Evaluating Performance of Machine Learning Methods for Credit Card Fraud Detection Using CatBoost

[1]ARJUN PARASHAR, [2]ANANYA BHARDWAZ, [3]RISHABH SHARMA, [4]MR ABHILASH SHARMA

[1]Team Lead, [2,3,4]Team Member
SRM Institute Of Science And Technology

Abstract- **Credit card fraud is a major problem in the financial services industry. Thousands of dollars are lost each year due to credit card fraud. Due to privacy concerns, there is not enough research that confirms credit card information. This article uses machine learning algorithms to detect credit card fraud. First, we use the template and then We will use another way of using CatBoost. By applying this algorithm to existing models, we aim to further improve the performance of these models. Finally, we compare the performance of the base model and the powered model with CatBoost and analyze the results. We expect the augmented model to outperform the base model while reducing false positives, especially in fraud detection. Overall, the project aims to prove the importance of feature engineering and algorithm selection in credit card fraud and the effectiveness of the CatBoost algorithm in improving model performance. Experimental results show that the CatBoost method has good accuracy in detecting credit card fraud.**

Index terms: **Credit Card Fraud Detection, Machine Learning, CatBoost.**

I. INTRODUCTION

'Fraud' in credit card transactions is unauthorized and unwanted usage of an account by someone other than the owner of that account [1]. Issuers, merchants and acquirers of merchant and ATM transactions collectively lost $28.58 billion to card fraud in 2020, equal to 6.8¢ per $100 in purchase volume [2]. Credit card fraud detection has been a hot topic in recent years, and machine learning algorithms have shown enormous potential in detecting fraudulent transactions. One such algorithm is CatBoost, a gradient-boosting algorithm that has shown superior performance in handling categorical features, which are prevalent in credit card transaction data.

In this research project, we focus on exploring the effectiveness of CatBoost in detecting credit card fraud. We use a publicly available dataset consisting of credit card transactions made by Europe cardholders in Sept 2013, which has been widely used in earlier studies on fraud detection.

The main aim is to evaluate the performance of the CatBoost algorithm and compare it with popular machine learning algorithms, such as Random Forest, Decision Tree, and Logistic Regression. To achieve this, we first preprocess the data, which involves data cleaning, feature engineering, and data scaling. We then split the dataset into train and test sets and train the CatBoost model on the training set.

We evaluate the performance of the CatBoost model using various metrics, such as accuracy, precision, recall, and F1-score and compare it with the performance of other machine learning algorithms. We also perform hyperparameter tuning to name the best set of hyper parameters for the CatBoost model and compare its performance with the default hyperparameters.

In addition to evaluating the performance of CatBoost, we also perform feature importance analysis to understand which features contribute most to the model's performance and find the most relevant features for credit card fraud detection. Our research project aims to contribute to the growing body of knowledge on credit card fraud detection and to supply insights into the effectiveness of the CatBoost algorithm. The results of our research can be useful for financial institutions, merchants, and consumers in detecting and preventing credit card fraud.

II. LITERATURE REVIEW

This section supplies a literature review of earlier research that used ML techniques for credit card fraud detection.

Ileberi et al. [3] implemented Various machine learning algorithms used for credit card identification in the European credit card fraud database developed in September 2013. Among the machine learning algorithms presented in this work are Decision Trees (DT), Random Forests (RF), Extra Trees (ET), XGBoost (XGB), Logistic Regression (LR), and Support Vector Machines (SVM). Each proposed algorithm is combined with the AdaBoost technique to improve classification quality and deal with non-uniform classes found in the European credit card fraud dataset. DT-AdaBoost,

RF-AdaBoost, ET-AdaBoost, and XGB-AdaBoost achieved 99.67%, 99%, 95%, 99.98%, and 99.98%, respectively. These results show that using the AdaBoost algorithm has a positive effect on the proposed ML method.

Tingfei et al. [4] In their contribution, data and testing of a new monitoring system are presented, which is necessary when the data application is the main source of different classes. Although this method achieves encouraging results, it has limitations. First, this method cannot be used in an uncontrolled environment. Second, the model is slightly inferior to the recovery performance of SMOTE and GAN, although not inferior to the base model in terms of recovery metric performance. Finally, because the plan falls under the category of monitoring work, the model will not perform well on novel false data.

