IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 9 June 2023, accepted 19 June 2023, date of publication 26 June 2023, date of current version 30 June 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3289220

RESEARCH ARTICLE

Federated Learning for Activity Recognition: A System Level Perspective

STEFAN KALABAKOV[1,2,3], BORCHE JOVANOVSKI[1], DANIEL DENKOVSKI[1], VALENTIN RAKOVIC[1], (Senior Member, IEEE), BJARNE PFITZNER[2], ORHAN KONAK[2], BERT ARNRICH[2], AND HRISTIJAN GJORESKI[1]

[1]Faculty of Electrical Engineering and Information Technologies, Ss. Cyril and Methodius University in Skopje, 1000 Skopje, North Macedonia
[2]Digital Health—Connected Healthcare Group, Hasso Plattner Institute, University of Potsdam, 14469 Potsdam, Germany
[3]Jožef Stefan International Postgraduate School, 1000 Ljubljana, Slovenia

Corresponding author: Valentin Rakovic (valentin@feit.ukim.edu.mk)

This work was supported by the WideHealth Project—European Union's Horizon 2020 Research and Innovation Programme under Grant 952279. The work of Stefan Kalabakov was supported by the Slovene Human Resources Development and Scholarship Fund (Ad Futura).

ABSTRACT The past decade has seen substantial growth in the prevalence and capabilities of wearable devices. For instance, recent human activity recognition (HAR) research has explored using wearable devices in applications such as remote monitoring of patients, detection of gait abnormalities, and cognitive disease identification. However, data collection poses a major challenge in developing HAR systems, especially because of the need to store data at a central location. This raises privacy concerns and makes continuous data collection difficult and expensive due to the high cost of transferring data from a user's wearable device to a central repository. Considering this, we explore the adoption of federated learning (FL) as a potential solution to address the privacy and cost issues associated with data collection in HAR. More specifically, we investigate the performance and behavioral differences between FL and deep learning (DL) HAR models, under various conditions relevant to real-world deployments. Namely, we explore the differences between the two types of models when (i) using data from different sensor placements, (ii) having access to users with data from heterogeneous sensor placements, (iii) considering bandwidth efficiency, and (iv) dealing with data with incorrect labels. Our results show that FL models suffer from a consistent performance deficit in comparison to their DL counterparts, but achieve these results with much better bandwidth efficiency. Furthermore, we observe that FL models exhibit very similar responses to those of DL models when exposed to data from heterogeneous sensor placements. Finally, we show that the FL models are more robust to data with incorrect labels than their centralized DL counterparts.

INDEX TERMS Human activity recognition, federated learning, deep learning, system-level aspects, different and heterogeneous sensor placements, FL optimizers, fraction fit, bandwidth efficiency, data errors, feature selection, model complexity.

I. INTRODUCTION

The ever-increasing ubiquity of devices such as smartphones, smartwatches, fitness trackers, and smart glasses has paved the way for many new applications that could be offered to users. This is mainly due to the incredibly valuable context information acquired through them, which enables applications such as (i) remote monitoring of patients [1], (ii) prevention and detection of high-risk situations, such as falls [2], (iii) fitness and lifestyle improvements [3], (iv) detection of cognitive diseases such as Parkinson's disease [4], [5], and (v) automatic activity log generation [6].

Although today's wearable devices contain many different types of sensors and can capture large amounts of diverse

The associate editor coordinating the review of this manuscript and approving it for publication was Mostafa M. Fouda.

data, data collection remains one of the most prevalent problems in Human Activity Recognition (HAR). This is due to the fact that the process is time-consuming, expensive, and usually performed only once, at the start of the development of any specific HAR pipeline. One of the reasons behind performing data collection only once is that the technologies used to develop HAR models require data to be centrally stored before being used. This introduces significant privacy risks and hinders continuous data collection both because of security concerns and the potentially substantial costs incurred by sending large volumes of data from a user's device to a central data store.

A possible solution to these problems could be the use of Federated Learning (FL) [7], instead of the widely used centralized classical machine learning (ML) and deep learning (DL) methods. FL is a distributed learning paradigm that focuses on developing a shared model using clients who each only have access to their own data. The primary advantage of FL is that a client's data never leaves their device, which substantially decreases any security risks related to sharing sensitive information. In addition, the only information that leaves the user's device when using FL is the computed updates/weights to the local model, which substantially reduces the volume of data that needs to be sent to a central data store compared to when users send actual sensor readings. Both the improved security and the reduced volume of data leaving a user's device increase the feasibility of performing continuous data collection, which, in turn, would significantly impact the ability of models to improve over time and adapt to changes in the distribution of the data.

However, efficient deployment and optimal operation of FL in real-world scenarios is far from a trivial task. FL is commonly deployed on communication and computationally constrained devices, and requires a better understanding of how various system-level factors impact its reliability and applicability. Such an understanding has immense potential to facilitate the development of more effective FL-based models, which would advance the practical application of FL in real-world settings.

Rather than proposing a new model or FL optimizer, this paper aims for a more significant and wider impact. Our main goal is to provide a comprehensive and rigorous system level analysis of federated learning for human activity recognition. This paper is the first, to our knowledge, to offer such a system-level perspective that covers various practical aspects and considerations. In particular, this work initially focuses on analyzing the performance gaps between the centralized deep learning and the distributed federated learning approaches using two different HAR datasets with different sensor placements. Finally, this paper aims to characterize the behavior of FL by comparing it to that of a centralized DL, when considering the following important practical aspects for real-world deployments:

- data from different sensor placements,
- heterogeneous sensor placements in clients that participate in the training at the same time,
- different server-side model aggregation strategies for FL (i.e., FL optimizers),
- a different percentage of clients participating in the learning process,
- communication bandwidth efficiency,
- model size and model complexity as a result of feature selection,
- data with corrupted labels.

The lessons learned throughout the paper can later serve as comprehensive guidelines for designing and optimizing federated learning systems for HAR.

The paper is organized as follows. Section II presents the related work at the intersection of FL and HAR. Section III presents the two HAR datasets used for training and evaluation of our models. Next, Section IV describes the methodology, namely, the feature extraction that was performed, the model architecture used as well as the FL system architecture used. Section V presents the evaluation setup, metrics and the details of the experiments. Section VI presents and discusses the results from the experiments. Section VII compiles the lessons learned though our results and, finally, Section VIII provides a summary of the paper and discusses potential directions for future work.

II. RELATED WORK

State-of-the-art ML and DL solutions for HAR usually require data from different sensors and users to be located in one central location before being used to develop models. The disadvantages of training centralized models appear in the form of privacy concerns and the inability to perform continuous data collection due to both the security risk and the high cost of transferring large amounts of data from a user's device to a central data store. FL can mitigate these disadvantages by constructing a shared model using only the updates/weights computed by each client on their local machine and data.

Over the past few years, numerous studies have investigated the use of FL in the field of HAR. The majority of these studies have concentrated on exploring new FL applications in the context of activity recognition or enhancing FL pipelines and methodologies [8], [9], [10], [11], [12], [13], [14], [15]. These works usually aim to enhance the accuracy and resilience of the FL model, but they seldom focus on a broad exploration of the real-world deployment requirements of FL at the system level. Specifically, there is a lack of research on how various factors, such as fusion of data from different sensor placements, exposure to clients with data from heterogeneous sensor placements, and exposure to data with corrupted labels, affect the accuracy, communication efficiency, and complexity of FL systems for HAR. Furthermore, a head-to-head comparison of DL and FL models under varying conditions, in order to quantify and define the differences between the two paradigms, is also rarely performed.

Only a limited number of studies have attempted to provide a deeper understanding of the system-level specifics

of FL in the field of HAR. One example of such analysis is presented in [16], where the impact of non-iid data on the performance of a FL-based activity recognition system was investigated. Specifically, the authors examined how the performance was affected by clients having access to different subsets of activities, unbalanced numbers of examples from the activities they performed, and corrupted data. They also proposed a technique to address the issue of corrupted data. Another study that focused on the heterogeneity of clients' data is [17], where a device selection strategy was proposed to alleviate problems such as activity class imbalance and varying data sizes per client. Finally, in [18], the authors evaluated and compared various FL optimizers, including personalized ones. The study found that the federated averaging approach provided better global performance than the other more complex personalized approaches.

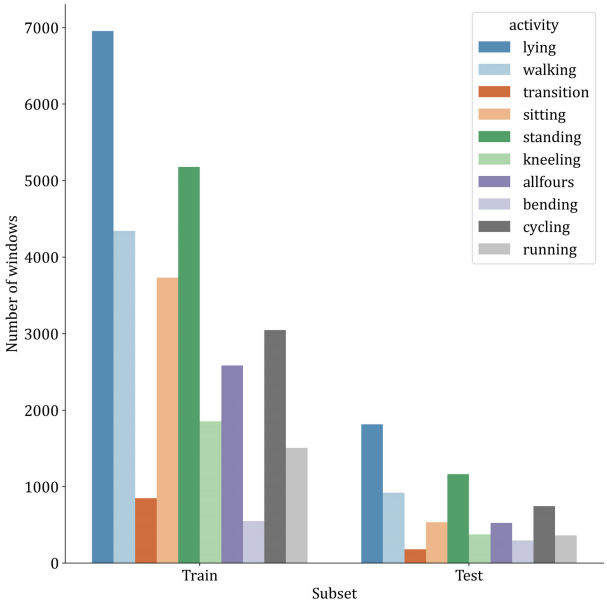
Although the mentioned papers have made contributions to the understanding of some system level aspects of FL-based HAR, they fail to provide a broad and thorough enough investigation of the requirements and implications in real-world deployments of these systems. More specifically, these works fail to investigate issues associated with diverse sensor placements in clients and data fusion, datasets containing corrupt labels, and the trade-off between optimal FL-specific hyperparameters, model accuracy, and communication and computational overhead. This work aims to build upon these limitations by exploring the effects of various FL-related factors on the overall system performance. Specifically, this work investigates how different sensor placements, FL optimizers and FL-specific hyperparameters, and data fusion affect model performance. In addition, the analysis encompasses the effects of communication bandwidth, model complexity, and data with corrupted labels on the overall precision, robustness and overhead efficiency of FL models.

