Performance Analysis of Credit Card Fraud Detection Algorithms: An Overview

Nneoma Okoroafor
Department of Computer Science
Prairie View A&M University
Prairie View, United States
nokoroafor@student.pvamu.edu

Abstract: In this paper, we discussed the increased dependency on Card transactions, including the proportionate increasing rate in fraudulent transactions. Related studies to the subject were discussed. Popular Machine Learning algorithms(XGBoost, Random Forest, and Bagging Decision Tree) were implemented and compared to a more recent proposed algorithm OLightGBM, which the author had suggested was the best. Indeed after analysis, this algorithm performed best in accuracy but was also the slowest.

Keywords—PCA(Principal Component Analysis), FDS, Machine Learning.

I. Introduction

According to a report from Fidelity National Information Services Inc., known as FIS, which assists about 3,200 U.S banks with fraud monitoring, there has been a big jump in attempted and successful credit and debit card fraud since the advent of coronavirus last year. The volume in dollars of attempted fraudulent transactions rose by 35% during this period, and this trend does seem to continue in May 2021. With the increased dependency on cards for online/offline shopping, the volume of fraudulent transactions through credit cards has skyrocketed. It has become imperative that traditional rule-based algorithms are no longer innovative enough for prevention, detecting, and managing risk with card transactions. Hence, modern-day credit card fraud detection systems would have to employ ML-based algorithms heavily. ML algorithms have been in existence to curb this menace. However, no existing algorithm has proved completely efficient as the methods of these fraudsters tend to change with newer inventions.

This paper seeks to analyze optimized LightGBM independently (which the author has claimed to be almost 100% effective at detecting fraud)[8] against the other popular algorithms, including bagging Decision Tree, Random Forest XGBoost Algorithms, and categorize them based on accuracy and speed. These algorithms will be subjected to the same datasets. A plethora of research has been carried out to explore the efficiency and loopholes that exist in each of them and are being exploited by fraudsters. Section II outlines related works, concepts and results achieved. In section III, we describe the methodology for the different Algorithms used. Section IV discusses the Result analysis and the summary in V.

II. Related studies

A. Fraud Classification:

As online/offline usage of cards for transactions increases, this channel becomes a more vulnerable and easy target for fraudsters. While promoting card ease of use during online transactions, only the card details are required, while in some cases, OTP is an added security. This authorization requirement applies to international transactions as well. This method is known as Card-Not-Present because the physical card is not required. Given this situation, it is effortless to steal card information through shoulder surfing, web traffic sniffing, card theft, and credit card information purchase[1][2].

B. Fraud Detection Systems

Fraud Detection Systems (FDS) are automated ML-based programs that companies such as financial institutions employ to detect and prevent fraudulent transactions before the actual owner is informed[3]. Such solutions are to blacklist fraudulent transactions before they hit the database, thereby managing this risk. Another function of the FDS is to ensure that actual transactions are not also misjudged as fraudulent; this could lead to customer discomfort. A solution designed to detect different types of fraud and differentiate the behavior of real users from fraudsters is known as a computational Fraud Detection System[1].

C. Supervised and Unsupervised Learning Algorithms

Supervised learning algorithms involve using labeled data from the history of card transactions and classifying them as fraudulent or non-fraudulent, then generating a model using the existing data to categorize other data samples. Unsupervised learning algorithms use patterns to classify the

credit card transactions that would be considered regular, ranking the remainder as fraudulent[5].
According to [1], their research on applying supervised and unsupervised learning algorithms were fairly successful with the conclusion that the unsupervised learning algorithms performed better at detecting fraudulent transactions and handling data skewness better. However, they also mentioned that there is still a lot of improvement to be made for data samples with highly imbalanced datasets to be handled.

Streaming Analytics is an FDS solution that uses the Hidden Markov model used to detect fraud in real-time. According to the authors, it also reduces false prediction. This model uses K-means to study the behavior of previous transactions based on certain features, then develop rules over time to detect anomalies in the future.[2].

