML algorithms do not bias towards the majority class.

iii) Training & Testing:

Training and Testing . The process of creating the dataset for the evaluation of ML models . During training, the model reads the patterns and relationships between the patterns in the data, adjusts the internal parameters to minimize the chance of making mistakes, and reads the patterns and relationships between the patterns in the data again. In test, the trained model is evaluated on unseen data to determine its generalisation skills and accuracy of the predictions it makes. The approach helps ensuring that the model is strong and can forecast correctly on new data that it has not seen before, and so detect fraudulent transactions well.

iv) Algorithms:

Random Forest: RF [19] ensemble training technique builds many decision trees in the learning process and returns the class which has the maximum vote from the decision trees. It is an average of the predictions of multiple trees to avoid overfitting and to improve prediction accuracy. The project uses Random Forest as it is robust enough to handle big data sets and can deal with imbalanced classes, which is perfect for detecting fraudulent transactions as the patterns may be complicated and varied.

Naive Bayes: Naive Bayes is a probabilistic classifier based on Bayes' theorem assuming the independence of the presence of a particular feature and any other feature in the class. It calculates the posterior probability for each class and chooses the class with the highest posterior probability. Appropriate for this project is a simple and efficient algorithm, namely Naive Bayes [18] classification, which is suitable for large data sets and cases where the features are independent which helps in the area of fraud detection.

In generalized notation we write:

$$p(A,B|A) = p(A) * p(B|A) \quad (1)$$

The probability of A and B when the probability of A is given, (or known to have occurred) is equal to the product of the probability of A and the probability of B given A. It is called conditional probability or rather joint conditional probability as the probability is calculated with respect to some condition or some event that has already happened.

XGBoost: XGBoost is a gradient boosting framework which builds the model tree by stages in order and improves their performance by regularization and boosting. It is characterized by great accuracy, fast and efficient. The study applied XGBoost [20] to improve fraud detection which was applied to large datasets and complex relations between features in an efficient way and improve the prediction precision and recall.

$$y^{\wedge}i=\sum_{k=1}^Kfk(xi) \quad (2)$$

where $y^{\wedge}i$ is the last prediction of the i th data point and K is the number of trees in the ensemble and $fk(xi)$ is the prediction of the K th tree for the i th data point.

Ensemble With Soft Voting Classifier (RF + NB + XGB): Soft Voting Classifier is the ensemble of different classifiers (RF, NB and XGB) by the average of their estimation probabilities for final classification. It improves predictions in general using the power of individual models. In the project, the ensemble approach is used to create a more effective fraud detection system that can detect a range of patterns and reduce the chances of false positives.

Stacking Classifier (Bagging Classifier with RF and DT with LightGBM): The Stacking Classifier stacks multiple classifiers (RF, DT, LightGBM) to boost the predictive accuracy, using the output of other classifiers as additional features for the meta-learner. This approach is a synthesis of both models and minimizes prejudice. In the project, it is used to build a

more accurate and generalized fraud detection system, by incorporating the predictions from multiple algorithms.

IV. RESULTS AND DISCUSSION

Accuracy: Accuracy measures the proportion of correctly classified transactions, including both fraudulent and legitimate transactions, among all transactions evaluated. Estimate the accuracy of the test by determining the percentage of true positive and true negative cases of all cases tested. This is mathematically:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (3)$$

Precision: Precision, the proportion of positive cases or samples that are correctly classified. The precision is calculated in the following way:

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \quad (4)$$

Recall: Recall in ML measures the ability of the model to identify all the class examples. It shows the model's ability to predict positive cases compared to the number of positive cases, for instances which fit a class.

$$Recall = \frac{TP}{TP + FN} \quad (5)$$

F1-Score: F1 score is used to measure the accuracy of the ML model. Combining the precision and recall values of the model. The accuracy is the percentage of occurrences for which the model has made a correct prediction in the data set.

$$F1\ Score = 2 * \frac{Recall * Precision}{Recall + Precision} * 100 \quad (6)$$

The performance metrics of the algorithms (F1, precision, recall and accuracy) are provided in tables (1-6). The Stacking Classifier always does better than all the other algorithms on all criteria. The tables also allow you to compare the statistics of the different algorithms.