III. DATA AND PREPROCESSING

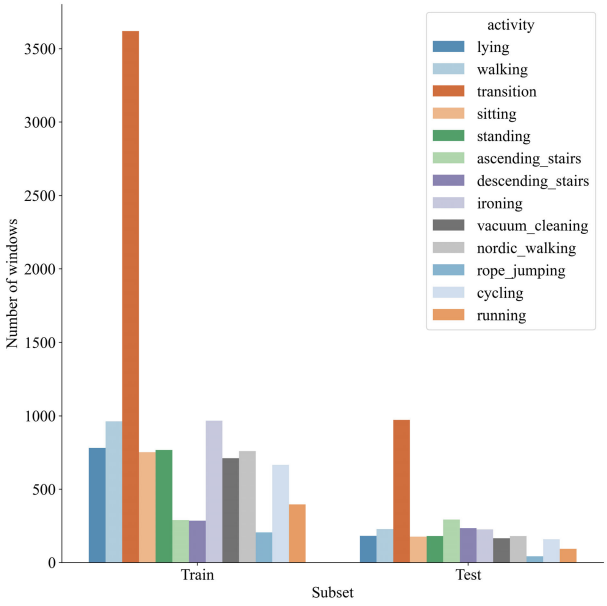
For the purposes of training and evaluation of our models, in this work, we used the JSI-FOS [19], [20] and PAMAP2 [21], [22] (hereinafter referred to simply as PAMAP) datasets. Both of these datasets contain recordings of activities of daily living (ADL) made using Inertial Measurement Units (IMUs) which users wore attached to different parts of their bodies.

More specifically, the JSI-FOS dataset consists of recordings collected from ten subjects while performing the following activities: walking, standing, sitting, running, lying, lying_exercising, kneeling, cycling, allfours_moving, allfours. Although more IMU placements were available, in this analysis, we considered only the data collected by the IMUs placed on the wrist of the dominant hand and the thigh of the dominant leg. Furthermore, we only considered data coming from the accelerometer and gyroscope. During data collection, values from the sensors were sampled using a frequency of 50 Hz.

Similarly to the JSI-FOS dataset, the PAMAP dataset consists of recordings collected from nine subjects while performing the following activities: lying, walking, transition, sitting, standing, ascending_stairs, descending_stairs, ironing, vacuum_cleaning, nordic_walking, rope_jumping, cycling, running. Here, as well, we chose to work with only a subset of the IMU locations and sensor modalities available in the dataset, limiting ourselves to data coming from the wrist of the dominant hand and the chest of the user and coming from either an accelerometer or a gyroscope. Originally, the values from the sensors were sampled using a frequency of 100 Hz.



(a) JSI-FOS dataset



(b) PAMAP dataset

FIGURE 1. The distribution of activities in the aggregated training and test subsets of the (a) JSI-FOS dataset, and the (b) PAMAP dataset.

Before applying the feature extraction procedure described in Section IV-A to both of these datasets, we first performed some common preprocessing steps. Firstly, we downsampled the data in the PAMAP dataset to a sampling frequency of 50 Hz to reduce the complexity of the problem. Then, we handled PAMAP's missing values by performing a backward fill operation followed by a forward fill operation (to handle missing values at the end of recordings).

Regarding the preprocessing steps applied to both datasets, we first segmented the continuous data streams into smaller windows. More specifically, we used windows of two seconds without any overlap. The label of each window was determined as the label most commonly found among the readings contained in the window. Next, we calculated the magnitude of the vectors provided by the accelerometer and the gyroscope at each sampling point. Finally, we filtered the raw sensor data using a band-pass filter to remove both the gravitational component and the noise inherently present in the data. It is important to mention that we kept and used both the unfiltered and the filtered versions of the data.

Fig. 1(a) and Fig. 1(b) show the distribution of the labels in the training and testing subsets (as described in Section V-A), after the segmentation of the original continuous recordings of the JSI-FOS and PAMAP datasets, respectively.

The data in both datasets are distributed equally among the subjects who participated in the data collection, except for 'subject 109' in the PAMAP dataset. This subject performed only a small subset of the activities and recorded very little data from them. It is also worth noting that the activity distribution in each subject's data is similar to the one shown in either Fig. 1(a) or Fig. 1(b), depending on the dataset to which the subject belongs. Additionally, it is evident that the activity distribution and frequency of occurrence are consistent between the test and train subsets, irrespective of the dataset under consideration.

IV. METHODOLOGY

This section explains the methodology used in our study, i.e., the feature extraction process, the deep learning model architecture and the federated learning setup, respectively elaborated in the following subsections.

A. FEATURE EXTRACTION

To reduce the complexity of both the training and inference phase in our experiments, we decided to use a simple Feed-Forward Neural Network (FFNN) that operates on extracted features instead of the raw sensor data. Moreover, utilizing a simplistic FFNN for HAR has more practical applicability, because of the limited computational and energy capacity of HAR-related Internet-of-Things (IoT) devices.

This means that after preprocessing the data, the next step in the ML pipeline is to extract an informative and diverse set of features with which the FFNN would be able to achieve high classification accuracy. Considering this, we extracted several types of features which have proved effective when analyzing time-series data and in particular, data from inertial sensors used for HAR [23]. The features we extracted, categorized in three groups, are the following:

- **generic:** mean, standard deviation, median, min, max, range, interquartile range, kurtosis, skewness, root mean square
- **HAR-specific:** integral, mean crossing rate, number of peaks, average height of peaks, peak-to-Average power ratio, sum, squared sum
- **frequency-domain:** energy, entropy, binned distribution, three largest PSD magnitudes and their frequencies, skewness, kurtosis

The same features were extracted from both the accelerometer and gyroscope data and more specifically, for each of their channels (including the magnitude). Furthermore, as was previously mentioned, features were extracted from both the filtered and the unfiltered versions of the signal values. In total, for each window we extracted 1184 features.

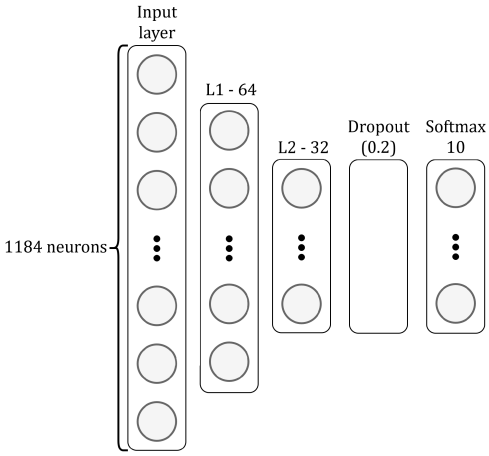


FIGURE 2. The architecture of the used feed-forward neural network for training/inference.

B. MODEL ARCHITECTURE

After performing feature extraction, the last step of the pipeline is the learning/inference step, performed by the aforementioned FFNN. The architecture of the FFNN is shown in Fig. 2. It consists of an input layer with as many neurons as there are features that describe a single window of data, followed by two fully-connected layers with 64 and 32 neurons, respectively. Both of these layers use the ReLU activation function [24]. The two fully-connected layers are followed by a single dropout layer with a rate of 0.2 [25]. Finally, the last layer in the network is a softmax layer with either 10 or 13 neurons, depending on the dataset used for training and evaluation. This network is used in the learning/inference step, regardless of whether the pipeline is used in a DL or FL context.

Finally, we want to point out that in all experiments, the models are trained using the Adam optimizer [26], with a learning rate of 0.0003, the categorical cross-entropy loss

function, and a batch size of 256. It is important to note that, when using FL, we also experimented with the use of the SGD (Stochastic Gradient Descent) optimizer, particularly because of the fact that it is stateless. However, the results suggested that there is no performance advantage of using SGD instead of Adam.

C. FEDERATED LEARNING SETUP

As previously mentioned, the core idea of FL is training a shared model using clients that never have to share data between themselves or with a server [29]. The depiction of a general FL implementation (and the one we use) is given in Fig. 3. A federated learning setup usually consists of a server that holds the shared model and coordinates the training process, as well as clients which all hold their own local data and models. The training of a shared model is achieved by aggregating the updates/weights that the clients make to their local models using their local data. This way, clients do not have to share their data with the server, but instead, only share the updates/weights of their local model. One training iteration of the shared model in FL is referred to as a round.

A more detailed illustration of what are the individual steps in a single round of training is given in Fig. 4. The whole process starts on the server-side with the initialization of the weights of the shared model. This only happens in the first round of training (thus, it is depicted with a dashed line). Next, the server picks a subset of clients (S) which will participate in the specific training round. This is done to simulate the fact that not all clients are available to participate in each round. The number of clients selected in each round of training is denoted as C. After picking the subset of clients that will participate, the server broadcasts the weights of the current shared model to all of the clients that are included in the training round.

