IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Maintaining Privacy in Face Recognition using Federated Learning Method

ABRAHAM WOUBIE [1], ENOCH SOLOMON [2] AND JOSEPH ATTIEH [3]

[1]Silo AI, Helsinki, Finland
[2]Virginia State University, Petersburg, Virginia, USA
[3]University of Helsinki, Helsinki, Finland

Corresponding author: Joseph Attieh (joseph.attieh@helsinki.fi)

ABSTRACT

The state-of-the-art face recognition systems are typically trained on a single computer, utilizing extensive image datasets collected from various users. Nevertheless, these datasets often contain sensitive personal information that users may hesitate to disclose. To address potential privacy concerns, we explore the application of federated learning, both with and without secure aggregators, in the context of both supervised and unsupervised face recognition systems. Federated learning facilitates the training of a shared model without necessitating the sharing of individual private data, achieving this by training models on decentralized edge devices housing the data. In our proposed system, each edge device independently trains its own model, which is subsequently transmitted either to a secure aggregator or directly to the central server. To introduce diverse data without the need for data transmission, we employ generative adversarial networks to generate imposter data at the edge. Following this, the secure aggregator or central server combines these individual models to construct a global model, which is then relayed back to the edge devices. Experimental findings based on the CelebA datasets reveal that employing federated learning in both supervised and unsupervised face recognition systems offers dual benefits. Firstly, it safeguards privacy since the original data remains on the edge devices. Secondly, the experimental results demonstrate that the aggregated model yields nearly identical performance compared to the individual models, particularly when the federated model does not utilize a secure aggregator. Hence, our results shed light on the practical challenges associated with privacy-preserving face image training, particularly in terms of the balance between privacy and accuracy.

INDEX TERMS edge computation, federated learning, privacy, secure aggregator, face recognition

I. INTRODUCTION

Face recognition is the process of automatically identifying or verifying the identity of an individual by analyzing facial patterns [1]. This technology has become an integral component in various security and authentication systems, ranging from smartphone unlocking [2] to airport security checks [3]. It encompasses two primary subfields: face identification and face verification. Face identification determines the identity of an individual, whereas face verification confirms or denies a claimed identity [1]. Ensuring accurate face recognition is integral for granting access to services, as permissions should only be accorded following correct identification or verification.

The rapid advancement of machine learning (ML) and the availability of facial datasets have significantly enhanced the accuracy and performance of face recognition systems. Typically, face recognition systems employ machine learning techniques to train deep neural networks using facial data samples. Data samples are commonly gathered on end-devices like smartphones, while the model training takes place on a computationally robust centralized server [4]. This setup raises two major concerns. First, since the model is trained on user face data, it is crucial to prevent unauthorized access or data breaches to protect user privacy. Second, such systems involve a heavy data transmission phase, which can place significant stress on the communication infrastructure. Federated learning offers a solution to both issues. Rather than sending the raw sensitive data to the central server for training the centralized model, federated learning advocates for a distributed training approach [5]. In this setup, each

device maintains its own instance of the model and trains it using its local data [6]. After this local training, only the model updates are transmitted to the central server. The server then aggregates these updates and applies them to the global model [7], [8]. This approach ensures that sensitive facial data remains local, strengthening privacy measures and minimizing data transfers. The most common aggregation strategy is federated averaging, which aggregates the updates using a weighted average [6].

Mobile phones and smart devices are examples of the modern distributed networks that generate huge amounts of data each day [9]. As these devices become more powerful and concerns about data privacy grow, federated learning has emerged as a notable solution to keep data on the device and shift the network's focus to the edge [9]. Various companies have adopted federated learning [7], [10], highlighting its importance in applications that need privacy, especially when training data is spread across devices [11]–[14]. The increasing demand for federated learning across various applications has led to the development of numerous tools, including TensorFlow Federated [15], Federated AI Technology Enabler [16], Leaf [17], and PaddleFL [18]. While privacy-preserving data studies have been of interest since the 1970s, it is only in recent times that they are being extensively employed at a large scale [19]. For instance, Google uses federated learning in Gboard [12] and Android messages [20], and Apple has incorporated it in iOS 13 [21] for features like "Hey Siri" [22].

As mentioned previously, privacy concerns are considered as one of the major challenges in face and speaker recognition systems [23], [24] as these systems usually involve the complete sharing of facial data, which can bring threatening consequences to people's privacy. Federated learning emerges as a promising approach to address these concerns. Unlike conventional methods that require raw data to be sent to a central server for processing, federated learning enables model training directly on the user's device, ensuring that sensitive facial data remains local. This decentralized approach not only enhances privacy but also reduces the need for data transmission, thereby saving bandwidth.

Thus, the main contribution of this work centers on the integration of federated learning techniques in the training of deep neural network-based face recognition classifiers, both supervised and unsupervised, with the primary aim of safeguarding user privacy. In the proposed system, each device independently trains its own model and subsequently transmits this local model to either a secure aggregator or directly to a central server. The secure aggregator, in turn, consolidates these local models originating from various devices, assembles a global model, and dispatches it to the central server. Alternatively, the central server may construct the global model directly, without intermediary interaction with the secure aggregator. Ultimately, the central server redistributes the global model to all participating devices.

The proposed system facilitates the training of a face recognition model grounded in a deep neural network. It accomplishes this by utilizing data stored exclusively on the respective devices, guaranteeing that this data never exits the confines of those devices. The cloud-based component of the system employs federated averaging to merge these local models, thereby forming a global model that is subsequently relayed back to the devices for inference. The implementation of secure aggregation ensures that, at a global level, individual updates from the devices remain completely confidential and inscrutable. As the edge devices solely transmit model updates, no raw data ever departs from the edge. Consequently, the aggregator only has access to a model trained for the purpose of identifying a local user, preserving the privacy of all other information pertaining to face image at the edge.

A second innovation lies in the deployment of a generative adversarial network (GAN) to produce counterfeit data directly on edge devices. Employing a GAN eliminates the necessity of transmitting counterfeit data to the edge or accumulating such data at the edge itself. Transmitting counterfeit data could place a substantial strain on available bandwidth and, more crucially, expose potential vulnerabilities by revealing distinct information about the local user. Conversely, collecting counterfeit data at the edge could prove unfeasible.

The potential applications of the proposed federated learning systems are diverse and could encompass tasks like smartphone-based learning. By collaboratively learning facial characteristics from a multitude of mobile devices, a shared statistical model can be developed to effectively identify individuals. Nevertheless, users might be hesitant to relinquish their data to a central server, driven by concerns about safeguarding their personal privacy. As a solution, federated learning can be employed to train a centralized, user-independent model without the need to expose or share private data.

