Human Activity Recognition with Sensor Data From Smart Devices

Taha Muhsin YILMAZ
161180505
Gazi University
Computer Engineering Dept.
tahamuhsinyilmaz@gmail.com

Görkem ŞÖLEN
151180055
Gazi University
Computer Engineering Dept.
realgorkem.23@gmail.com

Tugay KAĞAN
141180037
Gazi University
Computer Engineering Dept.
tugay.kagan@gmail.com

***Abstract*: Nowadays, with increasing technology, many people have mobile phones. In this article, we aimed to develop a system which can detect the human activity recognition with the gyroscope information found in the phones. In this way, the human activity recognition can be detected and used in healthcare applications or for Industry 4.0. When developing this system, we used a data set containing 6 movements. These movements are 'Biking', 'Sitting', 'Standing', 'Walking', 'Stair Up' and 'Stair down'. We tested this dataset with different machine learning algorithms and artificial neural networks. We got the most successful result with Random Forest. This system can be used for improve the results at future studies.**

Keywords: Human Activity Recognition (HAR), Machine Learning Algorithms, Artificial Neural Networks

I. INTRODUCTION

Nowadays, with the development of technology, many ideas and applications that can make people's lives easier are developing. Human Activity Recognition, which is one of these developing ideas, provides a lot of convenience to people's lives. If this estimate can be achieved successfully, this study can contribute positively to human life in many ways. These contributions may be about health. For example, the analysis and follow-up of the movements of the elderly people in their physical treatments can be achieved with this idea. Another example is that even people who do not have any health problems can have a healthy life by following their movement. In addition, this technology can be a big step towards the smart cities that will be the technology of the future and may contribute to the development of this smart technology. This human activity recognition is provided through sensors on people. These sensors can be used in people's daily lives with the accelerometer, gyroscope sensors in the phone. In this way, instead of using sensors that affect the quality of life, people do not feel the effect on the them by taking data from the sensors found in devices such as phones and watches they use in daily life.

II. RELATED WORKS

Mohammed Mobark and colleagues investigated recognizing of complex activities with common classifiers those using in recognizing human activities. The effect of two positions investigated; armband and waist-mounted. The subject performs four main activities; preparing for the breakfast, preparing tea, preparing sandwich, then clean up. Those activities are categorized into three levels as hierarchy. The processed data was classified by seven classifiers: C4.5(J48), Logistic Regression (LR), Neural Network (NN, multilayer perception), Naïve Bayes (NB), K-nearest Neighbor (KNN, IBK), Decision Table (DT), and Support Vector Machine (SVM, SMO). The experiment shows that the recognition accuracy of low-level activities is higher than high level in all seven classifiers. Also, it was noticed that the highest accuracy was got by IBK classifier (KNN) [2].

Sebastian Scheurer and colleagues explored three methods for reducing the dimensionality of a HAR problem in the context of an emergency first responders monitoring system. The study empirically estimates the accuracy of KNN, SVM and Gradient Boosted Trees when using different combinations of accelerometer, gyroscope and pressure sensors. The results show that the best combination is that which includes all three sensors (Mean Absolute Error: 3.6%) where the best performance was achieved with the GBT algorithm, followed by the A/G (MAE: 3.7%), and the A/P combination (MAE 4.3%): the same as that when using the accelerometer alone. The results also show that the Kruskal-Wallis test can be used to discard up to 50% of the features, and yet improve the performance of classification algorithms [6].

Purpose of the data preprocessing is to reduce the noise from the user and the sensors themselves. Training and testing on the same group of multiple data has the second highest accuracy. The accuracy decreases when the test data is collected from same subject but on different days. The lowest accuracy is in the setting where the training data is collected from one subject on one day and testing is conducted on another subject on a different day. Different classifiers and rule-based strategies are used for different activity recognition and the transition states. Activity recognition is a core building block behind many interesting applications. The classified applications of mobile activity recognition according to their targeted beneficial subjects are; health and fitness monitoring, personal biometric signature, elder-care and assistance, app activity log, indoor localization and navigation [1].

