Baghdad Science Journal

Volume 22 | Issue 9 Article 25

9-23-2025

# Towards Efficient and Privacy Preserving ECG Classification: Federated Transfer Learning Enhanced by CKKS-Based Homomorphic Encryption

Anmar A. Al-Janabi
*Collage of Computer Science, University of Technology - Iraq, Baghdad, Iraq*, anmar.a.aljanabi@uotechnology.edu.iq

Sufyan Al-Janabi
*Department of Computer Science, College of Computer Science and Information Technology, University of Anbar, Ramadi, Anbar, Iraq*, sufyan.aljanabi@uoanbar.edu.iq

Belal Al-Khateeb
*Department of Computer Science, College of Computer Science and Information Technology, University of Anbar, Ramadi, Anbar, Iraq*, belal-alkhateeb@uoanbar.edu.iq

## How to Cite this Article

Al-Janabi, Anmar A.; Al-Janabi, Sufyan; and Al-Khateeb, Belal (2025) "Towards Efficient and Privacy Preserving ECG Classification: Federated Transfer Learning Enhanced by CKKS-Based Homomorphic Encryption," *Baghdad Science Journal*: Vol. 22: Iss. 9, Article 25.
DOI: https://doi.org/10.21123/2411-7986.5066

BAGHDAD SCIENCE JOURNAL 2025;22(9):3094–3113

RESEARCH ARTICLE

# Towards Efficient and Privacy Preserving ECG Classification: Federated Transfer Learning Enhanced by CKKS-Based Homomorphic Encryption

**Anmar A. Al-Janabi [1,*], Sufyan Al-Janabi [2], Belal Al-Khateeb [2]**

[1] Collage of Computer Science, University of Technology - Iraq, Baghdad, Iraq
[2] Department of Computer Science, College of Computer Science and Information Technology, University of Anbar, Ramadi, Anbar, Iraq

## ABSTRACT

In healthcare, maintaining both the accuracy and privacy of medical diagnoses collaboratively is a significant challenge. To the best of our knowledge, this research proposes the first end-to-end Privacy-Preserving Federated Transfer Learning (PPFTL) framework for 2-D ECG arrhythmia classification. Incorporating Transfer Learning (TL) narrows the gap between the encrypted and non-encrypted framework versions in the training step. It involves the transformation of raw Electrocardiogram (ECG) signals into 2-D ECG grayscale images. The dataset is disseminated after transformation to images and then fed as input into the local models, where MobileNetV2 serves as a feature extractor. The training process for each client incorporates data balance and augmentation techniques to improve the model's performance. Deep Learning (DL) models are subject to various privacy attacks to gain sensitive data. As a result, the Homomorphic Encryption Cheon-Kim-Kim-Song (HE-CKKS) scheme encrypts only model parameters to protect deep models from adversary attacks, preventing the sharing of sensitive raw data. Experimental results on the MIT-BIH Arrhythmia dataset achieved 88.12% accuracy. Incorporating HE-CKKS increased computation times by 1.08%, 1.27%, and 1.43% for 2, 3, and 4 clients, respectively.

**Keywords:** Data privacy, Electrocardiogram, Federated learning, Homomorphic encryption, Transfer learning

## Introduction

According to estimates from the World Health Organization (WHO), Cardiovascular Diseases (CVDs) are currently recognized as the main cause of death worldwide, resulting in around 17.9 million deaths annually.[1,2] Several risk factors, such as smoking, unhealthy eating habits, and excessive alcohol consumption, have been shown to significantly elevate the probability of experiencing adverse cardiac conditions, such as heart attacks and heart failure.[3]

In general, doctors classify various shapes of waveforms in an electrical signal to recognize cardiac abnormalities. The development of Machine Learning (ML) technology within Artificial Intelligence (AI) and its employment within the medical domain offered significant assistance in disease diagnosis through the building of classification models.[4]

However, utilizing distributed ML over a large amount of scattered medical data poses serious challenges. A key challenge is data privacy. Working with medical records containing sensitive patients' information normally raises security and privacy concerns.[5,6] For example, the confidential information about an individual's health can be used for identification purposes. Hence, the privacy and security

Received 9 January 2024; revised 3 June 2024; accepted 5 June 2024.
Available online 23 September 2025

* Corresponding author.
E-mail addresses: anmar.a.aljanabi@uotechnology.edu.iq (A. A. Al-Janabi), sufyan.aljanabi@uoanbar.edu.iq (S. Al-Janabi), belal-alkhateeb@uoanbar.edu.iq (B. Al-Khateeb).

*https://doi.org/10.21123/2411-7986.5066*

of such data have the highest priority, which must be maintained. In order to preserve data privacy and limit the availability of data, laws and regulations have been adopted. The European Union General Data Protection Regulation (GDPR) regulatory framework has implemented a rigorous standard for protecting and preserving the privacy of sensitive data.[7]

Securing private data can be achieved through various approaches. An effective one is to employ Federated Learning (FL), where multiple entities utilize collaborative computations in a distributed environment to protect sensitive data. The aforementioned approach eliminates the need for raw data centralization and vulnerability, which enhances medical data privacy and security. A promising approach that has significant possibilities in the healthcare industry as a subcategory within ML is known as Transfer Learning (TL). TL utilizes previously gained knowledge from a particular task in order to enhance or improve the performance of another one. Hence, utilizing previously trained models to extract features improves the efficiency and accuracy of other different tasks.[8,9]

MobileNetV2 specifically designed for mobile and edge devices concentrating on computing efficiency. It achieves a compact model size while maintaining high accuracy by employing depthwise separable convolutions, effectively decreasing the parameter count and computational cost. As a result, it works perfectly for real-time applications and devices with limited working power, like smartphones, IoT devices, and embedded systems. MobileNetV2 has achieved extensive implementation in TL scenarios where lightweight and practical models are required for object detection, classification, and segmentation tasks. It established new standards for building Deep Learning (DL) models that are both powerful and efficient while also being conscious of the available resources.[10]

In order for the two-dimensional CNN to function properly, its input data must be images. Therefore, ECG signals are converted into 2-D images by plotting the ECG beats as grayscale images. Converting the signals into a visual representation enables the Convolutional Neural Network (CNN) model to extract complex patterns and features that might not be recognized in their original form. Signals are segmented into separate heartbeats, utilizing the detailed annotations associated with the records and generating two-dimensional grayscale standardized images. This procedure enables CNN to classify and analyze cardiac arrhythmias.

This research employs the MobileNetV2 architecture as a previously trained CNN model. The deep model analyzes ECG signals after being transformed into 2-D grayscale images for automating and assisting in diagnosing cardiac arrhythmia.

The transmission of model parameters or gradients within the FL approach is more effective and preferable than transmitting raw data. Certain data owners, such as hospitals, have rigorous privacy standards that add additional restrictions to the disclosure of any information beyond the final outcomes, including intermediate model weights that can be exploited to retrieve some of the training data.[11] A possible solution for the protection against adversarial collaborators can be achieved using the Homomorphic Encryption (HE) mechanism. HE is a cryptographic technology that enables data encryption while allowing operations to be done on the encrypted data.[12] This approach has the potential to protect the ML model from adversarial attacks. This research introduces an innovative paradigm for ECG data analysis by integrating TL for feature extraction, FL for collaborative learning without direct data exchange, and HE for secure aggregation of model updates. The combined approaches enhance efficiency, accuracy, and security in privacy-preserving ECG data analysis. This paper's primary contributions are as follows:

- To the best of our knowledge, this research proposes the first end-to-end Privacy-Preserving Federated Transfer Learning (PPFTL) framework for 2-D ECG arrhythmia classification.
- Develop privacy-preserving healthcare analytics, demonstrating that sensitive information related to ECG analysis can be safely shared among multiple participants without compromising the privacy or security of the underlying patient data.
- Highlights the potential for hospitals and other healthcare agencies to engage in collaborative learning using FTL by implementing a shared model that preserves data privacy.
- This work empirically validates the proposed system using a real-world ECG dataset.

## Related work

The use of data-driven ML models in the healthcare sector offers significant advantages, particularly when paired with private medical information. In order to protect sensitive health-related data, these models are trained on-premises. However, it can be challenging to construct reliable models without large, diversified datasets covering a wide range of health conditions. Previous studies have suggested using FL techniques as a potential solution to address the problem. For instance, research studies[13–15] reviewed the potential applications of FL within the digital health domain. It highlights the challenges and considerations that have to be tackled with FL in order to be successfully implemented and protect fragmented and private biomedical data. The influence of FL is investigated by several stakeholders, including patients, physicians, healthcare institutions, and manufacturers.

The study in[16] proposed a PPFL approach using the BraTS 2018 dataset for brain tumour segmentation. The proposed FL system is built on a client-server architecture using the federated averaging technique. The server manages the global Deep Neural Network (DNN) model and coordinates clients' local Stochastic Gradient Descent (SGD) updates. The Differential Privacy (DP) method is also utilized to ensure patient data privacy. Experimental results of the proposed approach assure data privacy protection with high accuracy. The study did not examine how privacy-preserving approaches impact brain tumour segmentation model accuracy and performance. Furthermore, the scalability of the method for complex medical imaging purposes remains unexplored.

The authors in[17] provided an innovative approach to improving the security of e-Healthcare systems through secure Multiparty Computation (MPC) and the Paillier encryption scheme. The strategy safeguarded the privacy and confidentiality of sensitive patient data. IoT-enabled healthcare devices are addressed for delivering accurate medical data. Also, the suggested model has the potential to be applied to the E-auction and E-voting models as well. The study needs more information regarding the scalability of the suggested approach, especially in the context of a large number of patients.

The studies presented by Wibawa et al.[6,18] presented a PPFL framework for medical data. They employed HE to protect sensitive data against privacy attacks, including those from collaborator adversarial attackers. In addition, to further protect the model from adversaries, a secure MPC protocol is being used. A real-world medical dataset of COVID-19 radiography images with two classifications, which are COVID and Normal, is utilized to evaluate the performance of the proposed method in terms of model accuracy. The developed framework has an accuracy of over 80% in both encrypted and plain data, demonstrating the framework's ability to maintain performance and preserve data privacy. However, processing time substantially increases because of HE employment, limiting its practicality for real-world applications.

Researchers in[19] suggested an innovative and efficient method utilizing Fully Homomorphic Encryption (FHE) in cloud computing. Their strategy employed a twin-key encryption technique, as well as a fragmentation technique of magic numbers to process encrypted data securely. The practicality of the suggested method is illustrated via cognitive applications regarding smart cities, and its effectiveness is evaluated through cryptanalytic attacks. The system demonstrated a high level of resistance against brute-force attacks. However, it lacks explicit information concerning the dataset used in the research.

The study conducted by[20] designed a deep, one-dimensional CNN-based model to classify heartbeats. Based on the standard specified by AAMI EC57, the model was capable of classifying five different types of arrhythmias. Furthermore, the authors were able to translate the knowledge obtained from this task to the Myocardial Infarction (MI) classification challenge. Results show that the proposed approach achieves average prediction accuracies of 93.4% for arrhythmia classification and 95.9% for MI classification.

Gao et al.[21] developed a new technique called Heterogeneous Federated Transfer Learning (HFTL), in which FL utilizes TL to deal with different feature spaces. They developed a privacy-preserving transfer learning method to remove the covariate shift of homogeneous feature spaces and bridge heterogeneous feature spaces of various data owners. An end-to-end secure multi-party learning protocol with two variations based on HE and Secret Sharing (SS) approaches shows that the HFTL is secure, efficient, and highly scalable on five benchmark datasets. However, it is crucial to perform a comprehensive analysis to evaluate the scalability of the proposed approach in cases that involve a larger number of clients.

