Efficient Privacy-Aware Financial Classification using Extra Trees Ensemble: A Comparative Evaluation with Deep Neural Networks

[1]Aakash Dongre, [2]Shaheen Ayyub, [3]Dr. Divyank Mishra
[1]M. Tech Scholar, Department of Computer Science & Engineering, Technocrats Institute of Technology
[2]Associate Professor, Department of Computer Science & Engineering, Technocrats Institute of Technology
[3]Assistant Professor, Department of Computer Science & Engineering, Technocrats Institute of Technology
[1]aakashdongre2@gmail.com , [2]shaheenayyub@gmail.com, [3]divyankmishra71@gmail.com

Abstract

This study proposes a unified privacy-preserving machine learning framework for financial anomaly detection using a Synthetic Financial Dataset. Thes framework integrates Differential Privacy, Homomorphic Encryption (overhead modeling), Federated Learning, and data anonymization to evaluate their impact on predictive performance and computational efficiency. Two models—a Deep Neural Network (DNN) and an Extra Trees ensemble—are comparatively analysed under identical preprocessing and privacy conditions. Experimental results show that the DNN achieves 94% accuracy under baseline conditions but degrades to 86% under combined privacy constraints, with increased training time. In contrast, the Extra Trees model consistently outperforms the DNN, achieving up to 99.99% accuracy with significantly lower training time (as low as 37.94 seconds) and minimal performance degradation across all privacy configurations. Federated Learning provides the highest efficiency, while anonymization preserves performance with negligible overhead. The findings highlight clear privacy–utility trade-offs and demonstrate that ensemble-based models offer superior robustness and efficiency for privacy-aware financial classification.

Keywords: Privacy-Preserving Machine Learning, Financial Anomaly Detection, Extra Trees Ensemble, Deep Neural Network.

1. Introduction

Big Data in mobile cloud computing (MCC) environments introduces significant challenges in ensuring data security and privacy due to the massive volume of sensitive information being generated, transmitted, and processed. Traditional encryption techniques often struggle to balance security with computational efficiency, especially in resource-constrained mobile devices. To address this, advanced privacy-preserving encryption methods such as homomorphic encryption, attribute-based encryption, and lightweight cryptographic algorithms have been developed to enable secure data storage, sharing, and processing without compromising performance [1], [2]. These approaches support scalable and efficient operations, making them suitable for Big Data applications while maintaining user trust and compliance with data protection regulations. MCC integrates the computational power of cloud computing with the mobility of handheld devices, evolving through advancements in wireless technologies like 4G, 5G, and edge computing to support real-time analytics, IoT, and AI-driven services [3]. The convergence of Big Data and MCC enables real-time data analysis and informed decision-making but also introduces challenges related to scalability, dynamic network conditions, and security vulnerabilities such as data breaches and unauthorized access [4], [5]. Given the distributed and dynamic nature of MCC, robust security mechanisms including encryption, authentication, and intrusion detection—are essential to safeguard sensitive data. Consequently, advanced encryption and secure data-sharing frameworks play a vital role in mitigating risks and ensuring the reliable adoption of MCC systems.

A. Mobile Cloud Computing (MCC)

Mobile Cloud Computing (MCC) represents a powerful integration of cloud computing and mobile technology, enabling seamless access to computational resources and storage through mobile devices while overcoming their inherent limitations such as restricted processing power, battery life, and storage capacity by offloading tasks to cloud servers [6]. Supported by advancements in wireless technologies like 4G, 5G, and Wi-Fi, MCC facilitates low-latency, high-bandwidth communication, allowing users to perform intensive tasks such as real-time analytics, machine learning, and multimedia processing across diverse domains including healthcare, finance, and entertainment. It also enhances business scalability and flexibility by centralizing resources and enabling efficient Big Data analytics from mobile and IoT-generated data [7]. However, the distributed and mobile nature of MCC introduces challenges related to security, privacy, and reliability, necessitating advanced protection mechanisms such as end-to-end encryption, multi-factor authentication, and secure cloud architectures [8]. The overall MCC structure, as illustrated in Figure 1, demonstrates how mobile devices connect through network providers and the internet to access cloud infrastructure, enabling scalable, secure, and efficient service delivery [9].

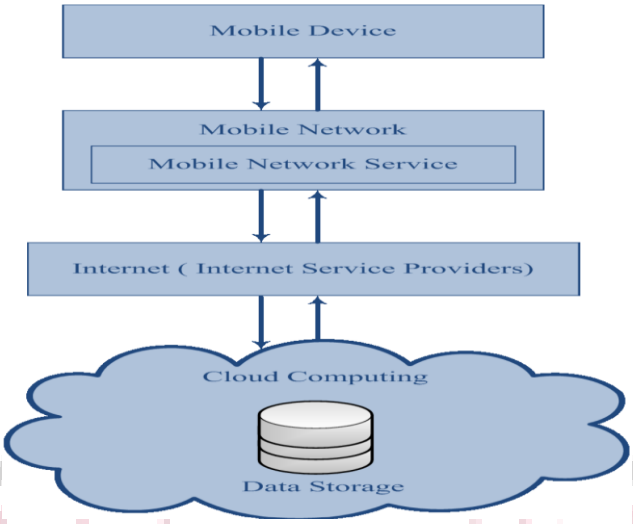

Figure 1. Architecture of Mobile Cloud Computing (MCC) [9]

B. Data Security for MCC

Data security is a major concern in Mobile Cloud Computing (MCC) due to its distributed architecture and the transmission of sensitive data over public networks [10]. The reliance on wireless communication and third-party cloud providers exposes MCC systems to risks such as data breaches, unauthorized access, and man-in-the-middle attacks, potentially leading to trust loss and legal consequences. Additionally, mobile devices are vulnerable to physical and cyber threats, while cloud infrastructures face sophisticated attacks, and data transmission further increases risks to confidentiality and integrity [11]. To address these challenges, a multi-layered security framework is required, incorporating encryption, access control, and real-time monitoring [12]. Techniques such as end-to-end and homomorphic encryption ensure data confidentiality, while multi-factor authentication (MFA) and secure protocols like HTTPS and VPNs strengthen access security [13]. Moreover, intrusion detection systems, blockchain integration, and zero-trust architectures enhance threat detection, data integrity, and secure access, collectively ensuring a reliable and secure MCC environment.

