Optimization of CNN-based Federated Learning for Cyber-Physical Detection

Ammar Kamal Abasi[a], Moayad Aloqaily[a], Bassem Ouni[b], Maher Hamdi[c]

[a]Mohamed Bin Zayed University of Artificial Intelligence (MBZUAI), Abu Dhabi, United Arab Emirates
[b]Technology Innovation Institute, Abu Dhabi, UAE
[c] George Washington University, Washington, USA
E-mails: {ammar.abasi; moayad.aloqaily}@mbzuai.ac.ae, bassem.ouni@tii.ae, mhamdi@ieee.org

2023 IEEE 20th Consumer Communications & Networking Conference (CCNC) | 978-1-6654-9734-3/23/$31.00 ©2023 IEEE | DOI: 10.1109/CCNC51644.2023.10059840

***Abstract*—With the increasing popularity of Cyber-physical Systems (CPS), there is a growing need for efficient and reliable methods for detecting and responding to threats. Federated Learning (FL) is a distributed Machine Learning (ML) technique that can be used to train models on data from multiple devices (i.e., edge devices) while keeping the data local. FL has the potential to improve the security and privacy of data while also reducing the training time and cost. Particularly, CNN-based FL has been shown to be effective for various tasks such as image classification and object detection. However, selecting suitable hyperparameters for constructing local ML models in FL is a significant challenge for practical inference and training on edge devices. In this paper, we focus on the optimization of CNN-based federated learning for the task of cyber-physical detection and we propose employing a novel metaheuristic optimization algorithm called Honey Badger Algorithm (HBA) for tuning the hyperparameters in local ML models (FL-HBA). To show the effectiveness of FL-HBA, we make an evaluation using an intelligent healthcare case study where we consider Sleep Apnea (SA) and use the PhysioNet apnea ECG dataset to diagnose SA. Our results show that the FL-HBA is superior to a Convolutional Neural Network (CNN) baseline, traditional ML techniques, and centralized learning models. Furthermore, we demonstrate that the proposed method for assigning the near-optimal hyperparameter values for centralized learning models improves accuracy by 2%.**

***Index Terms*—Cyber-physical Systems (CPS), Federated Learning (FL), CNN Hyper-parameter, Honey Badger Algorithm (HBA), Sleep Apnea (SA).**

I. Introduction

Cyber-physical Systems (CPS) applications are pivotal in leading the 21st-century information technology revolution. With heterogeneous data produced by interconnected sensors, the Internet of Things (IoT) has evolved to serve humans more cognitively, based on the CPS's architecture and design [1]. Wearable sensors like Electrocardiograms (ECG) and other Internet-of-Medical-Things (IoMT) devices play an essential role in collecting medical data in the intelligent healthcare environment. This data are then analyzed using sophisticated data analytics facilitated by Machine Learning (ML) to achieve a wide variety of exciting, intelligent healthcare applications, like diagnosing diseases and monitoring sleep quality monitoring [2].

Formerly, intelligent healthcare systems frequently depend on centralized ML functions hosted in a data center or the cloud to perform health analytics and data learning. This centralized solution has become less practical with the growing quantity of IoMT devices, health data, and raw data transmission in modern healthcare networks. In addition, the confidence in such a third party for data learning or a centralized server creates significant patient privacy issues, e.g., data breach and user information leakage [3].

In light of this, Federated Learning (FL) is an essential solution for producing high-privacy, low-cost intelligent healthcare systems [4]. Fundamentally, FL is an ML mechanism that allows high-quality ML models, such as Sleep Apnea (SA) detection models, to be trained by aggregating local trained models from various healthcare at edge devices without sharing their data. A more detailed discussion of FL and IoMT can be found in a recent work [5]. So far, the majority of the research that has been done on FL has concentrated on investigating communication, privacy, and global optimization. However, only a limited amount of attention has focused on analyzing the critically important matter of effectively carrying out local training and inference at the edge devices.

SA is the most prevalent respiratory condition, characterized by respiratory difficulties [6]. This disorder's severity is determined by the duration of an individual's inability to breathe. This condition is referred to as Obstructive Sleep Apnea (OSA) if the interval is notably long. A subtle form of this condition is characterized by symptoms such as fatigue upon awakening and loud snoring. The inhalation muscles cease functioning for an extended period of time, resulting in damage to the hippocampal area of the brain and other catastrophic conditions. It is estimated that around 1 billion of the world's 7.3 billion adults between the ages of 30 and 69 have OSA sleep-disordered breathing [7]. Early detection can be the key to protecting oneself from potentially dangerous diseases.