In order to enhance statistical modeling within a data federation, the study[22] presented a new technique and framework known as federated transfer learning. The paper also offers some novel approaches for Two-Party Computation (2PC) with Neural Network (NN) under the FTL framework, integrating additively HE and SS using beaver triples so that the accuracy is almost lossless and only minimal modifications of the NN are required. Experiments were conducted on publicly available datasets, including the NUS-WIDE dataset and Default-of-Credit-Card-Clients dataset, to validate the suggested technique. The study results show that the suggested FTL framework achieves similar or superior performance compared to existing methods while also preserving data privacy.

Singh et al.[23] introduced a novel architecture that combines FL and blockchain to improve privacy in IoT healthcare systems for smart cities. The aim is to protect data through a distributed method, employing blockchain for secure data exchange and FL for maintaining local data privacy. It features a sensor network for data collection from IoT devices, a blockchain cloud network for data validation and processing, and the distribution of processed data to various healthcare and monitoring devices.[24] Empirical results indicated increasing network overheads

**Table 1.** Related works strengths and weaknesses.

| Reference | Strength(s) | Weakness(es) |
|---|---|---|
| Li, Wenqi [16] | • Similar segmentation performance compared to a central server without sharing data.<br>• Improved convergence speed<br>• Sharing larger proportions of the model achieves better performance. | • The researchers did not investigate the scalability for a larger dataset or other intricate medical tasks.<br>• A thorough examination of communication and computational costs would be beneficial for practical, real-world deployment. |
| Vijaya Kumar A [17] | • The suggested framework can be expanded to include diagnostic centers for secure e-medical advice system. | • Operations are performed over encrypted data which makes it relatively slow and hence impacts the system's efficiency. |
| Wibawa et al. [18] | • The BFV crypto scheme does not degrade model performance. | • Single public key.<br>• Affected by the length of the encryption key. |
| Kara, Mostefa et al. [19] | • Offers efficient, secure, and private processing in cloud computing environments. | • The study does not investigate the limitations of the suggested FHE utilizing twin-key encryption and Magic Number Fragmentation. |
| Kachuee, Mohammad [20] | • Propose a deep CNN classifying 1-D ECG signals with high average accuracy. | • Further detailed evaluations of the suggested method over others' real-world scenarios are required. |
| Gao, Dashan [21] | • Handling Covariate Shift and Feature Heterogeneity. | • Due to DP usage, the proposed model is vulnerable to privacy leakage among participants during model training. |
| Liu, Yang et al. [22] | • Flexible and highly adaptable to various ML tasks. | • The utilized secret sharing approach has to generate and store many triplets before online computation. |
| Singh et al. [23] | • The proposed framework provides privacy-preserving, security, reliability, and scalability with a low overhead. | • The work relies only on the performance of the Blockchain-FL. |
| Walskaar et al. [25] | • Robust model performance. | • Increased execution times per round.<br>• Higher memory usage per IoT device and server. |

proportional to service allocation probability, with overheads rising from below 500ms to approximately 1500ms as the probability approaches 1.0 across various latency scenarios. Future improvements are anticipated in developing a blockchain-based trust model, a novel consensus mechanism for FL nodes, and improvements in latency, storage, and a federated reward system.

Walskaar et al. [25] published a study which enhanced the research conducted by Wibawa et al. [6] by integrating an enhanced multi-key homomorphic encryption (xMK-CKKS) approach into the FL framework. The authors used the xMK-CKKS scheme introduced by Ma et al. [26] to enhance the security of model updates in medical data applications. Specifically, they used a COVID-19 X-ray lung scan dataset from the study, [27] which consisted of COVID and non-COVID classes. Their modified Ring Learning with Error (RLWE) scheme and changes to the Flower FL framework endorsed enhanced client-server communications. The experimental results showed that the model maintained an accuracy of 89%–97% even when the number of clients varied from 2 to 5. Despite this, the performance with 10 clients revealed a wide variation in accuracy of 50%, indicating challenges with scalability.

Most previous works have focused on using simple model architectures in conjunction with privacy techniques, including DP, MPC, and HE, because of the noise generated from the computations within the convolutional layers. Due to the extensive multiplication operations within deep convolutional layers, noise increases exponentially in HE. However, our suggested framework addresses this issue using transfer learning technology, where the weights are kept frozen in the feature extraction section of the architecture since all the heavy computations are performed within. In addition, implementing transfer knowledge reduces the cost of computations over the distributed models since the training process does not have to begin from scratch. Integrating cutting-edge technologies, including FL, TL, and HE balances and model performance, noise management, and data privacy. Table 1 summarizes the related works, including key strengths and weaknesses.

## Preliminaries

The following subsections offer essential theoretical foundations for the current study. First, examine cryptographic algorithms like CKKS. Next, continue to review the regulations concerning the privacy of healthcare data. Finally, delve into the research of Federated and Transfer Learning technologies.

### Homomorphic encryption

Encrypting data is usually used for securing data at rest or while in transmission, both in enterprise and personal settings. However, this traditional method leaves security vulnerability during computing processes, especially in highly sensitive fields like healthcare and personal information management.

As a response to this challenge, HE has emerged, enabling mathematical operations to be performed directly over encrypted data without the need for decryption.[28] When decrypted, the outcomes stay encrypted as well as produce identical or almost identical results. HE complies with the stringent privacy requirements of the modern, digitally interconnected world by enabling secure data processing without disclosing actual data.

Denoting encryption as *Enc*, decryption as *Dec*, $\odot$ represents homomorphic addition or multiplication operations over ciphertext, and $f$ as a function applied to actual plaintext values x and y using encryption key $pk$, then the property of HE can be presented by Eq. (1).

$$f\ (x,\ y)\ =\ Dec\left(Enc\left(pk,\ x\right)\ \odot\ Enc\left(pk,\ y\right)\right) \qquad (1)$$

The adoption of HE enables privacy-preserving in outsourcing storage and computation. It enables the encryption and outsourcing of data processing to commercial cloud environments while the data remains encrypted.

HE can be classified into partial, somewhat, and fully homomorphic encryptions based on the operations they facilitate.[29,30] Partial Homomorphic Encryption (PHE) supports one type of mathematical addition or multiplication operation infinitely. In contrast, Somewhat Homomorphic Encryption (SHE) supports mathematical addition and multiplication operations, increasing the noise. Additionally, it is bound by the number of operations, and decryption fails when it overpasses a predefined threshold. Last but not least, FHE supports both addition and multiplication arbitrarily for an unlimited number of operations.[31] The bootstrapping strategy is utilized to decrease noise and enable continued accurate decryption. This work employs SHE, which enables both multiplication and addition on encrypted data, both of which are required for the secure aggregation of DL model weights.

### Cheon-Kim-Kim-Song (CKKS) scheme

Cheon et al.[32] proposed the CKKS scheme, an approximative homomorphic encryption scheme with a tunable level of approximation error for secure data computation over real or complex numbers directly. It operates over polynomials in a ring, and its security is primarily based on the Ring Learning with Errors (RLWE) problem. CKKS is well-suited for privacy-preserving computations in ML/DL and data analytics, for which approximate arithmetic tends to be sufficient. The following paragraph describes the scheme under consideration in brief.

Let n, q, $\Delta$ be initialized parameters, where n is a ring dimension, q is a large prime coefficient modulus, and $\Delta$ is a scaling factor. The secret key sk is sampled from $R_2$, and the public key $pk = (pk_1,\ pk_2)$ is derived. A vector of complex numbers $\vec{z} \in \mathbb{C}^{\frac{n}{2}}$, and $\Delta\ > 1$ being encoded into a single object $a$ in the plaintext domain represented by Eq. (2).

$$Encode\left(\vec{z},\ \Delta\right) = \left\lfloor \Delta.\pi^{-1}\left(\vec{z}\right) \right\rceil \qquad (2)$$

Encryption yields $C = (C_1, C_2)$ via pk and error distribution $\chi$. Homomorphic evaluation, denoted as $\mathrm{EvalAdd}(C^{(1)}, C^{(2)})$ produces$(C_1^{(3)}, C_2^{(3)})$, adds polynomial components, while multiplication EvalMult $(C^{(1)}, C^{(2)})$ produces $C^{(3)} = (C_1^{(3)}, C_2^{(3)}, C_3^{(3)})$, which undergoes relinearization to reduce dimensionality represented by Eqs. (3) and (4), respectively.

$$EvalAdd\left(C^{(1)},\ C^{(2)}\right) = \left(C_1^{(3)},\ C_2^{(3)}\right) \qquad (3)$$

$$EvalMult\ \left(C^{(1)},\ C^{(2)}\right) = \left(C_1^{(3)},\ C_2^{(3)},\ C_3^{(3)}\right) \qquad (4)$$

Following, Rescaling procedure after a multiplication operation is performed to manage noise so that the output ciphertext closely resembles the input, Eq. (5) denotes rescaling.

$$Rescale\left(C,\ \Delta\right) = \frac{1}{\Delta}.\ \left[C_1,\ C_2\right]_q \qquad (5)$$

Employing sk on ciphertext to decrypt and obtain an encoded approximate plaintext is represented by Eq. (6).

$$\overset{\prime}{m} = \left[C_1 +\ C_2.sk\right]_q \qquad (6)$$

Decoding is the inverse of the encoding process which returns a vector of complex numbers as denoted in Eq. (7).

$$Decode\left(a,\ \Delta\right) =\ \pi\left(\frac{1}{\Delta}.\ a\right) \qquad (7)$$

The CKKS scheme is significantly similar to other RLWE-based schemes like BGV; however, a study presented by Li and Micciancio[33] claims that a passive adversary with the following capabilities can launch an efficient passive attack:

- The attacker can choose and encrypt any number of messages to generate their corresponding ciphertext messages.
- The attacker acts as a server providing outsourced computations, choosing a function to be evaluated homomorphically.
- Based on the application context, the attacker could choose some ciphertext messages and ask for decryption.

The attack model with the abovementioned properties is known as INDistinguishability under Chosen Plaintext Attack (IND-CPA). They claim the vanilla CKKS scheme is less secure than BFV and BGV. The attack uses the decryption function and the approximate decryption result to infer information about the RLWE error. In the best-case scenario, the attacker potentially recovers the secret key with a simple algebraic manipulation in one attempt, utilizing mathematical approaches such as the Extended Euclidean algorithm and Bezout's identity for computational efficiency.

The authors also recommended modifying the CKKS scheme by adding extra noise to the decryption results, aiming to conceal the RLWE noise and mitigate attacks. However, if the decrypted results are not shared with any untrusted participants, the vanilla CKKS scheme in this case is secure enough. In our framework context, once clients receive the encrypted parameters from the central server, they decrypt them locally and never share the decrypted results with any other parties, making them suitable and secure enough.

Additional security measures to address CKKS crypto scheme vulnerabilities include secure selection and periodically updating encryption parameters. Enhancing key management is critical through the implementation of secure key storage, such as hardware security modules, to prevent unauthorized access. Additionally, data privacy can be enhanced by integrating cutting-edge technologies from DP or MPC.

