Human Activity Recognition Using Machine Learning Algorithms

By

Mim Rahman
181-35-2292

A thesis submitted in partial fulfillment of the requirement for the degree of Bachelor of Science in Software Engineering

Department of Software Engineering

DAFFODIL INTERNATIONAL UNIVERSITY

Summer – 2021

APPROVAL

This thesis titled on "Human Activity Recognition Using Machine Learning Algorithms", submitted by Mim Rahman, ID: 181-35-2292 to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Software Engineering and approval as to its style and contents.

BOARD OF EXAMINERS

---------- Chairman

Dr. Imran Mahmud
Associate Professor and Head
Department of Software Engineering
Daffodil International University

---------- Internal Examiner 1

Kaushik Sarker
Assistant Professor
Department of Software Engineering
Daffodil International University

---------- Internal Examiner 2

Md. Shohel Arman
Senior Lecturer
Department of Software Engineering
Daffodil International University

---------- External Examiner

Md. Fazle Munim
Technology Expert
Access to Information (a2i) Programme

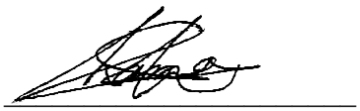
DECLARATION

It hereby declares that this thesis has been done by me under the supervision of **Syeda Sumbul Hossain**, Senior Lecturer, Department of Software Engineering, Daffodil International University. It is also declared that neither this thesis nor any part of this has been submitted elsewhere for award of any degree.

Mim Rahman

Student ID: **181-35-2292**
Batch**:** 25th
Department of Software Engineering,
Faculty of Science & Information Technology,
Daffodil International University

Certified by:



Syeda Sumbul Hossain

Senior Lecturer,
Department of Software Engineering,
Faculty of Science & Information Technology,
Daffodil International University

ACKNOWLEDGEMENT

First of all, I am grateful to the Almighty for giving us the ability to complete the final thesis.

I would like to express my gratitude to my supervisor **Syeda Sumbul Hossain** for the consistent help of my thesis and research work, through her understanding, inspiration, energy, and knowledge sharing. Her direction helped me to find the solutions of research work and reach my final theory.

I would like to express my extreme sincere gratitude and appreciation to all of our teachers of the **Software Engineering** department for their kind help, generous advice and support during the study.

I am also expressing my gratitude to all of our friends, seniors, juniors, who directly or indirectly have lent their helping hand in this venture.

Last but not the least, I would like to thank our family for giving birth to me in the first place and supporting me spiritually throughout my life.

Mim Rahman

181-35-2292

TABLE OF CONTENTS

LIST OF ABBREVIATION

Terms	Abbreviation
HAR	Human Activity Recognition
DL	Deep Learning
MLP	Multi-layer perceptron
KNN	K-Nearest Neighbors
RF	Random Forest
NB	Naive bayes
DT	Decision Tree

ABSTRACT

Human activity recognition (HAR) is an important part of people's daily lives because it allows them to get high-level information about human actions from raw sensor input data. HAR is the difficulty of characterizing daily human activity using data acquired from smartphone sensors in a single statement. The accelerometer and gyroscope collect data on a continual basis, and these data are crucial in forecasting human behaviors like walking or standing. On this subject, there are several databases and continuing study. The COVID-19 pandemic is the world's most pressing problem now, thus it's critical to follow or record the everyday actions of those who live alone or isolate themselves. As a result, human activity recognition is crucial in the medical profession. Human Activity Identification (HAR) is used in a variety of applications, including eldercare, medical, sport activity monitoring, surveillance, emotion recognition, and training. Human Activity Recognition is the subject of a lot of study and research. However, in most of the paper, there are just two or three models. What we know is that the more models we evaluate with more data, the better model and accuracy we will find. Pre-processing of data, training and testing with selected models, evaluation of outcomes (accuracy), and better model prediction for HAR are the phases of the system model. I have taken the "Human Activity Recognition with Smartphones Dataset (2019)" to apply machine learning methods. Kaggle was used to get this dataset. The Human Activity Recognition database was created using recordings of 30 research participants conducting activities of daily living (ADL) while wearing a smartphone with inertial sensors attached to their waist. The datasets I took in this paper were separate for train and test, and the data was taken to a smart device through a sensor as already mentioned in the dataset description.

If I compare between my proposed methods, then the Random Forest method works well in Human Activity Recognition which is evidenced by the comparison table.

I evaluate the models with the confusion matrix of almost ten thousand + data with train and test data, so the table proves that the Random Forest model gives better performance than other models. I will build an Activity Recognizer app in that recognition process using the primary dataset up front. This model may be used to predict Human Activity. To pass on estimations and execute the model on such estimations, an Android application may be used. Furthermore, in the near future, we will achieve greater accuracy if we employ more unique and richer datasets. This will benefit our medical and robotics industries.

Keywords: Human Activity Recognition, Random Forest, KNN, XGBoost, Naive Bayes, Machine Learning Algorithms.

