

# User-centric Activity Recognition and Prediction Model using Machine Learning Algorithms

Namrata Roy, Rafiul Ahmed, Mohammad Rezwatul Huq, Mohammad Munem Shahriar

Department of Computer Science and Engineering  
East West University, Dhaka, Bangladesh

**Abstract**—Human Activity Recognition has been a dynamic research area in recent years. Various methods of collecting data and analyzing them to detect activity have been well investigated. Some machine learning algorithms have shown excellent performance in activity recognition, based on which many applications and systems are being developed. Unlike this, the prediction of the next activity is an emerging field of study. This work proposes a conceptual model that uses machine learning algorithms to detect activity from sensor data and predict the next activity from the previously seen activity sequence. We created our activity recognition dataset and used six machine learning algorithms to evaluate the recognition task. We have proposed a method for the next activity prediction from the sequence of activities by converting a sequence prediction problem into a supervised learning problem using the windowing technique. Three classification algorithms were used to evaluate the next activity prediction task. Gradient Boosting performs best for activity recognition, yielding 87.8% accuracy for the next activity prediction over a 16-day timeframe. We have also measured the performance of an LSTM sequence prediction model for predicting the next activity, where the optimum accuracy is 70.90%.

**Keywords**—Machine learning algorithms; activity recognition; gradient boosting; next activity prediction; LSTM sequence prediction model

## I. INTRODUCTION

Rapid advancement in machine learning addresses a significant area of research, recognition, and human activity prediction. Predicting the next activity ahead of time can have a substantial impact on shaping and designing future technologies. A system needs to know the daily activities of a human to predict the next activity, which requires activity recognition. Recognition of an activity depends on capturing the movements and gestures made by different body parts. It is pretty challenging to correctly detect daily activities from a whole bunch of body movements. Again, the same activity can be performed in different ways. So, activity recognition has gained increasing interest in research in the past years. Activity prediction is the next step into the advancement of technology. From the recognized activities, a system would be able to predict what is going to be the next activity. It would be a massive leap towards reshaping the future, and the promise of such systems and technologies motivated this work to contribute to this field.

Activities are parts of Human behavior. An Activity consists of Actions. Three important terminologies are to be considered: Action, Activity, and Behavior [1]. Actions are a

more straightforward form of conscious body movements like moving hands up and down, which form a specific action such as eating, running, walking, etc. All the actions a human performs in daily life contribute to creating his behavior, a complex structure of activities with a hidden pattern. Action data is needed to be captured by using different sensors for recognition of Activity. It can also be done from a video feed. Thus, we have two approaches for activity recognition- vision-based and sensor-based [1]. The vision-based activity recognition approach often turns into privacy concerns, so sensors get more attention in research and thus our area of interest.

In recent years, the consumer electronics industry has made a considerable investment in wearable technology. Companies produce many different wearable devices: fitness trackers, smartwatches, connected headsets, smart glasses, wrist bands, etc. Despite wearable devices not being new, the development of mobile technologies and the quantified-self movement related to fitness and sports activities have led to their explosion [2], [3]. Among the wide variety of wearable devices, wrist-wearables such as smartwatches and wrist bands seem to have become mainstream. Estimations indicated that by 2019 [4], wrist wearable devices would reach 1 million sold units, while all the other wearable devices together will achieve just 7.3 million units. In addition to other features associated with their reduced size and comfortable use, wrist-wearable devices include many sensors providing continuous data about vital signs (e.g., heart rate, skin temperature, acceleration) and environmental variables (e.g., movements). Such advancements in wearable technology create opportunities to study further by analyzing the data and developing new applications, technologies, and solutions.

Different context-aware systems such as personalized assistants, home automation, health monitoring, security management systems, etc., can benefit from activity data. The anticipation of the following activity will empower such systems to interact and perform more efficiently with users to improve context-aware experiences.

Predicting the next activity of a user requires previous activity data. So, a system that recognizes activity and stores activity information (e.g., activity name, timestamp, etc.) can be used for activity prediction. For activity recognition, it requires data for training a machine learning model. Our work includes data collection using a wearable device to build a recognition model to recognize the activity. Then the predictor model predicts the next activity from the sequence of recognized activities. We will focus on the following facts:

- 1) Collecting sensor data for building a machine learning model.
- 2) Deploying a machine learning model that best recognizes activity from collected data.
- 3) Storing activity information in a log file for creating an activity sequence.
- 4) Build another machine learning model to learn patterns in the activity sequence for predicting the next activity.

We intend to record sensor data generated from body movements and use the data to train Machine Learning Models to recognize activities. Further, the detected activities will be used for predicting the next activity for a specific user. There are different sensors available to collect required data, and these sensors can be positioned in various body parts. Sensor position is an essential factor in HAR (Human Activity Recognition) [5]. They can also be attached to home entities (e.g., bedroom doors, kitchen doors, refrigerators, washing machines, etc.) in a smart home setting. Some sensors are integrated into smartphones and smartwatches/fitness trackers. We propose to record data from a wrist wearable device because wrists are engaged in most activities in daily life, and the position is ideal for collecting data for activity recognition purposes [5]. Choosing the suitable machine learning model for this task requires effort [6]. In our proposed work, the prediction of the next activity depends on the recognized activity sequence. Though activity recognition is a widespread research interest, activity prediction is still moderately new and challenging. Several approaches have been adopted to predict activities which include Hidden Markov Models (HMM) [7], Recurrent Neural Networks (RNN), Long-Short Term Memory (LSTM) [5], etc. In this work, we intend to collect data from sensors positioned at the wrist, recognize activity and predict the next possible activity of a specific user in the nearest future by implementing machine learning models. This paper contributes to the following sectors:

- 1) We derive three new features from collected data and build an Activity Recognition Model that performs moderately well based on a reduced dataset.
- 2) We propose a novel approach to predict activity by converting the sequence prediction problem into a supervised learning problem.
- 3) We also explore an LSTM sequence prediction approach for the next activity prediction.

The study schemes to propose a model to recognize human activity from the data collected by the sensors of a wrist-wearable device and then predict the next possible activity from the sequence of previous activities. Section 2 contains a brief discussion about Activity Recognition and Activity Prediction and prior works related to these fields. The Architecture and System Workflow of our proposed model is described in Section 3. The evaluation of our work and results are covered in Section 4. Section 5 includes an insightful discussion of our findings.

## II. LITERATURE REVIEW

Activity recognition is the process of recognizing an activity performed by a human. It is a machine learning

approach to detect activity by analyzing data given as input to a machine learning model built on a machine learning algorithm. It is a way to teach a machine to recognize an activity.