Several strategies for identifying apnea using ECG centralized data have been developed. One such method is the K-Nearest Neighbor (KNN) classifier, and the wavelet transform with linear discriminant [8]. The contemporary method for diagnosing SA consists of two phases. The ECG centralized data is preprocessed using a weighted computation approach to generate appropriate groups for the upcoming phase [6]. Recently, a one-dimensional Convolutional Neural Network

(CNN) has been proposed as a deep learning technique for SA detection [9].

Moreover, there are several hyperparameters in deep learning, and it is difficult to assign the optimum value manually; hence, automation has been extensively studied. The metaheuristic algorithms such as Lemur Optimizer (LO) [10] and other algorithms [11] have been developed and applied to enhance the CNN performance by selecting the best hyperparameter values.

Honey Badger Algorithm (HBA) [12] is a new metaheuristic algorithm. HBA is a mathematically efficient search method that was designed to tackle optimization problems. HBA was motivated by the sophisticated foraging behavior of honey badgers. In HBA, exploration and exploitation depict the dynamic search activity of honey badgers, which includes digging and honey-finding tactics. The HBA offers several benefits over other algorithms. During a particular search, it is trying to provide a good balance between exploitation and exploration. No mathematical derivation of specific data is necessary; therefore, just a few parameters must be established during the initialization phase. In addition, it is straightforward, sound-and-complete, adaptable, expandable, and versatile. Consequently, HBA was used for various optimization challenges, such as medical image classifications [13]. This paper uses the HBA to optimize CNN hyperparameters-based FL (HBA-FL) for SA detection using a decentralized collection of medical data.

The remainder sections of this paper are structured as follows. Section II examines related work. Section III explains the proposed method in depth. Section IV presents the analytical and empirical findings on the proposed method's efficient performance. The conclusion and suggestions for further research are included in the section V.

II. Preliminaries

The Honey Badger Algorithm (HBA) is an algorithm that attempts to mimic the natural behaviors of honey badgers in the wild. This section describes the inspiration and mathematical concepts behind HBA. Besides, it briefly introduces a Convolutional Neural Network (CNN).

A. Honey Badger Algorithm (HBA)

1) Inspiration: HBA is a new metaheuristic algorithm proposed by Hashim *et al.* [12]. HBA replicates the honey badger's behavior of hunting and gathering food. The honey badger either follows the honeyguide bird or digs and smells to locate food sources. We refer to the first scenario as "honey mode" and the second scenario as "digging mode." In the previous phase, honey badgers directly locate the beehive with the aid of honeyguide birds. In the latter technique, it utilizes its sense of smell to approximate the position of prey; upon arrival, it wanders about the prey to determine the optimal spot for digging and capturing it.

2) Mathematical Model: HBA is a global optimization technique since it mathematically includes exploitation and exploration phases. The HBA's mathematical steps are described as follows. Here, the population of potential HBA solutions is depicted as:

$$\text{HB} = \begin{bmatrix} hb_1^1 & hb_1^2 & \cdots & hb_1^D \\ hb_2^1 & hb_2^2 & \cdots & hb_2^D \\ \vdots & \vdots & \cdots & \vdots \\ hb_N^1 & hb_N^2 & \cdots & hb_N^D \end{bmatrix}. \tag{1}$$

where HB is a candidate solutions to a given problem, $hb_i = [hb_i^1, hb_i^2 ..., hb_i^d]$, ith is a position index of the honey badger.

1) **Step 1: The initialization phase.** Initialize N honey badgers and their positions using Eq.(2).

$$hb_i = lb_i + rand1() \times ((ub_i - lb_i)), \tag{2}$$

wherein ub_i and lb_i are the upper and lower search domain boundaries, respectively, and hb_i is the ith honey badger position corresponding to a solution space in a population containing N honey badgers.