In the context of smartphone-based learning, a collective approach to learning face image characteristics from a substantial pool of mobile and similar devices enables the development of a unified statistical model for user identification. However, users may understandably harbor reservations about transferring their data to a central server, driven by privacy concerns. As a solution, federated learning can be employed to train a central, user-independent model without compromising the confidentiality of their private data.

In the context of learning across organizations, entities like universities can be likened to remote devices, each housing a wealth of student data. Nevertheless, universities are typically bound by stringent privacy regulations and practices, and any data leakage could lead to legal, administrative, or ethical complications. Federated learning offers a viable solution, allowing for confidential learning to take place across diverse devices and organizations while safeguarding the sensitive data of these institutions.

Our experiments conducted on the CelebA datasets reveal that federated learning brings notable advantages to both supervised and unsupervised facial recognition systems. This is achieved by avoiding the transmission of sensitive user

data to central servers, while still delivering promising results when compared to individual local models. Consequently, the experimental outcomes provide a quantitative understanding of the challenges associated with the practical application of privacy-preserving training for facial recognition. These challenges are particularly evident in the trade-off between privacy and accuracy.

The remainder of this paper is structured as follows. Section III provides a detailed description of the proposed system's architecture. The experimental results are outlined in Section IV, while Section V presents the conclusions drawn from the work.

II. RELATED WORK

Various methodologies have been proposed to enhance the privacy and security of face recognition systems.

A. PRIVACY-PRESERVING FACE RECOGNITION

Various methods have been explored to safeguard facial data. Instead of using real images of individuals' faces, the authors of [25] propose to generate synthetic images by training a class-conditional GAN. The synthetic data generator was trained on the original face dataset and the identities of the individuals as class labels. The authors then generate the synthetic dataset to train the face recognition model. PriFace [26] is another method for privacy-preserving face recognition. PriFace uses locality-sensitive hashing to add randomness to facial data, preventing potential misuse or reconstruction of the images. Further, the work in [27] uses the Householder matrix to protect both model and facial data. This method combines additive and multiplicative perturbations, ensuring efficient user-side computations. For smart home settings, the authors of [28] propose to protect the face feature data of the users using a face recognition approach that combines random matrix and BLS short signature with FaceNet. The work in [29] proposes to protect the privacy of the faces by encrypting them through affine transformation, which consists of permutation, diffusion and shift transformations. Another scheme presented in [30] performs privacy-preserving face recognition scheme in the frequency domain. This scheme integrates an analysis network that gathers components with the same frequency from different blocks and a fast masking method to further secure the remaining frequency components. We can also highlight the work of [31] in which the normalized face feature vectors are encrypted using the CKKS algorithm from the SEAL library. To save computation costs that comes with encryption of query face images, [32] proposes to match an encrypted face query against clustered faces in the repository through a novel multi-matching scheme.

Other studies use local differential privacy to ensure that individual data points cannot be reverse-engineered or identified. The work of [33] proposes a general privacy protection framework for edge-based face recognition systems. This is done through a local differential privacy algorithm based on the proportion difference of feature information. Furthermore, identity authentication and hash technology are used to ensure the legitimacy of the terminal device and the integrity of the face image in the data acquisition phase. The authors of [34] introduce a new privacy-preserving face recognition protocol referred to as Privacy using EigEnface Perturbation (PEEP). This protocol uses local differential privacy to apply perturbation to Eigenfaces. Only the perturbed data is stored in third-party servers, and a standard Eigenface recognition algorithm is run on this data.

B. FEDERATED LEARNING FOR FACE RECOGNITION

Multiple methods use federated learning to ensure privacy-preserving face recognition. The authors of [35] introduce PrivacyFace, which leverages privacy-agnostic clusters during model training. These clusters are indifferent to privacy concerns (i.e., the data in these clusters do not reveal sensitive personal information). PrivacyFace consists of two main components: the Differently Private Local Clustering (DPLC) algorithm, which derives privacy-independent group features, and a consensus-aware face recognition loss that refines the global feature space distribution using these desensitized group features. FedFace [36] presents a federated learning framework that learns from face images across multiple clients without sharing the images with other clients or a central host. Each client, typically a mobile device, contains face images of only its owner. Face Presentation Attack Detection (FedPAD) [37] aims to develop generalized fPAD models while ensuring data privacy. Each data owner trains a local fPAD model, and a server aggregates these models without accessing individual private data. Once the global model is refined, it's used for fPAD inference. FedFR [38] is a federated learning-based framework for privacy-aware generic face representation. The framework optimizes personalized models for clients using the Decoupled Feature Customization module, improving both the global model for face representation and the personalized user model.

III. PROPOSED SYSTEM

The challenge of federated learning revolves around the task of constructing a unified global statistical model using data distributed across a limited number to possibly millions of remote devices. More specifically, the primary objective commonly pursued in federated learning is the minimization of the following objective function:

$$\min_{w} F(w), \text{ where } F(w) := \sum_{k=1}^{m} p_k F_k(w), \qquad (1)$$

where m represents the total number of devices, F_k denotes the local objective function for the kth device, and p_k signifies the relative impact of each device with $p_k \geq 0$ and $\sum_{k=1}^{m} p_k = 1$.

Federated learning empowers the distributed training of face recognition models, accommodating a diverse range of client devices. As illustrated in Fig. 1, the envisioned federated learning system for face recognition functions across

three key locations: edge devices, a secure aggregator, and a central main server. These edge devices encompass a variety of hardware, including mobile phones, laptops, and similar devices. In contrast, the aggregator and main server typically operate as cloud-based services.

Fig. 1 depicts the training of a central model using a distributed dataset. Here, a multitude of nodes, which could represent user devices, possess subsets of data with varying sizes. At the device level, each node computes a model update, which is subsequently conveyed to a central server. During each training iteration, a substantial volume of these updates or gradients is amalgamated at the central server. The central server then derives a global update for the central model by computing the average of these individual local updates.

Note that the architecture of the proposed system remains consistent for both supervised and unsupervised face recognition systems. The key distinction lies in the utilization of labels for training in supervised systems, while unsupervised systems do not rely on labels for training individual face recognition models.

While it is feasible to train individual face recognition models in the supervised system using only client images of a given person on a specific device, our preference is to enhance model robustness and improve the ability to distinguish impostor images. To achieve this, as depicted in Fig. 1, we employ two distinct methods for generating impostor image data for each individual on the edge device:

- In the first method, we randomly select the image of other persons from the CelebA dataset as impostor image data for a given person.
- In the second method, we train a GAN model to generate impostor image data as it is not always easy to find image data of different persons in edge devices. Thus, we use the work of [39] to train a GAN model on the CelebA dataset. Once the impostor images are generated using the trained GAN model, they are combined with client image data to train an individual face recognition model on a specific edge device.

