Accredited SINTA 2 Ranking
Decree of the Director General of Higher Education, Research, and Technology, No. 158/E/KPT/2021
Validity period from Volume 5 Number 2 of 2021 to Volume 10 Number 1 of 2026

Published online at: **http://jurnal.iaii.or.id**

JURNAL RESTI
(Rekayasa Sistem dan Teknologi Informasi)
Vol. 8 No. 4 (2024) 563 - 570 e-ISSN: 2580-0760

Optimizing Multilayer Perceptron with Cost-Sensitive Learning for Addressing Class Imbalance in Credit Card Fraud Detection

Wowon Priatna[1]*, Hindriyanto Dwi Purnomo[2], Ade Iriani[3], Irwan Sembiring[4], Theophilus Wellem[5]
[1]Informatic Department, Computer Science Faculty, Universitas Bhayangkara Jakarta Raya, Jakarta, Indonesia
[1,2,3,4,5]Department Information Technology, Universitas Kristen Satya Wacana, Salatiga, Indonesia
[1]wowon.priatna@dsn.ubharajaya.ac.id, [1]982023018@student.uksw.edu, [2]hindriyanto.purnomo@uksw.edu, [3]ade.iriani@uksw.edu,[4]irwan@uksw.edu, [5]theophilus.wellem@uksw.edu

Abstract

The increasing use of credit cards in global financial transactions offers significant convenience for consumers and businesses. However, credit card fraud remains a major challenge due to its potential to cause substantial financial losses. Detecting credit card fraud is a top priority, but the primary challenge lies in class imbalance, where fraudulent transactions are significantly fewer than non-fraudulent ones. This imbalance often leads to machine learning algorithms overlooking fraudulent transactions, resulting in suboptimal performance. This study aims to enhance the performance of Multilayer Perceptron (MLP) in addressing class imbalance by employing cost-sensitive learning strategies. The research utilizes a credit card transaction dataset obtained from Kaggle, with additional validation using an e-commerce transaction dataset to strengthen the robustness of the findings. The dataset undergoes preprocessing with RUS and SMOTE techniques to balance the data before comparing the performance of baseline MLP models to those optimized with cost-sensitive learning. Evaluation metrics such as accuracy, recall, F1 score, and AUC indicate that the optimized MLP model significantly outperforms the baseline, achieving an AUC of 0.99 and a recall of 0.6. The model's superior performance is further validated through statistical tests, including Friedman and T-tests. These results underscore the practical implications of implementing cost-sensitive learning in MLPs, highlighting its potential to significantly enhance fraud detection accuracy and offer substantial benefits to financial institutions.

Keywords: optimizing multilayer perceptron; fraud detection; class imbalance; cost-sensitive learning; credit card

How to Cite: W. Priatna, Hindriyanto Dwi Purnomo, Ade Iriani, Irwan Sembiring, and Theophilus Wellem, "Optimizing Multilayer Perceptron with Cost-Sensitive Learning for Addressing Class Imbalance in Credit Card Fraud Detection", *J. RESTI (Rekayasa Sist. Teknol. Inf.)*, vol. 8, no. 4, pp. 563 - 570, Aug. 2024.
DOI: https://doi.org/10.29207/resti.v8i4.5917

1. Introduction

The increased use of credit cards in global financial transactions has provided significant convenience for consumers and businesses[1]. However, this development also presents a major challenge in the form of credit card fraud[2]. Such fraud can cause substantial financial losses for financial institutions and consumers, making credit card fraud detection (CCFD) a top priority to minimize these losses[3].

Class imbalance in classification data is a significant challenge in CCFD[4]. In the context of CCFD, fraudulent transactions are considerably fewer compared to non-fraudulent ones[5]. This imbalance often leads to machine learning models ignoring fraudulent transactions due to the dominance of non-fraudulent transactions in the training data[6]. Consequently, the resulting models perform poorly in detecting rare but critical fraud transactions[7].

Various techniques have been employed to mitigate class imbalance in credit card fraud detection (CCFD). Resampling techniques, such as SMOTE and Random Under Sampling (RUS), have been widely adopted to balance the data distribution[7]. SMOTE generates synthetic samples for the minority class to enhance its representation, though it may lead to overfitting in certain scenarios[5]. Additionally, ensemble methods like Random Forest, XGBoost, AdaBoost, and Easy Ensemble have also been proposed as effective strategies to improve classification performance. These methods work by combining multiple models to better detect minority classes, but they come with increased computational complexity and the need for extensive hyperparameter tuning, which can be challenging and

Received: 11-07-2024 | Accepted: 26-08-2024 | Published Online: 28-08-2024

may impact the model's overall effectiveness if not properly managed[8]. Recent studies [9] have also explored hybrid techniques, such as combining SMOTE with Generative Adversarial Networks (GANs), to further enhance the effectiveness of resampling strategies. Hybrid SMOTE-GAN techniques have shown promising results in addressing class imbalance by generating more realistic fraud samples while reducing the risks associated with overfitting. However, these techniques still face challenges such as the complexity of training GAN models and the potential for introducing noise through SMOTE, which can reduce the overall effectiveness of the approach.

One approach involves the use of cost-sensitive learning (CSL), which introduces different penalties or costs for misclassification errors depending on the importance of the class[10]. Studies have shown that CSL improves performance in handling class imbalance in various domains, including medical data[11]. Additionally, CSL can manage high-dimensional data and address class imbalance by adaptively adjusting the loss function, thus bridging the distribution between classes[10].