Randhawa et al. [5] In their studies, they used some models such as NB, SVM, and DL for their visual evaluation. Publicly available credit card data is used to evaluate the stand-alone model (model) and the hybrid model using a combination of AdaBoost and majority voting. The MM metric is used to measure performance as it considers the prediction of true and false positive and negative results. The best MCC score by majority vote is 0.823. The test also used real credit card information from financial institutions. Use the same person and mixed models. An excellent MCC score (out of 1) was achieved using AdaBoost and majority voting methods. Ten percent to 30% noise was added to the sample data to further evaluate the mixed model. The best MCC score generated by the majority voting method is 0.942, being 30% noise added to the data. This shows that most voting methods perform well in the presence of noise.

For his future work, the way of learning continues the e-learning model. Other online learning models will also be examined. Using e-learning will allow fraud cases to be detected quickly, possibly in real-time. This will help detect and prevent fraudulent transactions before they happen, thereby reducing the one-day damage done to the financial industry.

Credit card verification is critical to improving credit card usage. Because financial institutions experience significant and ongoing financial losses, and given the difficulty of detecting credit card fraud, it is important to create better value by investigating credit card fraud.

Taha et al. [6] A clever method for detecting credit card fraud is proposed using an optical gradient enhancement machine (OLightGBM). We do a lot of experiments using two real-world data. Combined with other scientific results and ultramodern machine learning algorithms (such as random forest, logistic regression, radial support vector machine, linear support vector machine, k-nearest close, decision tree, and naive Bayesian), the method performed better. achieved the best performance in terms of accuracy, AUC, precision, and F1 scores. The results showed that the proposed algorithm outperformed other classifications. The results also highlight the importance and value of using optimization techniques to improve the predicted performance of the plan.

III. BACKGROUND ON MACHINE LEARNING ALGORITHMS

A. ALGORITHMS

Logistic regression (LR) is a statistical method used to analyze the relationship between a dependent variable and one or more independent variables. It is a type of regression analysis that is commonly used in the field of machine learning for classification problems. Logistic regression works by modeling the log odds of the dependent variable as a linear combination of the independent variables. This linear combination is then transformed using the logistic function, which maps the linear combination to a probability value between 0 and 1.

A Decision Tree (DT) is a machine learning algorithm used for classification and regression analysis. It is a simple and intuitive model that resembles a tree-like structure, where each node is a feature or attribute, each branch is a value or outcome, and each leaf node is a class or prediction. The basic idea is to recursively partition the input space (i.e., the space of feature values) into smaller and smaller regions, based on the values of the input features, until a stopping criterion is met. This process creates a tree-like structure, where each internal node is a test on a particular feature, and each leaf node is a class or prediction.

Random Forest (RF) is a machine learning algorithm that is used for both classification and regression problems. It is an ensemble learning method that combines multiple decision trees to improve the accuracy and stability of predictions. In Random Forest Model, we create a large number of decision trees that are trained on different subsets of the training data and then combine their predictions by taking the majority vote (in classification problems) or the average (in regression problems). Each decision tree is constructed using a random subset of the features in the dataset, which helps to reduce overfitting and improve the generalization of the model.

XGBoost (eXtreme Gradient Boosting) is used for regression, classification, and ranking problems. It is an extension of the gradient-boosting algorithm that uses an ensemble of weak decision trees to create a more accurate and robust model. The basic idea behind XGBoost is to iteratively add decision trees to the model, where each tree is trained on the residuals (i.e., the difference between the predicted and actual values) of the previous trees. In each iteration, XGBoost computes the gradients and Hessians of the loss function concerning the predicted values and then uses these values to construct a new decision tree that minimizes the loss function. The final prediction is the weighted sum of the predictions of all the decision trees.

B. CATBOOST

CatBoost is a gradient-boosting algorithm that is used for classification, regression, and ranking problems. It is an open-source machine learning library developed by Yandex that is particularly known for its ability to handle categorical features in the dataset.

The basic idea behind CatBoost is like other gradient boosting algorithms, such as XGBoost and LightGBM, which is to iteratively add decision trees to the model, where each tree is trained on the residuals of the earlier trees.

