Proceedings of CONF-MLA 2025 Symposium: Intelligent Systems and Automation: AI Models, IoT, and Robotic Algorithms
DOI: 10.54254/2755-2721/2025.LD26477

A Comparative Study on Supervised Machine Learning Algorithms for Credit Card Transaction Fraud Detection

Emma Li

Cypress Bay High School, Weston, USA
jiayie.li11@gmail.com

Abstract. The global cost of credit card fraud continues to rise, driven by the increasingly concentrated and sophisticated attacks. This situation underscores the necessity for more effective detection and prevention methods. In response to the growing need for better fraud detection and prevention, machine learning has witnessed significant advancements in recent years. This paper provides an overview and comparison of various models. On one hand, there are traditional supervised learning models, such as Logistic Regression, Decision Trees, and Support Vector Machines (SVM). On the other hand, ensemble methods like Random Forest, Gradient Boosting, and XGBoost are also covered. Given the highly imbalanced nature of credit card fraud datasets, the study also examines the impact of the Synthetic Minority Over-sampling Technique (SMOTE) on classification performance. While SMOTE has been shown to improve a model's performance for weaker classifiers, its benefits for advanced ensemble methods remain less clear. Consequently, this paper will identify which models benefit most from oversampling and assess whether high-performing classifiers can mitigate the effects of imbalance without the need for data augmentation. When comparing the models' performances, Random Forest and XGBoost demonstrated superior performance both with and without SMOTE. Without SMOTE, two models, Logistic Regression and SVM, yielded high accuracy but near-zero performance on key classification metrics, highlighting their inability to effectively detect minority class instances.

Keywords: fraud detection, credit card, machine learning, imbalanced data

1. Introduction

Much of the success of E-commerce manifests in the credit card, a physical card which allows users to purchase goods and services using borrowed money, relying on the privacy of the credit card number to ensure security. Credit card fraud results in large financial losses [1]. It becomes important to create an efficient detection and prevention system. Machine learning solutions gained traction during this period and have since grown to replace outdated and ineffective rule-based systems [2]. Several of such machine learning models are the decision tree, random forest, logistic regression, gradient boosting, support vector machine, and XGBoost. Logistic regression is an easy-to-implement classification technique that works well with the classification of unknown data [3]. The decision tree is a structure of nodes and branches which can be used for classification through

DOI: 10.54254/2755-2721/2025.LD26477

decision rules inferred from the data [3]. Random forest is an ensemble model made up of multiple decision trees, giving it flexibility and improvements in accuracy [3]. Gradient boosting is an optimizing machine learning technique which addresses regression and classification problems; the method combines with a weak prediction model like and improves its predictive power through iterative improvements [4]. The Support Vector Machine algorithm is resistant to overfitting and has achieved high-generalization performance in problems involving classification [5]. Finally, XGBoost, short for Extreme Gradient Boosting, is an optimized and efficient version of gradient boosting which has great potential in fraud applications, as seen in a study by Mohammad Tayebi and Said El Kafhali on hyparameter tuning, outperforming other machine learning models [6]. This paper overviews and compares multiple models, covering traditional supervised learning models like Logistic Regression, Decision Trees, SVM, and ensemble methods such as Random Forest, Gradient Boosting, XGBoost. Given the highly - imbalanced credit card fraud datasets, it explores the impact of SMOTE on classification performance. This study uses publicly available Kaggle datasets. Six machine learning models (Logistic Regression, Decision Trees, Support Vector Machines, Random Forest, Gradient Boosting, and XGBoost) are trained and tested on Google Colab and a personal computer. The impact of the Synthetic Minority Over - sampling Technique (SMOTE) on classification performance is also examined. The models' performance is evaluated using precision, recall, F1 score, accuracy, and training time. By comparing different machine learning models and the effect of SMOTE, this research aims to identify high - performing classifiers that can balance predictive performance and training time. This helps in developing more efficient credit card fraud detection systems, which is crucial for businesses to avoid financial losses and maintain customer trust.