It is also argued that it is improper to assume that the class for each transaction would be available for most real-time FDS[3]. This has been the assumption for most previous performance analyses. This research provides a very detailed instance of applying FDS in real-time transactions and concluded that more emphasis should be laid on the precision of reported alerts. It was also shown that semi-supervised algorithms have proven more promising when used against unlabeled transactions.

Many algorithms have been proposed to detect fraudulent credit card transactions. Amongst the popular ones are Bagging Decision Tree, Random Forest, and XGBoost Algorithms. These algorithms have not proven completely efficient but are topmost efficient[1][5]. However, a newly introduced algorithm, Optimized LightGBM, is suggested to have been more promising at detecting fraudulent and non-fraudulent with the least issue of false categorization, which is the bane of most fraud detection systems[8].

D. Challenges

Though Machine Learning Algorithms have been used in credit card fraud detection[6], there are still several issues surrounding the efficiency and accuracy of these algorithms in detecting fraud. These include a change in card user behavior, domain metrics, data skewness, and real-time classification problem[4].

E. Research Focus

This research aims to evaluate some popular machine learning algorithms(Random Forest Classification, XGBoost, Bagging Decision Tree) against a newly developed ML algorithm(Optimized LightGBM) that suggested being better than the above-aforementioned algorithms. The goal is to establish which algorithm performs better in accuracy and speed than the rest.

III. Research methodology

A. Dataset

The dataset used in this paper is a publicly available real dataset. These data were generated and analyzed during a research collaboration of Worldline and the Machine Learning Group of ULB(Université Libre de Bruxelles) on big data mining and fraud detection. The total number of transactions is 284,807 made in September 2013 by European cardholders. The details of cardholders have been anonymized using a PCA transform because of confidentiality issues. However, amongst the 31 features provided, the time and amount have not been tampered with. It also contains a highly imbalanced data set of 284,315 legitimate transactions and 492 fraudulent transactions which are about 0.17% of total transactions.

Total No. of Transactions	No. of Legitimate transactions	No. of fraudulent transactions	No. of Features	Ref
282,809	284,315	492	31	[3]

Fig 1 Summary of Analyzed Datasets[3].

B. Model Design and ML Algorithms

For the research, five models were designed. They are Bagging Decision Tree, RF, XGBoost, and Optimized Light Gradient Boost Classifier.
Below are steps taken for each of the models.

- Parameters were initialized by importing appropriate python libraries and setting hyper-parameters.
- Import the datasets and assign the x and y values.
- Datasets were split into the Training set and Test set by 75% and 25%, respectively.
- Fit the training data to each model.
- Model performance was evaluated on test data.
- K-fold Cross-validation was used against all models to ensure that a high-performing model was not randomly picked. This was used to calculate the accuracy mean and standard deviation.
- Bagging Decision Tree: is a supervised ML technique that performs ensemble technique on decision trees. Here, the decision tree classifier was applied to the training data, followed by the bagging classifier. It has a predefined target variable which generally is used in problem classification. It is useful for classification and regression. The advantage of using this algorithm is due to its stability and accuracy.

- Random Forest: The Random Forest (RF) is an ensemble method classifier that combines various tree predictors. It is an extension of the Bagging Decision Tree. The advantage of using RF is that it handles higher dimensionality of data better and also maintains higher accuracy for missing data.
- XGBoost is a well-known machine learning algorithm. It is a scalable and accurate implementation of gradient boosting machines. It is based upon the ANN Artificial Neural Network. This algorithm is well known for its high computational power and speed, thereby making it preferable when dealing with large datasets.
- OLightGBM: this is based on tree learning algorithms. In this approach, a Bayesian-based hyperparameter optimization algorithm is intelligently integrated to tune the parameters of the LightGBM algorithm. This is expected to perform better when it comes to accuracy.

