Light Weight Human Activity Recognition using Raspberry PI IoT Edge and Reduced Features from Smartphones

Ayman Wazwaz
Computer Engineering Department, College of Information Technology and Computer Engineering, Palestine Polytechnic University, Palestine
aymanw@ppu.edu

Khalid M. Amin
Department of Information Technology, Faculty of Computers and Information, Menofia University, Shebin El Kom, Egypt
k.amin@ci.menofia.edu.eg

Noura Semari
Department of Information Technology, Faculty of Computers and Information, Menofia University, Shebin El Kom, Egypt
noura.samri@ci.menofia.edu.eg

Tamer Ghanem
Department of Information Technology, Faculty of Computers and Information, Menofia University, Shebin El Kom, Egypt
tamer.ghanem@ci.menofia.edu.eg

***Abstract*— Different applications used cloud computing, machine learning, and the Internet of Things (IoT). Transferring data from the local network to the cloud for processing causes huge traffic and delay. IoT services, like Human Activity Recognition (HAR), use IoT edge options to be near the place of telemetry data generation that decreases traffic and speeds up the results. This study used three smartphones with built-in accelerometers; three parameters from each accelerometer to predict human activities. While building the models at the Raspberry PI edge, the most important features were determined using Principal Component Analysis (PCA). Light GBM, Extra Trees, and Random Forest algorithms were employed to evaluate the best models. Significant performance improvements in training and real-time results were achieved using the top related features at the IoT edge. The Light GBM recognized four different activities with 99.6% accuracy when all nine features were used, and with more than 98% accuracy when less than half of the features were used. To process one prediction, Raspberry PI 3 took 6.1 milliseconds, Raspberry PI 4 took less than 3 milliseconds if all features are used, while Microsoft Azure cloud took 5.8 seconds, including prediction time and network latency.**

Keywords— ***Internet of Things, IoT Edge, Raspberry PI, Human Activity Recognition, Feature Selection, Machine Learning.***

I. INTRODUCTION

Internet of Things (IoT) provides a framework that enables devices to connect to the Internet and collect information about their environments. IoT and Machine Learning (ML) are used in smart systems such as smart healthcare, smart energy management, smart cities, and smart machines [4, 10].

Human Activity Recognition (HAR) using smartphones is a popular and cost-effective approach, used to automatically identifying and classifying human activities. Smartphones have sensors like inertial measurement unit (IMU) that contains magnetometers, accelerometers, and gyroscopes. These sensors can be used to gather data about human activities [6, 8].

HAR using smartphones can be used in healthcare where it can be employed for monitoring and predicting health-related behaviors such as physical activities, sleep, and medication adherence [24, 26-30]. Fitness tracking and monitoring physical activity levels are also applications of HAR; this can be useful for individuals who want to improve their fitness and overall health. In sports and performance tracking, HAR can be used to track and analyze sports performance, including running, cycling, and swimming. This can help athletes to track their progress, set goals, and improve their performance [18 ,32]. Overall, HAR using smartphones has the potential to provide valuable insights into human behavior and can be used in a variety of smart applications [16, 32-35, 38-40].

In cloud computing systems, where there are plenty of resources, machine learning can be used to process data using processing power, storage, and memory as needed. However, network delay from the local environment to the servers and responses back increases the volume of data and adds latency in applications that consume a lot of traffic [4, 10-12].

In order to stay up with application needs, machine learning needs to move from being managed in the cloud to being handled closer to end user devices at the IoT edge. Smart systems apply IoT edge devices to offload work from the cloud to the IoT edge, where requests are handled close to the environment [36-37].

In this paper, two versions of Raspberry PI micro-computers were used as edge devices to speed up response time while maintaining accuracy. Depending on the significance of each feature, a different number of features was employed. Nine values were used, and the accuracy was 99.6% across all three smartphones. By reducing the number of features and exploiting the IoT edge, prediction time was significantly improved without having noticeable effects on accuracy. Accuracy was 94% in real-time testing and 99% during training with only five features.

The related articles review and datasets are covered in the next sections. Following an explanation of the methodology and experiments, the findings will be examined and contrasted, and finally, comparisons and conclusions will be presented.

II. LITERATURE REVIEW

Smartphones were utilized in [1, 5, 7, 17], the authors of these articles suggested HAR systems using deep learning and data from smartphone sensors to detect human body motion. In these studies, machine learning was used to train over data from smartphone inertial sensors to distinguish a range of behaviors, including standing, running, walking, leaping, and sleeping, as well as actions that take place in between distinct activities. The findings demonstrate that human behaviors can be consistently identified.

A cloud-based method for detecting falls in older persons was employed in [11], and the data was transferred there to be categorized and utilized to develop a profile of the person being tracked. On the basis of data collected and processed by the smartphone's accelerometer and gyroscope, fall detection was implemented on smartphones. Wearable sensor data was handled by authors in [9, 14, 20-21, 31, 41] to carry out classification. They utilized the derived patterns over a variety of time scales using additional hardware like microcontrollers and various classification algorithms to differentiate between activities.

Authors in the field of internet-based healthcare presented a three-dimensional inertia signal with thirteen time-stamped human movements, including walking, walking upstairs, walking downstairs, writing, smoking, and other activities [8], the HAR model was provided using the Random Forest classifier and effective handcrafted features. A new machine learning (ML) approach for HAR systems was proposed by researchers in [17], which entails data gathering, data cleaning, feature extraction, feature engineering, and modeling with classification algorithms for forecasting human activities. They compared the performance of traditional ML algorithms and tree-based boosting techniques while using motion sensors to detect human movements.