2. Methodology

2.1. Data collection

The hardware utilized for system implementation includes: 32 GB RAM, 1.82 TB SSD,AMD Ryzen 9 7950X 16-Core Processor, 16 GB VRAM.

The software employed consists of: Windows 11 Pro, Python 3.13.5, Google Colab, Jupiter Notebook

The dataset is publicly available on Kaggle: https://www.kaggle.com/datasets/kartik2112/fraud-detection/data. The dataset consists of two CSV files, one for model training and the other for model testing. The dataset is created by user kartik2112 and includes 23 columns, of which not all are relevant to fraud. It is based on simulated data generated using a Sparkov Data Generation tool by Brandon Harris. Only the training csv file was used, which consisted of 1289169 non-fraud cases and 7506 fraud cases before data cleaning. Experimentation was done on Google Colab and ran on a personal computer. This research predominantly relies on qualitative data, as it provides in-depth insights into the nature and characteristics of the phenomenon under study in the context of this research.

2.2. Proposed algorithm

Step 1- Import libraries

The comparison leverages the following essential tools:

Pandas: For data reading, manipulation, and analysis.

DOI: 10.54254/2755-2721/2025.LD26477

Python: Chosen for its variety of libraries, the programming language used to train and run all models.

Scikit-learn: A Python library for machine learning models and preprocessing tools.

Matplotlib and Seaborn: For data visualization.

Imblearn: Library for importing tools to upsample the minority class.

Step 2- Load and summarize the data

Step 3- Preprocess the data

The data will undergo cleaning, during which rows containing missing values will be removed. The following numerical data will also be normalized on the 0 to 1 scale: amt, lat, long, city_pop, merch_lat, and merch_long, ensuring that features with different scales do not disproportionately affect the model's learning.

Furthermore, the following rows will be removed from model training: Unnamed: 0, cc_num, first, last, street, city, state, zip, dob, and trans_num, as they don't correlate with fraud when visualized. The Trans_date_trans_time variable is extracted into three separate rows: month, day, and hour. By breaking down the timestamps, the model may learn patterns of fraud more effectively. Hours could show how fraud is more common at night while month can reveal seasonal or holiday trends in behavior. Then, trans_date_trans_time and unix_time were removed to avoid giving the model redundant information.

Label encoding was done for the following categorical variables: merchant, category, gender, job. This technique converts categorical variables into a numerical format that machine learning models can understand and is a common preprocessing practice.

Rather than using both the training and testing datasets, we separate features and targets within the training dataset. This approach allows the data to be utilized for both training and testing purposes.

Step 4- Mitigate effects of imbalanced data using SMOTE

SMOTE, short for Synthetic Minority Over-sampling Technique, is one of many over-sampling methods designed to address data imbalances in machine learning models, along with under-sampling, balancing class weight, and other novel techniques. Kumar et al. explored many of these approaches in their paper, and also introduced their innovative BDT approach [7]. For this comparison, SMOTE will be used, as it is an established method with a decent comparative performance. The performance of each model will be tested with and without SMOTE to determine where SMOTE may not be needed to enhance model results.

Step 5- Train one of six proposed models

Step 6- Classification using one of six proposed models

Step 7- Summarize results using confusion matrix

Step 8- Repeat for all 6 models

Results will be compared using four parameters: precision, recall, F1 score, and accuracy. In the following equations, TP, FP, TN, and FN represent true positive, false positive, true negative, and false negative, respectively. A true positive is when the model correctly classifies an instance as positive, a true negative when it correctly classifies an instance as negative. A false positive occurs when the model misclassifies an instance as positive, and a false negative when it misclassifies an instance as negative. In the context of these models, a positive indicates a fraud transaction, and a negative indicates a non - fraud transaction.

Precision is the ability of a model to identify true positives among all instances predicted as positive, with the equation $\text{Precision} = \frac{TP}{TP+FP}$ [5].