Dengpan Tian and colleagues proposed a HAR method based on inertial sensors and barometer. The proposed method recognizes eight human activities following a multi-layer strategy: sitting, standing, lying, walking, downstairs, upstairs, running and fall. Activities are classified into two categories: dynamic and static activities; then explicit activity recognition is taken individually in the two categories by dynamic or static classification. Three

classifiers are adopted for different classification, including random forest and SVM. Different feature sets have been selected for different classifiers which are more targeted and effective. In addition, the classifier result is further verified by additional parameters and previous recognition results to decide the final recognition result. The average recognition accuracy is above 96% [4].

Study of Dengpan Tian and colleagues collected data of five behaviors: stand, walk, run, upstairs, downstairs from the accelerometer on a smart watch. These data are classified with CART decision tree. In proposed method, the frequency domain features are replaced with polar coordinate conversion, which reduces the dimensions of the feature vector. After calculations are done, the normal method has highest accuracy with running as 95%, when the proposed method has %98 accuracy with same activity [3].
Hui Huang and colleagues designed a novel triboelectric motion sensor in wearable body sensor network for HAR. The experiments are conducted to collect five common activity data: sitting and standing, walking, climbing upstairs, downstairs, and running. The KNN clustering algorithm is adopted to recognize these activities and validate the feasibility of this new approach. The results show that the system can perform physical activity recognition with a successful rate over 80% for walking, sitting and standing. But climbing upstairs and downstairs have a really low successful rate %50 [5].

Murat Cihan Sorkun and colleagues' study analyzes the performance of activity classification when accelerometer and gyroscope are used separately or in combination. By using a dataset which is collected from fifteen participants with each has a three mobile phones in seperate positions. The performing activites are: standing, sitting, climbing, running, transportation, texting, calling and opening apps. Various features are extracted from raw data and afterwards supervised machine learning algorithms like NB, KNN, RF, MLP ve SVM are used to train and validate the results. With both sensors are in combination the most successful algorithm is RF. But walking and stairs activity has most successful rate with KNN [19].

Song-Mi Lee and colleagues aimed estimation of human recognition from smartphone's triaxial accelerometer data. Data was collected five graduate students' same smartphones in same brands (Nexus 6P). Then data was labelled three result states which are walking, running and staying. It was three axis of accelerometer data which has x, y, z axis accelerometer values. Magnitude evaluating with this equation. Data was collected with 2 method. Ten second and twenty second vectors. Ten second vectors are measured 10 second with one second interval then twenty second vectors are measured 20 second with one second interval. 7206 ten-second data and 6666 twenty second data with total of 13692 data. Performance of random forest method is %89 and CNN is %92.7 [7].

Other experiment again triaxial accelerometer was located on left wrist, right wrist and waist of a people. Sensors measure the 3-axis acceleration for standing for 12 minutes, lying 11 minutes, walking 15 minutes, sitting and lying 43 minutes and sitting and non-dining for 40 minutes. So, after that data was collected by the researcher and again Magnitude evaluating with Euclidean norm. After training with Random Forest, Decision Tree, and Support Vector Machine algorithm. Result was like multi sensor accuracy %81, dominant-hand accuracy %80 and non-dominant hand accuracy %73 with respect to all algorithms which they used. On the other hand, Dining algorithm performance at value of dining accuracy measured. Best performance was Decision Tree performance [8].

In other experiment researchers again collected triaxial accelerometer data for determining walking, running, snowboarding, skiing and driving. Accelerometer run with Wi-Fi Photon System on Board microcontroller. This microcontroller sends data to amazon dynomoDB in real time. Fifty thousand sample collected from 2 male 1 female people for each activity and four machine learning algorithms were implemented. Such that KNN, Naïve Bayes, Random Forest and Support Vector Machine. Best performance came from J48 algorithm [20].

The other study researchers used integrated sensors on smartphones. There are six labels in this data such as walking, standing, sitting, lying down, up the stairs and down the stairs. Support Vector Machine algorithm had best performance in this study. Model was worked with the accuracy of %89. Data was collected 3 types of sensors which located in smartphones: Acceleration sensor, gyro sensor and accelerometer sensor and each sensor return x, y and z dimension. Data was separated as %70 of training set and %30 of test set and was collected 10939 sample and. Learning completed in SVM with a %89 accuracy [9].