Table 3.1

Model	Strengths	Limitations
Logistic Regression	Simple and easy to interpret. Suitable for binary classification Low computational cost	Assumes linearity and independence of features. Not suitable for nonlinear or complex relationships
Decision Trees	Simple and easy to interpret. Can handle both categorical and numerical data. Can handle missing values	Prone to overfitting Not suitable for continuous or regression problems
Random Forest	Reduces overfitting and improves generalization. Can handle both categorical and numerical data. Can handle missing values	Tends to have a higher computational cost than decision trees. Not suitable for continuous or regression problems
XGBoost	Handles complex, nonlinear relationships. High prediction accuracy Regularization and feature selection capabilities	Sensitive to hyperparameter tuning. May overfit the training data
CatBoost	Handles categorical features effectively. High prediction accuracy Automatic hyperparameter tuning	May require more computational resources than other models. Sensitive to hyperparameter tuning

IV. RESEARCH METHODOLOGY

A. FRAUD DETECTION FRAMEWORK

Fig. (4.1) depicts the fraud detection framework that was implemented in this research. In the first step, we load the Credit Card Fraud (CCF) Dataset. In the second step, The CCF dataset is then divided into two sets Namely Training and Test Sets. In the third step, we instantiate the machine learning models (LR, DT, RF, XGB). After Instantiation, the models are trained and tested. In the fourth step, the Instantiated models undergo the CatBoost module. Afterward, the models are trained and tested. The Fraud Detection module evaluates the performance of both the non-boosted and boosted models. As shown in Fig (4.1)

B. DATASET

The Kaggle Credit Card Fraud Detection dataset is a publicly available dataset that holds credit card transaction data from a European bank over two days in September 2013. The dataset holds a total of 284,807 transactions, of which 492 (0.17%) are fraudulent.

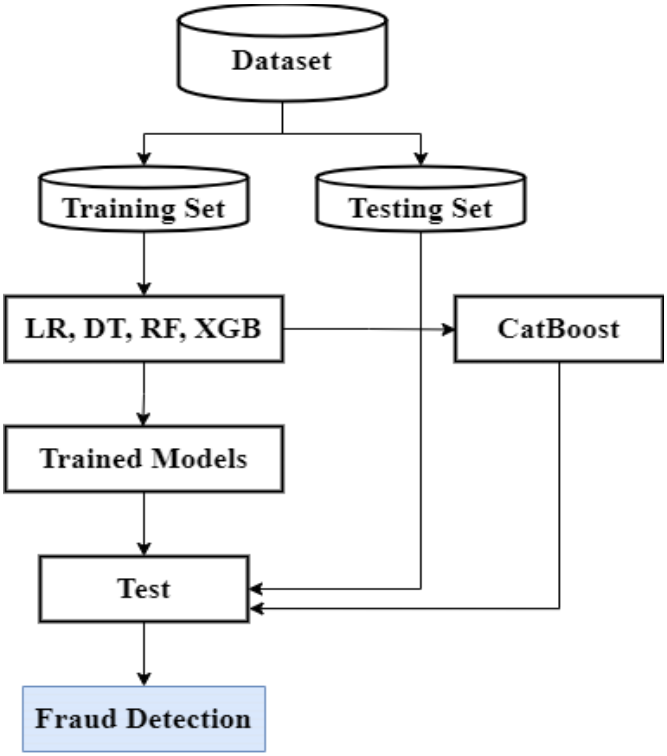

Fig. (4.1)

The dataset has thirty-one features, of which twenty-eight features are anonymized and numeric, and the remaining three features are time, the amount of the transaction, and the target variable showing whether the transaction is fraudulent or not. We have added some variables in the dataset to check how our model performs on Online Transactions the variables include Transaction_Type which is whether the transaction was online or offline Merchant_Category_Code is the code for the category of goods or services the merchant offers. Card Status is whether the card is Active, Inactive, or Cancelled. The anonymized features are PCA-transformed and are not interpretable, which makes it difficult to understand the underlying patterns in the data. Fig (4.2) depicts the Class Distribution of the values.