2) **Step 2: Defining intensity (I).** The concentration of the prey and the prey's distance from the honey badger are both factors that influence the intensity of the honey badger's attack. S_i represents the strength of the prey's smell; if the scent is strong, the move will be fast, and vice versa, as dictated by the Inverse Square Law [14] and Eq.(5).

$$Dis_i = hb_{prey} - hb_i \tag{3}$$

$$S_i = (hb_i - hb_{i+1})^2 \tag{4}$$

$$I_i = rand2() \times \frac{S_i}{4\pi {Dis_i}^2} \tag{5}$$

where Dis_i represents the distance between the position of the prey hb_{prey} and the badger hb_i, and Si represents the strength of the concentration or source.

3) **Step 3: Update density factor.** The time-varying randomness is controlled by the density factor (ð), providing for a consistent move from exploration to exploitation. The randomness is then reduced by updating the decreasing factor at each iteration, as shown in Eq.(6):

$$ð = P2 \times \exp \frac{-itr}{max_{itr}} \tag{6}$$

where $P2$ is a constant (the default is 2) that decreases with each iteration. max_{itr} is a maximum number of iterations, itr is a current iteration.

4) **Step 4: Escaping from the local optimum.** Along with the two succeeding stages, this step is utilized to escape local optima zones. Specifically, the HBA utilizes a flag U that adjusts the search direction in order to give honey badgers with several opportunities to explore the search space fully.

5) **Step 5: Changing the positions of agents.** As already mentioned, the HBA position update process (hb_{new})

consists of two phases: "honey phase or exploitation." and "digging phase or exploration". The following is explained in further detail:

a) Step **5-1: Digging phase or exploration.** Honey badgers approximate the shape of a Cardioid during the digging phase. The modeling of the cardioid motion is given in Eq.(7).

$$\begin{aligned} hb_{new} = hb_{prey} + U \times \wp \times I_i \times hb_{prey} \\ + A \times rand3() \times \eth \times Dis_i \\ \times (\cos(2\pi \times rand4()) \\ \times (1 - \cos(2\pi \times rand5()))) \end{aligned} \tag{7}$$

where hb_{prey} is the position of the prey, which is the best position identified, or the position of the best solution. $\wp$ is a constant value (the default value = 6 ; $\wp \geq 1$) indicates the honey badger's ability to get food. According to Eq.(3): Dis_i is the distance between the prey and the ith honey badger. Random values rand3(), rand4(), and rand5() range between [0, 1]. U serves as a flag that affects the search direction; it is calculated by Eq.(8).

$$U = \begin{cases} -1 & rand6() >, 0.5, \\ 1 & , Otherwise \end{cases} \tag{8}$$

where rand6() generates a random number within the range [0, 1], during the digging phase, honey badgers rely heavily on the scent intensity I of prey hb_{prey}, the distance between the badger and the prey Dis_i, and the time-varying search influence factor. Moreover, the badger's digging process could be interrupted by any disturbance U, allowing it to choose an even more advantageous ambush position.

b) **Step 5-2: Honey phase or exploitation.** The scenario in which a honey badger follows a honey guide bird to a beehive may be reproduced using Eq.(9).

$$hb_{new} = hb_{prey} + U \times rand7() \times \eth \times Dis_i \tag{9}$$

Where hb_{new} represents the honey badger's new position and hb_{prey} represents the prey's position, $\eth$ and U are determined using Eqs.(6) and (8), respectively. Based on the distance information Dis_i, Eq.(9) indicates that a honey badger conducts a search near the prey location hb_{new} that has been located thus far. Variable search behavior over time ($\eth$) affects the search at this stage. It is worth mentioning that all HBA equations were taken from [12].

B. Convolutional Neural Network (CNN)

In recent years, CNN has acted as a focal point for artificial intelligence research. It has been utilized well in voice recognition, Natural Language Processing (NLP), and image classification [15]. By utilizing a deep neural network, it is able to reproduce the complex hierarchical structure of the human vision. CNN is also applied in developing complex signal analysis systems [16] because of its efficiency in automated feature extraction. For instance, the authors in [17] classified ECGs using CNN. The proposed SA detection model is developed based on modified LeNet-5, an effective CNN implementation. The modified LeNet-5 given in [9] will be used in the next section.

III. Proposed Method

This section introduces the proposed method to optimize the CNN hyperparameter. First, the dataset is given. Second, preprocessing is provided. Then, the overall framework of the proposed method in an FL environment is then shown.