ZhenyuHe's study again 3-axis accelerometer used and raw data was transformed with Wavelet transform. Accelerometer was located on human body and data was sent to computer via Bluetooth. There are three type of label in data set, such as walking, jumping, still, running. They were use Wavelet transform which decomposed signal into detailed signal. After wavelet transform signal was processed with AR model. SVM algorithm used and Wavelet-AR based model learn with accuracy of %95 [10].
Researchers in other experiment used open access dataset from UCI named Mobile Health (mHealth). Dataset consist of accelerometer, gyroscope and magnetometer data. Data collected from smartphones and other wearable devices. There were 3 features in data which were Sensor name of sensor type, Location name of sensor's location on human body and Orientation name of x, y, z value of sensors. There were twenty-four types of labels which separated into two part as Activity and Transition. Activity labels were twelve label that contains normal activities such as running, cycling, walking etc. Transition labels were activity transitions which consist of stand to sit, sit to lie, walk to climb etc. The sensitivity improved %50 in this experiment that had transition label with j48 algorithm [11].

In other study researchers estimate the prayer activities in addition to basic human activities such as walking, running, sitting etc. Data was collected ten people which were consist of five female, three male and two juniors. Data set again

consist of triaxial accelerometer data. Muslim prayer positions standing, bowing, prostrating, sitting and other human activities were estimated by KNN, j48 and Naïve Bayes algorithms. Performance of each algorithm given below [12].

Other research, researchers first generate feature map for the training sample and identify feature images. Secondly, the training sample and identifying feature images extracted feature and calculated feature vector. Thirdly, word bag models of feature vectors for feature images are obtained. Fourthly, they use the visual words bag characteristics quantified by the bag of model to obtain visual feature histogram. Finally a classification algorithm is utilized for training and classification results are obtained. The Bow which is feature map algorithm by adding SVM classifier has accuracy rate lower than other methods which SVM, KNN, Naive Bayes, Tree, Discriminant in identifying walking is %85 but it has much higher accuracy rate in identifying falling and staggering are %100 and %96 [13].

Tahmina Zebin and colleagues aimed to show that signal sequences of accelerometers and gyroscopes can be processed by Deep Convolutional Neural Networks to automatically learn from the input the optimal features for the activity recognition task. They have collected data from 12 healthy volunteers who have five different sensor locations on the lower body in order to classify activities more accurately. The activities are walking, walking upstairs, downstairs, sitting, standing and lying down. The comparison of algorithms show that CNN has %97.01 accuracy, SVM has %96.4 accuracy, MLP has %91.7 accuracy [14].

The main aim of the other work which mentioned in the research is to compare the performance of different algorithms for human activity recognition by extracting various statistical the domain and frequency domain features from the inertial sensor(accelerometric and gyroscopic), improve classification accuracy by utilize machine learning algorithms and prediction speed of daily life activities. The daily life activities are walking, walking upstairs, downstairs, sitting, standing and lying down and then some algorithms used such as Decision Tree, Linear and Quadratic Discriminant Analysis, Support Vector Machine, Ensemble. The results demonstrate that Ensemble classifier has the highest accuracy rate which is %94.3 [15].

In other research researcher consider military operations, where the group members perform motions on a mission and their aim was to develop a real-time smartphone-based application to get soldier localization. The data-set was split into train-test-validation sets in the ratio of 50:30:20. They choose 6 activities which are walk, jog, run, jump, jump-rope and cycle. They used Decision Tree, Naive Bayes Classifier, K-Nearest Neighbor, Support Vector Machines and Random Forest classifiers. The classifiers tested with sensor(accelerometer, gyroscope, magnetometer) data and results showed that Random Forest is the best classifier with %97 precision, %97 recall, %97 f1-score and %9712 accuracy [16].

Luis F. Mejia and colleagues used accelerometer, gyroscope and pedometer sensors. They collected the sensor data using a smartphone in the left side pocket and the smartwatch strapped onto the left-hand wrist. In HAR, most works use supervised learning, but they purposed that unsupervised approach is to reveal patterns in data that could be used as activity primitives for further, higher level activity recognition. They extracted a total of 114 features from the time domain. Then they used 6 clustering algorithms are K-means, Spectral Clustering, Hierarchical clustering, DBSCAN, Mean Shift and Average Linkage. When they compared to their labeled dataset, the results of their clustering algorithms Spectral Clustering, K-means and Hierarchical Clustering using Ward's method appear to reinforce the use of their selected action primitives. The three algorithms show that they can use together samples from the lying down, standing idly and sitting down [17].

