

An Optimization-Driven Feature Selection and Deep Learning Framework for Accurate Credit Card Fraud Detection

Chisom Chukwuemeka Wisdom

atlascreativehub9@gmail.com

Obafemi Awolowo University

Method Article

Keywords: Credit card fraud detection, feature selection, deep learning, optimization algorithms, class imbalance, neural networks, financial cybersecurity, anomaly detection

Posted Date: October 20th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11399301/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: The authors declare no competing interests.

An Optimization-Driven Feature Selection and Deep Learning Framework for Accurate Credit Card Fraud Detection

Author : Chisom Chukwuemeka Wisdom

Publication Date : July 30, 2026

Abstract

The rapid growth of digital financial services has increased the volume and complexity of credit card transactions, creating significant challenges for financial institutions seeking to identify fraudulent activities accurately and efficiently. Traditional fraud detection approaches often struggle with extreme class imbalance, high-dimensional transaction data, evolving fraudulent behaviors, and the need for real-time decision-making. This paper proposes an optimization-driven feature selection and deep learning framework for accurate credit card fraud detection. The framework integrates feature selection, hyperparameter optimization, class-imbalance management, and deep neural network classification into a unified detection pipeline. An optimization mechanism identifies informative transaction features while reducing redundant variables and computational overhead. The selected features are subsequently processed by a deep learning classifier trained to distinguish legitimate transactions from fraudulent activities. To improve practical reliability, the framework incorporates cost-sensitive learning, precision-recall-based evaluation, probability calibration, and threshold optimization. The widely used European credit card transaction dataset is discussed as a potential benchmark for empirical evaluation. The proposed framework is expected to provide a systematic approach to improving fraud detection performance, although its actual effectiveness must be established through controlled experiments. The paper contributes a reproducible

methodological foundation for developing efficient, adaptive, and scalable fraud detection systems in modern financial environments.

Keywords: Credit card fraud detection, feature selection, deep learning, optimization algorithms, class imbalance, neural networks, financial cybersecurity, anomaly detection.

1. Introduction

Credit card fraud represents a persistent cybersecurity and financial risk for consumers, payment processors, merchants, and financial institutions. The expansion of electronic commerce, mobile banking, contactless payments, and online financial services has created opportunities for legitimate digital transactions while also increasing the number of channels that criminals can exploit. Fraudulent activities may involve stolen card details, unauthorized transactions, account compromise, card-not-present fraud, and coordinated transaction patterns intended to avoid conventional security controls.

The European Central Bank and European Banking Authority have documented the continuing significance of payment fraud. Their 2025 report states that the total value of payment fraud in the European Economic Area increased from €3.5 billion in 2023 to €4.2 billion in 2024. The report also identifies changing fraud strategies as an ongoing challenge for payment security (EBA & ECB, 2025). These findings emphasize the need for detection mechanisms that can adapt to evolving transaction patterns.

Machine learning has become an important component of modern fraud detection because it can identify statistical relationships and behavioral patterns that are difficult to express through manually defined rules. Deep learning extends these capabilities by learning complex nonlinear representations from transaction data. Nevertheless, deep learning performance depends heavily on data quality, feature representation, model architecture, and training procedures.

One major challenge is that fraudulent transactions constitute only a small proportion of most transaction datasets. A classifier may therefore achieve high overall accuracy simply by predicting that nearly every transaction is legitimate. Another challenge is the presence of redundant or weakly informative features, which can increase training costs and introduce noise.

This paper proposes an optimization-driven feature selection and deep learning framework designed to address these challenges systematically. Its primary objective is to improve the reliability of fraud classification while balancing detection performance, computational efficiency, and operational cost.

The main contributions are:

1. A unified framework combining optimized feature selection with deep neural network classification.
2. An imbalance-aware training strategy that accounts for the rarity and financial importance of fraud.
3. A rigorous evaluation methodology emphasizing precision, recall, precision-recall area under the curve, and financial cost.
4. A practical deployment strategy incorporating decision thresholds, monitoring, and model adaptation.

2. Literature Review

2.1 Machine Learning for Credit Card Fraud Detection

Machine learning-based fraud detection systems typically use historical transaction records to learn distinctions between legitimate and fraudulent activities. Common approaches include logistic regression, decision trees, Random Forest, gradient boosting, support vector machines, and neural networks.

