International Journal of Computational Science and Engineering.
ISSN 2249-4251 Volume 13, Number 1 (2023), pp. 11-34

https://dx.doi.org/10.37622/IJCSE/13.1.2023.11-34

Smart Fit On Using Machine Learning and Deep Learning Techniques

***Dr. N. Kalyani, **Begari Shailaja, **Daripali Aparna, **Gayatri Shambhu Shastri and **Taluka Tejaswi**

**Professor, **UG Students*
G. Narayanamma Institute of Technology & Science (For Women)
**email: nara.kalyani@gnits.ac.in*

Abstract

Our society needs to know what people are doing for different reasons such as keeping people safe and tracking lifestyle and behavior. It's hard to watch everyone all the time, but sensors that respond to movement can help recognize what someone is doing. Many people have smartphones with sensors, making this kind of recognition more useful. To classify human activities this research looks at using different techniques using data from smartphone sensors. They compare the accuracy and efficiency of different methods.

Keywords: Word Prediction, LSTM.

1. Introduction

The healthcare industry has a growing demand for a better understanding of human conditions, especially in elder care, recuperation, diabetes, and cognitive diseases. Using detectors and surveillance cameras can save money by detecting abnormalities in real-time. Wearable sensors have been shown to accurately identify human activities with low error rates, but most studies were done in laboratory settings that are not practical for real-world use. The use of smartphones as sensors for identifying human activities has the potential to revolutionize the way we gather data about human behavior. By using the built-in sensors of modern smartphones, we can obtain high-quality data that accurately captures a person's movements and activities, without the need for complex and intrusive sensors. The ability to place the phone in any position around the waist, such as a pocket, further increases the practicality and user-friendliness of this approach. It also has the potential to save significant amounts of money for caretakers, as real-time recording and reporting of abnormal gestures can help prevent and detect health issues early on. Overall, the use of smartphones as sensors has the potential to greatly improve the lives of patients.

2. Related Work

The proposed approach by the authors [1] involves the utilization of smartphone accelerometer and gyroscope data to detect activities, incorporating long short-term memory (LSTM), autoencoders (AEs) and convolutional neural networks (CNN).These techniques complement each other, with LSTMs being proficient at temporal modelling, AEs used for dimensionality reduction, and CNNs excelling at automatic feature extraction. To fully leverage their complementary capabilities, the authors integrated CNNs, AEs, and LSTMs into a single architecture in their study.

In this particular study [2], multiple machine learning classification techniques were utilized to detect human activities using sensor data from smartphones. The data obtained from the sensors of smartphone such as gyroscope and accelerometer were used to identify human activities. The study evaluated the efficiency and accuracy of the different methods applied in the classification process. The results of these methods were compared and analyzed.

Author. [3] introduced a novel approach for recognizing activities of human based on the principle of compressed sensing. The article reported an 86% accuracy rate for the recognition of human motion using this new approach. The authors also investigated the scenarios and examples where the recognition results could be accepted or rejected, which could lead to advancements in the area of recognition. The major challenges in measuring the performance of the approach and methods used for recognition is the validity of metrics. However, the new technique of model parameter optimization can now be applied to improve the quality of recognition. It is worth noting that activity recognition has applications beyond information technology, and medical researchers utilize it to enhance healthcare services.

Z Chen proposed a reliable human activity recognition (HAR) system that utilizes transformation of coordinates, Support Vector Machines (SVM) and CT-Principal Component Analysis (PCA) [4]. To mitigate the impact of direction fluctuations, the system employs CT-PCA. The proposed online SVM is independent and uses only a small amount of information from the hidden location, making it an efficient and effective approach for HAR.

For examining human activity recognition (HAR) operations using cellphones Ronao. [5] suggested utilizing a deep convolutional neural network (CNN). The results of their study showed that this technique has more sophisticated and relevant properties, making it a promising approach for HAR.

While the vision-based technique has a identification rate for HAR, utilizing cameras and video in interior settings raises privacy concerns and presents other technological challenges, making it impractical [6]. On the other hand, the wearablebased (smartphone) method is practical because of its compactness and adaptability, allowing for the use of sensorequipped devices for HAR.

In reference [7], the author utilized a data that is obtained by fixing smartphone on one of five body positions, including the left upper arm, right pocket, right upper arm and right wrist as a classifier for Support Vector Machines (SVM) to identify seven activities. The study achieved a maximum classification accuracy of 82%.

Gjoreski et al. [8] conducted a comparison between deep learning and machine learning algorithms and reported that deep learning algorithms performed somewhat better.

Abu Alsheikh analyzed recognition of activities in various datasets and employed paradigms of deep learning, which resulted in significant improvements in recognition compared to other methods for human activity recognition [9]. Smartphone activity identification systems [10] can be trained to identify data in the real-life using support vector machines as the classifier. The system requires both a recognition phase and a training phase, during which the acquired data is analyzed using activity models. One advantage of using sensors attached to these smart devices is that machine learning algorithms can be applied offline to analyze the data gathered for human activity detection. Even with smartphones having more powerful resources such as CPUs, memory, and batteries, the system can still be implemented in an online pattern.

In reference [11], the authors used a data by fixing smartphone in one of the user's front-right and back-left pant pocket along with a classifier called neural network to identify five different activities. The overall accuracy achieved was 93.311%. However, since the neural network needs to be trained using MATLAB on a PC, preparing the training datasets for this task can be challenging and time-consuming.

The work of author. [12] provides valuable insights into the main challenges faced in physical activity monitoring, which is dealing with problems of classification that exceed the capabilities of current classifiers.

Author. [13] used a smartphone with an sensor called accelometer to track a person's movement while performing different physical activities. The authors were able to correctly classify the different physical activities with a high accuracy rate of 98%.

Researchers in [14] searched for new approaches and strategies to improve the reliability and accuracy of smartphone sensors for activity recognition. They proposed a novel technique that combined the accelerometer of smartphone with a special chest sensor to identify human activities based on timing.

The use of various techniques of machine learning for HAR is an active research area, and LSTM RNNs are one of the popular choices due to their ability to capture long-term dependencies in sequential data. The combination of smart home sensors with LSTM RNNs is also interesting, as it can provide more comprehensive data about human activity in the home environment. The use of ANN for data capture module prototypes is also a common practice, as ANNs can efficiently process and extract features from large datasets[15].

In [16], the authors employed a fixed wearable sensor (Zephyr BioHarness) in the user's chest together with a smartphone fixed in one of the user's trouser pockets to identify 12 activities. The highest performance this study was able to obtain was 95%. Due to its size and ability to restrict breathing by tightening around the chest, the Zephyr BioHarness sensor is difficult for patients to use.

That sounds like an interesting study on using a digital low-pass filter to identify specific forms of human physical activity using mobile phone-generated acceleration data. The use of multiple human subjects and real-world usage simulation for training and evaluating the system is also promising. It's good to hear that different statistical features were considered and multiple classifiers were chosen to identify activities, and the effort to find the best possible set by combining these classifiers is commendable. Overall, this approach could potentially improve the accuracy and reliability of HAR systems using mobile phone sensors[17].

In this study, Hoseini-tabatabaei.[18] suggested a context-aware activity recognition framework for smartphones using multiple sensors, including accelerometer, gyroscope, and magnetometer. They utilized a sliding window approach to segment the sensor data and extract relevant features. A similarity search was then conducted to identify the most similar activity patterns, and a classifier called decision tree was employed to classify the activities. The suggested framework accomplished an average accuracy of 92.5% for six different activities.