In [5], researchers proposed a two-level scheme for recognition of human physical activities and the corresponding contexts based on the smartphone accelerometer data. The suggested method consists of four steps that are based on accelerometer data: data pre-processing, feature analysis, activity detection, and context classification. Cooking, and in a meeting are examples of context recognition. Other articles related to HAR contexts recognition are discussed in [26-28].

According to the earlier articles, embedded sensors in smartphones and wearable sensors are frequently utilized in machine learning, and they are being used to identify many types of daily activities. While some recent research publications did focus on real-time experiments, the majority of them concentrated on increasing recognition accuracy and updating algorithms to enhance processing speed. Some articles used edge computing, but the speed and accuracy were not equivalent as in this work. Table 10 at the end of the results section compares the outcomes of this proposal and the outcomes of some recent related articles.

This paper used two versions of Raspberry PI IoT edge, to enhance the performance of HAR using several models, Microsoft Azure cloud infrastructure and three smartphones were used. Utilizing the built-in accelerometer of three smartphones, HAR used the features created from the accelerometer readings. Depending on the relative importance of each value, various feature datasets were used. In addition to the cloud, numerous tests were created and conducted at the IoT edge close to mobile devices. At IoT edge, processing speed for several models were measured. We discovered that combining the IoT edge with the high impact features would greatly increase prediction speed without affecting accuracy.

III. Dataset and Methodology

A. Dataset and configurations

The basic dataset used in this work is based on a dataset created by the German university of Manheim's Data and Web Science group (DWS) [19]; they captured movements on smartphones for numerous people using various sensors, such as the accelerometer, the gyroscope, and others. The exercises include walking, running, jumping, lying, standing, and going up and down stairs. They used seven smartphones, one on each of their heads, chests, forearms, waists, shins, thighs, and upper arms. Here, each smartphone has an accelerometer, which generates three values (x, y, and z) as illustrated in Figure 1. Smartphone's orientation will make accelerometer produce different values when the body is moving [1,6].

Using just three smartphones, we were able to extract a subset of the dataset to identify four activities. Running, standing, sitting, and walking are the chosen categories among other categories. The locations of smartphones are shown in Figure 2. Keep in mind that the smartphone on the waist is horizontal, which changes how the x and y coordinates of the accelerometer are affected.

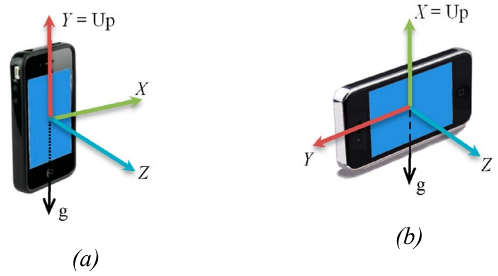

Figure 1: smartphone accelerometer [6]

Figure 2 depicts the distribution of the three places that were chosen to cover the categories that we are targeting. The waist movement is essential for the selected categories, and the head and chest were not chosen because they are mostly stable in the selected movements. Some locations also cover other locations, such as the upper arm substituting the forearm and the shin substituting the thigh. Therefore, there were only three smartphones instead of seven. The accelerometer sensor, which was selected in accordance with the four categories, has a greater impact on detecting the specified activities than the gyroscope and other sensors. However, if fall detection is required, for instance, a gyroscope will be essential to detect rotation while falling.

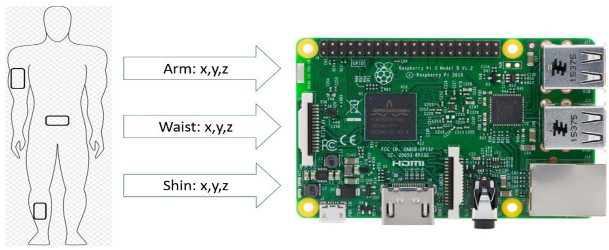

Figure 2: Three smartphones and Raspberry PI as an IoT Edge

Popular algorithms in smart systems include Extra Trees, Random Forest, and Light Gradient-Boosting Machine (Light GBM) [10]. Light GBM and Extra Trees are both widely used machine learning algorithms for classification tasks including HAR [14, 23, 39]. The Random Forest ensemble learning approach combines different decision trees to improve the accuracy and dependability of HAR models [27]. Light GBM is faster than Extra Trees and Random Forest due to its use of gradient-based boosting and histogram-based binning. This makes it well-suited for large datasets and real-time applications. These algorithms were used and compared in this paper in terms of accuracy and prediction speed.

A sample of a dataset with nine features and one projected activity is shown in Table 1. Starting with the smartphone's accelerometer on the arm provided the first three columns arm(x,y,z); the other six columns came from the smartphones located on the waist and the shin. The dataset contains 120000 entries, 30000 for each recorded category.