IV. RESULT ANALYSIS

A. *Metrics*

The dataset has two classes, 1 and 0. 1 indicating real transactions while 0 indicates fraudulent transactions. We assume that we are dealing with a classification problem. Table 1 shows the accuracy and confusion matrix for metrics evaluation. We also take into consideration that the data is highly skewed classes. See figures 2, 3, 4 and 5, for the Confusion matrix for each model.

Predicted	Actual	
	Positive Prediction	Negative Prediction
Non-Fraud(Class 1)	True Positive(TP)	False Negative(FN)
Fraud(Class 0)	False Positive(FP)	True Negative(TN)

Table 1 Confusion Matrix Table

- Precision(Pr): This metric focuses on analyzing the rate at which the model predicts the true positives against all true positives and false positives. This evaluates the number of positives returned from all positives. See equation 1
- Recall(Rc): This provides an analysis of predicted true positives against all true positives and false negatives. This is illustrated in equation 2.
- F-measure or score(F-score): This quantifies a balance between precision and recall. See equation 3.
- K-Fold Validation(K-fold): This divides the test dataset into the specified number of sets and calculates the average performance.
- Accuracy(Accu): This calculates all correct predictions made from the test dataset.
- Speed checks the amount of time it takes to train a model for each algorithm. This is measured in seconds.

$$\mathrm{Pr} = \frac{TP}{TP + FP} \quad (1)$$

$$Rc = \frac{TP}{TP + FN} \quad (2)$$

$$Fscore = \frac{2}{\frac{1}{\mathrm{Pr}} + \frac{1}{Rc}} \quad (3)$$

Models	Accu	Pr	Rc	f1 score	K-fold	Speed (s)
RF	0.9995	0.93	0.79	0.85	99.95	18
Bagging	0.9994	0.94	0.72	0.82	99.95	32
XGBoost	0.9995	0.92	0.77	0.84	99.95	52
OlightGBM	0.9999	0.95	0.80	0.86	99.96	197