After receiving the broadcasted weights, each of the included clients (client $x \in S$) creates a local copy of the shared model. This local model is then trained using their local data for a few epochs. Subsequently, each of the included clients sends only their updates/weights of the local model to the server. It is important to note that when referring to updates, we mean the difference between the received model and the local model after training using local data.

Finally, after receiving the updates/weights from all participating clients, the server is ready to update the shared model. This is done using some form of aggregation of the multiple received updates/weights. The updated shared model is used as the starting point for training in the next round.

V. EXPERIMENTAL SETUP

In the following subsections, we provide detailed information about the evaluation setup, metrics of interest, and experiments conducted in our study.

A. EVALUATION SETUP

Instead of using a Leave-One-Subject-Out strategy, we opted for a more personalized evaluation setup due to the unique suitability of FL for developing personalized models. To implement this setup, we divided the data of each user in both datasets into training and test subsets. The training subset typically consisted of approximately 80% of the user's data, equivalent to around 100 minutes of labeled data (about 3100 windows/instances) in the JSI-FOS dataset and around 46 minutes of labeled data (about 1390 windows/instances) in the PAMAP dataset (except for 'subject 109'). The remaining 20% of the user's data, equivalent to around 20 minutes of labeled data (about 700 instances/windows) in the JSI-FOS dataset and around 13 minutes of labeled data (about 390 instances/windows) in the PAMAP dataset, formed the test set. No validation sets were used in this study as there was no parameter tuning involved, and our focus was solely on reporting performance changes using different setups on the test data from each user.

To mitigate the potential issue of high similarity between windows containing data from the same user in close temporal proximity, we took precautions during the data splitting process. We ensured that windows belonging to a continuous performance of a specific activity (activity segment) were only present in either the training or test subset, but not both, in each of the two datasets. This was achieved through the following steps: (i) identifying activity segments in the data of each user, (ii) grouping activity segments based on the performed activity, (iii) iterating through the groups of activity segments and assigning each segment to either the training or test subset.

During step (iii), we assigned activity segments from each group to the training or test subset in such a way that approximately 80% of the windows in the group belonged to the training subset of the user, while around 20% of the windows in the group belonged to the test subset of the user. This approach ensured that the evaluation of the model was not biased by unintentional repetition of similar data during training and testing, and helped maintain the integrity of the evaluation process.

Due to the inherent differences between DL and FL, the utilization of the training and test subsets varied for each paradigm during the training and evaluation process. For DL models trained on one of the two available datasets, the training subsets of all users in that dataset were concatenated to update the model in each epoch. The concatenated test subsets of all users in the same dataset were used to evaluate the model after each epoch and at the end of the training procedure. In contrast, for FL models trained on one of the two datasets, the training subset of each user was used to train a local model in each round of FL. Simultaneously, the test set of each user was used to evaluate the respective local model's performance. However, after each training round, the shared global model was also evaluated using the concatenated data from the test subsets of all users in the dataset. This distinct approach in utilizing training and test subsets in DL and FL models accounts for the differences in how data is aggregated and utilized in each paradigm, taking into consideration the distributed and collaborative nature of federated

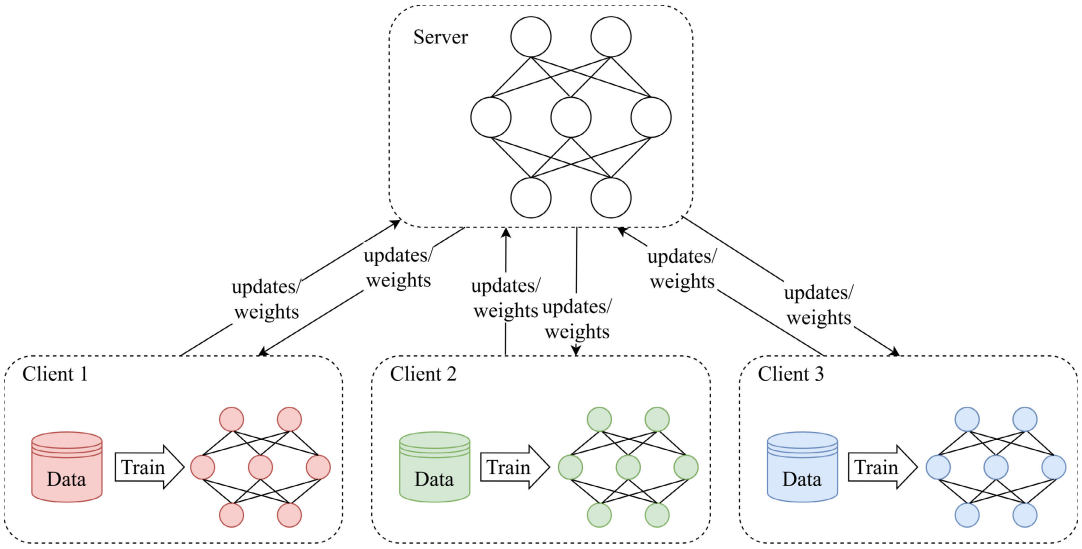


FIGURE 3. The architecture of a typical FL system.

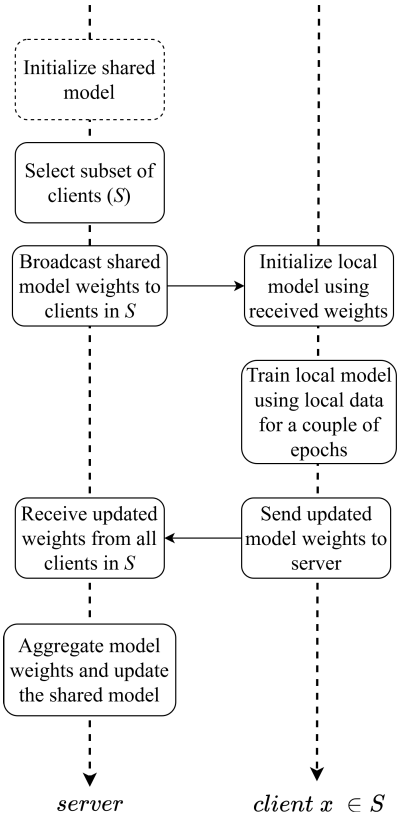


FIGURE 4. A step-by-step depiction of a single round of training when using the federated learning paradigm.

learning compared to the centralized training in deep learning.

B. METRICS

To account for the imbalanced distribution of activities in the JSI-FOS and PAMAP datasets, the macro F-score was utilized as the performance metric in this study. The macro F-score avoids bias towards activities with a larger number of examples, as it calculates the F-score for each activity separately and reports the average of those results.

The F-score is a harmonic mean of the precision and recall metrics for a specific label. While it may not be as easily interpretable as accuracy, higher F-score and macro F-score values (closer to 1.0) indicate better classification performance, while lower values (closer to 0.0) indicate poorer performance. It is worth noting that the macro F-score and accuracy metric may report similar values on datasets with a balanced distribution of activities.

C. EXPERIMENTS DEFINITION

The following section introduces all the experiments we conveyed in our study, providing descriptions, configurations and targets of the experiment analysis.

1) SENSOR PLACEMENT IMPACT

Our first experiment performs a head-to-head comparison between DL and FL models. Specifically, the comparison includes observing the performance of FL models and their gap to DL models when (i) using data from different sensor placements and (ii) when using different numbers of training epochs/rounds. In particular, the experiment investigates whether FL models exhibit similar behavior to DL models when the above-mentioned conditions are varied, and analyzes the performance gap between the two learning paradigms.

More specifically, we trained six models (three DL models and three FL models based on the three different sensor placements), on each of the two datasets, and evaluated their performance after each epoch/round of training. The maximum number of epochs/rounds used for training both DL and FL models was 50. It is important to note that when

comparing the DL and FL models, we treated one epoch (DL) and one round (FL) as equivalent. This approach is intended to provide fairness in the comparison, as FL locally operates on smaller amount of data, compared to DL, but exploits more local epochs. Also, the updates of the shared (global) FL model, occur in every round, which is equivalent to the model update at each epoch in the DL case. When training FL models, the C parameter was set to 6 and the number of local epochs used, was 5.

Furthermore, as already mentioned, we also varied the sensor placement whose data we used for training and testing. Namely, we used three possible sensor placements: (i) the wrist of the dominant hand, (ii) the thigh of the right leg, when using the JSI-FOS dataset, or the chest when using the PAMAP dataset, and (iii) a combination of both available sensor placements. When using data from two different sensor placements, the data were simply concatenated and examples from both placements had the same weight while training.

2) FL OPTIMIZER IMPACT

The goal of this experiment is to explore the behavior of different FL optimizers - FedAdagrad [27], FedYoGi [28], and FedAvg [29], with respect to their macro F-score performance. Specifically, we will evaluate these optimizers when using both sensor locations. Due to the different approach in computing the global model, it is expected that some optimizers should operate more accurately for the case of HAR.

3) IMPACT OF CLIENTS WITH HETEROGENEOUS SENSOR PLACEMENTS

Our third experiment investigates the impact of building a shared model using clients that have access to data from different sensor placements. In real-world scenarios, not every person who uses an activity recognition service will wear their sensor-equipped device at the same location on their body. For example, if that device is a smartphone, one person may wear it in the pocket of their trousers, and another might wear it in the pocket of their jacket or even in a backpack. This means that some of the clients of a FL model might send updates computed on data from one sensor placement, while others send updates computed on data from another sensor placement. Considering this, this experiment aims to explore the effects that receiving updates corresponding to data from heterogeneous sensor placements might have on the performance of the shared model.