### Privacy and trustworthiness in healthcare sector

In today's healthcare systems, trust and privacy are essential aspects of the digital ecosystem that cannot be compromised. Trust between stakeholders, such as patients, healthcare professionals, health regulatory authorities, and technological solutions providers, is complicated and multifaceted. Due to the sensitivity of medical data, it is crucial to protect the following types of information:[18]

- **Identifiable personal information:** Consisting of a patient's name, address, social security number, date of birth, and other financial information such as bank account numbers.
- **Health status metrics:** Detailed data regarding a patient's medical and psychological conditions, effectively serving as a snapshot of their current state of health.
- **Healthcare services data:** This includes medical and psychiatric consultations, prescribed medications, equipment used, and surgical or therapeutic treatments.
- **Institutional and provider identifiers:** Information on healthcare facilities or professionals who offered medical and psychological treatment.

The European Union's General Data Protection Regulation (GDPR) remains a prominent legislative instrument that governs data privacy. It significantly impacts public and private sector healthcare data privacy regulations by emphasizing individuals' right to control how their private data is used. Healthcare systems can achieve unprecedented data privacy and enable secure sharing and analysis by integrating FTL with CKKS-based homomorphic encryption. This integrated strategy promotes healthcare systems toward a more secure and patient-centric model while simultaneously complying with GDPR standards.[34]

### Federated learning

FL is a decentralized ML framework that allows stakeholders, such as healthcare and other data owners, to collaboratively build a shared model while storing their data locally. Since only the model's parameters or gradients are transmitted in this method, it is particularly attractive to entities with stringent privacy policies. Participants can train a common model on their local datasets in FL, even though they may have distinct feature spaces and include very different distributions of the shared features. The trained models are either aggregated on a central server or coordinated across participants to create a global model. FL can be implemented on various technical platforms and in applications in centralized, decentralized, or heterogeneous environments. The localized nature of the data and the collaborative method for training make FL an effective strategy for the privacy and security of data, particularly in fields where data heterogeneity and privacy are essential, such as healthcare.[18]

### Transfer learning

The target domain of some real-world applications may suffer from insufficiency in data labeling or a lack of features for various reasons, such as

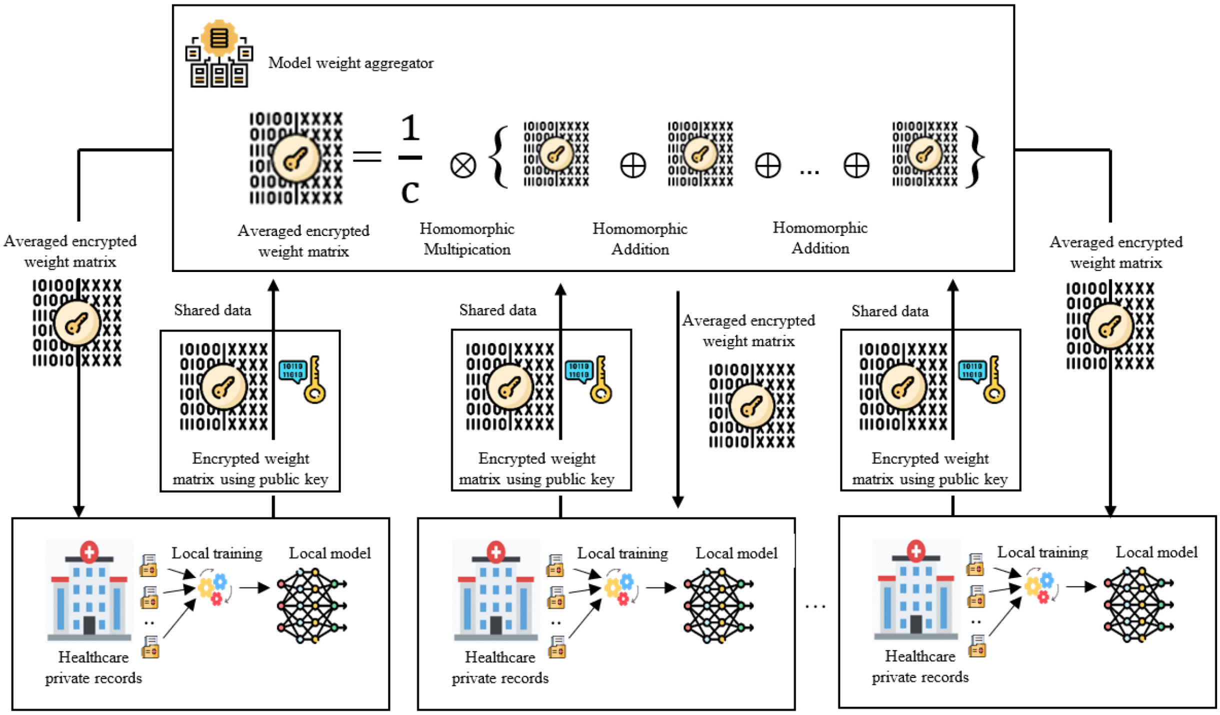
Model weight aggregator
$= \frac{1}{c} \otimes \{ \oplus \oplus \dots \oplus \}$
Averaged encrypted weight matrix
Homomorphic Multipication
Homomorphic Addition
Homomorphic Addition
Averaged encrypted weight matrix
Averaged encrypted weight matrix
Averaged encrypted weight matrix
Shared data
Shared data
Shared data
Encrypted weight matrix using public key
Encrypted weight matrix using public key
Encrypted weight matrix using public key
Local training
Local model
Healthcare private records
Local training
Local model
Healthcare private records
Local training
Local model
Healthcare private records
...

Fig. 1. System overview of the proposed PPFTL framework.[18]

domain-specific expertise, data privacy, and cost and time constraints, to name a few. The emergence of TL enables these target domains to take advantage of insights and patterns derived from more extensive source domains.[35,36] Automatic feature extraction can be implemented through a successful, previously trained deep CNN model. The feature maps generated from the intermediate convolutional layers during model training contain knowledge about the patterns from the source dataset. These extracted features surpass hand-crafted features and are effective for feature extraction.[37,38]

In the healthcare domain, the use of the TL strategy enables medical personals to fine-tune previously trained models on medical data, such as radiology scans, for particular tasks, such as ECG analysis for early detection, as well as make decisions about the best course of treatment. Despite the challenges of ECG signals, which may vary substantially between individuals, TL enhances the accuracy and efficiency of an ECG's cardiac interpretations. Hence, minimizing the need for extensive annotated ECG data labels and accelerating the advancement of robust, reliable, and accurate diagnostic tools within the healthcare industry.[39–41]

## System model

In order to fully understand the suggested procedure for analyzing 2-D ECG images' beats, a comprehensive understanding of the suggested Privacy Preserving Federated Transfer Learning (PPFTL) framework is introduced in this section. The CKKS encryption scheme was utilized in the suggested framework due to its ability to support direct operation on real numbers rather than approximation, leading to improved model accuracy. The framework comprises client-model and server-model stages. The first primary stage performs data training, while the second, represented by the server, performs a secure aggregation of the models' parameters or weights gathered from the various distributed clients. Each client locally trains a deep CNN model to extract the most critical features from its own locally stored data (beats' images) via a previously trained model through TL, thus maintaining the privacy of the data during the training phase. The encryption of the model weights is performed using the public key within the CKKS scheme. Subsequently, the server aggregates the encrypted weight matrices received from various clients. The server transmits the final aggregated weight matrix to the clients during the second step. Each client updates its model's weights by decrypting the aggregated encrypted weight matrix. After that, the aggregated model serves as an organizing structure for the final classification stage, providing a secure and private method for analysis. Fig. 1 and the following algorithms provide a complete system overview.

**Algorithm 1:** Client model training

```
Input: The dataset at client c: $\mathrm{D_c} = \{(\mathbf{x},\ \mathbf{y})|\mathbf{x} \in \mathbb{R}^{m},\ \mathbf{y} \in \mathbb{R}\}_{i=0}^{m}$; public key: $\mathrm{K_{pub}}$; $\mathrm{M_{global}}$: global_model;
    $m_c$ : Local_model; W: model weights.
Output: Ciphertext of a matrix W: [[W]].
Begin
Step 1: $\mathrm{X_{train}, X_{test}, y_{train}, y_{test}} \leftarrow$ split_local_dataset ($\mathrm{D_c}$)
Step 2: $\mathrm{m_c} \leftarrow \mathrm{M_{global}}$
Step 3: $\mathrm{m_c}$.compile(loss = “categorical_crossentropy″, optimizer = “Adam″)
Step 4: $\mathrm{m_c}$.fit $\leftarrow (\mathrm{X_{train}, y_{train}})$
Step 5: W $\leftarrow$ {}
Step 6: For each $l \in \mathrm{m_c}$ do
        Step 6.1: [[W]] $\leftarrow$ encrypt_fractional($l$.W, $\mathrm{K_{pub}}$)    // Encrypt the layer weights ($l$.W $\in \mathbb{R}^{m}$) with public key
Step 7: End For
Step 8: return [[W]]                                               // weights' matrix in encrypted form
End
```

**Algorithm 2:** Secure model aggregation

```
Input: No._of_clients: $c$; client_model_weights: $\mathcal{W} = \{[\![W]\!]_0, \ldots, [\![W]\!]_c\}$
Output: Encrypted aggregated weight matrix: $[\![W]\!]_{aggregated}$
Step 1: $[\![W]\!]_{aggregated} \leftarrow \{\}$
Step 2: For each $[\![W]\!]_i \in \mathcal{W}$ do
        Step 2.1: For each [[r]] $\in [\![W]\!]_i$ do
            Step 2.1.1: $[\![W]\!]_{aggregated} \leftarrow [\![W]\!]_{aggregated} \oplus$ [[r]]    // Additive homomorphic operation
        Step 2.2: End For
Step 3: End For
Step 4: For each [[r]] $\in [\![W]\!]_{aggregated}$ do
        Step 4.1: [[r]] $\leftarrow$ [[r]] $\otimes c^{-1}$                  // Multiplicative homomorphic operation
Step 5: End For
Step 6: Return $[\![W]\!]_{aggregated}$           // The aggregated matrix of weights in its encrypted form.
End
```

### Client initialization

The steps in Algorithm 1 demonstrate the entire sequence of steps occurring through the initialization stage. Each client receives the global model architecture with their weights and independently retrains the model's upper layers, utilizing their private set of data within batches of size 128 for 40 rounds. The generic feature extraction layers, whose weights are frozen, remain unchanged during this process. The categorical cross-entropy loss function and Adam optimizer were utilized to compile the model. Consequently, the trained model's weight matrix is encrypted and transmitted to the aggregator using the homomorphic CKKS scheme. Before the training step, private and public keys are generated through the PYthon For Homomorphic Encryption Libraries (PYFHEL) library for successful encryption. Once local model training ends, the algorithm iterates over each layer within the trained model to encrypt its parameters by employing the generated public key and then appending them to the encrypted weight matrix. This procedure continues until all weights are encrypted and formed properly before transmission.

### Model aggregation

Algorithm 2 describes the aggregation process in detail. First, the central server initializes a storage structure for the average encrypted weights. It generates an encrypted version of the denominator that represents the reciprocal of the total number of clients. Second, the aggregator receives all encrypted weight matrices, denoted as $\{[\![W]\!]_0, \ldots, [\![W]\!]_c\}$, and adds them homomorphically element-wise, where each weight's value participates equally in the final aggregated weight matrix. Finally, the central entity employs the FedAvg algorithm and securely computes the average value for each neuron in the output encrypted matrix in the encrypted domain.

