A Federated Learning Approach for Authentication and User Identification based on Behavioral Biometrics

Rafael Veiga[1], Cristiano B. Both[2], Iago Medeiros[1], Denis Rosário[1], Eduardo Cerqueira[1]

[1]Federal University of Pará (UFPA), Belém – Brazil

[2]University of Vale do Rio dos Sinos (UNISINOS), São Leopoldo – Brazil

rafael.teixeira.silva@icen.ufpa.br, cbboth@unisinos.br,

iago.medeiros@itec.ufpa.br, {cerqueira, denis}@ufpa.br

Abstract. *A smartphone can collect behavioral data without requiring additional actions on the user's part and without the need for additional hardware. In an active or continuous user authentication process, information from integrated sensors, such as touch, and gyroscope, is used to monitor the user continuously. These sensors can capture behavioral (touch patterns, accelerometer) or physiological (fingerprint, face) data of the user naturally interacting with the device. However, transferring data from multiple users' mobile devices to a server is not recommended due to user data privacy concerns. This paper introduces an Federated Learning (FL) approach to define a user's biometric behavior pattern for continuous user identification and authentication. We also evaluate whether FL can be helpful in behavioral biometrics. Evaluation results compare CNNs in different epochs using FL and a centralized method with low chances of wrong predictions in user identification by the gyroscope.*

1. Introduction

Mobile devices, such as smartphones and tablets, are part of the population's daily life, and their use and importance have grown globally [Li et al. 2020a]. These devices store private and sensitive information, such as financial, personal, professional, and social data. In this context, a growing demand emerges to provide better security on mobile devices. For example, current authentication systems started to be based on "something you are" rather than "something you have" (cards or keys) or "something you know" (password) since cards, keys, or passwords can be lost, stolen, discovered, or copied [Barros et al. 2020]. In this way, the behavioral biometric approach aims to map unique behavioral characteristics along the interaction of a person with their mobile device, which can be used for human identification and authentication [Stylios et al. 2021].

It is fundamental, in this context, to collect data from the various sensors of a mobile device to define the biometric behavior of a given user. For instance, an accelerometer, light sensor, orientation sensor, magnetometer, gyroscope, Global Positioning System (GPS), proximity, and touch sensor are used without requiring extra user actions and the need for additional hardware [Pires et al. 2020]. Specifically, these sensors relate to something the user regularly does, such as typing, walking, using a mobile application, etc. In this way, the data generated by the sensors can be used to predict the occurrence of events

that tend to repeat themselves over time, where changes in this context may reflect device misuse [Acien et al. 2020]. Therefore, it is possible to identify patterns of biometric and behavioral profiles without users needing to remember or possess any object at the time of authentication [Kandar et al. 2021].

Traditional authentication approaches assume single events, such as entering a password. However, the security level is reduced over time in authenticated sessions [Barros et al. 2020]. In this way, it is fundamental to offer solutions that treat authentication as a continuous process and not based on isolated events. Therefore, active or ongoing user authentication can be based on behavioral biometrics, where information from built-in sensors is used to identify the user continuously [Dahia et al. 2020]. In this context, the classification of behavioral biometrics for continuous authentication is traditionally based on Machine Learning (ML) methods, where relevant user characteristics are sent to a central server for processing and classification.

Data privacy is violated during the execution, interpretation, error calculations, and weight readjustments, considering a conventional ML approach to identify behavioral biometrics. Usually, this violation is because the entire process takes place centrally, *i.e.*, not guaranteeing user privacy. In addition, current policies for protecting private data have become increasingly stringent, setting limits on obtaining knowledge of user data and determining responsibilities related to security and data access control. Moreover, transferring raw data from multiple users to a central server to identify behavioral biometrics for continuous user authentication could violate user privacy [Zhu et al. 2019]. The new personal data protection laws define the rights of data subjects and duties to institutions holding such data. A good example is the General Data Protection Regulation (GDPR), an European Union law on data protection and privacy.

Federated Learning (FL) paradigm enables the preservation of privacy by processing the data on the client to deal with such data privacy issues where this data is generated [Lobato et al. 2022]. Specifically, FL considers distributed ML process by allowing mobile devices (*i.e.*, clients) to train models locally. The locally trained models' ML parameters (*i.e.*, Neural Network Weight) are sent to a central server that aggregates the parameters of different clients under a given aggregation policy (*e.g.*, average). Afterward, the server returns to the clients a consolidated model, which is retrained with local data by the clients participating in the federation. In this context, clients do not share their private raw data with a centralized server, reducing data transmission between the client and a server and mitigating user privacy issues [Barros et al. 2021]. The user data is kept private while, indirectly, it is used to build improved models. The authentication based on FL improves performance, allowing users to be more secure. Discreteness in user identification and authentication is a fundamental point in a platform that provides a protection system. However, authentication systems can be inappropriate and ineffective, causing problems such as error authentication, other users recognized at the wrong moments, and other inconveniences [Liang et al. 2020].