DOI: 10.54254/2755-2721/2025.LD26477

Recall is the ability of the model to identify true positives among all instances that are definitively positive, with the equation Recall = $\frac{TP}{TP+FN}$.

F1 score is the weighted mean of precision and recall [8], ranging from 0 to 1 and with the equation F1 Score = $\frac{2 \cdot precision \cdot recall}{precision+recall}$.

Accuracy measures the total number of correct predictions, both positive and negative, among all predictions made by the model [8], with the equation Accuracy = $\frac{TP+TN}{TP+TN+FP+FN}$.

In addition to these measurements, time taken for the model to train will also be taken into consideration.

3. Results and analysis

Without applying SMOTE, the training dataset consisted of 1011335 non - fraud cases and 6005 fraud cases. The testing dataset consisted of 257834 non - fraud cases and 1501 fraud cases. Below are the confusion matrices of models without the use of SMOTE.

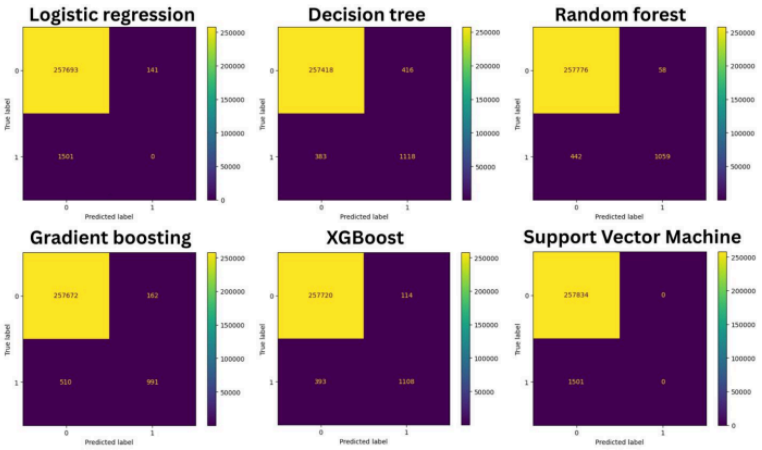

Figure 1. Confusion matrix for all models trained without SMOTE

Table 1. Classification report summary for all models trained without SMOTE

	Logistic Regression	Decision Tree	Support Vector Machine	Gradient Boosting	XGBoost	Random Forest
Precision	0	0.7288	0	0.8595	0.9067	0.9481
F1-Score	0	0.7367	0	0.7468	0.8138	0.8090
Recall	0	0.7448	0	0.6602	0.7382	0.7055
Accuracy	0.9936	0.9969	0.9942	0.9974	0.9980	0.9981
Training time (s)	0	7	420	180	1	120

When applying SMOTE with a sampling strategy of 0.5, the resampled training dataset was augmented to contain 1011335 non - fraud cases and 515667 fraud cases.

