Journal of Cloud Computing:
Advances, Systems and Applications

RESEARCH

Open Access

PPDNN-CRP: privacy-preserving deep neural network processing for credit risk prediction in cloud: a homomorphic encryption-based approach

Vankamamidi S. Naresh[1*] and Ayyappa D[1]

Abstract

This study proposes a Privacy-Preserving Deep Neural Network for Credit Risk Prediction (PPDNN-CRP) framework that leverages homomorphic encryption (HE) to ensure data privacy throughout the credit risk prediction process. The PPDNN-CRP framework employs the Paillier homomorphic encryption scheme to secure sensitive loan application data during both the training and inference phases. Implemented using TensorFlow for deep neural network operations and TenSEAL for HE, the framework uses the Kaggle loan dataset to evaluate its performance. The results show that PPDNN-CRP achieved an accuracy of 80.48%, demonstrating competitive performance compared to Privacy-Preserving Logistic Regression (PPLR) at 77.23% and a slight decrease from the non-private DNN-CRP model at 86.18%. The model exhibited strong metrics with a precision of 0.84, recall of 0.91, F1-score of 0.87, and an AUC of 0.74. Security analysis confirms that PPDNN-CRP effectively defends against various privacy attacks, including poisoning, evasion, membership inference, model inversion, and model extraction, through robust encryption techniques and privacy measures. This framework offers a promising approach for achieving high-quality credit risk prediction while maintaining data privacy and complying with legal and ethical standards.

Keywords Credit risk prediction, Deep neural network, Homomorphic encryption, Privacy preserving, Bank loan prediction, Cloud computing

Introduction

The modern banking industry is pivotal in the economic development of nations by providing financial assistance to individuals and businesses. A key mechanism used by banks for financial transactions is the loan system. However, the sensitive nature of borrowers' personal and financial information underscores the critical importance of ensuring data security and privacy. Deep Neural Networks (DNNs) have gained substantial attention across various domains for their ability to learn complex patterns and make precise predictions. In loan systems, DNNs emerge as powerful tools for decision-making, capable of analyzing diverse factors and historical data to assess the creditworthiness of borrowers.

While accurate predictions are integral to the loan system, addressing security concerns related to handling sensitive data is equally imperative. This is where Homomorphic Encryption (HE) assumes significance. HE facilitates computations on encrypted data without decryption, thereby preserving data privacy. This presents a significant advantage over other privacy-preserving techniques such as differential privacy, secure multi-party computation (SMPC), and federated learning in the context of credit risk prediction. Differential

*Correspondence:
Vankamamidi S. Naresh
vsnaresh111@srivasaviengg.ac.in
[1] Department of CSE, Sri Vasavi Engineering College, Tadepalligudem, Andhra Pradesh, India

privacy aims to protect individual data points by adding random noise to the dataset or query results, which can compromise the accuracy of the data analysis, leading to less precise creditworthiness assessments and suboptimal lending decisions. In contrast, HE allows for computations on exact encrypted data without introducing noise, ensuring predictions remain precise and reliable while maintaining high levels of privacy. SMPC, though effective in ensuring privacy through collaborative computations, can be complex to implement and involve significant communication overhead, posing a bottleneck in real-time systems. HE, on the other hand, offers simplicity and efficiency by enabling a single entity to handle encrypted data without complex multi-party protocols, thereby reducing latency and computational overhead. Federated learning enables multiple entities to collaboratively train models without sharing their data but still poses privacy risks through potential inference attacks and requires coordination among parties. HE surpasses these limitations by ensuring that all computations are performed on encrypted data, completely eliminating the risk of sensitive information leakage and simplifying implementation through centralized control. Thus, Homomorphic Encryption provides a robust and effective solution for privacy-preserving credit risk prediction, enhancing security, accuracy, and efficiency while ensuring the confidentiality of borrowers' personal and financial information.

Combining DNNs and Homomorphic Encryption offers a promising solution to enhance the efficiency, accuracy, and security of bank loan systems. Loan application data can be encrypted using HE techniques, ensuring the confidentiality of borrowers' information. This encrypted data serves as input for the neural network, which evaluates the loan application and provides a creditworthiness prediction without accessing personal and financial data. This approach guarantees the privacy of borrowers' information throughout the loan evaluation process while facilitating accurate creditworthiness assessments. Banks can thus make informed decisions regarding loan approvals, mitigating the risk of unauthorized access to sensitive data.

Integrating DNNs with Homomorphic Encryption presents a comprehensive solution to enhance the efficiency and security of bank loan systems. By harnessing the predictive capabilities of neural networks alongside the privacy-preserving computations of HE, banks can streamline their loan approval processes, mitigate risks associated with data breaches, and ensure the confidentiality of borrowers' personal and financial information. This symbiotic approach marks a significant advancement for the banking industry, elevating the customer experience while upholding the highest standards of data security and privacy. Overall, the trade-off for adopting PPDNN-CRP is justified due to its privacy benefits with slight accuracy drop. However, decisions should be made on a case-by-case basis, considering regulatory environments and market conditions, with a phased approach recommended for implementation.

Contributions

The main contribution of this paper is to build a HE-enabled PPDNN-CRP system with the following features:

- Proposed an HE-enabled DNN Processing Framework that can provide privacy in training and inference phase with training data privacy, model privacy, input privacy, and output privacy.
- We made a security analysis that shows that the proposed system can defend against poison, evasion, member inference, model inversion, and model extraction in the respective stages of machine learning.
- We conduct experiments using the Tenseal package on real datasets from the Kaggle to evaluate the performance of both DNN-CRP (over unencrypted data) and the proposed PPDNN-CRP.

The remainder of this paper is organized as follows: Section 2 reviews related work in machine learning for credit risk prediction, privacy-preserving techniques, and homomorphic encryption. Section 3 covers the background knowledge, including key notations and DNN workflow. Section 4 details the PPDNN-CRP framework, its architecture, data flow, and privacy mechanisms. Section 5 provides a security analysis, demonstrating resilience against various attacks. Section 6 presents implementation details and results, including dataset analysis, performance metrics, and comparative analyses. Finally, Sect. 7 concludes the paper with key findings and future research directions.

Related work

Various researchers worked on Machine learning Models for credit risk predication: Anshika Gupta et al. [1] developed a machine learning system that predicts loan application results based on credit history and income. Golak Bihari Rath et al. [2], machine learning and classification algorithms can help anticipate qualified applicants and improve loan approval processes. Mayank Anand et al. [3] use various machine learning models to predict loan default behavior in secure banking.

Syed Zamil Hasan Shoumo et al. [4] developed a machine learning model for assessing credit risk and

predicting loan defaults in the banking sector. Mohammad Abdullah et al. [5] apply machine learning approaches to anticipate nonperforming loans in financial institutions in developing nations. Their findings show that the random forest model outperforms the other models, with an accuracy of 76.10%. In a study by Bhargavet et al. [6], random forest and decision tree algorithms were compared for predicting loan approval using machine learning. The results show that Random Forest outperforms other algorithms.

Dansana et al. [7] use various loan approval criteria, such as gender, educational qualification, employment type, business kind, loan length, and marital status, to predict defaults. The Random Forest technique is used in the study to perform prediction analysis. Blessie et al. [8] use various machine learning techniques, including logistic regression, decision trees, SVM, and Naive Bayes, to forecast loan sanctions from a loan dataset. Naive Bayes is selected as the most effective model, with the highest loan predicting accuracy of 80.42%.

Zhu et al. [9] use different machine learning techniques, including logistic regression, decision trees, SVM, and Naive Bayes, to forecast loan sanctions from a loan dataset. The study identifies Naive Bayes as the most effective model, with an accuracy of 80.42% in loan forecasting. Alsaleem et al. [10] compared machine learning methods, including decision trees, random forests, Bayesian networks, and neural networks, to forecast bank loan risks using a dataset of 1000 loan applications. The results highlight the better performance of the Multilayer Perceptron neural network, which achieved an 80% accuracy, suggesting its efficacy in facilitating data-driven loan approval choices for banks. Di Wang et al. [11] created NeuCredit, a deep neural network model for evaluating and forecasting consumer credit risk in e-commerce settings. The model improves existing methods by partitioning default likelihood into willingness to repay, ability to repay, and behavioral risk variables, resulting in interpretable predictions.

Uddin et al. [12] offer an ensemble machine learning technique to predict bank loan acceptance using a Kaggle dataset. The study investigates various models, applies preprocessing approaches, and introduces two ensembles, with the top three models having a peak accuracy of 87.26%. Furthermore, the study illustrates practical implementation via a desktop program with a user interface.

Various researchers worked on Integrating Privacy Techniques such as Differential Privacy, Homomorphic Encryption into Machine Learning: Zhigang Lu et al. [13] developed a novel differentially private framework for deep learning. This entails inserting DP noise into a randomly chosen neuron in the output layer of a non-private neural network trained with a convexified loss function. Ma et al. [14] provided an overview of DP and deep learning, covering both DP and GANs. Cristiano and colleagues [15] introduce D-ZOA, a privacy-preserving distributed algorithm employing zeroth-order optimization to minimize a regularized empirical risk function. D-ZOA ensures $(\in,\partial)-$DP and surpasses the accuracy of current differentially-private methods.

Various researchers worked on Federated Learning for PPCRA: Jean-François et al. [16] propose a privacy-preserving framework for probabilistic voltage forecasting in local energy communities. The approach employs federated learning and DP strategies. Abdullah Lakhan et al. [17] propose an approach for healthcare job scheduling that combines federated learning and blockchain. The suggested system intends to protect privacy, identify fraud, and meet energy and delay restrictions in healthcare applications. Xu et al. [18] proposed μDFL, a hierarchical IoT network fabric using microchains and a hybrid consensus protocol. This system is intended to ensure efficiency, privacy, scalability, and security in the context of decentralized federated learning via IoT networks.

Other Privacy-Preserving Techniques: Wang et al. [19] propose a privacy-preserving distributed machine learning system that uses local randomization and ADMM perturbation. This approach tries to protect confidential data while providing users with varied levels of privacy. Huadi Zheng et al. [20] proposed an approach called BDP to protect a machine learning model's decision boundary by obscuring predicted responses through noise. Liu et al. [21] discuss privacy and security concerns in deep learning. Huang et al. [22] proposed a trainable picture encryption approach to protect privacy in deep learning applications, particularly for medical images. Furthermore, they have reported on enhancements to existing encryption algorithms, with a special emphasis on advances in the keyspace of images encrypted with their suggested scheme.

Various researchers worked on HE enabled NNP for PPCRA: Bernardo Pulido-Gaytan et Chandramohan et al. (2013) [23] examined data secrecy in cloud storage and proposed an evolutionary model for privacy preservation, highlighting the need for innovative approaches to enhance security. Paroda et al. (2023) [24] introduced a chaotic image encryption model that combines advanced techniques for high security against various attacks. Movassagh et al. (2020) [25] developed a neural network training algorithm that improves accuracy through optimization techniques. Alzubi et al. (2021) [25] presented a deep learning model for secure medical data transmission using homomorphic encryption. Gheisari et al. (2021) [26] created the OBPP framework to improve privacy and service quality in IoT-based smart cities.

In homomorphic encryption for credit risk analysis, Pulido-Gaytan et al. [27] provided a comprehensive survey on integrating HE with neural networks. Stephanie et al. [28] combined multiple privacy methods, while AONO et al. [29] introduced the PPLR method for secure data collaboration. Chaudhuri et al. [30] developed PPLR with differential privacy techniques to enhance accuracy. Zheng et al. [31] proposed the PCAL framework to balance privacy and utility. Han et al. [32] combined HE and MPC for secure logistic regression. Divakar et al. [33] created a cloud-based credit risk analysis method using homomorphic encryption to maintain data anonymity.