CHAPTER 1

INTRODUCTION

1.1 Background

The number of old individuals has been steadily growing over the last few decades, raising serious worries among academics about their comfort and quality of life. This is because advances in information and communication technologies (ICTs) have piqued the interest of a large number of individuals (particularly young people), resulting in desolation and isolation among chronic patients. As a result of the shortage of keepers' services, it is difficult for individuals to take care of themselves. In this context, great emphasis is placed on developing new methods to harness the benefits of ICTs in order to enable chronically sick and special-needs persons to live independently while also encouraging a sense of blessing. Although, employing sensor and actuator networks, the design and deployment of internet of things (IoT) based smart home technologies can aid in the management of daily functioning.

Human activity recognition (HAR) is an important part of people's daily lives because it allows them to get high-level information about human actions from raw sensor input data. HAR is the difficulty of characterizing daily human activity using data acquired from smartphone sensors in a single statement. The accelerometer and gyroscope collect data on a continual basis, and these data are crucial in forecasting human behaviors like walking or standing. On this subject, there are several databases and continuing study.

HAR is an extremely tough issue that has been used in a variety of fields, including restoration and limiting. This analysis has adjusted for several models, the number of sensors and sensor placements, and highlight developments over the preceding 10 years. What is evident from this

collection of previous opinions is that the unambiguous applications drive the assessment of the best solutions for each circumstance.

1.2 Motivation

In the biomedical and healthcare systems, monitoring physical activity and posture allocation is critical. Obesity may be induced by extended automobile driving and physical activity such as walking, standing, and lying, according to study, and prostate cancer is linked to sitting. Aside from that, aberrant physical patterns and everyday activities might lead to additional disorders of this nature. Children with fewer physical activities have a higher risk of developing autism, according to research. Individuals with amyotrophic lateral sclerosis and post-stroke have a similar situation. To build a trustful mechanism in current healthcare systems, great precision (for example: more than 99 percent) and dependability of activities are required. As a result, walking and other activities are significant clinical parameters for predicting the risk of falling, which is a regular occurrence. The evaluation of an activity metric, on the other hand, can only be useful if real-time data is retrieved.

Moreover, the COVID-19 pandemic is the world's most pressing problem now, thus it's critical to follow or record the everyday actions of those who live alone or isolate themselves. As a result, human activity recognition is crucial in the medical profession.

Human Activity Identification (HAR) is used in a variety of applications, including eldercare, medical, sport activity monitoring, surveillance, emotion recognition, and training.

1.3 Problem Statement

Human Activity Recognition is the subject of a lot of study and research. However, in most of the paper, there are limited models. What we know is that the more models we evaluate with more data, the better model and accuracy we will find.

1.4 Research Questions

1. Which techniques are more effective for Human Activity Recognition (HAR)?

2. Which methods give better results through evaluation metrics?

1.5 Research Objectives

To propose a model which would assist in recognizing human activity (For example: Walking, Standing, Sitting, Laying, Climbing Stairs etc.) information taken from different sensors data by comparing different machine learning algorithms.

1.6 Research Scopes

On a continuous basis, this model may be used to predict Human Activity. To pass on estimations and execute the model on such estimations, an Android application may be used. Furthermore, in the near future, we will achieve greater accuracy if we employ more unique and richer datasets. This will benefit our medical and robotics industries.

1.7 Thesis Organization

The following is a list of the contents of this paper: The first chapter introduced Human Activity Recognition (HAR), including the history and motivation for the issue, the problem description, scopes, and so on. The overview of recent material in Chapter 2 is followed by the development of certain HAR models. The datasets utilized in this investigation are described in Chapter 3, which is followed by an explanation of each component of the technique employed in this study. In Chapter 4, the findings of numerous tests are given and debated, and then the conclusion is offered in Chapter 5.

CHAPTER 2

LITERATURE REVIEW

2.1 Background

Machine learning and deep learning approaches have been developed in response to extensive study in order to improve the HAR. Depending on the application requirements, previous research on the topic may differ in a variety of ways. The aims and scope of such projects vary greatly, for example, data gathering techniques, signal processing, feature extraction and selection, classification, and prediction. However, before discussing categorization and prediction, it appears that one of the most important needs is the availability of real-time data sets. Because proper categorization using machine learning algorithms is based on the genuine data set.

Kholoud Maswadi, Norjihan Abdul Ghani[1], Suraya Hamid and Muhammads Babar Rasheed have used Naive Bayes (NB), Decision Tree (DT) algorithm for this paper. They attained highest accuracies of 89.5 percent and 99.9 percent using NB and DT classifiers with Gaussian filters.

To increase accuracy and save processing time, some characteristics can be added or deleted using the filter, wrapper, or embedding techniques. This research employed a filter-based feature selection method. As a result of the recursive feature removal, future study may examine wrapper-based techniques.

The authors have used Decision Tree[2] (DT), Support Vector Machine (SVM), Random Forest (RF), Artificial Neural Network (ANN) algorithms to get the accuracy and for prediction. To discover the optimal hyperparameters, they employed 5-fold stratified cross-validation and grid search. Then, using test data, run the models with the best parameters, followed by the statistical significance test. The SVM method outperforms the other two algorithms.

Simple machine learning algorithms can perform well with correct parameter tuning, and statistical testing can be used to establish the significance of the outcome.