The proposed system employs distributed gradient descent to train a deep neural network across training data residing on user-held devices, with the aim of analyzing the impact of a secure aggregator. In the system that incorporates a secure aggregator, the process unfolds as follows:

1) Local Training: Initially, an individual model is trained locally on each user's device.
2) Model Transmission to Secure Aggregator: Subsequently, each user's device transmits its locally trained model to the secure aggregator.
3) Global Model Creation: The secure aggregator aggregates these individual models to construct a global model.
4) Aggregated Model Transmission: The aggregated model is then sent to the central main server.

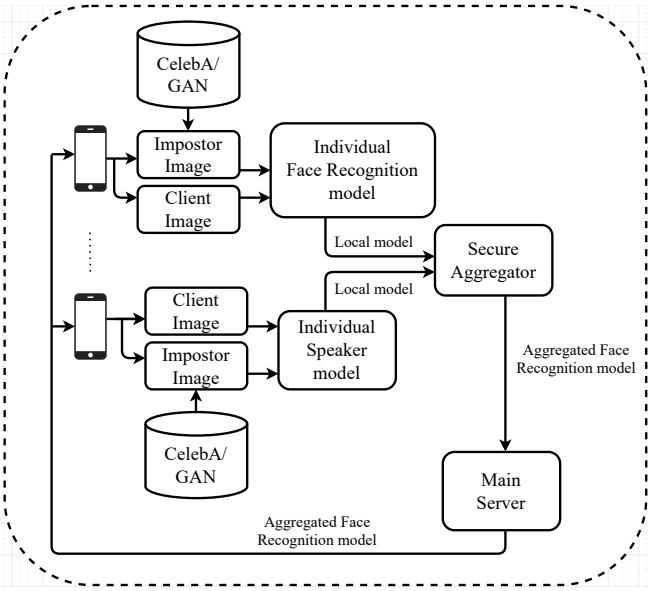


FIGURE 1: The proposed face recognition system incorporates federated learning. Through the implementation of a secure aggregator, we empower a collective of inherently untrusting devices to collaborate and calculate an aggregate value without disclosing their individual private data.

5) Distribution to Devices: Finally, the main server redistributes the global model to each individual device.

In contrast, in the system where a secure aggregator is not utilized, the workflow proceeds as follows:

1) Local Training: Each device independently conducts local training to create an individual model.
2) Model Transmission to Main Server: These individual models are directly transmitted to the central main server.
3) Global Model Creation: The main server combines these individual models to form a global model.
4) Aggregated Model Transmission: The global model is sent back to each individual device for further use and updates.

This dual approach allows for a comparative analysis of the system's performance with and without the incorporation of a secure aggregator.

Privacy serves as a significant driving force behind the adoption of federated learning applications. These systems are designed to safeguard user data by prioritizing the sharing of model updates, such as gradient information, rather than the raw and potentially sensitive data itself. This innovative approach to collaborative machine learning not only enhances data privacy but also enables the collective training of robust and accurate models without exposing individual user information to undue risks or breaches [40]–[42]. While federated learning mitigates some privacy risks by not directly sharing raw data, it's important to recognize that sending model updates during the training process can still pose potential privacy challenges [43]. While recent advancements in federated learning have made strides in enhancing privacy

through tools like secure multiparty computation (SMC) or differential privacy (DP), these approaches have trade-offs between privacy and model performance. The secure aggregator belongs to the class of secure multi-party computation algorithms, where a set of inherently distrustful devices denoted as $d \in U$ individually possess private values x_u. These devices collaborate to calculate an aggregate value, such as the sum $\sum_{u \in U} x_u$, while ensuring that no device discloses any information about its private value to others, except what can be inferred from the resulting aggregate value.

The proposed system aims to uphold the privacy of federated learning by incorporating the use of secure multiparty computation (SMC) techniques. [44], [45]. The adoption of secure multiparty computation serves to safeguard individual model updates, ensuring their privacy and confidentiality. [44]. The central server is unable to observe individual local updates; it can only access the aggregated results at each round.

The proposed work employs the classical federated learning average (FedAvg) [6]. The process involves local optimization executed on participating clients and a subsequent server step to update the global model. Notably, Algorithm 1 illustrates that devices communicate only the updated weights rather than face image data, preserving the security and privacy of the user's facial information locally.

Addressing the transfer of a substantial volume of updated model parameters from users to a server, which is often restricted in throughput [8], [9], [46], [47], poses a significant obstacle in federated learning. This difficulty can be addressed through strategies such as minimizing the number of participating users, achieved through the implementation of scheduling policies [48], [49].

Algorithm 1 In FedAvg, the *C* devices are denoted by index *c*, epochs are indicated by index *e*, and *n* represents the number of face image samples.

0: Initialize w^0
0: **for each** $epoch\ e = 1, 2, ...$ **do**
0: $D \leftarrow (random\ subset\ of\ M\ clients)$
0: **for each** $client\ c \in C$ **do**
0: $\hat{w}_c^e \leftarrow ClientUpdate(c, w^e)$
0: $\Delta w_c^e = w^e - \hat{w}_c^e$
0: **end for**
0: $\bar{w}^e = \sum_{c=1}^{C} \frac{n_c}{n} \Delta w_c^e$ ▷ $weighted\ average$
0: **end for**
0: $w^{e+1} = w^e - \eta \bar{w}^e$ ▷ $Serverupdate$ =0

IV. EXPERIMENTS

A. DATASET

CelebA (Celebrities Attributes Dataset) [50], is a popular dataset in the field of computer vision and machine learning. It was created by researchers at the Chinese University of Hong Kong and is often used for various facial recognition and image analysis tasks. CelebA is known for its large collection of celebrity images and the annotations associated with them. CelebA contains more than 200,000 celebrity images. These images cover a wide range of celebrities from different backgrounds and professions. Each image in the CelebA dataset is annotated with a set of 40 binary attributes. These attributes include characteristics like "smiling," "wearing glasses," "wearing a hat," and so on. These annotations are valuable for tasks like facial attribute prediction and facial attribute manipulation. In addition to attribute annotations, CelebA also provides identity labels for the celebrities in the dataset. This can be useful for tasks involving face recognition. The images in CelebA showcase a wide variety of poses, expressions, lighting conditions, and backgrounds, making it suitable for a broad range of computer vision tasks. The dataset is typically split into training, validation, and test sets to facilitate model training and evaluation.