A. Dataset

In this study, the PhysioNet apnea ECG dataset was employed. This dataset is publicly accessible [18]. Recordings of 70 people's single-lead ECG ranging from 401 to 587 minutes are included. Each 1-minute segment of ECG signal recording was professionally labeled (in the event of apnea, it was classified as SA; otherwise, it was classified as normal). In addition, the Apnea-Hypopnea Index (AHI) value was used to classify these recordings into three main classes (i.e., classes A, B, and C). Class A indicates that there was at least 100 SA segments in the recording over the entirety of the recording and that each hour included at least (AHI $\geq$ 10) segments. Class B indicates that the recording had at least 99 SA segments and that each hour included at least (AHI $\geq$ 5) segments. Normal or class C indicates that each hour of the recording included at least (AHI $<$ 5) segments.

B. Preprocessing

In this study, an automated method is utilized to extract features from amplitudes as well as RR intervals. This purpose required a preprocessing approach to acquire the RR intervals and amplitudes. ECG signal segments labeled with "SA" and $\pm$ adjacent segments were extracted for processing because previous studies [9] found that adjacent segment information (five 1-minute segments in total) is essential for per-segment SA detection. We initially utilize the Hamilton technique [19] to identify the R-peaks, then calculate the distance between R-peaks (i.e., RR intervals) based on their positions and extracted the R-peak values (amplitudes). Since the retrieved RR intervals contained medically uninterpretable points, the median filter recommended in [9] was utilized. Because the generated amplitudes and RR intervals did not correspond to equal time intervals, as needed by the proposed method, we apply cubic interpolation across 5-minute segments to get 900 points of amplitude and 900 points of RR intervals. Fig.1 comprehensively illustrates the preprocessing approach.

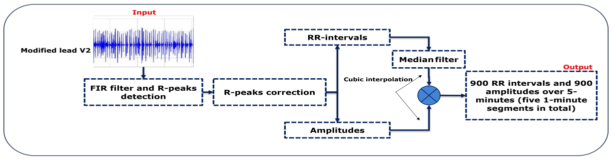

Fig. 1. Preprocessing scheme for the PhysioNet Apnea-ECG datasets.

C. *Adapted HBA for CNN hyperparameters fine-tuning*

In this study, a modified leNet-5 is employed as the primary implementation of the CNN architecture. This reasonably simple architecture consists of two convolutional layers and two fully linked layers frequently used in studies as a comparative design. The baseline and optimization parameters of LeNet-5 is shown in Table I. The optimal range for the number of filters in convolution layers Conv1.1D and Conv3.1D is 32 and 64, respectively, and the kernel size is 3, 5, or 7. Each layer's activation function utilizes "sigmoid", "relu", and "tanh", while the batch size is tuned between 10 and 128. The optimizer employs Adam or Stochastic Gradient Descent (SGD) with a respective learning rate of 0.01. The open-source neural network framework Keras is utilized to construct the local models at the edge devices.

TABLE I
HYPERPARAMETER OF MODIFIED LENET-5 ADAPTED FOR SA

Related Layer	Hyperparameter	Optimization Value	Baseline	Output Shape
Input	-	-	-	(None, 900, 2)
Conv1.1D	Number of filters	32	32	(None, 448, 32)
Conv1.1D	Activation function	tanh,relu,sigmoid	relu	
Max pooling2	Pool size	3	3	(None, 149, 32)
Conv3.1D	Number of filters	32	32	(None, 73, 64)
Conv3.1D	Activation function	tanh,relu,sigmoid	relu	
Max pooling4	Pool size	3	3	(None, 24, 64)
Dropout5	rate	0.4-0.8	0.8	(None, 24, 64)
FC6	units	4-200	32	(None, 32)
FC6	Activation function	tanh,relu,sigmoid	relu	
Output	Activation function	softmax	softmax	(None, 2)
-	Optimizer	SGD,Adam	Adam	-
-	Batch size in the training	10 – 128	128	-

The overall framework of the proposed method in an FL environment to optimize the hyperparameters for modified leNet-5 local models trained at the edge devices is shown in Fig.2. First, the selected edge devices are received the initial model from the server, and then at each edge device, a set of honey badgers HB (i.e., solutions) are initialized randomly using Eq.(1). Each solution is considered a different hyperparameter configuration. Second, the solutions are evaluated using the modified LeNet-5. In this study, one-dimensional data is used for the time series. Fully connected layer nodes, strides in the convolution layer, and feature maps in the basic LeNet-5 may not be appropriate for the SA detection scenario. Therefore, we used the modified LeNet-5, which was introduced by [9] as follows:

- In the feature extraction step, one-dimensional convolution is used rather than a two-dimensional convolution operation.
- A dropout layer is added to prevent overfitting between the fully connected layer and the convolution layer.
- One fully connected layer is kept to reduce the complexity of the network.
- The number of fully connected and the size of strides in the convolution layer is adjusted.