The authors have implemented[3] K-Nearest Neighbors (KNN), Random Forest (RF), Feature Selection Using Neighborhood Component Analysis (FSCNCA) models. The precision attained by utilizing K-NN decreases as the range of feature selection decreases. It changes depending on the raw data characteristics that were chosen. This indicates that the precision attained by employing an RF classifier decreases as the range of the feature selection decreases. In the future, the categorization presentations will be expanded and executed on a constant basis.

In this paper K-Nearest Neighbors [4] (KNN), Decision Tree (DT), Naive Bayes (NB), Artificial Neural Network (ANN), Support Vector Machine (SVM) models have been used. The findings show that using a compressed training dataset can reduce the time it takes to update the HC classifier, and that the k-NN and DT methods are the best two. That suggests the DL-based HC classifier might perform better in terms of accuracy while categorizing 12 unique activities. Future work will take into account a greater number of actions as well as more complicated circumstances as new challenges.

The authors have used Artificial Neural Network (ANN) [5], K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Naive Bayes (NB), Classification and Regression Tree (CART), Decision Tree (DT), REP Tree, LAD Tree, Random Tree (RT), Random Forests (RF) models here. All of the approaches used functioned admirably. The top degree of accuracy for SVM is 99.43%, which is the best of all the classification techniques tested. To attain high accuracy and increase healthcare functionality, quality, and safety, future HAR should be built to foresee and recognize these concurrent processes and be capable of handling ambiguity.

In this paper, Multilayer Perceptron [6], Support Vector Machine, Gaussian Naive Bayes (NB), K-Nearest Neighbor (KNN), Random Forest (RF) have been implemented. Although these solutions are quick and simple to use, they do have certain drawbacks owing to poor performance in a variety of scenarios. They suggest a unique way for boosting the performance of existing machine learning algorithms for HAR in this study, which is based on ensemble learning.

The authors have implemented [7] Random Forest (RF) algorithm. This study on the classification algorithms used in human activity identification using a smartphone is incredibly important for analysts to have a better understanding of the research flow patterns in the field of human activity recognition. Modified RF is used to measure the dataset's precision. On Android stages with limited assets, the modified RF grouping performs far better than the RF classifier in terms of precision. In terms of execution times, we also evaluated the execution of modified RF. In addition, setup times are highly dependent on the gadget model and capabilities.

2.2 Summary

Maximum work applies Naive Bayes and Decision Tree machine learning methodologies, as explained in the preceding section. It is obvious that deep learning methods should be investigated for HAR detection. For this study, the performance of numerous machine learning models was compared to that of the deep learning model.

CHAPTER 3

RESEARCH METHODOLOGY

3.1 Design of the System Model Related to this Research

Pre-processing of data, training and testing with selected models, evaluation of outcomes (accuracy), and better model prediction for HAR are the phases of the system model linked to this research are showing in Figure 3.1:

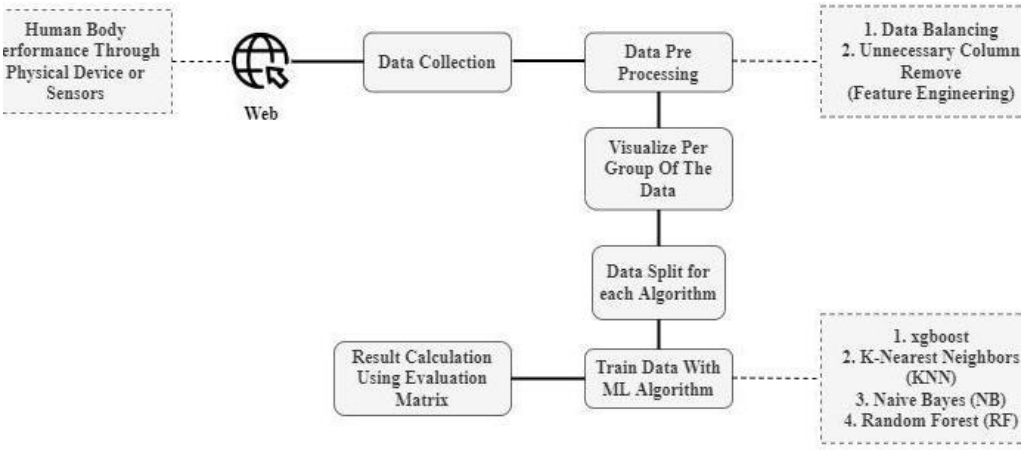


Figure 3.1: Working Procedure

3.2 Data Collection and Dataset

I have taken the "Human Activity Recognition with Smartphones Dataset (2019)" to apply machine learning methods. Kaggle was used to get this dataset. The Human Activity Recognition database was created using recordings of 30 research participants conducting activities of daily living (ADL) while wearing a smartphone with inertial sensors attached to their waist.