The accurate information is one of the challenges that HAR [19] recognition faces because of the vast amount of processed data.Due to the necessity for daily life activities and accurate information, supervised learning presents privacy concerns because it uses actual information. However, in order to address the privacy concern, researchers developed a backgroundoperating smartphone sensor that is producing positive results in a cost-effective manner.

In their research paper, the author referenced as [20] utilized a smartphone that was securely attached to the user's chest and employed Support Vector Machine (SVM) as a classifier to successfully identify a total of 16 distinct physical activities. The overall recognition rate achieved through their experimental method was an impressive 95.03 percent, indicating a accuracy of high level in the classification of the activities recorded by the smartphone. This demonstrates the effectiveness of using SVM as a method for activity recognition, and provides valuable insight into the potential applications of such technology in fields such as healthcare and fitness tracking.

Anjum utilized sensors in smartphones to track various physical activities such as jogging, climbing, cycling, walking and driving. In order to measure the accuracy of the tracking, Anjum tracked the performance of several algorithms including Naive Bayes, Decision Trees, KNN (K Nearest Neighbors), and SVM[21].

The author of [22] utilized a smartphone attached to any of the locations-the waist, wrist, chest, or thighs-and applied SVM as a classification method to track activities. However, with only six actions being tracked, this may not be sufficient to predict user health. Nonetheless, the study achieved a 96% accuracy rate.

In [23], the authors were able to distinguish between five physical exercises by using a single accelerometer worn on the wrist. The exercises that were differentiated include strolling, sitting, lying down, standing and jogging. Additionally, the wearable device accelerometer was able to identify eight different activities, including computer usage.

In [24] explores the use of advanced sensors in mobile phones and smart wearables for identifying physical activities. The authors applied five different classifiers to recognize nine activities, including sitting, strolling, cycling, standing, going up, running and down stairs, and lifting heavy objects. However, the authors did not combine the sensor data from the two devices and instead analyzed each device's data separately. For the mobile phone, they employed a magnetometer, an accelerometer, pressure sensors, and a gyroscope, while for the smart wearable, they only used an accelerometer. In [25] used a stationary smartphone placed in the user's front trouser pocket along with Naive Bayes and Clustered KNN classifiers to recognize four activities. However, with only four activities being tracked, this may not be sufficient to predict user health. The study achieved a highest performance rate of 92%.

Baccouche [26] introduced a fully automated deep learning model for a video-based dataset, which does not require any prior knowledge. This model utilizes high-level feature extraction and takes advantage of recent advancements in deep learning to achieve promising results in various fields.

In [27], the authors utilized a smartphone positioned around the waist, such as in a jacket or jeans pocket, with arbitrary orientation, a classifiers called support vector machine, k-nearest neighbor and artificial neural networks were used to classify five activities. However, with only five activities being tracked, this may not be sufficient to predict user health. The study achieved an accuracy rate of 84.4%.

Sun proposed a HAR method utilizes smartphone accelerometer data with variable position and rotation. The fourth dimension of the data is the acceleration magnitude. The model also incorporates both generic and site-specific SVM [28]. Krishnan [29] conducted a study using data collected from ten individuals, where three wearable accelerometers were utilized to evaluate seven lower body actions. In the study by Mannini & Sabatini [30], five tri-axial accelerometers were employed, and machine learning techniques were used to identify users' movements and postures. Choudhury et al [31] employed seven separate sensors to recognize actions in their model.

HAR has been a subject of research for years, and numerous solutions have been proposed. Current methods often combine inertial sensors, visual sensors, or both, using either threshold-based techniques or machine learning. While thresholdbased algorithms are faster and easier to implement, machine learning methods typically produce more accurate and reliable results. Additionally, one or more cameras have been utilized to record and identify body posture [30, 33].

In addition to accelerometers, other available sensors have been the subject of studies. According to Maurer et al. [31], an "ewatch" device is placed in shirt pockets, pants pockets, backpacks, and other locations to identify typical user actions. The system created by the authors [32] employs bi-axial accelerometers located at five specific locations on the user's body to detect performed actions, which can aid in the promotion of health and fitness.

Later on, researchers combined accelerometers with other sensors for activity recognition. Parker, for instance, developed a mechanism that recognizes specific actions such as standing, moving about, jogging, playing ball, and using the restroom. Lee and Mase, on the other hand, investigated user location and activity recognition [34].

This study aims to explore the effectiveness of various algorithms of machine learning and deep learning in recognizing human activities using smartphone sensors. The authors propose using a similarity-based approach that compares the current sensor readings with previously recorded sensor data to identify the activity being performed. The performance of several classifiers, including Naive Bayes, Decision Trees, KNN (K Nearest Neighbors), and SVM, is evaluated. The study also aims to trackthe impact of different sensor placements on the accuracy of activity recognition. Overall, the paper seeks to contribute to the growing body of research using sensors of smartphone on recognition of human activities, with the goal of improving the accuracy and effectiveness of activity monitoring system.

3. Existing Systems

The previous system for monitoring and guiding human activities was entirely manual, requiring an individual to sit in front of a monitor and perform the necessary tasks. Unfortunately, this process was both arduous and timeconsuming, resulting in a costly system that was susceptible to human errors and negligence. As a result, there was a growing need for a more efficient and reliable system that could accurately recognize human activities in an open environment.

To address this challenge, to recognize human activities some systems began using sensor data. However, these systems typically required the user to wear sensors, which limited the scope of activity recognition in general. As a result, these solutions were not always effective in monitoring human behavior in public spaces or other open environments. Overall, there is a continued need to develop new and innovative solutions that can accurately recognize and monitor human activities in real-time, without imposing limitations on individuals or relying on manual processes. By doing so, we can enhance the effectiveness and efficiency of existing systems and promote a safer and more secure future for all.

4. Methodology

Our project aims to develop a smartphone application that can accurately classify human activities using sensor data from the phone's accelerometer and gyroscope. We explored several machine learning and deep learning techniques to categorize the sensor data and identify the physical activity performed by the user. The primary objective is to provide users with a seamless experience of tracking their physical activities using their smartphones. We compared the performance and accuracy of different classification methods to determine the most effective approach

4.1 Architecture

First, we gather a dataset from Kaggle. We check for errors and pre-process the data by converting categorical values to numerical ones and scaling larger values. We then divide the data and select the necessary features into training and testing sets. After applying the algorithm, we use performance metrics like accuracy, recall, and precision to evaluate the system's performance. Lastly, the system predicts the activity as output.