Third, the positions of the solutions are updated according to the HBA mechanism and their fitness values. Next, all updated local trained models from the selected edge devices are collected by the server and then aggregated using the Federated Averaging (FedAvg) method [20]. The pseudo-code of the proposed method's steps is provided in Algorithm 1.

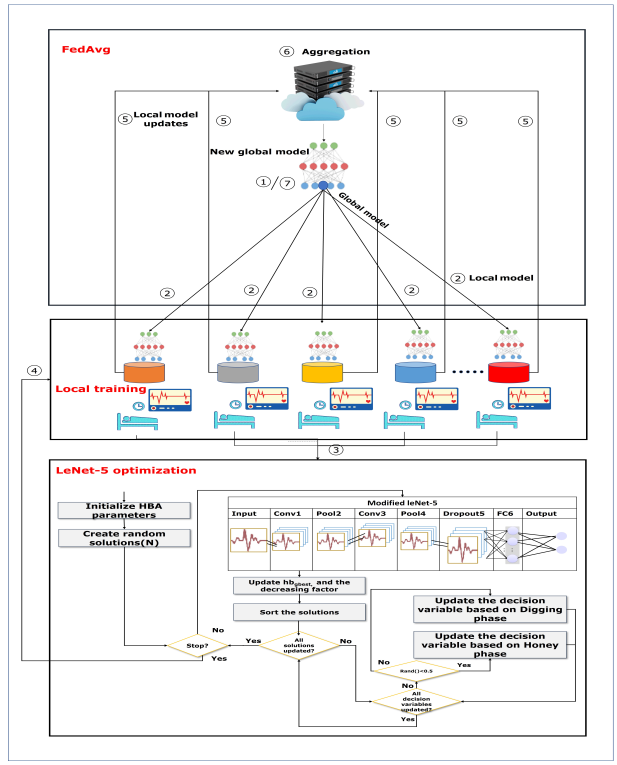

Fig. 2. Proposed method framework.

IV. RESULTS AND DISCUSSION

This section uses extensive experiments to evaluate the performance of HBA for SA detection in an FL environment (FL-HBA). All experiments are carried out on a single widely used PhysioNet apnea ECG dataset as a case study, and traditional HBA and five state-of-the-art methods are utilized to compare comparable studies.

A. *Experimental settings*

In the field of SA detection using a single-lead ECG signal, the existing methods primarily rely on extracting suitable features based on the knowledge and experience of experts and then building a model utilizing those features, a technique known as "feature engineering". Numerous well-known engineering-based ML techniques, such as Support Vector Machine (SVM), Logical Regression (LR), and KNN, were used to evaluate the efficacy of the proposed method. In

Algorithm 1: FedAvg-HBA for Parameter Optimization of CNN Pseudo-code.

Input $P2$, Rounds R, max_{itr}, $\wp$, N, Number of edge devices
Output weight w.
Function CLOUDSERVER:
Initialize global model (w_0).
for round r ∈ R **do**
  D_r= set of maximum devices.
  for each devices $d \in D$ **do**
    $w^t + 1$=LocalDeviceUpdate($P2$, max_{itr}, $\wp$, N)
  end for
  $w^t + 1$=average the weights
end for
Return w.
Function LocalDeviceUpdate:
Initialize the population N.
for each solution $n \in N$ **do**
  Evaluate using modified LeNet-5.
end for
while $itr \leq max_{itr}$ **do**
  Sort (N).
  Update hb_{prey}
  Update I using Eq.(5)
  Update $\eth$ using Eq(6)
  for each solution $n \in N$ **do**
    Generate flag U
    if $rand() < 0.5$ **then**
      Update hb_{new} using Eq(7)
    else
      Update hb_{new} using Eq(9)
    end if
  end for
  for each solution $n \in N$ **do**
    Evaluate using modified LeNet-5.
  end for
end while Stop criteria satisfied.
Return w.

previous studies, various features that may have provided helpful information for SA identification have been constructed. For the feature engineering-based methods in this study, we used the features (amplitudes: 6 features, RR intervals: 12 features) that substantially influenced SA detection. The HBA parameters are listed in Table II. A swarm's maximum number of iterations is 100, with a maximum size of 10. The $P2$ parameter is 2.0, and the $\wp$ parameter is 6.0. It concludes when the maximum number of iterations is achieved, which serves as the end condition. The code is available in the FL-HBA GitHub [21].