Table 1: A sample of HAR dataset using accelerometer for 3 smartphones and four categories

arm_x	arm_y	arm_z	waist_x	waist_y	waist_z	shin_x	shin_y	shin_z	category
-0.02753	9.977836	1.795053	9.941324	-0.42078	0.716465	-0.1227	10.04667	-0.48363	Running
-0.16101	10.0341	1.704672	9.877279	-0.44412	0.846949	-0.13827	10.04128	-0.42018	Running
-0.34058	10.10054	1.669357	9.985018	-0.3711	0.888249	-0.21189	10.05984	-0.48662	Running
-4.66391	8.535928	2.562394	9.944317	0.134075	0.64404	6.265626	7.480683	-2.74615	Sitting
-4.75189	8.607155	2.583942	9.943718	0.155623	0.637456	6.244078	7.475296	-2.73777	Sitting
-4.7495	8.542512	2.61387	9.944916	0.129885	0.612916	6.21834	7.484274	-2.71921	Sitting
-0.39085	10.13765	1.237802	9.932944	0.126294	0.387262	0.013168	10.11431	-0.38666	Standing
-0.37828	10.06463	1.238401	9.971252	0.240019	0.358532	-0.10714	10.06642	-0.44173	Standing
-0.42976	10.00118	1.182736	9.941922	0.172981	0.319626	-0.0808	10.13525	-0.36332	Standing
0.637456	12.07815	1.475427	8.647857	-0.35973	-1.59454	1.092355	7.026383	-4.89135	Walking
1.231218	10.5692	1.936909	6.501455	-1.49219	-0.26875	0.815226	7.100604	-5.06553	Walking
0.994791	8.868123	2.03148	5.975927	0.104148	0.696713	0.764349	7.782352	-4.12521	walking

In this work, Light GBM produced the highest accuracy and the fastest processing speed. In some experiments, we used Light GBM to compare the performance of models with different number of features on Raspberry PI micro-computer.

B. Methodology and Experiments

This section describes the methodology from datasets extraction, features importance and selection, training and accuracy measurements, real-time analysis, and comparing cloud and edge performance. The following methodology and experiments were used:

1. Training and accuracy:
    - The dataset with nine inputs and one output (Table 1) was extracted from the original dataset [19].
    - The importance of each feature was measured using PCA algorithm [3, 16].
    - Using subsets of the dataset based on the importance of the features, 11 models were extracted and compared.
    - Accuracy of several models was compared using various training and testing percentages.
    - The accuracy using three algorithms and the confusion matrix for the best four models were measured and compared.
2. Real-time prediction accuracy:
    - The model with nine features was used as a reference to compare with the other models in real-time experiments.
    - The best four models were tested using real-time experiments with different algorithms to measure accuracy.
3. Real-time prediction time:
    - Prediction time was measured for the top four models on IoT edge using two versions of Raspberry PI.
    - Prediction time was compared at the edge and using Microsoft Azure Cloud.

IV. Experiments and Results

A. Training and features selection

The model with nine values (Top 9 model) used all features from the accelerometers of the three smartphones. Here, Light GBM produced 99.65% accuracy compared to 99.43% for Extra Trees and 96.83% for Random Forest as shown later in Table 4.

PCA was utilized to reduce features in the context of recognizing human activities. It is known that not all of these features are equally significant for accurately categorizing human activities. The most important parameters that are responsible for the majority of the variance in the data were found. The reduced features representation for the classification job can then be created from these primary components.

In the first experiment, the variation in the importance of the features is depicted in Figure 3. Based on Figures 1 and 2, the orientation of the smartphones is either vertical or horizontal, and none of the anticipated behaviors rely on the z-component, but this is not the case if we have sleeping or falling categories. Additionally, the horizontal orientation of the smartphone switches the impact of the x and y components on the waist, making the x-component on the waist the most important feature, followed by the shin_y and arm_y, and none of the top 5 features include a z-component.

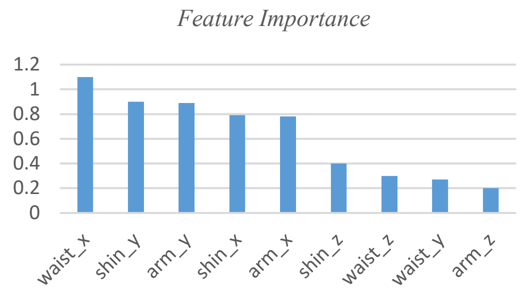

Figure 3: Feature importance for three smartphones

Based on Figure 3, various models were proposed as listed in table 2, where each model in the table is described by its features listed in the table, the accuracy was measured using Light GBM for 11 models. The models are grouped as:

- The first three models are based on one smartphone each; arm, waist, or shin.
- The next five models from Top1 to Top 5, used the most important features sequentially, Top 1 used only the most important feature, and Top 2 used the first and second most important features, and so on.
- WSS (Waist_x, Shin_x, Shin_y), and WAA (Waist_x, Arm_x, Arm_y) are models with only two smartphones, but with three features. WSS used x from the waist which is the most important feature, followed by top two features from the shin, while WAA used top two features from the arm instead. In this case, we reduced the number of smartphones as well as the number of features.
- Lastly the Top 9 model that used all features from three smartphones.

The outcomes for 11 models are represented in Table 2 and Figure 4. Results shows that the accuracy using one smartphone with three features approached 93.4% for the Arm, and around 90% using Waist or Shin, which are sufficient in some applications.

It should be mentioned that the accuracy when employing the most significant feature was close to 80%. However, only one feature (11% of the features) was utilized. Adding the next top important feature will improve the result to 93.4% which is equivalent to Arm model with three features.