### Client decryption

Clients can decrypt the weights after they have been successfully encrypted and securely aggregated.

**Algorithm 3:** Decryption process and local model update

```
Input: private key: $K_{priv}$; encrypted_aggregated_weights: $[\![W]\!]_{aggregated}$; global model: $M_{global}$
Output: Updated local model $m_c$
Begin
Step 1: $m_c \leftarrow \mathrm{M_{global}}$
Step 2: For each $l \in m_c$ do                                         // $l$ is a layer within the model.
        Step 2.1: $[\![r]\!] \leftarrow [\![W]\!]_{aggregated}\ (l)$      // Retrieve the row that corresponds to the given layer.
        Step 2.2: $l \leftarrow decrypt\_fractional\ ([\![r]\!], K_{priv})$   // Row decryption and layer weight updates
Step 3: End For
Step 4: $m_c.save\_model$                                  // Client save aggregated model as global model.
End
```

Each client accepts the updated global model weight's matrix and traverses through its layers to decrypt them with the private key previously generated utilizing the PYFHEL library. It is essential to know that PYFHEL loses reference to the encrypted version of the decrypted floating point weight values and lacks a direct link to the encrypted version; therefore, re-referencing is required. Because of the unidirectional nature of the decryption procedure, these newly decrypted weights are treated as a new entity. So, for the decryption to be completely successful, must first re-embed the decrypted weights into the model. The decryption process is a key step in updating their local models, as illustrated in Algorithm 3.

## Experimental evaluation

This section assesses the effectiveness of our method and discusses the experimental setup, including datasets, architecture, and performance metrics. After that, compare the outcomes and discuss privacy and computational overhead.

### Dataset and preprocessing

This study used the MIT-BIH Arrhythmia Database,[42] known for its high-quality, labeled ECG records from eight different types of arrhythmia (NOR, PVC, PAB, LBB, RBB, APC, VFW, and VEB). Raw ECG records were converted to 2-D grayscale images of size 96 × 96 for use as input to the MobileNetV2-based feature extractor. The choice of this image size is to maintain a balance between computational complexity and sufficient details. ECGs inherently hold fewer complex details than other medical images, hence reducing the need for higher resolution. FTL was employed for decentralized training across several clients, with data privacy guaranteed by the Homomorphic Encryption Cheon-Kim-Kim-Song (HE-CKKS). Concurrently, MobileNetV2 served as a feature extractor to extract significant features from the 2-D ECG images. From the original dataset, 107,620 samples were obtained in total, then split into 80% as a training set (86,092) and 20% as a testing set (21,528). Oversampling minority classes and under sampling dominant classes were both used to address class imbalance. Since CNN was used as a classifier, data augmentation techniques such as rotation, zoom, shifting, and flipping were employed in the training dataset, reducing overfitting and balancing the distribution between classes on the federated clients. This preprocessing method allowed us to effectively exploit the spatial features of ECG data in a federated and privacy-preserving manner, providing a solid foundation for subsequent analyses.

### Implementation and experimental setup

Table 2 provides an overview of the CNN architecture derived from MobileNetV2 and used in the ECG study.

The implementation was developed using Python 3.8.16 and utilized pre-existing third-party libraries. Standard libraries were utilized, including Keras and TensorFlow for ML. The NumPy library was also used for processing weight arrays and the structure of data. Furthermore, Pickle has been used for serializing exported weights. Most importantly, PYFHEL[43] is a Python wrapper for Microsoft SEAL,[44] was utilized for HE, and offers the same functionalities as SEAL. Microsoft published the SEAL library for HE in 2015. It employs both BFV[45] and CKKS[32] schemes. It offers SHE from key generation to evaluation, with homomorphic addition, multiplication, and relinearization.

This study uses the standard parameters for HE context generation to implement the CKKS scheme within the PYFHEL library. The security parameter n = 8192, scale factor = $2^{**30}$, and $q_i\_sizes =$ [60, 30, 30, 30, 60]. The dataset is randomly distributed across clients C ∈ (2, 3, and 4), then the

**Table 2.** Model summary.

| Layer (Type) | Output | Shape | No. of Parameters |
|---|---|---|---|
| input_2 | (InputLayer) | (None, 96, 96, 3) | 0 |
| Conv1 | (Conv2D) | (None, 48, 48, 32) | 864 |
| bn_Conv1 | (BatchNormalization) | (None, 48, 48, 32) | 128 |
| Conv1_relu | (ReLU) | (None, 48, 48, 32) | 0 |
| expanded_conv_depthwise | (DepthwiseConv2D) | (None, 48, 48, 32) | 288 |
| . . .. . . | | | |
| out_relu | (ReLU) | (None, 3, 3, 1280) | 0 |
| avg_pool | (GlobalAveragePooling2D) | (None, 1280) | 0 |
| flatten | (Flatten) | (None, 1280) | 0 |
| dropout_3 | (Dropout) | (None, 1280) | 0 |
| d1 | (Dense) | (None, 1024) | 1311744 |
| dropout_4 | (Dropout) | (None, 1024) | 0 |
| d2 | (Dense) | (None, 256) | 262400 |
| dropout_5 | (Dropout) | (None, 256) | 0 |
| classifier | (Dense) | (None, 8) | 2056 |

Total params: 3,834,184. Trainable params: 1,576,200. Non-trainable params: 2,257,984.

encrypted domain's predictive results regarding performance are compared to the plaintext domains.

### *Practical considerations for real-world implementation of PPFTL in healthcare sector*

Several aspects must be considered to implement and deploy the proposed PPFTL framework in a real-world healthcare environment, including infrastructure requirements, regulatory compliance, user adoption, security and privacy assurance, and conducting practical studies. These aspects are briefly explained in the following:[46]

- **Infrastructure requirements:** To deploy the PPFTL successfully, a robust infrastructure, including hardware and software to handle FL and HE computational demands, is necessary. Healthcare organizations require access to high-performance servers and clients' devices for efficient cryptographic computation. In addition, a reliable network is required to exchange the data required by FL-distributed models without exposing raw data.
- **Regulatory compliance:** Following healthcare regulations is essential for ensuring PPFTL deployment based on legal standards regarding patients' sensitive data. Compliance with GDPR and HIPAA is critical for the implementation, ensuring that the transmitted and processed data is encrypted and only minimal data is shared. Also, developing and maintaining an audit trail, which is a mechanism that is crucial to recording every access and process of a patient's data in addressing potential breaches.
- **User adoption:** several factors must be considered for the PPFTL to be adopted by healthcare professionals; these include training and support to educate healthcare staff with a comprehensive training course on the proposed system and continuously provide support to address any issues that arise while using. A user-friendly User Interface (UI) design for ease of use, especially for non-technical staff. A feedback mechanism with end-users will help to continuously optimize and update the PPFTL framework based on practical and real-world feedback from healthcare providers.
- **Security and privacy assurance:** Beyond regulatory compliance, transparency, including comprehensive documentation of how the data are being collected, stored, processed, and protected, should be available to the parties involved. Also, emphasizing privacy guarantees of the PPFTL by explaining how data exposure being minimized in contrast to centralize approaches.
- **Practical studies:** Before deploying and implementing the proposed framework into the real world, PPFTL must be tested in controlled healthcare configurations to monitor performance and gather real-time feedback. Additionally, evaluating the proposed framework scalability on these tests ensures it can handle additional loads generated by the increased number of participants.

## Results and discussion

At first, experiments were conducted using the MIT-BIH Arrhythmia dataset for ECG-based arrhythmia classification without the FTL and CKKS encryption scheme. Only one client participated in this baseline examination.

To understand the efficiency of the proposed framework in real-life environments, performance metrics

Table 3. Performance metrics for this non-federated, non-encrypted model.

| Preprocessing Techniques | | | | | |
|---|---|---|---|---|---|
| Balancing | Augmentation | Accuracy | Precision | Recall | F1-score |
| No | No | 0.923076928 | 0.829913127 | 0.853783358 | 0.832628812 |
| Yes | No | 0.820605695 | 0.617478568 | 0.872002949 | 0.677675383 |
| Yes | Yes | 0.73722595 | 0.573148724 | 0.838614315 | 0.618775105 |

were evaluated to assess the model's classification performance across various categories. Accuracy is a general indicator represented by Eq. (8), which measures the overall correctness of the model. It is calculated by the sum of the total true positives and true negatives divided by the total number of cases. Precision, denoted by Eq. (9), measures the ability of the model to classify positive values correctly and is calculated by dividing the true positives by the total number of predicted positive values. Recall, denoted by Eq. (10), which calculates the model's ability to detect positive values among the actual positives. It is the ratio of the true positives to the total values of actual positives. The F1-score, denoted by Eq. (11), balances precision and recall, where 1 reflects perfection while 0 is the worst.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{8}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{9}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{10}$$

$$\text{F1} - \text{score} = \frac{2* (\text{Precision} * \text{Recall})}{\text{Precision} + \text{Recall}} \tag{11}$$

Table 3 shows the performance metrics for this non-federated, non-encrypted model and the total execution time was 2347.022s. The model was compiled with Adam optimizer for the learning process, and ReLU was utilized as an activation function, while the last dense layer was a Softmax. Other hyperparameters include a minibatch size of 128, the number of epochs fixed at 40, and a dropout value of 0.5.

Table 3 shows that the highest accuracy has been achieved at 92.31% when no augmentation and balancing preprocessing techniques are performed, while the F1_score is approximately 83.26%, indicating that the model performs best with this original configuration without preprocessing procedures. Applying the balancing technique lowers the accuracy to 82.06% and precision to 61.75% while increasing recall to 67.77%, which helps the balanced model identify more positive cases across various classes. Furthermore, applying augmentation with data balancing reduces the model's accuracy to 73.72%. Similarly, the F1_score drops to 61.88%, reducing model performance. Fig. 2 shows the performance evaluation of the MobileNetV2 model of 8 classes for loss and accuracy of the analyzed model's training set under different configurations concerning data balancing and augmentation techniques.

Fig. 3 shows the confusion matrix for ECG arrhythmia classification conducted on a test set under various configurations for MobileNetV2 of 8 classes. Fig. 3 (a) displays the confusion matrix of the base model without implementing preprocessing techniques and has the highest overall performance metrics, as illustrated in Table 3. The model classifies most classes correctly due to the high values of true positives in the main diagonal, validating the high-performance metrics. However, some classes, like RBB and VFW, show a low recall value, raising concern. In Fig. 3 (b), there is a shift in the main diagonal values, which represent the correct predictions, being decreased, particularly for the NOR labels compared to their respective values of the confusion matrix in Fig. 3 (a). This indicates that the model decreases in its ability to correctly identify this class when the balancing technique is applied. The precision decremented value of performance metrics in Table 3 is evident in Fig. 3 (b), in which false positive values (off-diagonal values) are noticeable, particularly in the RBB class of rate 0.18. Implementing the augmentation procedure with data balancing, the model significantly degrades overall performance metrics, including accuracy, precision, and F1_score, as seen in Table 3. Regarding the confusion matrix, Fig. 3 (c) shows that the true values (main diagonal values) are high, referring to the classification of each class as good. The model is strong in identifying APC and VFW classes with a 0.98 rate value. Also, some class confusions are being noticed; specifically, the NOR is being misclassified as APC.

Subsequently, FTL was applied to the architecture, observing the model's performance based on evaluation matrices. Table 4 shows the performance results of the FTL approach without encryption.

