Machine Learning Modeling and Evaluation for E-commerce Transaction Fraud Detection

Zixiao Ding

{t330036016@mail.uic.edu.cn}

Beijing Normal–Hong Kong Baptist University, No. 18 Jinfeng Road, Tangjiawan, Zhuhai, Guangdong 519087, China

Abstract. With the continuous growth of e-commerce transaction volume, transaction fraud is becoming increasingly sophisticated in terms of concealment and harm. Addressing the characteristics of e-commerce transaction data—multi-dimensional features, complex patterns, and imbalanced categories—this paper first cleans and performs feature engineering on the raw data, constructing feature representations from dimensions such as time, transaction behavior, and account attributes. Then, under a unified process, logistic regression, gradient boosting tree, and multilayer perceptron (MLP) models are trained respectively, and a systematic comparison is conducted using 5-fold cross-validation based on six metrics: accuracy, precision, recall, F1, and ROC-AUC. Experimental results show that the gradient boosting tree outperforms the competition across all metrics (Accuracy=0.7431, Precision=0.7588, Recall=0.7120, F1=0.7343, ROC-AUC=0.8019, FLOPs= 4.16×10^{6}) with smaller fluctuations between folds, indicating a stronger ability to characterize complex fraud patterns and more robust generalization performance. Logistic regression shows relatively stable performance but its overall level is limited. MLP has high precision but low recall, resulting in a limited F1 score. The research findings provide quantitative basis for the selection of fraud detection models and the optimization of risk control strategies for e-commerce platforms.

Keywords: E-commerce; Fraud detection; Machine learning; Imbalanced classification; Risk control.

1 Introduction

In recent years, with the rapid development of internet and mobile payment technologies, e-commerce platforms have gradually become an important channel for residents' daily consumption. However, while the transaction volume continues to expand, various types of online fraud have also shown a rapid growth trend [3]. Fraudsters carry out illegal transactions through various means such as forging identity information, stealing accounts, and malicious refunds, causing serious economic losses to platform operators and consumers [13]. Traditional risk control systems based on manual rules and expert experience rely on manually designed rules, have long update cycles, limited adaptability, and are difficult to cope with complex and ever-changing fraud patterns [6, 10].

ICIAAI 2026, May 22-24, Singapore, Singapore

DOI 10.4108/eai.22-5-2026.2365358

With the development of big data and artificial intelligence technologies, fraud detection methods based on machine learning have gradually become a research hotspot. These methods, through modeling and analyzing historical transaction data, can automatically uncover potential fraud features and demonstrate high detection efficiency in practical applications [8,9,11]. However, e-commerce transaction data typically features high dimensionality, large scale, and extremely imbalanced class distribution, posing significant challenges to model training and performance optimization [1, 2, 4]. Therefore, improving the ability to identify minority class samples while maintaining detection accuracy has become an important research issue.

Based on this background, this paper focuses on the task of e-commerce transaction fraud detection, systematically studying the application effects of different machine learning models on open-source e-commerce transaction datasets. Through experimental comparative analysis of model performance and stability, it provides technical support for risk identification and risk control strategy optimization for e-commerce platforms.

2 Dataset and Data Analysis

2.1 Data Sources

The data employed in this experiment are of two parts, the main data comprising around 1.47 million records of transactions, which are to be utilized in the training and testing of the model, and the independent test data which comprises around 23, 000 records of transactions, which are to be utilized in testing the generalization aptitude of the model to unknown data.

The original data is made up of multi-dimensional transaction data such as transaction amount, payment method, product category, amount of purchase, user age, type of device, period of account registration, time of transaction, and address of transaction. It also contains a fraud label field, Is Fraudulent with 0 being a regular transaction and 1 being a fraudulent transaction. This discipline is the target variable used in modeling in this paper. The characteristics of the data include information on user behavior, transaction attribute behavior, and device information, which presents an abundant information base regarding fraud behavior modeling. Figure 1 presents the results in the form of data visualization.