DOI: 10.54254/2755-2721/2025.LD26477

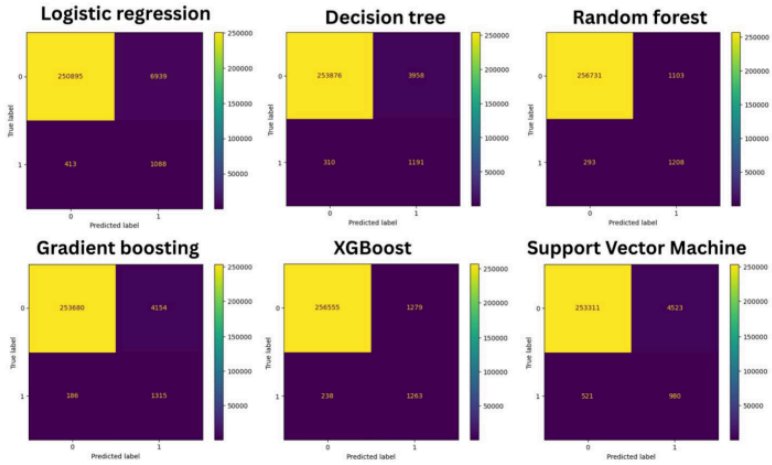

Figure 2. Confusion matrix for all models trained with SMOTE

Table 2. Classification report summary for all models trained with SMOTE

	Logistic Regression	Decision Tree	Support Vector Machine	Gradient Boosting	XGBoost	Random Forest
Precision	0.1355	0.2313	0.1781	0.2404	0.4969	0.5227
F1 Score	0.2284	0.3582	0.2798	0.3773	0.6248	0.6338
Recall	0.7249	0.7935	0.6529	0.8761	0.8414	0.8048
Accuracy	0.9717	0.9835	0.9806	0.9833	0.9942	0.9946
Training time (s)	5	21	25200	900	6	360

Arranged in ascending order of training times, the models are logistic regression, XGBoost, decision tree, random forest, gradient boosting, and SVM.

The times taken for logistic regression and XGBoost to train are the most comparable—in both Tables 1 and 2, XGBoost took only one second longer than logistic regression to train. When the dataset increased by 50%, none of the models scaled linearly with this increase. The training times of both gradient - boosting decision tree and random forest increased by approximately 200%, while that of SVM increased by around 5900%. Due to the significant time requirements for SVM training as compared to other models, it may not be applicable to real-world fraud scenarios.

A model that performs well without SMOTE is ideal for training. The reason is that preprocessing and handling additional data, such as those involved in SMOTE, place greater demands on the computing resources of the machine. This, in turn, increases the costs associated with using a machine - learning model. As can be observed from Tables 1 and 2, the use of SMOTE did not lead to an improvement in accuracy. While accuracy was high for logistic regression and SVM without the use of SMOTE in Table 1, a further look at the confusion matrixes in Figure 1 reveals that the accuracy was only high because the extremely imbalanced testing dataset allowed these models to categorize all cases as negative and still achieve a high outcome, evidenced again in Table 1 where precision, F1 score, and recall are all 0 for both models. This was proven to be the case in Table 2, where logistic regression and SVM achieved a higher precision, F1 score, and recall with more balanced training data through the usage of SMOTE, albeit at the cost of accuracy. Figure 2 additionally shows that the model predicted cases as positive with the help of SMOTE.

DOI: 10.54254/2755-2721/2025.LD26477

For this reason, having high accuracy is not the distinguisher of a good model. False negatives and false positives occur when using each model. However, false negatives are generally considered more costly for fraud detection. When the detection system fails to identify a fraudulent transaction, businesses may suffer significant financial losses. Moreover, the unrecovered funds can damage a company's reputation and erode customer trust. False positives can lead to poor customer experience and could incur operational costs associated with manual reviews of false alarms. It is in the company's best interest to minimize false negatives, even at the expense of some false positives, thus making recall the most important variable among them. However, since both errors are undesirable, it would also be helpful to investigate the model's F1 score, which balances the metrics of precision and recall.

In contrast to logistic regression and SVM models, decision tree, gradient boosting, XGBoost, and random forest all experienced a decline in precision and F1 - score when SMOTE was applied, while recall increased. This can be attributed to the model's tendency to over - predict positives during testing. The testing was conducted using severely imbalanced data, while balanced data was used for training. Using SMOTE, the greatest precision and F1 score was by the random forest model (0.5227 and 0.6338 respectively) while the greatest recall was achieved by gradient boosting (0.8761) although its precision was a low 0.2404, as shown in Table 2.

However, without the application of SMOTE, Table 1 indicates that XGBoost had the greatest recall and F1 - score, at 0.7382 and 0.8138 respectively, while the highest precision was obtained by random forest at 0.9481. These models achieved comparably similar scores, yet XGBoost had a faster training time in both SMOTE and non-SMOTE scenarios, putting it above random forest.

The next best performing model was gradient boosting. Although it exhibited a low precision and F1 score when used with SMOTE, its recall increased by a substantial 0.2159. Without SMOTE, the model performance was only slightly worse than XGBoost and random forest, however its training time was by far longer than both. The performance of gradient boosting can be compared to the decision tree model, the difference being that decision tree took a far shorter time to train.