The trials were carried out on a group of 30 participants ranging in age from 19 to 48 years old. Each participant did six activities while wearing a smartphone (Samsung Galaxy S II) around their waist (Walking, WalkingUpStairs, WalkingDownStairs, Sitting, Standing, Laying). 3-axial linear

acceleration and 3-axial angular velocity at a constant rate of 50Hz using the device's internal accelerometer and gyroscope has been recorded. The tests were videotaped so that the data could be manually labeled. The collected dataset was randomly partitioned into two sets, with 70% of the volunteers being chosen to provide training data and 30% being chosen to generate test data. The sensor data (accelerometer and gyroscope) were pre-processed using noise filters before being sampled in 2.56 sec fixed-width sliding windows with 50% overlap (128 readings/window). A Butterworth low-pass filter was used to separate the gravitational and body motion components of the sensor acceleration data into body acceleration and gravity. Because it is expected that the gravitational force has only low frequency components, a filter with a cutoff frequency of 0.3 Hz was utilized. Calculating variables from the time and frequency domain yielded a vector of characteristics from each frame.

The use of smart electronic devices integrated into wearable objects or directly with the body to measure biological and physiological sensor signals such as heart rate, blood pressure, body temperature, accelerometers, or other attributes of interest like motion and location is referred to as the wearable sensor method. These sensors are linked to an integration device, such as a smartphone, laptop, or bespoke embedded system. As a result, raw signals are forwarded to an application server for monitoring, visualization, and analysis in real time. For activity tracking, a smart phone with multiple sensors, such as a gyroscope, camera, microphone, light, compass, accelerometer, proximity, and GPS, can be highly useful. The raw data from a smartphone, on the other hand, is only useful for simple actions and not for sophisticated ones. As a basis, new sensors or monitoring devices should be employed to improve identification performance.

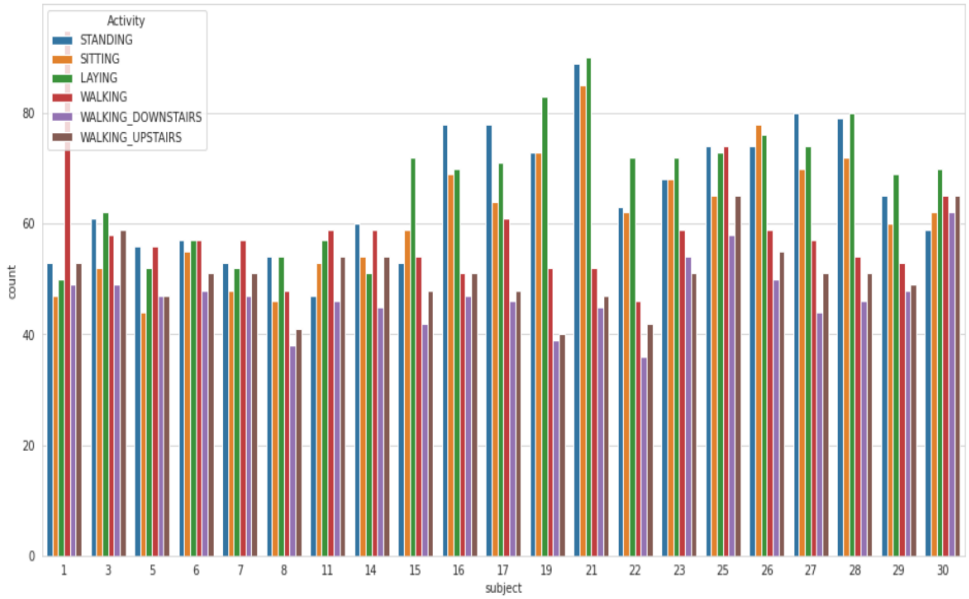


Figure 3.2: Subject per activity count

Here the activities are counted according to each field subject.

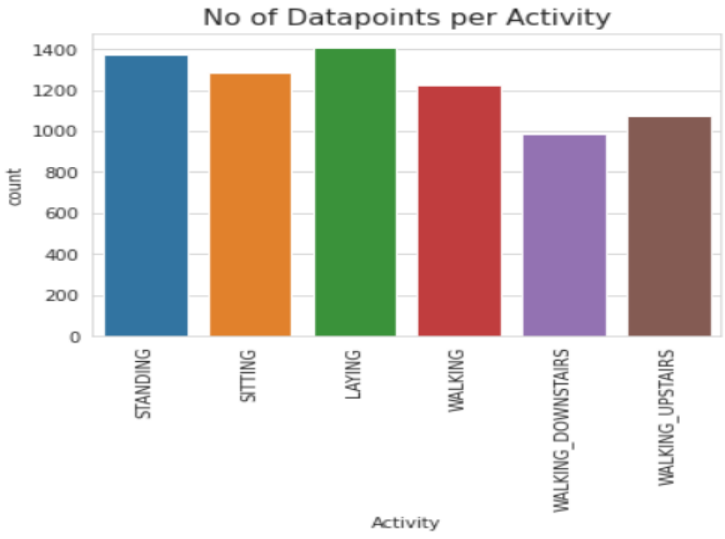


Figure 3.3: Data Points per Activity

Activities are counted in the form of data points for each field.