Table.1 Performance Evaluation Metrics for Oversampling – European dataset

Model	Accuracy	F1 Score	Recall	Precision
Random Forest	0.925	0.925	0.925	0.928
Naive Bayes	0.793	0.801	0.793	0.867
XGBoost	0.937	0.937	0.937	0.939
Ensemble	0.910	0.911	0.910	0.921
Stacking Classifier	1.000	1.000	1.000	1.000

Graph.1 Comparison Graphs for OverSampling – European dataset

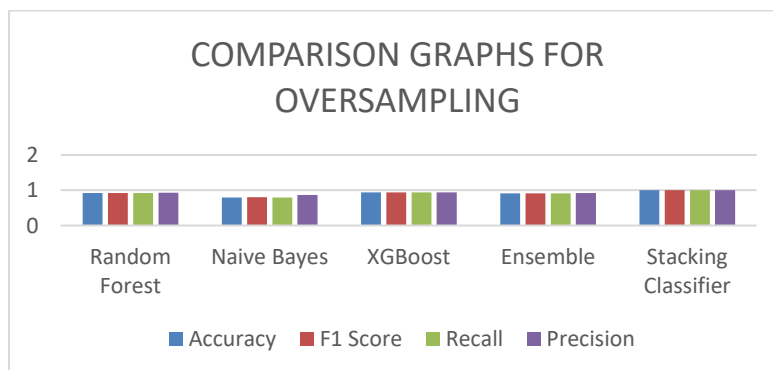
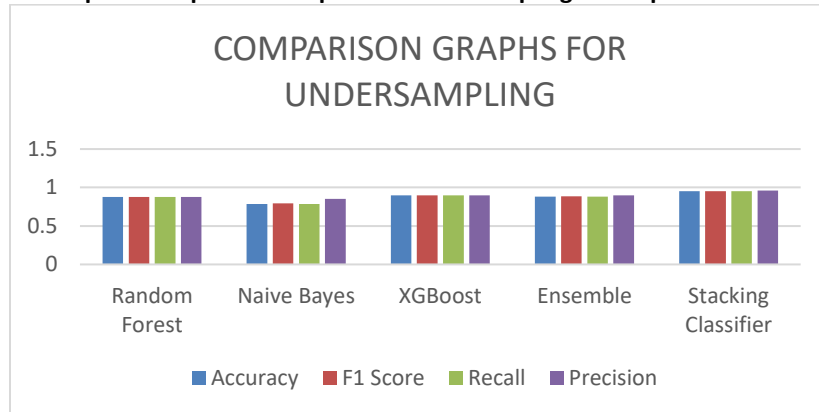
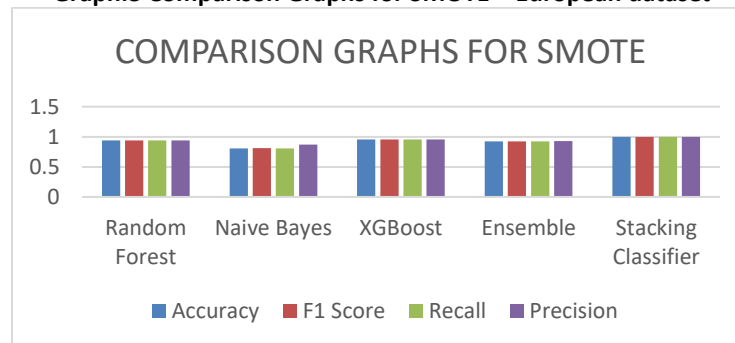


Table.2 Performance Evaluation Metrics for UnderSampling – European dataset

Model	Accuracy	F1 Score	Recall	Precision
Random Forest	0.878	0.878	0.878	0.878
Naive Bayes	0.787	0.794	0.787	0.854
XGBoost	0.898	0.899	0.898	0.900
Ensemble	0.883	0.884	0.883	0.898
Stacking Classifier	0.954	0.954	0.954	0.959

Graph.2 Comparison Graphs for UnderSampling – European dataset**Table.3 Performance Evaluation Metrics for SMOTE – European dataset**

Model	Accuracy	F1 Score	Recall	Precision
Random Forest	0.941	0.941	0.941	0.941
Naive Bayes	0.807	0.813	0.807	0.871
XGBoost	0.955	0.955	0.955	0.956
Ensemble	0.923	0.923	0.923	0.930
Stacking Classifier	1.000	1.000	1.000	1.000

Graph.3 Comparison Graphs for SMOTE – European dataset**Table.4 Performance Evaluation Metrics for OverSampling – Sparkov dataset**

Model	Accuracy	F1 Score	Recall	Precision
Random Forest	0.861	0.862	0.861	0.881
Naive Bayes	0.806	0.811	0.806	0.860
XGBoost	0.913	0.914	0.913	0.921
Ensemble	0.864	0.866	0.864	0.888
Stacking Classifier	1.000	1.000	1.000	1.000