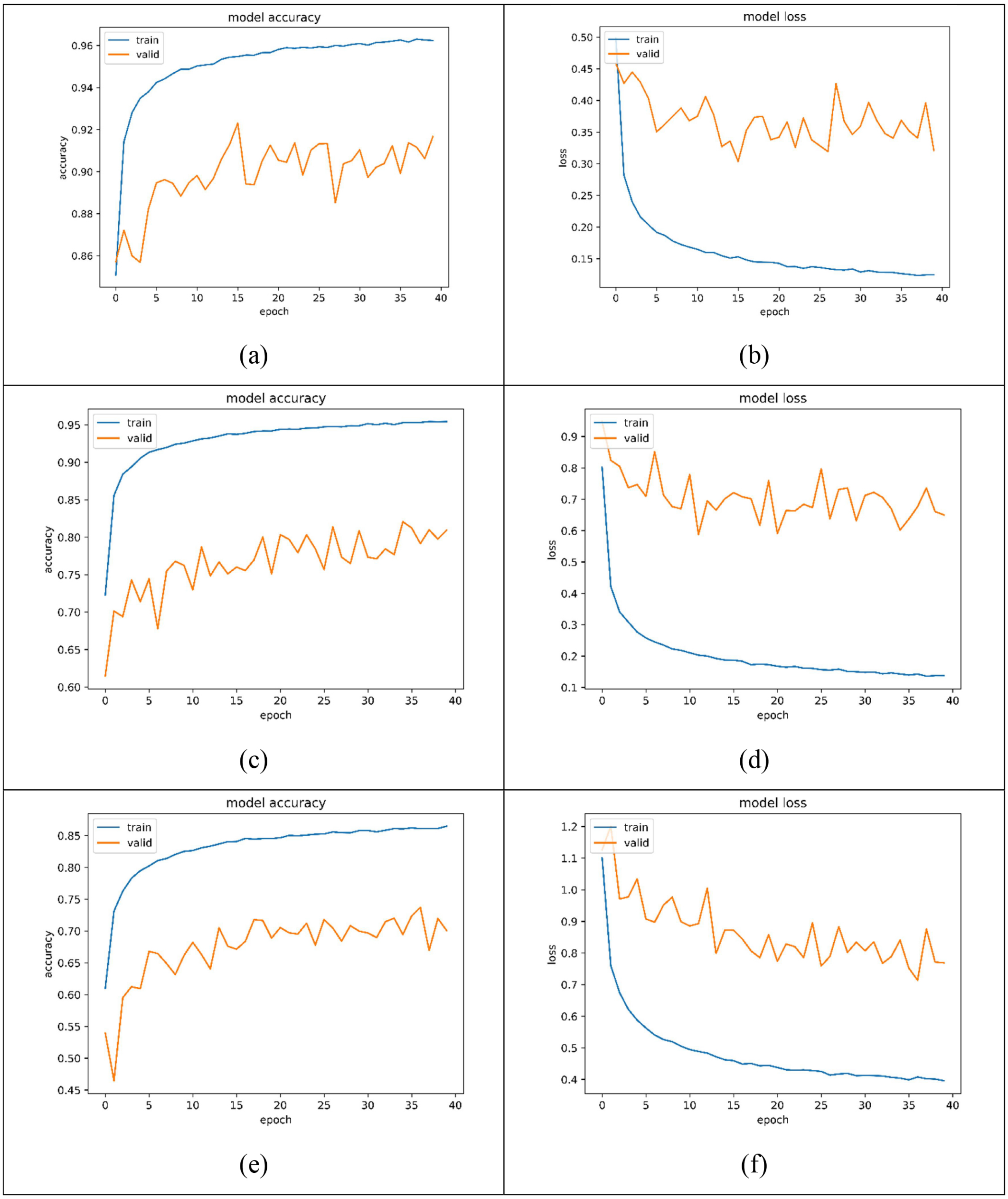

**Fig. 2.** MobileNetV2: Performance evaluation under various training configurations: (a, b) accuracy and loss without balancing or augmentation; (c, d) with data balancing only; (e, f) with balancing and augmentation strategies.

Then, encryption was implemented to the model using the CKKS scheme, observing the performance measurements and adjusting various parameters regarding each run. Table 5 shows the performance measurements of FTL with encryption.

Figs. 4 to 6 show accuracy, precision, recall, and F1-score in federated environments across various data configurations grouped by client numbers 2, 3, and 4. HE-CKKS encryption approach was applied once the initial model was trained without encryption.

Aside from the evaluation metrics computed above, the computation complexity of the entire process starts from the training at the client's side over the GPU until the end of the collaborative training process, then classification results utilizing a secure aggregated model. Tables 6 and 7 display the impact of encryption and the computation complexity in the collaborative process with a different number of clients. Fig. 7 shows the total execution time of an 8-class MobileNetV2 pertained model in 2, 3, and 4 clients.

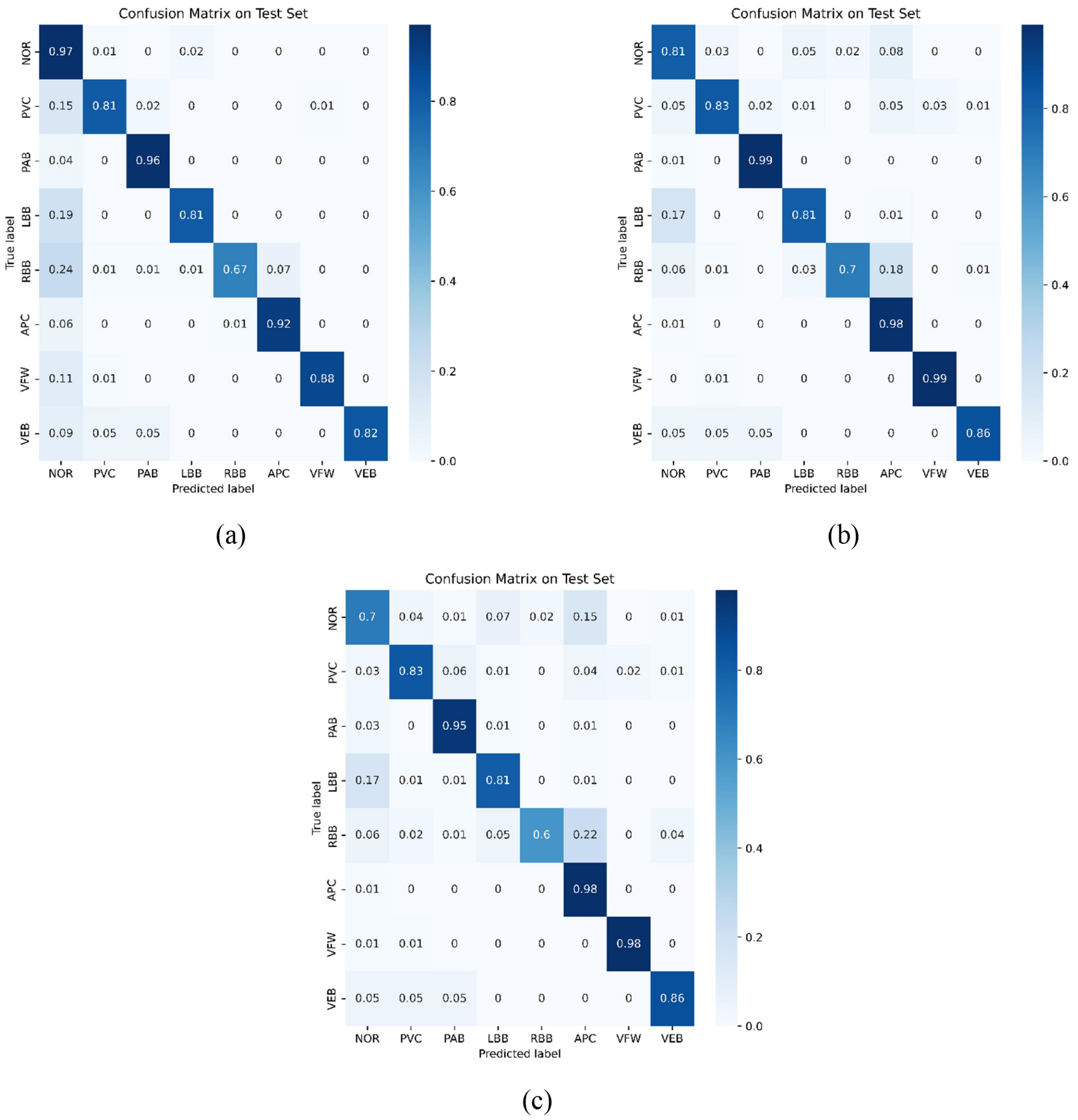

**Fig. 3.** Confusion matrix for heartbeat classification on a test set for the baseline MobileNetV2 model under various configurations: (a) without data balancing and augmentation; (b) with balancing only; (c) with both balancing and augmentation.

**Table 4.** Performance measurements: federated transfer learning without encryption.

| Balance-Augment configuration | Clients | Accuracy | Precision | Recall | F1-score |
|---|---|---|---|---|---|
| NoBalance, NoAugment | 2 | 0.852935731 | 0.925331249 | 0.599296672 | 0.682568185 |
| | 3 | 0.785024166 | 0.830279854 | 0.483591325 | 0.552187302 |
| | 4 | 0.814381242 | 0.968411290 | 0.479613559 | 0.565636811 |
| Balance, NoAugment | 2 | 0.859996259 | 0.670932913 | 0.872986862 | 0.712459160 |
| | 3 | 0.839232624 | 0.656849202 | 0.859086509 | 0.691721481 |
| | 4 | 0.871098101 | 0.684399758 | 0.851597916 | 0.699958531 |
| Balance, Augment | 2 | 0.771042347 | 0.570097763 | 0.833888412 | 0.619122848 |
| | 3 | 0.777173936 | 0.608232374 | 0.781058720 | 0.616837991 |
| | 4 | 0.751486421 | 0.590910337 | 0.784364743 | 0.602211150 |

A histogram was utilized to show the correlation between running time and homomorphic encryption. Fig. 8 shows that HE-CKKS implementation resulted in a slight increase in running time. Utilizing a pre-trained model for training classifier layers reduces overhead.