To that end, we varied the number of users who only had access to data from one sensor placement but not the other, and observed the performance changes that occurred. In each training round, all clients, regardless of what data they had access to, were eligible to be used for training, while the selection of which clients had access to a particular type of data was done randomly. The whole process was repeated ten times to reduce the effects of randomness. It should be pointed out that in each repetition, the test subset of each user contained data from only one sensor placement, depending on what type of data the user was chosen to have access to.

As was the case previously, after each round, the model was evaluated on a test subset that was a combination of the individual test subsets of all users (clients). This effectively meant that the test subset used to evaluate the model, had roughly the same ratio of examples from different placement as the ratio of users who had access to data from different sensor placements.

Furthermore, aside from varying the number of users who had access to each location, we also varied the number of clients used for training in each round and the number of rounds used to train each model.

4) BANDWIDTH EFFICIENCY ANALYSIS

One of the most prominent advantages of FL is the exchange of the model information, instead of the complete dataset. This results in decreased volume of shared information, that facilitates higher bandwidth efficiency and easier collaboration and model building. However, the improved bandwidth efficiency can result in performance decline. This experiment aims at analyzing the effects of bandwidth efficiency on the overall FL model performance. Specifically, the experiment strives to analyze how the number of clients and the volume of the exchanged data impacts the precision and robustness of the FL model.

It is intuitive that DL will have an advantage compared to FL due to the larger volume data that is available to the model at any point in time. However, this larger data volume hampers the deployment of DL in real-world scenarios, where bandwidth limitation and efficiency is of utmost importance to IoT-based HAR systems. Conversely, the experiment also compares the FL and DL performances for the same amount of exchanged data. The comparison provides further insights regarding the applicability of FL when compared to DL.

The experiment setup and system configuration for the bandwidth efficiency analysis is the same as described in Section V-C1. The performance analysis is conducted with respect to the attained macro F-score as a function of the volume of data transmitted to a server. For FL, the data transfer volume is calculated as:

$$D_{FL} = C \cdot N_{tr} \cdot N_w \cdot P \tag{1}$$

where C is the number of random clients that participate in the round, N_{tr} is the number of training rounds executed in order to attain the given macro F-score, N_w is the number of weights of the client's model and P is the memory size of each weight in the model (i.e., 4B per weight, assuming single precision floating point). For DL, the data transfer volume is calculated as:

$$D_{DL} = F \cdot N_f \cdot N_{dr} \cdot P \tag{2}$$

where F is the fraction of data used for training the DL model, N_f is the number of features used for training (1184 in total), N_{dr} is the total amount of data rows (cumulative for

all clients), and P is the feature precision (i.e., 4B, assuming single precision floats).

For both, the DL and the FL strategy, multiple runs were conducted to calculate the 95% confidence intervals of the macro F-score. For comparability reasons, only two DL variations were considered, i.e., DL trained with 10% and 50% of the training part of the dataset.

5) MODEL COMPLEXITY AND THE EFFECTS OF FEATURE SELECTION

Often HAR-based systems rely on devices that have limited energy, computational and communication capabilities. Since FL relies on local model building, it is crucial to minimize the model complexity. However, straight-forward minimization of the model complexity can have detrimental effects on the overall performance of FL. As a result, there exists a requirement for exploring the possibilities that minimize the model complexity without significantly decreasing the FL performances.

Feature selection represents one of the most auspicious ways of minimizing the model complexity while attaining a certain level of robustness and precision of the FL model. This experiment analyzes the effects of model complexity minimization by feature selection, and discusses the potential benefits and pitfalls.

For the purposes of this experiment, the performed feature selection process is a Recursive Feature Elimination (RFE). The goal of the feature selection was set as selecting the best 100 features out of the total of 1184. Afterwards, the models were trained and tested on these 100 most important features

6) EFFECTS OF DATA WITH CORRUPTED LABELS

In real-world deployments, the available data is non-ideal and exhibits different negative properties, such as data will be noisier and data labels can be incorrect. This experiment analyzes the performance behavior of FL when considering non-ideal datasets. Specifically, the experiment analyzes the FL performances when there exist errors in the labeling of the data. The amount of erroneous data (wrong labels) is varied for both DL and FL. Since FL relies on a subset instead of all clients during each round of the training phase, it is very important to analyze how the volume of erroneous data correlates with the number of active clients per round, and how it compares to the DL case.

The dataset with corrupted labels is generated from the JSI-FOS dataset. The process of generating the erroneous labels, is as follows: (i) randomly select specific amount of labels (i.e., 1%, 10% or 20%) that will be incorrect; (ii) for the selected labels, choose a different label based on a uniform random distribution from all available ones in the dataset; (iii) use the newly generated dataset for training.

VI. RESULTS AND DISCUSSION

This section presents and elaborates on the main results we obtained from all the experiments introduced in Section V-C.

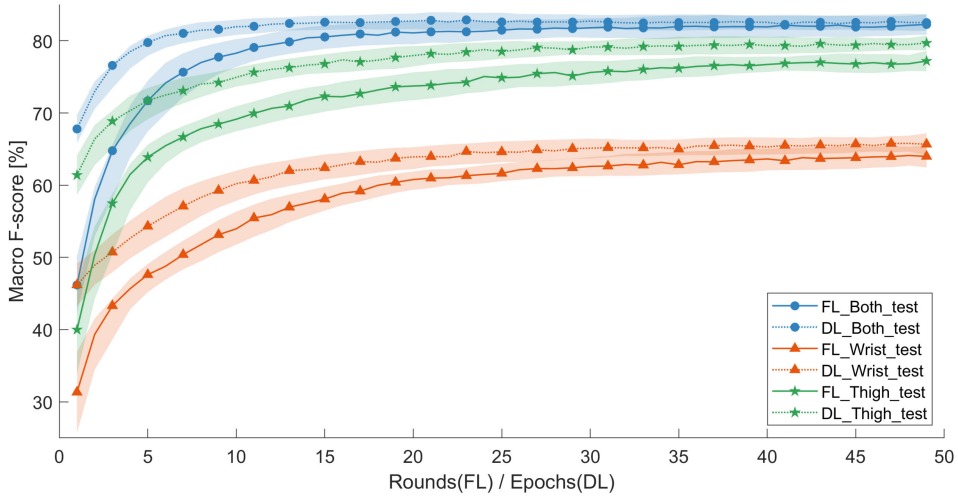
A. SENSOR PLACEMENT IMPACT

Fig. 5 presents the main results from our first experiment. More specifically, Fig. 5(a) and Fig. 5(b) show the achieved macro F-score in dependence of the number of training rounds/epochs, for the DL (shown using dotted lines) and the FL models (solid lines) when using either the JSI-FOS or PAMAP dataset for training and evaluation, respectively. The three models per learning paradigm differ only in the sensor placement that provided the data they processed.

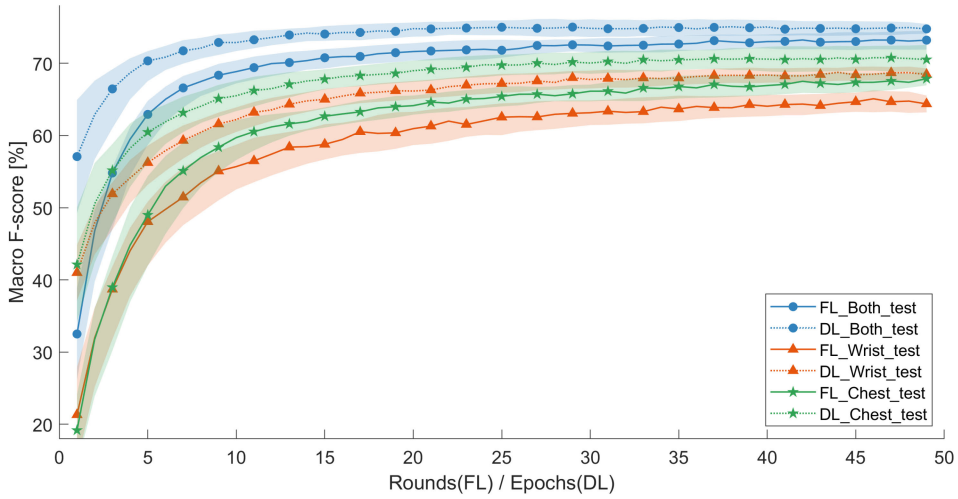
When using JSI-FOS for training and evaluation, Fig. 5(a) shows a clear ranking between the models that differ only in the sensor placement they used, regardless of whether DL or FL was used. For example, the worst performance was generated by DL and FL models that used data from the wrist sensor placement, while substantially better results were produced by those using either the thigh placement or a combination of both sensor placements. In fact, the best DL and FL models were produced using the combination of both placements. Furthermore, the results show that all models tend to plateau once the number of training epochs/rounds reaches 20, with models that use either both sensor placements or the thigh sensor placement, converging slightly faster than the models that use the wrist sensor placement.

When comparing models based on their type, i.e., DL or FL, the results show that DL models always produced slightly better results across the whole range of training epochs/rounds when compared to the corresponding FL model. Additionally, this performance gap between the two types of models seems to remain almost constant across the whole range of epochs/rounds, with the exception of the case when DL and FL models are trained on data from both sensor locations and when the number of epochs/rounds is above 20. It is also evident that these models behave very similarly and usually generate test macro F-score curves that have nearly identical shapes, with FL models taking a slightly larger number of rounds to achieve their best performance.