Other researchers used the algorithms on activity recognition mobile application which is named Activity Logger and they used embedded accelerometer sensors on smart phones. Four main activities; which are walking, running, standing and sitting, the application creates different training data files in which raw data from the 3-axes of the accelerometers is being logged. The Clustered KNN performance is around %92 accuracy considering all activities. The clustered KNN method exhibited a much better performance than the KNN classifier in terms of accuracy [18].

Burak Çatalbas and colleagues used the dataset which was created by 6 different activity which are walking, stair up, stair down, sitting, standing and lying down. The best classifier was Support Vector Machine. They proved the Neural Network algorithm can get better accuracy rate. In their tests the Neural Network algorithm which including reservoir and used two layers get the best accuracy rate which is %96.2. They used 330000 epoch to get the accuracy rate [21].

III. METHODOLOGY

1. Logistic Regression

Logistic regression is a statistical analysis method used to predict a data value based on prior observations of a data set. A logistic regression model predicts a dependent data variable by analyzing the relationship between one or more existing independent variables.

For example, a logistic regression could be used to predict whether a political candidate will win or lose an election or whether a high school student will be admitted to a particular college. The resulting analytical model can take into consideration multiple input criteria -- in the case of college acceptance, things such as the student's grade point average, SAT score and number of extracurricular activities. Based on historical data about earlier outcomes involving the same input criteria, it then scores new cases on their probability of falling into a particular outcome category.

Logistic regression has become an important tool in the discipline of machine learning. The approach allows an algorithm being used in a machine learning application to classify incoming data based on historical data. As more relevant data comes in, the algorithm should get better at predicting classifications within data sets. For that reason, logistic regression has become particularly popular in online advertising, enabling marketers to predict as a yes or no percentage the likelihood of specific website users who will click on particular advertisements.

Logistic regression can also play a role in data preparation activities by allowing data sets to be put into specifically predefined buckets during the extract, transform, load (ETL) process in order to stage the information for analysis.

2. Naïve Bayesian

The Naive Bayesian classifier is based on Bayes' theorem with the independence assumptions between predictors. A Naive Bayesian model is easy to build, with no complicated iterative parameter estimation which makes it particularly useful for very large datasets. Despite its simplicity, the Naive Bayesian classifier often does surprisingly well and is widely used because it often outperforms more sophisticated classification methods.

Bayes theorem provides a way of calculating the posterior probability, $P(c|x)$, from $P(c)$, $P(x)$, and $P(x|c)$. Naive Bayes classifier assume that the effect of the value of a predictor (x) on a given class (c) is independent of the values of other predictors. This assumption is called class conditional independence.

$$P(c \mid x) = \frac{P(x \mid c)P(c)}{P(x)}$$

- $P(c|x)$ is the posterior probability of *class* (*target*) given *predictor* (*attribute*).
- $P(c)$ is the prior probability of *class*.
- $P(x|c)$ is the likelihood which is the probability of *predictor* given *class*.
- $P(x)$ is the prior probability of *predictor*.

3. Knn (k-Nearest Neighbors)

The model for kNN is the entire training dataset. When a prediction is required for a unseen data instance, the kNN algorithm will search through the training dataset for the k-most similar instances. The prediction attribute of the most similar instances is summarized and returned as the prediction for the unseen instance.

The similarity measure is dependent on the type of data. For real-valued data, the Euclidean distance can be used. Other other types of data such as categorical or binary data, Hamming distance can be used.

In the case of regression problems, the average of the predicted attribute may be returned. In the case of classification, the most prevalent class may be returned.

The kNN algorithm is belongs to the family of instance-based, competitive learning and lazy learning algorithms.

Instance-based algorithms are those algorithms that model the problem using data instances (or rows) in order to make predictive decisions. The kNN algorithm is an extreme form of instance-based methods because all training observations are retained as part of the model.

It is a competitive learning algorithm, because it internally uses competition between model elements (data instances) in order to make a predictive decision. The objective similarity measure between data instances causes each data instance to compete to "win" or be most similar to a given unseen data instance and contribute to a prediction.