Multilayer Perceptron (MLP) is a type of neural network commonly used in various classification tasks, including fraud detection[12]. Although MLP can model non-linear relationships in data, it has several weaknesses in handling class imbalance issues[13]. These weaknesses include overfitting to the majority class, underfitting to the minority class, and the need for complex hyperparameter tuning to achieve optimal performance[14].To address these issues, this research aims to optimize MLP to handle class imbalance by utilizing CSL techniques and advanced weighting methods.

Compared to these approaches, the proposed method leverages the non-linear modeling capabilities of MLP and enhances it with CSL. CSL specifically adjusts the loss function by assigning higher penalties to misclassification errors in the minority class, thereby improving the model's sensitivity to fraudulent transactions without the risk of overfitting. This integration offers a more balanced and accurate detection mechanism for credit card fraud in highly imbalanced datasets.

The novelty of this research lies in developing a more effective method for optimizing MLP to address class imbalance and offering a better solution compared to previous approaches. This study presents a novel approach that integrates CSL with MLP, providing a comprehensive analysis of its performance. The main contributions of this research include the development of a novel method for optimizing MLP to handle class imbalance and providing a comprehensive analysis of the performance of MLP optimized with CSL techniques. This study also offers practical guidelines for financial practitioners in implementing more accurate fraud detection models and significantly improving the accuracy of CCFD.

2. Research Methods

This research addresses class imbalance in credit card transaction data by optimizing MLP through weight adjustments using Cost-Sensitive Learning (CSL).

This research uses a credit card transaction dataset from Kaggle in September 2013[15]. The dataset contains 31 variables, with the target classes being fraud, comprising 492 records, and non-fraud, comprising 284,315 records. This imbalance leads to a significant challenge in training effective models for fraud detection. Figure 1 shows the distribution between the fraud and non-fraud classes.

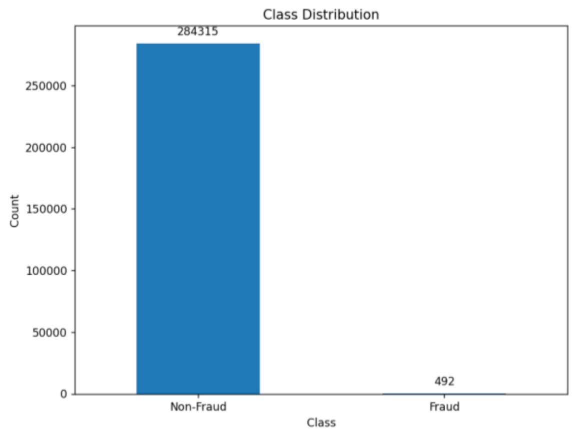

Figure 1. Visualization Data

In addition to using the credit card fraud dataset, this research employs another dataset to test the reliability of the proposed MLP model optimized with cost-sensitive learning weighting. This additional dataset includes e-commerce transactions containing fraud and credit card transaction data from January to April 2022[16]. Table 1 provides information on the credit card dataset from September 2013 and the e-commerce fraud transactions dataset.

Table 1. Dataset Information

Dataset	Sample	Fraud	Ratio
Credit Card 1	284807	492	0.0017
E-Commerce	23634	1222	0.0545
Credit Card 2	1048575	140473	0.1547

The preprocessing stage ensures that the data is clean and ready for modelling. The preprocessing steps begin with removing records with missing values. In this research, out of a total of 284,807 records, no records were found to be missing after detecting missing data. The next step is to separate the features and labels, consisting of 28 features and 1 class label. The target label 0 represents non-fraud, and the value 1 represents fraud. The final step is data normalization to ensure that the feature types support MLP modelling. Data normalization is performed using Z-score standardization, as shown in Equation 1[11].

$$X_{std} = \frac{X-\mu}{\sigma} \quad (1)$$

Random Under Sampling (RUS) is a simple technique used to address class imbalance in datasets by reducing the size of the majority class[17]. This method randomly selects a subset of data from the majority class to reduce its number, making it equivalent to the minority class. RUS helps balance class distribution and can potentially improve the performance of machine learning models by focusing more on the minority class[18].

Synthetic Minority Over-sampling Technique (SMOTE) is a sophisticated method used to handle class imbalance in machine-learning datasets[19]. Instead of simply copying existing samples, SMOTE generates new synthetic samples for the minority class by interpolating between current minority samples [20]. This approach boosts the representation of the minority class, making machine learning models better at identifying patterns specific to that class.

A Multilayer Perceptron (MLP) is a type of artificial neural network that consists of at least three layers of nodes: the input layer, one or more hidden layers, and the output layer. This layered configuration allows the MLP to perform information processing tasks with higher complexity, as each additional layer can capture more features and patterns in the processed data[21]. This process ensures that the input data is sequentially processed through each layer until it reaches the final output layer. With these characteristics, the MLP can be applied in various fields, including pattern recognition, classification, and prediction[22]. The stages of MLP are [23]: Initialization of Weights (w) and Biases (b): Randomly initialize weights and biases using Equation 1; Forward Propagation: Input data is forwarded to the hidden layer and output using Equations 2 and 3; Loss Function Calculation: Use binary cross-entropy for binary classification using Equation 4; Backpropagation: Calculate the gradient of the loss function concerning weights and biases using Equations 5 and 6; Weight and Bias Update (Gradient Descent): Update weights and biases using Equations 7 and 8; Model Evaluation: Evaluate the model using Equations 11 – 15; Fine-tuning and Hyperparameter Optimization: Improve the model by adjusting the learning rate, hidden layers, number of neurons, and activation functions.