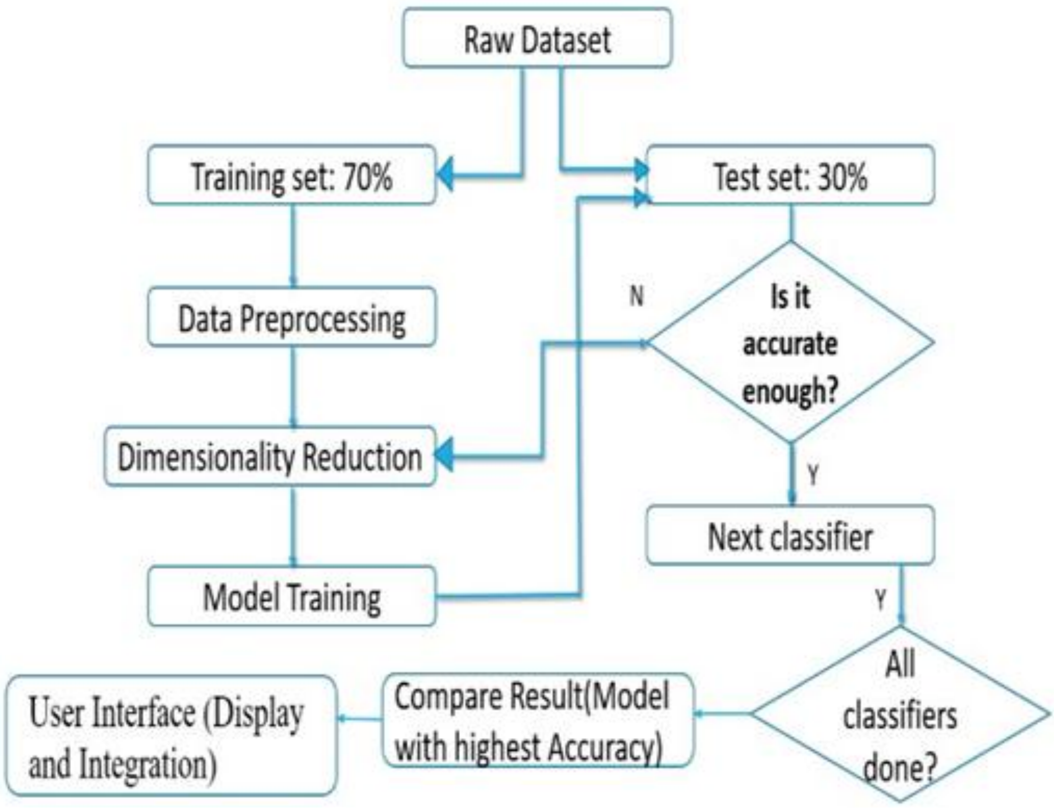


Fig 4.1: Architecture of Human activity recognition system

4.2 Module Design

The proposed system is designed with a module-based approach, where each module represents a specific step in the process of recognition of human activities. The design ensures a clear and complete flow of information throughout the system.

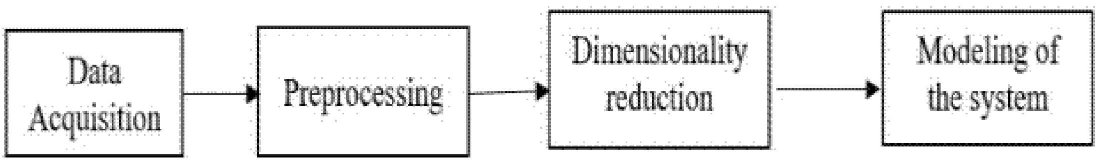


Fig 4.2: Process of Activity Recognition

4.2.1.Data Acquisition

To gather data, we used sensors like accelerometers and gyroscopes to measure physical conditions and converted the samples into digital numeric values.During the study, the researchers gathered data from 30 participants, aged 19-48 years old, who wore a Samsung Galaxy S II smartphone on their waist while performing six activities. The smartphone's embedded sensors, including the accelerometer and gyroscope, recorded the linear acceleration and angular velocity of the movements at a consistent rate of 50Hz. The data was split into two sets, with 70% for training and 30% for testing. We applied noise filters and sampled the signals in windows of 2.56 seconds with a 50% overlap. A low-pass filter separated the acceleration signal into body acceleration and gravity. Lastly, we extracted features from each window.

Accelerometer

Accelerometers are important in smart devices and wearables. They detect changes in the device's orientation relative to a reference point and adjust the direction to the user's viewing angle. For instance, when you turn your phone sideways, you can see a webpage in landscape mode. The camera mode also adjusts depending on the device's orientation. The accelerometer measures acceleration in 3D (X, Y, Z axes) during free fall to detect changes in orientation.

Fig 4.3: Smartphone Accelerometer Directions

Gyroscope

Gyroscopes help to maintain position, level, or direction based on angular momentum. When combined with accelerometers, they detect movement along the six axes (left, right, up, down, forward, and backward) and identify movements of yaw, roll, and pitch along the X, Y, and Z axes. MEMS gyroscopic sensors aid navigation and recognize gesture frameworks used in smartphones and tablets.

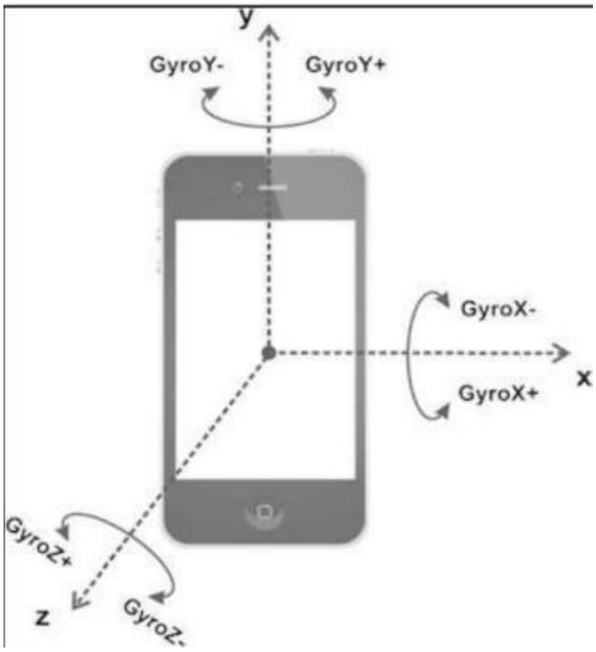


Fig 4.4: Gyroscope on Smartphone

4.2.2.Preprocessing

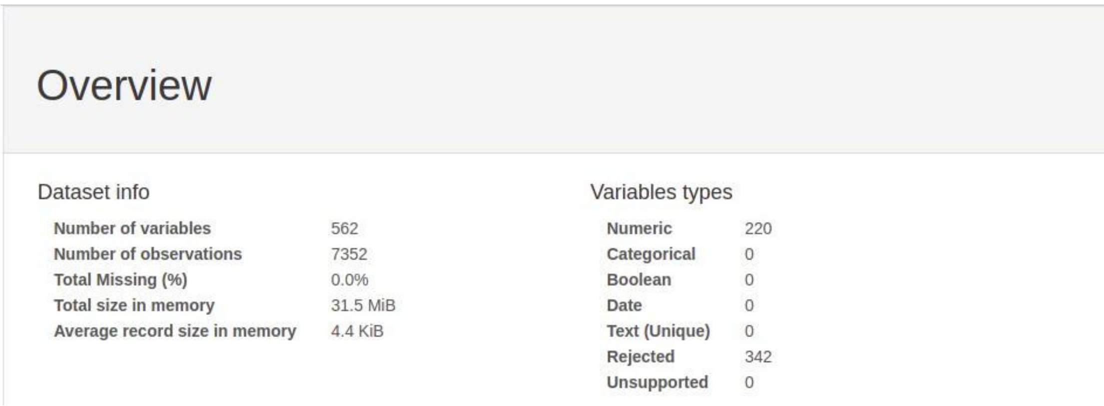
Data preprocessing is an essential step in preparing raw data for analysis or modeling. It involves various techniques to clean, transform, and format data to make it suitable for downstream processing. This step is crucial for machine learning projects, where input data quality can significantly impact the model's accuracy and reliability. Data may be incomplete, inconsistent, or contain errors or outliers. Preprocessing techniques, including cleaning, feature selection, and normalization, address these issues. Filtering methods are often used, with output from one step serving as input for the next. Data preprocessing ensures data quality and is critical for accurate and reliable data-driven analysis or modeling.