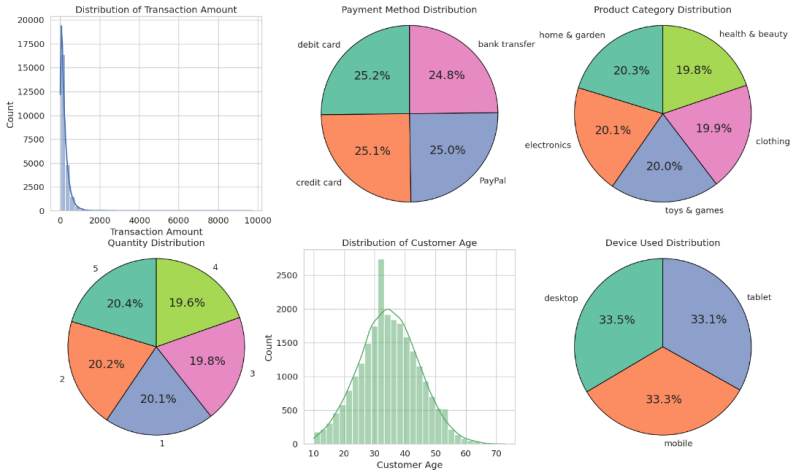

Fig. 1. Data Visualization Results (Picture credit: Original)

2.2 Data Preprocessing

Data Cleaning. Prior to model training, systematic cleaning of the raw data had been done. Fields that are not directly predictive, like transaction numbers, customer numbers, and IP addresses, were eliminated in the preprocessing process. At the same time, standardized conversion of time fields took place, the layout of abnormal records occurred or was eliminated, and blank values were verified. Statistical outcomes demonstrate that the cleansed data are of good integrity, and it does not have massive values missing.

Feature Construction. In a bid to enhance the capability of the model in characterizing the possible patterns of fraud, this paper further builds the derived features upon the original features. Transaction weekday and transaction month information are first extracted at the transaction time in order to describe the periodicity of user behavior. Second, the consistency attribute between shipping address and billing address is also added to represent the presence or absence of abnormal matches in terms of transaction information. Moreover, the time period of the transaction is further divided to help the model to detect the abnormal time-related behavior.

The Standardization and Encoding of Features. The differentiated processing strategy used in this paper depends on the nature of different types of features. In the case of numerical characteristics like the amount of transactions and the time of account as well as other equivalent features, a standardization technique applies to normalize the effect of dimensional variation during model training. In features that are categorical like payment method, the type of product, and the type of device, numerical transformation is performed by one-hot encoding. Categories of information can be retained in such a way that spurious order relationships are not introduced. All features are then encoded and standardized and then transformed into machine learning model input, which is in numerical form.

Data Imbalance Processing. Due to a significant class imbalance problem in the dataset, i.e., the number of normal transaction samples is much higher than the number of fraudulent samples, this paper introduces a class weight mechanism during model training. This mechanism differentiates the contribution of different class samples to the loss function, assigning higher penalty weights to the minority class of fraudulent samples to enhance the model's attention to fraudulent samples.

Let the sample set be $\{(x_i, y_i)\}_{i=1}^{N}$, where $y_i \in \{0,1\}$ denotes the class label. When $y_i = 1$, the sample is a fraudulent instance; when $y_i = 0$, the sample is a normal instance. In the subsequent model training phase, this paper introduces a class weight coefficient w_c into the original loss function to construct a weighted loss function:

$$\mathcal{L} = -\sum_{i=1}^{N} w_{y_i}[y_i \log(p_i) + (1 - y_i)\log(1 - p_i)] \tag{1}$$

where p_i denotes the predicted probability that sample x_i belongs to the fraudulent class, and w_{y_i} is the weight coefficient corresponding to the respective class. For minority-class fraudulent samples, a larger weight value is assigned so that the model incurs a greater penalty when misclassifying fraudulent samples, thereby placing more emphasis on learning fraud-related features during the parameter optimization process.