$$w_{ij}^{(i)} dan\ b_j^{(i)} \tag{2}$$

i is the layer index, i is the neuron index for the input, and j is the neuron index in the next layer.

$$z^{(1)} = W^{(1)}.a^{(i-1)} + b^{(i)} \tag{3}$$

$$a^{(l)} = f(z^{(l)}) \tag{4}$$

$$L = -\frac{1}{m}\sum_{i=1}^{m}(yi\log(\bar{\Upsilon}i) + (1-yi)\log(1-\bar{\Upsilon}i)) \tag{5}$$

L represents the total loss value, m represents the number of data samples in the dataset, yi is the actual target value for the i and $\bar{\Upsilon}i$ is the predicted probability value for the i-th data point generated by the model.

$$\partial^{(L)} = \propto^{L} - y \tag{6}$$

$$\partial^{(l)} = (W^{l+1)})^T \delta^{(l+1)}.f'(Z^{(1)}) \tag{7}$$

$$W^{(l)} := W^{l} - \propto \frac{\partial L}{\partial W^{(l)}} \tag{8}$$

$$b^{(l)} := b^{(l)} - \propto \frac{\partial L}{\partial b^{l}} \tag{9}$$

$\partial^{(L)}$ is the gradient of the loss function, $(W^{l+1)})^T\delta^{(l+1)}$ is the weight matrix connecting layer $l+1$. $\dot{f}(z^l)$ is the derivative of the activation function.

Cost-Sensitive Learning is an approach in machine learning that considers the costs of different misclassification errors. Cost-sensitive learning (CSL) has two approaches in binary classification to maximize weights: the Weighted Loss Function and the Weighted Cross-Entropy Loss. The Weighted Loss Function is used in this research, with the formula provided in Equation 10.

$$L(cost) = -\frac{1}{m}\sum_{i-1}^{m} C_{yi}(yi\ \log(\bar{\Upsilon}i) + (1-yi)\log(1-\bar{\Upsilon}i)) \tag{10}$$

L_{cost} is the cost-sensitive loss function, m is the number of samples, and C_{yi}a is the cost associated with y_i, where C_{yi} is more significant for the minority class.

To enhance the performance of the MLP model, the weakness of MLP in handling class imbalance is addressed during the Loss Function Calculation stage by optimizing it with cost-sensitive learning through the implementation of a weighted loss function. The weighted loss function assigns different costs for errors in the majority and minority classes based on the consequences of each error, thereby reducing errors, and improving the model's performance. The proposed method is outlined in Algorithm 1.

Algorithm 1: Optimization of MLP Using CSL

Input:
  Training dataset: D={(xi, yi)}
  Cost for misclassification errors
  K: Number of training iterations
Output:
Final Model G(X)
Cost Structure Determination
  Set the cost C_0 for errors in the majority and C1 for errors in the minority class based on the respective consequences of the error.
Initialization and Forward Propagation
  Use Equation (2) to initialize weight W and bias b in the MLP
    For each batch of data (X,y): perform forward propagation by calculating the output $a^{(l)}$ at each layer l using the appropriate activation functions with Equations (3) and (4).
Calculate the Cost-Sensitive Loss Function: Compute the loss function using the cost-sensitive learning formula (10).
Backpropagation and Weight Update:
    Calculate the gradients of the cost-sensitive loss function using Equations (6) and (7).
    Update weight W and biases b using the gradient descent optimization method with Equations (8) and (9)
Evaluation and Fine-Tuning.
  Evaluate the model using Equations (11), (12), (13), (14), and (15).
  Perform fine-tuning on hyperparameters and class costs to improve model performance.
Final Model G(X)

The MLP model in this study was configured to balance complexity with performance. A learning rate of 0.001 was selected as it optimally balances convergence speed with model stability, avoiding premature convergence and excessively slow training[24]. The hidden layer was set to 32 neurons, based on cross-validation experiments that indicated this number effectively captures data complexity without leading to overfitting[25]. The Rectified Linear Unit (ReLU) activation function was employed for its effectiveness in addressing the vanishing gradient problem and its ability to accelerate convergence in complex neural network training tasks[24].

To further enhance the model's performance in detecting fraudulent transactions, particularly within the context of class imbalance, CSL was implemented. This approach involved adjusting the classification error costs by assigning higher weights to the minority class (fraud)[26]. This adjustment ensures that misclassification errors for fraud transactions are penalized more heavily during training, thereby improving the model's sensitivity to these critical cases. The specific weights were determined through an analysis of class distribution and the potential financial impact of misclassification, ensuring that the model remains focused on minimizing false negatives in the fraud class. Additionally, the loss function was modified to a weighted cross-entropy function, integrating class weights directly into the error calculation[27]. This modification was chosen because it effectively increases the model's focus on the minority class during training, thus enhancing the model's ability to correctly classify rare fraudulent transactions.