TABLE 1: The architecture employed for the supervised face verification system..

Layer	Kernel	Filters	Output size
Conv-1	3 X 3	64	350 X 80 X 64
Conv-2	3 X 3	128	175 X 40 X 128
Conv-3	3 X 3	256	87 X 20 X 256
fc-1	-	1000	-
fc-2	-	400	-
fc-3	-	1	-

B. EXPERIMENTAL SETUP

The system architectures are:

- The CNN architecture utilized in our work closely mirrors VGG-M [51], a widely adopted architecture for image classification and speech technology applications [52]. Furthermore, we incorporate a max-pooling layer with dimensions of 2 by 2, along with batch normalization and dropout layers.
  The supervised system has been implemented using the Keras deep learning library [53] to train the model. The network is trained on Titan X GPUs for 100 epochs or until the validation error stops decreasing, whichever is sooner, using a batch-size of 64. We use SGD with momentum (0.9), weight decay (5×10^{-4}) and a logarithmically decaying learning rate (initialised to 10^{-2} and decaying to 10^{-8}).
- An autoencoder is employed to train the unsupervised system with the primary objective of enabling the network to acquire a representation of person-specific facial data. The CNN component, identified as the encoder, is optimized to learn a sophisticated representation of the provided facial image, while the decoder component is fine-tuned to reconstruct the encoder's output into the corresponding facial image. Following the training phase, the decoder component is discarded, and the already learned encoding representation is repurposed for the face verification task. The unsupervised system does not utilize impostor data since its primary focus is on acquiring a compact vector repre-

sentation of distinct individual faces.

The proposed face verification system has been carried out on the following databases namely CelebA [50]. We randomly selected 1000 persons' face images. We allocated 90% of each person's face images for training an individual, face-dependent model, while the remaining 10% was reserved for evaluation. For instance, if a given person had 100 face images in the development set from the database, 90 images were utilized for training the individual face model, and the remaining 10 images were used for evaluation. Additionally, impostor data was introduced into the test set for comprehensive assessment.

Initially, our intention was to train individual face models exclusively using the authentic client face data for each person on every device. However, due to the limited number of files for each individual in the dataset—most individuals having fewer than 100 face images—this approach resulted in an overfitting problem. To address this, we modified our strategy and trained individual face models by incorporating both the true face of the individual and the face images of other individuals as impostor face data.

We adopted two distinct methods to generate impostor face images for each individual device. In the first method, we selected face images of other individuals from each dataset as impostor face images, with 100 samples chosen for each individual on a given device. For the second method, impostor data was created using a GAN model, leveraging the approach outlined in [39] to train the GAN model on each dataset. Similar to the first method, we generated 100 impostor face images for each individual device.

The primary challenge in training the GAN model to generate impostor face images lies in its time-consuming training phase. The computational cost of training the GAN model for 50 hours on the CelebA dataset with a Quadro P2000 GPU amounts to 3.5 hours. However, once the GAN model is trained, the extraction of impostor face image samples on edge devices becomes significantly faster. It's important to note that the training of the GAN model is a one-time task.

The proposed system's performance is assessed using the Equal Error Rate (EER), a metric that measures the point at which the rates of acceptance and rejection errors are equal.

C. EXPERIMENTAL RESULTS

As it is mentioned in Section III, we have analyzed the impact of federated learning both for supervised and unsupervised face verification systems with and without using the secure aggregator. Thus, the experimental results of the supervised and unsupervised systems with and without using the secure aggregator are described below.

1) Supervised Systems without Secure Aggregator

Fig. 3 illustrates the comparative performance of both individual and aggregated models within the supervised system, with and without the utilization of GAN. Notably, the distinction between Fig. 3 (a) and Fig. 3 (b) lies in the method of generating impostor face image samples. In Fig. 3 (a), impostor face image samples are created by selecting face images of different individuals (i.e., extracting face images from CelebA to serve as impostors for a given face image). Conversely, in Fig. 3 (b), the GAN model is employed to generate the impostor face images.

The primary distinction between the individual and aggregated face image models lies in their training approach. For the individual model, a dedicated face image model is initially trained for each specific face image, utilizing the corresponding individual's face image data. Subsequently, the face image samples are assessed using this personalized face image model. In this work, the individual model serves as the baseline system, wherein 1000 individual face image models are trained using face image samples from 1000 devices.

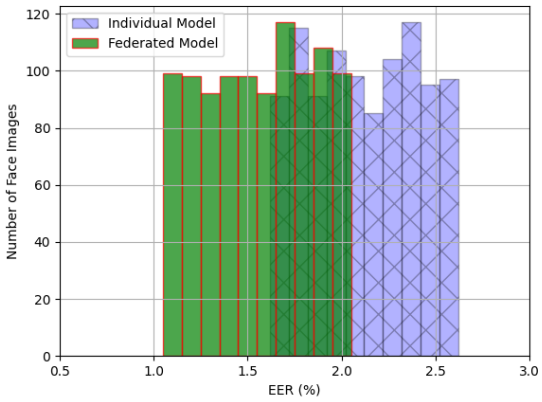
In contrast, the aggregated model employs a collaborative approach. Each of the 1000 devices transmits their parameters to a secure aggregator. The aggregator computes the average of these parameters, establishing them as its updated weight parameters, and then redistributes them to the 1000 devices. Consequently, this collaborative model is referred to as the aggregated (federated) face image model.

In Fig. 3, it is evident that when each of the 1000 devices employs its own individual model, the Equal Error Rate (EER) surpasses 1.98. However, with the utilization of federated/aggregated models, irrespective of the impostor generation method, a majority of the devices exhibit EER values below 1.98. Specifically, employing the first method, which uses the face images of other individuals as impostor data, around 324 devices yield an EER below 1.98. Meanwhile, in the second aggregation method involving GAN-generated impostor face images, a similar number of devices achieve an EER below 1.98.