Logistic regression and SVM performed the worst both with and without SMOTE, the only models with F1 scores below 0.30. While logistic regression has the fastest training time, it does not make up for its poor performance.

4. Conclusion

High accuracy alone is insufficient in evaluating models trained on highly imbalanced datasets. Weak classifiers, such as SVM and logistic regression, when trained without SMOTE, had difficulty identifying minority - class instances. As a result, despite reporting high accuracy, they exhibited low precision, recall, and F1 scores. The SMOTE application improved the weak classifiers' ability to identify minority classes, enhancing the precision, recall, and F1 scores of SVM and logistic regression.

Among the evaluated algorithms, XGBoost and Random Forest emerged as the top - performing classifiers. They are capable of balancing predictive performance and reasonable training time. XGBoost holds a slight edge over Random Forest, attributed to its higher recall and shorter training time. In contrast, SVM, while improved with SMOTE, incurred an extremely high computational cost. It has been observed that SMOTE generally increases recall but decreases precision in return, to the extent that the F1 score is also decreased. Therefore, it is recommended to explore novel methods for handling imbalanced datasets without resorting to oversampling and undersampling techniques.

DOI: 10.54254/2755-2721/2025.LD26477

The scope of the study only addressed a few machines and limited machine learning techniques. A future direction would be to analyze a wider range of training techniques and focus on more modern machines. In this paper, many weak classifiers such as SVM and logistic regression were tested. As illustrated in the discussion section, it is commonly believed that SMOTE has become outdated. This is because there is a wide range of state-of-the-art strong classifiers available, such as XGBoost, which do not benefit from oversampling. Thus, it would be beneficial to analyze such newer technologies to further advance machine learning for fraud detection.

References

[1] Xuan, S., Liu, G., Li, Z., Zheng, L., Wang, S., & Jiang, C. (2018, March). Random forest for credit card fraud detection. In 2018 IEEE 15th international conference on networking, sensing and control (ICNSC) (pp. 1-6). IEEE.

[2] Kurshan, E., Shen, H., & Yu, H. (2020, September). Financial crime & fraud detection using graph computing: Application considerations & outlook. In 2020 second international conference on transdisciplinary AI (transAI) (pp. 125-130). IEEE.

[3] Aditi, A., Dubey, A., Mathur, A., & Garg, P. (2022, July). Credit card fraud detection using advanced machine learning techniques. In 2022 Fifth International Conference on Computational Intelligence and Communication Technologies (CCICT) (pp. 56-60). IEEE.

[4] Simarmata, N., Wikantika, K., Tarigan, T. A., Aldyansyah, M., Tohir, R. K., Fauzi, A. I., & Fauzia, A. R. (2025). Comparison of random forest, gradient tree boosting, and classification and regression trees for mangrove cover change monitoring using Landsat imagery. The Egyptian Journal of Remote Sensing and Space Sciences, 28(1), 138-150.

[5] Chen, M. Y. (2011). Bankruptcy prediction in firms with statistical and intelligent techniques and a comparison of evolutionary computation approaches. Computers & Mathematics with Applications, 62(12), 4514-4524.

[6] Tayebi, M., & El Kafhali, S. (2025). A Novel Approach based on XGBoost Classifier and Bayesian Optimization for Credit Card Fraud Detection. Cyber Security and Applications, 100093.

[7] Kumar, S., Singh, S. K., & Nagar, V. (2024). BDT: A Novel Approach to Handle Imbalanced Data in Machine Learning Models. International Journal of Intelligent Systems and Applications in Engineering, 12(20s), 691-703.

[8] Afriyie, J. K., Tawiah, K., Pels, W. A., Addai-Henne, S., Dwamena, H. A., Owiredu, E. O., ... & Eshun, J. (2023). A supervised machine learning algorithm for detecting and predicting fraud in credit card transactions. Decision Analytics Journal, 6, 100163.