Table 1 Notations

Symbol	Description
HE	Homomorphic Encryption
DNN	Deep Neural Network
DNN-CRP	Deep Neural Network Processing for Credit Risk Prediction
PPDNN-CRP	Privacy-Preserving Deep Neural Network Processing for Credit Risk Prediction
LR	Logistic Regressions
CSP	Cloud Service Provider
DP	Differential Privacy
PII	Personally Identifiable Information
MPC	Multi Party Computation
GANs	Generative Adversarial Networks
BDP	Boundary Differential Privacy
TP	Training Phase
TSP	Testing Phase
PUK	Public Key
PRK	Private Key
PPLR	Privacy-Preserving Logistic Regressions

Background knowledge

This part gives a brief description of pallier and DNN for the proposed system.

Notations (Table 1)

Workflow of a deep neural network (DNN)

The workflow of DNN is depicted in Fig. 1 and DNN processing steps were presented as follows

DNN Processing Steps

Step 1: The input layer: In a DNN, the input layer is the first layer that receives the raw input data. It acts as the interface between the external data and the internal computation that occurs within the network.

Step2: First Hidden Layer:

Linear transformation: $Z_1 = W_1 X + b_1$

Activation function (ReLU): $A_1 = ReLu(Z_1)$

Step3: Batch Normalization (after the first hidden layer):

Batch normalization operation, typically involving scaling and shifting.

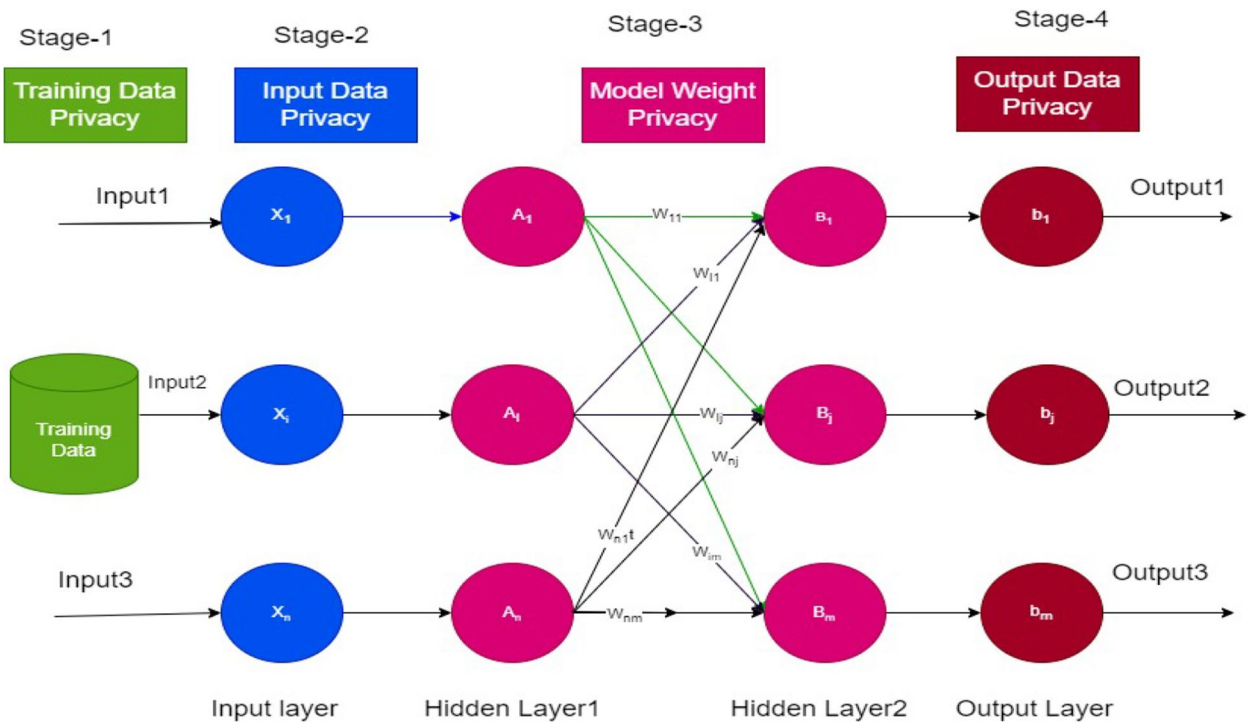


Fig. 1 Workflow of a deep neural network (DNN)

DNN Processing Steps

Step4: Dropout (after the batch normalization):

Step5: Dropout operation, randomly setting a fraction of input units to zero.

Step6: Second Hidden Layer:

Linear transformation: $Z_2 = W_2 .X + b_2$

Activation function (ReLU): $A_2 = ReLu(Z_2)$

Step7: Dropout (after the batch normalization):

Dropout operation.

Step8. Output Layer:

Linear transformation: $Z_{output} = W_{output} .A_2 + b_{output}$

Activation function: $A_{output} = sigmoid(Z_{output})$

Where:

- X is the input data.
- $w_1, b_1, w_2, b_2, w_{output}$, $utput, b_{output}$, $utput$, *are the weight matrices and bias vectors for each layer*.
- $Z_1, A_1, Z_2, A_2, Z_{output}, A_{output}$, *are the pre* $-$ *activation, post* $-$ *activation values for each layer*.
- ReLU is the rectified linear unit activation function.
- Sigmoid is the sigmoid activation function.

Paillier homomorphic encryption (PHE) cryptosystem

PHE is an asymmetric homomorphic cryptosystem [34] with key generation, encryption, and decryption module as follows:

PHE. Key Generation(λ):

1. Choose Two Large Prime Numbers: - Select two large prime numbers, p and q.

2. Compute n and λ:

- Calculate $n = pq$ and $\lambda = lcm(p-1, q-1)$

(where lcm is the least common multiple).

3. Choose g Value:

- Select a random integer g in the range $(n+1)^2$ such that $gcd(g, n) = 1$.

4. Public Key (n, g) and Private Key λ:

- The public key is (n, g), and the private key is λ.

PHE. Encryption(m,*n*):

To encrypt a plaintext m, where $0 \leq m < n$, compute the ciphertext c using the formula:

$c = (g^m \cdot r^n) mod n^2$

Where r is a random integer in the range $1 \leq r < n$.

PHE. Decryption(c,λ):

To decrypt a ciphertext , compute the plaintext using the formula:

$m = Lc(c^\lambda mod n^2)\ \mu mod n$

where $L(x) = \frac{x-1}{n}$ and μ is the modular inverse of $(g^\lambda mod n^2)$ modulo n.

Homomorphic Properties:

- Paillier encryption is homomorphic concerning addition.

Given ciphertexts c_1 and c_2 encrypting plaintexts m_1 and m_2 respectively, the product $c_1 \ mod\ n^2$ will decrypt to $m_1 + m_2$.

The homomorphic property allows computations on encrypted data without decryption, which is particularly useful in privacy-preserving applications such as Proposed Credict Risk Prediction.

DNN-CRA architecture

The block diagram stepwise processing of DNN –CRA is presented in Fig. 2 as flow chart that starts with financial dataset and then apply preprocessing and feature engineering to get the processed dataset. Next divide this dataset into training and testing through which DNN model was build and evaluated its performance.

Proposed system

In this section we have presented system model, proposed framework

System model for PPDNN-CRA

A system model involving a Bank, a Loan Applicant(user), and a CSP for building a DNN model to train on a loan application dataset with sensitive information is presented in Fig. 3. In this model, we will emphasize the importance of privacy-preserving techniques to handle sensitive data responsibly.

Entities

i. *Loan Applicants* (*Users*): that are individuals applying for bank loans provides their personal and financial information to the bank for the purpose of loan approval decision making.
ii. *Bank: a* financial institution that provides loans and other banking services whichcollects applicant data through loan applications. Owns a dataset containing sensitive information about income, assets, credit history and other financial details. And ensures secure storage of datasets. May offer additional services for privacy-preserving techniques to encrypt the dataset.
iii. *Cloud Service Provider (CSP):* a third party thatprovides cloud infrastructure for processing, and training of machine learning models and hosts the DNN model training process.

Data Flow

The loan applicant provides their information to the bank through the loan application process, allowing the collection of sensitive financial data.

The Bank aggregates and anonymizes the collected data to remove PII and reduce the risk of privacy breaches.