In this paper, we define a user's biometric behavior pattern using a FL approach for continuous user identification and authentication. To this end, we considers gyroscope data to extract representative features with high discriminability related to user behavior. We also examine and evaluate whether FL can be helpful in behavioral biometrics. In our evaluation, we consider gyroscope data from a BrainRun database with a large volume collected (26.7 GB) to extract representative features related to user behavior with

high discriminability. We examine two types of Convolutional Neural Networks (CNNs), *i.e.*, ResNet [Wang et al. 2017] and Inception [Ismail Fawaz et al. 2020], with different configurations to compare their performance in a context of FL approach for continuous authentication. Evaluation results compare CNNs in different epochs using FL and a centralized method with low chances of wrong predictions in user identification by the gyroscope.

This paper is organized as follows. Section 2 presents an overview of works that explore similar proposals related to user authentication and identification. Section 3 describes our methodology for data analysis and the tools we used for this paper. Section 4 explores the simulation model and obtained results related to user authentication and identification. Finally, Section 5 concludes the paper with final remarks and directions for future work.

2. Related Work

Several studies demonstrate the use of behavioral biometrics for continuous user authentication, which employ user behavior characteristics such as walking mode, screen touch operations, hand gestures, and keyboard usage patterns to generate behavioral characteristics. Such data are obtained naturally from the movement, orientation, and position of a device considering the surrounding environment. In this context, Liang *et al*. bring a systematic and comprehensive review, discussing challenges and future directions to guide further studies of continuous authentication and artificial intelligence [Liang et al. 2020]. The authors present a framework for developing innovative approaches to ongoing user authentication and identification, defining the overall workflow.

Mekruksavanich and Jitpattanakul introduced a new continuous authentication framework called DeepAuthen, which identifies mobile device users based on their physical activity patterns as measured by the accelerometer, gyroscope, and magnetometer sensors on their mobile devices [Mekruksavanich and Jitpattanakul 2021]. In the study, the authors used three sets of reference data and demonstrated that the framework could accurately authenticate users using a variety of sensors. Li *et al*. also used deep learning techniques in a continuous authentication system called SCANet, based on a smartphone accelerometer and gyroscope, to monitor user behavior patterns [Li et al. 2020b]. The SCANet system was composed of five modules: data collection, pre-processing, feature extraction, classification, and authentication, and the experimental results showed an accuracy of 90.04%.

Deep learning was also applied by Nemes and Antal by employing supervised and unsupervised approaches to gait recognition based on learning accelerometer features [Nemes and Antal 2021]. The results showed that autoencoders have a feature learning ability close to discriminative models. Moreover, convolutional models have high learning ability, regardless of the training strategy. Another important study was conducted by Espín López *et al*., presenting the S3 platform, *i.e.*, a continuous authentication system based on artificial intelligence that combines sensor data, application statistics, and voice to authenticate users on smartphones [Espín López et al. 2021]. Lin *et al*. introduced a feature called DeFFusion, a continuous authentication system based on accelerometer and gyroscope data [Li et al. 2021b]. The authors evaluated the system authentication performance by considering an experimental environment.

Regarding using FL in a continuous authentication system, it is possible to men-

tion the work done by Li *et al.* and Ek *et al*. First, Li *et al.* designed the Meta-HAR framework for human activity recognition in this work using FL [Li et al. 2021a]. The results showed that Meta-HAR efficiently obtains and maintains high test accuracy values, considering individual users, significantly outperforming oou ther models and even centralized learning in some instances. Ek *et al*. also evaluated FL for human activity recognition, presenting several experiments to analyze the performance of FL compared to a centralized training approach [Ek et al. 2020]. The experiments showed different behaviors considering the same FL algorithm but with other datasets and models.

We can confirm from analyzed studies that using a continuous authentication method based on behavioral characteristics is applicable and can obtain satisfactory results. However, we can observe that the existing studies have some limitations, such as rely on centralized ML models. In this context, it is important to protecting individual user data stored in applications, where FL enables a balanced approach to privacy maintenance and data utilization.

3. Behavioral Biometrics for Authentication and User Identification using Federated Learning

Behavioral biometrics is an approach that refers to unique behavioral characteristics in the interaction of a person and their mobile device, which can be used for human identification and authentication [Stylios et al. 2021]. Therefore, this section presents a FL approach for authentication and user identification based on behavioral biometrics. To this end, we introduce the modules and the implementation details for obtaining behavioral biometrics through FL for authentication and user identification.

3.1. Scenario Overview for User Identification

The behavioral biometrics recognition process has the following steps: data acquisition, pre-processing, feature extraction, classification, and authentication [Dahia et al. 2020]. The data acquisition step has input from the various sensors that integrate a mobile device, such as an accelerometer, light sensor, orientation sensor, magnetometer, gyroscope, GPS, proximity sensor, touch, etc. These sensors provide information related to something the user repeatedly does, such as typing, walking, using an app, etc. On modern devices, biometrics registration is based on information provided by APIs on mobile device operating systems. This data goes through a pre-processing step, where outliers are removed.

After pre-processing the data, extracting relevant features from the raw data is necessary. Based on this information, it is possible to identify a biometric pattern of the user, where characteristics are considered and captured discretely, representing natural movements and the activities continuous by the user. For example, a touch on a modern smartphone screen generates multiple interaction events throughout this touch. The first event is when the finger touches the screen, accompanied by several others produced while the finger is still in contact with the screen. These collected characteristics are used for the classification pattern to distinguish genuine and imposter behavioral biometrics data.