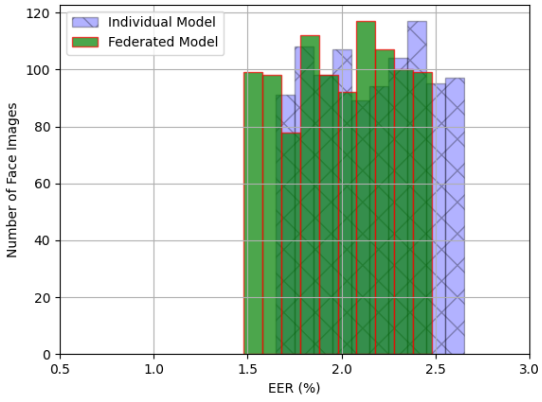
In both Fig.3 (a) and Fig.3 (b), it is noticeable that the aggregated face image model consistently outperforms the individual model in terms of average EER, regardless of whether GAN or other persons' face images are used as impostor data. These figures also indicate that the two aggregated methods yield nearly identical average EER values. This suggests the feasibility of employing GAN for on-device generation of impostor face images, eliminating the need to transfer impostor data from external sources to edge devices.

Table 2 presents a comprehensive overview of the results. The average EER of individual models across the 1000 devices/face images in the supervised face image verification system stands at 2.11. This average EER serves as the baseline system, calculated by utilizing data trained specifically for each face image/device.

Additionally, Table 2 highlights that the average EER for the 1000 devices under the federated model, without employing a secure aggregator and GAN, is 1.55. This represents a noteworthy 26.5% relative improvement compared to the baseline systems. Similarly, leveraging GAN to generate impostor data during the training of face image models yields superior EER results compared to the baseline system. The table illustrates that the federated model utilizing GAN-



(a) Impostors selected from CelebA dataset.



(b) Impostors generated using GAN.

FIGURE 2: Histograms depicting the Equal Error Rate (EER) across 1000 devices are presented for the comparison between individual and federated models in the supervised systems. Notably, this evaluation focuses on models that do not utilize a secure aggregator (SA).

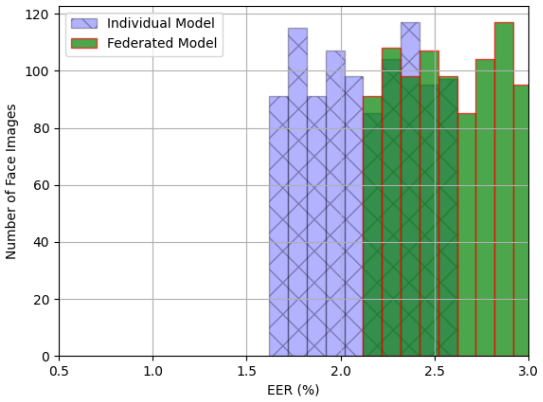
generated data achieves an average EER of 1.98, reflecting a 6

2) Supervised Systems using Secure Aggregator

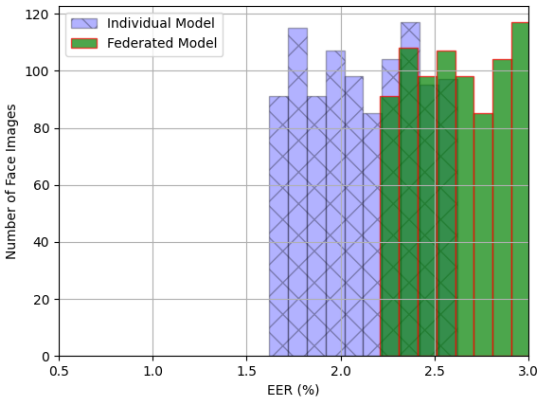
Table 2 reveals that the average Equal Error Rate (EER) of the 1000 devices within the federated model of the supervised system, incorporating both a secure aggregator and impostor face images from the CelebA dataset, is 2.61. Similarly, the average EER for the 1000 devices in the federated model of the same system, employing the GAN technique to generate impostor face images, is 2.73. These findings suggest that, regardless of the impostor generation method, the inclusion of a secure aggregator leads to inferior results compared to both individual systems and federated systems that do not involve a secure aggregator.

The decline in EER when employing a secure aggregator in the federated system can be attributed to the trade-off between privacy enhancement and model performance or system efficiency. While recent approaches aim to bolster the privacy of federated learning through secure aggregation, this often comes at the expense of reduced model performance or overall system efficiency. Consequently, it becomes essential to weigh the privacy aspect alongside the EER values. Nevertheless, the results from systems incorporating secure aggregators remain acceptable despite the observed trade-offs.

In Figure 4, the distribution of Equal Error Rate (EER) among the 1000 devices is presented for both individual and aggregated models within the supervised system, with and without the inclusion of a secure aggregator. The figure also highlights the influence of using Generative Adversarial Networks (GAN) for impostor face image generation. The depicted elements include the minimum, lower quartile, median, upper quartile, and maximum EER values.



(a) Impostors selected from CelebA dataset.



(b) Impostors generated using GAN.

FIGURE 3: Histograms illustrating the Equal Error Rate (EER) distribution across 1000 devices are provided for a comparison between individual and federated models in the supervised system. This analysis specifically considers models that incorporate a secure aggregator (SA).

As evident from the figure, aggregated models, particularly

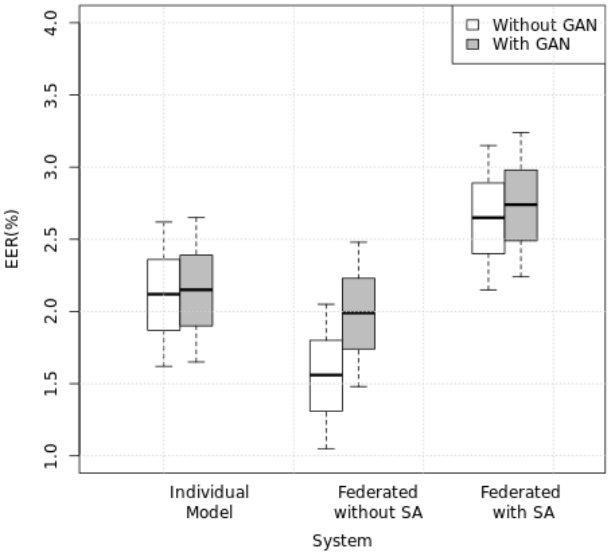


FIGURE 4: The box plot depicts the distribution of Equal Error Rates (EER) for both supervised individual and federated models across 1000 devices. The analysis considers scenarios both with and without using a Secure Aggregator (SA). Additionally, the influence of impostor selections, with and without the incorporation of Generative Adversarial Network (GAN), is highlighted.