Traditional classifiers can provide effective baselines and may perform particularly well on structured transaction data. Ensemble methods such as

Random Forest and gradient boosting can capture nonlinear relationships and interactions between variables. Deep learning models, meanwhile, can learn layered representations that may be useful when transaction data contains complex behavioral relationships.

Dal Pozzolo et al. (2018) highlighted several practical issues in credit card fraud detection, including class imbalance, changing customer behavior, and delays in obtaining verified fraud labels. Their work demonstrates why realistic evaluation protocols are essential when developing operational fraud detection systems.

2.2 Feature Selection and Optimization

Feature selection identifies a subset of input variables that preserves useful predictive information while reducing dimensionality. It differs from feature extraction, which transforms existing variables into a new representation.

Feature selection methods can be divided into three broad categories.

Filter methods evaluate features using statistical measures, such as mutual information, correlation, or statistical significance. They are computationally efficient but may overlook interactions between variables.

Wrapper methods evaluate feature subsets according to a model's predictive performance. Search strategies such as genetic algorithms, particle swarm optimization, and sequential selection can be used to explore candidate subsets. However, wrapper methods can be computationally expensive.

Embedded methods select or suppress features during model training. Examples include L1 regularization and tree-based feature importance.

An optimization-driven approach can combine these methods by using inexpensive statistical screening to eliminate uninformative variables and an optimization algorithm to search for a smaller, high-performing feature subset.

2.3 Deep Learning and Class Imbalance

Deep neural networks can learn complex relationships through successive nonlinear transformations. Multilayer perceptrons are suitable for structured transaction features, while recurrent networks may be useful when transaction sequences and temporal dependencies are represented explicitly.

However, class imbalance can cause a model to favor legitimate transactions because the majority class dominates the training objective. Researchers commonly address this issue through class-weighted losses, minority-class oversampling, majority-class undersampling, and carefully designed sampling strategies.

Chawla et al. (2002) introduced Synthetic Minority Over-sampling Technique (SMOTE), which generates synthetic minority-class examples by interpolating between neighboring minority instances. Although useful in some settings, synthetic samples can distort feature relationships if they are generated without sufficient attention to data structure.

Consequently, feature selection and deep learning should be evaluated together with imbalance-management techniques rather than treated as independent solutions.

3. Problem Formulation

Credit card fraud detection can be formulated as a binary classification problem. Let the transaction dataset be represented as:

$$D = \{(x_i, y_i)\}_{i=1}^N$$

where x_i denotes the feature vector for transaction i , $y_i \in \{0, 1\}$ represents its class, and N is the number of transactions. A value of zero represents a legitimate transaction, while one represents a fraudulent transaction.

The objective is to learn a model that estimates:

$$P(y_i=1|x_i)P(y_i=1|W \text{ mid } x_i)$$

This probability represents the model's estimated likelihood that a transaction is fraudulent.

The challenge is not merely to maximize classification accuracy. The system must detect as many fraudulent transactions as possible while controlling false alarms, computational latency, and unnecessary transaction declines.

The framework therefore seeks to optimize a multi-objective function:

$$J(S, \theta, \tau) = \lambda_1 L_{\text{cost}} + \lambda_2 (1 - \text{PR} - \text{AUC}) + \lambda_3 |S|d + \lambda_4 T$$

$$\lambda_1 L_{\text{cost}} + \lambda_2 (1 - \text{PR} - \text{AUC}) + \lambda_3 \frac{|S|}{d} + \lambda_4 T$$

where:

- S is the selected feature subset.
- θ represents model hyperparameters.
- τ is the classification threshold.
- L_{cost} is the expected financial loss associated with classification errors.
- PR-AUC is the area under the precision-recall curve.
- d is the original number of features.
- T represents an appropriately normalized computational cost.
- $\lambda_1, \lambda_2, \lambda_3, \lambda_4$ are nonnegative weights expressing the importance of each objective.

The weights must be selected according to the institution's operational priorities and validated using held-out data. The formulation is a proposed design, not a claim of experimentally established optimality.

4. Proposed Optimization-Driven Deep Learning Framework

4.1 Overall Architecture

The proposed framework consists of seven interconnected stages:

1. Transaction data collection.
2. Data preprocessing and temporal organization.
3. Feature evaluation and selection.
4. Class-imbalance management.
5. Deep learning model training.
6. Hyperparameter and threshold optimization.
7. Performance evaluation and continuous monitoring.