We considered a generic FL architecture for processing the behavioral biometrics data. Specifically, FL allows mobile devices (*i.e.*, clients) to train ML models to extract features without sharing their private data with a centralized server. In this context, a model is trained on client devices using each user's private data, and only the update of

a local model is sent back to a server. Figure 1 shows the FL approach to define a user's biometric behavior pattern for continuous user identification and authentication.

The central server selects a subset of clients to participate in the training round. Each client of this subset gets the model from the central server and performs the training steps from its local data. The client improves the model by training its specific data since user data used in the training process never leaves the device. Afterwards, model updates, *i.e.*, learned parameters or gradients, are sent periodically to the central server. Finally, the central server aggregates the parameters learned from the clients that took part in the training round, using the Federated Averaging (FedAvg) algorithm to improve the model which will create a generic model for classifying all users in data based on the average of each user. Moreover, the learned parameters are shared with customers.

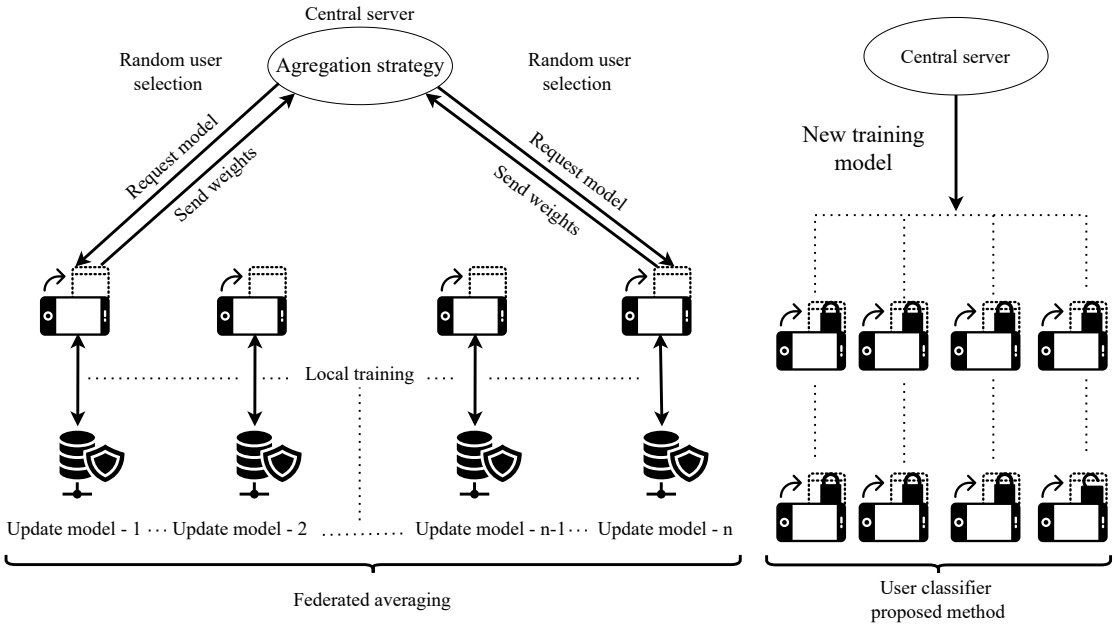


Figure 1. FL approach to define a user's biometric behavior pattern overview

In the authentication step, a set of features is checked against the patterns stored in the database so that it can assert whose identity it is, which is known as a one-to-one matching system. The authentication process responds if the subject is who it claims to be, while identification responds if it is someone registered in a previous database.

3.2. Biometric Data

We considered a database of user interaction with mobile devices obtained in a non-intrusive way from the application BrainRun [Papamichail et al. 2019]. The BrainRun is a mobile mental exercise game application with obtained biometric data. Its main purpose is to collect information from users in a non-intrusive way and store it in a database. One significant feature of this dataset is that it was not created in a strictly controlled environment. This feature is important because, under controlled conditions where users are aware of how their data is being used, there is a possibility that users will not behave as they would typically behave. Therefore, any conclusions and results cannot be representative of reality.

The application contains a list of brain training games through which data is collected from sensors. BrainRun has five distinct game types, called “Focus”, “Mathisis”, “Memoria”, “Reacton”, and “Speedy”. Each game mode is specifically designed to collect specific gestures such as taps, horizontal swipes, vertical swipes, combined taps, etc. Moreover, measurements from sensors were collected, including data from the accelerometer, gyroscope, and magnetometer. For this work, the gyroscope sensors were selected to reproduce how the user interacts with his device. The data obtained from the motion sensors have location variables (X, Y, and Z) and the game screen. Each variable represents the device’s position according to the three axes, and the game screen is the type of game that was recorded. For example, the gyroscope detects changes in the phone’s orientation concerning the three axes with a high level of accuracy. In this paper, the data collected by the sensors is related to time, which collects several measurements per second. We need the time series for user identification.