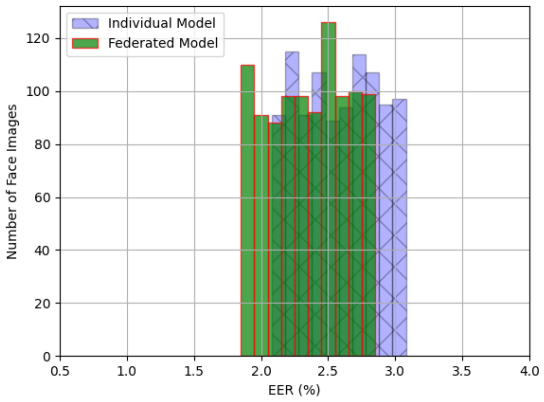
TABLE 2: The table provides a comparison of Equal Error Rates (EER) for supervised face verification systems, considering both individual and federated approaches, with and without the use of a Secure Aggregator (SA). The inclusion of GAN-generated data for impostor face images is also accounted for in the comparison.

System	With SA	Without SA
Individual Model	2.11	
Federated Model without GAN	2.61	1.55
Federated Model with GAN	2.73	1.98

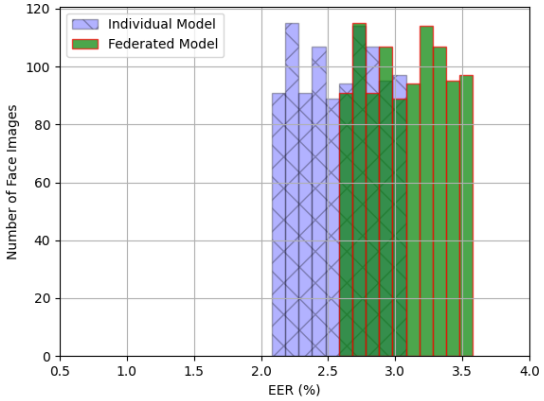
those not incorporating a secure aggregator, consistently outperform individual models in terms of average EER, regardless of the impostor generation method employed. The visual representation of EER distribution provides a clear indication of the superior performance of aggregated models, reinforcing the efficacy of collaborative approaches in contrast to individual models within the supervised system.

3) Unsupervised Systems

In Fig. 5, the Equal Error Rates (EERs) are presented for both individual and aggregated face image models within the unsupervised system, with and without the implementation of a secure aggregator. As depicted, the EER for all 1000 devices exceeds 2.57 when each device utilizes its individual model. However, when the federated model does not employ a secure aggregator, as shown in Fig. 5 (a), approximately 680 devices achieve an EER below 2.35.



(a) Without using a secure aggregator (SA).



(b) Using a secure aggregator.

FIGURE 5: The Equal Error Rate (EER) across 1000 devices is reported for the comparison between the individual and federated models in the unsupervised system.

This visual representation underscores a notable improvement in EER when transitioning from individual models to federated models without a secure aggregator in the unsupervised system. The collaborative approach appears to enhance the performance of the face image models, contributing to lower EER values for a significant portion of the devices.

In contrast to the results depicted in Fig.5 (b), the use of a secure aggregator results in a deterioration of Equal Error Rates (EERs), leading to inferior results when compared to the individual models. This observation highlights a significant discrepancy in performance when incorporating a secure aggregator within the unsupervised system. The visual representation in Fig. 5 (b) underscores the importance of carefully evaluating the influence of a secure aggregator on EER results, revealing a potential trade-off between privacy-enhancing measures and model performance in the context of the unsupervised system.

Table 3 displays the average Equal Error Rate (EER) of individual models across the 1000 persons within the

TABLE 3: The Equal Error Rate (EER) is compared between the unsupervised face verification systems, considering both individual and federated approaches, with and without the use of a Secure Aggregator (SA).

System	With SA	Without SA
Individual Model	2.57	
Federated Model	3.07	2.35

unsupervised face verification system, amounting to 2.57. In contrast, the table reveals that the average EER for the 1000 devices in the federated model, under the same system but without a secure aggregator, is 2.35. This signifies an 8.56% relative improvement in EER compared to the baseline unsupervised system. However, it is noteworthy that the inclusion of a secure aggregator in the federated model results in a worse outcome compared to the baseline system, as indicated in the table.

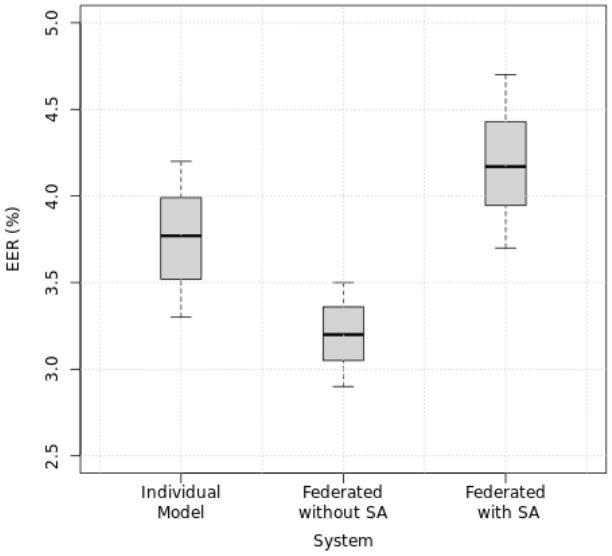


FIGURE 6: A box plot is presented to illustrate the distribution of Equal Error Rates (EER) for unsupervised individual and federated models across 1000 devices. The analysis encompasses scenarios both with and without the utilization of a Secure Aggregator (SA).

D. DISCUSSIONS

The results presented in Table 2 and 3 consistently highlight that, regardless of the face verification system (supervised or unsupervised), the introduction of a secure aggregator tends to reduce the performance of the federated model. Conversely, when the federated model operates without a secure aggregator, EER results improve in comparison to the individual models. Despite the slight deterioration caused by the secure aggregator, it's essential to consider that its inclusion ensures the privacy of the data.

It's worth noting that, while the EER results experience a slight decline with the secure aggregator, the overall performance remains satisfactory. The compromise in EER is balanced by the privacy-preserving benefits offered by the secure aggregator. The EER results, even with the use of a secure aggregator, are acceptable, emphasizing the trade-off between privacy protection and model performance.

In addition to the individual and federated models, another experiment was conducted by pooling all face image samples from the 1000 persons and training a single generic face image model on a single computer. The results demonstrate that the average EER of the global model on the supervised and unsupervised face verification systems is 1.2% and 2.2%, respectively. These results are comparable to the federated model's performance (see Fig. 4 and 6). The federated model achieves similar EER values as the global model while preserving the privacy of face image data. This underscores the advantage of using federated learning for face recognition systems.

The work employs 1000 devices to compare the performance of individual versus federated models. Statistical analysis using Student's t-test supports the significance of the observed differences. The computed P-values for both comparisons, where the federated model selects impostor face images from CelebA (federated model 1) and where GAN is used for impostor face image generation (federated model 2), are both less than the standard significance level of 0.05. Thus, we reject the null hypothesis, affirming that the mean EER differences between individual and federated models are statistically significant.