It is a fundamental and challenging problem to track and understand agents' behavior through videos taken by various cameras—the primary technique employed in computer vision. Vision-based activity recognition has found many applications such as human-computer interaction, user interface design, robot learning, and surveillance. In vision-based activity recognition, a great deal of work has been done. Researchers have attempted many methods such as optical flow, Kalman filtering, Hidden Markov models, etc., under different modalities such as single-camera, stereo, and infrared. In addition, researchers have considered multiple aspects of this topic, including single pedestrian tracking, group tracking, and detecting dropped objects. Recently some researchers have used RGBD cameras like Microsoft Kinect to detect human activities. Depth cameras add an extra dimension, i.e., the depth which a regular 2d camera fails to provide. Sensory information from these depth cameras has been used to generate real-time skeleton models of humans with different body positions. This skeleton information provides meaningful information that researchers have used to model human activities, which are trained and later used to recognize unknown activities.

Despite the remarkable progress of vision-based activity recognition, its usage for most actual visual surveillance applications remains a distant aspiration. Conversely, the human brain seems to have perfected the ability to recognize human actions. This capability relies not only on acquired knowledge but also on the aptitude of extracting information relevant to a given context and logical reasoning. Based on this observation, it is proposed to enhance vision-based activity recognition systems by integrating commonsense reasoning and contextual and commonsense knowledge.

Sensor-based activity recognition integrates the emerging area of sensor networks with novel data mining and machine learning techniques to model a wide range of human activities. Mobile devices (e.g., smartphones) provide sufficient sensor data and calculation power to enable physical activity recognition to estimate energy consumption during everyday life. Sensor-based activity recognition researchers believe that these computers will be better suited to act on our behalf by empowering ubiquitous computers and sensors to monitor agents' behavior (under consent). Sensor-based activity recognition is a challenging task due to the inherently noisy nature of the input. Thus, statistical modeling has been the main thrust in this direction in layers, where the recognition at several intermediate levels is conducted and connected. At the lowest level where the sensor data are collected, statistical learning concerns how to find the precise locations of agents from the received signal data. At an intermediate level, a statistical inference may be concerned about recognizing individuals' activities from the inferred location sequences and environmental conditions at the lower levels. Furthermore, at the highest level, a significant concern is to find out the overall goal or sub-goals of an agent from the activity

sequences through a mixture of logical and statistical reasoning.

Activity Recognition can be done in various ways. Some AR works include only accelerometer data. Fernando G.D Silva [8] designed a recognition system for simple human body movements using a tri-axial accelerometer sensor integrated with a sports watch. Min-Cheol Kwon and Sunwoong Choi built a system for recognizing activity using accelerometer and location data generated from a wrist-worn smartwatch using an Artificial Neural Network. Their approach is location-specific; a user can only perform certain actions in a specific location. An activity recognition model based on a wavelet using one or more accelerometers was proposed by Mannini and Sabitini [9]. Casale et al. [10] used a wearable device for collecting accelerometer data for human activity recognition. Some works include multiple accelerometers. Chung, Lim, Noh, Kim, and Jeong [11] built a testbed to collect motion data using a tri-axial accelerometer, gyroscope, and magnetometer by attaching eight Inertial Measurement Units (IMU) devices on different parts of the human body. They trained that dataset using the Long Short-Term Memory (LSTM) network to recognize a few Activities of Daily Living (ADLs). Foerster and Fahrenberg [12] collected data using five accelerometers and proposed a hierarchical model to classify different body postures and movements. Beddiar, Nini, Sabokrou and Hadid [13] surveyed numerous vision-based human activity recognition research papers to describe the method of HAR. They featured three essential components of this approach, which are video frame segmentation for activity recognition, action representation of the body postures and motions and ML algorithms to recognize activities by learning process. Bao and Intille [14] used five biaxial accelerometers worn on other body parts to monitor 20 types of activities using C4.5 and Naive Bayes classifiers. Wallace [15] Ugulino proposed another ML-based recognition classifier to detect five different activities (sitting, standing, sitting down, standing up, and walking) using body-worn accelerometer data collected from 4 participants. Krishnan et al. [16] collected data from ten participants using three accelerometers to detect lower body motions. Samad Zabihi [17] used transformation of the accelerometer data ( $x$ ,  $y$ ,  $z$ ) to a spherical coordinate system ( $r$ ,  $\phi$ ,  $\theta$ ) for activity recognition and extracted features from transformed data. Zhen-Yu [18] used tri-axial accelerometer data to build an autoregressive model to detect human activity. Different activities (run, still, jump, and walk) were classified using AR coefficients feature extraction. Huawei Wang [19] used Principal component analysis to reduce the dimensionality and selected 3 out of the 12 features of a dataset. Magnetic-induction based communication system is used for sensing and transmitting data generated by every movement of the body part to recognize a physical activity in [20] Acceleration data of a smartphone is investigated in [21]. Twenty-nine users participated in data collection, and each of them carried an Android phone in their pocket. They were asked to perform six activities: Walking, Standing, Sitting, Jogging, Stairs-Up and Stairs-Down. They used Logistic Regression, J48, and Multilayer Perceptron for the model evaluation. The accuracy was above 90%. They found it a little hard to differentiate between Stairs-Up versus Stairs-Down. For activity prediction

purposes, the most popular approaches are to use Recurrent Neural Networks (LSTM) and Hidden Markov Model (HMM) [1]. This paper [22] shows the importance of prediction in intelligent environments. Most of the Prediction tasks are carried out as a sequence prediction.

Activity Prediction is the process of predicting an activity ahead of time that will be performed in the nearest future. It is a way to teach a machine to predict an activity by using machine learning models. Most of the prediction tasks carried out in the past are either in a Smart Home scenario to identify the next sensor that would generate the next event or from a video to infer what will happen next. Personalized activity prediction is still a new concept and is merely investigated as a research topic. An activity can be predicted for a specific user by learning from the pattern of activities performed previously by that user. So, the activity prediction problem can be formulated as a sequence prediction problem. Sequence prediction is a problem that includes using historical sequence information to predict the next value or values in the sequence. Various methods are available for sequence prediction, but Recurrent Neural Networks, especially LSTMs, have been the best in use.

Predicting future activity empowers different applications like personal assistants and context-aware systems to interact more efficiently with the user. The problem of next activity prediction is often addressed as sequence prediction, which can be adapted to predict the label of the activity that will occur next in the sequence. Du, Lim, and Tan [23] performed activity recognition on some ADLs to generate a series of activities and then implemented LSTM and Naive Bayes to find the accuracy of predicting the next activity. The active LeZi algorithm is implemented in this [24] work to identify the sensor in a home that would generate the next event. There have been works on predicting the next location [25] and user's location-based mobility [26]. Location-based human behavior is focused in these papers and is subject primarily to using the Markov models. Markov models lack the flexibility to explore past activities instead of making a prediction based on only the most recent previous state. This feature restricts Markovian models from getting high-level insight into the data. Another popular method is Sequence mining that can be used to address such problems [27]. A dataset consisting the name of activities was generated from a collection of human actions using mapping and word embedding using LSTM algorithms to predict the future activity was implemented in [28]. Some activity prediction works are also found to be vision-based [7], [29], [2]. Alfaifi and Artoli [30] evaluated recent improvements in activity prediction and proposed a 3D-convolutional neural network model that extracted features and classified them to predict the action by LSTM.