Lazy learning refers to the fact that the algorithm does not build a model until the time that a prediction is required. It is lazy because it only does work at the last second. This has the benefit of only including data relevant to the unseen data, called a localized model. A disadvantage is that it can be computationally expensive to repeat the same or similar searches over larger training datasets.

Finally, kNN is powerful because it does not assume anything about the data, other than a distance measure can be calculated consistently between any two instances. As such, it is called non-parametric or non-linear as it does not assume a functional form.

4. Decision Tree

A decision tree is a tree like collection of nodes intended to create a decision on values affiliation to a class or an estimate of a numerical target value. Each node represents a splitting rule for one specific Attribute. For classification this rule separates values belonging to different classes, for regression it separates them in order to reduce the error in an optimal way for the selected parameter *criterion*.

The building of new nodes is repeated until the stopping criteria are met. A prediction for the class label Attribute is determined depending on the majority of Examples which reached this leaf during generation, while an estimation for a numerical value is obtained by averaging the values in a leaf.

This Operator can process ExampleSets containing both nominal and numerical Attributes. The label Attribute must be nominal for classification and numerical for regression.

After generation, the decision tree model can be applied to new Examples using the Apply Model Operator. Each Example follows the branches of the tree in accordance to the splitting rule until a leaf is reached.

5. Artificial Neural Network

The field of Artificial Neural Networks (ANN) is concerned with the investigation of computational models inspired by theories and observation of the structure and function of biological networks of neural cells in the brain. They are generally designed as models for addressing mathematical, computational, and engineering problems. As such, there is a lot of interdisciplinary research in mathematics, neurobiology and computer science.

An Artificial Neural Network is generally comprised of a collection of artificial neurons that are interconnected in order to performs some computation on input patterns and create output patterns. They are adaptive systems capable of modifying their internal structure, typically the weights between nodes in the network, allowing them to be used for a variety of function approximation problems such as classification, regression, feature extraction and content addressable memory.

Given that the focus of the field is on performing computation with networks of discrete computing units, the field is traditionally called a 'connectionist' paradigm of Artificial Intelligence and 'Neural Computation'.

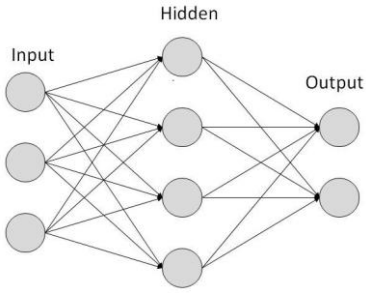

Figure 3.1. Artificial Neural Network illustration

There are many types of neural networks, many of which fall into one of two categories:

- **Feed-forward Networks** where input is provided on one side of the network and the signals are propagated forward (in one direction) through the network structure to the other side where output signals are read. These networks may be comprised of one cell, one layer or multiple layers of neurons. Some examples include the Perceptron, Radial Basis Function Networks, and the multi-layer perceptron networks.

- **Recurrent Networks** where cycles in the network are permitted and the structure may be fully interconnected. Examples include the Hopfield Network and Bidirectional Associative Memory.

Artificial Neural Network structures are made up of nodes and weights which typically require training based on samples of patterns from a problem domain. Some examples of learning strategies include:

- **Supervised Learning** where the network is exposed to the input that has a known expected answer. The internal state of the network is modified to better match the expected result. Examples of this learning method include the Back-propagation algorithm and the Hebb rule.

- **Unsupervised Learning** where the network is exposed to input patterns from which it must discern meaning and extract features. The most common type of unsupervised learning is competitive learning where neurons compete based on the input pattern to produce an output pattern. Examples include Neural Gas, Learning Vector Quantization, and the Self-Organizing Map.

Artificial Neural Networks are typically difficult to configure and slow to train, but once prepared are very fast in application. They are generally used for function approximation-based problem domains and prized for their capabilities of generalization and tolerance to noise. They are known to have the limitation of being opaque, meaning there is little explanation to the subject matter expert as to why decisions were made, only how.

6. Random Forest

A Random Forest consists of a collection or ensemble of simple tree predictors, each capable of producing a response when presented with a set of predictor values. For classification problems, this response takes the form of a class membership, which associates, or classifies, a set of independent predictor values with one of the categories present in the dependent variable. Alternatively, for regression problems, the tree response is an estimate of the dependent variable given the predictors. The Random Forest algorithm was developed by Breiman.