In the BrainRun dataset, each timestamp is a time series created from continuous data samples collected every twenty milliseconds. Figure 2 shows the biometric data distribution of users in the BrainRun dataset, which includes 3.11 million gestures from 2218 users; where each number in the x-axis is a number ID with its respective number of appearance in the dataset in the y-axis. Due to the time series data, it is necessary a technique to minimize its size and thus continuously send for the CNN training. Therefore, we chose a sliding window partitioning technique to compress the data and eliminate possible measurement errors. The logic of this methodology is to create sections with data of equal size to extract the characteristics. In this case, we propose a continuous biometric authentication in the user using the gyroscope as a unique identifier to ensure the data continuity to be processed.

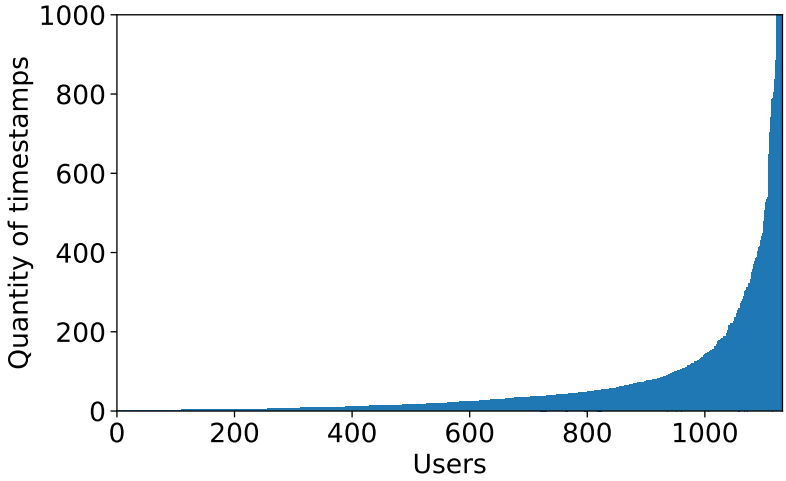


Figure 2. BrainRun data overview

Based on the BrainRun dataset analysis, we can observe that this dataset’s properties do not have the same distribution probability. This heterogeneity is a consequence of the interaction of clients’ records on their own devices. Therefore, many clients’ data have different features and sizes, impacting the distribution of the dataset.

3.3. Pre-processing and features extraction

[Hammerla et al. 2016] have shown through thousands of experiments with public Human Activity Recognition (HAR) datasets that Recurrent Neural Networks (RNN) and Long Term Memory (LSTM) are the most efficient in recognizing short-term activities with the natural order. At the same time, CNNs are better with repetitive and long-term activities. In this sense, CNN has been used to model the temporal structure in several types of tasks [Centeno et al. 2018]. In a continuous authentication context, advanced features serve as input to a subsequent model and establish authentication metrics.

Segmenting the data into fixed time intervals is fundamental to obtaining enough samples for data modeling. The CNN model processes segmented data and comprises several one-dimensional convolution layers together with the rectified linear activation function (ReLu). The ReLu function is a linear function that allows faster learning and better performance, especially for neural network models such as CNN. ReLu produces the output directly if the input goes positive; otherwise, it will output zero. The extraction of features from the input windows is performed by each convolution layer based on filters with size 200, each with a different size kernel. It is a challenging task to select the appropriate CNN parameters, so we used filters with varying kernel sizes in each layer to obtain meaningful information from CNN.

Figure 3 presents an overview of extracting features and classification steps to obtain behavioral biometrics through FL for authentication and user identification. First, each user starting from their ID, had their JSON files joined. Subsequently, we structured the data to obtain information related to the sensors. In Phase 1, the division of users by file demonstrates the need to merge them into a single data frame from raw data. The data labeling of Phase 2 is needed to choose the number of timestamps required for each user collected, this being the first time the user uses the application. Each user's samples were labeled with a numerical identifier so that each sample segment was labeled and assigned to its respective user after segmenting the samples in the windows.

In Phase 3, it is necessary to establish which data will collect from the sensors passed by BrainRun, collecting the gyroscope data of each client in the X, Y, and Z direction for each user training and testing. That way, proceeding to Phase 4 uses the data from each axis in the fourth phase to form the data windows parameters sent to CNN's training and test process. Additional pool layers were introduced to downsampling the sampling and applying them locally or globally. Local pooling takes an input time series and reduces its length by aggregating (average or maximum) in a sliding time series window. The global pool aggregates the time series into a single value. In this context, Phase 5 matches the windows with the labels. The labels are the names of the users that must be within the timeline carried out, sent together with the data necessary for the classification of the user by the data presented by the gyroscope windows in the three axes. Phase 6 shows the end of data extraction. This extraction is obtained from the data collected from the temporal windows of the data frame, developing the data distribution and passing it on in the data frame used in our FL approach to defining a user's biometric behavior pattern.