### III. ARCHITECTURE OF PROPOSED SOLUTION

In this paper, we intend to propose a conceptual model for activity recognition and next activity prediction. The model complements the existing wearable technology architecture [31]. As illustrated in Fig. 1, the current architecture has three main components: a wrist-worn device (i.e., fitness tracker/smartwatch), an intermediate medium (i.e., Smartphone, computer), and a server/cloud service. It follows

a Proprietary system. Wearable vendors use this system for data collection and analytics and send the analytical result to the user and the authorized third parties [31]. A wrist-worn device is used to collect data using the sensors integrated into it. Theoretically, the device should be capable of sending data directly to the server for permanent storage, but it is not in use [31].

It sends data to the server/cloud service via the intermediate platforms. The server performs analytical operations to generate insight from data and sends it back to the middle layer as an interface for the wearable device. Our work will enhance the analytical capability by adding the feature to recognize an activity with a machine learning model. The model will take a chunk of sensor data from the device as input and generate a label for that movement (i.e., activity). After successfully recognizing activities and storing them as an activity sequence for a certain period, another machine learning model will take the sequence of activities as input and predict the next possible activity.

The first task of our proposed work requires data for building an activity recognition model. A fitness tracker or smartwatch is the best solution for collecting such data, integrated with different sensors. Still, we have used a smartphone for data collection and storage simplicity, tying it on the participants' hands like a tracker/smartwatch. This approach has collected the desired data but with more straightforward storage options as we could save it directly as CSV files. We have used an android app named 'AndroSensor'<sup>1</sup> available on the Google Play Store to collect data. We have recorded accelerometer, gyroscope, and sound level data and also kept timestamps. The data recording rate has been set to: 0.125 s/per data (instance).

Table I shows the value of the day-of-the-week and corresponding value. One is a weekday, and another is the weekend. We have considered Sunday, Monday, Tuesday, Wednesday, Thursday as weekdays while Friday and Saturday as weekends according to local holidays. Table II illustrates time-of-the-day data. Furthermore, the datasets have used in this study can be categorized into two types: recognition dataset and prediction dataset. We have selected ten activities (Table III). Here, we have held a common assumption: A user will not perform multiple activities simultaneously. The recognition dataset consists of 330993 rows and ten columns. It has contained acceleration values of x, y, z-axis, orientation values of x, y, z-axis, sound level (in DB), day-of-the-week (Weekday/Weekend), time-of-the-day (Morning, Noon, Evening, Night). The last two features have been generated from a timestamp—the acceleration data of x, y, and z-axis have depicted hand movement through the corresponding axis. We have also considered the orientation data of the x, y, z-axis because sometimes there were activities that did not generate insightful acceleration data. Some activities have been performed in specific places, e.g., we have taken transport in noisy places rather than a closed room. So, the sound level has played an essential role in recognizing activities accurately.

In the data pre-processing task, we have converted timestamp data into day-of-the-week (Table I) and time-of-the-day (Table II). Sound level, day-of-the-week, and time-of-the-day are three novel features we have introduced in our research. A human annotator has labeled the collected data according to activity labels (Table III). There have been three participants; two of them were male and one female. Each of them has collected data for three consecutive days consisting of two weekdays and one weekend. This dataset has been used to detect activity.

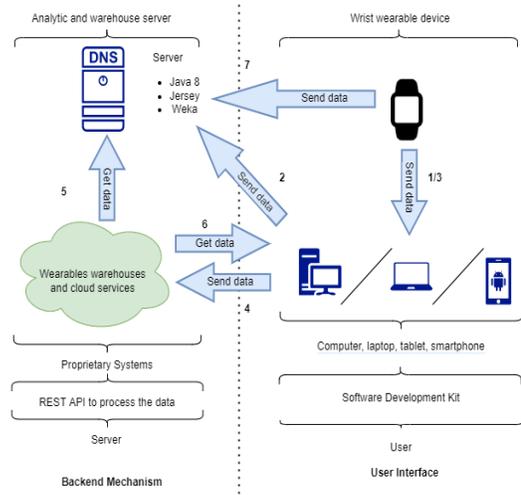


Fig. 1. Wearable Technology Architecture.

TABLE I. VALUE OF DAY-OF-THE-WEEK

| Day-of-the-week                                        | Value |
|--------------------------------------------------------|-------|
| Weekday (Sunday, Monday, Tuesday, Wednesday, Thursday) | 1     |
| Weekend (Friday, Saturday)                             | 2     |

TABLE II. VALUE OF TIME-OF-THE-DAY

| Time-of-the-day                | Value |
|--------------------------------|-------|
| Morning (5:00 AM- 11:59 AM)    | 1     |
| Noon (12:00 PM- 16:59 PM)      | 2     |
| Afternoon (17:00 PM- 19:59 PM) | 3     |
| Night (20:00 PM- 4:59 AM)      | 4     |

TABLE III. ACTIVITY LABELS

| Activity               | Label |
|------------------------|-------|
| Brushing teeth         | 1     |
| Drinking tea or coffee | 2     |
| Eating                 | 3     |
| Walking                | 4     |
| Taking Transport       | 5     |
| Working on PC          | 6     |
| Using Mobile Phone     | 7     |
| Reading                | 8     |
| Cooking                | 9     |
| Cleaning Utensils      | 10    |

<sup>1</sup><https://play.google.com/store/apps/details?id=com.fivasim.androsensor&hl=en&gl=US>

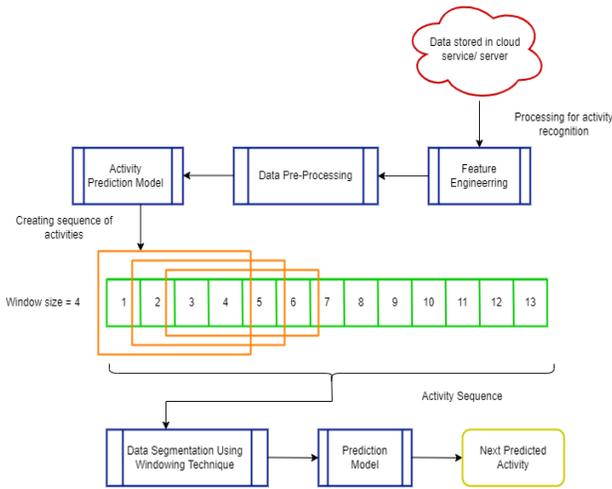


Fig. 2. System Workflow.