The stages are integrated into a single pipeline to ensure that feature selection, model training, and decision-making are evaluated consistently.

4.2 Data Collection and Preprocessing

The framework receives transaction information from payment processing systems, including transaction amount, timestamp, merchant category, device information, transaction location, authentication results, and recent transaction history, where such information is legally available.

Data preprocessing includes missing-value treatment, duplicate detection, data-type validation, and normalization of appropriate numerical variables. Categorical features may be represented using suitable encodings, while temporal features can describe transaction frequency, unusual spending intervals, and changes in customer behavior.

Sensitive information should be minimized or tokenized, and access to transaction records should be restricted according to established security and privacy requirements.

Preprocessing operations must be fitted using training data only. Applying transformations based on the complete dataset can introduce information leakage and produce misleading performance estimates.

4.3 Optimization-Based Feature Selection

The feature selection stage identifies a subset of variables that preserves fraud-related information while reducing unnecessary model complexity.

Initially, filter methods can remove constant features, highly redundant variables, and features with limited predictive relevance. Mutual information or other appropriate measures can be used to estimate the relationship between candidate features and the fraud label.

A genetic algorithm is proposed as one possible optimization method. Each candidate solution is represented as a binary vector:

$$\mathbf{z} = (z_1, z_2, \dots, z_d), z_j \in \{0, 1\} \quad \mathbf{z} = (z_1, z_2, \dots, z_d), \quad z_j \in \{0, 1\}$$

Here, $z_j = 1$ indicates that feature j is selected, whereas $z_j = 0$ indicates that it is excluded.

The optimization process follows four operations:

Initialization: Generate a population of candidate feature subsets.

Fitness evaluation: Train a computationally controlled classifier or evaluate a consistent validation procedure for each subset.

Selection and variation: Apply selection, crossover, and mutation to create new candidate solutions.

Termination: Stop when the computational budget is reached or improvements become insufficient.

The fitness function can penalize large subsets while rewarding predictive performance. The best subset is then used to train the final deep learning classifier.

This approach can improve efficiency when redundant features are present. Nevertheless, optimization does not guarantee that the selected subset will generalize to unseen transactions. Feature selection must therefore occur inside the training and validation procedure, rather than before the data is divided.

4.4 Deep Learning Classification

A multilayer perceptron (MLP) is proposed as the primary classifier because it can model nonlinear relationships among structured transaction features.

For an input representation $h(0)=x$, each hidden layer can be expressed as:

$$h(l)=\sigma(W(l)h(l-1)+b(l))$$

where $W(l)$ and $b(l)$ are trainable parameters and σ is a nonlinear activation function.

The final layer applies a sigmoid function to estimate fraud probability:

$$p_i = \frac{1}{1 + e^{-s_i}}$$

where s_i is the output logit for transaction i .

Dropout, weight regularization, and early stopping can reduce overfitting. Model depth, hidden-layer width, learning rate, dropout rate, and batch size should be selected through a controlled hyperparameter search.

For transaction streams where temporal behavior is important, a sequence-aware model can also be evaluated. However, sequential architectures should be justified by the availability of meaningful transaction histories and evidence that temporal modeling improves performance over simpler alternatives.

4.5 Imbalance-Aware Training

The training objective should account for the disproportionate cost and rarity of fraudulent transactions.

One option is weighted binary cross-entropy:

$$L = -\frac{1}{N} \sum_{i=1}^N [w_1 y_i \log(p_i) + w_0 (1 - y_i) \log(1 - p_i)]$$

where w_1 and w_0 are weights assigned to fraudulent and legitimate transactions, respectively.

A higher fraud-class weight can increase the penalty for missed fraud cases. However, excessive weighting may increase false positives and distort probability calibration.

Alternative strategies include carefully controlled undersampling, minority-class oversampling, and focal loss. These approaches should be compared empirically rather than assumed to be universally superior.

Resampling must be applied only within training partitions. Validation and test datasets should retain their natural class distributions so that performance reflects the actual evaluation environment.

4.6 Joint Hyperparameter and Threshold Optimization

After selecting features, the framework searches for suitable model hyperparameters and a decision threshold.