The results presented in Fig. 5(b) indicate that using PAMAP as the dataset for model training and evaluation yields similar outcomes. It is worth noting that models of the same type maintain a consistent ranking. In particular, deep learning (DL) and federated learning (FL) models that utilize data from both sensor locations perform better than those using data from the chest alone, which in turn perform better than those using data solely from the dominant wrist. However, a key difference when training and evaluating on the PAMAP dataset is that the gap in the performance between models trained on wrist sensor data and models trained using the chest location or data from multiple sensor locations is substantially smaller compared to that which is present when using the JSI-FOS dataset. For instance, the FL model trained on chest data performs worse than the DL model trained on wrist data, which is not observed in the case of using the JSI-FOS dataset. We hypothesize that this discrepancy arises because data from the chest sensor placement is inherently less informative for predicting the target activities compared to data coming from a sensor placed at the user's thigh.



(a) JSI-FOS dataset



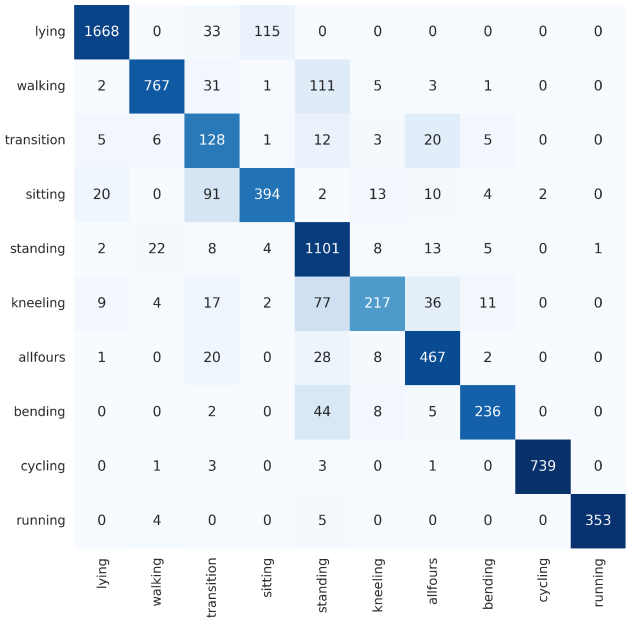
(b) PAMAP dataset

FIGURE 5. Comparison of macro F-scores [%] between DL and FL models at varying numbers of training epochs/rounds when using the (a) JSI-FOS dataset, and the (b) PAMAP dataset.

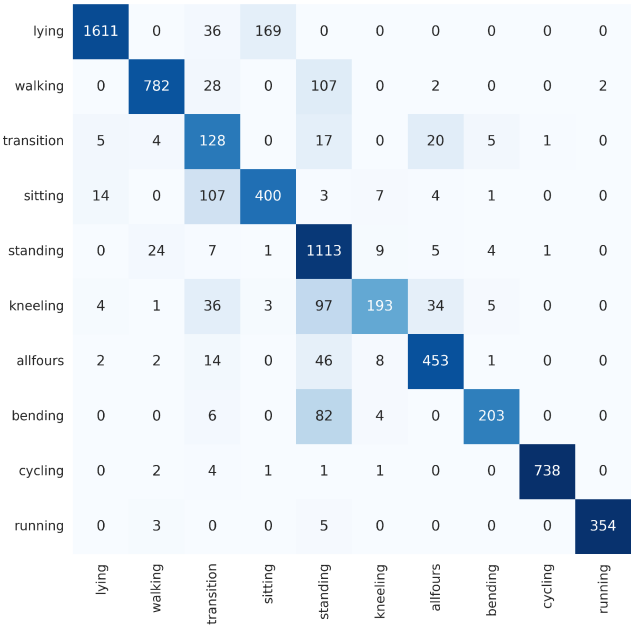
Regarding the relative behavior of DL and FL when using the PAMAP dataset, things remain unchanged. Again, DL models always produce slightly better results across the whole range of training epochs/rounds when compared to the corresponding FL model. Additionally, the performance gap between these two models seems to remain constant as the training of the model progresses. Furthermore, as was the case when using the JSI-FOS dataset, the results show that all models tend to plateau around the 20^{th} epoch/round, with models that use either both sensor placements or the chest sensor placement converging slightly faster than the models that use the wrist sensor placement. Finally, here we can once more observe that the different models produce test macro F-score curves that have nearly identical shapes.

Given that the relative performance of DL and FL models does not appear to change when using different datasets for training and evaluation, and to streamline our analysis, we decided to exclusively present the results obtained on the JSI-FOS dataset from this point forward.

Fig. 6 takes an even closer look into the relative performance of the FL models compared to the DL models. It presents two confusion matrices, generated from the predictions of a DL model and an FL model, both using data from both sensor locations for training and evaluation on the JSI-FOS dataset. By comparing the confusion matrices, we can observe that both DL and FL models exhibit very similar detection performance per activity class. Specifically, both models achieve the best performance for activities such as standing, lying, cycling and running. The worst performances are attained for activities such as kneeling. It is also interesting to note that DL and FL models make mistakes in roughly the same situations, namely, confusing lying for



(a) Deep learning



(b) Federated learning

FIGURE 6. The confusion matrices generated by the (a) DL model, and the (b) FL model, trained using both sensor placements on the JSI-FOS dataset.

sitting, walking for standing, kneeling for standing, etc. However, the FL model tends to confuse lying for sitting and bending for standing a lot more than the DL model.

B. FL OPTIMIZER IMPACT

Fig. 7 investigates the performances of different FL optimizers, i.e., the FedAdagrad, the FedYoGi and FedAvg, when combining both sensor placements. The optimizers do not undergo a hyperparameter tuning process, in order to foster a more generic and fair comparison. The results show that FedAvg provides best performances in terms of the achieved macro F-score. The figure also shows that the FedYoGi optimizer has comparable performances to FedAvg for higher number of training rounds. The worst performance is achieved by the FedAdagrad optimizer, attaining the lowest macro F-score, and exhibiting large performance oscillations. In all remaining experiments we use FedAvg, as it provides best performances.

C. IMPACT OF CLIENTS WITH HETEROGENEOUS SENSOR PLACEMENTS

Fig. 8 depicts the macro F-scores attained by FL models that were trained for 50 rounds on the JSI-FOS dataset, using varying values for the C parameter, and for different amounts of data from the two sensor placements. The quantity of data from each sensor placement is regulated by the number of users who have access to the data from that particular placement. The x-axis shows the sensor split i.e. how many clients had access to data from the wrist sensor placement (w) or the thigh sensor placement (t). The y-axis shows the achieved macro F-score. The red line presents the results of a DL model, while the rest of the lines correspond to FL models that use different values for the C parameter. It is evident that both DL and FL behave in a very similar manner. They achieve the best performances for the case when all data is derived from only the thigh (i.e. w0_t10), and achieve the worst performance when all data is derived from the only the wrist sensors (i.e. w10_t0). Moreover, it is noticeable that FL closes the performance gap to DL for the case of w10_t0.

Fig. 8 also shows that FL models require a slightly larger number of clients with data from the thigh sensor placement before achieving more substantial improvements in performance. Notably, when the data split corresponds to w10_t0, w9_t1, w8_t2, w7_t3, or w6_t4 FL models perform at a level close to the maximum achieved by models that use only wrist sensor data in the first experiment (Fig. 5(a)). It is only when at least five users have access to thigh sensor data that FL models in this experiment start to show more substantial improvement.

Our hypothesis is that the FL models' inability to leverage data from a potentially more informative sensor placement results from the fact that, during each training round, only a subset of clients (four, six, or eight) contribute their data for training, thereby limiting the models' exposure to the entire training set and hindering their ability to properly adapt to using data from two different sensor placements. However, the results from the first experiment (Fig. 5(a)) demonstrate that when models have access to twice the amount of training data, they can more easily utilize data from two sensor placements and generate superior results. Thus, a possible solution to mitigate these negative effects is to involve clients who can provide more data than those included in these experiments.

It is also interesting to note that there does not seem to be a noticeable advantage to using any of the investigated C values over the other, as they perform comparable to one another across the whole range of possible training data compositions.

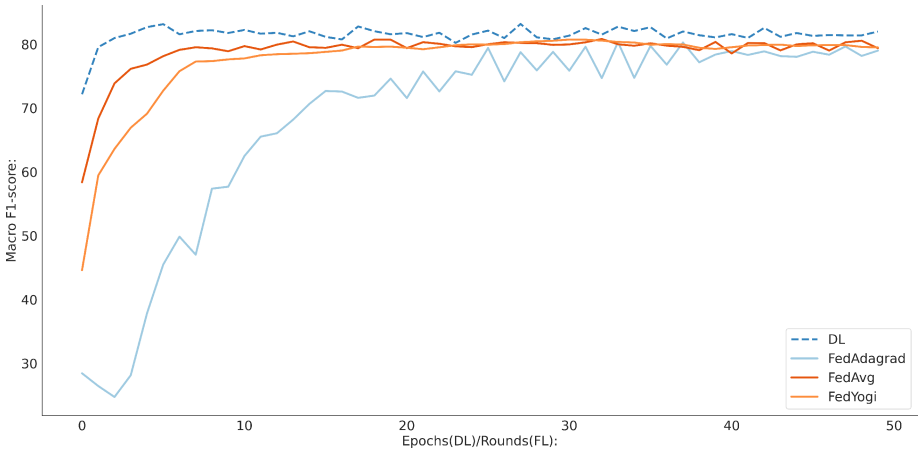


FIGURE 7. Macro F1-scores [%] of different FL optimizers.