4.2.4 Dimensionality Reduction

Dimensionality reduction is a technique used in activity recognition to reduce the number of features while retaining information. This is useful because many datasets have a features in large number, leading to overfitting. Methods of dimensionality reduction include PCA, LDA, and t-SNE. PCA finds the principal components of a dataset, LDA maximizes the separation between different classes of activities, and t-SNE is used for visualization. Dimensionality reduction improves accuracy and provides insight into relationships between activities.

pandas profiling

Pandas profiling is a powerful tool for conducting quick and comprehensive exploratory data analysis of a Pandas Dataframe. It provides a summary of the dataset through descriptive statistics such as mean, data types, standard deviation, missing values and median. The primary objective of profiling is to gain a thorough understanding of the data to enable it to be queried and visualized in a variety of ways. The screenshots below demonstrate the usefulness of pandas profiling in gaining insights into a dataset.



Overview

Dataset info

Number of variables	562
Number of observations	7352
Total Missing (%)	0.0%
Total size in memory	31.5 MiB
Average record size in memory	4.4 KiB

Variables types

Numeric	220
Categorical	0
Boolean	0
Date	0
Text (Unique)	0
Rejected	342
Unsupported	0

Fig 4.5: Overview of the dataset

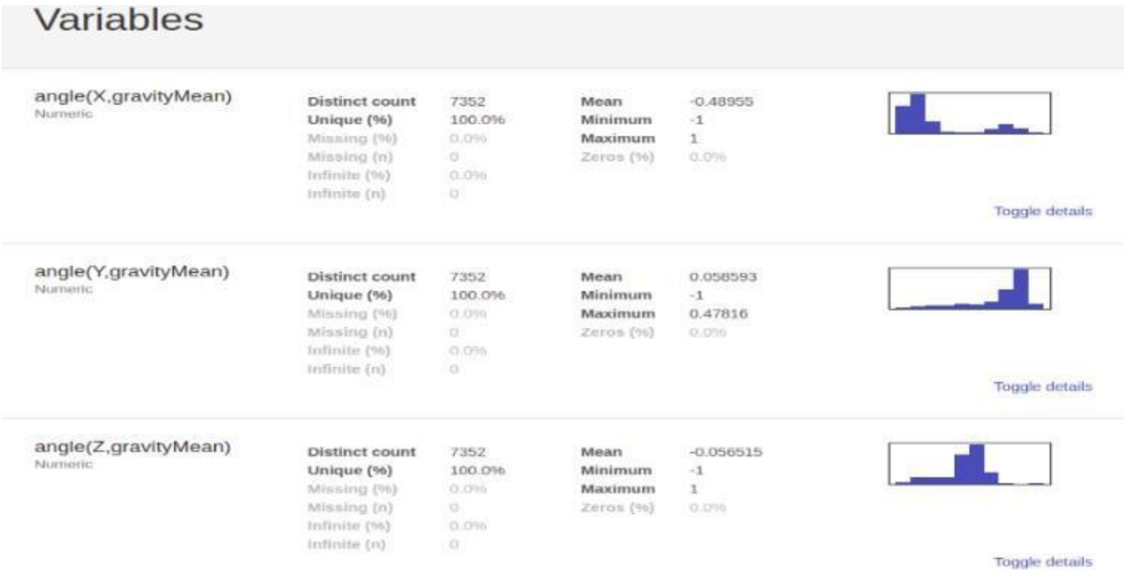


Fig 4.6: Details about the features in the dataset

Fig 4.7: Details about each individual feature

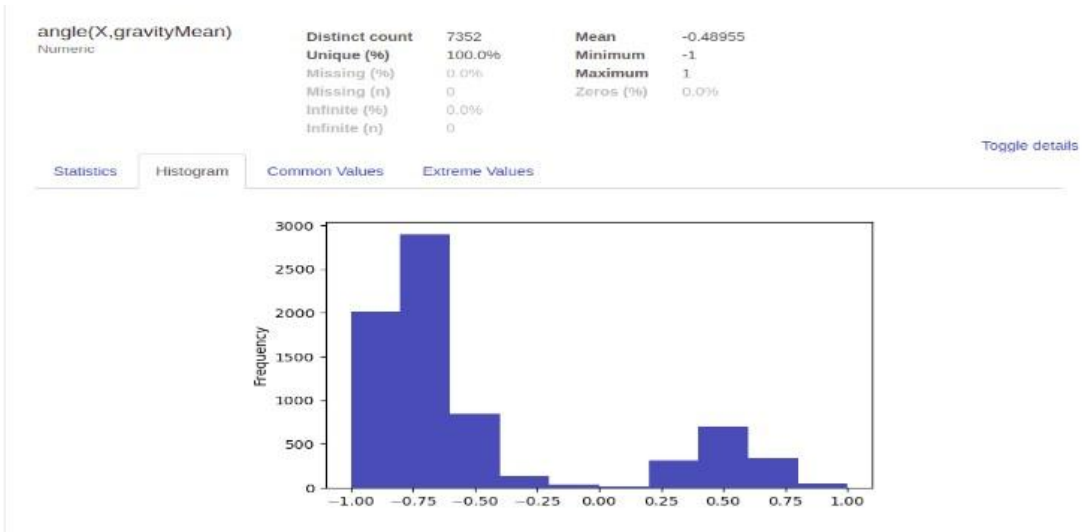


Fig 4.8: Distribution of the values of the features

4.2.5 Modeling of the system

The modeling process also involves choosing an appropriate algorithm or combination of algorithms based on the specific problem and data characteristics, selecting the best hyperparameters for the chosen algorithm(s), and implementing any necessary preprocessing steps to prepare the data for modeling. To consider factors such as scalability, computational efficiency and interpretability when selecting a model. Overall, the goal of modeling is to create a model that can accurately and efficiently make predictions on new, unseen data.

Importing the data using pandas and analyzing its shape and structure is an essential first step. Checking for duplicates and null values is important to ensure data quality. Visualizing the data using graphs and determining the balance of the data is also crucial. Feature engineering, such as changing feature names and differentiating between dynamic and static activities, can improve the quality of the input data. Using techniques like TSNE can also aid in visualizing and clustering the data. Converting categorical columns to numerical columns using LabelEncoder can also be useful for machine learning models. Finally, testing accuracy using deep learning and models of machine learning can help evaluate the effectiveness of the preprocessing and modeling techniques use.