For a transaction with estimated fraud probability $\hat{p}_i$, the classification rule is:

$$y_i = \begin{cases} 1, & \hat{p}_i \geq \tau \\ 0, & \hat{p}_i < \tau \end{cases}$$

A low threshold generally increases fraud detection sensitivity but may also increase false alarms. A high threshold can reduce unnecessary declines while allowing more fraudulent transactions to pass undetected.

The threshold should therefore be chosen using validation data and an explicit cost model incorporating potential fraud losses, investigation expenses, customer inconvenience, and the costs of incorrectly blocking legitimate transactions.

The final threshold and model configuration must be frozen before the test set is evaluated.

5. Experimental Design and Evaluation Metrics

5.1 Dataset Selection

The European credit card fraud detection dataset published through the Machine Learning Group at Université Libre de Bruxelles and distributed on Kaggle is a useful public benchmark.

It contains 284,807 transactions recorded over two days in September 2013, including 492 fraudulent transactions. Fraud therefore accounts for approximately 0.173% of the dataset. Twenty-eight features are anonymized principal components, while Time and Amount are provided separately.

[Dataset source](#).

This dataset is useful for evaluating class-imbalance methods, but its limitations are substantial. It covers a short historical period, and its transformed features provide limited insight into the original transaction characteristics. It cannot, by itself, establish that a model will generalize to contemporary banking systems.

Where possible, evaluation should be supplemented with more recent and diverse datasets, including datasets containing merchant, device, authentication, and temporal behavioral variables.

5.2 Experimental Protocol

A rigorous evaluation should use chronological training, validation, and testing partitions when transaction timestamps permit. This approach better approximates deployment, where a model learns from past transactions and predicts future events.

The experimental comparison should include:

- Logistic regression as a simple baseline.
- Random Forest or gradient boosting as a strong structured-data baseline.
- A deep learning model using all eligible features.
- The proposed feature-selected deep learning model.
- The proposed model with alternative imbalance-management strategies.

All models should use consistent data partitions and comparable tuning budgets. Repeated runs with different random seeds can help assess stability.

Feature selection, resampling, hyperparameter tuning, and calibration must not use test-set labels. Where labels become available only after an investigation delay, the evaluation should also account for realistic label availability.

5.3 Performance Metrics

Overall accuracy is insufficient for highly imbalanced fraud datasets. The following metrics provide a more informative evaluation.

Precision measures the proportion of predicted fraud cases that are genuinely fraudulent:

$$\text{Precision} = \frac{TP}{TP + FP}$$

Recall measures the proportion of actual fraud cases correctly identified:

$$\text{Recall} = \frac{TP}{TP + FN}$$

F1-score represents the harmonic mean of precision and recall:

$$F1 = 2 \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

PR-AUC summarizes the precision-recall trade-off across classification thresholds and is particularly useful when fraud cases are rare.

ROC-AUC measures the model's ability to rank positive cases above negative cases across thresholds, but should not be used alone because extreme class imbalance can make its practical interpretation less direct.

Additional operational metrics should include false positives per thousand transactions, monetary fraud loss, fraud value detected, inference latency, and processing throughput.

6. Results Interpretation and Discussion

Because this paper proposes a framework rather than reporting a completed experiment, no numerical performance improvement is claimed. The expected benefits must be verified through the experimental protocol described above.

The first research question is whether optimization-driven feature selection improves the trade-off between detection performance and computational cost. A smaller feature set may reduce inference latency and model complexity, but removing informative variables could also reduce recall.

The second question is whether deep learning offers measurable benefits over established structured-data classifiers. Neural networks are capable of learning complex nonlinear relationships, but this capacity does not guarantee superior performance on every tabular dataset. Tree-based models should therefore remain part of the comparison.

The third question concerns class imbalance. Weighted loss functions may improve sensitivity to fraudulent transactions, but their usefulness depends on

the relationship between false negatives and false positives. A model that identifies nearly all fraud while incorrectly flagging too many legitimate transactions may be unsuitable for real-world payment processing.

Finally, evaluation must consider generalization. Fraud patterns evolve as criminals modify their behavior, customers change spending habits, and institutions introduce new authentication procedures. A model that performs well on a historical benchmark may deteriorate when deployed in a different period or customer population.

These considerations support a central principle: reliable fraud detection depends on the complete modeling and evaluation pipeline, not solely on the complexity of the neural network.