Fig. 2 depicts the workflow of our approach in this study. Our proposed model is initiated after data is collected by a wearable device and sent to the server. Feature engineering and data pre-processing are performed on the stored data for efficient modeling.

In our conceptual system architecture, data is then sent to the recognition model. In this study, we have investigated six machine learning algorithms: Artificial Neural Network (ANN), K-Nearest Neighbor (KNN), Naïve Bayes (NB), Random Forest (RF), Support Vector Machine (SVM), and Gradient Boosting (GB) to recognize activities. When the model has started identifying activities, the labels were sent to a pipeline to create a sequence of activities. We have implemented a windowing technique to extract data from the pipeline to prepare the dataset for the next activity prediction.

Activity prediction is a user-centric process. When the activity recognition algorithm is implemented in the wearable technology architecture, it starts detecting the user's activities. So, we propose that the sequence of activities will be stored in a log file and used as the dataset for prediction purposes. We have collected data for some consecutive days to make a sequence of activities for demonstrating the activity prediction problem.

The prediction dataset used in this paper has been prepared by recording activities for eight consecutive days and sixteen days to evaluate the model performance regarding time. As prediction of the next activity depends on a specific user's behavior, data has been collected from one participant. The prediction model has taken previous activities and the current activity into consideration for predicting the next activity. Table IV shows a glimpse of the sequence of activities performed by the participant. Hence, it has a various number of attributes depending on the window size we choose. Depending on different window sizes concerning 8 days and 16 days, the number of rows has differed in the prediction dataset.

TABLE IV. SEQUENCE ACTIVITIES

|   |   |   |   |    |   |   |   |   |       |
|---|---|---|---|----|---|---|---|---|-------|
| 1 | 6 | 4 | 5 | 10 | 2 | 6 | 3 | 1 | ..... |
|---|---|---|---|----|---|---|---|---|-------|

TABLE V. TRANSFORMED DATA FOR NEXT ACTIVITY PREDICTION (WINDOW SIZE = 3)

|   | Prev_act2 (Feature 1) | Prev_act1 (Feature 2) | Curr_act (Feature 3) | Next_act (Target) |
|---|-----------------------|-----------------------|----------------------|-------------------|
| 1 | 1                     | 6                     | 4                    | 5                 |
| 2 | 6                     | 4                     | 5                    | 10                |
| 3 | 4                     | 5                     | 10                   | 2                 |
| 4 | 5                     | 10                    | 2                    | 6                 |
| 5 | 10                    | 2                     | 6                    | 9                 |

We have introduced a novel approach for the next activity prediction. We have transformed the sequence of activities into a feature and target-shaped data frame by implementing the windowing technique. It creates the opportunity to use the transformed sequential dataset in a supervised learning problem predicting human activities. Table V shows a sample of the dataset for the next activity prediction, where the window size is 3, which features were used as input and the target as output for the prediction model.

We propose another approach to predict the next activity, as shown in Fig. 3. An LSTM sequence prediction approach is adopted to predict the next value in the activity sequence.

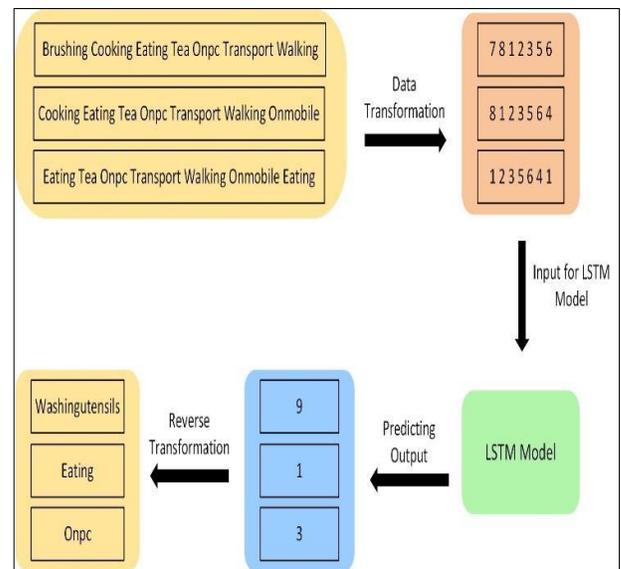


Fig. 3. LSTM Sequence Prediction Workflow.

#### IV. RESULT AND DISCUSSION

We have used python with scikit-learn ML packages for the Machine Learning algorithms implemented in our study [32]. Keras and Tensorflow are used to implement LSTM. Some modifications have been made in the default hyperparameters of the ML algorithms to achieve better accuracy. The codes run in the Anaconda Jupyter Notebook (Version3).

Fig. 4 illustrates the accuracy comparison of the six algorithms for activity recognition. For the Artificial Neural Network, the number of hidden layers has played an important role. The accuracy for the 3-layer network was 98.99% and for the 7-layer network was 99.01%. However, we have

considered the outcome of five hidden layer networks, as it performed the best for this dataset with an accuracy of 99.1%. On the other hand, the K-Nearest Neighbor yielded an accuracy of 99.2%. Where the accuracy of the Random Forest was 99.3%, for the Support Vector Machine, it was 98.7%. Though the Naïve Bayes performed poorly, having an accuracy of 61.9%, among all the six ML algorithms, the Gradient Boosting showed the highest accuracy of 99.7%.

As mentioned earlier, in this study, we have introduced three novel features: sound level, day-of-the-week, and time-of-the-day. These trio features help to detect the activity more accurately. As a comparison, we have run all six algorithms on the same dataset without these trio features.

Fig. 5 clearly describes that all six algorithms' accuracy had reduced compared to Fig. 4. Yet, Gradient Boosting had the best accuracy of 98.8%, and Naïve Bayes yielded the lowest accuracy of 54.4%. In comparison, the accuracy differed by around 2% for all the algorithms except for Naïve Bayes. Naïve Bayes accuracy difference was almost 7%.

After studying the accuracy score of the six implemented ML algorithms in the previous section for activity recognition, we can see that GB gave the best result [99.7%], where the nearest result was shown by KNN and the bagging algorithm Random Forest with an accuracy rate of 99.2% and 99.3% respectively. As we know, both bagging and boosting are ensemble methods and perform better compared to other ML algorithms. So, as expected, both of them have the highest accuracy among all six algorithms.

Fig. 6 explains the confusion matrix of our best model for detecting human activity, Gradient Boosting. Almost all the instances of the actual class have been predicted accurately by the model. Specifically, the activity 'Walking' is labeled as activity no. 4. Of all the instances of 'Walking,' data has been predicted correctly as 'Walking' with an accuracy of 99.9%. Also, the instances of other activities have been predicted correctly with varying accuracy from 98.8% to 99.8%.