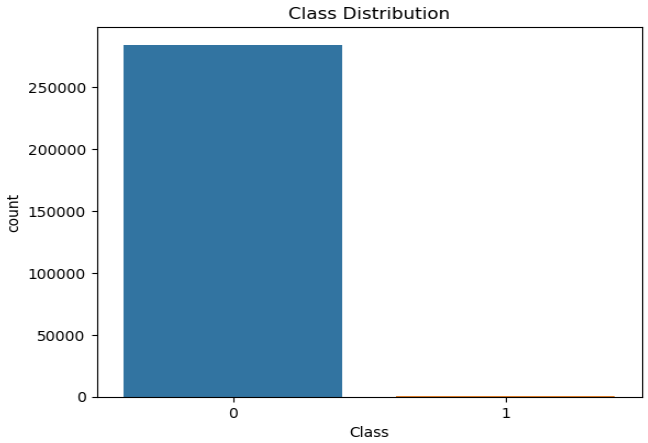

Fig. (4.2)

D. EXPERIMENT RESULT AND DISCUSSION

The results of our experiments on five different machine-learning models are shown above. We have evaluated the performance of Logistic Regression, Decision Tree, Random Forest, XGBoost, and CatBoost algorithms, based on five evaluation metrics: Accuracy, Precision, Recall, F1-score, and ROC AUC.

Logistic Regression achieved high accuracy of 0.999, and ROC AUC of 0.939, which suggests that the model is highly effective in predicting the target variable. However, the recall value of 0.6 is relatively low, indicating that the model's ability to identify true positives is limited. The precision value of 0.808 is moderate, implying that the model's ability to limit false positives is relatively good. The F1 score of 0.690 is relatively low, which suggests that the model is unable to achieve a balance between precision and recall. The model comparison is mentioned above in Fig 5.6

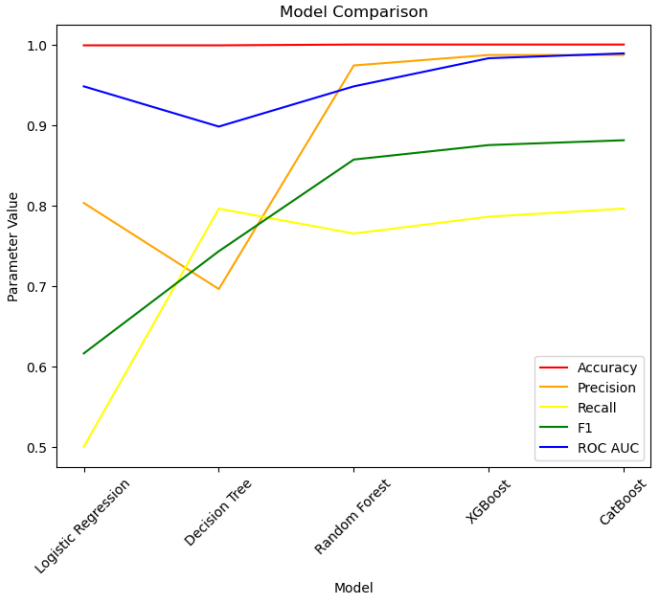

Fig. (4.3)

The Decision Tree model also achieved high accuracy of 0.999, but its performance is not as good as Logistic Regression in terms of other metrics. The model's recall value of 0.796 is much higher than Logistic Regression, but its precision value of 0.729 is relatively low, indicating that the model is more prone to false positives.
The F1 score of 0.761 is lower than Logistic Regression, suggesting that this model is also unable to achieve a balance between precision and recall. The ROC AUC value of 0.898 is the lowest among all models, showing that the model's ability to distinguish between positive and negative classes is relatively weak.

Table 4.3

Model	Accuracy	Precision	Recall	F-1 Score	ROC-AUC
LR	**0.999**	**0.808**	**0.602**	**0.690**	**0.939**
DT	**0.999**	**0.729**	**0.796**	**0.761**	**0.898**
RF	**1.000**	**0.987**	**0.776**	**0.869**	**0.958**
XGB	**1.000**	**0.987**	**0.776**	**0.869**	**0.983**
CatBoost	**1.000**	**0. 988**	**0. 806**	**0.888**	**0.988**