7. Practical Implications and Deployment Considerations

The proposed framework can support payment processors, banks, and digital financial service providers by prioritizing suspicious transactions for additional verification or investigation.

In a production environment, the system could assign each transaction a risk score and route it to one of three decision categories: approve, request additional authentication, or escalate for investigation. High-risk decisions should be governed by explicit policies, with human review available for ambiguous or consequential cases.

The deployment architecture should include secure event ingestion, low-latency inference, model version control, decision logging, and monitoring dashboards. Transaction-level explanations can help analysts understand whether a decision was influenced by unusual amounts, transaction frequency, device changes, or other relevant features.

Operational monitoring should track changes in fraud prevalence, feature distributions, precision, recall, calibration, and latency. Retraining should

occur under a controlled process that validates new models before deployment and preserves rollback capability.

Financial institutions should also ensure compliance with applicable data protection, payment security, consumer protection, and model governance requirements. Automated decisions should be proportionate to the evidence available and should not rely on inappropriate or discriminatory proxy variables.

8. Limitations and Future Research

The proposed framework has several limitations. First, optimization-based feature selection can be computationally expensive when the number of candidate features is large. Search efficiency may be improved through staged selection, parallel evaluation, or more efficient optimization algorithms.

Second, deep learning models can be difficult to interpret, particularly when complex feature interactions contribute to a risk score. Explainability methods should be evaluated for stability and usefulness to investigators rather than treated as definitive explanations of causation.

Third, historical datasets may not adequately represent emerging fraud techniques. Future studies should evaluate temporal drift, delayed labels, adversarial adaptation, and cross-institution generalization.

Fourth, the framework's effectiveness depends on the quality and availability of transaction information. Highly anonymized datasets may not support all proposed feature engineering methods.

Future research should compare genetic algorithms with particle swarm optimization, mutual-information selection, and embedded regularization. Researchers should also investigate calibrated ensemble systems, graph-based transaction analysis, privacy-preserving collaborative learning, and adaptive threshold policies.

Most importantly, future studies should publish reproducible code, dataset descriptions, parameter settings, confidence intervals, and complete baseline comparisons. Such practices will help establish whether the proposed framework offers consistent improvements under realistic financial conditions.

9. Conclusion

This paper has proposed an optimization-driven feature selection and deep learning framework for accurate credit card fraud detection. The framework integrates data preprocessing, feature subset optimization, imbalance-aware learning, deep neural network classification, hyperparameter tuning, threshold selection, and operational monitoring.

Its principal advantage is methodological integration: feature selection and classification are treated as connected components of a fraud detection system rather than isolated tasks. The approach also emphasizes realistic evaluation through precision, recall, PR-AUC, financial cost, and inference latency.

Nevertheless, the proposed framework should not be considered experimentally validated until it has been compared with established baselines using leakage-free and temporally appropriate evaluation. Feature selection may reduce complexity, and deep learning may capture nonlinear transaction patterns, but their actual benefits depend on the data, fraud distribution, model configuration, and operating environment.

A reliable credit card fraud detection system must balance detection sensitivity, false alarms, computational efficiency, explainability, and adaptation to changing threats. By providing a structured methodology for evaluating these trade-offs, the proposed framework offers a foundation for future empirical research and the development of more accurate, efficient, and responsible financial fraud detection systems.

References:

1. Kravchik, M., Biggio, B., & Shabtai, A. (2021, March). Poisoning attacks on cyber attack detectors for industrial control systems. In *Proceedings of the 36th annual ACM symposium on applied computing* (pp. 116-125).
2. Anthi, E., Williams, L., Rhode, M., Burnap, P., & Wedgbury, A. (2021). Adversarial attacks on machine learning cybersecurity defences in industrial control systems. *Journal of Information Security and Applications*, 58, 102717.
3. Bej, S. R., Mann, K., Pakalapati, A., & Pillai, B. (2026, May). Investigating the Robustness of Machine Learning Models Against Data Poisoning in Industrial Control Systems. In *2026 7th International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)* (pp. 291-297). IEEE.
4. Wang, Z., Ma, J., Wang, X., Hu, J., Qin, Z., & Ren, K. (2022). Threats to training: A survey of poisoning attacks and defenses on machine learning systems. *ACM Computing Surveys*, 55(7), 1-36.
5. Bhattacharjee, S., Islam, M. J., & Abedzadeh, S. (2022, May). Robust anomaly based attack detection in smart grids under data poisoning attacks. In *Proceedings of the 8th ACM on cyber-physical system security workshop* (pp. 3-14).
6. Clement, T., Gbaja, C., & Onayemi, H. (2025). Adversarial machine learning: Defense mechanisms against poisoning attacks in cybersecurity models. *International Journal of Engineering and Computer Science*, 14(06), 27286-27308.
7. Kalejaiye, A. N. (2024). Adversarial machine learning for robust cybersecurity: strengthening deep neural architectures against evasion, poisoning, and model-inference attacks. *International Journal of Computer Applications Technology and Research*, 13(12), 72-95.
8. Bej, S. R., Mann, K., Pakalapati, A., & Pillai, B. (2026, May). Investigating the Robustness of Machine Learning Models Against