Performance Evaluation*:* The subsequent phase of this research involves assessing the effectiveness of the CCFD model. This performance testing aims to determine the model's suitability for practical use. Several evaluation parameters are used, including Accuracy (A_c), Recall (R_e), Precision (P_r), F1 Score (F1), and Area Under the Curve (AUC)[5]. These parameters provide a comprehensive assessment of the effectiveness and reliability of the classification model. The formulas for each parameter are provided in Equations 11 - 15[28].

$$Accuracy = \frac{TP+TN}{TP+FP+TN+PN} \tag{11}$$

$$Precision = \frac{TP}{TP+TF} \tag{12}$$

$$Recall = \frac{TP}{TP+TN} \tag{13}$$

$$F1\ Score = \frac{2\ x\ Precision\ x\ recall}{precision+recall} \tag{14}$$

$$AUC = \int_0^1 TPR(FPR)d\ (FPR) \tag{15}$$

In this research, statistical validation is performed using the Friedman Test and the Paired T-test. The Friedman non-parametric test compares the performance of several classification models on the same dataset[29]. This test examines the null hypothesis that there are no significant differences in the performance of these models. If the Friedman Test results indicate substantial differences, further analysis is conducted using the Paired T-test to identify which pairs of models have significant performance differences. The combination of these two tests provides comprehensive validation, ensuring that the developed model is statistically superior and has practical significance in its application[30].

3. Results and Discussions

The fraud detection model for credit card transactions was designed to optimize the weights of the MLP using CSL during the loss function stage. This approach modifies the standard MLP by incorporating a cost-sensitive loss function, which ensures that the model is more cautious in its predictions to minimize the consequences of misclassification, particularly for the minority class. The model is composed of multiple layers, including a Dense layer activated by ReLU, a BatchNormalization layer, and a Dropout layer to prevent overfitting. For binary classification, the output layer employs a sigmoid activation function. The model is compiled with the Adam optimizer, set to a learning rate of 0.001, and uses the binary_crossentropy loss function, with accuracy as the evaluation metric. This configuration was applied to the datasets described in Table 1.

The performance of the implemented model was evaluated and compared with several other configurations, including the baseline MLP algorithm, MLP with SMOTE, and MLP with RUS. This evaluation utilized key metrics such as Accuracy (Ac), Recall (Re), F1-Score (F1), and Area Under the Curve (AUC), as shown in Table 2.

Table 2. Result Evaluation of Model

Algorithm	A_c	R_e	F_1	AUC
MLP Baseline	0.998	0.1	0.181	0.55
MLP+Cost Sensitive	0.996	0.6	0.363	0.99
MLP+SMOTE	0.998	0.4	0.47	0.699
MLP + RUS	0.846	0.8	0.17	0.82

The results indicate that the CSL-optimized MLP significantly outperforms the baseline model in terms of recall and AUC, demonstrating its effectiveness in prioritizing the detection of fraudulent transactions. Specifically, the CSL-optimized MLP achieved a recall of 0.6 and an AUC of 0.99, which is substantially higher than the baseline MLP's recall of 0.1 and AUC of 0.55.

Subsequently, the model's performance was further analyzed using ROC curves, presented in Figure 2, providing a visual representation of the discriminatory power of each model. The ROC curve for the CSL model shows a steeper rise, indicating better classification performance for the minority class (fraudulent transactions). For a more detailed analysis, confusion matrices were constructed for each model, offering insights into their predictive capabilities, particularly in distinguishing between fraud and non-fraud cases. The confusion matrices for the Baseline, Cost-Sensitive, SMOTE, and Under Sampling models are displayed in Figures 3, 4, 5, and 6, respectively. These matrices highlight the number of true positives, true negatives, false positives, and false negatives for each model, providing a clear visual understanding of their classification performance.

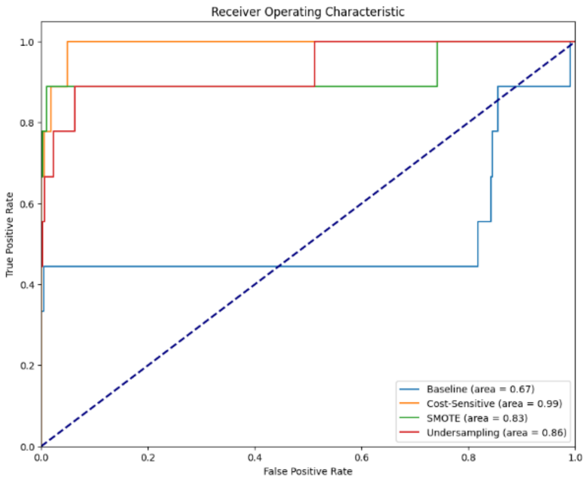

Figure 2. ROC Curve for Measuring Model Performance

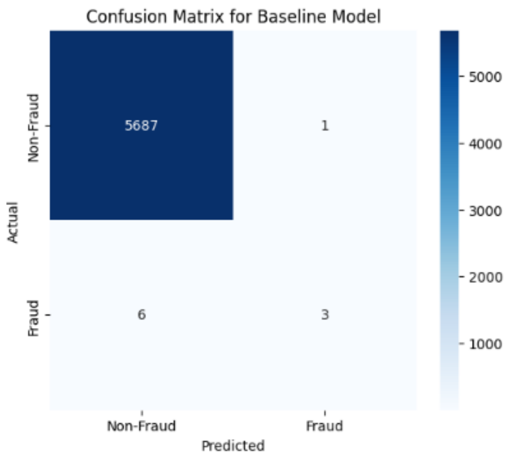

Figure 3. Confusion Matrix for Baseline Model

Among the four models, the Cost-Sensitive model shown in Figure 4 provides the best balance between detecting fraud cases (true positives) and minimizing false positives, thus being considered the most effective model for fraud detection. The higher recall of 0.6 observed in the CSL model indicates its superior ability to capture fraudulent transactions, compared to the lower recall values of the baseline and SMOTE-enhanced models. Furthermore, the lower false positive rate in the CSL model underscores its robustness in practical scenarios, where minimizing false alarms is critical to maintaining trust in the detection system. However, it is important to note that the performance gains observed in the CSL-optimized model may be influenced by the specific weighting strategy employed during training. The higher penalties assigned to misclassification errors in the minority class could lead to a trade-off between precision and recall, as evidenced by the moderate F1-Score of 0.363. This suggests that while the CSL model is highly effective in increasing sensitivity to fraud, it may also introduce a higher number of false positives, requiring further fine-tuning to achieve optimal performance.