Graph.4 Comparison Graphs for OverSampling – Sparkov dataset

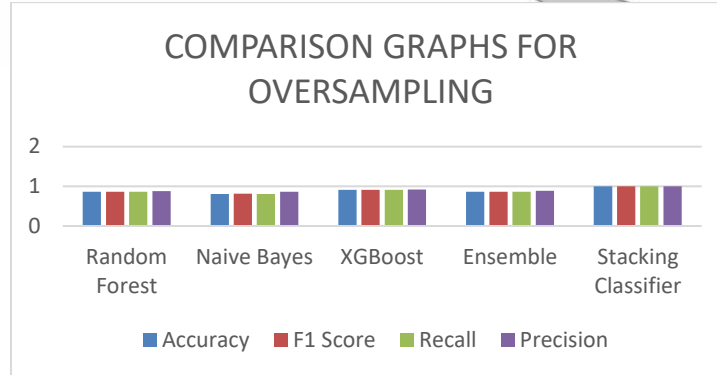


Table.5 Performance Evaluation Metrics for UnderSampling – Sparkov dataset

Model	Accuracy	F1 Score	Recall	Precision
Random Forest	0.861	0.862	0.861	0.884
Naive Bayes	0.843	0.845	0.843	0.867
XGBoost	0.911	0.911	0.911	0.919
Ensemble	0.865	0.866	0.865	0.888
Stacking Classifier	0.995	0.995	0.995	0.995

Graph.5 Comparison Graphs for UnderSampling – Sparkov dataset

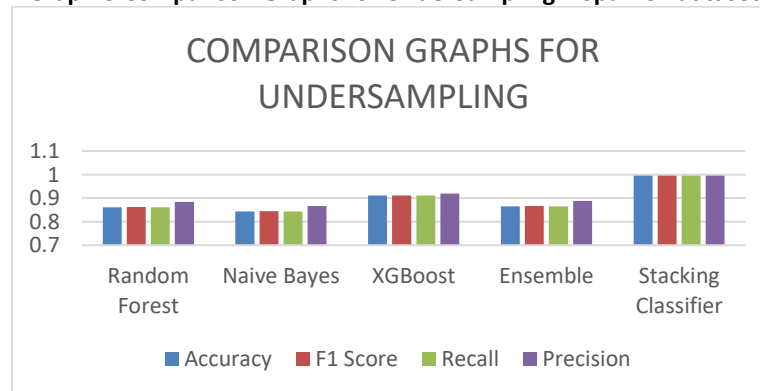
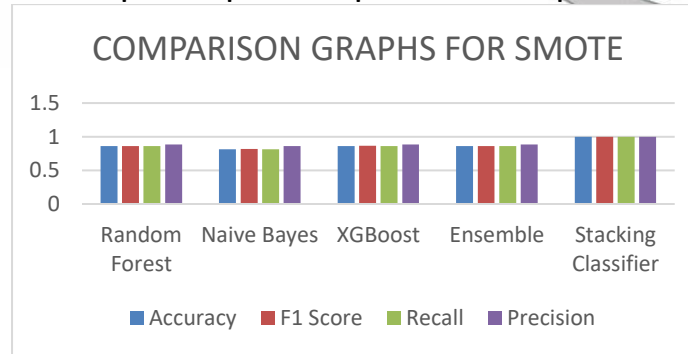


Table.6 Performance Evaluation Metrics for SMOTE – Sparkov dataset

Model	Accuracy	F1 Score	Recall	Precision
Random Forest	0.861	0.863	0.861	0.885
Naive Bayes	0.814	0.819	0.814	0.863
XGBoost	0.863	0.865	0.863	0.885
Ensemble	0.860	0.861	0.860	0.883
Stacking Classifier	0.999	0.999	0.999	0.999

Graph.6 Comparison Graphs for SMOTE – Sparkov dataset



For Graphs (1 to 6), the accuracy, F1-score, recall, and precision are indicated in blue, orange, gray, and light yellow, respectively. The best performance in all the criteria is obtained by the Stacking Classifier, which gets the highest values of the criteria among the other classifiers. The results of these are presented graphically above.

V. CONCLUSION

Credit card malicious transaction identification helps to thwart huge financial losses for institutions. Though the financial stakeholders are making a lot of efforts, all such losses continue to rise and it is significant to have better detection systems. ML methods, especially ensemble models, have shown a lot of promise in solving this problem. The ensemble models have been found to be very effective in detecting fake transactions among various models, with the Soft Voting Classifier and the Stacking Classifier models based on RF, Naive Bayes, and XGBoost being the best models. These are hybrid models that combine the positive aspects of various classifiers, to be better at detecting anomalies by improving accuracy, precision, f1 score and recall. Especially stacking Classifier is one of the best Credit Card Fraud Detection algorithm as well as very accurate. These are very high performing algorithms that can help financial organisations to a great extent to minimise the risk of fraud and enhance the security of transactions in the digital world and avoid considerable financial losses.