For predicting the next activity, we have used two timeframes. The first one was a prediction on an 8 days activity sequence and the second one was on 16 days. We have used 3 ML algorithms to check the accuracy, e.g., ANN, KNN, GB. We have experimented with different window sizes to observe the effect on accuracy.

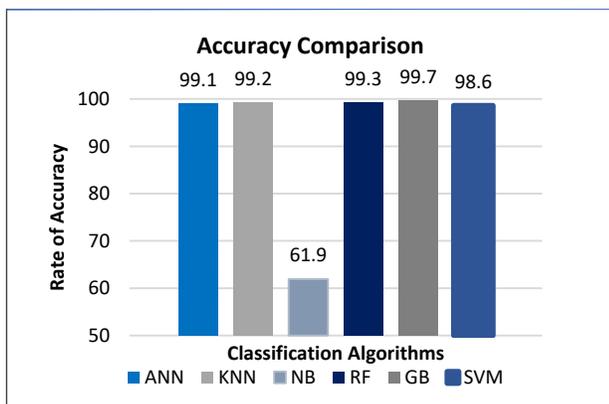


Fig. 4. Accuracy Comparison for Activity Recognition.

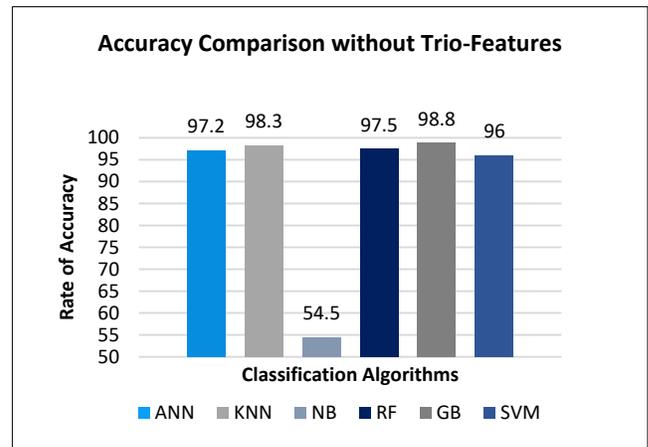


Fig. 5. Accuracy Comparison without Trio-Features.

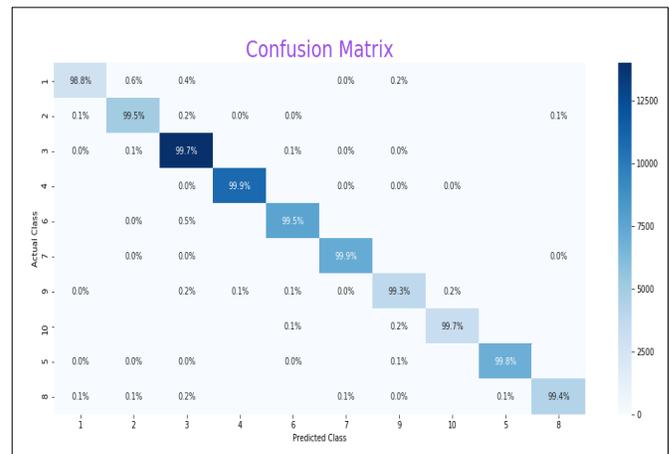


Fig. 6. Confusion Matrix of the Best Model (Gradient Boosting).

Here we have kept both training and test accuracy in determining if the models have overfitting or underfitting issues. For an 8-days timeframe, window size (W) was taken from 2 to 6 with an interval of 1 to explore the effect of window size on accuracy comparison. Fig. 7, Fig. 8, Fig. 9 show the accuracy comparison of training and test for 8 days on ANN, KNN, GB for predicting the next activity, respectively.

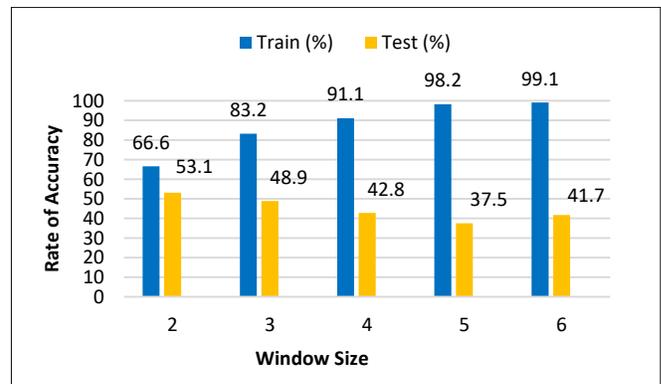


Fig. 7. Accuracy Comparison (8 Days) with varied Window Size for ANN.

Fig. 7 illustrates that an increase in window size increases training accuracy but decreases test accuracy. At  $W = 2$ , ANN yielded the highest test accuracy of 53.1% but kept lowering with the window size increase.

Similarly, in Fig. 8, training accuracy increases with the increase of window size, but there is a sudden drop at  $W = 6$ . However, the highest test accuracy for KNN was 46.9% at  $W = 2, 3$ . The test accuracy gradually kept decreasing with the change of window size. KNN showed similar behavior as ANN to predict the next activity from the activity sequence of 8 days.

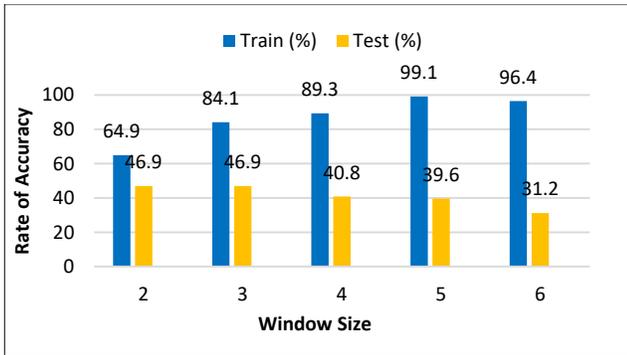


Fig. 8. Accuracy Comparison (8 Days) with varied Window Size for KNN.

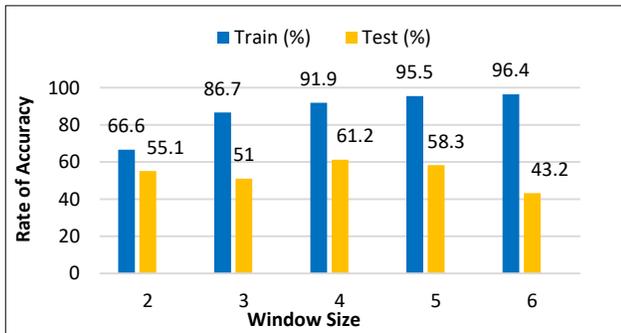


Fig. 9. Accuracy Comparison (8 Days) with varied Window Size for GB.

Fig. 9 displays that with the change of window size, training accuracy kept increasing. In the case of testing accuracy, there was some fluctuation of performance for varied window sizes. Notably, for  $W = 4$ , GB showed the best result of 61.2% accuracy.