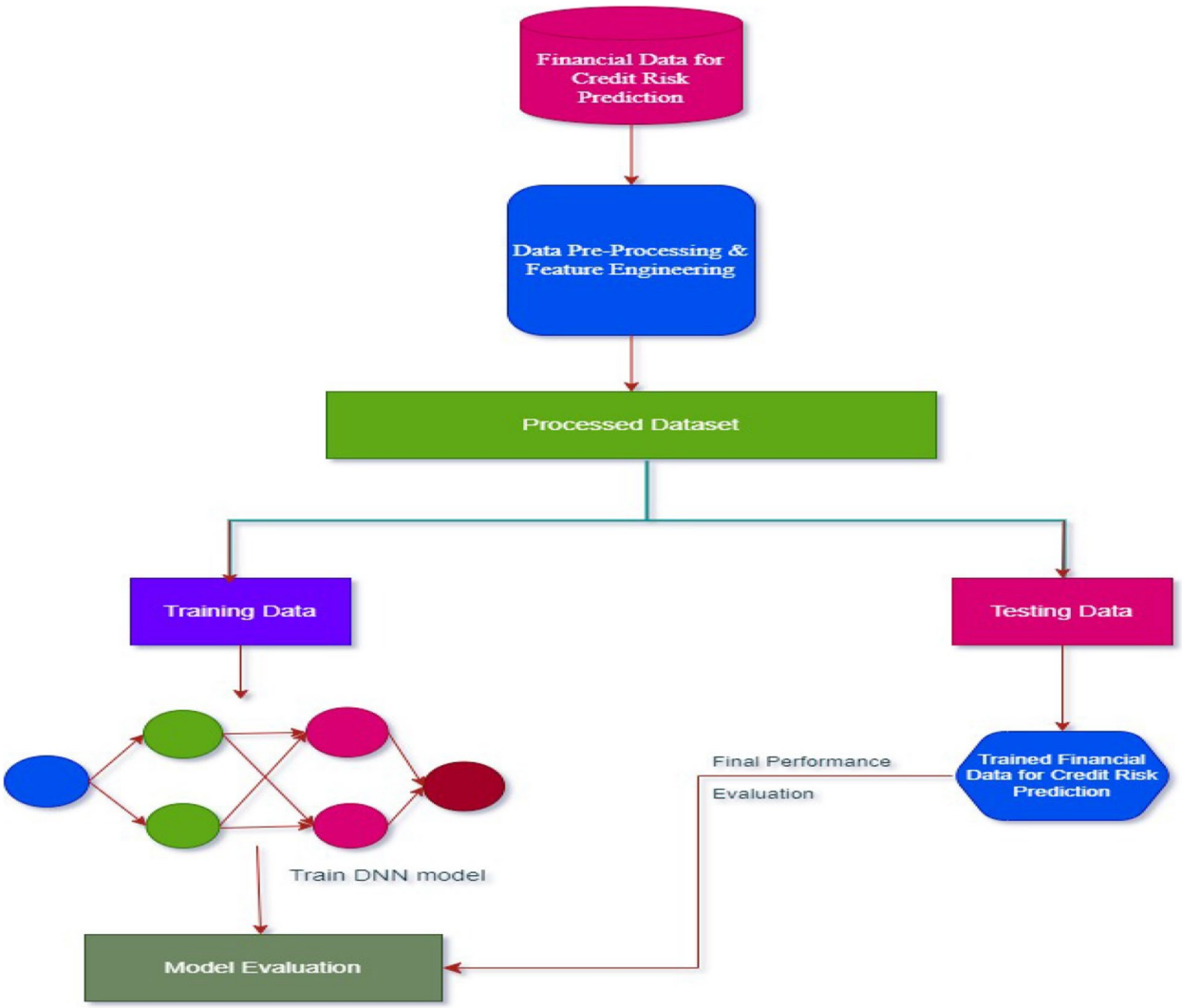


Fig. 2 Block diagram for DNN-CRA

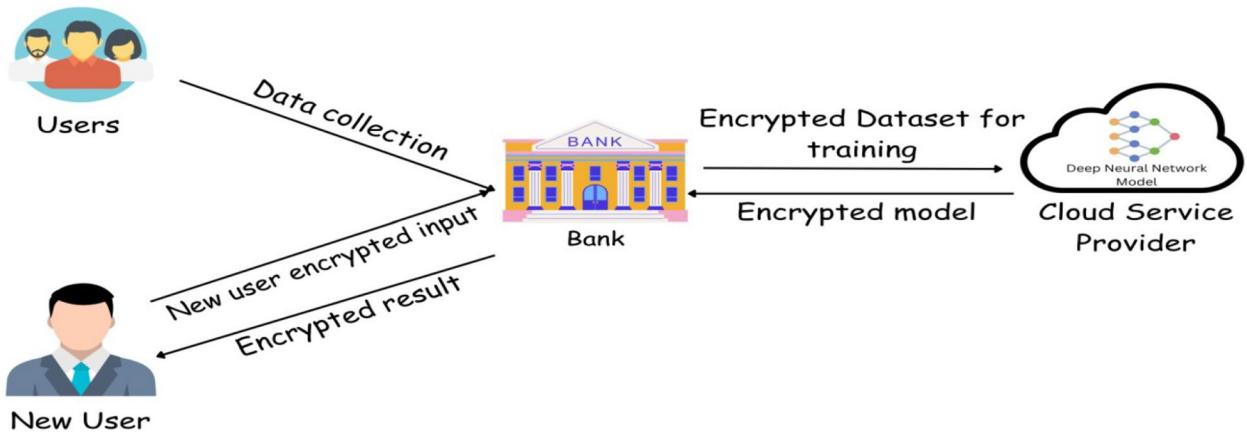


Fig. 3 System model for PPDNN-CRA

The dataset is anonymized and encrypted with the BPUK to create an encrypted dataset.

The encrypted dataset is then securely transmitted to the Cloud Service Provider, ensuring data integrity and confidentiality during the transfer process.

The Cloud Service Provider hosts the DNN model training process, utilizing the encrypted loan application dataset to develop predictive models for credit risk analysis, loan approval optimization or other relevant insights.

After training, the DNN model is deployed back to the Bank's infrastructure for local inference, reducing the need for continuous data transmission to the cloud.

Privacy-preserving techniques

Pailler Homomorphic Crypto systems is used to allow certain computations to be performed on encrypted data without decryption are utilized. This enables processing of sensitive loan application data while preserving confidentiality.

Security measures

Encryption is applied during data transmission to protect against unauthorized access or interception of the sensitive loan application dataset.

Access controls and authentication mechanisms are implemented at both the Bank and CSP ends to ensure that only authorized personnel can access and manipulate the data.

By integrating these privacy-preserving techniques and security measures into the system model, the Bank can benefit from the insights generated by the DNN model while safeguarding the privacy of loan applicants and complying with ethical and legal standards.

Privacy-preserving credit risk prediction (PPCRP) Framework

The methodology for the bank loan processing system utilizing a DNN with partially homomorphic encryption consists of two key phases as presented in Fig. 4.

Training phase

In the *training privacy* subset (steps 1 to 3), the process begins with data collection and storage in step 1, wherein the bank gathers information from local sites and securely stores as a loan dataset on its server. Feature scaling is then applied to enhance model performance, resulting in normalized datasets. Subsequently, in step 2 (data encryption), the normalized datasets undergo encryption using the banker's PUK, creating an encrypted dataset. In

PHASE	PRIVACY	STEPS
TRAINING	TRAINING	1 TO 3
	MODEL	4 TO 5
TESTING	INPUT DATA	6 TO 8
	OUTPUT DATA	9 TO 11

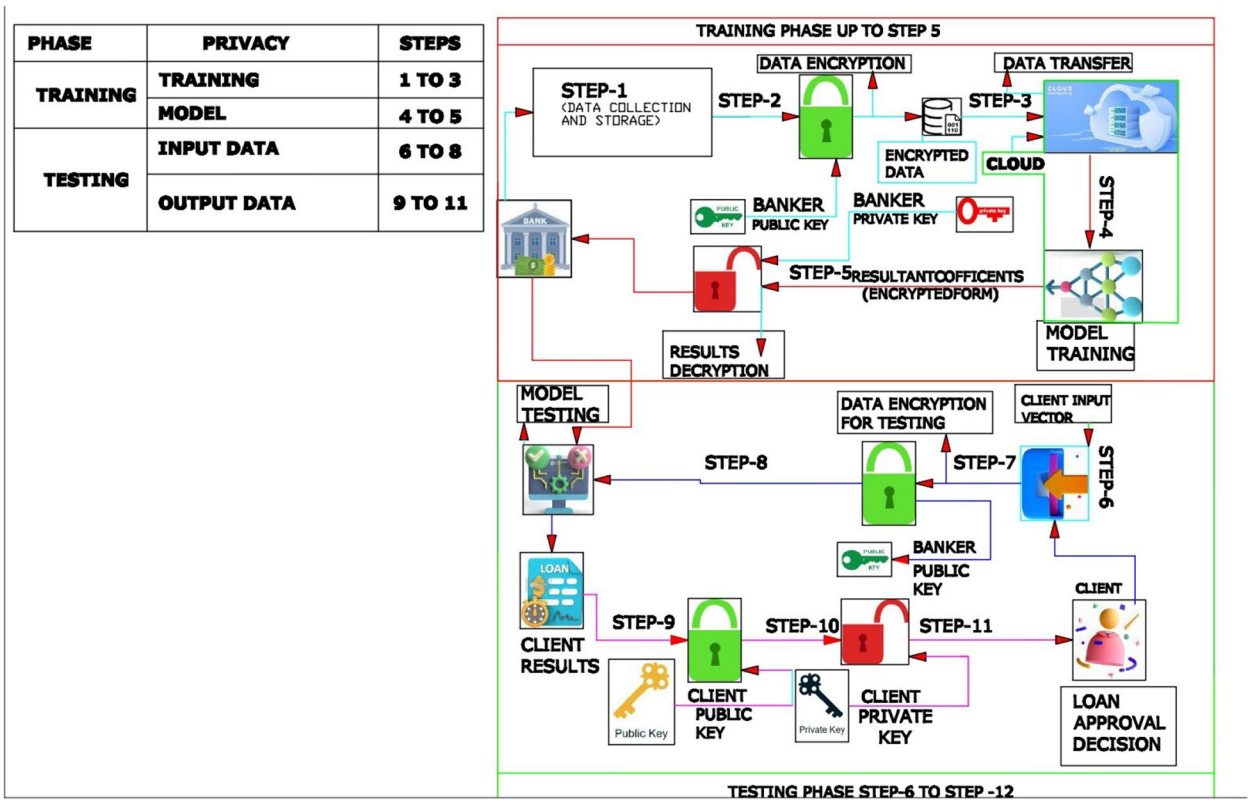


Fig. 4 Privacy-preserving credit risk analysis framework (PPCRAF)

step 3 (data transfer) involves transferring this encrypted dataset to a CSP, where a CSP provider trains the model on the encrypted dataset.

Model privacy (steps 3–5), step 4 (data decryption) involves a neural network model operating on the encrypted dataset to generate resultant coefficients. In step 5, the resultant coefficients are decrypted using the banker's PRK, marking the end of the training phase and initiating the testing phase.

Testing phase

Input data privacy (Steps 6 to 8), Step 6 (Single Client Input Vector Transfer) involves transmitting a single input vector from the client to the banker. In Step 7 (Client Data Encryption), the single-input variable is encrypted using the banker's PUK, resulting in encrypted data sent to the bankers. Step 8 (data testing): providing the encrypted single input vector to the resultant coefficients of the bank.

Output Data Privacy (Steps 9 to 11), Step 9 (Results), provides clients with loan processing outcomes. In Step 10 (Results Encryption), the obtained results are encrypted with a client-PUK, generating encrypted results sent to the client. Finally, in Step 11 (Results Decryption), clients decrypt the results using their PRK, decisively determining the banker's decision on loan approval or rejection. This methodology, seamlessly integrating a neural network with partially homomorphic encryption, ensures robust data security and privacy throughout the bank loan processing system.

Security analysis

The PPDNN-CRP framework employs a robust security model to defend against a wide array of security threats, ensuring the integrity and confidentiality of the credit risk prediction process. By leveraging Homomorphic Encryption (HE) and other advanced techniques, the framework safeguards data at various stages of machine learning. This section delves into the specific attack scenarios and the defenses employed by the system, highlighting the framework's comprehensive security measures.

Poisoning attacks

Objective: Poisoning attacks aim to manipulate the training data by injecting malicious inputs, thereby degrading the model's performance.

Defense Mechanisms:

- Homomorphic Encryption: Ensures that the training data remains encrypted and confidential during the learning process, significantly reducing the risk of adversaries injecting malicious data.
- Validation Checks and Anomaly Detection: Implement regular validation checks and anomaly detection mechanisms to identify and mitigate unusual patterns in the encrypted data.

Mathematical Concept:

- Loss Function: Defined as L(w,X,y) where w represents the model parameters, is the input dataset, and is the true label.
- Training Objective: Minimize the loss function:

$$\min_{w} L(w, X, y)$$

- Encrypted Gradient Update: The encrypted computation of the gradient and loss functions during training updates the model parameters as follows:

$$Enc(w_{new}) = Enc(w_{old}) - \eta \cdot \nabla_{Enc}(L(w_{old}, X, y))$$

This encrypted update rule prevents adversaries from manipulating gradients or injecting poisoned data, as they do not have access to the decrypted data.

Evasion attacks

Objective: Evasion attacks aim to manipulate input data to deceive the model into making incorrect predictions.

Defense Mechanisms:

- Confidentiality of Model Parameters: HE preserves the confidentiality of the model's parameters and gradients, preventing adversaries from gaining insights necessary for crafting malicious inputs.
- Encryption Techniques: The robust encryption techniques used in PPDNN-CRP make it extremely difficult for attackers to manipulate inputs in ways that would affect the model's predictions.

Mathematical Concept: - Prediction Function: Expressed as M(X, Enc(w)), where X is the input data and Enc(w) represents the encrypted model parameters.

- Prediction on Encrypted Data:

$$Enc(\hat{y}) = M(X, Enc(w))$$

Without access to the decryption key, adversaries are unable to manipulate $\hat{y}$, thereby securing the model from evasion attacks.

Membership inference attacks

Objective: Membership inference attacks attempt to determine whether a specific data point was part of the training set.

Defense Mechanisms:

- Homomorphic Encryption: Ensures that even if an attacker gains access to the model's output, they cannot infer whether a particular data point was included in the training data.
- Differential Privacy (DP): Adds noise to the training process to obscure individual records and protect against membership inference.

Mathematical Concept:

- Probability Distribution of Predictions: Expressed as $P(Enc(y)|X, Enc(w))$. DP mechanisms ensure that for two datasets differing by a single record, the ratio of the probabilities is bounded:

$$\left|\log\left(\frac{P(Enc(y)|X, Enc(w), D_1)}{P(Enc(y)|X, Enc(w), D_2)}\right)\right| \leq \epsilon$$

This noise addition preserves privacy and limits the attacker's ability to infer membership status.

Model inversion attacks

Objective: Model inversion attacks attempt to extract sensitive information about the training data from the model's parameters.

Defense Mechanisms:

- Encryption of Model Parameters: HE ensures that model parameters remain encrypted and inaccessible, preventing attackers from extracting information about the training data.
- Obfuscation Techniques: Further obscure the relationships between model inputs and outputs, mitigating the risk of model inversion.