Table 2 Performance Analysis of the ML Algorithms

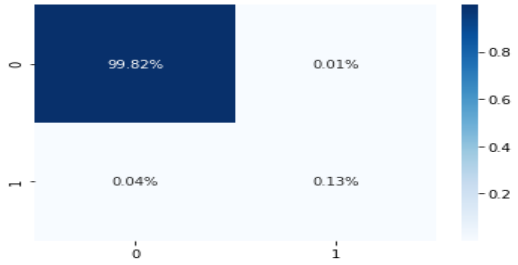

Fig 2. RF Confusion Matrix

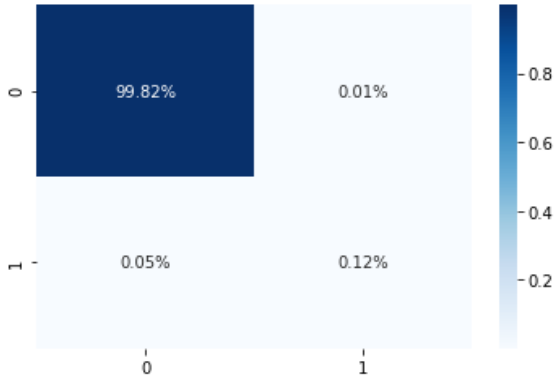

Fig 3. Bagging DT Confusion Matrix

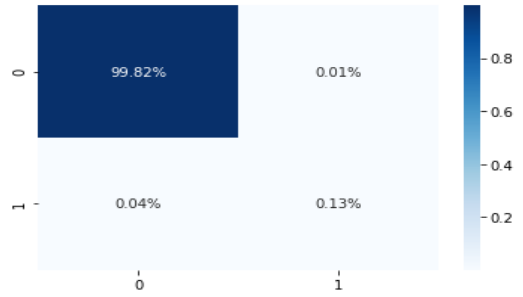

Fig 4. XGBoost Confusion Matrix

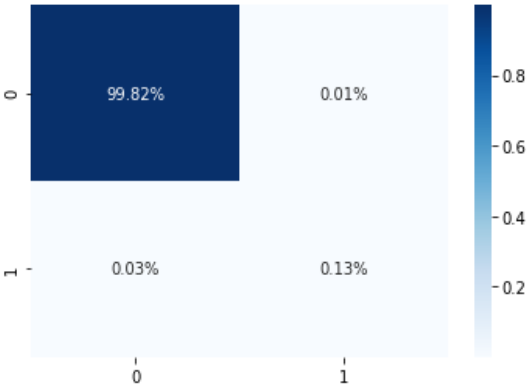

Fig 5. OLightGBM Confusion Matrix

B. Findings

The performance of each machine learning model is documented in Table 2. The result of the analysis is summarised below:

- RF performed best in speed than any other model. Also, it was better than Bagging and XGBoost in terms of Accuracy, Rc, and F1 score.
- OLightGBM performed better than the rest in accuracy, however, was the slowest compared to the other models.
- Bagging performed better than XGBoost and OlightGBM when it comes to speed but was the least in terms of other metrics.

V. Summary

In conclusion, all models performed as posed by previous researchers. However, the author[8] did not indicate that OlightGBM was deficient in speed. The speed issue becomes a huge disadvantage as the accuracy value it brings will be diminished.

A. Limitations

As much as this paper indicates the proposition of research[reference] is true for OLightGBM, it could be argued that the test model might have been oversampled for the algorithm. This poses another problem because there are limited datasets to test with due to the confidentiality of the transactions. Another limitation is that there were many variations for hyperparameter tuning for the Olight algorithm. It is not clear if this might have impacted the poor execution rate.

B. Future Scope

Perhaps in the future, bayesian optimization would be applied to Random Forest so that both accuracy and speed will be leveraged. Also, getting more datasets to train would greatly improve the certainty of these machine learning model results.

Acknowledgment

I would like to thank Dr. Bellam, associate professor of the Computer Science department, Prairie View A&M University, for her guidance and all open source contributors on the Kaggle and TowardDataScience platforms.

References

[1] S. Mittal and S. Tyagi, "Performance Evaluation of Machine Learning Algorithms for Credit Card Fraud Detection," *2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India,* 2019, pp. 320-324, doi: 10.1109/CONFLUENCE.2019.8776925.

[2] U Rajeshwari and B. S. Babu, "Real-time credit card fraud detection using Streaming Analytics", *2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)*, pp. 439-444, 2016.

[3] A. Dal Pozzolo, G. Boracchi, O. Caelen, C. Alippi and G. Bontempi, "Credit Card Fraud Detection: A Realistic Modeling and a Novel Learning Strategy", *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 8, pp. 3784-3797, Aug. 2018.

[4] Y. Sahin and E. Duman, "*Detecting Credit Card Fraud by Decision Trees and Support Vector Machines*", *Int. Multiconference Eng. Comput. Sci.*, vol. I, pp. 442-447, 2011.

[5] S. Dhankhad, E. Mohammed and B. Far, "*Supervised Machine Learning Algorithms for Credit Card Fraudulent Transaction Detection: A Comparative Study*," *2018 IEEE International Conference on Information Reuse and Integration (IRI)*, Salt Lake City, UT, USA, 2018, pp. 122-125, doi: 10.1109/IRI.2018.00025.

[6] M. Kavitha and M. Suriakala, "*Hybrid Multi-Level Credit Card Fraud Detection System by Bagging Multiple Boosted Trees (BMBT)*", *2017 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, pp. 1-5, 2017.

[7] E. W. T. Ngai, Y. Hu, Y. H. Wong, Y. Chen and X. Sun, "The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature", *Decis. Support Syst.*, vol. 50, no. 3, pp. 559-569, 2011.

[8] A. A. Taha and S. J. Malebary, "*An Intelligent Approach to Credit Card Fraud Detection Using an Optimized Light Gradient Boosting Machine,"* in IEEE Access, vol. 8, pp. 25579-25587, 2020, doi: 10.1109/ACCESS.2020.2971354.