3 Model

Logistic Regression is a discriminative probabilistic model designed for classification tasks[7], with the objective of directly modeling the conditional distribution $P(y \mid \mathbf{x})$. Let the feature vector of a sample be $\mathbf{x} \in \mathbb{R}^d$ and the label $y \in \{0,1\}$. The model computes the score of a sample belonging to the positive class via a linear predictor $z = \mathbf{w}^\top \mathbf{x} + b$, where $\mathbf{w} \in \mathbb{R}^d$ is the weight vector to be learned, $b \in \mathbb{R}$ is the bias term, and $\mathbf{w}^\top \mathbf{x}$ denotes the weighted sum of features. Subsequently, Logistic Regression applies the sigmoid function $\sigma(\cdot)$ to map the real-valued score z to the interval $[0,1]$, yielding the probability output $p = P(y = 1 \mid \mathbf{x})$. From a statistical modeling perspective, its fundamental assumption is that the log-odds (log-odds) are linearly related to the feature vector, i.e., $\log \frac{p}{1-p} = z$. Given a training set $\{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$, where n is the number of samples and y_i is the groundtruth label of the i-th sample, $p_i = P(y = 1 \mid \mathbf{x}_i)$ is the model's predicted probability for that sample. The parameters $(\mathbf{w}, b)$ are typically learned by maximizing the conditional likelihood, which is equivalent to minimizing the negative log-likelihood (cross-entropy) loss. This loss function is convex with respect to $\mathbf{w}$ and b under standard settings, and therefore admits favorable interpretability and stability in the optimization sense.

$$z = \mathbf{w}^\top \mathbf{x} + b \tag{2}$$

$$p = P(y = 1 \mid \mathbf{x}) = \sigma(z) = \frac{1}{1 + \exp(-z)} \tag{3}$$

$$\log \frac{p}{1-p} = \mathbf{w}^{\top}\mathbf{x} + b \tag{4}$$

$$\mathcal{L}(\mathbf{w}, b) = -\sum_{i=1}^{n} [y_i \log p_i + (1 - y_i) \log(1 - p_i)] \tag{5}$$

Gradient Boosting Decision Trees (GBDT) is an additive ensemble model whose core idea is to minimize empirical risk iteratively in the function space [5]. Let the input be $\mathbf{x}$, the model output be $F(\mathbf{x})$, and the loss function be $\ell(y, F(\mathbf{x}))$, where y is the supervision signal (which can be a class label or a continuous value). GBDT represents the final prediction as the sum of multiple base learners: $F_M(\mathbf{x}) = \sum_{m=1}^{M} \nu f_m(\mathbf{x})$, where M is the number of boosting iterations (also interpretable as the number of trees), $f_m(\mathbf{x})$ is the base learner obtained at the m-th round, typically a regression tree, and $\nu \in (0,1]$ is the learning rate used to shrink the contribution of each newly added base learner. The training process starts from an initial model $F_0(\mathbf{x})$; at each round, the pseudo-residual under the current model F_{m-1} is first computed as r_{im}, defined as the negative gradient of the loss function with respect to the model output:

$$r_{im} = -\left.\frac{\partial \ell\big(y_i, F(\mathbf{x}_i)\big)}{\partial F(\mathbf{x}_i)}\right|_{F=F_{m-1}} \tag{6}$$

where $\mathbf{x}_i$ and y_i are the features and label of the i-th sample, respectively. r_{im} characterizes the direction and magnitude by which the model output should be adjusted at sample point $\mathbf{x}_i$ in order to most rapidly reduce the loss. A regression tree f_m is then fitted to approximate the mapping relationship of $\{(\mathbf{x}_i, r_{im})\}$, and the new model is obtained via an additive update: $F_m(\mathbf{x}) = F_{m-1}(\mathbf{x}) + \nu f_m(\mathbf{x})$. This iterative mechanism is equivalent to performing gradient descent in the function space, enabling the model to progressively approximate complex target functions through multi-round boosting.