2. Literature Review

Recent studies highlight significant advancements and challenges in Mobile Cloud Computing (MCC) driven by emerging technologies such as 5G, edge computing, and intelligent optimization. **Asghari and Sohrabi (2024) [14]** emphasize the importance of efficient server placement in MCC, where relocating cloud resources to the network edge improves Quality of Service (QoS), reduces latency, and enhances energy efficiency through approaches including machine learning, optimization, and meta-heuristics. Similarly, **Pramanik et al. (2024) [15]** explore mobile crowd computing, leveraging smart mobile devices for cost-effective high-performance computing, while outlining its architecture, sustainability benefits, and research challenges. **Sen et al. (2024) [16]** focus on computation offloading as a key technique to enhance performance, reduce latency, and optimize resource utilization by transferring tasks from mobile devices to cloud servers. Addressing security concerns, **Alabdeli et al. (2024) [17]** propose a BOA-SVM-based intrusion detection system achieving 99.78% accuracy, demonstrating the effectiveness of hybrid machine learning models in MCC security. Furthermore, **Devi et al. (2024) [18]** highlight the role of cloud computing in managing IoE-generated big data, emphasizing load balancing and task scheduling using machine learning and optimization techniques. **Ajaz et al. (2024) [19]** extend MCC applications to vehicular networks, proposing cloud-integrated VANET architectures to improve scalability, security, and connectivity. **Y. A. N. G. Shouyi et al. (2024) [20]** discuss Mobile Edge Computing (MEC) as a transformative paradigm enabling low-latency, secure, and scalable communication systems, particularly in 6G environments, while addressing challenges such as interoperability and mobility management. Additionally, **Malgwi et al. (2024) [21]** investigate security routing protocols in cloud-based networks, identifying key threats such as data breaches and DoS attacks and validating enhanced security mechanisms through simulation. Collectively, these studies demonstrate that MCC is rapidly evolving through integration with edge computing, AI-driven optimization, and advanced security frameworks, while still facing challenges related to scalability, security, and efficient resource management.

Table 1 Comparative Summary of Recent MCC Research Studies and Identified Gaps

Study	Core Focus	Models Used	Application Area	Key Contribution	Research Gap
Asghari & Sohrabi (2024) [14]	Server placement in MCC (Edge + Cloud)	ML-based, optimization, heuristics, meta-heuristics	Cloudlet, Fog, Edge Computing	Improved QoS, reduced latency & energy via optimal server placement	Lack of unified scalable framework for dynamic environments
Pramanik et al. (2024) [15]	Mobile Crowd Computing (MCC) architecture	Conceptual frameworks, classification models	HPC using smart mobile devices	Comprehensive taxonomy, architecture & sustainability analysis	Limited real-world deployment and performance evaluation
Sen et al. (2024) [16]	Computation offloading	Decision engines, profiling-based strategies	Mobile apps (AI, gaming, recognition)	Enhances battery life, reduces latency & improves performance	Complex decision-making under dynamic network conditions
Alabdeli et al. (2024) [17]	MCC Security (IDS)	BOA-SVM, GWO feature selection	Intrusion Detection (UNSW-NB15 dataset)	Achieved 99.78% accuracy outperforming existing models	High computational cost, limited real-time deployment
Devi et al. (2024) [18]	Load balancing & task scheduling in cloud	ML, optimization algorithms (SLR of 63 papers)	IoE & cloud environments	Comprehensive review of scheduling & balancing techniques	Need for real-time adaptive and scalable solutions
Ajaz et al. (2024) [19]	VANET + Cloud integration	Cloud architectures (HVC, VC, VuC)	Vehicular networks	Improved scalability, connectivity & traffic efficiency	Security and privacy issues still unresolved
Y. A. N. G. Shouyi et al. (2024) [20]	Mobile Edge Computing (MEC)	AI-assisted MEC, blockchain, IoT integration	6G communication systems	Enables low-latency, intelligent & secure communication	Challenges in interoperability, mobility, scalability
Malgwi et al. (2024) [21]	Cloud network security routing	Encryption, SSH, routing protocols (EIGRP, OSPF, BGP)	Cloud-based networks	Validated secure routing via simulation (Cisco Packet Tracer)	Limited real-world validation and integration with AI systems

3. Research Objectives

- To analyse the limitations of conventional machine learning models in handling sensitive financial data under privacy constraints.
- To design a unified privacy-preserving learning framework integrating Differential Privacy, Homomorphic Encryption modeling, Federated Learning, and data anonymization techniques.
- To develop and evaluate an Extra Trees–based ensemble model for financial anomaly detection and compare its performance with a Deep Neural Network baseline.
- To investigate the impact of different privacy mechanisms on classification accuracy and computational efficiency.
- To assess the privacy–utility trade-offs and identify an efficient model–privacy combination that balances data protection and predictive performance.

4. Research Methodology

This section presents the research methodology for designing, implementing, and evaluating privacy-preserving machine learning models using a structured synthetic financial transaction dataset. With the growing reliance on data-driven techniques for sensitive information, conventional machine learning methods that require unrestricted

access to raw data pose risks of privacy leakage. To address this, the study integrates multiple privacy-preserving mechanisms with supervised learning models while maintaining predictive performance and computational efficiency. A Deep Neural Network (DNN) is utilized to capture non-linear relationships, while an Extra Trees ensemble model is employed for its computational efficiency and robustness to noise, enabling comparative analysis across different model complexities. The privacy framework incorporates Differential Privacy through controlled noise injection, Homomorphic Encryption via modeling Paillier-based computational overhead, and data anonymization through feature generalization. The methodology follows a structured pipeline including data preprocessing, feature encoding, privacy integration, model training, and evaluation, with performance assessed using metrics such as classification accuracy, training time, and throughput. This systematic approach establishes a foundation for analysing privacy–utility trade-offs and supports the implementation results presented in the subsequent section.