Table 2: Accuracy level for 11 models using Light GBM

Model	Accuracy (%)	Features
Arm	93.4	arm_x, arm_y, arm_z
Waist	90.2	waist_x, waist_y, waist_z
Shin	89.9	shin_x, shin_y, shin_z
Top 1	79.72	waist_x
Top 2	93.4	waist_x, shin_y
Top 3	98.49	waist_x, arm_y, shin_y
Top 4	98.83	waist_x, arm_y, shin_x, shin_y
Top 5	99.14	waist_x, arm_x, arm_y, shin_x, shin_y
WSS	95.52	waist_x, shin_x, shin_y
WAA	97.48	waist_x, arm_x, arm_y
Top 9	99.65	All the 9 features from 3 smartphones

The accuracy was improving with additional features and approached 98.49% with top three features from three smartphones. Here, Top 3 model still needed the three smartphones despite using only one-third of the features to create a model with a relatively high level of accuracy. More features didn't significantly enhance performance over the Top 3 as seen in Table 2, where Top 4 accuracy was 98.83% and Top 5 accuracy was 99.14%, and they still need three smartphones.

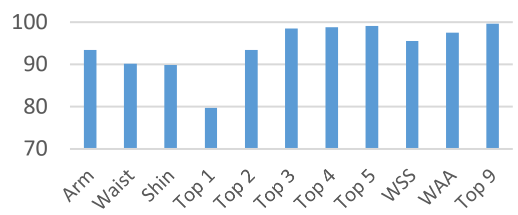

Figure 4: Accuracy for 11 models using Light GBM

Reducing number of smartphones will reduce costs and make configuration easier, WSS model as shown in Table 2 uses three features from two smartphones (Waist and Shin), and the model produced 95.52% accuracy, while WAA model produced 97.48% using three features from the Waist and the Arm. It is clear that combining features from the Arm with the Waist is better than combining features from the Shin with the Waist.

Top 9 model used all features, the accuracy was 99.65% using Light GBM, this level made it possible to consider this model as a benchmark to compare it with others in real-time experiments. Later, Top 9 was compared with Top 3, 4, and 5.

The previous experiments used 50% (60000 entries) of the dataset for training and the remaining for testing. Table 3 shows that if the percentage of the training entries are between 80% and 40%, it will not affect the accuracy significantly.

Table 3: Accuracy for different training and testing percentages using 9 features with Light GBM

Train/Test percentage	80/20	60/40	50/50	40/60	20/80	5/95
Accuracy (%)	99.67	99.66	99.65	99.62	99.49	99.14

For example, when 40% of the dataset was used for training, then 48000 entries were sufficient to capture the information from the dataset to build the model. Even 20% produced high accuracy for such a dataset. The number of predicted categories, and the nature of features made it possible to have sufficient accuracy even for training by 5% (3000 entries) of the total entries of the dataset. But, this will not be the case if more features and more categories are added.

Table 4: Accuracy percentages for the best four models using three algorithms

Model	Random Forest	Extra Trees	Light GBM
Top 3	94.69	98.16	98.49
Top 4	95.06	98.68	98.83
Top 5	96.22	98.97	99.14
Top 9	96.83	99.43	99.65

As mentioned earlier, Light GBM, Extra Trees, and Random Forest are popular algorithms in smart systems including HAR. Table 4 shows the accuracy results of using these three algorithms on the best four models shown in Figure 4. It is clear that Light GBM produced best accuracy among the other algorithms listed in the table, followed by Extra Trees algorithm.

The frequency of system confusion to find the proper category is measured by the confusion matrix. While there was little confusion between walking and running, and this confusion decreases when additional features are included, as was the case when nine features were employed, there was almost no confusion between standing and sitting on one side and running from the other side as seen in Table 5.

B. *Real –time prediction*

In this experiment, two models were running simultaneously, with Top 9 serving as the standard for comparison. To evaluate accuracy, Top 3, Top 4, and Top 5 were compared to Top 9. The results of three tests are shown in Table 6, which compares the number of categories that were accurately predicted to the overall number of messages received at the edge. It was discovered that the accuracy for Top 3 model was approximately 89% in practice, even though the same model had more than 98% in training. Top 4 performed better with 93.84%, and Top 5 model had 94.07% accuracy.

The models in practice clearly provided less accurate results than those achieved during training, where the type of smartphone and the precision of the accelerometer vary from one brand to another, and where the quantity of motions captured in a time interval also had an effect on results. The posture of the smartphone on the body affect the accuracy as well. In this case, we made use of various smartphone brands and various data rates, and obtained reasonable practical accuracy using fewer number of features.

Table 5: Confusion matrix for the best four models with Light GBM

True Label \ Predicted Label	running	sitting	standing	Walking
running	14589	1	72	312
sitting	2	14736	111	77
standing	0	41	15021	29
walking	107	32	124	14746

(a) Top3

True Label \ Predicted Label	running	sitting	standing	walking
running	14715	1	73	185
sitting	4	14781	79	62
standing	0	41	15019	31
walking	93	10	125	14781

(b) Top 4

True Label \ Predicted Label	running	sitting	standing	walking
running	14751	1	72	150
sitting	2	14842	40	42
standing	2	20	15043	26
walking	78	10	72	14849

(c) Top 5

True Label \ Predicted Label	running	sitting	standing	walking
running	14860	1	11	102
sitting	3	14885	17	21
standing	2	1	15084	4
walking	34	2	13	14960

(d) Top 9

Table 6: Real-time prediction accuracy at the edge

Model	Real-time Accuracy
Top 3	9643/10828 (89.05 %)
Top 4	8381/8931 (93.84 %)
Top 5	6270/6665 (94.07 %)

C. Prediction time at the edge and the cloud

In this part, Light GBM algorithm was used, and the best four models were deployed at the edge in real-time prediction process. Smartphones used UDP to send messages through Wi-Fi to the Raspberry PI device as shown in Figure 2, and the Raspberry PI recorded the time before and after the prediction process. Smartphones were configured to read sensors and send values every 50 milliseconds (20 Hz).