Finally, the experiment considered the impact of updating local models more than once. The results indicate that updating local models more than once does not lead to an improvement in EER. This could be attributed to the similarity in training data among devices during each training phase. Although updating more frequently did not yield enhanced performance, this decision was driven by the need to maintain data privacy.

V. CONCLUSIONS

In this work, we propose the adoption of federated learning as a safeguard for the privacy of facial image data residing on edge devices, applicable to both supervised and unsupervised face recognition systems. Our approach centers on decentralized training, eliminating the need for devices to transmit their raw image data to centralized servers. Instead, each user's data remains securely stored and processed solely on their respective edge device. Consequently, training occurs exclusively at the local level, with each device contributing updates to a central model. Subsequently, a secure aggregator consolidates these local models into a single federated model, which is then distributed via the main server back to the individual devices. Furthermore, our research delves into an analysis of the influence of the secure aggregator on the performance of face recognition systems.

Our proposed system offers two primary advantages.

Firstly, as raw data remains confined to individual devices, the privacy of facial images is preserved. Secondly, experimental findings reveal that the federated model, devoid of a secure aggregator, achieves a superior average Equal Error Rate (EER) compared to individual models. However, when the federated model incorporates the secure aggregator, the aggregated model yields EER results that are slightly less favorable than those of individual models. Nonetheless, the EER results remain commendable, emphasizing the importance of weighing the trade-offs between privacy and performance.

Future research works should delve into refining aggregation techniques beyond simplistic averaging methods. Additionally, exploring the effects of scaling up the number of devices beyond the 1000 devices employed in this work holds promise for further enhancing the effectiveness of privacy-preserving face recognition systems.

REFERENCES

[1] Xinyi Wang, Jianteng Peng, Sufang Zhang, Bihui Chen, Yi Wang, and Yandong Guo. A survey of face recognition, 2022.

[2] Guillaume Dave, Xing Chao, and Kishore Sriadibhatla. Face recognition in mobile phones. Department of Electrical Engineering, Stanford University, 01 2010.

[3] Jyri Rajamäki, Tuomas Turunen, Aki Harju, Miia Heikkilä, Maarit Hilakivi, Sami Rusanen, and Laurea Leppävaara. Face recognition as an airport and seaport security tool. 6, 07 2009.

[4] Jiasi Chen and Xukan Ran. Deep learning with edge computing: A review. Proceedings of the IEEE, 107(8):1655–1674, 2019.

[5] Jeffrey Dean, Greg S Corrado, Rajat Monga, Kai Chen, Matthieu Devin, Quoc V Le, Mark Z Mao, Marc'Aurelio Ranzato, Andrew Senior, Paul Tucker, et al. Large scale distributed deep networks. 2012.

[6] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In Artificial Intelligence and Statistics, pages 1273–1282. PMLR, 2017.

[7] Keith Bonawitz, Hubert Eichner, Wolfgang Grieskamp, Dzmitry Huba, Alex Ingerman, Vladimir Ivanov, Chloe Kiddon, Jakub Konečnỳ, Stefano Mazzocchi, H Brendan McMahan, et al. Towards federated learning at scale: System design. arXiv preprint arXiv:1902.01046, 2019.

[8] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Keith Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. arXiv preprint arXiv:1912.04977, 2019.

[9] Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. Federated learning: Challenges, methods, and future directions. IEEE Signal Processing Magazine, 37(3):50–60, 2020.

[10] WA Group et al. Federated learning white paper v1. 0, 2020.

[11] Mingkai Huang, Hao Li, Bing Bai, Chang Wang, Kun Bai, and Fei Wang. A federated multi-view deep learning framework for privacy-preserving recommendations. arXiv preprint arXiv:2008.10808, 2020.

[12] Andrew Hard, Kanishka Rao, Rajiv Mathews, Swaroop Ramaswamy, Françoise Beaufays, Sean Augenstein, Hubert Eichner, Chloé Kiddon, and Daniel Ramage. Federated learning for mobile keyboard prediction. arXiv preprint arXiv:1811.03604, 2018.

[13] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. ACM Transactions on Intelligent Systems and Technology (TIST), 10(2):1–19, 2019.

[14] Yang Zhao, Jun Zhao, Linshan Jiang, Rui Tan, and Dusit Niyato. Mobile edge computing, blockchain and reputation-based crowdsourcing iot federated learning: A secure, decentralized and privacy-preserving system. 2020.

[15] The TFF Authors. TensorFlow Federated. https://www.tensorflow.org/federated/, 2019. Accessed: 2021-09-20.

[16] The FATE Authors. Federated AI Technology Enabler. URLhttps://www.fedai.org/, 2019. Accessed: 2021-09-20.

[17] The Leaf Authors. Leaf. https://leaf.cmu.edu/, 2019. Accessed: 2021-09-20.

[18] The PaddleFL Authors. PaddleFL. https://github.com/PaddlePaddle/PaddleFL/, 2019. Accessed: 2021-09-20.

[19] Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. Rappor: Randomized aggregatable privacy-preserving ordinal response. In Proceedings of the 2014 ACM SIGSAC conference on computer and communications security, pages 1054–1067, 2014.

[20] support.google. Your chats stay private while Messages improves suggestions. https://support.google.com/messages/answer/9327902/, 2019. Accessed: 2021-09-20.

[21] Apple. Private Federated Learning (NeurIPS 2019 Expo Talk Abstract). https://nips.cc/ExpoConferences/2019/schedule?talk_id=40, 2019. Accessed: 2021-09-20.

[22] Apple. Designing for privacy. https://developer.apple.com/videos/play/wwdc2019/708/, 2019. Accessed: 2021-09-20.

[23] Yogachandran Rahulamathavan, Kunaraj R Sutharsini, Indranil Ghosh Ray, Rongxing Lu, and Muttukrishnan Rajarajan. Privacy-preserving ivector-based speaker verification. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 27(3):496–506, 2018.

[24] Abraham Woubie and Tom Bäckström. Federated learning for privacy-preserving speaker recognition. IEEE Access, 9:149477–149485, 2021.

[25] Fadi Boutros, Marco Huber, Patrick Siebke, Tim Rieber, and Naser Damer. Sface: Privacy-friendly and accurate face recognition using synthetic data. In 2022 IEEE International Joint Conference on Biometrics (IJCB), pages 1–11, 2022.