Figure 2 illustrates a structured workflow for developing a privacy-preserving machine learning framework using a synthetic financial dataset from Kaggle. The process begins with preprocessing steps, including target–feature separation, removal of identifier columns, categorical encoding, train–test splitting, and class imbalance handling to prepare clean and balanced data. This processed data is then fed into the model architecture stage, where both a Deep Neural Network (DNN) and an Extra Trees algorithm are implemented to capture different learning capabilities. Subsequently, appropriate privacy techniques such as differential privacy, homomorphic encryption, and anonymization are integrated to ensure data security. Finally, the models are evaluated based on performance metrics, completing the end-to-end pipeline.

Figure 2 Workflow of the Proposed Privacy-Preserving Machine Learning Framework

A. Dataset Description

A single benchmark dataset was utilized to evaluate the proposed privacy-preserving learning framework, enabling a controlled analysis of classification performance, computational overhead, and the impact of various privacy protection mechanisms.

1. Synthetic Financial Dataset

The study utilizes a Synthetic Financial Transaction Dataset obtained from Kaggle, designed to simulate real-world financial behavior with labeled records for supervised learning. The dataset includes both numerical and categorical features describing transaction-level information, with a binary target variable indicating normal and anomalous transactions. During preprocessing, identifier attributes were removed and categorical features were encoded to ensure model compatibility. Additionally, the dataset exhibits class imbalance, making it suitable for evaluating model robustness and performance under privacy-preserving conditions. The preprocessing starts by loading the Kaggle-based synthetic financial dataset in CSV format, where each record contains numerical, categorical features, and a binary label, preserved in raw form for further processing.

Table 2 Feature Categories and Description of the Synthetic Financial Dataset

Feature Category	Description
Numerical Features	Transaction amount, origin account balance before transaction, origin account balance after transaction, destination account balance before transaction, destination account balance after transaction
Categorical Features	Transaction type indicating the nature of the financial operation
Identifier Features (Removed)	Sender account ID, receiver account ID
Target Variable	Binary class label (Normal / Anomalous transaction)

B. Data Preprocessing

Data preprocessing is a critical component of the proposed privacy-preserving machine learning framework, as it enhances data consistency, removes redundant attributes, and prepares the dataset for reliable learning under privacy constraints. The process begins with target–feature separation to avoid data leakage, followed by the removal of identifier-based attributes to reduce bias and improve generalization. Categorical features are transformed using one-hot encoding, and missing or invalid values are handled to maintain numerical stability. Additionally, controlled Gaussian noise is injected into the feature space to simulate real-world uncertainty and introduce a lightweight privacy-enhancing effect, thereby preventing overly optimistic model behavior.

Subsequently, the dataset is split into training and testing subsets using a stratified 80:20 ratio to preserve class distribution and ensure unbiased evaluation. Class imbalance, a key characteristic of the dataset, is addressed through a weak oversampling strategy applied only to the training data, improving the model's ability to detect anomalous transactions without causing overfitting. Overall, this structured preprocessing pipeline establishes a robust foundation for model training, fair comparison, and effective integration of privacy-preserving mechanisms.

C. Model Architecture

This section outlines the learning models used in the privacy-preserving framework to evaluate performance, computational efficiency, and robustness under privacy constraints. Two models are considered: a Deep Neural Network (DNN) as a parametric baseline capable of capturing complex non-linear relationships, and an Extra Trees ensemble as a non-parametric model known for its robustness to noise and feature variability. By analysing both models under identical preprocessing and privacy settings, the study enables a systematic comparison of privacy–utility trade-offs, with the DNN specifically designed to process pre-processed features while maintaining controlled complexity to prevent overfitting.

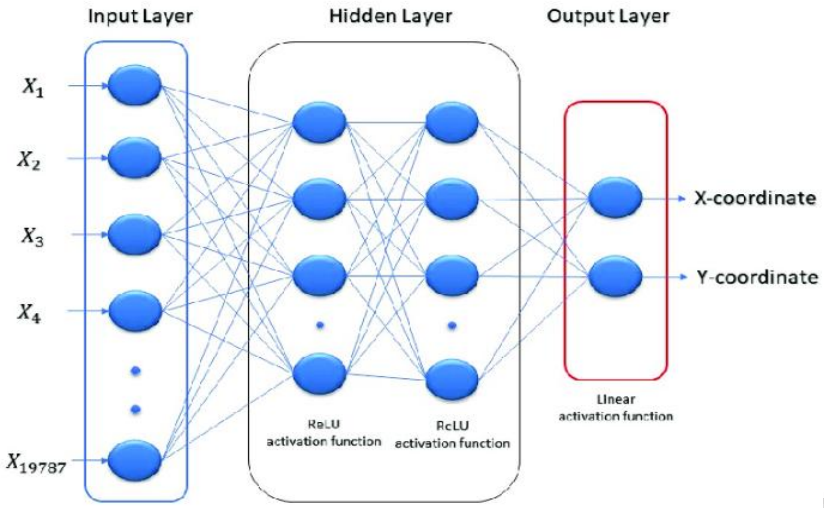

Figure 3 DNN Architecture Used for Privacy-Preserving Anomaly Detection

The Deep Neural Network (DNN) architecture consists of an input layer that receives the standardized feature vector $X_1, X_2, \ldots, X_n$ without transformation, followed by two fully connected hidden layers that apply weighted summation and ReLU activation to capture complex non-linear relationships in transaction data. A dropout mechanism is incorporated between hidden layers to enhance generalization and robustness, particularly under privacy-induced perturbations. The output layer contains a single neuron with a sigmoid activation function to produce a probability score for binary classification (normal vs anomalous), which is converted into class labels using a fixed threshold. The model is trained using the Adam optimizer with binary cross-entropy loss for efficient convergence. In contrast, the Extra Trees ensemble model is adopted as a non-parametric alternative that constructs decision boundaries through highly randomized trees, offering improved robustness to noise, feature variability, and privacy-preserving transformations while maintaining consistency with the same preprocessed feature space.