Two versions of Raspberry PI were used, the key differences between the two models are the processor and the RAM; Raspberry Pi 4 is powered by a quad-core ARM Cortex-A72 processor, is faster and more powerful than the quad-core ARM Cortex-A53 processor used in Raspberry Pi 3. Raspberry Pi 4 comes with up to 8GB of RAM, compared to 1GB available on Raspberry Pi 3 [2].

Table 7 compares between the prediction times of the two hardware models using Light GBM; the table shows the average prediction time per request for 3000 sample requests. Considering the arrival rate of messages is one every 50 milliseconds, both systems could handle predictions in reasonable times; Raspberry PI 4 was capable of predicting one request in less than 3 milliseconds, the Raspberry PI 3 could predict at a rate of less than 6.1 milliseconds for each prediction if all features were employed. In comparison to the model using nine features, Table 7 shows that less features led to faster replies. The faster replies indicates simpler models and less processing time is needed when we use fewer features.

Table 7: Average Prediction time using IoT edge for different models using Light GBM

Model	Prediction time on Raspberry PI 3 (milliseconds)	Prediction time on Raspberry PI 4 (milliseconds)
Top 3	5.811	2.387
Top 4	5.827	2.401
Top 5	5.877	2.589
Top 9	6.097	2.964

Light GBM is generally faster than Random Forest and Extra Trees due to its use of gradient-based boosting and histogram-based binning. This makes it well-suited for real-time applications. Extra Trees can be slower when dealing with high-dimensional data, as it generates more trees than Light GBM. Random Forest uses a combination of bagging and random feature subsets, which needs more time [1, 10, 27].

Table 8 compares prediction time using the three algorithms, Light GBM was the fastest among the three, it took less than 3 milliseconds using nine features on Raspberry PI 4, Extra Trees came next with 3.6 milliseconds, and Random Forest with more than 27 milliseconds. Using Raspberry PI 3, predictions took longer; it consumed about 6.1 milliseconds using Light GBM, 15.681 milliseconds using Extra Tress, and more than 166 milliseconds using Random Forest algorithm.

Table 8: Prediction time for Top 9 model using different algorithms on Raspberry PI 3 and 4

Model	Prediction time on Raspberry PI 3 (milliseconds)	Prediction time on Raspberry PI 4 (milliseconds)
Light GBM	6.097	2.964
Extra Tree	15.681	3.622
Random Forest	166.75	27.313

About cloud computing, the last experiment used Microsoft Azure cloud, the edge send requests to the cloud for processing. To compare processing speed and latency, a web service was used to deploy the Top 9 model in the cloud. The cloud server used a container with a 1 GHz CPU and 1GB of RAM in this case to handle requests. Faster cloud processing container with larger memory will speed up prediction, but processing costs per hour will also increase, but will not decrease the network latency.

There are numerous reasons that might contribute to cloud latency, such as network congestion, physical distance between the device and the cloud server, and the processing time needed by the cloud server [11].

Table 9: Azure cloud delay testing

Total time (milliseconds)	Network delay (milliseconds)	Prediction time in the cloud (milliseconds)
5834	5727	107

Here, the network between the edge and Azure cloud and inside the cloud itself caused the majority of the delay. The configuration of the container affects how quickly a request is processed in the cloud. Less than 2% of the total time in this scheme, as shown in Table 9, was consumed in prediction; the rest time was consumed by network delays caused by transmissions inside the cloud, queuing, and sending between edge devices and the cloud. The overall duration was here about 5.8 seconds per prediction, compared to a few milliseconds at the IoT edge in the previous experiments.

About scalability, if the arrival rate is one request every 50 milliseconds and nine features are employed, the Raspberry PI 4 edge can provide one prediction in 3 milliseconds, which means it can support more than 16 users. For each IoT edge, the number of users can be increased by reducing the amount of features and slowing down the arrival rate. For example if one request is sent every 250 milliseconds, and one request needed less than 2.5 milliseconds as in Top3 and Top4 models using Raspberry PI 4 as shown earlier in Table 7, then one edge can handle more than 100 users. The system may support a greater number of users by adding more IoT edges.