Mathematical Concept:

- Inversion Function: Consider a function $G(Enc(w), Enc(X))$ that attempts to invert the model's parameters and input data. Security against model inversion relies on the difficulty of solving:

$$Enc(w) = G^{-1}(Enc(X), Enc(\widehat{y}))$$

Homomorphic Encryption makes it challenging for attackers to retrieve meaningful data from the encrypted parameters.

Model extraction attacks

Objective: Model extraction attacks aim to recreate the model's architecture and parameters.

Defense Mechanisms:

- Confidentiality of Model Architecture and Parameters: HE ensures that the model's architecture and parameters are kept confidential.
- Secure Access Controls and Storage: Implement strict access controls and key management practices to safeguard the decryption key, preventing unauthorized extraction of the model's internals.

Mathematical Concept:

- Extraction Attempt: Described as $E(Enc(X), Enc(\widehat{y}))$, aiming to recover the model parameters. Security against extraction attacks is based on the difficulty of solving:

$$Enc(w) = E^{-1}(Enc(X), Enc(\widehat{y}))$$

Strict access controls and key management practices protect against unauthorized extraction of the model parameters.

The PHE-enabled PPDNN-CRP system employs a comprehensive suite of security measures, including Homomorphic Encryption, Differential Privacy, anomaly detection, and obfuscation techniques, to defend against a variety of attacks. Each attack scenario is addressed through specific defenses that leverage advanced cryptographic principles and privacy-preserving technologies. The mathematical concepts underlying these defenses demonstrate how HE and related mechanisms work together to protect against poisoning, evasion, membership inference, model inversion, and model extraction attacks. This robust security framework ensures the integrity and confidentiality of the credit risk prediction process, making PPDNN-CRP a highly secure solution for sensitive financial data.

Implementation and results

In this session, we present a dataset and perform analysis, the implementation of various models, and their performance metrics. training and validation, accuracy, and performance, Layer-wise and epoch-wise analysis and

comparison of proposed models in various performance metrics and aspects:

Dataset and it's feature analysis

Loan dataset is derived from Kaggle [35], exhibits features gathered from loan applications, while offers an overview of the dataset—614 rows and 13 columns, split into 80% training and 20% testing. Notably, an imbalance exists, with more approved loans. Mitigating this, min-max scaling normalizes features in a DNN model with 117 input nodes, two hidden layers, and one output layer. ReLU activates the hidden layers, and rmsprop activation function is used in the output layer. Optimized via stochastic gradient descent, the DNN, employing categorical cross-entropy loss and dropout for regularization, trains on 491 scaled samples. Evaluation on 123 unseen scaled test samples gauges generalization performance in terms of accuracy, precision, recall and F-score.

Heat map correlation matrix

Figure 5 shows a heatmap correlation matrix indicating the relationships between different variables in a dataset used for credit risk prediction. The matrix highlights that no two variables are strongly correlated, making it clear that we have kept all variables for the credit risk prediction model.

Implementation

Proposed DNN-CRP and PPDNN-CRP were implemented using the Keras library with a TensorFlow backend. The architecture comprises three dense layers with rectified linear unit (ReLU) activation functions. For the DNN-CRP model, the input layer has 117 neurons, the hidden layer has 104 neurons, and the output layer has one neuron with a sigmoid activation function suitable for credit risk prediction. In other hand, the PPDNN-CRP model has corresponding layers with 91, 78, and

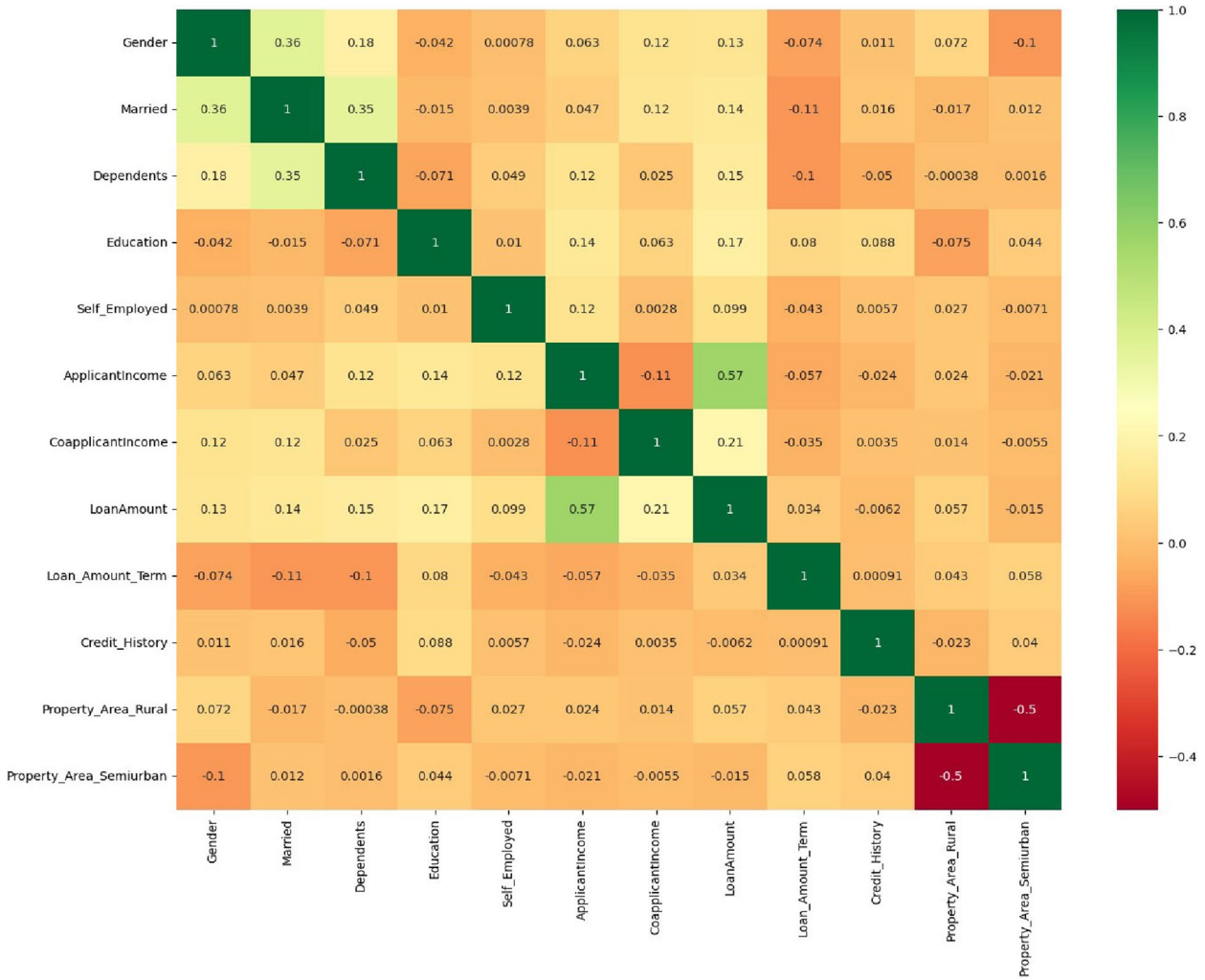


Fig. 5 Heatmap correlation matrix

1 neuron, respectively. Batch normalization is applied after the input and hidden layers to enhance training stability, and dropout layers are incorporated to prevent overfitting.

The model is compiled using the rmsprop optimizer and binary cross-entropy as the loss function, a standard choice for binary classification problems. During training, the accuracy metric is employed, and the model undergoes 100 epochs with a batch size of 26, with monitoring of the validation data (x_test and y_test).

The primary goal is to optimize model parameters, minimize binary cross-entropy loss, and achieve high accuracy in predicting binary outcomes.

Layer-wise and epoch-wise accuracy during the training of proposed models

Exploring layer-wise and epoch-wise implementation during the training of proposed models offers a detailed understanding of how the model's performance evolves across different layers and training iterations.

i) Layer-wise and epoch-wise accuracy during the training of DNN-CRP:

The Fig. 6, provides a comprehensive view of the DNN-CRP performance over 100 epochs. Each epoch reports the accuracy of various layers, such as dense_262, batch_normalization_196, dropout_196, and dense_263, among others. The model demonstrates a consistent improvement in accuracy throughout training, reaching a peak accuracy of 80.65% at epoch 95.

The first few epochs show a steady increase in accuracy across different layers, indicating effective learning. The model exhibits resilience against overfitting, with validation accuracy aligning closely with training accuracy. Notably, the accuracy values for batch_normalization_197, dropout_197, and dense_264 consistently match their counterparts in the preceding layers, emphasizing the model's uniform learning across different components.

Figure 6 serves as a valuable visual aid, offering a clear depiction of the model's convergence and accuracy trends. This comprehensive overview aids in understanding the DNN dynamic behavior and provides insights into the learning patterns of individual components, contributing to the model's overall success in credit risk prediction.

ii) Layer-wise and Epoch-wise Accuracy Analysis for PPDNN-CRP:

Figure 7 presents a thorough comparison of accuracy across different layers and epochs in a PPDNN-CRP model subject to privacy constraints. The model encompasses layers such as dense, batch normalization, and dropout, with accuracy measured over 100 epochs. Accuracy values span from 0.547861516 in the initial epoch to 0.826882601 in the final one.

Notably, the accuracy exhibits improvement with each epoch, signifying the model's ongoing learning progress. Different layers, including dense and batch normalization, showcase distinct accuracy levels, illustrating their respective contributions to the overall model performance. This in-depth examination of layer-specific accuracy offers valuable insights into the impact of privacy constraints on the model's learning dynamics. It assists in enhancing the PPDNN-CRP for credit risk prediction while maintaining privacy through the use of PHE.

Training and validation accuracy

A graphical representation of the training and validation accuracy over epochs for the proposed models.

i. Training vs. Validation Accuracy for DNN-CRP

Training and validation accuracy graphs with reference to Epoch was presented in the Fig. 8.

The DNN model underwent training from 0 to 100 epochs, with the training accuracy in the range [0.778, 0.8416]. While the training accuracy exhibited noticeable variations across epochs, it generally demonstrated an upward trend, peaking at 0.8416 by the end of the training.

In contrast, the validation accuracy displayed greater consistency and reached saturation earlier. This indicates that during the initial eighteen epochs, the model's generalization performance was at its optimal level. The model exhibited some overfitting, as evidenced by its superior performance on the training data compared to the validation data. However, this disparity diminished in subsequent epochs, indicating effective control over overfitting. With a validation set accuracy of 0.8618, the model showcased strong performance, suggesting the acquisition of efficient credit risk prediction skills that generalized well to new, unobserved data.

For DNN-CRP the validation accuracy is plateau and efficient overfitting management is evident.

ii. Training and validation accuracy performance of PPDNN-CRP is presented in Fig. 9

The training and validation accuracy patterns show the PPDNN-CRP model's continuous learning process. In the early epochs, both accuracies increased as the model understood underlying data patterns, with training accuracy increasing from 53% to over 80% upto epoch 20 and validation accuracy peaking at 86–87% upto epoch 25.