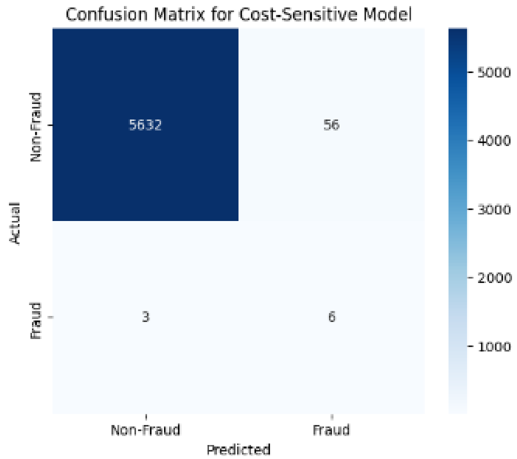

Figure 4. Confusion Matrix for Cost-Sensitive Learning

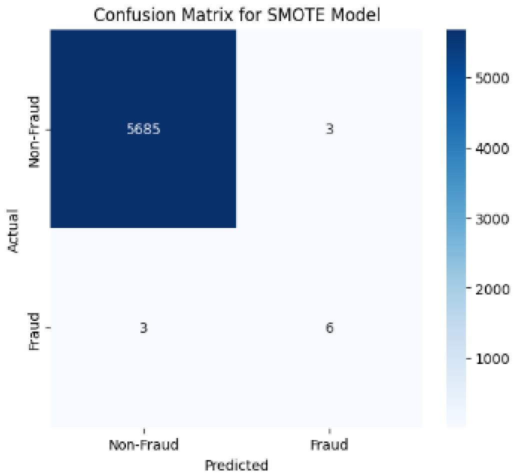

Figure 5. Confusion Matrix for SMOTE Model

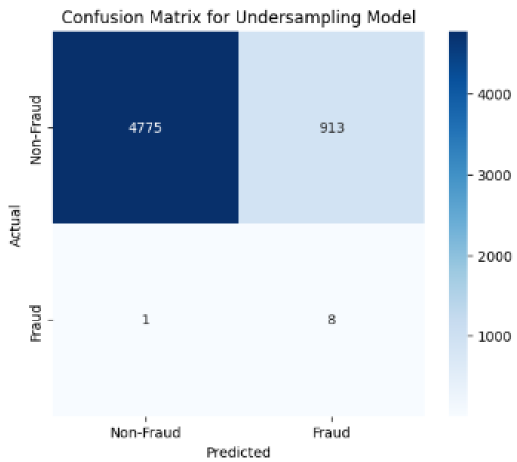

Figure 6. Confusion Matrix for Under Sampling Model

Statistical Validation: To evaluate the reliability of the developed model, we will employ statistical tests, specifically the Friedman test and the T-test, to assess its performance relative to other models. The proposed model will serve as the control method in this experiment, with the significance level (α) set at 0.0016 for the statistical tests. Typically, smaller p-values signify significant differences among the comparison methods. The results of the Friedman test and T-test for the baseline MLP model, which has not been optimized with cost-sensitive learning, are shown in Table 3.

Table 3. Statistical Test Results

	MLP Baseline	MLP+SMOTE	MLP+RUS
Friedman	6.333	6.333	6.333
T-Test	4.181	-1.775	-0.207

Analysis of Performance Differences: The differences in performance between the CSL-optimized MLP and other models, such as the baseline MLP, MLP with SMOTE, and MLP with RUS, can be attributed to several key factors. Firstly, the impact of the cost-sensitive learning approach on model performance is significant. By assigning higher penalties to misclassification errors in the minority class (fraud), the CSL model inherently prioritizes the detection of fraudulent transactions. This leads to higher recall values compared to the baseline MLP, which tends to underperform in detecting fraud due to the overwhelming dominance of non-fraudulent transactions. The weighted loss function used in CSL plays a crucial role in this improvement by forcing the model to focus more on the minority class during training, thus enhancing the model's sensitivity to fraud. Secondly, the use of SMOTE and RUS techniques for handling class imbalance introduces different effects on model performance. SMOTE, which generates synthetic samples for the minority class, typically increases recall but can sometimes introduce noise, leading to potential overfitting. Conversely, RUS reduces the size of the majority class, which can improve model performance by balancing the dataset but might also result in the loss of valuable information, thus affecting the model’s ability to generalize well. The CSL model, by directly addressing the class imbalance through weighted penalties rather than data modification, avoids these pitfalls, resulting in a more robust and reliable performance. Lastly, the complexity of the CSL model compared to the baseline MLP could also contribute to the observed performance differences. The additional computational overhead required to calculate and apply cost-sensitive weights may lead to better optimization of the model parameters, thus enhancing its ability to detect fraudulent transactions more effectively. However, this increased complexity could also mean that the model is more prone to overfitting, particularly if not carefully fine-tuned, as indicated by the moderate F1-Score observed in some instances. Overall, while the CSL-optimized MLP demonstrates superior performance in detecting fraudulent transactions, these gains necessitate careful consideration of potential trade-offs, such as the balance between recall and precision, and the risk of overfitting, which must be managed through further refinement of the model.