$$F_M(\mathbf{x}) = \sum_{m=1}^{M} \nu f_m(\mathbf{x}) \tag{7}$$

$$r_{im} = -\left.\frac{\partial \ell\big(y_i, F(\mathbf{x}_i)\big)}{\partial F(\mathbf{x}_i)}\right|_{F=F_{m-1}} \tag{8}$$

$$F_m(\mathbf{x}) = F_{m-1}(\mathbf{x}) + \nu f_m(\mathbf{x}) \tag{9}$$

Multilayer Perceptron (MLP) is the most fundamental form of feedforward neural networks, consisting of an input layer, one or more hidden layers, and an output layer[12]. It achieves the modeling of complex mappings through affine transformations between layers combined with nonlinear activation functions. Let the input feature vector be $\mathbf{x} \in \mathbb{R}^d$, and let $\mathbf{h}^{(0)} = \mathbf{x}$denote the representation of layer 0 (the input layer). For the l-th hidden layer ($l = 1, \ldots, L-1$), its output is represented as $\mathbf{h}^{(l)} \in \mathbb{R}^{d_l}$. By applying a weight matrix $\mathbf{W}^{(l)} \in \mathbb{R}^{d_l \times d_{l-1}}$and a bias vector $\mathbf{b}^{(l)} \in \mathbb{R}^{d_l}$to perform a linear transformation on the representation of the previous layer

$\mathbf{h}^{(l-1)}$, followed by an element-wise nonlinearity via an activation function $\phi(\cdot)$, the output of each layer is obtained. The output layer produces the final prediction according to the task type.

$$\mathbf{h}^{(0)} = \mathbf{x} \tag{10}$$

$$\mathbf{h}^{(l)} = \phi\left(\mathbf{W}^{(l)}\mathbf{h}^{(l-1)} + \mathbf{b}^{(l)}\right), l = 1, \ldots, L-1 \tag{11}$$

$$p = P(y = 1 \mid \mathbf{x}) = \sigma\left(\mathbf{w}^{\top}\mathbf{h}^{(L-1)} + b\right) \tag{12}$$

4 Experiment

4.1 Experimental Results and Analysis

Table 1. Results of 5-fold Cross-validation

Model	Accuracy	Precision	Recall	F1	ROC-AUC	FLOPs
Logistic Regression	0.6972	0.7060	0.6784	0.6912	0.7710	5.82×10^{3}
Gradient Boosting	0.7431	0.7588	0.7120	0.7343	0.8019	4.16×10^{6}
MLP	0.7046	0.7298	0.6498	0.6873	0.7777	2.39×10^{7}

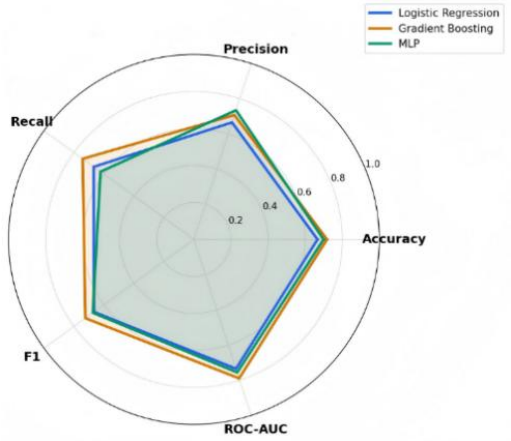

Fig. 2. Radar Chart of Models (Picture credit: Original)