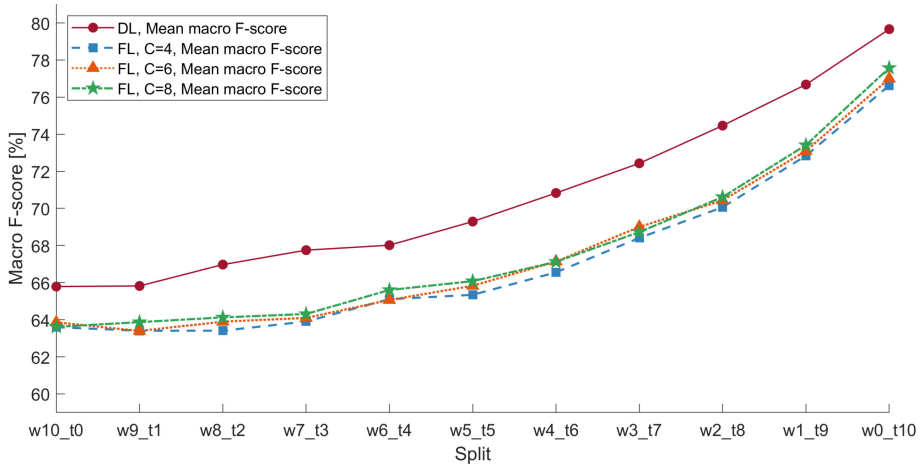


FIGURE 8. Macro F-scores [%] achieved by DL and FL models using different compositions of the training data.

Additionally, the analysis in this section focuses on the statistical behavior of the FL models. Fig. 9 shows the statistical performances of FL models (mean and 95% confidence interval) that had been trained for either 10, 30 or 50 rounds, that used eight clients for training in each round ($C = 8$), and that used different ratios of clients with heterogeneous sensor placements. The results reveal a substantial performance gap depending on the number of rounds chosen for training. Specifically, opting for a low number of rounds, such as 10, yields relatively poor results in terms of mean macro F-score values, whereas a higher value like 30 or 50 leads to better performance. However, the difference between choosing 30 and 50 rounds for training is small, consistent with the results from the first experiment, where the models tend to plateau in performance after the 20^{th} round. Furthermore, choosing a larger number of rounds for training (e.g., above 30) and/or using only thigh sensor data, yields results with a lower standard deviation (i.e. smaller 95% confidence interval).

D. BANDWIDTH EFFICIENCY ANALYSIS

The results of our analysis regarding bandwidth efficiency are presented in Fig. 10(a). More specifically, Fig. 10(a) shows a comparison between DL and FL models that use different amounts of training data from a full-featured version of the JSI-FOS dataset. As a distributed learning strategy, FL, transfers the model weights to the centralized server in each round of operation. In contrast, for DL, the data needs to be completely transferred to the central server to perform the training of the model. The FL-based macro F-score curves are presented as continuous with respect to data transfer volume and the DL results are depicted as discrete points on the macro F-score vs. data transfer volume plots.

In terms of the FL performances, Fig. 10(a) shows that the FL strategy with one active client per round ($C = 1$) can achieve the near optimal macro F-score with about 15MB of data transferred, while FL with five and nine active clients per round needs ∼30 and ∼45MB, respectively, to achieve the near-optimal macro F-scores. The FL results also show

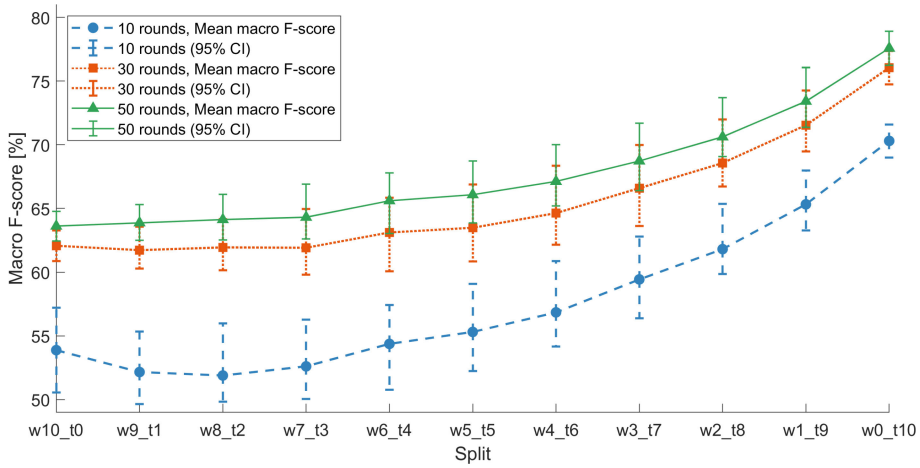


FIGURE 9. Impact of the number of training rounds on a FL model's ($C = 8$) macro F-score [%] performance for different compositions of training data from the JSI-FOS dataset.

that the confidence intervals for the macro F-score decrease as the number of active clients increases, meaning that a bit of bandwidth efficiency needs to be sacrificed for an increased stability of the FL models. In conclusion, there is a clear trade-off between the bandwidth efficiency, model accuracy and model stability for the FL strategy.

Fig. 10(a) also depicts the DL results for the macro F-scores and confidence intervals vs. the data transfer volume. It is clear that the DL model using only 10% of the dataset for training is outperformed by all FL scenarios in terms of bandwidth efficiency. The DL model trained with 50% of the dataset, shows slightly better macro F-scores at the price of a wider confidence interval (lower model stability) than FL with a larger number of active clients per round ($\geq$5).

E. MODEL COMPLEXITY AND THE EFFECTS OF FEATURE SELECTION

The results of our analysis regarding model complexity are presented in Fig 10(b). They are consistent with the ones presented in Section VI-D. The data volumes are reduced in compliance with equations 1 and 2.

Comparing the results between Fig. 10(a) and Fig. 10(b), there is a significant improvement of the bandwidth efficiency of the FL strategy. In particular, FL with one active client per round needs about 4MB to achieve a near-optimal macro F-score. FL with a higher number of clients (five and nine) does not converge in the inspected data volume range. The increase of the bandwidth efficiency comes at the price of a reduced model accuracy. Comparing Fig. 10(a) and Fig. 10(b), there is a noticeable drop in performances for the FL strategy. There is about a 5% drop in macro F-score at a lower number of rounds, as well as a noticeable increase in the confidence intervals (model instability) for all inspected FL use-cases ($C = 1, 5, 9$).

On the contrary, the DL strategy preserves the macro F-score performances with the reduced feature set, compared to DL with the full feature set (Fig. 10(a)). These are the only differences: an increase in the confidence interval for DL trained with 10% of the dataset and a slight increase in macro F-score for the DL trained with 50% of the dataset. DL with the reduced feature set (100 features) provides a dominant bandwidth efficiency, i.e., a macro F-score of $\approx$0.83 for 5MB of data volume transferred.

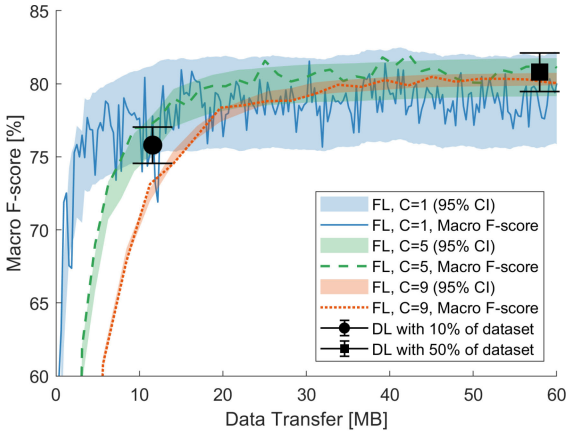
In conclusion, DL with optimized feature set might come as a satisfactory solution for bandwidth efficient ML for HAR. However, the online principle of operation, privacy preservation, reasonable performances and bandwidth efficiency, still remain the main benefits of the FL strategy. Furthermore, the drop in macro F-score performances of FL with reduced feature set may come as a result of the low number of epochs used to train the local FL models (=5), i.e., the inability of the local models to converge for the reduced feature set. The optimization of these aspects will be part of the authors' future work.

F. EFFECTS OF DATA WITH CORRUPTED LABELS

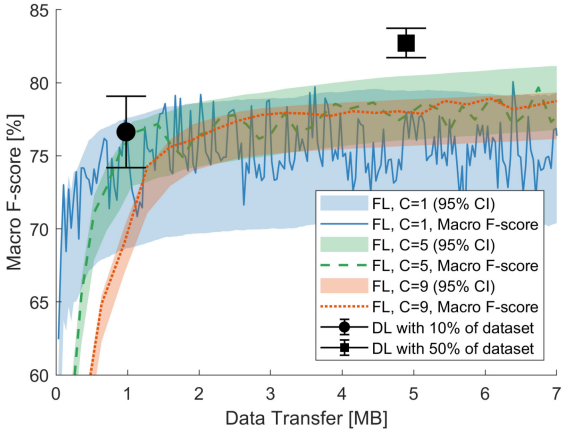
The results of our analysis into the effects of data with corrupted labels are presented in Fig. 11.