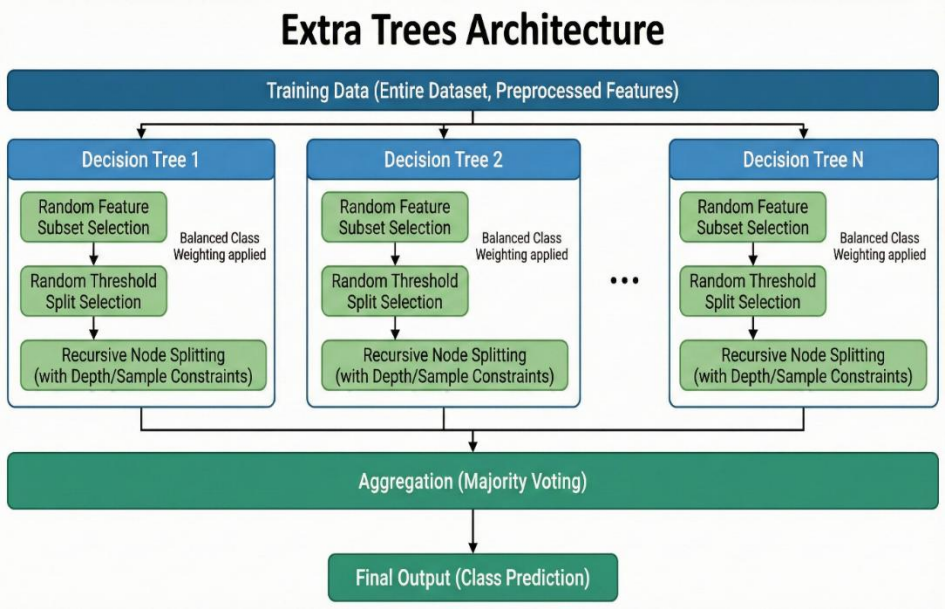

Figure 4 Extra Trees–Based Architecture for Privacy-Preserving Anomaly Detection

The Extra Trees ensemble classifier is composed of multiple decision trees trained in parallel on the same pre-processed dataset, where each tree is built independently using strong randomization to ensure diversity and improve generalization. Randomness is introduced by selecting a subset of features at each split and choosing split thresholds randomly, enabling faster training and enhanced robustness to noise and privacy-induced perturbations. Tree growth is controlled through constraints such as maximum depth and minimum samples to

prevent overfitting and ensure stable performance, while class imbalance is addressed through balanced weighting. During prediction, each tree outputs a class label, and the final result is obtained באמצעות majority voting, which stabilizes predictions under privacy transformations. Additionally, the model is computationally efficient, as it avoids iterative optimization and scales linearly with data size, making it suitable for repeated evaluations in privacy-preserving settings.

The integration of privacy-preserving techniques in the proposed framework is systematically implemented after preprocessing to ensure data confidentiality while maintaining analytical utility. Four key mechanisms are considered: Differential Privacy, Homomorphic Encryption overhead modeling, Federated Learning, and data anonymization, each applied independently and in combination for comprehensive evaluation. Differential Privacy is achieved by injecting Gaussian noise into training features to limit information leakage, while Homomorphic Encryption is simulated through computational overhead modeling to reflect increased training cost and reduced precision. A combined configuration of both techniques is also analyzed to assess cumulative privacy impact. Federated Learning is incorporated by partitioning data into multiple subsets for decentralized model training and aggregation without sharing raw data, whereas anonymization is applied through feature generalization to reduce identifiability. All techniques are evaluated using consistent metrics such as accuracy, training time, and throughput, enabling a structured analysis of privacy–utility trade-offs across different models and configurations.

D. Evaluation Metrics

The proposed privacy-preserving models are evaluated using standard classification metrics to assess predictive performance, class-wise discrimination, and robustness under class imbalance, providing quantitative insight into privacy–utility trade-offs.

Accuracy represents the proportion of correctly classified samples relative to the total number of samples. While accuracy provides an overall performance measure, it may not fully reflect model effectiveness when class distributions are imbalanced.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

Where:

- TP = True Positives (correctly identified anomalous transactions)
- TN = True Negatives (correctly identified normal transactions)
- FP = False Positives (normal transactions misclassified as anomalous)
- FN = False Negatives (anomalous transactions misclassified as normal)

Precision measures the proportion of correctly predicted anomalous transactions among all samples predicted as anomalous. It reflects the reliability of positive predictions and is particularly important when false alarms are undesirable.

$$\text{Precision} = \frac{TP}{TP+FP} \quad (2)$$

Recall also known as the true positive rate, quantifies the proportion of actual anomalous transactions that are correctly identified by the model. High recall indicates effective detection of minority-class samples.

$$\text{Recall} = \frac{TP}{TP+FN} \quad (3)$$

F1-Score is the harmonic mean of precision and recall, providing a balanced evaluation metric that considers both false positives and false negatives. It is especially useful for imbalanced datasets.

$$\text{F1-Score} = 2 \times \frac{Precision \times Recall}{Precision+Recall} \quad (4)$$

Confusion Matrix provides a tabular summary of prediction outcomes, showing the distribution of correct and incorrect classifications for each class. It enables visual inspection of classification strengths and weaknesses.

Table 3 Predicted vs. Actual Outcomes in a Confusion Matrix

Predicted \ Actual	**Positive**	**Negative**
Positive	True Positives (TP)	False Positives (FP)
Negative	False Negatives (FN)	True Negatives (TN)

The confusion matrix is further used to compute recall and false positive rates for each class.

Macro Average computes evaluation metrics independently for each class and then calculates their unweighted mean. This approach treats all classes equally, regardless of their sample size.

$$\text{Macro Average} = \frac{1}{N}\sum_{i=1}^{N} Metrics_i \quad (5)$$

Where N is the number of classes, and $Metrics_i$ is the evaluation metric (precision, recall, F1) for each class.

Weighted Average accounts for class imbalance by weighting each class-specific metric by its support (number of samples). This metric provides a distic measure of overall performance when class distributions are uneven.

$$\text{Weighted Average} = \frac{\sum_{i=1}^{N}(Metrics_i \times Support_i)}{\sum_{i=1}^{N} Support_i} \quad (6)$$

ROC Curve and AUC curve illustrates the trade-off between the true positive rate and false positive rate across varying decision thresholds. The Area Under the Curve (AUC) summarizes the model's discriminative capability. An AUC value closer to 1 indicates strong class separability, while a value near 0.5 suggests random performance.