5. Implementation

a. Input Dataset

The HAR dataset used in the study is a well-known dataset in the area of human activity recognition, and has been widely used in various studies and competitions. The dataset consists of smartphone sensors (accelerometer and gyroscope) measurements, providing detailed information about the participants' movements during daily activities. The manual labeling of the data ensures the accuracy of the ground truth labels, which is essential for the development of machine learning models.By randomly splitting the data into training and test sets, it becomes possible to assess the models' performance on new, unseen data, and also prevent overfitting.. Overall, the dataset provides a valuable resource for research in the area of human activity recognition.

b. Classification Algorithms

5.2.1 K-Nearest Neighbour

The k-nearest neighbour (kNN) algorithm divides the labels of the k closest training examples into the feature space and uses the results to classify unlabeled occurrences. Since it delays data processing until a classification request is made, kNN is a lazy learning algorithm., The code imports necessary libraries, loads train and test datasets, converts categorical values to numerical values, and initializes KNN classifier. It uses a loop to find the optimal number of neighbors for KNN algorithm and plots the accuracy scores. It also tests KNN algorithm with reduced features and principal components to find the optimal parameters for predicting human activities using smartphone sensor data.

KNN Algorithm

- Load data using a library like pandas.
- Choose the number of neighbors, K.
- To compute the K-Nearest Neighbors, determine the distance between the query example and each of the other examples in the dataset.
- Create an ordered collection that includes both the distance and the index of each example, then sort the collection in ascending order based on the distances.
- Select the first K entries in the sorted collection..
- Get the labels of the selected K entries.
- For regression, return the mean of the K labels; for classification, return the mode of the K labels.

- To determine the best K value for the K-Nearest Neighbors algorithm, it is necessary to run the algorithm several times using varying K values, and then select the K value that produces the highest accuracy when tested on new and unseen data.

Hyper parameters used: PCA(n_components=170, random_state=0) n_neighbors=17

5.2.2 Support Vector Machine

Support Vector Machines (SVMs) are a widely used type of supervised machine learning algorithm. They have been adapted for use in a variety of applications, including multiclass classification and regression analysis. SVM model trains with the default kernel 'rbf' and displays its train and test accuracy. It then loops through different kernel functions, trains the SVM model using each kernel, and displays its test accuracy. The test accuracy values for each kernel are stored in a list, which is later used to plot a bar chart using Seaborn's barplot function.The code then trains the SVM model again using the 'linear' kernel, as it provided the highest accuracy among the tested kernels. It displays the train and test accuracy values.Next, the code shows how to choose the value of the regularization parameter C, which is used to control the comprmise between the smoothness of the decision boundary and the degree to which it correctly classifies the training data.The SVM model is then trained using the PCA-transformed train dataset, and its train and test accuracy values are displayed. The results show that the SVM model's accuracy decreases slightly after PCA, but it is still high enough to be useful for classification.
Next, It loops through different numbers of principal components and trains the SVM model using each number. It then displays the test accuracy values for each number of principal components, which are stored in a list that is later used to plot a scatterplot. The results show that the SVM model's accuracy peaks at around 190 principal components and then starts to decrease. This suggests that reducing the dimensionality of the dataset beyond this point does not provide much benefit in terms of classification accuracy.

SVM Algorithm

- Begin by importing the libraries: numpy, pandas, seaborn, and matplotlib.
- Load the train and test datasets into pandas DataFrames using the read_csv() function.
- Encode the activity labels of both the train and test datasets using the LabelEncoder() function.
- Train the model with a linear kernel and calculate both the training and test accuracy.
- Evaluate the efficiency of the SVM model by testing different kernel functions, including 'rbf', 'linear', 'poly', and 'sigmoid'.
- Use the seaborn library to plot the accuracy of the SVM model for each kernel function.
- Select a regularization parameter value (C) and evaluate the efficiency of the SVM model with various C values, using a linear kernel.

- To simplify the training dataset, implement Principal Component Analysis (PCA) to reduce its dimensionality.
- Assess the effectiveness of the Support Vector Machine (SVM) model using the reduced dimensionality dataset that was obtained through PCA.
- Determine the optimal number of components for maximum accuracy and plot the accuracy against the number of components.
- Evaluate the performance of the SVM model using the optimal number of components obtained from PCA.
- Use the seaborn library to plot the confusion matrix for the SVM model.

Hyper parameters used: kernel='linear', C=1.0 PCA(n_components=190, random_state=0)

5.2.3.XG Boost

XGBoost is a powerful distributed gradient boosting library designed to be highly effective, adaptable, and portable.

It is built on the Gradient Boosting framework and is capable of implementing a range of machine learning algorithms. Based on sensor data from a smartphone XGBoost classifier classifies activities. It reads and encodes the data, trains and tests the classifier, performs feature selection and dimensionality reduction, and plots the results. The code then iterates over learning rates and principal components to optimize accuracy, applies the classifier, evaluates performance, and outputs a confusion matrix and classification report.

XG Boost Algorithm

- Load the training and testing data from CSV files.
- Encode the labels using sklearn's LabelEncoder.
- Train a default XGBoost classifier and report the train and test accuracy.
- Evaluate the impact of different learning rates on accuracy by training classifiers with varying learning rates and recording their test accuracy.
- Evaluate the impact of selecting different features by training classifiers with different feature subsets.
- Evaluate the impact of dimensionality reduction using PCA on accuracy by training classifiers with different numbers of components and recording their test accuracy.
- Evaluate the impact of different learning rates on accuracy after dimensionality reduction by training classifiers with varying learning rates and recording their test accuracy.
- Report the final train and test accuracy of the XGBoost classifier using the selected features and optimal number of components.
- The algorithm outputs multiple accuracy scores and visualizations to aid in interpreting the results.

Hyper parameteres used:

learning_rate=1

PCA(n_components=180, random_state=0)

5.2.4.Logistic Regression

Logistic regression is a statistical method that examines the correlation between one or more independent variables, which can be classified as nominal, ordinal, interval, or ratio-level, and a binary dependent variable. It is particularly suitable when the dependent variable has a binary or dichotomous nature, meaning it can take only two possible outcomes. Firstly, it separates the subject information and labels from the training and testing data. The subject information is stored separately, while the labels are stored as the target variable.

Next, it drops the subject and activity columns from the data, as they are not needed for classification.It then uses the LabelEncoder function from the scikit-learn library to encode the categorical activity labels as numerical target labels.The next part is to initializes a logistic regression model and trains it on the training data and target labels. The trained model is then used to predict the activity labels for the testing data.

5.2.5. Convolution Neural Networks

Classifier s	accuracy	precision	recall	F1
XGBoost	0.934	0.9358	0.9326	0.9335
KNN	0.908	0.9136	0.9032	0.905
Logistic Regressio n	0.9643	0.9666	0.9639	0.9643
SVM	0.9633	0.965	0.9627	0.9634
CNN	0.9602	0.9613	0.9602	0.9601

Convolutional Neural Networks (CNNs) are a particular kind of neural network that are often used in the context of Human Activity Recognition (HAR) to analyse time series data, such as accelerometer and gyroscope signals obtained from wearable sensors.First, it reads the train and test data from CSV files, separates the features and labels, and replaces the activity names with integer labels.The Keras library is then used to define the CNN model. Two 1D convolutional layers make up the model's foundation, which is then followed by two dense layers and a final output layer with softmax activation. The sparse categorical cross-entropy loss function is used in the model's construction along with the stochastic gradient descent optimizer.Using the fit() function, the model is trained across 100 epochs.