Table 10: Comparing results of other articles with this proposal

Reference	Sensor	Activities	ML models	Accuracy
[2] , 2020	Accelerometer, MPU6050 Using Raspberry PI edge	Walking, sitting, Running	ANN , LSTM-RNN	Training 98% Real-time 86%
[15] , 2022	IMU, and heart rate	walking, sitting walking up/down, standing, laying	CNN Fusion, GRU	96.28%
[27] , 2023	IMU	walking, driving, active status	tree-based models	93%
[22] , 2019	IMU	Walking upstairs/downstairs, sleeping, standing	LSTM, CNN	96.4%
This proposal	Accelerometer (different number of features)	Walking, sitting, Running, standing	Light GBM, Extra Trees, Random forest	Training 99.14% Real-time 94%

The outcomes of this proposal is compared in Table 10 with some recent articles and their findings; the listed works made use of smartphones and other smart sensors to identify various activities. The majority of the studies rely on signal processing with multiple types of sensors, with the main objective being to improve forecast speed or accuracy. To improve response time, they used developed algorithms, combinations of sensors, or edge computing. But this proposal produced a significant increase in speed and accuracy using only one type of sensors, and with reduced features with around half number of features without affecting accuracy. One Raspberry PI can handle scalable number of users depending on the application in use and the required rate of predictions.

V. Conclusions

Monitoring information generated by IoT devices is used in a variety of applications that operate in real time. Machine learning is used to assess and predict outcomes whereas cloud computing is utilized to store data for a range of applications. These methods are used by intelligent systems to offer services to the end users. Smartphones and other devices create a lot of sensor data for IoT systems like HAR, which requires complex computations.

A variety of situations, like healthcare and rehabilitation, where it is crucial to continuously monitor patients. When necessary, HAR systems should be quick to react, and precise. HAR may use edge machine learning to improve the process and solve challenges. In order to speed up the prediction process and lower the amount of data that needs to be transferred to the cloud, we suggested using IoT edge computing in this study with fewer features.

The accuracy of the accelerometer measurements from three smartphones utilized in this research was approximately 99.6%. By lowering the number of features and utilizing the IoT edge, prediction time decreased considerably without significantly affecting accuracy level. In real-time testing, accuracy was 94%, while during training it was 99% using just five features. The results in this proposal are more accurate in training and real-time compared to the previous publications shown in Table 10, and the prediction speed was sufficient even for tens of users on the same edge.

In practice, smartphones might be replaced by microcontrollers and IMU sensors, some kits like MPU6050 include built-in accelerometer and gyroscope. The raspberry PI was used as a small, well-known microcomputer, and the dataset used in this study was derived from data generated by smartphones. In some situations, standard computers can serve as an IoT edge, and in others, smartphones can function both as an IoT edge and a mobile sensing device.

The use of machine learning models on smartphones and wearables might be considered as an extension of this research, sensors and microcontrollers might be used to support more services. Different IoT edge devices may manage groups of users in a scalable solution, and integrate together to cover different places and use the cloud for storage and analysis.

References

[1] Christian Krupitzer, Timo Sztyler, Janick Edinger, Martin Breitbach, Heiner Stuckenschmidt, Christian Becker, Beyond position-awareness—Extending a self-adaptive fall detection system, Pervasive and Mobile Computing, Volume 58, 2019, ISSN 1574-1192, https://doi.org/10.1016/j.pmcj.2019.05.007

[2] Jmal, A., Barioul, R., Meddeb Makhlouf, A., Fakhfakh, A., Kanoun, O. (2020). An Embedded ANN Raspberry PI for Inertial Sensor Based Human Activity Recognition.. ICOST 2020. Lecture Notes in Computer Science, vol 12157. Springer, Cham. https://doi.org/10.1007/978-3-030-51517-1_34

[3] Shaeela Ayesha, Muhammad Kashif Hanif, Ramzan Talib, Overview and comparative study of dimensionality reduction techniques for high dimensional data, Information Fusion, Volume 59, 2020, Pages 44-58, ISSN 1566-2535, https://doi.org/10.1016/j.inffus.2020.01.005

[4] Mohammad Aazam, Sherali Zeadally, Eduardo Feo Flushing, Task offloading in edge computing for machine learning-based smart healthcare, Computer Networks, Volume 191, 2021,

108019, ISSN 1389-1286, https://doi.org/10.1016/j.comnet.2021.108019

[5] Muhammad Ehatisham-ul-Haq, Muhammad Awais Azam, Yusra Asim, Yasar Amin, Usman Naeem, Asra Khalid, Using Smartphone Accelerometer for Human Physical Activity and Context Recognition in-the-Wild, Procedia Computer Science, Volume 177, 2020, Pages 24-31, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2020.10.007

[6] Saeedi, S.; El-Sheimy, N. Activity Recognition Using Fusion of Low-Cost Sensors on a Smartphone for Mobile Navigation Application. Micromachines 2015, 6, 1100-1134. https://doi.org/10.3390/mi6081100

[7] Mohammed Mehedi Hassan, Md. Zia Uddin, Amr Mohamed, Ahmad Almogren, A robust human activity recognition system using smartphone sensors and deep learning, Future Generation Computer Systems, Volume 81, 2018, Pages 307-313, ISSN 0167-739X, https://doi.org/10.1016/j.future.2017.11.029

[8] Issa, Mohamed E., Ahmed M. Helmi, Mohammed A. A. Al-Qaness, Abdelghani Dahou, Mohamed Abd Elaziz, and Robertas Damaševičius. 2022. "Human Activity Recognition Based on Embedded Sensor Data Fusion for the Internet of Healthcare Things" *Healthcare* 10, no. 6: 1084. https://doi.org/10.3390/healthcare10061084

[9] Emanuele Lattanzi, Valerio Freschi, Evaluation of human standing balance using wearable inertial sensors: A machine learning approach, Engineering Applications of Artificial Intelligence, Volume 94, 2020, 103812, ISSN 0952-1976, https://doi.org/10.1016/j.engappai.2020.103812