E. Model Evaluation Process

The evaluation of the proposed privacy-preserving learning models follows a structured process:

- **Training and Testing**: Models are trained on the training dataset and evaluated on an independent test dataset.
- **Confusion Matrix Generation**: Confusion matrices are computed to analyse prediction outcomes and misclassification patterns.
- **Metric Computation**: Precision, recall, F1-score, and accuracy are calculated for each class.
- **Macro and Weighted Averages**: These averages are computed to assess overall model behaviour under class imbalance.
- **ROC Curve and AUC Analysis**: ROC curves and AUC values are generated to evaluate threshold-independent discriminative performance.

5. Results

This section analyses the performance of the proposed privacy-preserving framework using DNN and Extra Trees under various privacy settings. The DNN achieves 94% accuracy but degrades under stronger privacy constraints, while Extra Trees consistently outperforms it with up to 99.94% accuracy and lower training time. Overall, Extra Trees demonstrates superior robustness, efficiency, and stability across all privacy configurations.

A. Software Tools and Frameworks

The proposed privacy-preserving machine learning framework was implemented using a set of standard software tools that support efficient data processing, model development, and evaluation. Python served as the primary programming language due to its simplicity and extensive ML ecosystem. Pandas and NumPy were used for data handling, preprocessing, and numerical operations, including feature transformation and noise injection. Scikit-learn facilitated train–test splitting, feature scaling, model implementation (Extra Trees), and performance evaluation, while Imbalanced-learn was used to handle class imbalance through controlled SMOTE. TensorFlow and Keras were employed to design and train the Deep Neural Network (DNN), and Google Colab provided a flexible computational environment for execution, experimentation, and result visualization.

B. Results for DNN Model

This section evaluates the Deep Neural Network (DNN) under various privacy-preserving configurations using the Synthetic Financial Dataset for binary classification of normal and anomalous transactions. A consistent preprocessing pipeline, including feature encoding, noise injection, masking, and scaling, is applied to ensure fair comparison. Privacy techniques such as Differential Privacy, Homomorphic Encryption (overhead modeling), their combination, Federated Learning, and data anonymization are integrated into the DNN framework. The model is trained with identical settings across all configurations, and performance is assessed using classification accuracy and training time to analyses privacy–utility trade-offs.

Table 4 Performance of DNN Model under Different Privacy-Preserving Techniques

S. No.	Technique	Training Time (sec)	Accuracy
1	DNN	167.35	0.94
2	DNN + Differential Privacy	164.78	0.94
3	DNN + Paillier Homomorphic Encryption	263.66	0.90
4	DNN + Differential Privacy + HE	263.66	0.86
5	DNN + Federated Learning	43.35	0.94
6	DNN + Anonymization	169.41	0.94

Table 4 presents the performance of the Deep Neural Network (DNN) under different privacy-preserving configurations using training time and classification accuracy. The baseline DNN achieves 94% accuracy with 167.35 seconds training time, serving as a reference. Under Differential Privacy, performance remains unchanged (94%) with similar training time, indicating robustness to noise. However, Homomorphic Encryption increases training time significantly (263.66 seconds) and reduces accuracy to 90%, while the combined privacy setting further lowers accuracy to 86%, highlighting the trade-off between privacy and utility. In contrast, Federated Learning maintains 94% accuracy with a significantly reduced training time (43.35 seconds), demonstrating computational efficiency. Data anonymization also preserves accuracy (94%) with minimal impact on training time, indicating that it effectively balances privacy and performance.

C. Performance Impact of Privacy Mechanisms on the DNN Model

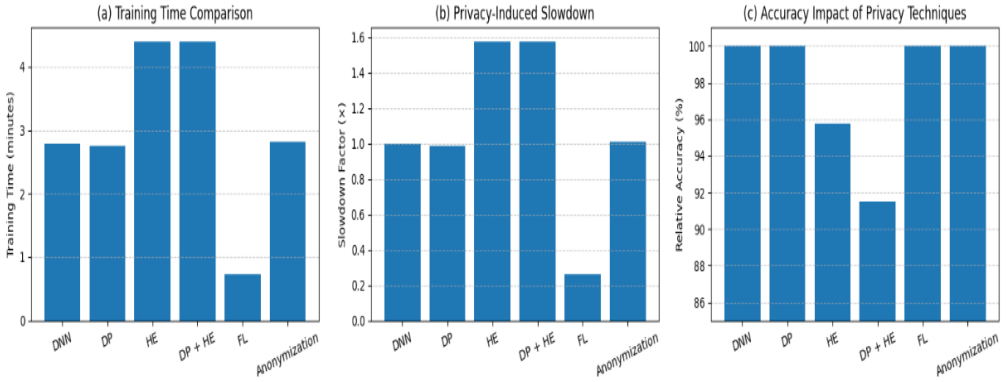

Figure 5 Comparative Analysis of Training Time and Accuracy for the DNN Model under Privacy-Preserving Techniques

The performance graph illustrates the impact of different privacy techniques on DNN efficiency and accuracy. Differential Privacy shows minimal effect on training time and accuracy, remaining close to the baseline. In contrast, Homomorphic Encryption and its combination with Differential Privacy significantly increase training time and reduce accuracy due to higher computational overhead and information loss. Federated Learning achieves the lowest training time while maintaining accuracy, and anonymization also preserves performance with minimal overhead. Overall, the results highlight a clear trade-off, where stronger privacy mechanisms increase computational cost and degrade accuracy, while lighter techniques maintain a better balance between privacy and performance.

D. Results for Extra Trees Model

This section summarizes the performance of the Extra Trees ensemble model, treated as the proposed approach, evaluated under the same privacy configurations as the DNN for fair comparison. Using identical preprocessing, Extra Trees achieves very high accuracy with low training time and demonstrates strong robustness to noise, data

transformations, and privacy constraints. Even under stricter privacy settings, performance degradation remains minimal, indicating that Extra Trees effectively balances predictive performance and computational efficiency, making it a reliable alternative to neural network-based models.