4. Evaluation

This section presents the experimental evaluation of FL approach to define a user's biometric behavior pattern, including dataset work description, FL-based classification model, metrics, and results obtained. Moreover, we used the Flower Framework for our simula-

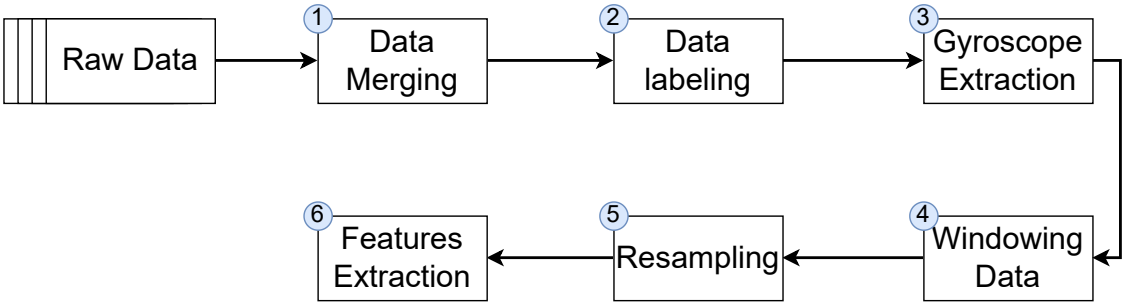


Figure 3. Data pre-processing pipeline

tion, a Python library that works as a Friendly Federated Learning framework. This framework enables the recreation of a simulation of several users locally. Moreover, SCOMNet is an open-source project available on GitHub[1].

4.1. Methodology

The raw data has 1,132 users with specific numbers of timestamps in a total of 101,065. Each JSON sample file has four types of available sensor data: gyroscope, accelerometer, device motion, and magnetometer. Table 1 compiles the simulation parameters, such as the values of each essential described information necessary to our user filter extracted from the Brainrun dataset. Table 1 contains information regarding the number of users, groups, timestamp, time of sample, and window sizes, among others.

Table 1. Simulation Parameters

Characteristics	Values
Time span of sample	15 - 20 ms
Number of users	1132
Number of users with non-null values	804
Number of groups users for timestamp	234
Usable users	169
Timestamp scale	1 - 7904
Number of samples to compose the timeline	200, 400 and 590 sample
Windows sizes for CNN application	100, 150 and 200 size
User for training model	60 users

We split the users according to the number of timestamps, but each timestamp has a different quantity of samples in the same user. Each timestamp is created when a user runs the gaming app Brainrun, recording a new sample span. This sample span (initially defined by the BrainRun developers) is necessary to understand the timeline we use to identify a user and his timeline ID. It is important to highlight that we need a minimum number of samples and multiple timelines for training CNN when we work on a time-series approach and CNNs. Our evaluation shows that our proposal excels with

[1]https://github.com/VeigarGit/BiometricBehaviorFL

400 minimum samples. Filtering data is necessary to search how many users could use it for training CNN and build a data frame for better accuracy in user identification and discarding non-usable users. In our case, we discarded more than 100 null users and only selected the others.

We used the CNN method due to the research initiatives of technologies in deep learning for a multivariate classification of images and time series, where we evaluate two CNN algorithms, Inception, and ResNet. Specifically, **Inception** is a proposal of time classification to resolve traditional supervised learning problems in structured data [Ismail Fawaz et al. 2020]. The ideal for novel deep learning for the time series is adding multiple cascading Inception modules that can multiply filters simultaneously to an input time series that automatically extract relevant features from the time series. It is possible to see the scalability and accuracy of this new classifier with the addition of the Inception module in the architecture. On the other hand, **ResNet** aims to resolve representations and classify time series data based on another deep learning approach of Multilayer Perceptrons (MLP), and Fully Convolutional Networks (FCN) [Wang et al. 2017]. While other neural networks need a stable database with the characteristics of the data analyzed and processed, ResNet proposes an approach of CNN in multi-scale (MCNN) for non-varied time series classification. The results shows that CNN is a robust time series classification for deep neural networks for evaluating the presentation error rate of diverse CNNs.

We created a scenario to show the evaluation of two different CNNs by comparing their behavior and changing the parameters of windows, epochs, and data size. We conducted several experiments with the raw data and the desired CNNs to find the best metrics for the fastest classification with the best evaluation. For this purpose, the evaluation will be made after the aggregation step and will use the global model using the weighted average of a round. In this context, we see how the metrics played an essential role in the final metrics used in this work. The Inception of each block is composed of three modules in the convolutional layers, called the bottleneck. This layer operates for sliding each user time series using multiple filters to classify users from the samples. However, ResNet uses a shortcut connection on the residual blocks to enable the gradient flow directly through the bottom layers and reuse them to form his ReLu function filters of the convolutional layer. In our evaluation, we analyze both CNNs with several numbers of rounds and epochs to analyze the impact of both issues on the performance of an FL approach for user authentication and identification.

We compare these algorithms with metrics commonly used for authentication, namely Accuracy, Loss, False Rejection Rate (FRR), and False Positive Rate (FPR), summarized in the following. The **Accuracy** metric is obtained by the number of hits (positive) divided by the total number of examples, which is used on data with the examples for each class and when the miss. Moreover, hit penalties for each class are the same. In the case of disproportionate classes, it gives a false impression of good performance, delivering a flawed result. The **Loss** metric is used for comparing the target and predicted output values, which helps see how the neural network models the training data. It calculates the average loss, weights, and biases in that case.