We can sense a trend in training and test accuracy with varied window sizes. In most cases, for all the algorithms, an increase in window size increases training accuracy but results in a gradual decrease in test accuracy. So, there is a hint of slight overfitting of data here. Now we will be analyzing the same for a more extended timeframe of 16 days of activity sequence.

For a 16-days timeframe, the window size varied from 2 to 8. For predicting the next activity, the accuracy comparison of training and test for 16 days on ANN, KNN, and GB is illustrated in Fig. 10, Fig. 11, and Fig. 12, accordingly.

Analyzing Fig. 10, it can be said that, while training accuracy increases with the increase of window size, testing

accuracy varies significantly. The highest test accuracy was yielded at  $W = 6$ , which was 81.6%.

Similarly, in Fig. 11, for KNN, training accuracy increased with the change of window size. KNN also varied for test accuracy, but like the previous one, it performed best at  $W = 6$ , 82.7%.

The accuracy comparison for GB on 16 days dataset is graphed in Fig. 12. With the increase of window size, training accuracy kept increasing. GB also yielded the highest accuracy of 87.8% at  $W = 6$ .

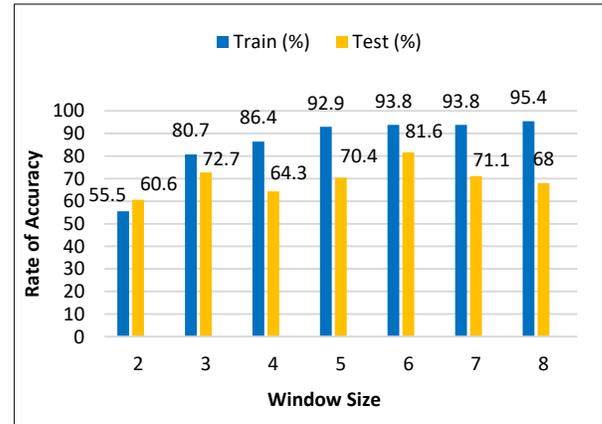


Fig. 10. Accuracy Comparison (16 Days) with varied Window Size for ANN.

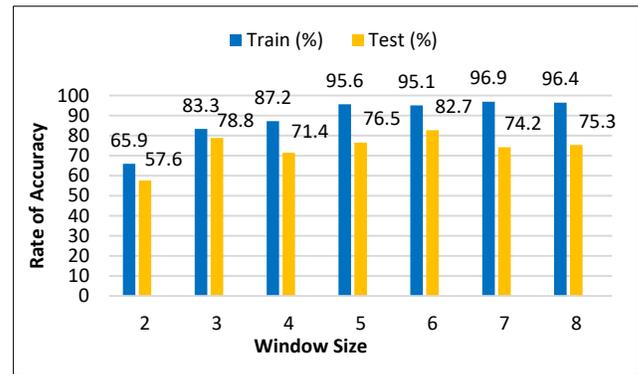


Fig. 11. Accuracy Comparison (16 Days) with varied Window Size for KNN.

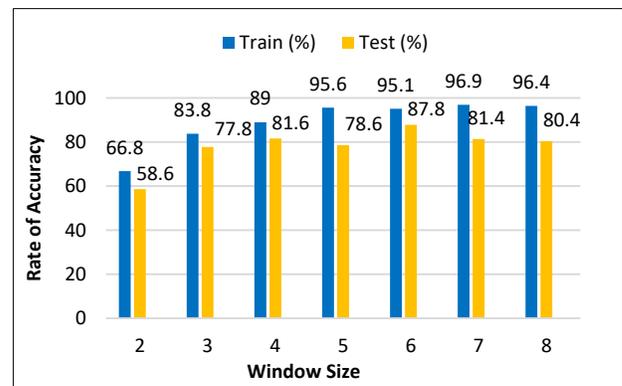


Fig. 12. Accuracy Comparison (16 Days) with varied Window Size for GB.

After vivid exploration of the above cases, a longer timeframe of activity sequence yields better performance for

predicting the next activity. Also, we can say that the algorithms are prone to overfitting with smaller timeframe datasets.

Observing Table VI, it can be said that increasing the window size gives a good test accuracy compared to the training accuracy for a specific limit. The training accuracy increases with the increase of window size. Just a small exception in the case of KNN when the window size was changed from 5 to 6, it decreased instead of increasing. But in the case of testing, while changing the window size from 4 to 5, the test accuracy started to decline and continued for all the algorithms. On the other hand, Table VII indicates the accuracy comparison between the training and testing dataset for 16 days. Unlike Table VI, after increasing the window size with an interval of 1, GB's testing accuracy was quite good, though the accuracy decreased for ANN and KNN on some points.

For training accuracy, we can say that the accuracy increased gradually with the change of the window size. Though after size 5, it decreased for some, at size 7, it rose again. It can be determined that for ANN and KNN the test accuracy varied while increasing the window size after analyzing Table VII. But for GB, the accuracy increased up to window size 6, and after that, it started to decrease even though the window size was increasing. This was the purpose of considering a higher range of window size for 16 days compared to 8 days.

TABLE VI. TRAINING AND TESTING ACCURACY (8 DAYS) WITH VARIED WINDOW SIZE

| Window Size | ANN       |          | KNN       |          | GB        |          |
|-------------|-----------|----------|-----------|----------|-----------|----------|
|             | Train (%) | Test (%) | Train (%) | Test (%) | Train (%) | Test (%) |
| 2           | 66.6      | 53.1     | 64.9      | 46.9     | 66.6      | 55.1     |
| 3           | 83.2      | 48.9     | 84.1      | 46.9     | 86.7      | 51.0     |
| 4           | 91.1      | 42.8     | 89.3      | 40.8     | 91.9      | 61.2     |
| 5           | 98.2      | 37.5     | 99.1      | 39.6     | 95.5      | 58.3     |
| 6           | 99.1      | 41.7     | 96.4      | 31.2     | 96.4      | 43.2     |

TABLE VII. TRAINING AND TESTING ACCURACY (16 DAYS) WITH VARIED WINDOW SIZE

| Window Size | ANN       |          | KNN       |          | GB        |          |
|-------------|-----------|----------|-----------|----------|-----------|----------|
|             | Train (%) | Test (%) | Train (%) | Test (%) | Train (%) | Test (%) |
| 2           | 55.5      | 60.6     | 65.9      | 57.6     | 66.8      | 58.6     |
| 3           | 80.7      | 72.7     | 83.3      | 78.8     | 83.8      | 77.8     |
| 4           | 86.4      | 64.3     | 87.2      | 71.4     | 89.0      | 81.6     |
| 5           | 92.9      | 70.4     | 95.6      | 76.5     | 95.6      | 78.6     |
| 6           | 93.8      | 81.6     | 95.1      | 82.7     | 95.1      | 87.8     |
| 7           | 93.8      | 71.1     | 96.9      | 74.2     | 96.9      | 81.4     |
| 8           | 95.4      | 68.0     | 96.4      | 75.3     | 96.4      | 80.4     |