Table 5. Performance of Extra Trees Model under Privacy-Preserving Techniques

Technique	Training Time (sec)	Accuracy
Extra Trees classifier	46.904	0.9994
Extra Trees + Differential Privacy	77.073	0.9958
Extra Trees + Paillier HE	70.356	0.9758
Extra Trees + DP + HE	115.609	0.9658
Extra Trees + Federated Learning	37.943	0.9999
Extra Trees + Anonymization	50.302	0.9994

Table 5. summarizes the performance of the Extra Trees model under various privacy configurations using training time and classification accuracy. The baseline achieves 99.94% accuracy with 46.90 seconds, demonstrating high efficiency. Under Differential Privacy, accuracy remains high (99.58%) with moderate increase in training time, indicating robustness to noise. Homomorphic Encryption increases training time and reduces accuracy to 97.58%, while the combined setting further increases cost and lowers accuracy to 96.58%. Federated Learning delivers the best performance with 99.99% accuracy and the lowest training time, highlighting computational efficiency. Anonymization maintains baseline accuracy with minimal overhead, showing that Extra Trees effectively balances performance and privacy across all configurations.

E. Performance Impact of Privacy Mechanisms on the DNN Model

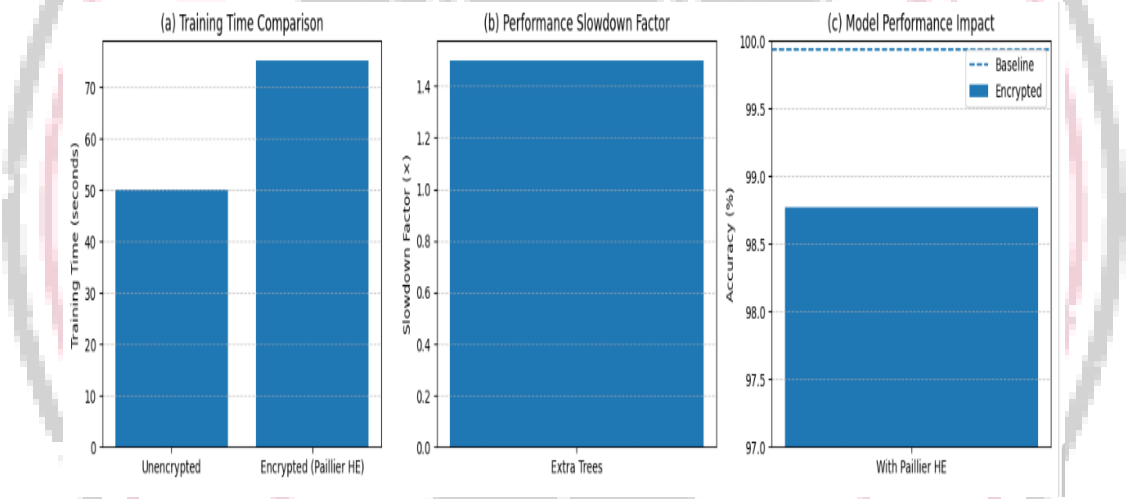

Figure 6 Comparative Analysis of Training Time and Accuracy for the Extra Tree Model under Privacy-Preserving Techniques

Figure 6 illustrates the effect of Paillier Homomorphic Encryption on the Extra Trees model in terms of training time, computational slowdown, and accuracy. Encryption increases training time and introduces a moderate slowdown due to additional operations on encrypted data, while causing only a slight reduction in accuracy. Despite this overhead, the model maintains high predictive performance, highlighting a trade-off where enhanced privacy is achieved with increased computational cost and minimal impact on accuracy.

F. Overall Observations for Extra Trees Model Performance

The Extra Trees model achieves very high accuracy with low training time, demonstrating strong suitability for structured financial data and high robustness to privacy-induced perturbations. Under Differential Privacy, performance degradation is minimal, while Homomorphic Encryption increases computational cost and slightly reduces accuracy. The combined privacy setting results in the highest overhead and largest accuracy drop, reflecting cumulative effects of stronger constraints. In contrast, Federated Learning provides the best efficiency with the highest accuracy and lowest training time, and anonymization preserves baseline performance with minimal overhead, indicating an effective balance between privacy and utility.

Table 8 Comparative Performance Analysis of DNN and Extra Trees Models under Privacy-Preserving Techniques

Privacy Technique	DNN – Training Time (sec)	DNN – Accuracy	Extra Trees – Training Time (sec)	Extra Trees – Accuracy
DNN and Extra Trees	167.35	0.94	46.90	0.9994
Differential Privacy (DP)	164.78	0.94	77.07	0.9958
Paillier Homomorphic Encryption (HE)	263.66	0.90	70.36	0.9758
DP + HE	263.66	0.86	115.61	0.9658
Federated Learning (FL)	43.35	0.94	37.94	0.9999
Anonymization	169.41	0.94	50.30	0.9994

The table 8 provides a comparative analysis of DNN and Extra Trees under various privacy-preserving techniques based on training time and classification accuracy. The Extra Trees model consistently outperforms the DNN, achieving higher accuracy (up to 99.94%) with significantly lower training time. While Differential Privacy has minimal impact on both models, Homomorphic Encryption and its combination with DP increase computational cost and reduce accuracy, with the DNN showing greater degradation. Federated Learning offers the best efficiency for both models, especially Extra Trees, which achieves the highest accuracy and lowest training time. Anonymization maintains performance with minimal overhead. Overall, Extra Trees demonstrates superior robustness, efficiency, and resilience to privacy constraints compared to DNN.