The small gap between training and validation accuracy throughout the first 25 epochs implies that the model is

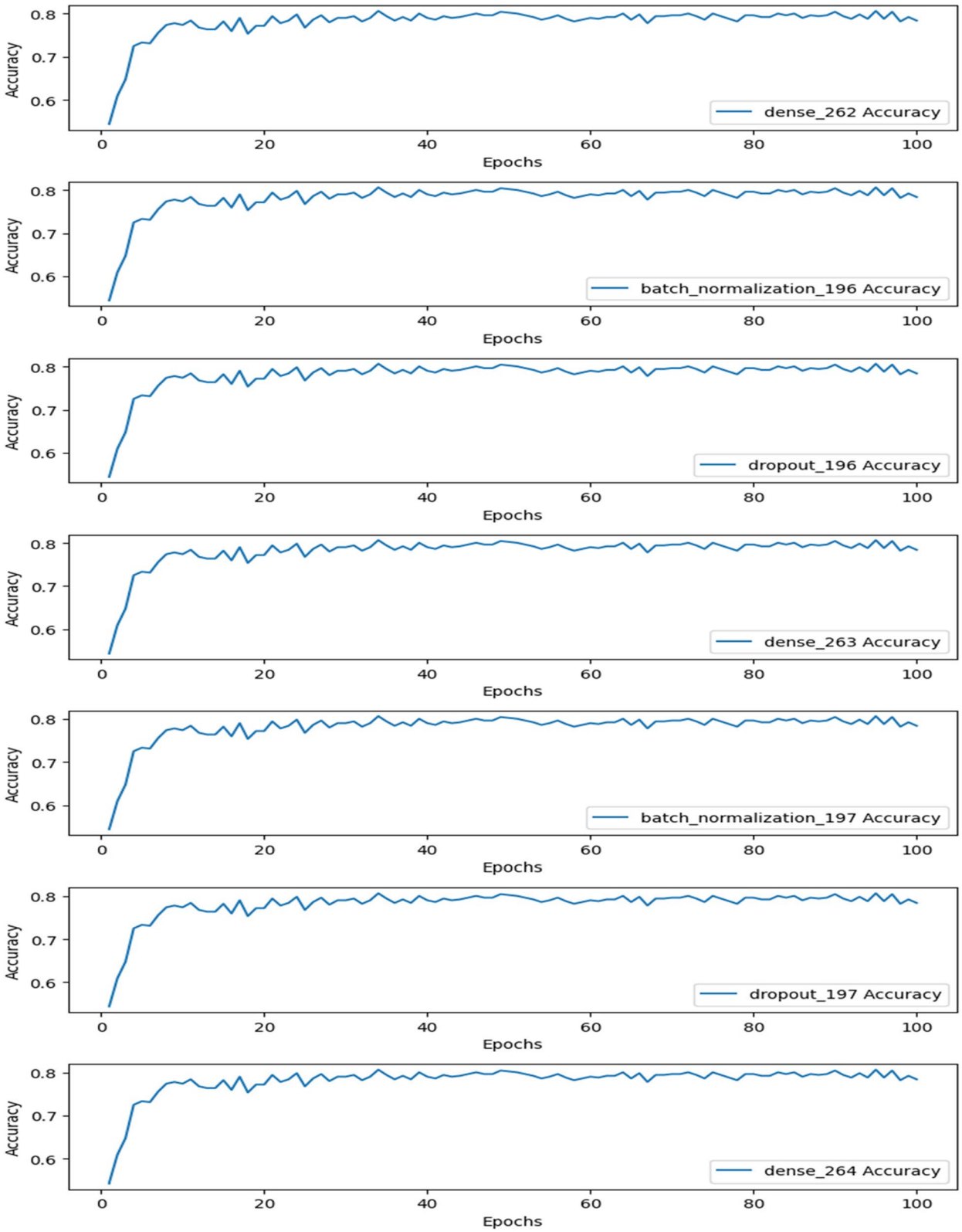


Fig. 6 Layer-wise and epoch-wise accuracy analysis for DNN-CRP

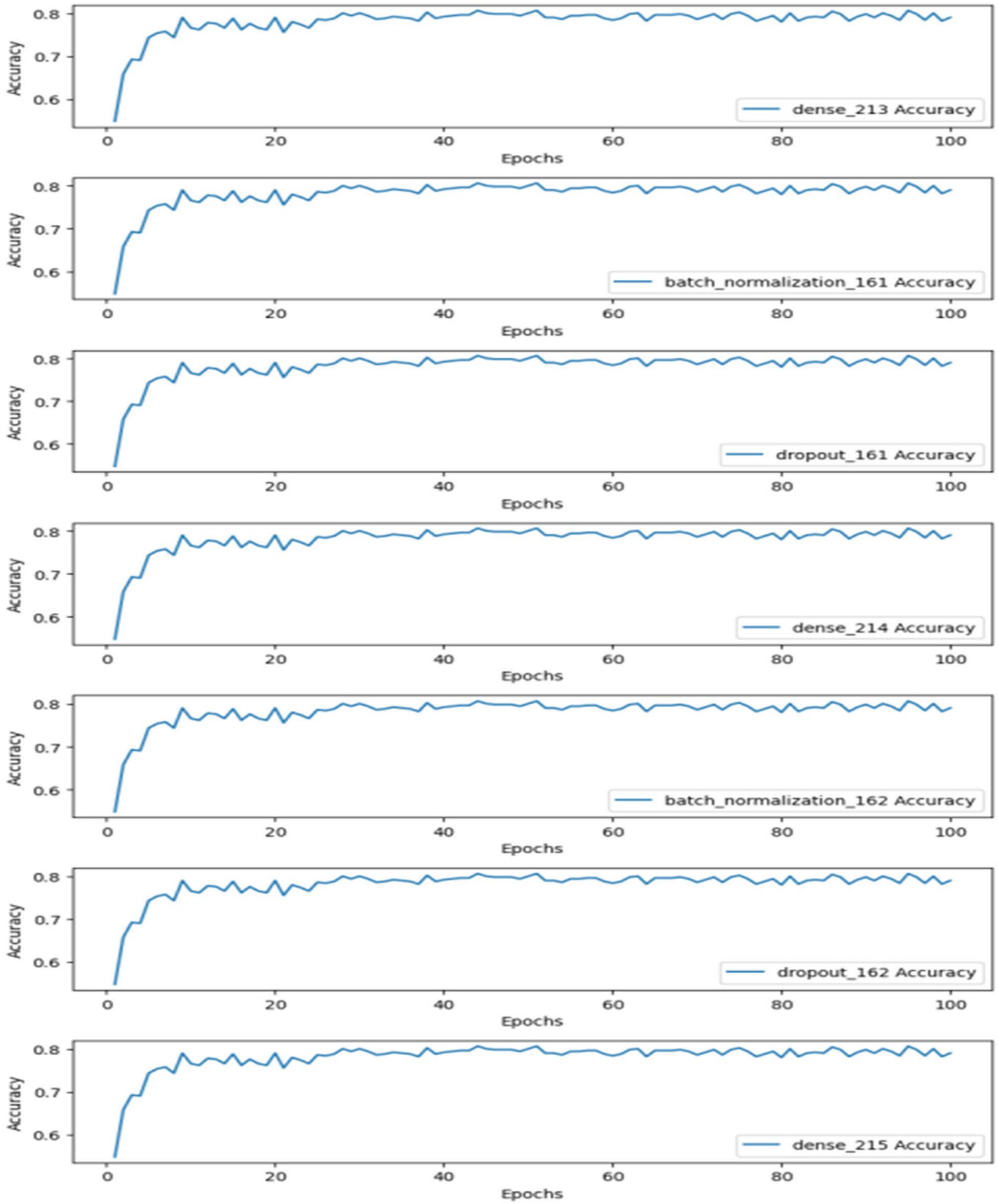


Fig. 7 Layer-wise and epoch-wise accuracy analysis for PPDNN-CRP

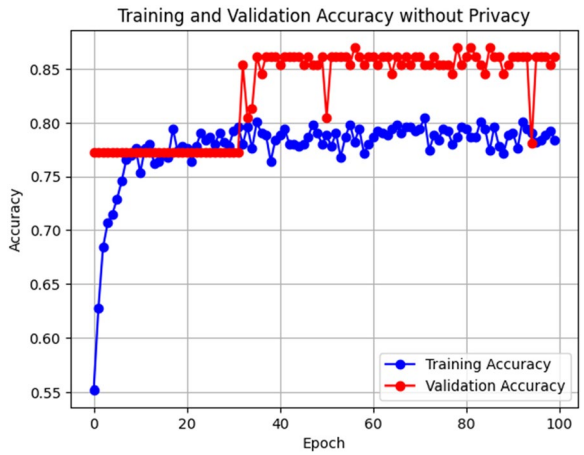


Fig. 8 Training accuracy and validation accuracy graph for DNN-CRP

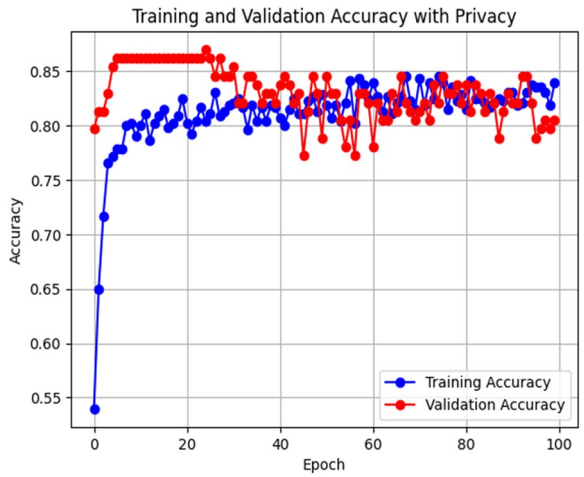


Fig. 9 Training accuracy and validation accuracy graph for PPDNN-CRP

fitting genuine data signals without major overfitting. However, at period 25, training accuracy increased while validation accuracy plateaued at 78%, indicating overfitting and input learning.

By epoch 100, the widening gap demonstrated the model's increasing reliance on irrelevant training data details. The optimal stopping point, identified around 25 epochs, showcased a balance between fitting meaningful patterns and avoiding overfitting. Further training led to diminishing returns, with improved training accuracy failing to generalize to enhanced validation performance.

In essence, the model demonstrated initial meaningful learning followed by overfitting, illustrating the crucial role of early action to achieve the best balance between underfitting and overfitting for successful credit risk prediction.

iii. Comparative Analysis of Training and Validation Accuracy Performance of PPDNN-CRP vs. DNN-CRP

Figure 10 illustrates the accuracy outcomes of DNN-CRP, emphasizing the disparities in training and validation accuracy across 100 epochs for the credit risk prediction model. The comparison is made between the DNN-CRP model operating DNN and the same model incorporating PHE for privacy.

In the absence of privacy protection, the DNN-CRP demonstrates higher and consistently accurate outcomes, indicating effective learning. The training accuracy ranges from 0.778 to 0.8416, while validation accuracy stabilizes between 0.8455 and 0.8618 after initial fluctuations.

Conversely, the PPDNN-CRP model, utilizing PHE, begins with a lower training accuracy of 0.5397 but steadily progresses to 0.8512 by the 100th epoch. The validation accuracy remains relatively stable, consistently surpassing 0.8 after epoch 20. Despite some observed overfitting, the PPDNN-CRP model exhibits a remarkable ability to learn from encrypted data.

In comparison, the DNN-CRP exhibits less overfitting, as shown by a smaller gap between training and validation accuracy. In conclusion, the PPDNN-CRP model shows reasonable accuracy, demonstrating its potential for credit risk prediction while maintaining data privacy via homomorphic encryption. Although the DNN-CRP model performs somewhat better overall, the balance of privacy and usability makes the safety-enhanced model an important consideration for safe credit risk assessment, as shown in Fig. 10.

Training loss and validation loss

Training Loss and Validation Loss Graphs for Proposed Model. It provides insights into the model's learning process and generalization performance.