TABLE II
PARAMETERS FOR THE PROPOSED METHOD

ID	Parameter	Value
1	$P2$	2
2	$\wp$	6
3	Number of iterations	100
4	Number of runs	10
5	Population size (N)	10

B. Comparison with existing schemes

Predicting SA using ECG segment is essential in this domain since it establishes a stable foundation for diagnosing patients who may be suffering from SA. For per-segment SA detection, it is worth mentioning that the proposed method is compared with the traditional machine learning and non-HBA methods. As demonstrated in Table III, the whole performance of the evaluation measures was used for comparison, including its Area Under the Curve (AUC), specificity, sensitivity, and accuracy [11]. As seen in Table III, the proposed method with automated feature extraction performed well with an AUC of 96.1%, a specificity of 92.4%, a sensitivity of 85.7%, and an accuracy of 88.92%. The total results increased by 1.1%, 2.1%, 2.6%, and 1.3%, respectively, compared to the second-best performance (i.e., modified LeNet-5). According to the results, KNN exhibited the lowest predictive performance of the six methods, probably because the obtained features from the ECG signal are less spatially related and unsuited for this case. In summary, the proposed method (i.e., FL-HBA) with automated feature extraction outperformed the existing feature engineering method for SA detection.

TABLE III
THE PERFORMANCE OF CONVENTIONAL MACHINE LEARNING IN COMPARISON TO THE PROPOSED METHOD IN THE DETECTION OF SA

Method	AUC	Specificity (%)	Sensitivity (%)	Accuracy (%)
MLP	89.8	87.2	71.3	81.1
KNN	82.6	83.4	68.1	77.5
LR	88.4	84	75.7	80.8
SVM	88.7	84.3	76.9	81.4
LeNet-5	95.0	90.3	83.1	87.6
FL-LeNet-5	93.2	88.8	81.6	85.4
FL-HBA	**96.1**	**92.4**	**85.7**	**88.9**

C. Ten-fold cross-validation

We employed ten-fold cross-validation to establish the consistency of the proposed method based on the PhysioNet Apnea ECG dataset. The entire collection (70 recordings) was randomly divided into ten groups, of which nine were used to train the classifiers (LeNet-5, KNN, LR, SVM, and HBA), and the tenth was used for testing. Fig.3 illustrates the accuracy of six classifiers' SA detection accuracy in ten test groups. As shown in Fig.3 , the accuracy obtained by the LeNet-5, MLP, KNN, LR, SVM, and HBA ranged from 84.2% to 93.7% (standard deviation ∓ mean, 3.05% ∓ 88.7%), 75.4% to 89.9% (standard deviation ∓ mean, 4.98% ∓ 81.9%), 72.5% to 84.8% (standard deviation ∓ mean, 4.53% ∓ 79.3%), 71.7% to 87.8% (standard deviation ∓ mean, 5.47% ∓ 80.6%), 71.9% to 88.6% (standard deviation ∓ mean, 5.50% ∓ 81.1%), and 90.71% to 95.29% (standard deviation ∓ mean, 2.01% ∓ 91.2%), respectively. These results indicate that the FL-HBA with automated feature extraction is very resilient and can achieve consistent and considerably superior outcomes based on the PhysioNet Apnea ECG dataset.

Numerous studies on detecting SA based on a signal from a single lead ECG have been reported thus far, with the majority of these efforts concentrating on feature engineering. As mentioned, the proposed method is compared with prior research that utilized both withheld and published PhysioNet Apnea-ECG dataset. Table IV illustrates the results of SA detection utilizing the same dataset. Again, the same training and testing datasets are used for validation. As proven, the classification existing works accuracy between [83.4%, 87.6%], which is less than the FL-HBA (with an accuracy of 88.9%). In summary, the FL-HBA outperforms the existing methods published in the literature.