G. Training Time Comparison

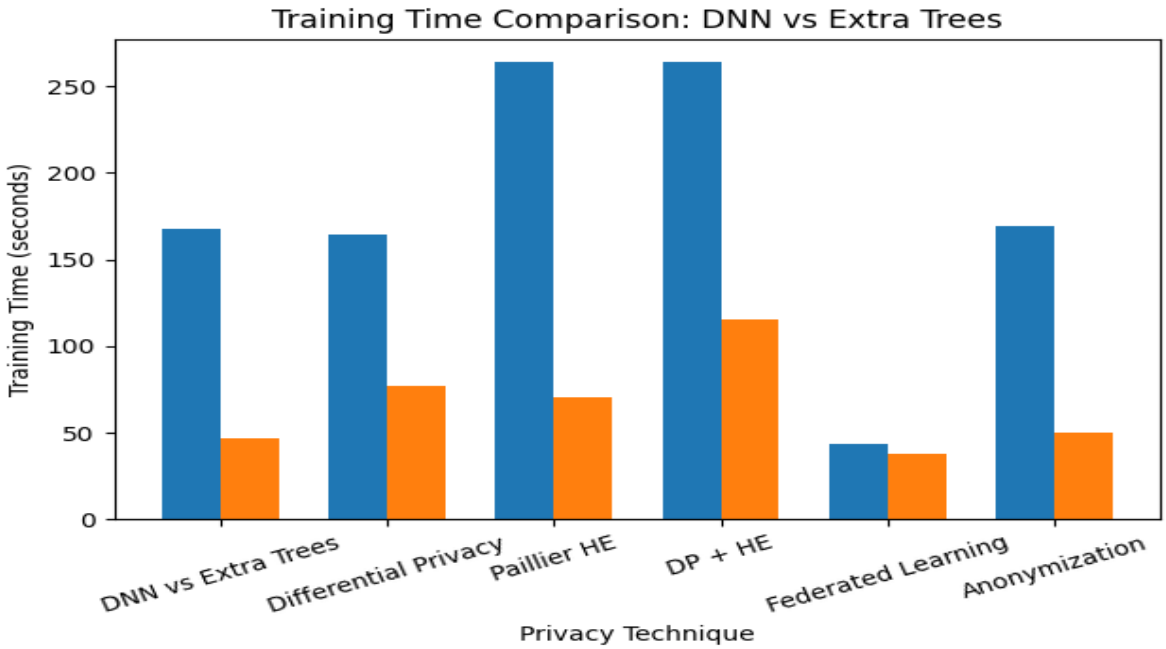

Figure 7 Training Time Comparison of DNN and Extra Trees Models under Privacy-Preserving Techniques

The training time comparison shows that the Extra Trees model consistently requires significantly less time than the DNN across all privacy configurations, highlighting its computational efficiency. While Differential Privacy slightly increases training time for both models, the impact is minimal for Extra Trees, indicating better tolerance to noise. In contrast, Homomorphic Encryption and the combined DP + HE setting cause a sharp increase in DNN training time, whereas Extra Trees experiences only a moderate rise. Federated Learning achieves the lowest training time for both models, with Extra Trees remaining the fastest, and anonymization introduces only minor overhead, again with Extra Trees maintaining superior efficiency.

H. Accuracy Comparison

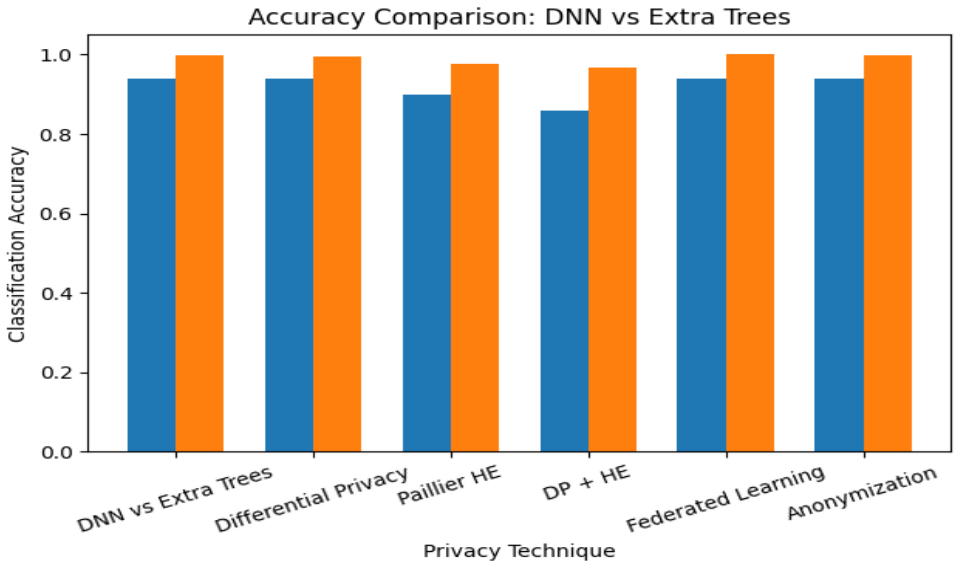

Figure 8 Accuracy Impact Comparison of DNN and Extra Trees Models under Privacy-Preserving Techniques

The accuracy comparison shows that Extra Trees consistently outperforms the DNN across all privacy techniques. While the DNN maintains stable accuracy under Differential Privacy and anonymization, it experiences significant degradation under Homomorphic Encryption and DP + HE due to sensitivity to information loss. In contrast, Extra Trees remains highly robust, with only minor accuracy reduction even under stronger privacy constraints. Federated Learning delivers the highest accuracy for both models, with Extra Trees achieving near-perfect performance.

6. Conclusion

This study presented a unified privacy-preserving machine learning framework for financial anomaly detection, integrating Differential Privacy, Homomorphic Encryption modeling, Federated Learning, and data anonymization. A comparative evaluation between Deep Neural Networks (DNN) and Extra Trees revealed clear differences in performance under privacy constraints. The DNN achieved a baseline accuracy of 94% but showed significant degradation to 86% under combined privacy mechanisms, along with a substantial increase in training time (up to 263.66 seconds), indicating sensitivity to information loss and computational overhead. In contrast, the Extra Trees model consistently demonstrated superior performance, achieving up to 99.99% accuracy with significantly lower training time (as low as 37.94 seconds), even under strict privacy configurations. Among the evaluated techniques, Federated Learning provided the best balance between efficiency and accuracy for both models, while Differential Privacy and anonymization preserved performance with minimal overhead. Homomorphic Encryption introduced the highest computational cost and moderate accuracy reduction, highlighting the trade-off between strong privacy guarantees and model utility. Overall, the results confirm that ensemble-based models, particularly Extra Trees, offer a more robust, efficient, and scalable solution for privacy-aware financial data analysis compared to neural network-based approaches.