Potential Bias and External Factors: The CSL-optimized MLP, while effective in detecting fraudulent transactions, is susceptible to overfitting due to its focus on the minority class. The weighted penalties applied during training might cause the model to become overly specialized in the patterns found in the training data, potentially reducing its ability to generalize to new, unseen datasets. This risk is particularly significant in highly imbalanced datasets, where the model may overemphasize certain features unique to fraudulent transactions within the training set. Additionally, the use of SMOTE and RUS for data preprocessing can introduce biases—SMOTE may generate synthetic samples that do not accurately represent real-world transactions, while RUS might lead to the loss of valuable information from the majority class, affecting the model’s overall robustness. External factors, such as evolving fraud tactics and variations in transaction data across different regions or industries, further challenge the model's long-term effectiveness. A model trained on historical data might underperform as fraud patterns change over time. Therefore, continuous model updates and recalibration are essential to maintaining its relevance and accuracy. Regular validation and the incorporation of new data are critical strategies to ensure that the model remains effective in real-world applications, where the dynamics of fraudulent behavior and data characteristics can shift rapidly.

Comparison with Other Research: To validate the success of this research, we compare it with other studies that use the same credit card dataset and address class imbalance. In this study [5] tackled class imbalance by testing the credit card dataset using feature extraction and data sampling, resulting in an AUC of 0.97 and the highest accuracy of 0.97. In contrast, this study's optimization of MLP with cost-sensitive learning achieved an AUC of 0.99 and an accuracy of 0.998.

Testing on Different Datasets: Additional tests were conducted using different datasets to further evaluate the modified MLP model's performance with cost-sensitive learning. The test datasets include e-commerce and credit card transactions containing fraud, as described in Table 1. Based on Table 4, the results show that the MLP optimized using CCL outperforms the baseline MLP algorithm, MLP + SMOTE, and MLP + RUS. For Dataset 1, the optimized MLP excels in recall and AUC compared to all other algorithms. For Dataset 2, all algorithms achieved a perfect 1.0 or 100% score. The ROC performance results demonstrate that the MLP optimized with Cost-Sensitive Learning achieved the best performance among all tested algorithms, as shown in Figures 7 and 8.

Table 4. Testing on Different Datasets

Dataset		MLP Baseline	MLP+ SMOTE	MLP+ RUS	MLP+ Cos
E-Commerce	Ac	0.94	0.78	0.57	0.80
	Re	0	0.56	0.64	0.60
	F1	0	0.21	0.13	0.24
	AUC	0	0.67	0.60	0.77
	Ac	0.98	0.99	0.96	1.00
Credit Card 2	Re	0.90	0.97	1.00	1.00
	F1	0.95	1.00	0.86	1.00
	AUC	0.95	0.99	0.98	1.00

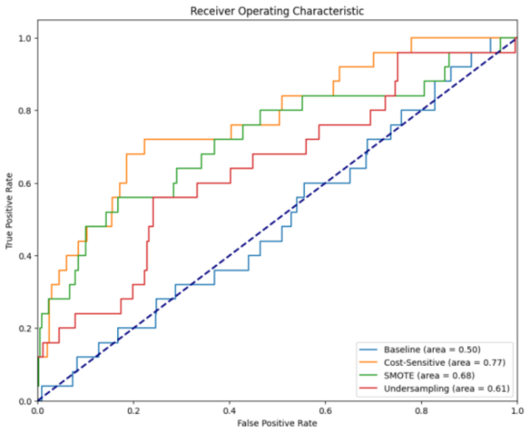

Figure 7. ROC Dataset E-Commerce

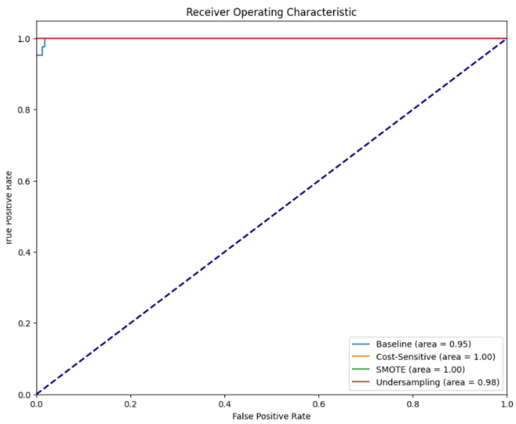

Figure 8. ROC Dataset Credit Card 2

The proposed model has been successfully implemented and tested for its performance in CCFD using a highly imbalanced dataset. The results indicate that the MLP optimized with CSL outperforms the baseline MLP and other methods, such as MLP with SMOTE and MLP with RUS, particularly in terms of AUC and recall metrics. Specifically, the CSL-optimized model achieved an AUC of 0.95 and a recall of 0.6, demonstrating its effectiveness in prioritizing the detection of fraudulent transactions. These findings were validated using statistical tests, including the Friedman test and T-test, which confirmed significant performance improvements over the baseline model.