In Figure 2, five points depict the five evaluation dimensions that include Precision, Recall, Accuracy, ROC-AUC, and F1. These three models are shown with their general performance using polygon outlines and areas filled by varying colors, and the greater the polygon area of the coverage depicts the higher overall performance. Comparing areas, Gradient Boosting (orange) covers the greatest area of polygon hence is the best performing model in nearly all respects; MLP (green) is second in area; Logistic Regression (blue) has smallest area but regular outline. The additional discussion of shape features shows that Gradient Boosting has more prominent outward expansion in terms of both Recall and ROC-AUC directions and still shows high values in F1 and Accuracy dimension with an overall balanced performance display with

higher Recall. In Precision, MLP has a slight outward convexity and in Recall, it has a large inward contraction resulting in an asymmetrical, so-called, convex Precision, concave Recall, which is a direct reflection of high precision but inadequate recall. The shape of Logistic Regression is similar to a standard pentagon, and the variations remain insignificant with respect to the dimensions, proving the more equalized distribution of metrics that is typical of a linear model. Altogether, the three contour lines are limited to the nearest proximity at the majority of the vertices, which suggests that the differences between the three model sets are rather minor, with the performance disparities primarily located in the two crucial areas, which are Recall and the F1 score that it determines.

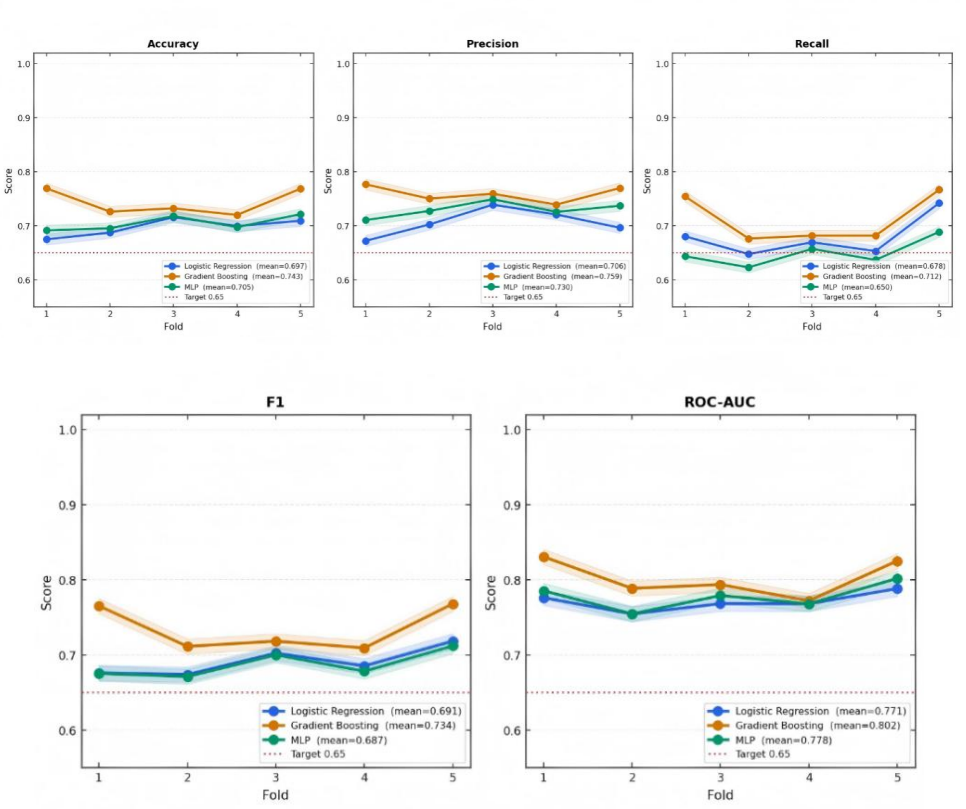

Fig. 3. 5-Fold Cross-Validation Metrics per Fold (Picture credit: Original)