The Random Forest model achieved a perfect accuracy score of 1.000, showing that the model is highly effective in predicting the target variable. The precision value of 0.987 and recall value of 0.776 is also relatively high, showing that the model is good at naming true positives and limiting false positives. The F1 score of 0.869 and ROC AUC of 0.958 suggest that the model has achieved a balance between precision and recall and is effective in distinguishing between positive and negative classes.
The XGBoost and CatBoost models have achieved equivalent results with perfect accuracy and high values for other metrics. Both models have high precision and recall values, showing that they are effective in finding true positives and limiting false positives. The F1 score values of 0.869 and 0.888 for XGBoost and CatBoost, respectively, say that both models have achieved a good balance between precision and recall. The ROC AUC values of 0.983 and 0.988 suggest that both models have a high ability to distinguish between positive and negative classes.
In summary, our experiments suggest that Random Forest, XGBoost, and CatBoost are the most effective models for our dataset based on the evaluation metrics. However, Logistic Regression and Decision Tree models also performed

reasonably well but with some limitations. The selection of a suitable model for a specific application would depend on the trade-off between model performance and computational efficiency. The model comparison is depicted in Fig 5.6

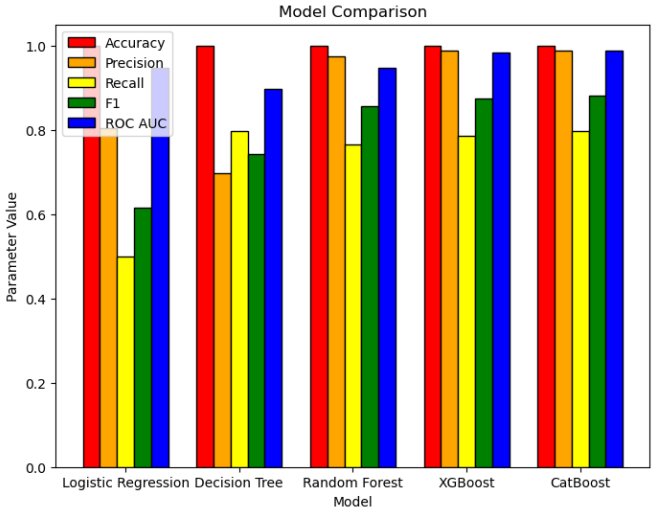

Fig. (4.4)

V. CONCLUSION

In the Paper, we have implemented different machine-learning models and compared the performance of the models on a classification task. The results show that all models achieved high accuracy, with Random Forest, XGBoost, and CatBoost achieving perfect accuracy.

When comparing the models based on other performance metrics, such as precision, recall, and F1 score, we can see that XGBoost and CatBoost outperformed the other models. However, it is important to note that the logistic regression model achieved high accuracy and ROC AUC score, showing that it was still able to effectively distinguish between the two classes despite having lower precision and recall scores.

Overall, the choice of model to use in each application depends on the specific requirements of the problem at hand. While some models may perform better in certain areas, such as XGBoost and CatBoost in the precision, recall, and F1 score, other models such as logistic regression can still supply superior performance on a classification task.

REFERENCES:

[1] S. P. Maniraj, A. Saini, S. Ahmed, and S. Sarkar, ‘‘Credit card fraud detection using machine learning and data science,’’ Int. J. Res. Appl. Sci. Eng. Technol., vol. 8, no. 9, pp. 3788–3792, Jul. 2021.

[2] The Nilson Report. Accessed: Sep. 27, 2021. [Online]. Available:https://nilsonreport.com/upload/content_promo/NilsonReport_Issue1209.pdf

[3] Emmanuel Ileberi, Yanxia Sun, Zenghui Wang, “Performance Evaluation of Machine Learning Methods for Credit Card Fraud Detection Using SMOTE and AdaBoost”. IEEE Access, 9:165286 - 165294, 2021 DOI: 10.1109/ACCESS.2021.3134330

[4] Huang Tingfei, Cheng Guangquan, Huang Kuihua,” Using Variational Auto Encoding in Credit Card Fraud Detection”, IEEE Access, 8:149841-149853,2020 DOI: 10.1109/ACCESS.2020.3015600

[5] Kuldeep Randhawa, Chu Kiong Loo, Manjeevan Seera; Chee Peng Lim, Asoke K. Nandi, “Credit Card Fraud Detection Using AdaBoost and Majority Voting”. IEEE Access, 6:14277 - 14284, 2018 DOI: 10.1109/ACCESS.2018.2806420

[6] Altyeb Altaher Taha; Sharaf Jameel Malebary, “An Intelligent Approach to Credit Card Fraud Detection Using an Optimized Light Gradient Boosting Machine” IEEE Access, 8:25579 - 25587, 2020DOI: 10.1109/ACCESS.2020.2971354