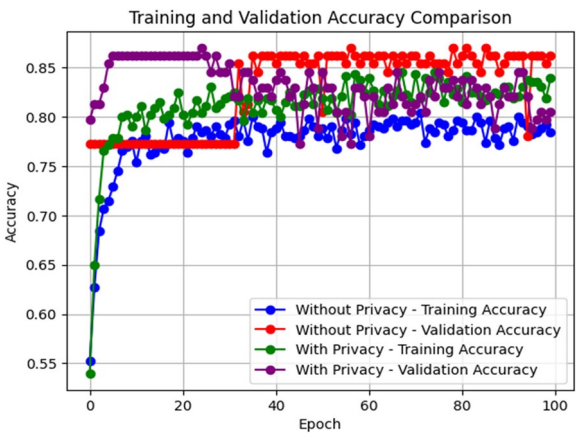


Fig. 10 Comparative analysis of training and validation accuracy for PPDNN-CRP and DNN-CRP

i. Training Loss and Validation Loss Graphs for DNN-CRP presented in Fig. 11

The training and validation loss data for the DNN-CRP model, which runs 100 epochs, is clearly reported. The training loss steadily reduces from 70.2209 to 1.0215, demonstrating the model's capacity to digest information from the training set and improve performance. Simultaneously, the validation loss decreases from 62.0963 to 0.9431, exhibiting greater fluctuation but getting around 1.0 around epoch 20.

The noticeable difference between the training and validation losses indicates some overfitting, as the training loss falls faster than the validation loss. However, as epochs advance, the validation loss decreases, showing strong generalization. In essence, the decreasing curves of both training and validation losses demonstrate the DNN-CRP effectiveness in pattern recognition and credit risk prediction.

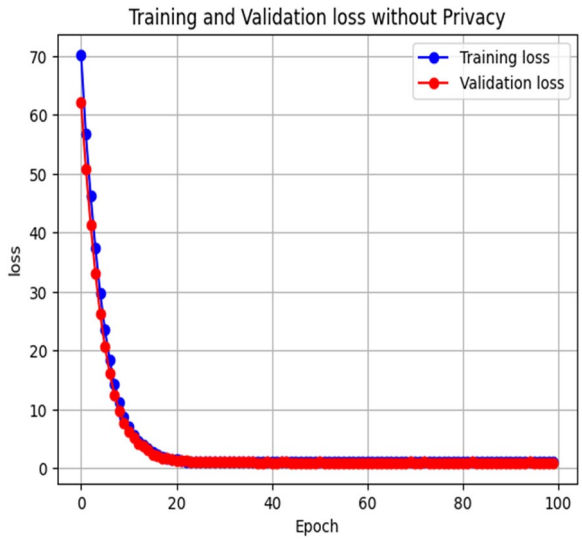


Fig. 11 Training loss and validation loss analysis graph for DNN-CRP"

ii. Training Loss and Validation Loss Graphs for PPDNN-CRP

Figure 12 shows the PPDNN-CRP model's training and validation loss over 100 epochs, with a focus on confidentiality via homomorphic encryption. The training loss has decreased from 4.6868 to 0.4299, indicating that the private model is effective at finding patterns in encrypted training data. The validation loss follows a similar decreasing trend over the epochs, ranging from 4.1436 to 0.8603, with a peak of roughly 0.8 around epoch 20.Despite showing overfitting in the private model, as seen in the gap between the training and validation loss curves, these losses gradually diminish over time, indicating an acceptable level of generalization. In conclusion, the private PPDNN-CRP model shows promise for robust learning from encrypted data and effective credit risk prediction.

Comparison of training loss and validation loss performances for the models with privacy (PPDNN-CRP) and without privacy (DNN-CRP)

Over the course of 100 epochs, the DNN-CRP model exhibits a reduction in validation loss from 62.0963 to 0.9431, and a gradual decline in training loss from 70.2209 to 1.0215. These trends underscore the model's effective learning and generalization from the original data.

On the other end, the PPDNN-CRP model, employing PHE, starts with a lower training loss of 4.6868, eventually decreasing to 0.4299. The validation loss similarly decreases from 4.1436 to 0.8603. This highlights the

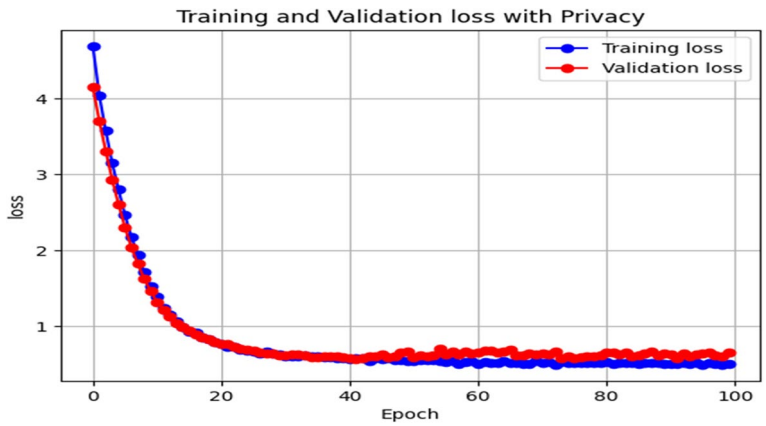


Fig. 12 Training loss and validation loss analysis graph for PPDNN-CRP

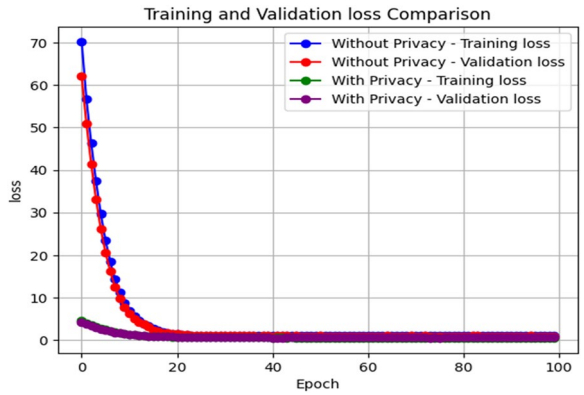


Fig. 13 Comparison analysis graph for training loss and validation loss performances for PPDNN-CRP And DNN-CRP is presented in Fig. 13

PPDNN-CRP capacity to discern patterns and make forecasts based on encrypted data.

However, the training and validation curves of the DNN -CRP model appear smoother with smaller gaps, indicating less overfitting compared to the private model, which shows greater variations between training and validation loss.

Both models exhibit respectable declining trends, reflecting their ability for learning and generalization. While the DNN-CRP demonstrates slightly more overfitting than the privacy-preserving model, it still maintains usable accuracy.

Comparative analysis performance metrics for the proposed models

Performing a comparative analysis of performance metrics for a comprehensive assessment of different models.

Emphasis of various metrics used for evolution

In evaluating the performance of the PPDNN-CRP model for Credit Risk Prediction, various metrics provide a comprehensive understanding of its effectiveness:

- *Accuracy* measures the proportion of correct predictions, giving a general sense of the model's overall performance.
- *Precision* assesses the accuracy of positive predictions, indicating the model's ability to avoid false positives.
- *Recall* evaluates how well the model identifies all relevant positive cases, reflecting its capability to capture true positives.
- *F1-Score* balances precision and recall, offering a single metric that accounts for both false positives and false negatives.
- *Area Under the Curve (AUC)* reveals the model's ability to distinguish between different classes, demonstrating its discriminative power.

These metrics collectively provide a robust framework for assessing the PPDNN-CRP model's predictive accuracy, reliability, and overall effectiveness in ensuring both high performance and data privacy.

Comparison of performance metrics of PPLR, DNN-CRP and PPDNN-CRP

Figure 14 Presents the Confusion matrix. of the Three models.

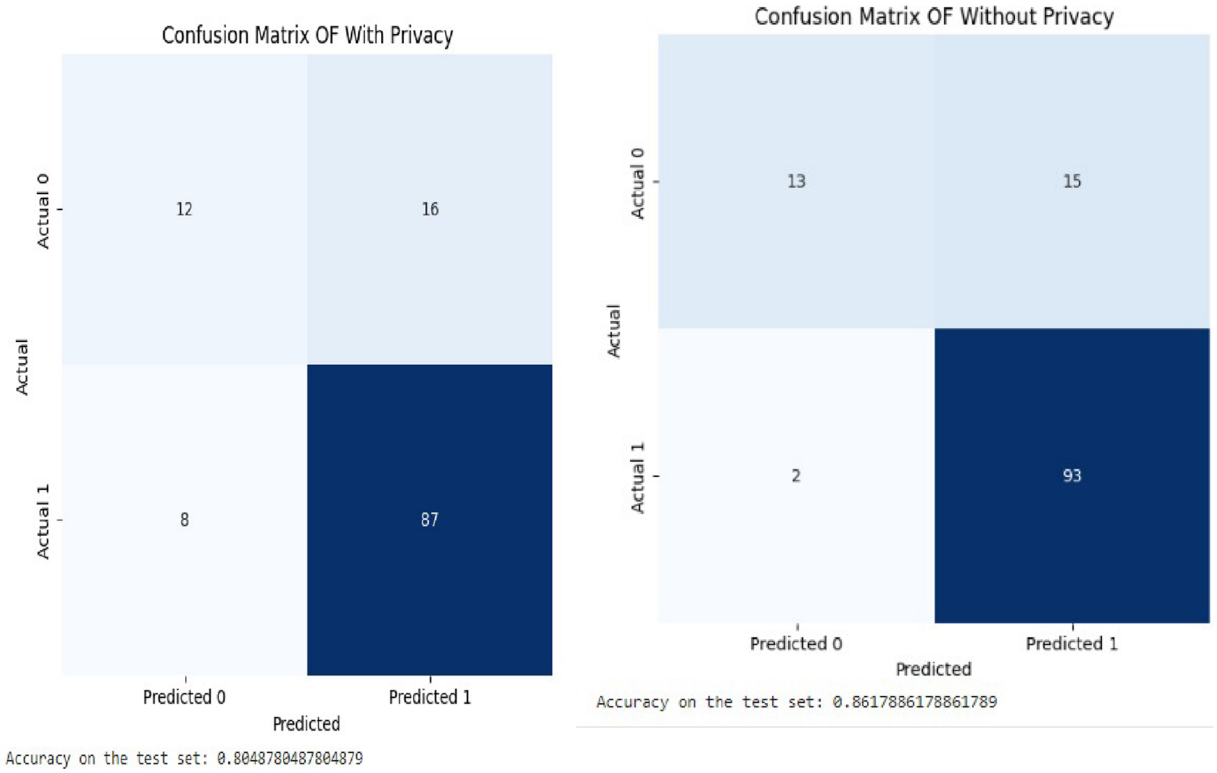
Table 2; Figs. 15, 16, and 17 present the performance metrics—Precision, Recall, F1-score, Accuracy, AUC, and ROC—for the three models: PPLR, DNN-CRP, and PPDNN-CRP. Among these, DNN-CRP achieves the highest precision at 0.86 and recall at 0.97, indicating its strong ability to identify true positives and actual defaults. In comparison, PPDNN-CRP shows competitive results with a precision of 0.84 and a recall of 0.91, reflecting its effective performance in default detection while maintaining an accuracy of 80.48%. PPLR, however, lags behind with a precision of 0.74, recall of 0.87, and accuracy of 77.23%. The F1-scores further confirm these trends, with DNN-CRP scoring 0.91, PPDNN-CRP at 0.87, and PPLR at 0.79. In terms of AUC, DNN-CRP excels with a value of 0.83, while PPDNN-CRP and PPLR score 0.74 and 0.65, respectively, indicating varying levels of model robustness.

The solid performance of PPDNN-CRP can be largely attributed to its implementation of Homomorphic Encryption (HE). This technique allows for the secure processing of sensitive data without compromising model accuracy, enabling effective learning from encrypted datasets. HE offers distinct advantages over other privacy-preserving methods by preserving data utility during computations and facilitating model training without exposing raw data. While DNN-CRP outperforms PPDNN-CRP in several metrics, the latter's integration of HE positions it as a compelling option for credit risk prediction, balancing the critical need for data privacy with reliable predictive performance.