CNN Algorithm

- Import necessary libraries and modules
- Read train and test dataset files into Pandas DataFrames
- Extract features and labels from DataFrames
- Convert activity labels to integers
- Define function to create CNN model with Keras API
- Reshape features to fit CNN model
- Compile model with loss function, optimizer, and metrics
- Train model with fit function using train and validation data

- Evaluate trained model with evaluate function using test data
- Predict labels on test data and calculate confusion matrix
- Visualize confusion matrix with Seaborn heatmap

Hyperparameters
Conv1D layer, Dense layers, Optimizer, Loss function, Metrics

5.4 Comparision of Algorithms

Table 1: Tabular representation of evaluation metrics for each classifier

CLASSIFIER	ACCURACY
Support Vector Machine	96.3352
K-Nearest Neighbour	90.8042
XG Boost	93.4509
Logistic Regression	93.3495
Convolusion Neural Network(CNN)	96.335

Table-1 displays the performance metrics of several classifiers used to classify data. The first column lists the name of each classifier, which includes XGBoost, KNN, Logestic Regression, SVM, and CNN. The remaining columns contain different performance metric for each classifier, including Accuracy, Precision, Recall, and F1-score.

6. Results

Table 2: Tabular representation of obtained accuracy

Classifiers	Training Accuracy	Test Accuracy
XGBoost	0.9994	0.9395
KNN	0.9842	0.9002
Logistic Regression	0.9955	0.9644
SVM	0.979	0.9504
CNN	0.9536	0.9223

Table-2 displays the training and test accuracy of different classifiers used in a machine learning task. The first column lists the names of each classifier, which includes XGBoost, KNN, Logistic Regression, SVM, and CNN. The second and third columns show the trainingand test accuracy, respectively.

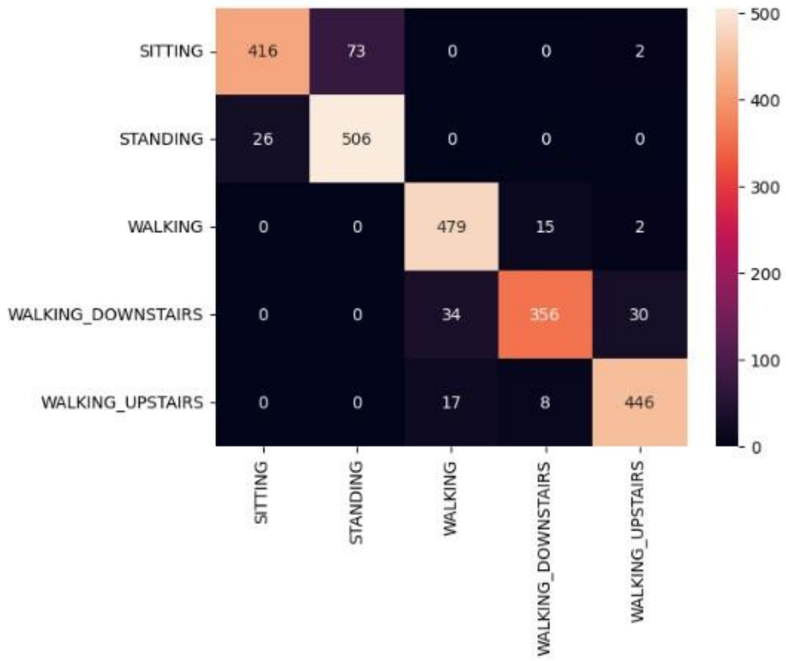


Figure 5.1: Confusion Matrix for XGBoost

In figure 5.1, we can observe that sitting and standing, walking and walking downstairs values are getting overlapped.

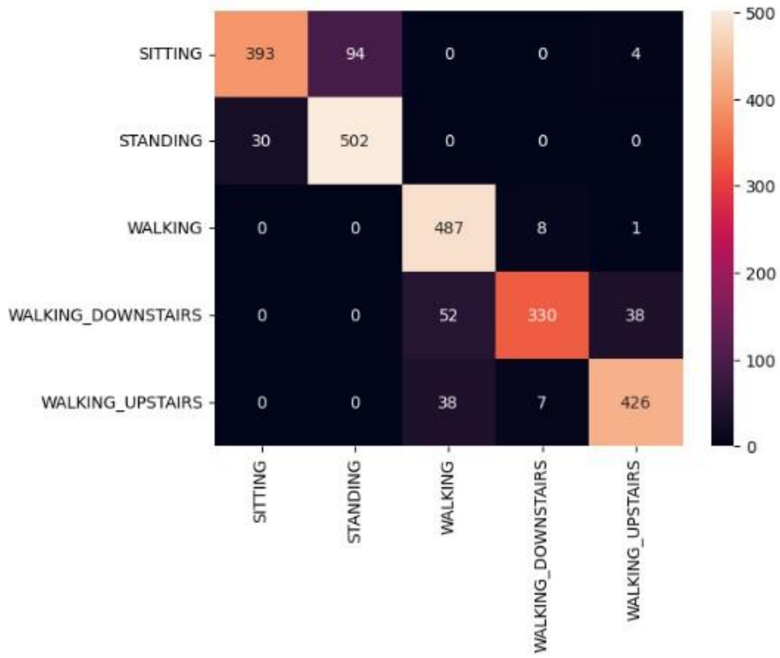


Figure 5.2: Confusion Matrix for KNN

In figure 5.2, we can observe that sitting and standing, walking and walking downstairs values are getting overlapped.

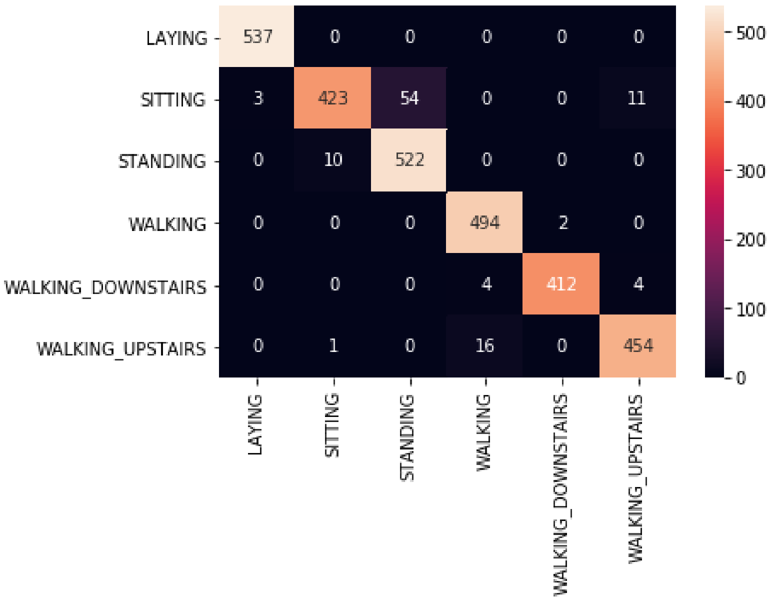


Figure 5.3: Confusion Matrix for Logistic Regression

In figure 5.3, we can observe that sitting and standing, values are getting overlapped.