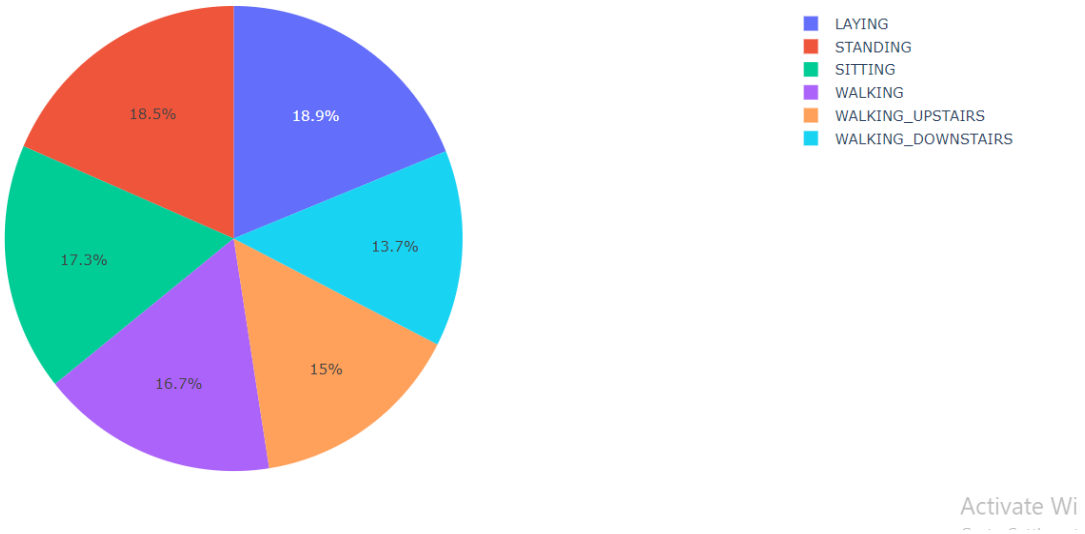


Figure 3.4: Activity of human bodies after drop some columns

After the data balance, the ratio is calculated by counting each field.

3.3 Machine Learning Based Classification Algorithms

The Classification method is a Supervised Learning approach that uses training data to identify the category of fresh observations. In the classification algorithms, our program learns from the dataset and classifies into some groups or classes based on the data. Such as email spam or not spam or animal classification (dog, cat etc).

Main goal of the classification algorithm is to identify categories from the given dataset and our classification algorithm used to predict the classification (Classification Algorithm in Machine Learning, n.d.). There are some types of classification algorithms such as **linear models** and **non-linear** models.

Whether linear models have some methods such as logistic regression and support vector machines. On the other hand, non-linear models have some methods like KNN (K-Nearest Neighbours), Naïve Bayes, Decision Tree, Random Forest etc.

However, in this paper we use non-linear methods for classification and compare those methods through evaluation metrics.

3.3.1 KNN (K-Nearest Neighbours)

Full abbreviation of KNN is “K-Nearest Neighbors”, its type of classification non-linear algorithm. Whether the KNN method is known as supervised machine learning algorithms. This method is used for both classification and regression problems. The sign 'K' represents the number of nearest neighbors to a new unknown variable that has to be predicted or categorized. The goal of KNN algorithms is to find all of a new unknown data point's nearest neighbors in order to figure out what class it belongs to. It's a strategy centered on distance. Can culate the distance from all the points in the nearest point of unknown data and filter the shortest distance to it through this method. That's why that method is called a distance based algorithm (Sharma, 2021).

KNN prefers the approach of "majority voting" when the problem statement is of the "classification" kind. The class with the most votes is picked from the range of K values provided.

3.3.2 KNN Model Training Procedure

In this paper we used 10000+ data of human bodies from smartphones, first of all remove all the unnecessary columns and prepare the dataset again.

```
# x, x_test, y, y_test = train_test_split(X,Y,test_size=0.2,train_size=0.8)
x_train, x_validation, y_train, y_validation = train_test_split(X,Y,test_size=0.2,random_state=254)
```

Figure 3.5: Split data and take random state

I take 20% (2947 data) data as a test set and 80% data (7352 data) as a train set (figure 3.5) and import the model library. Take class as K values seven (figure 3.6) and calculate the accuracy through evaluation metrics.

```
knn = KNeighborsClassifier(n_neighbors=7)

knn.fit(x_train, y_train)
```

```
KNeighborsClassifier(n_neighbors=7)
```

Figure 3.6: Take K neighbors as seven

3.3.3 XGBoost

The XGBoost algorithm is a decision tree based ensemble of machine learning algorithms that is used as a gradient boosting framework. Artificial neural networks surpass all other algorithms or frameworks in prediction issues involving unstructured data (pictures, text, etc.). "Extreme Gradient Boosting" is what XGBoost stands for. XGBoost is a distributed gradient boosting toolkit that has been tuned for efficiency, flexibility, and portability. It uses the Gradient Boosting framework to create Machine Learning algorithms. It uses parallel tree boosting to tackle a wide range of data science issues quickly and accurately. On the other hand, XGBoost is used in supervised learning as a regression and classification problem (Subia, n.d.).