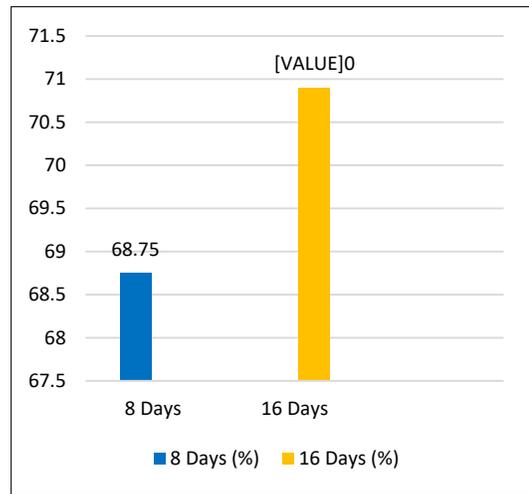


Fig. 13. Accuracy Comparison (8 Days vs. 16 Days) LSTM Activity Prediction.

Fig. 13 shows an accuracy comparison for the LSTM model used for predicting the next activity. Here we have assessed the performance of the model in a limited way. The model is assessed under timeframes of 8 and 16 days but considered a fixed window size for each timeframe, unlike the previous method that used varied window sizes. For the 8 days' timeframe, window size 6 was taken, and it was 8 for 16 days. It showed slightly better performance for a longer sequence of activities of 16 days than 8 days. For the 8 days' timeframe, the model yielded an accuracy of 68.75% and 70.90% for 16 days. We observe a similar behavior of the LSTM model performing with better accuracy for a longer timeframe, just like the previous method.

Both the supervised model and LSTM had performed well for predicting the next activity for an individual when there was a long activity sequence. Fig. 14 shows a performance comparison between LSTM and the supervised model used for predicting next activity. For 8 days timeframe and window size 6, LSTM outperforms our best-supervised model. But in the longer timeframe of 16 days and window size 8, the supervised model performs better than LSTM.

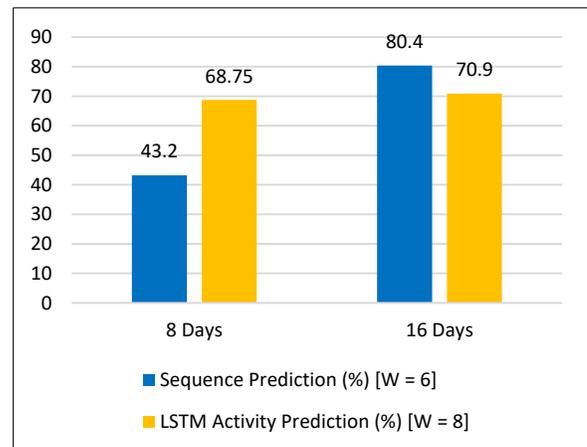


Fig. 14. Accuracy Comparison between Supervised Model and LSTM Activity Prediction (8 Days vs. 16 Days).

Hence, we can conclude that both the methods used for predicting the next activity perform better with an extended timeframe of activities.

Compared to the current work, our proposed model includes three new novel features: sound level, day-of-the-week, and time-of-the-day while collecting data. Moreover, the activities chosen to be recognized shows deviation quantitatively. Various machine learning algorithms have been applied, and their accuracy is as good as the current work. Furthermore, we have converted a sequential dataset into a supervised learning model, which was not conducted in previous work. The accuracy rate of the LSTM model for activity prediction performed better than the existing surveys.

To summarize, our proposed models show a significant outcome. The Gradient Boosting produced 99.7% accuracy out of all six algorithms for recognizing activity. In addition, a longer timeframe with a longer day count performs better than others for predicting next activity. Sequence and LSTM activity predictions provide 80.4% and 70.9% accuracy respectively for a 16-day timeframe with window size 8.

## V. CONCLUSION

Due to resource constraints, we could not use a smartwatch or any wearable device to collect data; instead, we used a smartphone tying it on the wrist, which worked just like a wrist wearable. For data collection, we had to depend on only 3 participants. The work can be best understood and explained by deploying it in real-time. But we had to work with batch processing due to the lack of high-end machines and components.

This paper proposes a conceptual model for activity recognition and prediction, which can be extended with the existing wearable architecture. We have shown how sound level, day-of-the-week, and time-of-the-day can improve activity recognition model accuracy. We adopted a straightforward but effective approach to convert a sequence prediction problem into a supervised learning problem to predict the next activity. The accuracy was moderate, considering the dataset we have. We have also used a Long-Short-Term Memory (LSTM) model to explore the prediction process. The model we developed has shown good performance, yet the result is not always accurate.

Our models have shown promise for both activity recognition and prediction. Wearable technologies and home automation, security systems, and health monitoring systems can significantly leverage this concept. By integrating our model in a smartwatch or a wrist-worn wearable architecture, this unique technology can be readily available to common people. Predicting the next human activity can create a significant impact on existing and future technologies.

The next activity predictor model predicts the next activity based on previously detected activities. Prediction model accuracy depends mainly on the timeframe. The comparison between the 8 days dataset and 16 days dataset clearly shows the difference. If the dataset is sufficiently large and the hyperparameters are appropriately tuned, we expect a far better result. We have noticed that both prediction models yield better accuracy with a larger window size when the time

frame increases from 8 days to 16 days. Though it is difficult to say which model can predict the next activity of a person more accurately, for different timeframes with different window sizes, both the supervised learning model and LSTM perform differently. We have a plan to extend our research with more participants.

Moreover, we want to determine if it is possible to predict which approach performs more precisely using even more timeframes with different window sizes. We will also investigate the fact that if there is any correlation between timeframe and window size. We intend to further mature this proposed model by implementing it in real-time with streaming data. In the future, we will try to make predictive modeling more efficient by incorporating timestamps in the model and the activity to create time series forecasting. We also intend to introduce some more valuable features and recognize more new activities. Furthermore, a recommendation system can be designed based on the predicted activity. The recommender system will use the next activity data indicated by the model to make corresponding recommendations.