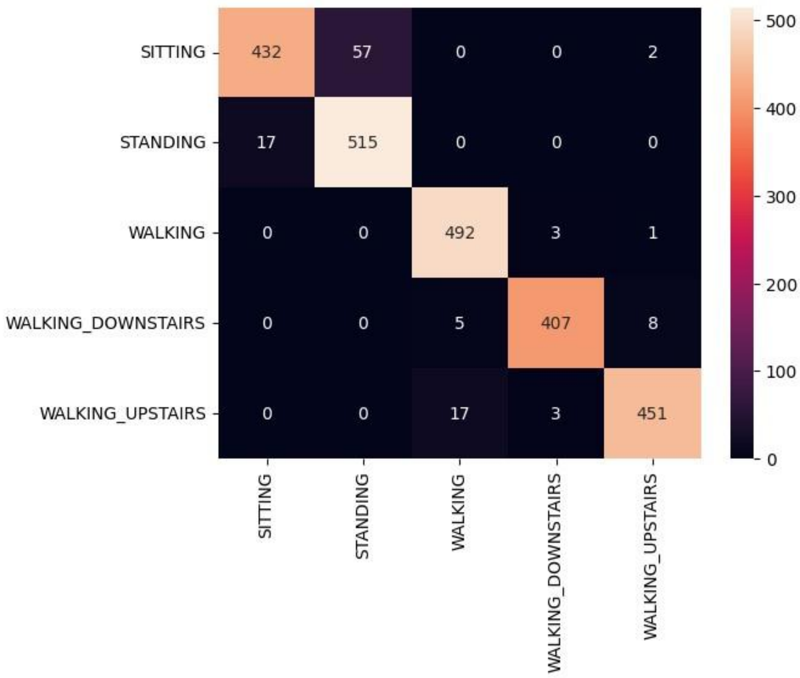


Fig 5.4: Confusion Matrix for SVM

In figure 5.4, we can observe that sitting and standing values are getting overlapped.

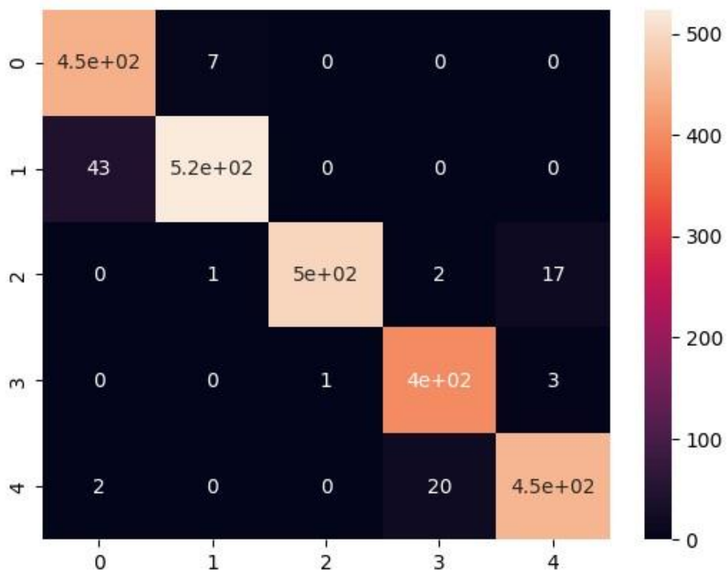


Fig 5.5: Confusion Matrix for CNN

DEMO:

Fig 5.5: Home page of SmartFitON

Figure 5.5 displays the home page of SmartFitOn's mobile application, which provides an overview of the app's functionalities and usage instructions.

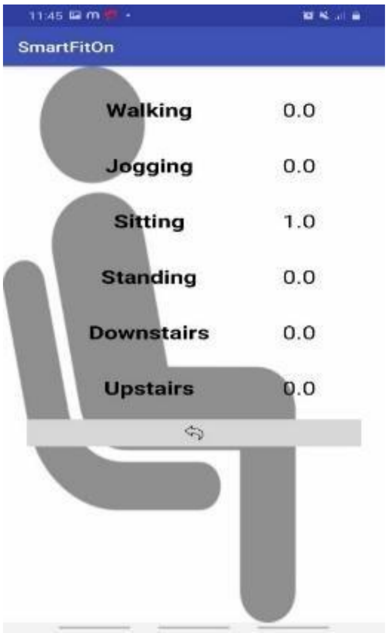


Fig 5.6: Sitting.

The image shown in Figure 5.6 depicts a representation of a person sitting, which has been identifiedby the system as having the higest confidence rate for the sitting posture.

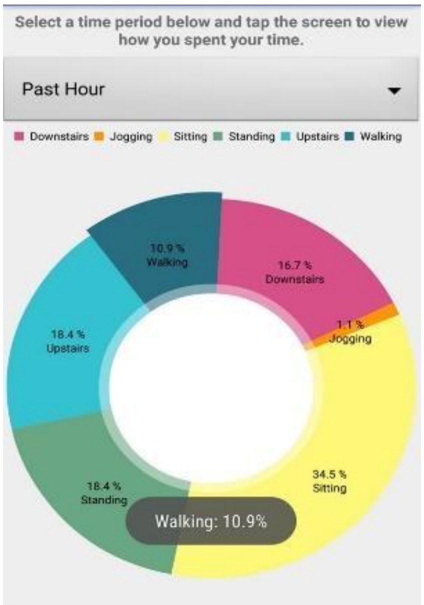


Figure 5.7: pie chart for past hour

Figure 5.11 presents a pie chart that displays the various activities performed by the user over the past hour.

Figure 5.8: piechart for past month.

The pie chart shown in Figure 5.8 displays the various activities performed by the user over the past month.

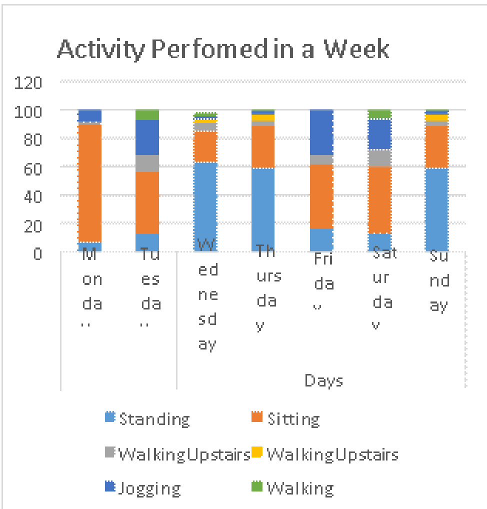


Fig:5.9 Activities perfromed in a week

The segmented bar graph shown in Figure 5.9 displays the various activities performed by the user in past week.

7. Conclusion

This project made a system that employs a smartphones accelerometer to identify six distinct human activities. We gathered data, extracted features, reduced dimensionality, and utilizing various machine learning algorithms including KNN, SVM, Logistic Regression, XGBoost and CNN to train and evaluate the model. The best accuracy achieved was 96.33% with SVM and CNN, showing the effectiveness of these algorithms and the importance of using machine learning for activity recognition.

8. Enhancements

Future work could include expanding the number of activities recognized by the system and deploying a real-time version on a smartphone platform. Additionally, it would be beneficial to explore other query strategies, such as variance reduction and density-weighted methods, to improve the performance of the active learning approaches used in this study.

References

[1] Dipanwita Thakur, Suparna Biswas, Edmond S. L. Ho, Samiran, “ConvAE-LSTM: Convolutional Autoencoder Long Short-Term Memory Network for Smartphone-Based Human Activity Recognition", vol. 10, pp. 99, January 2022.