The performance measure of each of the three models with 5-fold cross was depicted in Figure 3 as the number of folds against the performance measure. Horizontal Axis is Fold (1 -5) and Vertical Axis are Accuracy, Precision, Recall, F1, and ROC-AUC respectively. These three lines represent Logistic Regression, Gradient Boosting and MLP. The line with red dashes represents the convergence point of 0.65, and each curve has its band of light color representing the uncertainty between the folds. In general, in all three models, the score of performance is always higher than 0.65 in each of the 5 folds and 5 metrics, without any cases of falling lower. This implies that the training process of the model can be considered to be reliably convergent and repeatable in the current conditions, unlike the large variation in values of various folds

when class balancing is not implemented. It can be further compared that Gradient Boosting has a high overall performance in four dimensions: Accuracy (mean 0.743), Precision (0.759), F1 (0.734), and ROC-AUC (0.802) with little variation in between folds showing that it is very stable. Peak in the first fold and a decrease in ROC-AUC subplot and stabilization are the evolutionary traits of the adaptive adjustment of ensemble methods to alternative data partitions by additive iteration. The curve of the Logistic Regression is the smoothest, on the whole is horizontally oriented and the difference between folds is the least which corresponds to the theory that linear models are not very sensitive to the distortions in data distribution. Nevertheless, its average Recall (0.678) is weak compared to Gradient Boosting, which indicates that there is a limit to the linear boundary that can be used to describe the complex patterns of fraud. MLP shows the strongest variations between folds in the Recall subgraph and folds 3 and 4 especially have dips. This is consistent with the fact that neural networks do not behave smoothly (when trained under stochastic optimization, like Adam and mini-batch stochastic gradients) with subsets of data in their training path. Nevertheless, in Precision between folds, MLP works best when compared to Gradient Boosting and comes in second, closely after it, with a mean of 0.730 and second close to 0.759. Based on the trends of the subgraphs, the most important indicator of the differentiation in the three models is Recall. Gradient Boosting has clearly shown a better performance in Fold 1, and Logistic Regression is more like MLP which is the primary origin of the F1 variation.

4.2 Discussion

As is apparent in Table 1, Figure 2, and Figure 3, the Gradient Boosting has the largest values in all six metrics which are: Accuracy (0.7431), Precision (0.7588), Recall (0.7120) F1 (0.7343), ROC-AUC (0.8019) and FLOPs(4.16×10^6). Gradient Boosting was more accurate, more precise, more recall, and more F1 score and more ROC-AUC by 0.0459, 0.0528, 0.0622, 0.0470, 0.0242, and by 0.0385, 0.0290, 0.0336, 0.0309, respectively, than Logistic Regression, MLP, respectively. The largest gain was in the recall and F1, which are in line with the largest increase in differentiation in recall in the line graph, and also in displayed radar chart as the broadening and highest coverage area of Gradient Boosting along the recall and ROC-AUC directions. Since fraud detection generally gives more emphasis on the cost of a false negative, when the recall is larger, and therefore the F1 score, it means a greater coverage of the risks, and a better designation of overall performance. Subsequently, the Gradient Boosting can be regarded as the best one in the conditions of the experiment.

Logistic Regression exhibits a "stable but limited capability" characteristic: its various metrics are relatively balanced, with minimal fluctuations across all intervals, the smoothest line graph, and a more regular radar profile, reflecting the stability of the linear discriminant structure under different data partitions; however, it lags behind Gradient Boosting in all metrics, especially in Recall and F1, indicating that the linear decision boundary has an upper limit in characterizing complex fraud patterns.

MLP, on the other hand, presents a pattern of "higher Precision but weaker Recall and more pronounced fluctuations between intervals": Precision reaches 0.7298, but Recall is only 0.6498, resulting in the lowest F1 (0.6873), consistent with the concave Recall in some intervals of the line graph and the "convex Precision, concave Recall" pattern in the radar graph. This indicates that MLP tends to conservatively classify positive classes under the current threshold decision, thus reducing false positives but sacrificing coverage of positive classes. Although its ROC-

AUC (0.7777) is slightly higher than Logistic Regression, showing that its ranking ability is not weak, it failed to effectively translate into higher Recall and F1 scores, suggesting that the key differences lie more in the threshold-sensitive region and the different handling of boundary samples.