This study utilizes a real-world credit card transaction dataset from Kaggle, with 284,807 records, of which only 492 are fraudulent, representing just 0.172% of the data. The CSL approach was implemented by modifying the loss function to apply higher penalties for misclassifications in the minority class. This ensures that the model becomes more sensitive to detecting fraud, an essential aspect when working with highly imbalanced datasets. The implementation of CSL involves several key stages, including initializing weights and biases, forward propagation, cost-based loss calculation, and backpropagation with gradient descent optimization.

When compared to other studies that have used similar datasets and approaches, this model demonstrates superior accuracy (0.996) and AUC (0.99). The results underscore the practical value of Cost-Sensitive Learning in enhancing the performance of MLPs for fraud detection in financial transactions. The implications for the financial industry are significant, as a more sensitive model can substantially reduce financial losses due to fraud and increase customer trust in payment systems. However, the approach is not without limitations. The need for careful parameter tuning to avoid overfitting, as well as the potential increase in computational resource requirements due to the complexity of the weighted loss function, are critical factors that must be managed. Future research should focus on refining these aspects to ensure that the proposed model can be effectively scaled and applied in real-world environments, where the dynamics of fraud and transaction data are continually evolving.

4. Conclusions

This research successfully implemented and tested the performance of a model for CCFD using a highly imbalanced dataset. The proposed model, which utilized CSL as the optimization approach for MLP, was compared with the baseline MLP, MLP optimized with SMOTE, and MLP optimized with RUS. The evaluation demonstrated that the MLP with CSL significantly outperformed other methods, particularly in terms of AUC and recall, achieving an AUC of 0.99 and a recall of 0.6. The Friedman and T-test further validated that the performance improvements achieved with CSL were statistically significant. Compared to other studies using similar datasets, this model showed a slight reduction in AUC but achieved higher accuracy (0.996), confirming the effectiveness of the CSL approach. This result suggests that CSL is a more reliable strategy for improving MLP performance in detecting credit card fraud within imbalanced datasets. The practical implications of this research are substantial for the financial industry, as the enhanced sensitivity of the model can lead to reduced financial losses and increased trust in payment systems. However, the research also identified several limitations, such as the use of static weights for the minority class and the limited validation across diverse datasets. Future research should focus on exploring dynamic weighting strategies, incorporating a broader range of datasets, and implementing automated hyperparameter optimization to further improve model performance. These steps will ensure that the proposed model can be effectively scaled and adapted to real-world environments where the characteristics of fraudulent behavior are continuously evolving.

References

[1] S. Bagheri, S. Taridashti, H. Farahani, P. Watson, and E. Rezvani, "Multilayer perceptron modeling for social dysfunction prediction based on general health factors in an Iranian women sample," *Front. Psychiatry*, vol. 14, no. December, pp. 1–12, 2023, doi: 10.3389/fpsyt.2023.1283095.

[2] Z. Yi *et al.*, "Fraud detection in capital markets: A novel machine learning approach," *Expert Syst. Appl.*, vol. 231, no. October 2022, p. 120760, 2023, doi: 10.1016/j.eswa.2023.120760.

[3] FBI Springfield, "Internet Crime Complaint Center Releases 2022 Statistics," *https://www.fbi.gov/*, 2023. https://www.fbi.gov/contact-us/field-offices/springfield/news/internet-crime-complaint-center-releases-2022-statistics.

[4] M. Habibpour *et al.*, "Uncertainty-aware credit card fraud detection using deep learning," *Eng. Appl. Artif. Intell.*, vol. 123, no. January, p. 106248, 2023, doi: 10.1016/j.engappai.2023.106248.

[5] Z. Salekshahrezaee, J. L. Leevy, and T. M. Khoshgoftaar, "The effect of feature extraction and data sampling on credit card fraud detection," *J. Big Data*, vol. 10, no. 1, 2023, doi: 10.1186/s40537-023-00684-w.

[6] M. Alamri and M. Ykhlef, "Survey of Credit Card Anomaly and Fraud Detection Using Sampling Techniques," *Electron.*, vol. 11, no. 23, 2022, doi: 10.3390/electronics11234003.

[7] H. Ahmad, B. Kasasbeh, B. Aldabaybah, and E. Rawashdeh, "Class balancing framework for credit card fraud detection based on clustering and similarity-based selection (SBS)," *Int. J. Inf. Technol.*, vol. 15, no. 1, pp. 325–333, 2023, doi: 10.1007/s41870-022-00987-w.

[8] L. Liu, X. Wu, S. Li, Y. Li, S. Tan, and Y. Bai, "Solving the class imbalance problem using ensemble algorithm: application of screening for aortic dissection," *BMC Med. Inform. Decis. Mak.*, vol. 22, no. 1, pp. 1–16, 2022, doi: 10.1186/s12911-022-01821-w.

[9] P. C. Y. Cheah, Y. Yang, and B. G. Lee, "Enhancing Financial Fraud Detection through Addressing Class Imbalance Using Hybrid SMOTE-GAN Techniques," *Int. J. Financ. Stud.*, vol. 11, no. 3, 2023, doi: 10.3390/ijfs11030110.

[10] J. Xian, "An Imbalanced Financial Fraud Data Model Based on Improved XGBoost and RUS Boost Fusion Algorithm with Pairwise," *BCP Bus. Manag.*, vol. 49, pp. 410–419, 2023, doi: 10.54691/bcpbm.v49i.5445.

[11] H. C. Du, L. Lv, H. Wang, and A. Guo, "A novel method for detecting credit card fraud problems," *PLoS One*, vol. 19, no. 3 March, pp. 1–26, 2024, doi: 10.1371/journal.pone.0294537.