Data Poisoning in Industrial Control Systems. In *2026 7th International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)* (pp. 291-297). IEEE.

9. Laghubhai Mangukiya, M., & Miyani, H. (2026). Smart electronics production: AI, machine vision, and industry 4.0 in PCB, harness, and system integration. Brilliant Publications House.
10. Takiddin, A., Ismail, M., Zafar, U., & Serpedin, E. (2020). Robust electricity theft detection against data poisoning attacks in smart grids. *IEEE Transactions on Smart Grid*, 12(3), 2675-2684.
11. Chen, J., Gao, X., Deng, R., He, Y., Fang, C., & Cheng, P. (2020). Generating adversarial examples against machine learning-based intrusion detector in industrial control systems. *IEEE Transactions on Dependable and Secure Computing*, 19(3), 1810-1825.
12. Elnour, M., Noorizadeh, M., Shakerpour, M., Meskin, N., Khan, K., & Jain, R. (2023). A machine learning based framework for real-time detection and mitigation of sensor false data injection cyber-physical attacks in industrial control systems. *IEEE Access*, 11, 86977-86998.
13. Dunn, C., Moustafa, N., & Turnbull, B. (2020). Robustness evaluations of sustainable machine learning models against data poisoning attacks in the internet of things. *Sustainability*, 12(16), 6434.
14. Nankya, M., Chataut, R., & Akl, R. (2023). Securing industrial control systems: Components, cyber threats, and machine learning-driven defense strategies. *Sensors*, 23(21), 8840.
15. Rahul, S., Awati, Y., Pakalapati, A., Goyal, H., Tatiya, S., & Baskaran, E. (2026, April). Deep Triple Spatial Diffractive Neural Network with Dollmaker Optimization Algorithm for Efficient Credit Card Fraud Detection. In *2026 12th International Conference on Communication and Signal Processing (ICCSP)* (pp. 316-321). IEEE.
16. Aslam, M. M., Tufail, A., & Irshad, M. N. (2025). Survey of deep learning approaches for securing industrial control systems: A comparative analysis. *Cyber Security and Applications*, 3, 100096.

17. Maabreh, M., Maabreh, A., Qolomany, B., & Al-Fuqaha, A. (2022). The robustness of popular multiclass machine learning models against poisoning attacks: Lessons and insights. *International Journal of Distributed Sensor Networks*, 18(7), 15501329221105159.
18. Alahmed, S., Alasad, Q., Yuan, J. S., & Alawad, M. (2024). Impacting robustness in deep learning-based NIDS through poisoning attacks. *Algorithms*, 17(4), 155.
19. Imran, M., Siddiqui, H. U. R., Raza, A., Raza, M. A., Rustam, F., & Ashraf, I. (2023). A performance overview of machine learning-based defense strategies for advanced persistent threats in industrial control systems. *Computers & Security*, 134, 103445.
20. Mangukiya, M., Miyani, H., & Yadav, V. (2026). Comment on “Case report: Electrocardiographic (ECG) recording during the hanging process”. *Forensic Science, Medicine and Pathology*, 1-2.
21. S. R. Bej, K. Mann, A. Pakalapati and B. Pillai, "Investigating the Robustness of Machine Learning Models Against Data Poisoning in Industrial Control Systems," 2026 7th International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), Tirunelveli, India, 2026, pp. 291-297, doi: 10.1109/ICICV68925.2026.11554655.