With the findings being presented above, analyses and insights can be deduced. Due to direct access to

**Table 5.** Performance measurements: federated transfer learning with encryption.

| Balance-Augment configuration | Clients | Accuracy | Precision | Recall | F1-score |
|---|---|---|---|---|---|
| NoBalance, NoAugment | 2 | 0.848522842 | 0.878016053 | 0.609438536 | 0.650703053 |
| | 3 | 0.763517261 | 0.831286286 | 0.428062989 | 0.498521329 |
| | 4 | 0.796915650 | 0.844882119 | 0.431418773 | 0.512032135 |
| Balance, NoAugment | 2 | 0.853818297 | 0.662478607 | 0.869734012 | 0.713714918 |
| | 3 | 0.856140852 | 0.669617578 | 0.859177516 | 0.704492939 |
| | 4 | 0.881224453 | 0.688576401 | 0.851417763 | 0.706130786 |
| Balance, Augment | 2 | 0.693143785 | 0.561076201 | 0.832628086 | 0.597602954 |
| | 3 | 0.679022670 | 0.563893396 | 0.726780494 | 0.547892088 |
| | 4 | 0.767837226 | 0.609095081 | 0.771069807 | 0.602702799 |

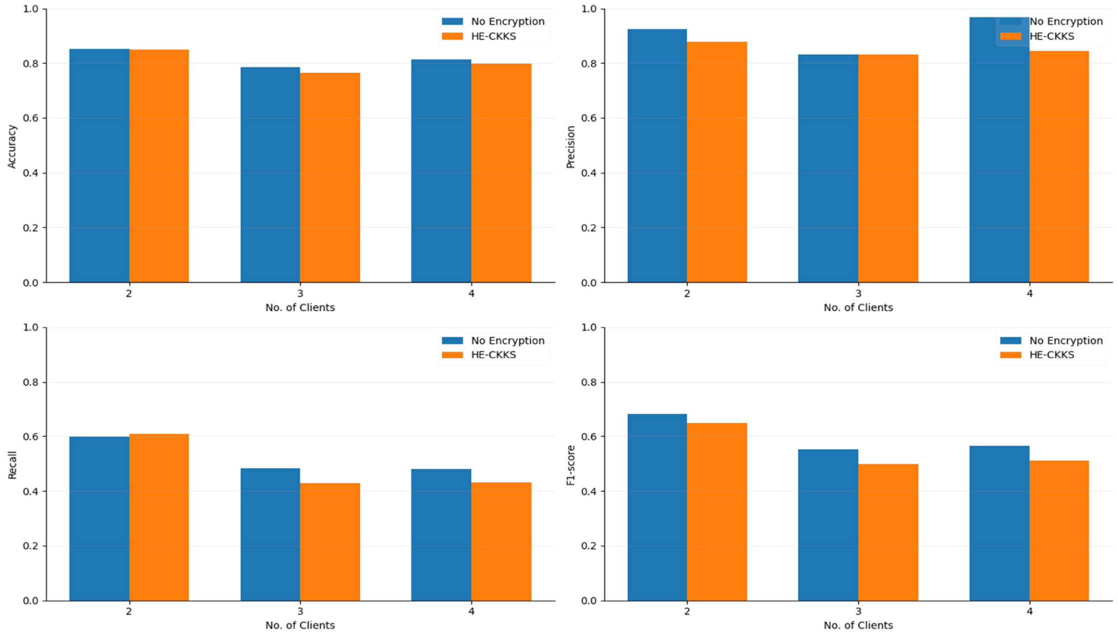

**Fig. 4.** Performance metrics under ‘nobalance, noaugment’ data configuration.

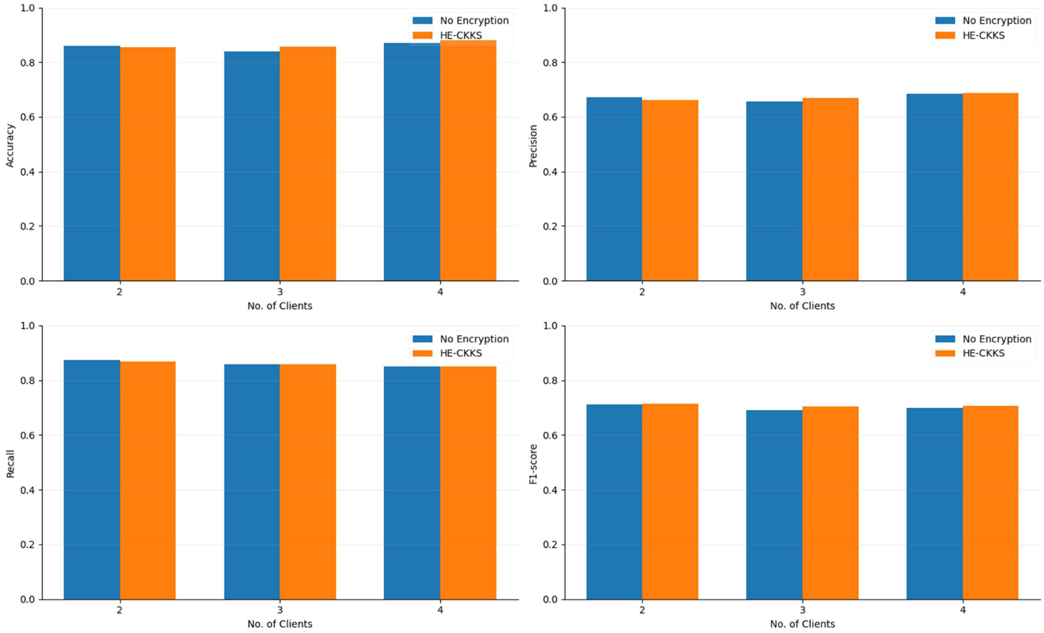

**Fig. 5.** Performance metrics under ‘balance, noaugment’ data configuration.

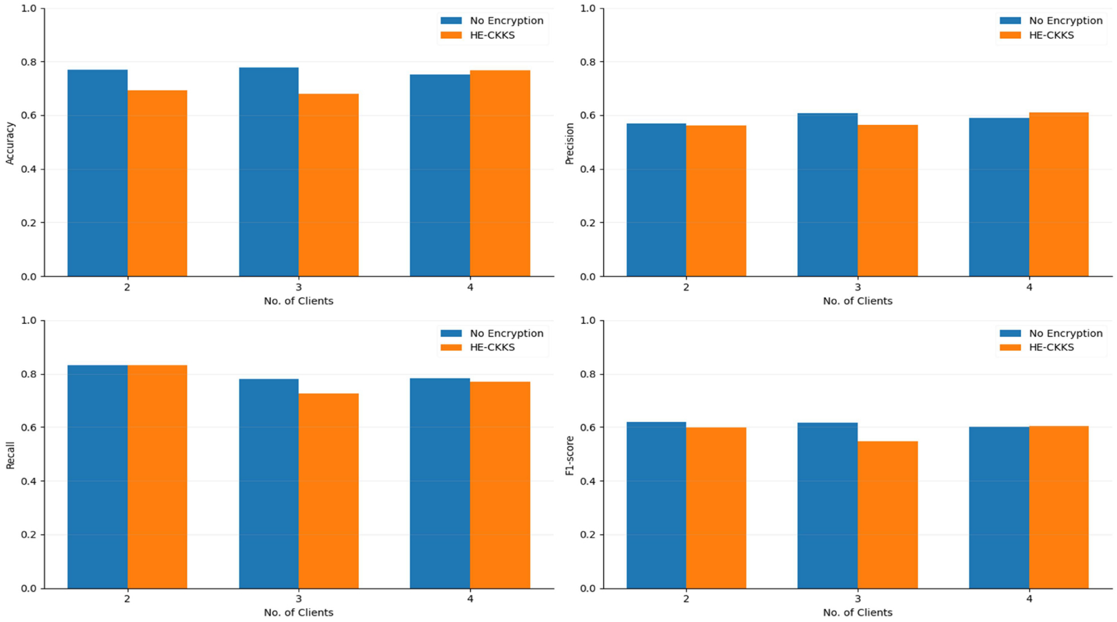

**Fig. 6.** Performance metrics under 'balance, augment' data configuration.

**Table 6.** Computation overhead of MobileNetV2 for 8 classes among different client counts.

| | Time (seconds) | | |
|---|---|---|---|
| Category | Two clients | Three clients | Four clients |
| Generating keys | 0.0425 | 0.0411 | 0.044 |
| Splitting data | 0.8013 | 0.8974 | 0.82 |
| Training client 0 | 1434.6969 | 1094.3825 | 951.0993 |
| Encrypting weights with public key client 0 | 11.9421 | 11.8538 | 11.8986 |
| Training client 1 | 1456.8292 | 1131.1527 | 975.2448 |
| Encrypting weights with public key client 1 | 11.642 | 11.7606 | 11.8525 |
| Training client 2 | | 1113.6686 | 930.612 |
| Encrypting weights with public key client 2 | | 11.5211 | 11.7796 |
| Training client 3 | | | 975.0884 |
| Encrypting weights with public key client 3 | | | 11.8228 |
| Averaging encrypted weights | 4.0318 | 4.0597 | 4.2669 |
| Export aggregated weights | 4.6476 | 4.9047 | 5.0477 |
| Decrypt encrypted aggregated weights | 4.6614 | 4.8288 | 4.7331 |
| Re-integrate decrypted weights to model | 1.1717 | 1.1858 | 1.0474 |
| Model's Evaluation Time | 15.4991 | 15.4991 | 15.4991 |
| Total Execution Time | 2945.9656 | 3405.7559 | 3910.8562 |

**Table 7.** Comparative analysis: computation Cost (Run Time in seconds).

| No_Of_Clients | No Encryption | HE-CKKS |
|---|---|---|
| 2 | 2914.522 | 2945.9656 |
| 3 | 3362.8996 | 3405.7559 |
| 4 | 3855.6382 | 3910.8562 |

unmodified data, the proposed convolutional model can learn from accurate patterns without noise or precision loss in circumstances without encryption. In contrast, encryption increases the computational overhead and affects the algorithm efficiency, influencing the accuracy of the results. Tables 3 and 4 showed experimental results that provided insights into the performance metrics of both encrypted and unencrypted methods. HE-CKKS scheme appears to have a minor influence on FTL model accuracy, which is reassuring since model degradation is a major concern. The balanced dataset outperforms the non-balanced dataset in terms of performance metrics. This indicates that class-balanced models are more robust, especially with regards to recall.

The model is less likely to suffer from client increments, at least in terms of learning, due to its use of TL. In essence, it is not good to increase clients' numbers in the issue at hand simply because the actual data distribution in the classes will not be authentic due to balancing and augmentation

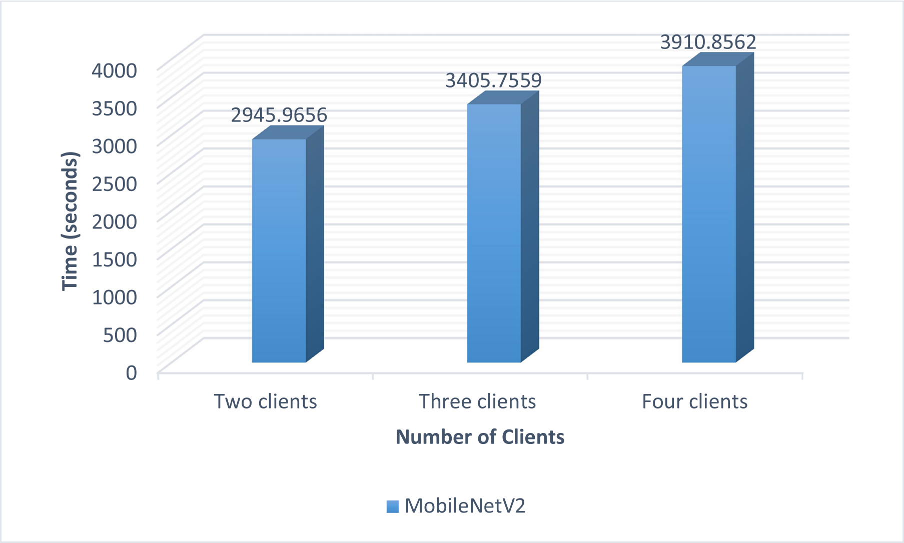

Fig. 7. Total execution time for MobileNetV2 of 8 classes with different client numbers.