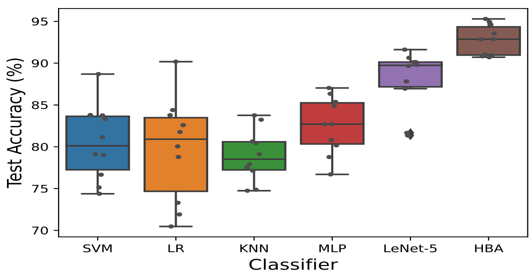

Fig. 3. Results of six classifiers' SA detection accuracy in ten test groups

TABLE IV
PROPOSED METHOD SA DETECTION PERFORMANCE VS. EXISTING WORKS.

Reference	Classifier	Features	Specificity (%)	Sensitivity (%)	Accuracy (%)
[22]	Decision fusion	Auto encoder	88.4	88.9	83.8
[23]	LS-SVM	Feature Engineering	88.4	79.5	83.4
[24]	HMM-SVM	Feature Engineering	88.4	82.6	86.2
[25]	LS-SVM	Feature Engineering	84.7	84.7	84.7
[9]	LeNet-5	CNN	90.3	83.1	87.6
The proposed method	FL-HBA	CNN	**92.4**	**85.7**	**88.9**

V. CONCLUSION

This paper has proposed the FL-HBA method to optimize the hyperparameters of local trained CNN models at the edge devices in an FL environment for SA detection. The results of the experiments show that the FL-HBA is helpful for SA detection and that its efficiency is more suitable than non-HBA and traditional machine learning methods. The proposed method can also be used to produce SA detection for use in home healthcare assistance via wearable devices with improved privacy protection. This is possible because only a trained model of the edge device is required. As a future extension, it is necessary to explore the use of FL-HBA on other CPS and investigate parameters that are related to CNN architecture and FL settings.

ACKNOWLEDGEMENTS

This work was supported by the Machine Learning Department, Mohamed Bin Zayed University of Artificial Intelligence (MBZUAI), UAE under grant no: 8481000021.

REFERENCES

[1] M. Wazzeh, H. Ould-Slimane, C. Talhi, A. Mourad, and M. Guizani, "Privacy-preserving continuous authentication for mobile and iot systems using warmup-based federated learning," *IEEE Network*, 2022.

[2] D. C. Nguyen, Q.-V. Pham, P. N. Pathirana, M. Ding, A. Seneviratne, Z. Lin, O. Dobre, and W.-J. Hwang, "Federated learning for smart healthcare: A survey," *ACM Computing Surveys (CSUR)*, vol. 55, no. 3, pp. 1–37, 2022.

[3] A. Boulemtafes, A. Derhab, and Y. Challal, "Privacy-preserving deep learning for pervasive health monitoring: a study of environment requirements and existing solutions adequacy," *Health and technology*, pp. 1–20, 2022.

[4] A. K. Abasi, M. Aloqaily, M. Guizani, and F. Karray, "Sine cosine algorithm for reducing communication costs of federated learning," in *2022 IEEE International Mediterranean Conference on Communications and Networking (MeditCom)*. IEEE, 2022, pp. 55–60.

[5] S. Banabilah, M. Aloqaily, E. Alsayed, N. Malik, and Y. Jararweh, "Federated learning review: Fundamentals, enabling technologies, and future applications," *Information Processing & Management*, vol. 59, no. 6, p. 103061, 2022.

[6] A. Zarei and B. M. Asl, "Automatic classification of apnea and normal subjects using new features extracted from hrv and ecg-derived respiration signals," *Biomedical Signal Processing and Control*, vol. 59, p. 101927, 2020.

[7] M. M. Lyons, N. Y. Bhatt, A. I. Pack, and U. J. Magalang, "Global burden of sleep-disordered breathing and its implications," *Respirology*, vol. 25, no. 7, pp. 690–702, 2020.

[8] A. H. Khandoker, C. K. Karmakar, and M. Palaniswami, "Automated recognition of patients with obstructive sleep apnoea using wavelet-based features of electrocardiogram recordings," *Computers in Biology and Medicine*, vol. 39, no. 1, pp. 88–96, 2009.

[9] T. Wang, C. Lu, G. Shen, and F. Hong, "Sleep apnea detection from a single-lead ecg signal with automatic feature-extraction through a modified lenet-5 convolutional neural network," *PeerJ*, vol. 7, p. e7731, 2019.