FRR is a metric that proposes to study a proportion of verification transactions with truthful claims of identity that were incorrectly denied. FRR is calculated using False Negative (FN) divided by False Negative (FN) plus true positive (TP). This metric occurs when a security parameter/system fails to recognize an authorized user, *i.e.*, a false

rejection occurs. On the other hand, **FPR** is a metric focused on justifying a prediction of when an assumption happened in which the value should be negative. FPR is calculated by using False Positive (FP) divided by False Positive (FP) plus True Negative (TN). FL gives the proposed result as positive.

4.2. Results

Figure 4 shows the accuracy for two FL models (ResNet and Inception) with two different epochs and a centralized model for user authentication and identification. By analyzing the results, we can observe that Inception and ResNet converge to a high accuracy rate only after a few rounds running (specifically, only after round 4 in the worst case). This behavior is because the FL models need more averaging weight from the server to the convolutions layer, i.e., more rounds to obtain the expected results. By analyzing the same two CNNs with a higher epoch (10), we see an improvement in the accuracy results, where ResNet achieved 98% in round 2 and Inception in round 3, which confirms that both worked better with a higher number of epochs. Finally, the centralized approach obtained 98% in all rounds since the FL models train with the full data instead of the weights and therefore present a possible breach in security when compared to fully decentralized approaches. However, the centralized approach requires sending all the data to the central server, while ResNet and Inception send only the weights to the server at each round.

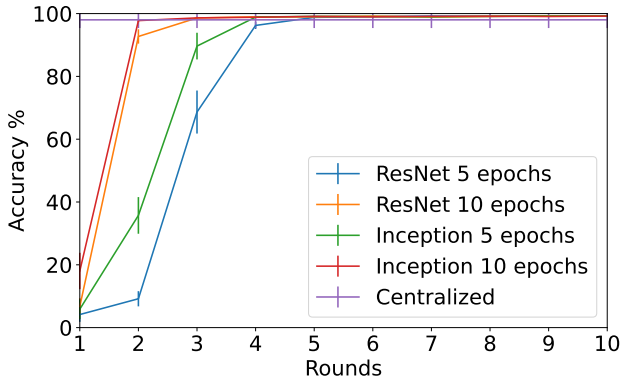


Figure 4. Accuracy measurements

Figure 5 shows the loss for two FL models (ResNet and Inception) with two different epochs and a centralized model for user authentication and identification. By comparing the results of both CNNs with five epochs, we can observe that Inception obtains good results faster than ResNet in rounds 4 and 5. The same observation is seen in the ten epochs evaluation, where Inception achieved a minimum loss faster with the higher epoch, *i.e.*, in round 2, while ResNet was in round 3. By comparing the performance between the two CNNs, we see how the activation response benefits their calculation. Inception arrives at the expected value much faster than ResNet in both situations. This behavior is because the aim of each CNN has some differences. For instance, ResNet uses "ResNet blocks" in the second layer in contrast to Inception, which uses the Inception module. In this case, Inception uses its modules to activate the ReLu function more times for more precision in training, thus obtaining better results. Finally, the centralized approach obtained fewer losses in the evaluations since it has an advantage in data access, even though this might be an undesired characteristic regarding data reliability.

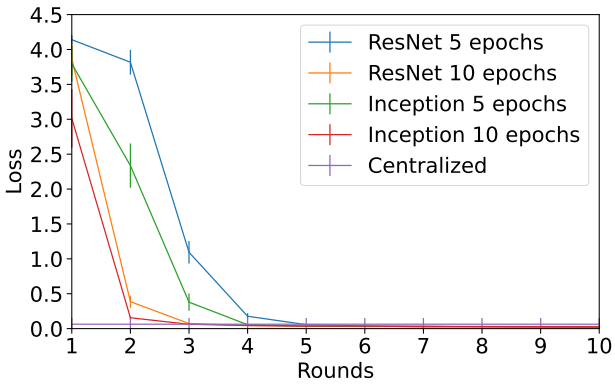
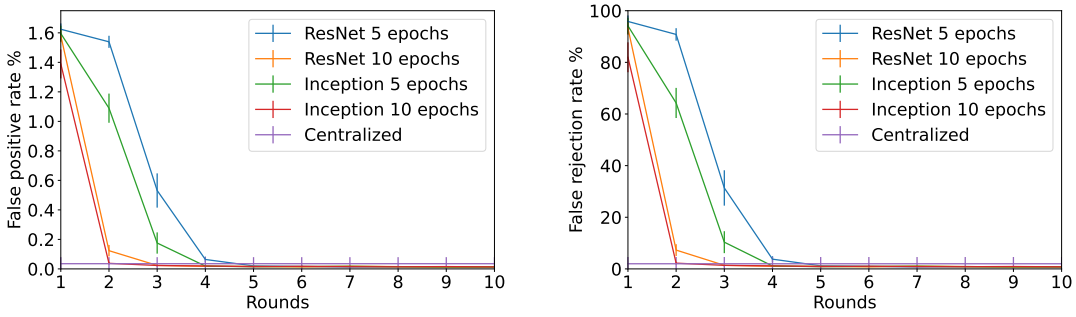


Figure 5. Loss measurements

It is even more necessary to obtain the results of false rejections and false positives obtained by CNN in the simulation of test results due to our objective based on the identification of users. Figure 6a demonstrates how the chance of false positives has been low since its beginning due to the low percentage of results. The FPR results show how faster the Inception CNN can reach low FPR values for 10 epochs in the second round, while ResNet achieved similar results in the third round. By analyzing the five epochs evaluation, both CNNs spend more time to converge, which takes more rounds to train these identification values and to obtain good FPR results. Figure 6a shows that the prediction to identify the user is possible with low chances of failure in this function.