A general observation is that DL is more vulnerable to this phenomenon than the FL models. It is intuitive that the increase of percentage of incorrect labels will decrease the macro F-score of the DL model, which is also confirmed by the results. Furthermore, as the number of epochs grows, the DL performances drop even more significantly, since the model has more opportunities to fine-tune to data with incorrect labels.

On the opposite, the FL strategy is more robust to label errors, dropping only 1-4% in macro F-score as the percentage of label errors grows to 20%, depending on the number of active clients. It is also clear that FL with more active clients ($C = 6$) is more robust to label errors. This is mostly due to the online operation and the weight averaging principle of the FL strategy. This is a very important advantage of the FL paradigm, since in real-world scenarios flawed or imprecise



(a) Full-featured dataset



(b) 100 most important features

FIGURE 10. A comparison between FL and DL in terms of bandwidth efficiency, i.e., macro F-score vs. data transfer volume to achieve the respective scores.

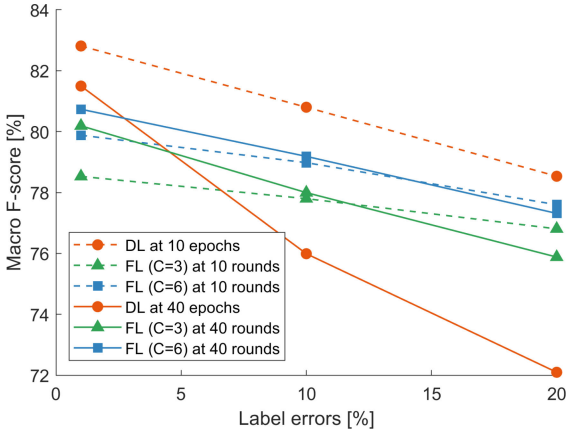


FIGURE 11. Impact of label errors on the macro F-score performance of FL and DL models.

data might seriously degrade performances. In each round of operation, the global FL model is calculated based on averaging local models from a random subset of clients. This means that erroneous local FL model weights are averaged out with more accurate ones and the effect of error propagation is diminished.

VII. LESSONS LEARNED

This section summarizes our observations from the multiple performed experiments related to FL for HAR using wearable sensors. The following lessons can be learned based on our findings:

1) **Federated vs. deep learning general observations**. It has been consistently observed that DL models outperform FL models in terms of classification performance. This is intuitive from an information theory perspective, since distributed learning cannot achieve higher accuracy than a centralized DL model when they use the same underlying neural network. DL models are trained on the entire dataset, while FL models only train local models on portions of the dataset and then combine them into a global model, which can result in valuable information being lost due to partitioning and averaging. However, this trade-off is necessary for increased privacy and data protection. After examining various use cases, it has been found that the macro F-score performance gaps between FL and DL typically range between 5-10% for the region of a lower number of rounds/epochs (<15), and below 5% for larger numbers of rounds/epochs (>20). These results and all previously discussed conclusions are consistent for two different datasets, namely the JSI-FOS and the PAMAP datasets, that were analysed using the same pipeline. Given that the performance of DL models serves as an upper limit to the performance of the FL models and the gap are not so significant when the models are trained in a sufficient number of epochs/rounds, choosing the appropriate neural network architecture is a critical step that can greatly impact the performance of FL. The goal should be to select a neural network architecture that maximizes the upper limit, thereby pushing the FL classification performances as high as possible.
2) **FL optimizer impact**. We conducted an investigation to compare the performance of various FL optimizers, namely FedAdagrad, FedYoGi, and FedAvg, for the purpose of HAR. The findings showed that despite its simplicity, FedAvg outperformed the other optimizers in terms of both convergence and macro F-score. Further investigation is required in this area since some of the optimizers were used with default initialization parameters.
3) **Sensor placement impact**. The impact of sensor placement on model performance highlights the importance of careful selection of sensor placements for accurate recognition of different activities. The results from the analysis in this paper show that the thigh (JSI-FOS dataset) and the chest placement (PAMAP dataset) prove to be more informative regarding HAR in comparison to the wrist sensor placement. It is important to note that this observation is true for both DL and FL models. Furthermore we observed that, the combination of either thigh or chest sensor data with wrist

sensor data yielded the best performances in terms of macro F-scores, again, for both the DL and the FL models. This is due to the fact the wrist sensor can contribute to better performances for some specific type of activities. The DL and FL results, as well as the gaps between the DL and the FL models are consistent for the two investigated datasets.

4) **Clients with heterogeneous sensor placements**. The experiment conducted on clients with heterogeneous sensor placements revealed that compared to DL models, FL models needed a slightly higher number of clients that have access to data from the more informative sensor placement before they are able to start leveraging this data source and improve their results. In addition, our results also showed that, when using clients with data from heterogeneous sensor placements, choosing one C value (fraction of clients) over the others does not make much sense as there was no substantial difference between their results.
5) **Bandwidth efficiency**. Regarding bandwidth efficiency, FL demonstrated better performance than DL by achieving a nearly optimal macro F-score with the transfer of only tens of megabytes of data. The investigation also looked into the C parameter and revealed that increasing the number of active clients per round led to improved model stability but required more data to be transferred for the FL models to converge. In other words, the study highlighted a clear trade-off between bandwidth efficiency, model accuracy, and model stability for the FL paradigm.
6) **Model complexity and feature selection**. The experiment used the Recursive Feature Elimination (RFE) to select the best 100 features, and the models were trained and tested on these 100 features. The results showed a substantial improvement in the bandwidth efficiency of the FL strategy when compared to the full feature set, with a 4MB data volume needed for near-optimal macro F-score. However, this increase in bandwidth efficiency came at the cost of reduced model accuracy, with a noticeable drop in macro F-score and an increase in confidence intervals for all inspected FL use cases. The main conclusion is that DL with optimized feature sets may be a satisfactory solution for bandwidth-efficient ML for HAR, but FL still remains the main choice for online operation, privacy preservation, and reasonable performances.
7) **Erroneous data effect**. The experiment compared the performance of FL to that of traditional DL when working with a dataset that has a varying percentage of erroneous labels. The results of the experiment show that the DL model is more vulnerable to label errors than the FL model. This finding highlights the advantage of FL in mitigating the effect of erroneous data, limiting the error propagation due to the averaging process for the global model update.

The previously discussed conclusions and lessons learned can serve as valuable and comprehensive guidelines for designing, developing and implementing efficient federated learning solutions for human activity recognition. Most of the conclusions are also generalizable to other federated learning applications beyond human activity recognition.

VIII. CONCLUSION

This paper presents a performance analysis for FL-based HAR, from a system level perspective and under various real-world conditions, such as communication cost/bandwidth efficiency, model complexity, erroneous data, etc. The analysis also provides a head-on comparison between FL and DL when using two different datasets. The results clearly show that various system parameters and configurations like the type of sensor placement, FL optimizer, model complexity, data volume as well as erroneous data can play a crucial role in the robustness and applicability of FL-based HAR.

Future work will focus on several different optimality and optimization aspects that will build upon the findings from this work. Specifically, the future work will investigate the analytical tractability and generalization of the optimization problem related to system-level parameters, including bandwidth efficiency, energy efficiency, model complexity, and the model performance. Additionally, it will broaden the analysis of the erroneous data effect, by including non-iid data points, noising of the data samples, as well as label smoothing.

REFERENCES

[1] M. Luštrek et al., "A personal health system for self-management of congestive heart failure (HeartMan): Development, technical evaluation, and proof-of-concept randomized controlled trial," *JMIR Med. Informat.*, vol. 9, no. 3, Mar. 2021, Art. no. e24501.
[2] I. Kiprijanovska, H. Gjoreski, and M. Gams, "Detection of gait abnormalities for fall risk assessment using wrist-worn inertial sensors and deep learning," *Sensors*, vol. 20, no. 18, p. 5373, Sep. 2020.
[3] I. Husain and D. Spence, "Can healthy people benefit from health apps?" *BMJ, Clin. Res. Ed.*, vol. 350, p. h2520, Apr. 2015.
[4] Y. Chen, X. Qin, J. Wang, C. Yu, and W. Gao, "FedHealth: A federated transfer learning framework for wearable healthcare," *IEEE Intell. Syst.*, vol. 35, no. 4, pp. 83–93, Jul./Aug. 2020.
[5] Y. Chen, X. Yang, B. Chen, C. Miao, and H. Yu, "PdAssist: Objective and quantified symptom assessment of Parkinson's disease via smartphone," in *Proc. IEEE Int. Conf. Bioinf. Biomed. (BIBM)*, Kansas City, MO, USA, Nov. 2017, pp. 939–945.
[6] M. Lee, A. M. Khan, J. Kim, Y. Cho, and T. Kim, "A single tri-axial accelerometer-based real-time personal life log system capable of activity classification and exercise information generation," in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol.*, Buenos Aires, Argentina, Aug. 2010, pp. 1390–1393.
[7] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. AISTATS*, Ft. Lauderdale, FL, USA, vol. 54, Apr. 2017, pp. 1273–1282.
[8] K. Kirsten, B. Pfitzner, L. Löper, and B. Arnrich, "Sensor-based obsessive-compulsive disorder detection with personalised federated learning," in *Proc. 20th IEEE Int. Conf. Mach. Learn. Appl. (ICMLA)*, Pasadena, CA, USA, Dec. 2021, pp. 333–339.
[9] S. Ek, F. Portet, P. Lalanda, and G. Vega, "A federated learning aggregation algorithm for pervasive computing: Evaluation and comparison," in *Proc. IEEE PerCom*, Kassel, Germany, Mar. 2021, pp. 1–10.