3.3.4 XGBoost Model Training Procedure

I take a parameter (figure 3.6) in the XGBoost method, in this parameter I take some values such as estimators - 100, learning rate - 0.2, sub sample - 0.927 max depth - 5, random state - 12 and so on. And as usual we train our model with 10000+ data of the human bodies.

```
xgb_params = {'n_estimators': 100,
              'learning_rate': 0.2,
              'subsample': 0.927,
              'colsample_bytree': 0.88,
              'max_depth': 5,
              'booster': 'gbtree',
              'reg_lambda': 38,
              'reg_alpha': 32,
              'random_state': 12}
model = XGBClassifier(**xgb_params)
model.fit(x_train, y_train)
```

```
XGBClassifier(colsample_bytree=0.88, learning_rate=0.2, max_depth=5,
              objective='multi:softprob', random_state=12, reg_alpha=32,
              reg_lambda=38, subsample=0.927)
```

Figure 3.7: XGBoost parameter

3.3.5 Naive Bayes

Naive Bayes is a classification approach that presupposes predictor stability and employs Bayes' Theorem. To put it another way, a Naive Bayes classifier assumes that the existence of one feature in a class is unrelated to the existence of any other feature (Ray, 2017). Mainly Naive Bayes used as a text classification from high dimensional training dataset. Bayes theorem known as Bayes rule or Bayes law (*Naive Bayes Classifier in Machine Learning*, n.d.). The formula is,

$$P(A \mid B) = \frac{P(B \mid A)\, P(A)}{P(B)} \text{ ----------(1)}$$

Where (1) P(A | B) is a posterior probability of hypothesis A's probability on the observed occurrence B. And P(B|A) is likelihood probability given that the likelihood of a hypothesis is true, the probability of the evidence (*Naive Bayes Classifier in Machine Learning*, n.d.).

Suppose I have a conditional based dataset in Naive Bayes method working, first of all convert the dataset into frequency table and then generate likelihood table by finding probability of the features. And finally use Bayes theorem to calculate the posterior probability.

3.3.6 Naive Bayes Model Training Procedure

First of all I fit our train data with the Naive Bayes method, and put the validation of X through a variable. As usual I train our model with 10000+ data of the human bodies.

3.3.7 Random Forest

Random Forest is a well-known supervised machine learning method. In machine learning, it may be utilized for both classification and regression issues. It is based on ensemble learning, which is a method of integrating several classifiers to solve a complicated issue and increase the model's performance. Random Forest is a classifier that combines the results of several decision trees on different subsets of a dataset and averages them to increase the dataset's predicted accuracy. In random forest when compared to other techniques, it takes less time to train, predicts output with good accuracy, and it runs quickly even with a huge dataset and when a considerable amount of the data is missing, it can still retain accuracy.

First of all random forest select random k points from the train set, secondly build the decision tree linked with the selected some points. Choose the N number of decision trees that we want to build. Finally, given new data points, find each decision tree's prediction.

3.3.8 Random Forest Model Training Procedure

Whenever training this method, we take one (1) as a random state (figure 3.7) when splitting our dataset. We find accuracy of this model through evaluation metrics with 10000+ data.

```
X_train, X_test, y_train, y_test = train_test_split(X, Y, random_state=1, stratify=Y)
```

```
forest = RandomForestClassifier()
forest.fit(X_train, y_train)
```

```
RandomForestClassifier()
```

```
y_pred_test = forest.predict(X_test)
```

3.7: Random Forest Data splitting

CHAPTER 4
RESULTS AND DISCUSSION

I have used confusion metrics to evaluate those methods in this paper. Through this way, I have calculated accuracy, precision value, recall value and f1 score.

I first import some of the required libraries of the method after data collection. I check whether the data has null value or missing value. I balance the data, eliminating some unnecessary columns. The data is reprocessed and trained through those methods, and I take a random state for each method separately. I split the data as 80% for the model train and 20% for the test. Finally I evaluate the models through the Confusion matrix. I compare all the methods through Confusion Matrix, which model will actually give better performance.

4.1 Confusion matrix

Confusion matrix is a prediction summary result of a classification problem. Moreover, the value is calculated by summarizing the correct or incorrect predictions through the Confusion matrix, divided according to their class (Brownlee, 2016). Precision, recall and f1 score are measured through this confusion matrix.

The TP, TN, FP, FN embedded in this matrix means,

- True Positive (TP): Which we have predicted as positive but which is true.
- True Negative (TN): Which we have predicted as negative but which is true.
- False Positive (FP): Which we have predicted as positive but which is false.
- False Negative (FN): Which we have predicted as negative but which is false.

Recall: Measurements are made through recall, how many classes have been correctly predicted from all the positive classes (Narkhede, n.d.).

Recall: TP / TP+FN

Precision: Measurements are made through precision, how many of the classes we have predicted are actually positive (Narkhede, n.d.).

Precision: TP / TP+FP

F-measure (Accuracy): Measurements are made through f1 score, from all the classes, whether it is positive or negative, we have been able to predict how many classes correctly.