[2] Erhan Bulbul, Aydin Cetin , Ibrahim Alper Dogru , “Human Activity Recognition using Smartphones", In Proc. of the 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), December 2018. [3] Lu Y., Wei Y., Liu L., Zhon J., Sun L., Liu Y., “Towards unsupervised physical activity recognition using smartphone accelerometers”, Multimedia Tools Applications, vol.76, pp.10701-10719, April 2017.

[4] Z. Chen, Q. Zhu, Y. C. Soh, and L. Zhang, “Robust human activity recognition using smartphone sensors via ct-pca and online svm”, IEEE Transactions on Industrial Informatics, vol. 13, no. 6, pp. 3070-3080, 2017.

[5] Ronao C. A. and Cho S.-B, “Human activity recognition with smartphone sensors using deep learning neural networks”, Expert Systems with Applications, vol. 59, no. 15, pp.235-244, October 2016.

[6] Aiguo Wang, Guilin Chen, Jing Yang, Shenghui Zhao, Chih-Yung Chang, “A comparative study on human activity recognition using inertial sensors in a smartphone”, IEEE Sensors Journal, vol.16, no.11, July 2016.

[7] D. Natarajasivan and M. Govindarajan, "Filter Based Sensor Fusion for Activity Recognition using Smartphone", International Journal of Computer Science and Telecommunications, vol. 7, no. 5, pp. 26-31, 2016.

[8] H. Gjoreski, J. Bizjak, M. Gjoreski, M. Gams, “Comparing deep a classical machine learning methods for human activity recognition using wrist accelerometer”, in Proc. the 25th International Conference on Artificial Intelligence, pp. 1-7, 2016.

[9] M. A. Alsheikh, A. Selim, D. Niyato, L. Doyle, S. Lin, H.P. Tan, ”Deep activity recognition models with triaxial accelerometers”, ArXiv e-prints, November 2015.

[10] Shoaib M., Bosch S., Incel O., Scholten H., Havinga P, “A Survey of Online Activity Recognition Using Mobile Phones”, Sensors, vol. 15, no. 1, pp. 2059-2085, January 2015.

[11] T. H. Cesar and A. L. Andres, "Accelerometer-Based Human Activity Recognition in Smartphones for Healthcare Services", in Mobile Health A Technology Road Map, Springer International Publishing Switzerland, pp.147-169, January 2015.

[12] Reiss A., Hendeby G., Stricker D., “A novel confidence-based multiclass boosting algorithm for mobile physical activity monitoring”, Personal and Ubiquitous Computing, vol. 19, pp.105-121, September 2014.

[13] Arif M., Bilal M., Kattan A., Ahamed S. I, “ Better Physical Activity Classification using Smartphone Acceleration Sensor”, Journal of Medical Systems, vol. 38, July 2014.

[14] Guiry J. J., Ven P. van de, Nelson J., Warmerdam L., Riper, H, “Activity recognition with smartphone support”, Medical Engineering and Physics,, vol. 36, no. 6, pp. 670-675, June 2014.

[15] S. Oniga and J. Suto, “Human activity recognition using neural networks, ”in Proceedings of the 2014 15th International Carpathian Control Conference (ICCC). IEEE, vol.6, no.1, pp. 403-406, May 2014.

[16] Y. Bingchuan, H. John and E. Yalda, "Smartphone-based Activity Recognition Using Hybrid Classifier-Utilizing Cloud Infrastructure for Data Analysis", in Proceedings of the 4th International Conference on Pervasive and Embedded Computing and Communication Systems, Lisbon, Portugal, January 2014.

[17] A.Bayat, M. Pomplun and D. Tran, "A Study on Human Activity Recognition Using Accelerometer Data from Smartphones*", Procedia Computer Science*, vol. 34, pp. 450-457, 2014.

[18] S. A. Hoseini-Tabatabaei, A. Gluhak, R. Tafazolli, “A Survey on smartphone-based systems for opportunistic user context recognition”, ACM Computing Surveys, vol. 45, no.3, pp. 1-51, June 2013.

[19] Martín H., Bernardos A. M., Iglesias J., Casar J.R, “Activity logging using lightweight classification techniques in mobile devices”, Personal and Ubiquitous Computing, vol. 17, no. 4, pp.675-695, April 2013.

[20] H. Yi and L. Ye, "Physical Activity Recognition Utilizing the Built-In Kinematic Sensors of a Smartphone", International Journal of Distributed Sensor Networks, vol. 9, April 2013. [21] A. Anjum and M. U. Ilyas, “Activity recognition using smartphone sensors”, in 2013 ieee 10th consumer communications and networking conference (ccnc). IEEE, pp. 914-919, January 2013.

[22] A. Davide, G. Alessandro, O. Luca, P. Xavier and L. R.-O. Jorge, "A Public Domain Dataset for Human Activity Recognition Using Smartphones", in European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, Bruges Belgium, ESANN, pp. 437-442, 2013.

[23] Varkey J.P., Pompili D., Walls T.A, “Human motion recognition using a wireless sensor-based wearable system”, Personal Ubiquitous Computing, vol. 16, no.7, pp. 897-910, September 2012.

[24] Scholl P.M., Van Laerhoven K, “A Feasibility Study of Wrist-Worn Accelerometer Based Detection of Smoking Habits”, Proceedings of the 2012 Sixth IEEE International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), Palermo, Italy, 4-6 July 2012.

[25] K. Mustafa, I. Ozlem Durmaz and E. Cem, "Online Human Activity Recognition on Smart Phones", in 2nd International Workshop on Mobile Sensing, Beijing, China, January 2012.

[26] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, A. Baskurt, “Sequential deep learning for human action recognition”, in Proc. the Second International Conference on Human Behavior Unterstanding, pp. 29-39, November 2011.

[27] R. Amin, C. Chien-An and L. Yan, "Human Activity Recognition using Smartphone", CSCE666, 2011.

[28] L. Sun, D. Zhang, B. Li, B. Guo, and S. Li, “Activity recognition on an accelerometer embedded mobile phone with varying positions and orientations”, in International conference on ubiquitous intelligence and computing, Springer, pp. 548-562, October 2010.

[29] N. Krishnan, D. Colbry, C. Juillard, S. Panchanathan, “Real time human activity recognition using tri-Axial accelerometers”, in Proc. Sensors, Signals and Information Processing Workshop, January 2008. [30] T.B.Moeslund, A.Hilton, V.Kruger, “A survey of advances in vision-based human motion capture and analysis”, Computer Vision Image Understanding, vol.104, no.2-3, pp.90-126, November-December 2006.

[31] U. Maurer, A. Smailagic, D. P. Siewiorek, M. Deisher, ”Activity recognition and monitoring using multiple sensors on different body positions”, in Proc. International Workshop on Wearable and Implantable Body Sensor Networks (BSN’06), May 2006.

[32] L. Bao and S. S. Intille, “Activity recognition from user-annotated acceleration data”, in Proc. Pervasive 2004, pp. 1-17, 2004.

[33] R. Bodor, B. Jackson, and N. Papanikolopoulos. “Vision-based human tracking and activity recognition”, In Proc. of the 11th Mediterranean Conf. on Control and Automation, June 2003. [34] S.W. Lee and K. Mase, ”Activity and location recognition using wearable sensors”, IEEE Pervasive Computing, vol. 1, no. 3, pp. 24-32, July 2002.