[26] S. Zhao, L. Zhang, and P. Xiong. Priface: a privacy-preserving face recognition framework under untrusted server. J Ambient Intell Human Comput, 14:2967–2979, 2023.

[27] Wenjing Gao, Jia Yu, Rong Hao, Fanyu Kong, and Xiaodong Liu. Privacy-preserving face recognition with multi-edge assistance for intelligent security systems. IEEE Internet of Things Journal, 10(12):10948–10958, 2023.

[28] Xiaoyu Kou, Ziling Zhang, Yuelei Zhang, and Linlin Li. Efficient and privacy-preserving distributed face recognition scheme via facenet. In Proceedings of the ACM Turing Award Celebration Conference - China, ACM TURC '21, page 110–115, New York, NY, USA, 2021. Association for Computing Machinery.

[29] Shangwei Guo, Tao Xiang, and Xiaoguo Li. Towards efficient privacy-preserving face recognition in the cloud. Signal Processing, 164:320–328, 2019.

[30] Yinggui Wang, Jian Liu, Man Luo, Le Yang, and Li Wang. Privacy-preserving face recognition in the frequency domain. Proceedings of the AAAI Conference on Artificial Intelligence, 36(3):2558–2566, Jun. 2022.

[31] Y. Yang, Q. Zhang, W. Gao, et al. Design on face recognition system with privacy preservation based on homomorphic encryption. Wireless Pers Commun, 123:3737–3754, 2022.

[32] Meng Liu, Hongsheng Hu, Haolong Xiang, Chi Yang, Lingjuan Lyu, and Xuyun Zhang. Clustering-based efficient privacy-preserving face recognition scheme without compromising accuracy. ACM Trans. Sen. Netw., 17(3), jun 2021.

[33] Y. Xie, P. Li, N. Nedjah, et al. Privacy protection framework for face recognition in edge-based internet of things. Cluster Comput, 26:3017–3035, 2023.

[34] M.A.P. Chamikara, P. Bertok, I. Khalil, D. Liu, and S. Camtepe. Privacy preserving face recognition utilizing differential privacy. Computers Security, 97:101951, 2020.

[35] Qiang Meng, Feng Zhou, Hainan Ren, Tianshu Feng, Guochao Liu, and Yuanqing Lin. Improving federated learning face recognition via privacy-agnostic clusters, 2022.

[36] Divyansh Aggarwal, Jiayu Zhou, and Anil K. Jain. Fedface: Collaborative learning of face recognition model, 2021.

[37] Rui Shao, Pramuditha Perera, Pong C. Yuen, and Vishal M. Patel. Federated face presentation attack detection, 2020.

[38] Chih-Ting Liu, Chien-Yi Wang, Shao-Yi Chien, and Shang-Hong Lai. Fedfr: Joint optimization federated framework for generic and personalized face recognition. Proceedings of the AAAI Conference on Artificial Intelligence, 36(2):1656–1664, Jun. 2022.

[39] Chris Donahue, Julian McAuley, and Miller Puckette. Adversarial audio synthesis. In ICLR, 2019.

[40] Nicholas Carlini, Chang Liu, Jernej Kos, Úlfar Erlingsson, and Dawn Song. The secret sharer: Measuring unintended neural network memorization & extracting secrets. arXiv preprint arXiv:1802.08232, 5, 2018.

[41] Martin J Wainwright, Michael Jordan, and John C Duchi. Privacy aware learning. Advances in Neural Information Processing Systems, 25, 2012.

[42] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. Found. Trends Theor. Comput. Sci., 9(3-4):211–407, 2014.
[43] H Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. Learning differentially private recurrent language models. arXiv preprint arXiv:1710.06963, 2017.
[44] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for privacy-preserving machine learning. In proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, pages 1175–1191, 2017.
[45] Badih Ghazi, Rasmus Pagh, and Ameya Velingker. Scalable and differentially private distributed aggregation in the shuffled model. arXiv preprint arXiv:1906.08320, 2019.
[46] H Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. ArXiv e-prints, pages arXiv–1602, 2016.
[47] Mingzhe Chen, Zhaohui Yang, Walid Saad, Changchuan Yin, H Vincent Poor, and Shuguang Cui. A joint learning and communications framework for federated learning over wireless networks. IEEE Transactions on Wireless Communications, 2020.
[48] Howard H Yang, Zuozhu Liu, Tony QS Quek, and H Vincent Poor. Scheduling policies for federated learning in wireless networks. IEEE transactions on communications, 68(1):317–333, 2019.
[49] Mohammad Mohammadi Amiri, Deniz Gündüz, Sanjeev R Kulkarni, and H Vincent Poor. Update aware device scheduling for federated learning at the wireless edge. In 2020 IEEE International Symposium on Information Theory (ISIT), pages 2598–2603. IEEE, 2020.
[50] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild, 2015.
[51] Ken Chatfield, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Return of the devil in the details: Delving deep into convolutional nets. arXiv preprint arXiv:1405.3531, 2014.
[52] Joon Son Chung and Andrew Zisserman. Out of time: automated lip sync in the wild. In Asian conference on computer vision, pages 251–263. Springer, 2016.
[53] François Chollet et al. Keras (2015), 2017.

ABRAHAM WOUBIE completed his Bachelor and Master of Science Degrees in Computer Science with very great distinctions from Arba Minch and Addis Ababa Universities in Ethiopia, respectively. He completed his Ph.D. in Signal Theory and Communications from the Technical University of Catalonia in Barcelona, Spain, in September 2017. He also undertook two post-doctoral research positions at University of Eastern Finland and Aalto University. Currently, he holds the position of Senior AI Scientist at Silo AI in Helsinki, Finland. His research interests include deep reinforcement learning, speech signal processing, machine learning and image processing. He is the author of papers on these topics.

ENOCH SOLOMON completed his Bachelor, Master and PhD degrees in Computer Science from Arba Minch University, Maharishi University of Management and Virginia Commonwealth University, respectively. He is currently an assistant professor at Virginia State University. He has substantial industry expertise in the fields of data science and machine learning. His research interests are in the area of Computer Vision, AI for Cybersecurity, and Natural Language Processing and understanding.

JOSEPH ATTIEH completed his Bachelor's in Computer Engineering with high distinction at the Lebanese American University. He then pursued a Master of Science in Security, Communication Systems, and Machine Learning, earning high distinction in an Erasmus Mundus joint master's program at Aalto University, Finland, and The Royal Institute of Technology (KTH) in Sweden. Currently, he is pursuing his Doctoral degree in Language Technology at the University of Helsinki. His research focuses on natural language processing, modularization, and parameter-efficient learning methods.