F-measure: 2 x (Precision x Recall) / (Precision + Recall)

4.2 Model Confusion matrix Result

I took data separately for each model, separately for the train and separately for the test. I have evaluated these methods in this paper using Confusion Matrix. The result of Confusion Matrix for each model is as under:

Table 3.9: Confusion Matrix of Data Tested with XGBoost

LAYING	1.00	1.00	1.00	381
SITTING	0.95	0.95	0.95	342
STANDING	0.95	0.96	0.95	384
WALKING	0.97	0.99	0.98	347
WALKING_DOWNSTAIRS	0.97	0.99	0.98	307
WALKING_UPSTAIRS	0.99	0.96	0.98	299
accuracy			0.97	2060
macro avg	0.97	0.97	0.97	2060
weighted avg	0.97	0.97	0.97	2060

Table 4.0: Confusion Matrix of Data Tested with K-nearest neighbours

	precision	recall	f1-score	support
LAYING	1.00	1.00	1.00	381
SITTING	0.91	0.88	0.90	341
STANDING	0.90	0.93	0.91	385
WALKING	0.98	1.00	0.99	352
WALKING_DOWNSTAIRS	1.00	0.98	0.99	311
WALKING_UPSTAIRS	1.00	1.00	1.00	290
accuracy			0.96	2060
macro avg	0.97	0.96	0.96	2060
weighted avg	0.96	0.96	0.96	2060

Table 4.1: Confusion Matrix of Data Tested with Naive Bayes

	precision	recall	f1-score	support
LAYING	0.99	0.79	0.88	381
SITTING	0.49	0.91	0.64	341
STANDING	0.85	0.35	0.49	385
WALKING	0.92	0.72	0.81	352
WALKING_DOWNSTAIRS	0.81	0.76	0.78	311
WALKING_UPSTAIRS	0.65	0.91	0.76	290
accuracy			0.73	2060
macro avg	0.78	0.74	0.73	2060
weighted avg	0.79	0.73	0.72	2060

Table 4.2: Confusion Matrix of Data Tested with Random Forest

	precision	recall	f1-score	support
LAYING	1.00	1.00	1.00	486
SITTING	0.98	0.95	0.97	444
STANDING	0.96	0.98	0.97	477
WALKING	0.99	0.98	0.98	431
WALKING_DOWNSTAIRS	0.97	0.97	0.97	351
WALKING_UPSTAIRS	0.97	0.98	0.98	386
accuracy			0.98	2575
macro avg	0.98	0.98	0.98	2575
weighted avg	0.98	0.98	0.98	2575

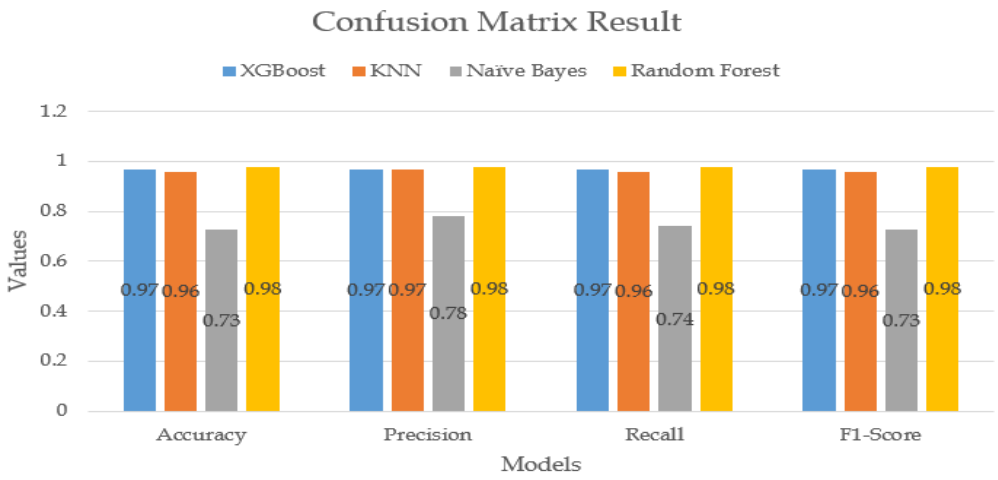


Figure 4.3: Confusion Matrix of all the Methods in Bar Graph

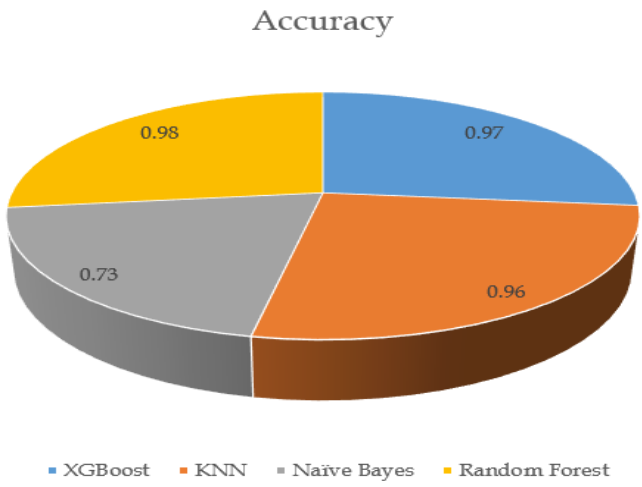


Figure 4.4: Accuracy of all the Methods in Pie Chart

Table 4.5 : The experimental results and comparison with the methods

Models	Accuracy
XGBoost	0.97
KNN	0.96
Naïve Bayes	0.73
Random Forest	0.98

From the table (table: 4.5) of our experimental results, it is seen that I have been able to get good accuracy of all the algorithms through that confusion matrix. I have already shown all the values of the confusion matrix through bar graphs (figure: 4.5), F1 score value, Precision value, Recall value respectively.