References

[1] Bashir, S. R., Raza, S., & Misic, V. (2024). Progress in Privacy Protection: A Review of Privacy Preserving Techniques in Recommender Systems, Edge Computing, and Cloud Computing. *arXiv preprint arXiv:2401.11305*. https://doi.org/10.48550/arXiv.2401.11305

[2] Amaithi Rajan, A., & V, V. (2024). Systematic survey: secure and privacy-preserving big data analytics in cloud. *Journal of Computer Information Systems*, *64*(1), 136-156. https://doi.org/10.1080/08874417.2023.2176946

[3] Roy, S., & Nag, S. (2024). Chapter-15 Privacy-Preserving Surveillance Systems in Smart Homes: Strategies and Implementations. *Synergy in Science and Engineering: An Integrative Approach*, 123.

[4] Dhinakaran, D., Sankar, S. M., Selvaraj, D., & Raja, S. E. (2024). Privacy-Preserving Data in IoT-based Cloud Systems: A Comprehensive Survey with AI Integration. *arXiv preprint arXiv:2401.00794*. https://doi.org/10.48550/arXiv.2401.00794

[5] Alabdulatif, A., Thilakarathne, N. N., & Kalinaki, K. (2023). A novel cloud enabled access control model for preserving the security and privacy of medical big data. *Electronics*, *12*(12), 2646. https://doi.org/10.3390/electronics12122646

[6] Nam, D. H. (2023, July). A Comparative Study of Mobile Cloud Computing, Mobile Edge Computing, and Mobile Edge Cloud Computing. In 2023 Congress in Computer Science, Computer Engineering, & Applied Computing (CSCE) (pp. 1219-1224). IEEE. https://doi.org/10.1109/CSCE60160.2023.00204

[7] Sheikh, B., Butt, A., & Hanif, J. (2023). Mobile cloud computing: A survey on current security trends and future directions. Engineering Proceedings, 32(1), 22. https://doi.org/10.3390/engproc2023032022

[8] Dhawan, S. (2024, October). Navigating the convergence: A comprehensive review of synergies and interplay between Mobile Cloud Computing, Edge Computing, and Fog Computing. In AIP Conference Proceedings (Vol. 3209, No. 1). AIP Publishing. https://doi.org/10.1063/5.0229035

[9] Modares, Hero & Lloret, Jaime & Moravejosharieh, Amir & Salleh, Rosli. (2014). Security in Mobile Cloud Computing. 3. 1548-1560. 10.4018/978-1-4666-6539-2.ch072.

[10] Mora, H., Pujol, F. A., Ramirez-Gordillo, T., & Jimeno, A. (2024). Mobile Cloud Computing Paradigm: A Survey of Operational Concerns, Challenges and Open Issues. Transactions on Emerging Telecommunications Technologies, 35(12), e70020. https://doi.org/10.1186/s13677-024-00697-7

[11] Aga, D. T., Chintanippu, R., Mowri, R. A., & Siddula, M. (2024). Exploring secure and private data aggregation techniques for the internet of things: a comprehensive review. Discover Internet of Things, 4(1), 28. https://doi.org/10.1007/s43926-024-00064-7

[12] Kaur, R., & Kaur, G. (2023). Security Challenges and Solutions in Mobile Cloud Computing Environments: A Review. Journal of Advanced Research in Embedded System, 10(1), 17-20. http://www.ijrar.org/

[13] Alakbarov, R. G. (2023). Security issues and solution mechanisms in cloud computing systems: a review. Problems of Information Technology, 12-22. http://doi.org/10.25045/jpit.v14.i2.02

[14] Asghari, A., & Sohrabi, M. K. (2024). Server placement in mobile cloud computing: A comprehensive survey for edge computing, fog computing and cloudlet. *Computer Science Review*, *51*, 100616. https://doi.org/10.1016/j.cosrev.2023.100616

[15] Pramanik, P. K. D., Pal, S., & Choudhury, P. (2024). Mobile crowd computing: potential, architecture, requirements, challenges, and applications. *The Journal of Supercomputing*, *80*(2), 2223-2318. https://doi.org/10.1007/s11227-023-05545-0

[16] Sen, P., Islam, T., Pandit, R., & Sarddar, D. (2024). A Comparative Review on Different Techniques of Computation Offloading in Mobile Cloud Computing. *Fog Computing for Intelligent Cloud IoT Systems*, 33-44. https://doi.org/10.1002/9781394175345.ch2

[17] Alabdeli, H., Yamsani, N., Anitha, D., Chaithra, K. N., & Bindu, G. (2024, February). Intrusion Detection System in Mobile Cloud Computing Using Bat Optimization Algorithm-Support Vector Machine. In *2024 International Conference on Integrated Circuits and Communication Systems (ICICACS)* (pp. 1-4). IEEE. https://doi.org/10.1109/ICICACS60521.2024.10498365

[18] Devi, N., Dalal, S., Solanki, K., Dalal, S., Lilhore, U. K., Simaiya, S., & Nuristani, N. (2024). A systematic literature review for load balancing and task scheduling techniques in cloud computing. *Artificial Intelligence Review*, *57*(10), 276. https://doi.org/10.1007/s10462-024-10925-w

[19] Ajaz, F., Naseem, M., Shabaz, M., & Khan, M. A. (2024). An architectural view of VANETs cloud: Its models, services, applications and challenges. *International Journal of Web and Grid Services*, *20*(3), 292-341. https://doi.org/10.1504/IJWGS.2024.139779

[20] Shouyi, Y. A. N. G., Yihang, C. H. E. N., Shuangling, Z. H. A. N. G., Haojin, H. A. N., Guangyuan, L. I., & Wanming, H. A. O. (2024). Research of Mobile Edge Computing for Future Mobile Communications: A Review. *Journal of Zhengzhou University: Engineering Science*, *45*(4).

[21] Malgwi, Y. M., George, F. K., Markus, C., & Chikaodiri, O. L. (2024). AN EFFICIENT SECURITY ROUTING PROTOCOL FOR CLOUD-BASED NETWORKS USING CISCO PACKET TRACER. Information Technology, 7(2), 49-67.