## REFERENCES

- [1] A. Almeida and G. Azkune, "Predicting Human Behaviour with Recurrent Neural Networks," in DeustoTech-Deusto Foundation, University of Deusto, Av. Universidades 24, 487 Bilbao, Spain, 2018.
- [2] Y. Fu and Y. Kong, "Human Activity Recognition and Prediction," JOURNAL OF LATEX CLASS FILES, VOL. 13, NO. 9, SEPTEMBER 2018, pp.23-48.
- [3] R. Rawassizadeh, B. A. Price, and M. Petre, "Wearable devices: Has the Age of Smart watches Finally Arrived?," Communications of the ACM, 2015, pp. 45-47.
- [4] M. Swan, "Sensor Mania! The Internet of Things, Wearable Computing, Objective Metrics, and the Quantified Self," 2.0. Journal of Sensor and Actuator Networks, 2012, pp. 217-253.
- [5] S. Muhammad, S. Bosch, O. D. Incel, H. Scholten and P. Havinga, "Complex Human Activity Recognition Using Smartphone and Wrist-Worn Motion Sensors," Sensors, 2016.
- [6] A. Moraru, M. Pesko, M. Porcius, C. Fortuna and D. Mladenic, "Using machine learning on sensor data," Proceedings of the ITI 2010, 32nd International Conference on Information Technology Interfaces, Cavtat, 2010, pp. 573-578.
- [7] M. S. Ryoo, "Human activity prediction: Early recognition of ongoing activities from streaming videos," 2011 International Conference on Computer Vision, Barcelona, 2011, pp. 1036-1043.
- [8] F. G. da Silva and E. Galeazzo, "Accelerometer based intelligent system for human movement recognition," 5th IEEE International Workshop on Advances in Sensors and Interfaces IWASI, Bari, 2013, pp. 20-24.
- [9] A. Mannini and A. Sabatini, "Machine Learning Methods for Classifying Human Physical Activity from On-Body Accelerometers," Sensors, Basel, Switzerland, 2010, pp. 1154-1175.
- [10] P. Casale, O. Pujol, and P. Radeva, "Human Activity Recognition from Accelerometer Data Using a Wearable Device," Pattern Recognition and Image Analysis: 5th Iberian Conference, IbPRIA 2011, Las Palmas de Gran Canaria, Spain, June 8-10, 2011, pp. 289-296.
- [11] S. Chung, J. Lim, K. J. Noh, G. Kim, and H. Jeong, "Sensor Data Acquisition and Multimodal Sensor Fusion for Human Activity Recognition Using Deep Learning," Sensors, vol. 19, no. 7. MDPI AG, p. 1716, Apr. 10, 2019.
- [12] F. Foerster and J. Fahrenberg, "Motion pattern and posture: Correctly assessed by calibrated accelerometers," Behavior research methods, instruments, & computers: a journal of the Psychonomic Society, Inc. 32. 450-7, 20.

- [13] D. R. Beddiar, B. Nini, M. Sabokrou, and A. Hadid, "Vision-based human activity recognition: a survey," *Multimed. Tools Appl.*, vol. 79, no. 41–42, pp. 30509–30555, 2020.
- [14] L. Bao and S. S. Intille, "Activity Recognition from User-Annotated Acceleration Data," In A. Ferscha & F. Mattern (eds.), *Pervasive*, 2014, pp. 1-17.
- [15] W. Ugulino, D. Cardador, K. Vega and E. Velloso, "Wearable computing: accelerometers data classification of body postures and movements." *Advances in Artificial Intelligence-SBIA 2012*. Springer Berlin Heidelberg, 2012, pp. 52-6.
- [16] N. C. Krishnan, D. Colbry, C. Juillard and S. Panchanathan, "Real time human activity recognition using tri-axial accelerometers, Sensors, signals and information processing workshop," *Sensors Signals and Information Processing Workshop*, Sedona, AZ, 28,2018, pp.915-922.
- [17] W. T. D. Souza, and R. Kavitha, "Human Activity Recognition Using Accelerometer and Gyroscope Sensors," *International Journal of Engineering and Technology*, 2017, pp.1171-1179.
- [18] Zhen-Yu He and Lian-Wen Jin, "Activity recognition from acceleration data using AR model representation and SVM," *28 International Conference on Machine Learning and Cybernetics*, Kunming, 28,2008 pp. 2245-2250.
- [19] H. Wang and J. Wu, "Classification of Human Posture and Movement Using Accelerometer Data," in *International Conference on Innovations in Computing & Networking (ICICN16)*, CSE, RRCE, ISSN: 0975-0282.
- [20] N. Golestani and M. Moghaddam, "Human activity recognition using magnetic induction-based motion signals and deep recurrent neural networks," *Nat. Commun.*, vol. 11, no. 1, p. 1551, 2020.
- [21] J. R. Kwapisz, G. M. Weiss and S. A. Moore, "Cell phone-based biometric identification," *2010 Fourth IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, Washington, DC, 2010, pp. 1-7.
- [22] D. J. Cook and S. K. Das, "How smart are our environments? An updated look at the state of the art" *t. Pervasive Mobile Comput.* 27, 3, pp.53–73.
- [23] Y. Du, Y. Lim, and Y. Tan, "A Novel Human Activity Recognition and Prediction in Smart Home Based on Interaction," *Sensors*, vol. 19, no. 20. MDPI AG, p. 4474, Oct. 15, 2019.
- [24] K. Gopalratnam and D. J. Cook, "Online Sequential Prediction via Incremental Parsing: The Active LeZi Algorithm," *IEEE Intelligent Systems*, vol. 22, no. 1, Jan.-Feb. 27, pp. 52-58.
- [25] S. Sigg, S. Haseloff and K. David, "An Alignment Approach for Context Prediction Tasks in UbiComp Environments," *IEEE Pervasive Computing*, vol. 9, no. 4, October-December 2010, pp. 90-97.
- [26] J. Mcinerney, S. Stein, A. Rogers, Alex and N. R. Jennings, "Breaking the habit: Measuring and predicting departures from routine in individual human mobility," *Pervasive and Mobile Computing*, 2013, pp. 808–822.
- [27] O. Brdiczka, N. S. Makoto and J. Begole, "Temporal task footprinting: Identifying routine tasks by their temporal patterns," *International Conference on Intelligent User Interfaces*, Proceedings IUI, 2010, pp. 281-284.
- [28] J. Septiadi, B. Warsito, and A. Wibowo, "Human activity prediction using long Short Term Memory," *E3S Web Conf.*, vol. 202, p. 15008, 2020.
- [29] L. Draschkowitz, C. Draschkowitz and H. Hlavacs, "Using Video Analysis and Machine Learning for Predicting Shot Success in Table Tennis," *EAI Endorsed Transactions on Creative Technologies*, 2015.
- [30] R. Alfaifi and A. M. Artoli, "Human action prediction with 3D-CNN," *SN Computer Science*, vol. 1, no. 5, 2020.
- [31] A. P. Francisco, M. C. Rodriguez, J. M. Santos, "Collection and Processing of Data from Wrist Wearable Devices in Heterogeneous and Multiple-User Scenarios," *Sensors*, 2016, pp. 1538.
- [32] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos and D. Cournapeau, "Scikit-learn: Machine Learning in Python," *J. Mach. Learn. Res.* 2011, 12, 2825–2830.