I have already shown the values of the confusion matrix in a separate table for each method. Accuracy of each method is good for dataset balancing (Sánchez & Patel, 2020), thus removing unnecessary data. So the correlation of one variable with another of the confusion matrix is strong. However, if I look at the accuracy of each method from our comparison table (table: 4.5), the result of the confusion matrix of the Random Forest method is better than other methods. Where Random Forest achieves **98% accuracy**, the accuracy of other methods is relatively low. Therefore, the Random Forest method plays a very good role in Human Activity Recognition, compared to some of the methods I use.

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1 Findings and Contribution

In fact, Human Activity Recognition plays a very good role, especially in various health care units. This type of machine learning based algorithm is also useful for various health tips or health care apps. In the human body, the function of the body changes with the movement of each of its organs, so it is difficult to understand in which condition the function of the body is changing. So it needs different types of electric devices to count immediately. However, it is quite challenging to count this functionality from the human body, then a good wearable sensor is needed, and it is possible to send data from that sensor to different smart devices.

However, there is some research on Human Activity Recognition that uses a single machine learning algorithm, but the main purpose of this paper is to recognize Human Activity through Multiple Machining Learning Algorithms and compare the performance of those Multiple Algorithms.

The datasets I took in this paper were separate for train and test, and the data was taken to a smart device through a sensor as already mentioned in the dataset description.

However, if I compare between my proposed methods, then the **Random Forest** method works well in Human Activity Recognition which is evidenced by the comparison table (Table: 4.5).

I evaluate the models with the confusion matrix of almost ten thousand + data with train and test data, so the table (table: 4.5) proves that the Random Forest model gives better performance than other models. I will build an Activity Recognizer app in that recognition process using the primary dataset up front.

5.2 Future Work

In the future I will work on Convolutional Neural Network and Artificial Neural Network Deep Learning by collecting data through wearable devices. Because the CNN method facilitates better feature selection. And finally, I will introduce a system through which the activity of the human body can be counted very quickly.

References

(n.d.). *Human Activity Recognition on Smartphones using Machine Learning Algorithms*.

Polu, S. K., & Polu, S. K. (2018). Human activity recognition on smartphones using machine learning algorithms. International Journal for Innovative Research in Science & Technology, 5(6), 31-37.

Brownlee, J. (2016, November 18). *What is a Confusion Matrix in Machine Learning*. Machine Learning Mastery. Retrieved January 1, 2022, from https://machinelearningmastery.com/confusion-matrix-machine-learning/

Classification Algorithm in Machine Learning. (n.d.). Javatpoint. Retrieved December 30, 2021, from https://www.javatpoint.com/classification-algorithm-in-machine-learning

Naive Bayes Classifier in Machine Learning. (n.d.). Javatpoint. Retrieved December 31, 2021, from https://www.javatpoint.com/machine-learning-naive-bayes-classifier

Narkhede, S. (n.d.). *Understanding Confusion Matrix | by Sarang Narkhede*. Towards Data Science. Retrieved January 1, 2022, from https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62

Ray, S. (2017, September 11). *Learn Naive Bayes Algorithm | Naive Bayes Classifier Examples*. Analytics Vidhya. Retrieved December 31, 2021, from https://www.analyticsvidhya.com/blog/2017/09/naive-bayes-explained/

Sánchez, E., & Patel, K. (2020, November 3). *Imbalance Dataset: Increasing Accuracy in Machine Learning Using 'imblearn'*. Medium. Retrieved January 1, 2022, from https://medium.com/swlh/imbalance-dataset-increasing-accuracy-in-machine-learning-using-imblearn-9cf1399e2319

Sharma, S. (2021, May 15). *KNN - The Distance Based Machine Learning Algorithm*. Analytics Vidhya. Retrieved December 30, 2021, from https://www.analyticsvidhya.com/blog/2021/05/knn-the-distance-based-machine-learning-algorithm/

Subia, J. (n.d.). *XGBoost Algorithm: Long May She Reign! | by Vishal Morde*. Towards Data Science. Retrieved December 31, 2021, from https://towardsdatascience.com/https-medium-com-vishalmorde-xgboost-algorithm-long-she-may-rein-edd9f99be63d

Wearable Sensor Data Based Human Activity Recognition using Machine Learning: A new approach. (n.d.). Nguyen, H. D., Tran, K. P., Zeng, X., Koehl, L., & Tartare, G. (2019). Wearable sensor data based human activity recognition using machine learning: a new approach. arXiv preprint arXiv:1905.03809.