It is recommended that future studies include the effect of factor multiplication, factor changes and randomizing in the factor in the authentication phase to further increase the robustness of fraud detection. It will also be crucial to pay special attention to the need for understandable and explainable algorithms. Knowing how these models make their decisions, and being able to explain them, can help to improve understanding of how the models work, transparency, and trust in the forecasts. The enhancements will lead to more accurate and meaningful systems of fraud detection.

References

1. Z. Faraji, "A review of machine learning applications for credit card fraud detection with a case study," *SEISENSE J. Manage.*, vol. 5, no. 1, pp. 49–59, Feb. 2022.
2. F. K. Alarfaj, I. Malik, H. U. Khan, N. Almusallam, M. Ramzan, and M. Ahmed, "Credit card fraud detection using state-of-the-art machine learning and deep learning algorithms," *IEEE Access*, vol. 10, pp. 39700–39715, 2022.
3. Nilson Report. Card Fraud Worldwide. Accessed: May 2023. [Online]. Available: <https://nilsonreport.com/>
4. N. S. Alfaiz and S. M. Fati, "Enhanced credit card fraud detection model using machine learning," *Electronics*, vol. 11, no. 4, p. 662, Feb. 2022.
5. B. Arora and Sourabh, "A review of credit card fraud detection techniques," *Recent Innov. Comput.*, pp. 485–496, 2022.
6. S.Srinidhi, K.Sowmya, and S.Karthika, "Automatic credit fraud detection using ensemble model," in *ICT Analysis and Applications*. Springer, 2022, pp. 211–224.
7. Alharbi, M. Alshammari, O. D. Okon, A. Alabrah, H. T. Rauf, H. Alyami, and T. Meraj, "A novel text2IMG mechanism of credit card fraud detection: A deep learning approach," *Electronics*, vol. 11, no. 5, p. 756, Mar. 2022.
8. Kaggle. (2022). European Cardholders Dataset. Accessed: May 2023. [Online]. Available: <https://www.kaggle.com/datasets/mlgulg/creditcardfraud>
9. Sparkov Data Generation on Github, Sparkv simulator, 2020.
10. D. Varmedja, M. Karanovic, S. Sladojevic, M. Arsenovic, and A. Anderla, "Credit card fraud detection–machine learning methods," in *Proc. 18th Int. Symp. INFOTEH-JAHORINA (INFOTEH)*, Mar. 2019, pp. 1–5.

11. S B. E. Raj and A A. Portia, "Analysis on credit card fraud detection methods," in Proc. Int. Conf. Comput., Commun. Electr. Technol. (ICCET), 2011, pp. 152–156.
12. J. M. V and D. Vimal, "Population based optimized and condensed fuzzy deep belief network for credit card fraudulent detection," Int. J. Adv. Comput. Sci. Appl., vol. 11, no. 9, 2020.
13. T. Razooqi, P. Khurana, K. Raahemifar, and A. Abhari, "Credit card fraud detection using fuzzy logic and neural network," in Proc. 19th Commun. Netw. Symp., 2016, pp. 1–5.
14. M. E. Lokanan, "Financial fraud detection: The use of visualization techniques in credit card fraud and money laundering domains," J. Money Laundering Control, vol. 26, no. 3, pp. 436–444, Apr. 2023.
15. B. Lebichot, F. Braun, O. Caelen, and M. Saerens, "A graph-based, semi supervised, credit card fraud detection system," in Proc. 5th Int. Workshop Complex Netw. Appl. (COMPLEX Network). Springer, 2017, pp. 721–733.
16. J.O.Awoyemi, A.O.Adetunmbi, and S.A.Oluwadare, "Credit card fraud detection using machine learning techniques: A comparative analysis," in Proc. Int. Conf. Comput. Netw. Informat. (ICCNi), Oct. 2017, pp. 1–9.
17. Sohony, R. Pratap, and U. Nambiar, "Ensemble learning for credit card fraud detection," in Proc. ACM India Joint Int. Conf. Data Sci. Manage. Data, Jan. 2018, pp. 289–294.
18. K. M. Leung, "Naive Bayesian classifier," Polytech. Univ. Dept. Comput. Sci./Finance Risk Eng., vol. 2007, pp. 123–156, Nov. 2007.
- A. Prinzie and D. Van den Poel, "Random forests for multiclass classification: Random MultiNomial logit," Expert Syst. Appl., vol. 34, no. 3, pp. 1721–1732, Apr. 2008.
19. T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, Aug. 2016, pp. 785–794.