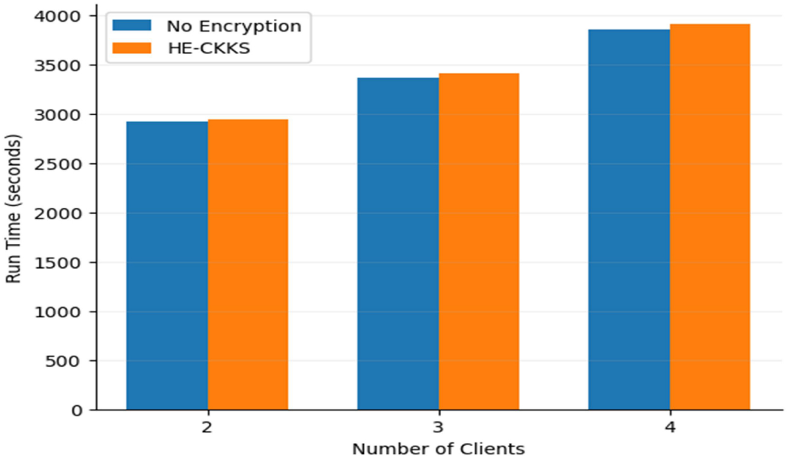

Fig. 8. A comparison of total execution times for MobileNetV2 of 8 classes with or without encryption under different client numbers.

techniques. Furthermore, it is important to note that classes with a small sample size are particularly susceptible to overfitting. When using a DL model and handling more clients, the risk also increases. Both models show improved performance by integrating data balance technology. However, applying balancing and augmentation techniques results in worse performance metrics than not using them. In general, measurements reveal that, from a model performance standpoint, implementing HE-CKKS encryption into FTL is an attractive choice that paves the way for secure, PPDL without affecting much accuracy or effectiveness.

As you can see in Table 5, more research into computational overhead shows that the runtime of both encrypted and unencrypted configurations grows as the number of clients increases, which is expected because of the correlation between the number of clients and computation time. Compared to unencrypted, the CKKS computational cost is slightly higher across board clients, approximately 1.08% for 2 clients, 1.27% for 3 clients, and 1.43% for 4 clients. However, the difference is negligible, indicating that the CKKS adoption is quite successful. Utilizing a pre-trained MobNetV2 as a feature extractor in FTL effectively decreases the computational cost. This is because there is no need for encryption or fine-tuning. When only the classification part of the model is encrypted, the computational overhead is much less than if the entire model architecture were to be encrypted. Table 8 compares performance accuracy and privacy guarantees between our proposed PPFTL framework and the related work approaches for privacy-preserving medical diagnosis.

**Table 8.** Comparison of performance accuracy and privacy guarantees between our proposed PPFTL framework and the related work approaches for medical diagnosis.

| Reference | Privacy guarantees | Dataset (s) | Accuracy (%) |
|---|---|---|---|
| Li, Wenqi [16] | DP | BraTS 2018 | 83%–85% |
| Vijaya Kumar A [17] | MPC, PHE (Additive -Paillier) | COVID-19 patient-related data | N/A |
| Wibawa et al. [18] | MPC, HE | COVID-19 | 82%–85% |
| Kara, Mostefa et al. [19] | FHE using twin key encryption | Patient's heart rate and other related healthcare data | N/A |
| Kachuee, Mohammad [20] | Not Applied | PhysioNet MIT-BIH Arrhythmia and PTB Diagnostic ECG Databases | Arrhythmia:93.4%<br>MI: 95.9% |
| Gao, Dashan [21] | MPC, Paillier's HE, SS, DP | • Spambase<br>• Wisconsin Diagnostic Breast Cancer ("WDBC")<br>• mfeat-fourier<br>• heart disease dataset ("heart")<br>• Default-of-Credit-Card-Clients ("Default-Credit")<br>• Real-world dataset (MIMIC-III) | • 0.8315 ± 0.0355 – Spambase ($\text{HFTL}_{\text{HE}}$).<br>• 0.9491 ± 0.0482 - WDBC ($\text{HFTL}_{\text{LR}}$).<br>• 0.6923 ± 0.0162 - mfeat-fourier ($\text{HFTL}_{\text{SS}}$).<br>• 0.7230 ± 0.0659 - heart ($\text{HFTL}_{\text{SS}}$, $\text{HFTL}_{\text{LR}}$).<br>• 0.5212 ± 0.0423 – Default Credit ($\text{HFTL}_{\text{HE}}$). |
| Liu, Yang et al. [22] | Additive HE, SS | • NUS-WIDE<br>• Default-of-Credit-Card-Clients ("Default-Credit") | N/A |
| Singh et al. [23] | FL for end-device privacy | IoT healthcare related dataset (weight meters, blood pressure, glucose meter, insulin pump) | N/A |
| Walskaar et al. [25] | MPC, xMK-CKKS | COVID-19 | • 89%–97% (clients: 2 to 5)<br>• 50% (clients: 10) |
| **Our proposed PPFTL framework** | **HE-CKKS** | **PhysioNet MIT-BIH Arrhythmia ECG database** | **88% (clients: 2 to 4)** |

## Conclusion

Regulations like Europe's GDPR have brought attention to the necessity for secure data handling methods as data privacy gains prominence, especially in the healthcare industry. FTL reduces data exposure by creating decentralized data training among institutions. Combined with homomorphic encryption, it allows for private and secure computations of sensitive data. These technologies not only improve performance but also increase computing complexity, a trade-off willing to accept as preserving data privacy is essential.

Performance scores within FTL usually surpass 80%. Still, accuracy drops slightly when data balancing and augmentation preprocessing strategies are used at the same time. However, the findings of the proposed paradigm show resilient scores regarding precision, recall, and F1-score despite the slight accuracy reduction. Hence, maintain a balance between precision and recall while at the same time classifying positive cases accurately. Overall model performance is preserved due to CKKS encryption scheme usage. It also preserves the model's privacy and adjusts as well as scales for various applications under different scenarios.

Data driven models within healthcare field are hindered through data security and patients' privacy attacks. As a result, improving the security of data and the processing methods are essential. The suggested framework takes advantage of utilizing FTL and HE technologies, and further improves the security and efficiency of the application via minimizing the shared actual raw data within the healthcare industry. Future research will explore and enhance the employed cryptographic scheme by utilizing multi-key CKKS to solidify the security of our framework. This feature promotes data privacy and model update confidentiality by making aggregated data inaccessible to participants. Also, adopting a weighted averaging strategy incorporating the size of each client's training data could result in a more nuanced model updating approach. Additionally, optimizing PYFHEL's built-in code for parallel processing will significantly reduce the total execution time under encrypted weight sharing, thereby decreasing computational overhead. Furthermore, a scalability assessment of the suggested framework needs to be conducted on a more complex medical dataset and tested with a more considerable number of participating clients.

## Authors' declaration

- Conflicts of Interest: None.
- We hereby confirm that all the Figures and Tables in the manuscript are ours. Furthermore, any Figures and images that are not ours have been included with the necessary permission for republication, which is attached to the manuscript.
- No animal studies are present in the manuscript.
- No human studies are present in the manuscript.
- Ethical Clearance: The project was approved by the local ethical committee at University of Anbar.

## Authors' contribution statement

A. A. A., S. A., and B. A. contributed to the design and implementation of the research, to the analysis of the results and to the writing of the manuscript.

## References

1. Roth GA, Mensah GA, Johnson CO, Addolorato G, Ammirati E, Baddour LM, *et al.* Global Burden of Cardiovascular Diseases and Risk Factors, 1990–2019. J Am Coll Cardiol. 2020 Dec 22;76(25):2982–3021. https://doi.org/10.1016/j.jacc.2020.11.010.
2. Kaptoge S, Pennells L, De Bacquer D, Cooney MT, Kavousi M, Stevens G, *et al.* World Health Organization cardiovascular disease risk charts: revised models to estimate risk in 21 global regions. Lancet Glob Heal. 2019 Oct;7(10):e1332–45. https://doi.org/10.1016/S2214-109X(19)30318-3.
3. Ying Z, Zhang G, Pan Z, Chu C, Liu X. FedECG: A federated semi-supervised learning framework for electrocardiogram abnormalities prediction. J King Saud Univ - Comput Inf Sci. 2023 Jun;35(6):101568. https://doi.org/10.1016/j.jksuci.2023.101568.
4. Almotairi KH, Hussein AM, Abualigah L, Abujayyab SKM, Mahmoud EH, Ghanem BO, *et al.* Impact of Artificial Intelligence on COVID-19 Pandemic: A Survey of Image Processing, Tracking of Disease, Prediction of Outcomes, and Computational Medicine. Big Data Cogn Comput. 2023 Jan 11;7(1):11. https://doi.org/10.3390/bdcc7010011.
5. Abouelmehdi K, Beni-Hssane A, Khaloufi H, Saadi M. Big data security and privacy in healthcare: A Review. Procedia Comput Sci. 2017;113:73–80. https://doi.org/10.1016/j.procs.2017.08.292.
6. Wibawa F, Catak FO, Sarp S, Kuzlu M. BFV-Based Homomorphic Encryption for Privacy-Preserving CNN Models. Cryptography. 2022 Jul 1;6(3):34. https://doi.org/10.3390/cryptography6030034.
7. Regulation P. Regulation (EU) 2016/679 of the European Parliament and of the Council. Regul. 2016;679. http://data.europa.eu/eli/reg/2016/679/oj.
8. Kim HE, Cosa-Linan A, Santhanam N, Jannesari M, Maros ME, Ganslandt T. Transfer learning for medical image classification: a literature review. BMC Med Imaging. 2022 Dec 13;22(1):69. https://doi.org/10.1186/s12880-022-00793-7.
9. Chakir O, Belfaik Y, Sadqi Y. Multi-Key Fully Homomorphic Encryption For Privacy-Preservation Within Federated Learning Environments. Edpacs. 2023;68(6):25–34. https://doi.org/10.1080/07366981.2023.2301832.
10. Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, *et al.* MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. arXiv: 1704.04861; 2017. https://doi.org/10.48550/arXiv.1704.04861.
11. Phong LT, Aono Y, Hayashi T, Wang L, Moriai S. Privacy-Preserving Deep Learning via Additively Homomorphic Encryption. IEEE Trans Inf Forensics Secur. 2018 May;13(5):1333–45. https://doi.org/10.1109/TIFS.2017.2787987.
12. Alloghani M, M. Alani M, Al-Jumeily D, Baker T, Mustafina J, Hussain A, *et al.* A systematic review on the status and progress of homomorphic encryption technologies. J Inf Secur Appl. 2019 Oct;48:102362. https://doi.org/10.1016/j.jisa.2019.102362.
13. Xu J, Glicksberg BS, Su C, Walker P, Bian J, Wang F. Federated Learning for Healthcare Informatics. J Healthc Informatics Res. 2021 Mar 12;5(1):1–19. https://doi.org/10.1007/s41666-020-00082-4.
14. Rieke N, Hancox J, Li W, Milletarì F, Roth HR, Albarqouni S, *et al.* The future of digital health with federated learning. NPJ Digit Med. 2020 Sep 14;3(1):119. https://doi.org/10.1038/s41746-020-00323-1.
15. Antunes RS, André da Costa C, Küderle A, Yari IA, Eskofier B. Federated Learning for Healthcare: Systematic Review and Architecture Proposal. ACM Trans Intell Syst Technol. 2022 Aug 31;13(4):1–23. https://doi.org/10.1145/3501813.
16. Li W, Milletarì F, Xu D, Rieke N, Hancox J, Zhu W, *et al.* Privacy-Preserving Federated Brain Tumour Segmentation. In: Suk HI, Liu M, Yan P, Lian C, editor. In Machine Learning in Medical Imaging: 10th International Workshop, MLMI 2019, Held in Conjunction with MICCAI 2019. Springer International Publishing; 2019:133–41. https://doi.org/10.1007/978-3-030-32692-0_16.
17. Vijaya Kumar A, Sujith MS, Sai KT, Rajesh G, Yashwanth DJS. Secure Multiparty computation enabled E-Healthcare system with Homomorphic encryption. IOP Conf Ser Mater Sci Eng. 2020 Dec 1;981(2):022079. https://doi.org/10.1088/1757-899X/981/2/022079.
18. Wibawa F, Catak FO, Kuzlu M, Sarp S, Cali U. Homomorphic Encryption and Federated Learning based Privacy-Preserving CNN Training: COVID-19 Detection Use-Case. In: EICC 2022: Proccedings of the European Interdisciplinary Cybersecurity Conference. New York, NY, USA: ACM; 2022:85–90. https://doi.org/10.1145/3528580.3532845.
19. Kara M, Laouid A, Yagoub MA, Euler R, Medileh S, Hammoudeh M, *et al.* A fully homomorphic encryption based on magic number fragmentation and El-Gamal encryption: Smart healthcare use case. Expert Syst. 2022 Jun 12;39(5):1–14. https://doi.org/10.1111/exsy.12767.
20. Kachuee M, Fazeli S, Sarrafzadeh M. ECG Heartbeat Classification: A Deep Transferable Representation. In: 2018 IEEE International Conference on Healthcare Informatics (ICHI). IEEE; 2018:443–4. https://doi.org/10.1109/ICHI.2018.00092.
21. Gao D, Liu Y, Huang A, Ju C, Yu H, Yang Q. Privacy-preserving Heterogeneous Federated Transfer Learning. In: 2019 IEEE International Conference on Big Data (Big Data). IEEE; 2019:2552–9. https://doi.org/10.1109/BigData47090.2019.9005992
22. Liu Y, Kang Y, Xing C, Chen T, Yang Q. A Secure Federated Transfer Learning Framework. IEEE Intell Syst. 2020 Jul 1;35(4):70–82. https://doi.org/10.1109/MIS.2020.2988525.
23. Singh S, Rathore S, Alfarraj O, Tolba A, Yoon B. A framework for privacy-preservation of IoT healthcare data using Federated Learning and blockchain technology. Futur Gener Comput Syst. 2022 Apr;129:380–8. https://doi.org/10.1016/j.future.2021.11.028.