[10] A. K. Abasi, S. N. Makhadmeh, M. A. Al-Betar, O. A. Alomari, M. A. Awadallah, Z. A. A. Alyasseri, I. A. Doush, A. Elnagar, E. H. Alkhammash, and M. Hadjouni, "Lemurs optimizer: A new metaheuristic algorithm for global optimization," *Applied Sciences*, vol. 12, no. 19, p. 10057, 2022.

[11] O. A. Alomari, S. N. Makhadmeh, M. A. Al-Betar, Z. A. A. Alyasseri, I. A. Doush, A. K. Abasi, M. A. Awadallah, and R. A. Zitar, "Gene selection for microarray data classification based on gray wolf optimizer enhanced with triz-inspired operators," *Knowledge-Based Systems*, vol. 223, p. 107034, 2021.

[12] F. A. Hashim, E. H. Houssein, K. Hussain, M. S. Mabrouk, and W. Al-Atabany, "Honey badger algorithm: New metaheuristic algorithm for solving optimization problems," *Mathematics and Computers in Simulation*, vol. 192, pp. 84–110, 2022.

[13] M. Abd Elaziz, A. Mabrouk, A. Dahou, and S. A. Chelloug, "Medical image classification utilizing ensemble learning and levy flight-based honey badger algorithm on 6g-enabled internet of things," *Computational Intelligence and Neuroscience*, vol. 2022, 2022.

[14] D. J. Kapner, T. S. Cook, E. G. Adelberger, J. H. Gundlach, B. R. Heckel, C. Hoyle, and H. E. Swanson, "Tests of the gravitational inverse-square law below the dark-energy length scale," *Physical review letters*, vol. 98, no. 2, p. 021101, 2007.

[15] A. Gaspar, D. Oliva, E. Cuevas, D. Zaldívar, M. Pérez, and G. Pajares, "Hyperparameter optimization in a convolutional neural network using metaheuristic algorithms," in *Metaheuristics in Machine Learning: Theory and Applications*. Springer, 2021, pp. 37–59.

[16] Y.-H. Kwon, S.-B. Shin, and S.-D. Kim, "Electroencephalography based fusion two-dimensional (2d)-convolution neural networks (cnn) model for emotion recognition system," *Sensors*, vol. 18, no. 5, p. 1383, 2018.

[17] S. Kiranyaz, T. Ince, and M. Gabbouj, "Real-time patient-specific ecg classification by 1-d convolutional neural networks," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 3, pp. 664–675, 2015.

[18] T. Penzel, G. B. Moody, R. G. Mark, A. L. Goldberger, and J. H. Peter, "The apnea-ecg database," in *Computers in Cardiology 2000. Vol. 27 (Cat. 00CH37163)*. IEEE, 2000, pp. 255–258.

[19] P. Hamilton, "Open source ecg analysis," in *Computers in cardiology*. IEEE, 2002, pp. 101–104.

[20] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*. PMLR, 2017, pp. 1273–1282.

[21] A. K. Abasi, M. Aloqaily, B. Ouni, and M. Hamdi, 2023. [Online]. Available: https://github.com/Artifitialleap-MBZUAI/Optimization-of-CNN-based-Federated-Learning-for-Cyber-Physical-Detection

[22] K. Li, W. Pan, Y. Li, Q. Jiang, and G. Liu, "A method to detect sleep apnea based on deep neural network and hidden markov model using single-lead ecg signal," *Neurocomputing*, vol. 294, pp. 94–101, 2018.

[23] C. Varon, A. Caicedo, D. Testelmans, B. Buyse, and S. Van Huffel, "A novel algorithm for the automatic detection of sleep apnea from single-lead ecg," *IEEE Transactions on Biomedical Engineering*, vol. 62, no. 9, pp. 2269–2278, 2015.

[24] C. Song, K. Liu, X. Zhang, L. Chen, and X. Xian, "An obstructive sleep apnea detection approach using a discriminative hidden markov model from ecg signals," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 7, pp. 1532–1542, 2015.

[25] H. Sharma and K. Sharma, "An algorithm for sleep apnea detection from single-lead ecg using hermite basis functions," *Computers in biology and medicine*, vol. 77, pp. 116–124, 2016.