(a) FPR (b) FRR

Figure 6. FPR and FRR measurements

Figure 6b shows the FRR results for two FL models (ResNet and Inception) with two different epochs and a centralized model for user authentication and identification. The FRR results show how the learning of the CNNs tends to be low from the beginning, implying that the CNNs Inception and ResNet obtained good results for user identification. The rate of false rejections is high only on the first simulations because all the CNN are in the training process. During this training process, on each round is necessary to look at how fast they got the expected results. Likely in the second round with ten epochs, Inception reaches the centralized model. However, with five epochs, it reaches only with fourth round. In less time and rounds, Inception can beat centralized on identifying users without data sharing, presenting itself as an important decentralized approach.

We also performed several other experiments with several numbers of samples, window sizes, and epochs, as shown in Table 2. Based on these results, we notice that distinct parameters have an essential role in the final results. Moreover, we can observe that 200 samples are insufficient for different user timelines. This behavior is due to the high variability of accuracy. For example, we can analyze slow processing in convergence data using many samples, as in the 590 samples, meaning the FL approach. In all cases of 590 samples of different epochs, from taking bad results to best accuracy but spending much time on this. We found examples of 400 samples using an average size in this process. In this case, we discovered the better parameters to use, *i.e.*, when we use 5 and 10 epochs, we can search the best metrics in a small number of rounds. Finally, the best selection for identifying users for the Brainrun dataset was 400 samples, 200 windows size, and 10 epochs. Although we chose 10 epochs as the high evaluation, this parameter needed more time to spend during training data but reached the result in less time.

Table 2. Summary of obtained results

Number of samples	Window size	Epochs	Results
590	150	10	Medium evaluation
590	100	10	Bad evaluation
590	200	10	Good evaluation
400	150	10	Slow and high loss
400	100	10	High Loss
400	**200**	**10**	**Best evaluation**
400	200	5	Good evaluation
200	150	10	High loss
200	100	5	low evaluation

5. Conclusion

In this paper, we analyzed the Brainrun dataset for user identification by gyroscope sensor using different CNNs on the Federated Learning approach. In our simulation, we use several experiments to find the best parameters with both CNNs to see which could better use for our purpose. Therefore, we found the best metrics and concluded that Inception better evaluates our scenery. Moreover, comparing centralized and FL can use for user identification. Although FL is a more reliable and independent method for user data.

In future work, we can extend the evaluation by adding more class users in the simulation and trying to search for how to work with them. This additional evaluation can be available with more impact in a real scenario training with a gyroscope and accelerometer. In contrast, an accelerometer can improve the accuracy of a gyroscope or can have more convergences between clients' data.

Acknowledgment

This work was conducted with partial financial support from the National Council for Scientific and Technological Development (CNPq) under Grant Nos. 311376/2020-7,

405111/2021-5 and 130555/2019-3, to PPI-Softex with support from the MCTI 01245.013778/2020-21, and also to the Coordination for the Improvement of Higher Education Personnel (CAPES) - Finance Code 001, Brazil. Moreover, MCTIC/CGI.br/São Paulo Research Foundation (FAPESP) partially supported the work through the Project SAMURAI - Smart 5G Core And MUltiRAn Integration under Grant 2020/05127-2 and Programmability, ORchestration and VIRtualization of 5G Networks (PORVIR-5G) under Grant No. 2020/05182-3.

References

[Acien et al. 2020] Acien, A., Morales, A., Fierrez, J., Vera-Rodriguez, R., and Bartolome, I. (2020). Becaptcha: Detecting human behavior in smartphone interaction using multiple inbuilt sensors. *arXiv preprint arXiv:2002.00918*.

[Barros et al. 2020] Barros, A., Resque, P., Almeida, J., Mota, R., Oliveira, H., Rosário, D., and Cerqueira, E. (2020). Data improvement model based on ecg biometric for user authentication and identification. *Sensors*, 20(10):2920.

[Barros et al. 2021] Barros, A., Rosário, D., Cerqueira, E., and da Fonseca, N. L. (2021). A strategy to the reduction of communication overhead and overfitting in federated learning. In *26th Workshop on Management and Operation of Networks and Service (WGRS)*, pages 1–13. SBC.

[Centeno et al. 2018] Centeno, M. P., Guan, Y., and van Moorsel, A. (2018). Mobile based continuous authentication using deep features. In *Proceedings of the 2nd International Workshop on Embedded and Mobile Deep Learning (EMDL)*, pages 19–24.

[Dahia et al. 2020] Dahia, G., Jesus, L., and Pamplona Segundo, M. (2020). Continuous authentication using biometrics: An advanced review. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 10(4):e1365.