[10] W. Y. B. Lim et al., "Federated Learning in Mobile Edge Networks: A Comprehensive Survey," in IEEE Communications Surveys & Tutorials, vol. 22, no. 3, pp. 2031-2063, 2020, https://doi.org/10.1109/COMST.2020.2986024

[11] Dariusz Mrozek, Anna Koczur, Bożena Małysiak-Mrozek, Fall detection in older adults with mobile IoT devices and machine learning in the cloud and on the edge, Information Sciences, Volume 537, 2020, Pages 132-147, ISSN 0020-0255, https://doi.org/10.1016/j.ins.2020.05.070

[12] Christine Mwase, Yi Jin, Tomi Westerlund, Hannu Tenhunen, Zhuo Zou, Communication-efficient distributed AI strategies for the IoT edge, Elsevier ,Future Generation Computer Systems, Volume 131, 2022, Pages 292-308, https://doi.org/10.1016/j.future.2022.01.013.

[13] Sen Qiu, Hongkai Zhao, Nan Jiang, Zhelong Wang, Long Liu, Yi An, Hongyu Zhao, Xin Miao, Ruichen Liu, Giancarlo Fortino, Multi-sensor information fusion based on machine learning for real applications in human activity recognition: State-of-the-art and research challenges, Information Fusion, Volume 80, 2022, Pages 241-265, ISSN 1566-2535, https://doi.org/10.1016/j.inffus.2021.11.006

[14] Jessica Sena, Jesimon Barreto, Carlos Caetano, Guilherme Cramer, William Robson Schwartz, Human activity recognition based on smartphone and wearable sensors using multiscale DCNN ensemble, Neurocomputing, Volume 444, 2021, Pages 226-243, ISSN 0925-2312, https://doi.org/10.1016/j.neucom.2020.04.151

[15] Zhang, C., Cao, K., Lu, L. *et al.* A multi-scale feature extraction fusion model for human activity recognition. *Sci Rep* 12, 20620 (2022). https://doi.org/10.1038/s41598-022-24887-y

[16] Hossein Shafizadeh-Moghadam, Fully component selection: An efficient combination of feature selection and principal component analysis to increase model performance, Expert Systems with Applications, Volume 186, 2021, 115678, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2021.115678

[17] Pratik Tarafdar, Indranil Bose, Recognition of human activities for wellness management using a smartphone and a smartwatch: A boosting approach, Decision Support Systems, Volume 140, 2021, ISSN 0167-9236, https://doi.org/10.1016/j.dss.2020.113426.

[18] Shreshth Tuli et al , HealthFog: An ensemble deep learning based Smart Healthcare System for Automatic Diagnosis of Heart Diseases in integrated IoT and fog computing environments, Future Generation Computer Systems, Volume 104, 2020, Pages 187-200, ISSN 0167-739X, https://doi.org/10.1016/j.future.2019.10.043

[19] Mannheim University, Germany - Research Group Data and Web Science , DataSet - RealWorld (HAR) https://sensor.informatik.uni-mannheim.de/#dataset_realworld

[20] Damini Verma, Kshitij RB Singh, Amit K. Yadav, Vanya Nayak, Jay Singh, Pratima R. Solanki, Ravindra Pratap Singh, Internet of things (IoT) in nano-integrated wearable biosensor devices for healthcare applications, Biosensors and Bioelectronics: X, Volume 11, 2022, 100153, ISSN 2590-1370, https://doi.org/10.1016/j.biosx.2022.100153

[21] Zhen Hong, Miao Hong, Ning Wang, Yong Ma, Xiaolong Zhou, Wei Wang, A wearable-based posture recognition system with AI-assisted approach for healthcare IoT, Future Generation Computer Systems, Volume 127, 2022, Pages 286-296, ISSN 0167-739X, https://doi.org/10.1016/j.future.2021.08.030

[22] T. Zebin, P. J. Scully, N. Peek, A. J. Casson and K. B. Ozanyan, "Design and Implementation of a Convolutional Neural Network on an Edge Computing Smartphone for Human Activity Recognition," in IEEE Access, vol. 7, pp. 133509-133520, 2019, https://doi.org/10.1109/ACCESS.2019.2941836

[23] Demrozi, F., Pravadelli, G., Bihorac, A., Rashidi, P., 2020. Human activity recognition using inertial, physiological and environmental sensors: A comprehensive survey. IEEE Access 8, 210816–210836. https://doi.org/10.1109/ACCESS.2020.3037715

[24] Sakorn Mekruksavanich, Ponnipa Jantawong, Anuchit Jitpattanakul, A Deep Learning-based Model for Human Activity Recognition using Biosensors embedded into a Smart Knee Bandage, Procedia Computer Science, Volume 214, 2022, Pages 621-627, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2022.11.220

[25] Yi Liu, Weiqing Huang, Shang Jiang, Bobai Zhao, Shuai Wang, Siye Wang, Yanfang Zhang, TransTM: A device-free method based on time-streaming multiscale transformer for human activity recognition, Defence Technology, 2023, ISSN 2214-9147, https://doi.org/10.1016/j.dt.2023.02.021