On the whole, 5-fold cross-validation confirmed at the same time convergence and reproducibility of the three models (all measures at each fold were necessarily larger than the threshold) and indicated that the distinction between models was not comprehensive, but concentrated rather around Recall and the derived F1 dimension. The implications of these results are that when a model is evaluated and selected in a cost-sensitive task (e.g. fraud detection, etc.), PrecisionRecall tradeoff should not be considered an outcome measurement as it does not favor bias judgements of model performance, and threshold-independent metrics (ROC-AUC) and threshold-related measures (Precision/Recall/F1) should be presented so that no one-sided judgment about model performance can be made using any single metric.

5 Conclusions

The paper will discuss the fraud detection and make a comparison of three representative models including Logistic Regression, gradient boosting and MLP through 5-fold cross-validation in a common state of data processing and assessment. It has been demonstrated through experimental results that all three models have balanced convergence on all metrics. In addition, the fold metrics of all fold metrics exceed the preset thresholds, meaning that the developed training and evaluation procedure has good repeatability and strength. Comparative analysis also reveals that Gradient Boosting has the highest performance regarding all metrics but had less variation when passing the transition, indicating greater generalization stability and overall discriminative capability. Logistic Regression demonstrated a more equalized dispersion across metrics but less on the whole, which offers a saline bottom. Precision favored MLP to some extent but Recall was comparatively lower, which led to low F1 and elevated variation across transitions. On the whole, the variations between models are largely localized in Recall and the F1 dimension, which implies that under high-cost conditions such as fraud-detecting Accuracy or AUC are not sufficient to describe the performance of models. The Precision-Recall tradeoff should be considered in full and overall detection performance should be assessed with respect to such measures as F1. The conclusions give empirical evidence of the experiment and the decision-making references at the metric level on the model choice and performance comprehension of comparable risk recognition work.

References

[1] Ali, A., Abd Razak, S., Othman, S. H., Eisa, T. A. E., Al-Dhaqm, A., Nasser, M., ... & Saif, A. (2022). Financial fraud detection based on machine learning: a systematic literature review. Applied Sciences, 12(19), 9637.

[2] Benchaji, I., Douzi, S., El Ouahidi, B., & Jaafari, J. (2021). Enhanced credit card fraud detection based on attention mechanism and LSTM deep model. Journal of Big Data, 8(1), 151.

[3] Btoush, E. A. L. M., Zhou, X., Gururajan, R., Chan, K. C., Genrich, R., & Sankaran, P. (2023). A systematic review of literature on credit card cyber fraud detection using machine and deep learning. PeerJ Computer Science, 9, e1278.
[4] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. Journal of Artificial Intelligence Research, 16, 321–357.
[5] Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. Annals of statistics, 1189-1232.
[6] Ileberi, E., Sun, Y., & Wang, Z. (2022). A machine learning based credit card fraud detection using the GA algorithm for feature selection. Journal of Big Data, 9(1), 24.
[7] McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics, 5(4), 115-133.
[8] Mienye, I. D., & Sun, Y. (2023). A machine learning method with hybrid feature selection for improved credit card fraud detection. Applied Sciences, 13(12), 7254.
[9] Mutemi, A., & Bacao, F. (2023). A numeric-based machine learning design for detecting organized retail fraud in digital marketplaces. Scientific Reports, 13(1), 12499.
[10] Mutemi, A., & Bacao, F. (2024). E-commerce fraud detection based on machine learning techniques: Systematic literature review. Big Data Mining and Analytics, 7(2), 419–444.
[11] Rodrigues, V. F., Policarpo, L. M., da Silveira, D. E., da Rosa Righi, R., da Costa, C. A., Barbosa, J. L. V., ... & Arcot, T. (2022). Fraud detection and prevention in e-commerce: A systematic literature review. Electronic Commerce Research and Applications, 56, 101207.
[12] Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. nature, 323(6088), 533-536.
[13] Zeng, Q., Lin, L., Jiang, R., Huang, W., & Lin, D. (2025). NNEnsLeG: A novel approach for e-commerce payment fraud detection using ensemble learning and neural networks. Information Processing & Management, 62(1), 103916.