Epoch-wise computation time analysis for proposed model

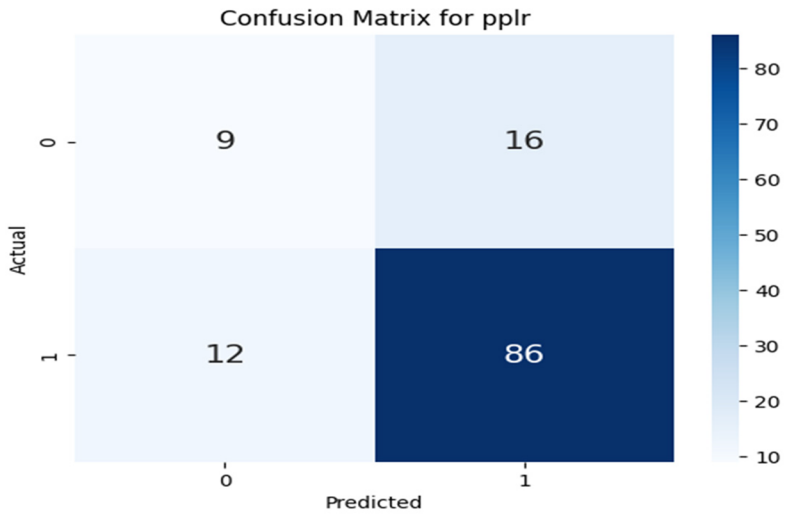
Epoch-wise computation time analysis for DNN-CRP model training model

The Fig. 18 explains the time required for training epochs in a DNN, where each epoch signifies one complete cycle



(a) DNN -CRP

(b) PPDNN-CRP



(c) PPLR-CRP

Fig. 14 Confusion matrix of the three models

Table 2 Performance metrics overview

ML Model	Performance metric parameters				
	Precision	Recall	F-measure	Accuracy (%)	AUC
PPLR	0.74	0.87	0.79	77.23	0.65
DNN-CRP	0.86	0.97	0.91	86.18	0.83
PPDNN-CRP	0.84	0.91	0.87	80.48	0.74

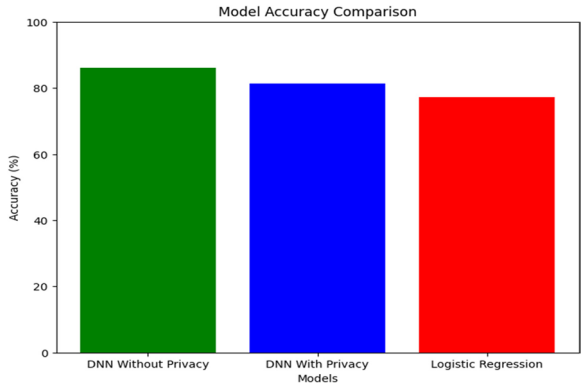


Fig. 15 Performance comparsion between PPLR, DNN-CRP and PPDNN-CRP

through the entire training data. Across 100 epochs, the first epoch stood out as the lengthiest, taking 6.34 s. Subsequently, the majority of epochs exhibited a significantly shorter duration, approximately 0.28 s each.

This pattern indicates that during the initial epoch, the DNN dedicated time to learning patterns within the credit risk data. The longer duration of the first epoch was attributed to the network adjusting its internal weights. By the second epoch, the network had already acquired substantial knowledge, resulting in faster subsequent epochs as it reinforced existing learnings.

The swift processing observed after the initial learning phase underscores the efficiency of DNN in credit risk prediction. Once the intricate patterns are learned, a DNN can rapidly process new applicant data. This aligns with the notion conveyed in the title, emphasizing that DNN excel in processing data for credit risk forecasting. The timing data serves as evidence that DNN effectively learn relationships and subsequently make rapid predictions regarding credit risk.

Epoch-wise computation time analysis for PPDNN-CRP Model Training

The Fig. 19 explains from the initial epoch had the longest duration, taking 0.3818 s. The subsequent epochs were relatively faster, ranging from 0.22 to 0.23 s. After epoch 20, the durations increased to a range of 0.3 to 0.4 s for several epochs. Following this, the times decreased once again and remained consistent between 0.2 and 0.25 s for the remaining epochs.

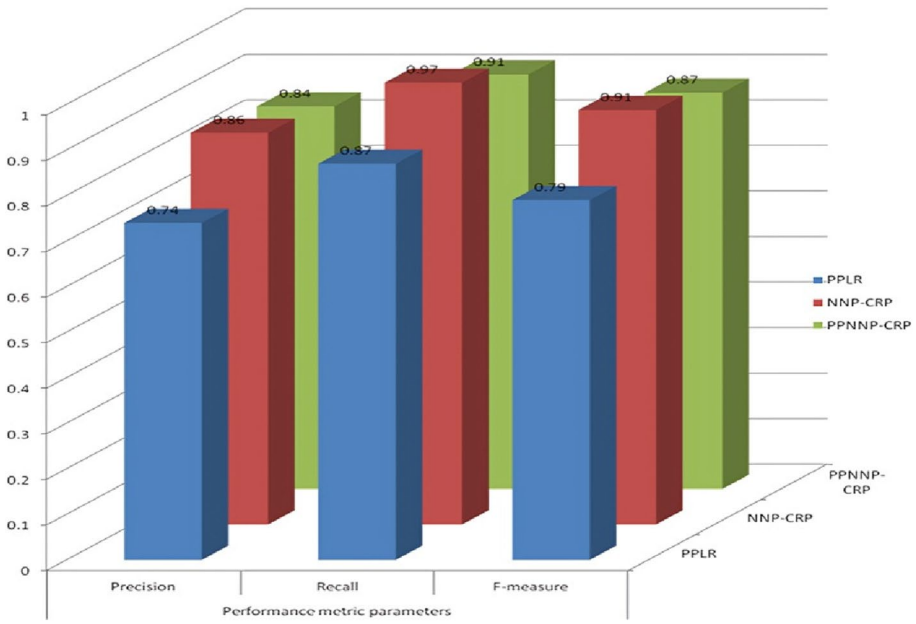


Fig. 16 Performance metric parameters comparison of PPLR, DNN-CRP and PPDNN-CRP

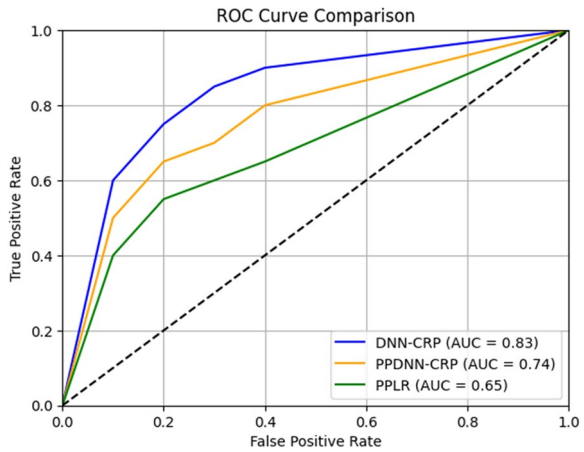


Fig. 17 Comparing ROC Curves: PPDNN-CRP, DNN-CRP and PPLR

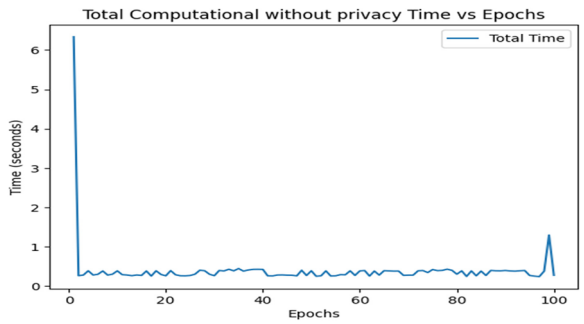


Fig. 18 "Epoch-wise computation time graph analysis for DNN-CRP model training"

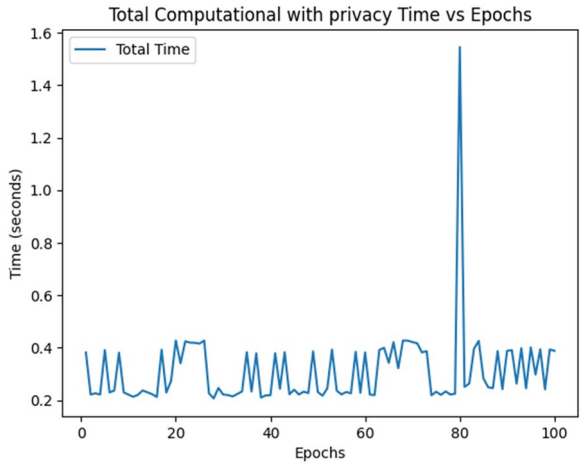


Fig. 19 Epoch-wise computation time graph analysis for PPDNN-CRP model training

The extended durations at the beginning and middle phases signify periods when the PPDNN-CRP was comprehending intricate patterns within the encrypted credit risk data. The subsequent faster durations suggest that, after a certain point (epoch 20), the network had learned the patterns and was primarily reinforcing its existing knowledge.

In essence, the timing patterns illustrate the PPDNN-CRP ability to process homomorphically encrypted data for credit risk predictions. The initial learning phase requires more time, but once the complex patterns are understood, subsequent predictions occur rapidly. This aligns with the theme conveyed in the title, asserting that PPDNN-CRP Model are proficient in handling encrypted data for credit risk forecasting.

Comparative analysis of computation time for DNN-CRP and PPDNN-CRP

Figure 20 provides a comparison of epoch times for a DNN trained on credit risk data, with one set conducted DNN-CRP and the other utilizing homomorphic encryption for privacy. In the absence of encryption, the initial epoch takes 2.4449 s, followed by faster durations in later epochs, ranging from 0.25 to 0.4 s.

Conversely, when HE is employed, the first epoch is slower, taking 1.6726 s. Subsequent epochs, however, exhibit similar durations, ranging from 0.2 to 0.4 s.This comparison reveals that encryption introduces a slight delay during the initial learning phase. However, after training, prediction times become comparable. Consequently, neural networks demonstrate the capability to process encrypted data for credit risk prediction. The titles accurately convey DNN-CRP, with only a minimal impact from privacy encryption.

Discussion on results

he PPDNN-CRP model offers a notable balance between accuracy in credit risk prediction and data privacy through Homomorphic Encryption (HE). When evaluated against the DNN-CRP and PPLR-CRP models, several trade-offs and benefits emerge.

The DNN-CRP model, which does not prioritize data privacy, exhibits the highest precision and recall, showcasing its strong predictive capabilities and ability to identify true positives. The PPDNN-CRP model, integrating HE for data privacy, shows a slight reduction in these metrics, reflecting the computational complexity introduced by encryption. Despite this, the PPDNN-CRP model surpasses the PPLR-CRP model in precision, recall, and accuracy, suggesting that HE strikes a better balance between accuracy and privacy than differential privacy.

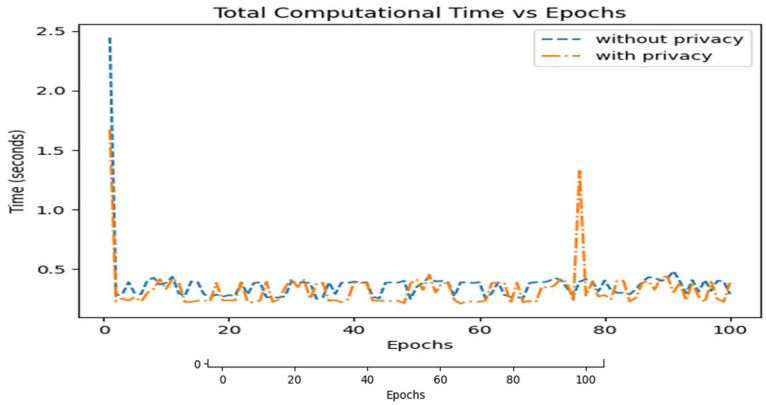


Fig. 20 Comparative analysis of computation time for DNN-CRP and PPDNN-CRP

Regarding precision, recall, and F1-Score, the PPDNN-CRP model performs well, though slightly lower than the DNN-CRP model, while outperforming the PPLR-CRP model. This highlights HE's effectiveness in maintaining high predictive performance while ensuring data privacy.