[26] Abdul Wasay Sardar, Farman Ullah, Jamshid Bacha, Jebran Khan, Furqan Ali, Sungchang Lee, Mobile sensors based platform of Human Physical Activities Recognition for COVID-19 spread minimization, Computers in Biology and Medicine, Volume 146, 2022, 105662, ISSN 0010-4825, https://doi.org/10.1016/j.compbiomed.2022.105662

[27] Daniel Garcia-Gonzalez, Daniel Rivero, Enrique Fernandez-Blanco, Miguel R. Luaces, New machine learning approaches for real-life human activity recognition using smartphone sensor-based data, Knowledge-Based Systems, Volume 262, 2023, ISSN 0950-7051, https://doi.org/10.1016/j.knosys.2023.110260

[28] Hnoohom, N., Mekruksavanich, S., Jitpattanakul, A., 2023. An efficient resnetse architecture for smoking activity recognition from smartwatch. Intelligent Automation & Soft Computing 35, 1245–1259. https://doi.org/10.32604/iasc.2023.028290

[29] Nancy Gulati, Pankaj Deep Kaur, An argumentation enabled decision making approach for Fall Activity Recognition in Social IoT based Ambient Assisted Living systems, Future Generation Computer Systems, Volume 122, 2021, Pages 82-97, ISSN 0167-739X, https://doi.org/10.1016/j.future.2021.04.005

[30] Qi, W.; Su, H.; Yang, C.; Ferrigno, G.; De Momi, E.; Aliverti, A. A Fast and Robust Deep Convolutional Neural Networks for Complex Human Activity Recognition Using Smartphone. Sensors 2019, 19, 3731 https://doi.org/10.3390/s19173731

[31] V. Bianchi, M. Bassoli, G. Lombardo, P. Fornacciari, M. Mordonini and I. De Munari, "IoT Wearable Sensor and Deep Learning: An Integrated Approach for Personalized Human Activity Recognition in a Smart Home Environment," in IEEE Internet of Things Journal, vol. 6, no. 5, pp. 8553-8562, Oct. 2019, https://doi.org/10.1109/JIOT.2019.2920283

[32] S. Mekruksavanich and A. Jitpattanakul, "Sport-related activity recognition from wearable sensors using bidirectional gru network," Intelligent Automation & Soft Computing, vol. 34, no.3, pp. 1907–1925, 2022, https://doi.org/10.32604/iasc.2022.027233

[33] E. Dhiravidachelvi, M. Kumar, L. D. Vijay Anand, D. Pritima, S. Kadry et al., "Intelligent deep learning enabled human activity recognition for improved medical services," Computer Systems Science and Engineering, vol. 44, no.2, pp. 961–977, 2023, https://doi.org/10.32604/csse.2023.024612

[34] M. Abdel-Basset, H. Hawash, R. K. Chakrabortty, M. Ryan, M. Elhoseny and H. Song, "ST-DeepHAR: Deep Learning Model for Human Activity Recognition in IoHT Applications," in IEEE Internet of Things Journal, vol. 8, no. 6, pp. 4969-4979, 15 March15, 2021, https://doi.org/10.1109/JIOT.2020.3033430

[35] Z. Chen, C. Jiang and L. Xie, "A Novel Ensemble ELM for Human Activity Recognition Using Smartphone Sensors," in IEEE Transactions on Industrial Informatics, vol. 15, no. 5, pp. 2691-2699, May 2019, https://doi.org/10.1109/TII.2018.2869843

[36] Preeti Agarwal, Mansaf Alam, A Lightweight Deep Learning Model for Human Activity Recognition on Edge Devices, Procedia Computer Science, Volume 167, 2020, Pages 2364-2373, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2020.03.289

[37] Mansoor Nasir, Khan Muhammad, Amin Ullah, Jamil Ahmad, Sung Wook Baik, Muhammad Sajjad, Enabling automation and edge intelligence over resource constraint IoT devices for smart home, Neurocomputing, Volume 491, 2022, Pages 494-506, ISSN 0925-2312, https://doi.org/10.1016/j.neucom.2021.04.138

[38] Y. Tang, L. Zhang, H. Wu, J. He and A. Song, "Dual-Branch Interactive Networks on Multichannel Time Series for Human Activity Recognition," in IEEE Journal of Biomedical and Health Informatics, vol. 26, no. 10, pp. 5223-5234, Oct. 2022, https://doi.org/10.1109/JBHI.2022.3193148

[39] Kaixuan Chen, Dalin Zhang, Lina Yao, Bin Guo, Zhiwen Yu, and Yunhao Liu. 2021. Deep Learning for Sensor-based Human Activity Recognition: Overview, Challenges, and Opportunities. ACM Comput. Surv. 54, 4, Article 77 (May 2022), 40 pages. https://doi.org/10.1145/3447744

[40] Shaik, T., Tao, X., Higgins, N., Li, L., Gururajan, R., Zhou, X., & Acharya, U. R. (2023). Remote patient monitoring using artificial intelligence: Current state, applications, and challenges. WIREs Data Mining and Knowledge Discovery, 13(2), e1485, https://doi.org/10.1002/widm.1485

[41] Kei Tanigaki, Tze Chuin Teoh, Naoya Yoshimura, Takuya Maekawa, and Takahiro Hara. 2022. Predicting Performance Improvement of Human Activity Recognition Model by Additional Data Collection. Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 6, 3, Article 142 (September 2022), 33 pages. https://doi.org/10.1145/3550319