[12] J. H. Joloudari, A. Marefat, M. A. Nematollahi, S. S. Oyelere, and S. Hussain, "Effective Class-Imbalance Learning Based on SMOTE and Convolutional Neural Networks," *Appl. Sci.*, vol. 13, no. 6, 2023, doi: 10.3390/app13064006.

[13] I. de Zarzà, J. de Curtò, and C. T. Calafate, "Optimizing Neural Networks for Imbalanced Data," *Electron.*, vol. 12, no. 12, pp. 1–26, 2023, doi: 10.3390/electronics12122674.

[14] F. Z. El Hlouli, J. Riffi, M. A. Mahraz, A. El Yahyaouy, and H. Tairi, "Credit Card Fraud Detection Based on Multilayer Perceptron and Extreme Learning Machine Architectures," *2020 Int. Conf. Intell. Syst. Comput. Vision, ISCV 2020*, pp. 3–7, 2020, doi: 10.1109/ISCV49265.2020.9204185.

[15] "Credit Card Fraud Detection," *kaggle*, 2018. https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud.

[16] N. F. S. Recruitment, "Credit Card Fraud Detection," *kaggle*, 2022. https://www.kaggle.com/competitions/nus-fintech-recruitment/data.

[17] P. Gupta, A. Varshney, M. R. Khan, R. Ahmed, M. Shuaib, and S. Alam, "Unbalanced Credit Card Fraud Detection Data: A Machine Learning-Oriented Comparative Study of Balancing Techniques," *Procedia Comput. Sci.*, vol. 218, pp. 2575–2584, 2023, doi: 10.1016/j.procs.2023.01.231.

[18] M. C. Untoro and M. A. N. M. Yusuf, "Evaluate of Random Undersampling Method and Majority Weighted Minority Oversampling Technique in Resolve Imabalanced Dataset," *IT J. Res. Dev.*, vol. 8, no. 1, pp. 1–13, 2023, doi: 10.25299/itjrd.2023.12412.

[19] F. Y. Chin, C. A. Lim, and K. H. Lem, "Handling leukaemia imbalanced data using synthetic minority oversampling technique (SMOTE)," *J. Phys. Conf. Ser.*, vol. 1988, no. 1, 2021, doi: 10.1088/1742-6596/1988/1/012042.

[20] P. Mrozek, J. Panneerselvam, and O. Bagdasar, "Efficient resampling for fraud detection during anonymised credit card transactions with unbalanced datasets," *Proc. - 2020 IEEE/ACM 13th Int. Conf. Util. Cloud Comput. UCC 2020*, pp. 426–433, 2020, doi: 10.1109/UCC48980.2020.00067.

[21] M. G. Rojas, A. C. Olivera, and P. J. Vidal, "Optimising Multilayer Perceptron weights and biases through a Cellular Genetic Algorithm for medical data classification," *Array*, vol. 14, no. April, p. 100173, 2022, doi: 10.1016/j.array.2022.100173.

[22] E. Safari and M. Peykari, "Improving the multilayer Perceptron neural network using teaching-learning optimization algorithm in detecting credit card fraud," *J. Ind. Syst. Eng.*, vol. 14, no. 2, pp. 159–171, 2022.

[23] R. Saputra, S. Sunardiyo, A. Nugroho, and S. Subiyanto, "Implementasi Multilayer Perceptron Artificial Neural Network untuk Prediksi Konsumsi Energi Listrik PT PLN (Persero) UP3 Salatiga," *Elektrika*, vol. 15, no. 2, p. 60, 2023, doi: 10.26623/elektrika.v15i2.6411.

[24] K. A. Rashedi, M. T. Ismail, S. Al Wadi, A. Serroukh, T. S. Alshammari, and J. J. Jaber, "Multi-Layer Perceptron-Based Classification with Application to Outlier Detection in Saudi Arabia Stock Returns," *J. Risk Financ. Manag.*, vol. 17, no. 2, 2024, doi: 10.3390/jrfm17020069.

[25] T. Deng, "Effect of the Number of Hidden Layer Neurons on the Accuracy of the Back Propagation Neural Network," *Highlights Sci. Eng. Technol.*, vol. 74, pp. 462–468, 2023, doi: 10.54097/nbra6h45.

[26] I. Araf, A. Idri, and I. Chairi, *Cost-sensitive learning for imbalanced medical data: a review*, vol. 57, no. 4. Springer Netherlands, 2024.

[27] O. Volk and G. Singer, "An adaptive cost-sensitive learning approach in neural networks to minimize local training–test class distributions mismatch," *Intell. Syst. with Appl.*, vol. 21, no. December 2023, p. 200316, 2024, doi: 10.1016/j.iswa.2023.200316.

[28] Y. Liu and L. Wu, "Intrusion Detection Model Based on Improved Transformer," *Appl. Sci.*, vol. 13, no. 10, 2023, doi: 10.3390/app13106251.

[29] J. Liu and Y. Xu, "T-Friedman Test: A New Statistical Test for Multiple Comparison with an Adjustable Conservativeness Measure," *Int. J. Comput. Intell. Syst.*, vol. 15, no. 1, pp. 1–19, 2022, doi: 10.1007/s44196-022-00083-8.

[30] R. Cao, J. Wang, M. Mao, G. Liu, and C. Jiang, "Feature-wise attention based boosting ensemble method for fraud detection," *Eng. Appl. Artif. Intell.*, vol. 126, no. PC, p. 106975, 2023, doi: 10.1016/j.engappai.2023.106975.