The ROC-AUC analysis indicates that the DNN-CRP model has good discriminative power, with the PPDNN-CRP model also demonstrating fair performance, though with a reduction due to privacy preservation. Nevertheless, the PPDNN-CRP model's performance remains acceptable given the privacy advantages.

Epoch-wise computation time analysis shows that the initial learning phase is more time-consuming for both models, with subsequent epochs becoming significantly faster. The delay introduced by encryption in the PPDNN-CRP model's initial epoch is minimal and manageable.

In summary, the PPDNN-CRP model achieves an optimal balance between accuracy and privacy, outperforming the PPLR-CRP model and approaching the performance of the DNN-CRP model. The slight reductions in accuracy and AUC are acceptable trade-offs for the significant privacy benefits provided by HE, making the PPDNN-CRP model a superior choice for credit risk applications requiring both predictive accuracy and data privacy. PPDNN-CRP with Neural Net work processing algorithms optimized for encrypted computations. This can involve designing neural network operations (like matrix multiplications and activations) that are more efficient under encryption. Hence Proposed PPNNP-CRP model can handle the potential impact on predictive accuracy due to its Homomorphic encryption used as privacy-preserving mechanisms.

Conclusions

In this paper we introduced the PPDNN-CRP framework, a Privacy-Preserving Deep Neural Network designed for secure credit risk prediction using homomorphic encryption. The proposed model demonstrates competitive performance with an accuracy of 80.48%, closely approximating the accuracy of a traditional deep neural network (86.18%) and significantly outperforming a privacy-preserving logistic regression method (77.23%). Additionally, the PPDNN-CRP model maintains robust predictive capabilities, as evidenced by a precision of 0.84, recall of 0.91, and an F1-score of 0.87. Our security analysis confirms that the PPDNN-CRP framework effectively mitigates a range of privacy threats, including poisoning, evasion, membership inference, model inversion, and model extraction. Although there is a slight accuracy trade-off compared to the non-private model, the robust privacy protections provided by homomorphic encryption justify this compromise. Epoch-wise analysis indicates that the model's training and inference times remain efficient and comparable to non-private approaches after the initial setup phase. In conclusion, the PPDNN-CRP framework offers a promising solution for high-quality credit risk prediction while ensuring data privacy and compliance with legal and ethical standards. This work significantly advances the potential for secure AI-driven financial services, and future research opportunities should focus on further optimizations and extensions to additional privacy-sensitive domains.

Acknowledgements
I am very much grateful to my Father V. Bala Suryanarayana for supporting and encouraging me throughout.

Authors' contributions
First author: the main idea, conceptualization, methodology, results and overall documentation. Second author: Data collection, literature review , Implementation, results and overall documentation.

Funding
No funding was received.

Data availability
No datasets were generated or analysed during the current study.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
All the author has given consent for publication.

Competing interests
The authors declare no competing interests.

Received: 29 May 2024 Accepted: 1 October 2024
Published online: 15 October 2024

References

1. Gupta A, Pant V, Kumar S, Bansal P (2020) Bank loan prediction system using machine learning. 2020 9th International Conference System Modeling and Advancement in Research Trends (SMART). p 423-426
2. Golak, Bihari R, Debasish D, Biswaranjan A (2021) Modern approach for loan sanctioning in banks using machine learning. https://doi.org/10.1007/978-981-15-5243-4_15
3. Mayank A, kulandai A, Pawan V (2022) Prediction of loan behaviour with machine learning models for secure banking. J Comput Sci Eng. https://doi.org/10.36596/jcse.v3i1.237
4. Shoumo SZH, Dhruba MIM, Hossain S, Ghani NH, Arif H, Islam S (2019) Application of machine learning in credit risk assessment: a prelude to smart banking. https://doi.org/10.1109/TENCON.2019.8929527
5. Ahamed KU, Islam M, Uddin A, Akhter A, Paul BK, Yousuf MA, Moni MA (2021) A deep learning approach using effective preprocessing techniques to detect COVID-19 from chest CT-scan and X-ray images. Comput Biol Med 139:105014. https://doi.org/10.1016/j.compbiomed.2021.105014
6. Bhargav P, Sashirekha K (2023) A machine learning method for predicting loan approval by comparing the random forest and decision tree algorithms. J Surv Fish Sci 10(1S):1803–1813. Vol. 10 No. 1S (2023): Special Issue 1
7. Dansana D, Patro SGK, Mishra BK, Prasad V, Razak AK, Wodajo AW (2023) Analyzing the impact of loan features on bank loan prediction using random forest algorithm. Eng Rep. https://doi.org/10.1002/eng2.12707
8. Blessie EC, Rekha R (2019) Exploring the machine learning algorithm for prediction the loan sanctioning process. Int J Innovative Technol Exploring Eng (IJITEE) 9(1):2714–2719. https://doi.org/10.35940/ijitee.A4881.119119
9. Zhu L, Wang Z, Wang L, Xie L, Li J, Cao X (2019) ZnSe embedded in N-doped carbon nanocubes as anode materials for high-performance Li-ion batteries. Chem Eng J 364:503–513. https://doi.org/10.1016/j.cej.2019.01.191
10. Alsaleem MYA, Hasoon S (2020) Predicting bank loan risks using machine learning algorithms. Al-Rafidain J Comput Sci Math. https://doi.org/10.33899/CSMJ.2020.164686
11. Wang D, Wu Q, Zhang W (2019) Neural learning of online consumer credit risk. arXiv: Risk management. https://ssrn.com/abstract=3398981
12. Uddin N, Ahamed MKU, Uddin MA, Islam MM, Talukder MA, Aryal S (2023) An ensemble machine learning based bank loan approval predictions system with a smart application. Int J Cogn Comput Eng 4:327–339. https://doi.org/10.1016/j.ijcce.2023.09.001
13. Lu Z, Asghar HJ, Kaafar MA, Webb D, Dickinson P (2022) A differentially private framework for deep learning with convexified loss functions. IEEE Trans Inf Forensics Secur 17:2151–2165. https://doi.org/10.1109/tifs.2022.3169911
14. Ma C, Li J, Ding M, Liu B, Wei K, Weng J, Poor HV (2023) RDP-GAN: a Rényi-differential privacy based generative adversarial network. IEEE Trans Dependable Secur Comput. https://doi.org/10.1109/tdsc.2022.3233580
15. Gratton C, Venkategowda NKD, Arablouei R, Werner S (2022) Privacy-preserved distributed learning with zeroth-order optimization. IEEE Trans Inform Forensics Secur. https://doi.org/10.1109/tifs.2021.3139267
16. Toubeau JF, Teng F, Morstyn T, Von Krannichfeldt L, Wang Y (2022) Privacy-preserving probabilistic voltage forecasting in local energy communities. IEEE Trans Smart Grid 14(1):798–809. https://doi.org/10.1109/tsg.2022.3187557
17. Lakhan A, Mohammed MA, Nedoma J, Martinek R, Tiwari P, Vidyarthi A et al (2022) Federated-learning based privacy preservation and fraud-enabled blockchain IoMT system for healthcare. IEEE J Biomed Health Inform 27(2):664–672. https://doi.org/10.1109/jbhi.2022.3165945
18. Xu R, Chen Y (2022) µDFL: a secure microchained decentralized federated learning fabric atop IoT networks. IEEE Trans Netw Serv Manage 19(3):2677–2688. https://doi.org/10.1109/tnsm.2022.3179892
19. Xin W, Hideaki I, Linkang D, Peng C, Jiming C (2020) Privacy-preserving distributed machine learning via local randomization and ADMM perturbation. IEEE Trans Signal Process. https://doi.org/10.1109/TSP.2020.3009007
20. Zheng H, Ye Q, Hu H, Fang C, Shi J (2020) Protecting decision boundary of machine learning model with differentially private perturbation. IEEE Trans Dependable Secur Comput 19(3):2007–2022. https://doi.org/10.1109/tdsc.2020.30433823
21. Ximeng L, Lehui X, Yaopeng W, Jian Z, Jinbo X, Zuobin Y, Athanasios V, Vasilakos (2021) Privacy and security issues in Deep Learning: a Survey. IEEE Access. https://doi.org/10.1109/ACCESS.2020.3045078
22. Huang QX, Yap WL, Chiu MY, Sun HM (2022) Privacy-preserving deep learning with learnable image encryption on medical images. IEEE Access 10:66345–66355. https://doi.org/10.1109/access.2022.3185206
23. Chandramohan D, Vengattaraman T, Ramachandran DR, Dhavachelvan BP (2013). A privacy preserving representation for web service communicators' in the Cloud. https://doi.org/10.1007/978-3-642-37949-9_44
24. Priyansi Paroda C, Pradhan JA, Alzubi A, Javadpour M, Liu Y, Lee C-C (2023) Elliptic curve cryptographic image encryption using Henon Map and Hopfield Chaotic Neural Network. Multimedia Tools Appl. https://doi.org/10.1007/s11042-023-14607-7
25. Movassagh AA, Alzubi JA, Gheisari M, Rahimi M, Mohan SK, Abbasi AA, Nabipour N. Artificial neural networks training algorithm integrating invasive weed optimization with diferential evolutionary model. J Ambient Intell Humaniz Comput. https://doi.org/10.1007/s12652-020-02623-6
26. Pulido-Gaytan B, Tchernykh A, Cortés-Mendoza JM, Babenko M, Radchenko G, Avetisyan A, Drozdov AY (2021) Privacy-preserving neural networks with homomorphic encryption: C hallenges and opportunities. Peer-to-Peer Netw Appl 14(3):1666–1691. https://doi.org/10.1007/s12083-021-01076-8
27. Gheisari M, Najafabadi HE, Alzubi JA, Gao J, Wang G, Abbasi AA, Castiglione A (2021) OBPP: an ontology-based framework for privacy-preserving in IoT-based smart city future generation computer systems. https://doi.org/10.1016/j.future.2021.01.028
28. Stephanie V, Khalil I, Rahman MS, Atiquzzaman M (2022) Privacy-preserving ensemble infused enhanced deep neural network framework for edge cloud convergence. IEEE Int Things J 10(5):3763–3773. https://doi.org/10.1109/jiot.2022.3151982
29. Wang L, Aono Y, Hayashi T, Phong LT, Wang L (2016) Privacy-preserving logistic regression with distributed data sources via homomorphic encryption. IEICE Trans Inform Systems. https://doi.org/10.1587/TRANSINF.2015INP0020
30. Chaudhuri K, Monteleoni C (2008) Privacy-preserving logistic regression. Advances in neural information processing systems, 21. ISBN: 9781605609492
31. Yuli Z, Zhenyu W, Ye Y, Tianlong C, Zhangyang W (2020) PCAL: a privacy-preserving intelligent credit risk modeling framework based on adversarial learning. arXiv: Crypt Secur. https://doi.org/10.48550/arXiv.2010.02529
32. Kyoohyung han, jinhyuck jeong, jung hoon sohn, and yongha son "Efficient privacy preserving logistic regression inference and training" Cryptology ePrint Archive. 2020. https://ia.cr/2020/1396
33. Divakar Allavarpu VVL, Naresh VS, Krishna Mohan A (2023) Privacy-preserving credit risk analysis based on homomorphic encryption aware logistic regression in the cloud
34. T Sridokmai, S Prakancharoen. The homomorphic other property of Paillier cryptosystem," 2015 International Conference on Science and Technology (TICST), Pathum Thani, Thailand, 2015, pp. 356–359. https://doi.org/10.1109/TICST.2015.7369385
35. Dataset avalible at: https://www.kaggle.com/datasets/burak3ergun/loan-data-set

Publisher's Note