[Ek et al. 2020] Ek, S., Portet, F., Lalanda, P., and Vega, G. (2020). Evaluation of federated learning aggregation algorithms: Application to human activity recognition. In *ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp-ISWC '20)*, page 638–643, New York, NY, USA. ACM.

[Espín López et al. 2021] Espín López, J. M., Huertas Celdrán, A., Marín-Blázquez, J. G., Esquembre, F., and Martínez Pérez, G. (2021). S3: An ai-enabled user continuous authentication for smartphones based on sensors, statistics and speaker information. *Sensors*, 21.

[Hammerla et al. 2016] Hammerla, N. Y., Halloran, S., and Plötz, T. (2016). Deep, convolutional, and recurrent models for human activity recognition using wearables. *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI'16)*, pages 1533–1540.

[Ismail Fawaz et al. 2020] Ismail Fawaz, H., Lucas, B., Forestier, G., Pelletier, C., Schmidt, D. F., Weber, J., Webb, G. I., Idoumghar, L., Muller, P.-A., and Petitjean, F. (2020). Inceptiontime: Finding alexnet for time series classification. *Data Mining and Knowledge Discovery*, 34(6):1936–1962.

[Kandar et al. 2021] Kandar, S., Pal, S., and Dhara, B. C. (2021). A biometric based remote user authentication technique using smart card in multi-server environment. *Wireless Personal Communications*, 120(2):1003–1026.

[Li et al. 2021a] Li, C., Niu, D., Jiang, B., Zuo, X., and Yang, J. (2021a). Meta-har: Federated representation learning for human activity recognition. In *The Web Conference 2021 (WWW '21), Virtual Event / Ljubljana, Slovenia, April 19-23, 2021*, pages 912–922. ACM.

[Li et al. 2020a] Li, T., Zhang, M., Cao, H., Li, Y., Tarkoma, S., and Hui, P. (2020a). "what apps did you use?": Understanding the long-term evolution of mobile app usage. In *Proceedings of The Web Conference 2020*, page 66–76, New York, NY, USA. ACM.

[Li et al. 2020b] Li, Y., Hu, H., Zhu, Z., and Zhou, G. (2020b). Scanet: Sensor-based continuous authentication with two-stream convolutional neural networks. *ACM Transactions on Sensor Networks*, 16.

[Li et al. 2021b] Li, Y., Tao, P., Deng, S., and Zhou, G. (2021b). Deffusion: Cnn-based continuous authentication using deep feature fusion. *ACM Trans. Sen. Netw.*, 18(2).

[Liang et al. 2020] Liang, Y., Samtani, S., Guo, B., and Yu, Z. (2020). Behavioral biometrics for continuous authentication in the internet of things era: An artificial intelligence perspective. *IEEE Internet of Things Journal*, PP:1–1.

[Lobato et al. 2022] Lobato, W., Da Costa, J. B., de Souza, A. M., Rosário, D., Sommer, C., and Villas, L. A. (2022). Flexe: Investigating federated learning in connected autonomous vehicle simulations. In *2022 IEEE 96th Vehicular Technology Conference (VTC2022-Fall)*, pages 1–5. IEEE.

[Mekruksavanich and Jitpattanakul 2021] Mekruksavanich, S. and Jitpattanakul, A. (2021). Deep learning approaches for continuous authentication based on activity patterns using mobile sensing. *Sensors*, 21(22).

[Nemes and Antal 2021] Nemes, S. and Antal, M. (2021). Feature learning for accelerometer based gait recognition. In *2021 IEEE 15th International Symposium on Applied Computational Intelligence and Informatics (SACI)*, pages 479–484.

[Papamichail et al. 2019] Papamichail, M. D., Chatzidimitriou, K. C., Karanikiotis, T., Oikonomou, N.-C. I., Symeonidis, A. L., and Saripalle, S. K. (2019). Brainrun: A behavioral biometrics dataset towards continuous implicit authentication. *Data*, 4(2).

[Pires et al. 2020] Pires, I. M., Marques, G., Garcia, N. M., Flórez-Revuelta, F., Canavarro Teixeira, M., Zdravevski, E., Spinsante, S., and Coimbra, M. (2020). Pattern recognition techniques for the identification of activities of daily living using a mobile device accelerometer. *Electronics*, 9(3):509.

[Stylios et al. 2021] Stylios, I., Kokolakis, S., Thanou, O., and Chatzis, S. (2021). Behavioral biometrics & continuous user authentication on mobile devices: A survey. *Information Fusion*, 66:76–99.

[Wang et al. 2017] Wang, Z., Yan, W., and Oates, T. (2017). Time series classification from scratch with deep neural networks: A strong baseline. In *2017 International joint conference on neural networks (IJCNN)*, pages 1578–1585. IEEE.

[Zhu et al. 2019] Zhu, T., Qu, Z., Xu, H., Zhang, J., Shao, Z., Chen, Y., Prabhakar, S., and Yang, J. (2019). Riskcog: Unobtrusive real-time user authentication on mobile devices in the wild. *IEEE Transactions on Mobile Computing*, 19(2):466–483.