[10] R. Presotto, G. Civitarese, and C. Bettini, "Semi-supervised and personalized federated activity recognition based on active learning and label propagation," *Pers. Ubiquitous Comput.*, vol. 26, no. 5, pp. 1281–1298, Oct. 2022.
[11] L. Tu, X. Ouyang, J. Zhou, Y. He, and G. Xing, "FedDL: Federated learning via dynamic layer sharing for human activity recognition," in *Proc. ACM SenSys*, Coimbra, Portugal, Nov. 2021, pp. 15–28.
[12] C. Li, D. Niu, B. Jiang, X. Zuo, and J. Yang, "Meta-HAR: Federated representation learning for human activity recognition," in *Proc. Web Conf.*, Ljubljana, Slovenia, Apr. 2021, pp. 912–922.
[13] G. K. Gudur and S. K. Perepu, "Resource-constrained federated learning with heterogeneous labels and models for human activity recognition," in *Proc. DL-HAR*, Kyoto, Japan, Jan. 2021, pp. 57–69.
[14] X. Ouyang, Z. Xie, J. Zhou, J. Huang, and G. Xing, "ClusterFL: A similarity-aware federated learning system for human activity recognition," in *Proc. ACM MobiSys*, Jun. 2021, pp. 54–66.
[15] X. Zhou, W. Liang, J. Ma, Z. Yan, and K. I. Wang, "2D federated learning for personalized human activity recognition in cyber-physical-social systems," *IEEE Trans. Netw. Sci. Eng.*, vol. 9, no. 6, pp. 3934–3944, Nov. 2022.
[16] K. Sozinov, V. Vlassov, and S. Girdzijauskas, "Human activity recognition using federated learning," in *Proc. IEEE ISPA/IUCC/BDCloud/SocialCom/SustainCom*, Melbourne, VIC, Australia, Dec. 2018, pp. 1103–1111.
[17] H. Cho, A. Mathur, and F. Kawsar, "Device or user: Rethinking federated learning in personal-scale multi-device environments," in *Proc. ACM SenSys*, 2021, pp. 446–452.
[18] S. Ek, F. Portet, P. Lalanda, and G. Vega, "Evaluation of federated learning aggregation algorithms: application to human activity recognition," in *Proc. ACM UbiComp-ISWC*, 2020, pp. 638–643.
[19] S. Kozina, H. Gjoreski, M. Gams, and M. Luštrek, "Three-layer activity recognition combining domain knowledge and meta-classification," *J. Med. Biol. Eng.*, vol. 33, no. 4, pp. 406–414, Jan. 2013.
[20] H. Gjoreski, B. Kaluža, M. Gams, R. Milić, and M. Luštrek, "Context-based ensemble method for human energy expenditure estimation," *Appl. Soft Comput.*, vol. 37, pp. 960–970, Dec. 2015.
[21] A. Reiss and D. Stricker, "Introducing a new benchmarked dataset for activity monitoring," in *Proc. 16th Int. Symp. Wearable Comput.*, Newcastle, U.K., Jun. 2012, pp. 108–109.
[22] A. Reiss and D. Stricker, "Creating and benchmarking a new dataset for physical activity monitoring," in *Proc. PETRA*, Crete, Greece, Jun. 2012, pp. 1–8.
[23] S. Kalabakov, S. Stankoski, I. Kiprijanovska, A. Andova, N. Reščič, V. Janko, M. Gjoreski, M. Gams, and M. Luštrek, "What actually works for activity recognition in scenarios with significant domain shift: Lessons learned from the 2019 and 2020 sussex-huawei challenges," *Sensors*, vol. 22, no. 10, p. 3613, May 2022.
[24] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. AISTATS*, Ft. Lauderdale, FL, USA, vol. 15, 2011, pp. 315–323.
[25] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.
[26] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[27] J. Wang, Z. Xu, Z. Garrett, Z. Charles, L. Liu, and G. Joshi, "Local adaptivity in federated learning: Convergence and consistency," 2021, *arXiv:2106.02305*.
[28] I. Tenison, S. A. Sreeramadas, V. Mugunthan, E. Oyallon, E. Belilovsky, and I. Rish, "Gradient masked averaging for federated learning," 2022, *arXiv:2201.11986*.
[29] H. Mcmahan, E. Moore, D. Ramage, and B. Yarcas, "Federated learning of deep networks using model averaging," 2016, *arXiv:1602.05629*.

STEFAN KALABAKOV received the B.Sc. degree in computer technologies and engineering from the Faculty of Electrical Engineering and Information Technologies (FEEIT), Skopje, North Macedonia, and the M.Sc. degree from the Joef Stefan International Postgraduate School, Ljubljana, Slovenia. He is currently pursuing the Ph.D. degree. He is also a Research Assistant with the Digital Health-Connected Healthcare Group, Hasso Plattner Institute (HPI), Germany. His research interests include federated learning, electronic health records, and human activity recognition.

BORCHE JOVANOVSKI received the B.Sc. degree in electrical engineering and information technologies and in the field of telecommunications and the M.Sc. degree in electrical and information technology and in the field of wireless systems, services and applications from the Faculty of Electrical Engineering and Information Technologies (FEEIT), Ss. Cyril and Methodius University in Skopje (UKIM), Skopje, Macedonia, in 2019 and 2021, respectively, where he is currently pursuing the Ph.D. degree in electrical engineering and information technologies. He is also a Research Associate and part of the Laboratory for Wireless and Mobile Networks, UKIM in Skopje. His research interests include wireless networks, wireless communications, cloud computing, and recently application of machine learning and federated learning in different domains.

DANIEL DENKOVSKI is currently an Associate Professor with the Faculty of Electrical Engineering and Information Technologies, Ss. Cyril and Methodius University in Skopje. His major research interests include concentrated on signal processing, information theory, wireless communications, cloud computing, and recently machine learning and federated learning and their application in different domains. He has notable research experience having worked on 12 internationally funded research projects (FP7, H2020, and NATO SpS) and several domestic projects in his research areas. Besides theoretical research, he has serious prototyping experience, which has resulted in several awarded ICT system prototypes. He has more than 60 publications, out of which 16 in top journals with impact factor, and seven chapters in Springer books. He was a recipient of the Award "Best Young Scientist" for 2014 from the President of the Republic of Macedonia.

VALENTIN RAKOVIC (Senior Member, IEEE) received the Dipl.-Ing., M.Sc., and Ph.D. degrees in telecommunications from the Faculty of Electrical Engineering and Information Technologies, Ss Cyril and Methodius University in Skopje (UKIM), in 2008, 2010, and 2016, respectively. He currently holds the position of an Associate Professor and the Head of the Laboratory for Wireless and Mobile Networks, Faculty of Electrical Engineering and Information Technologies (FEEIT), UKIM in Skopje. He has coauthored more than 70 publications in international conferences and journals. His research interests include wireless networks, signal processing, optimization theory, machine learning, and the prototyping of wireless networking solutions.

BJARNE PFITZNER received the M.Eng. degree in computing from the Imperial College London. He is currently pursuing the Ph.D. degree with the Digital Health—Connected Healthcare Group, Hasso Plattner Institute (HPI), Germany. He is also a Research Assistant with the Digital Health—Connected Healthcare Group, HPI. For the last four years, he worked in the area of federated learning with a focus on privacy-preserving algorithms using differential privacy and healthcare applications, such as medical imaging and risk stratification for the intensive care unit.

ORHAN KONAK received the Diploma degree in mathematics and the M.Sc. degree in mathematics—computational engineering from the University of Applied Sciences Berlin (BHT), Germany. He is currently pursuing the Ph.D. degree with the Digital Health—Connected Healthcare, Hasso Plattner Institute (HPI), Germany. His research interests include sensor-based human activity recognition, especially in developing new algorithms to determine the optimal sensor placement and augmenting given datasets through synthetically generated data.

BERT ARNRICH is currently the Head of the Chair Digital Health—Connected Healthcare, joint Digital-Engineering Faculty, Hasso Plattner Institute (HPI), and the University of Potsdam. He has been a PI in several European and national research projects. He studied "Informatics in the Natural Sciences." In his Ph.D. thesis, he implemented an early big data approach that collects and consolidates patient data for scientific data analysis. At ETH Zurich, he established and headed the Wearable Computing Laboratory, Research Group Pervasive Healthcare. He received a Marie Curie Cofound Fellowship from the European Union and was appointed to tenure track professorship with the Computer Engineering Department, Bosporus University. He was the Science Manager of Emerging Technologies with Accenture Technology Solutions.

HRISTIJAN GJORESKI received the M.Sc. and Ph.D. degrees in information and communication technologies from the Jozef Stefan Postgraduate School, Slovenia, in 2011 and 2015, respectively. From 2010 to 2016, he was a Researcher with the Department of Intelligent Systems, Jozef Stefan Institute, Slovenia. In 2017, he was a Postdoctoral Research Fellow with the University of Sussex, U.K. Currently, he is an Associate Professor with the Ss. Cyril and Methodius University in Skopje, North Macedonia. His research interests include artificial intelligence, machine learning, and wearable computing. He was highly successful at several machine learning competitions and received first place award at the EvAAL Activity Recognition Challenge, in 2013, ChallengeUP Fall Detection Competition, in 2019, and Emteq Activity Recognition Challenge at Ubicomp 2019 in London, U.K. He has received the award "Best Young Scientist" for 2016 from the President of Republic of Macedonia and was selected in the top-cited 2% scientists worldwide, in 2021.