24. Faheem M, Kuusniemi H, Eltahawy B, Bhutta MS, Raza B. A lightweight smart contracts framework for blockchain-based secure communication in smart grid applications. IET Gener Transm Distrib. 2024 Feb 13;18(3):625–38. https://doi.org/10.1049/gtd2.13103.
25. Walskaar I, Tran MC, Catak FO. A Practical Implementation of Medical Privacy-Preserving Federated Learning Using Multi-Key Homomorphic Encryption and Flower Framework. Cryptography. 2023 Oct 4;7(4):48. https://doi.org/10.3390/cryptography7040048.
26. Ma J, Naas S, Sigg S, Lyu X. Privacy-preserving Federated Learning based on Multi-key Homomorphic Encryption. Int J Intell Syst. 2021 Apr 14;37(9):5880–901. https://doi.org/10.1002/int.22818.
27. Maftouni M, Shen B, Law ACC, Yazdi NA, Hadavand F, Ghiasvand F, *et al.* A mask-guided attention deep learning model for COVID-19 diagnosis based on an integrated CT scan images database. IISE Trans Healthc Syst Eng. 2023 Apr 3;13(2):132–49. https://doi.org/10.1080/24725579.2022.2142866.
28. Yousif HM, Hameed SM. Review of Challenges and Solutions for Genomic Data Privacy-Preserving. Iraqi J Sci. 2023 Sep 30;64(9):4729–46. https://doi.org/10.24996/ijs.2023.64.9.35.
29. Marcolla C, Sucasas V, Manzano M, Bassoli R, Fitzek FHP, Aaraj N. Survey on Fully Homomorphic Encryption, Theory, and Applications. Proc IEEE. 2022 Oct;110(10):1572–609. https://doi.org/10.1109/JPROC.2022.3205665.
30. Alsaedi EM, Kadhim FA. Retrieving Encrypted Images Using Convolution Neural Network and Fully Homomorphic Encryption. Baghdad Sci J. 2023 Feb 1;20(1):0206. https://doi.org/10.21123/bsj.2022.6550.
31. Challa R, Gunta V. A Modified Symmetric Key Fully Homomorphic Encryption Scheme Based on Read-Muller Code. Baghdad Sci J. 2021 Jun 20;18(2(Suppl.)):0899. https://doi.org/10.21123/bsj.2021.18.2(Suppl.).0899.
32. Cheon JH, Kim A, Kim M, Song Y. Homomorphic Encryption for Arithmetic of Approximate Numbers. In: Takagi TPT, editor. International Conference on the Theory and Application of Cryptology and Information Security. Springer. 2017:409–37. https://doi.org/10.1007/978-3-319-70694-8_15.
33. Li B, Micciancio D. On the Security of Homomorphic Encryption on Approximate Numbers. In: Canteaut A, Standaert FX, editors. Advances in Cryptology – EUROCRYPT. Springer International Publishing; 2021:648–77. https://doi.org/10.1007/978-3-030-77870-5_23.
34. van Veen EB. Observational health research in Europe: understanding the General Data Protection Regulation and underlying debate. Eur J Cancer. 2018 Nov;104:70–80. https://doi.org/10.1016/j.ejca.2018.09.032.
35. Gajendran MK, Khan MZ, Khattak MAK. ECG Classification using Deep Transfer Learning. In: 2021 4th International Conference on Information and Computer Technologies (ICICT). IEEE;2021:1–5. https://doi.org/10.1109/ICICT52872.2021.00008.
36. Zhuang F, Qi Z, Duan K, Xi D, Zhu Y, Zhu H, *et al.* A Comprehensive Survey on Transfer Learning. Proc IEEE. 2021 Jan;109(1):43–76. https://doi.org/10.1109/JPROC.2020.3004555.
37. Salem M, Taheri S, Shiun YJ. ECG Arrhythmia Classification Using Transfer Learning from 2- Dimensional Deep CNN Features. In: 2018 IEEE Biomedical Circuits and Systems Conference (BioCAS). Cleveland; OH; USA: IEEE; 2018:1–4. https://doi.org/10.1109/BIOCAS.2018.8584808.
38. Ali AH, Yaseen MG, Aljanabi M, Abed SA, GPT C. Transfer Learning: A New Promising Techniques. Mesopotamian J Big Data. 2023 Feb 9;25(3):29–30. https://doi.org/10.58496/MJBD/2023/004.
39. Nguyen CV, Do CD. Transfer Learning in ECG Diagnosis: Is It Effective. 2024 Feb 2. https://doi.org/10.48550/arXiv.2402.02021.
40. Li AS, Iyengar A, Kundu A, Bertino E. Transfer Learning for Security: Challenges and Future Directions. arXiv preprint arXiv:2403.00935, 2024. https://doi.org/10.48550/arXiv.2403.00935.
41. Akram A, Rashid J, Jaffar MA, Faheem M, Ul AR. Segmentation and classification of skin lesions using hybrid deep learning method in the Internet of Medical Things. Ski Res Technol. 2023;29(11):1–14. https://doi.org/10.1111/srt.13524.
42. Moody GB, Mark RG. The impact of the MIT-BIH Arrhythmia Database. IEEE Eng Med Biol Mag. 2001;20(3):45–50. https://doi.org/10.1109/51.932724.
43. Ibarrondo A, Viand A. Pyfhel. In: Proceedings of the 9th on Workshop on Encrypted Computing & Applied Homomorphic Cryptography. New York, NY, USA: ACM; 2021:11–6. https://doi.org/10.1145/3474366.3486923.
44. Chen H, Laine K, Player R. Simple Encrypted Arithmetic Library - SEAL v2.1. Cryptology ePrint Archive. 2017:3–18. https://doi.org/10.1007/978-3-319-70278-0_1.
45. Fan J, Vercauteren F. Somewhat practical fully homomorphic encryption. IACR Cryptol ePrint Arch. 2012;2012:144.
46. Vemuri N. AI-Driven DevOps Practices for Healthcare Data Security and Compliance. Int J Intell Syst Appl Eng. 2024;12(16s):297–305.

# نحو تصنيف تخطيط القلب بشكل كفؤ والمحافظة على الخصوصية: باستخدام طريقة تعلم النقل الاتحادي والمعزز بواسطة التشفير المتماثل القائم على خوارزمية CKKS

**انمار علي الجنابي[1]، سفيان تايه فرج الجنابي[2]، بلال اسماعيل الخطيب[2]**

*[1] كلية علوم الحاسوب، الجامعة التكنولوجية - العراق، بغداد، العراق.*

*[2] قسم علوم الحاسبات، كلية علوم الحاسوب وتكنولوجيا المعلومات، جامعة الانبار، الرمادي، الانبار، العراق.*

## المستخلص

في مجال الرعاية الصحيه، يُعد الحفاظ على دقة وخصوصية التشخيص الطبي بشكل تعاوني تحديًا كبيرًا. على حد علمنا، يقترح هذا البحث أول إطارعمل متكامل للتعلم الاتحادي للنقل الحافظ للخصوصيه (PPFTL) لتصنيف عدم انتظام ضربات القلب لتخطيط القلب باستخدام صور ثنائية الأبعاد. يساهم دمج التعلم المنقول (TL) في تقليص الفجوة بين نسختي إطارالعمل المشفرة وغير المشفرة في خطوة التدريب. يتضمن هذا النهج تحويل إشارات تخطيط القلب الكهربائيه (ECG) الخام إلى صور ثنائية الأبعاد ذات تدرج رمادي لتخطيط كهربائية القلب (ECG). يتم توزيع مجموعة البيانات بعد تحويلها إلى صور ثم يتم تغذيتها كمدخلات في النماذج المحليه، حيث يعمل MobileNetV2 كمستخرج للميزات. تشمل عملية التدريب الخاصه بكل مشترك ضمن اطار العمل العام بتقنيات تعمل على موازنة فئات التصنيف المتختلفه بالاضافة الى زيادة حجم البيانات لكل من هذه الفئات وبالتالي تحسين اداء وزيادة كفاءة النموذج المقترح. تتعرض نماذج التعلم العميق (DL) للعديد من هجمات الخصوصيه للحصول على بيانات حساسه. ونتيجة لذلك، يقوم نظام التشفير المتماثل(HE-CKKS) Son-Kim-Kim-Cheong بتشفيراوزان النموذج فقط لحماية النماذج العميقه من هجمات الخصوم، مما يمنع مشاركة البيانات الخام الحساسه. أظهرت النتائج التجريبيه على مجموعة البيانات (BIH Arrhythmia-MIT) دقه بنسبة 88.12%. أدى دمج نظام التشفيرالمتماثل CKKS-HE إلى زيادة التوقيتات الخاصه بالتفيذ بنسبة 1.08%، 1.27%، و1.43% لـ 2 و 3 و4 مشتركين، على التوالي.

الكلمات المفتاحية: خصوصية البيانات، تخطيط كهربائية القلب، التعلم الاتحادي، التشفير المتماثل، نقل التعلم.