A Random Forest consists of an arbitrary number of simple trees, which are used to determine the final outcome. For classification problems, the ensemble of simple trees vote for the most popular class. In the regression problem, their responses are averaged to obtain an estimate of the dependent variable. Using tree ensembles can lead to significant improvement in prediction accuracy (i.e., better ability to predict new data cases).

The response of each tree depends on a set of predictor values chosen independently (with replacement) and with the same distribution for all trees in the forest, which is a subset of the predictor values of the original data set. The optimal size of the subset of predictor variables is given by log2 M+1, where M is the number of inputs.

For classification problems, given a set of simple trees and a set of random predictor variables, the Random Forest method defines a margin function that measures the extent to which the average number of votes for the correct class exceeds the average vote for any other class present in the dependent variable. This measure provides us not only with a convenient way of making predictions, but also with a way of associating a confidence measure with those predictions.

For regression problems, Random Forests are formed by growing simple trees, each capable of producing a numerical response value. Here, too, the predictor set is randomly selected from the same distribution and for all trees. Given the above, the mean-square error for a Random Forest is given by:

mean error = (observed - tree response)^2

The predictions of the Random Forest are taken to be the average of the predictions of the trees:

$$\text{Random Forest Prediction } s = \frac{1}{K}\sum_{K=1}^{K} K^{th} \text{ tree response}$$

where the index k runs over the individual trees in the forest.

Typically, Random Forests can flexibly incorporate missing data in the predictor variables. When missing data are encountered for a particular observation (case) during model building, the prediction made for that case is based on the last preceding (non-terminal) node in the respective tree. So, for example, if at a particular point in the sequence of trees a predictor variable is selected at the root (or other non-terminal) node for which some cases have no valid data, then the prediction for those cases is simply based on the overall mean at the root (or other non-terminal) node. Hence, there is no need to eliminate cases from the analysis if they have missing data for some of the predictors, nor is it necessary to compute surrogate split statistics.

IV. DATASET

The dataset is based on Human Activity Recognition (HHAR) from Smartphones and Smartwatches from the public repository [22].
The dataset contains the readings of two motion sensors commonly found in smartphones recorded while users executed activities scripted in no specific order carrying smartwatches and smartphones.
The data is split into 4 files in total divided by device (phone or watch) and sensor (gyroscope and accelerometer). The files for phones are: Phones_accelerometer.csv, Phones_gyroscope.csv for the accelerometer and gyroscope respectively, and for the Watch_accelerometer.csv, Watch_gyroscope.csv for the accelerometer and gyroscope as well.
Activities are ‘Biking’, ‘Sitting’, ‘Standing’, ‘Walking’, ‘Stair Up’ and ‘Stair down’.
Sensors are two embedded sensors, i.e., Accelerometer and Gyroscope sampled at the highest frequency possible by the device
Devices are 4 smartwatches (2 LG watches, 2 Samsung Galaxy Gears)
8 smartphones (2 Samsung Galaxy S3 mini, 2 Samsung Galaxy S3, 2 LG Nexus 4, 2 Samsung Galaxy S+)
Recordings are 9 users currently named: a, b, c, d, e, f, g, h, i, consistently across all files.
The data set is structured in the following way:
All the csv files have the same structure of following columns are 'Index', 'Arrival_Time', 'Creation_Time', 'x', 'y', 'z', 'User', 'Model', 'Device', 'gt'. Description of all features were given in Table 4.1.

Feature	Description
Index	is the row number.
Arrival Time	The time the measurement arrived to the sensing application
Creation Time	The timestamp the OS attaches to the sample.
x, y, z	The values provided by the sensor for the three axes, x, y, z.
User	The user this sample originates from, the users are named a to i.
Model	The phone/watch model this sample originates from
Device	The specific device this sample is from. They are prefixed with the model name and then the number, e.g., nexus4_1 or nexus4_2.

Table 4.1. Description of features.

The data was labeled from user activities i.e. bike, sit, stand, walk, stairs up, stairs down and null.
Each accelerometer sample if represented as a single row in the file and with all columns having repeated values.
Also due to issues with sampling some users have few collected samples for specific activities, e.g., User h and activity sit in the Phones_accelerometer.csv.
The null class is defined as null in the gt (groundtruth) column, whereas the rest of the classes are called bike sit, stand, walk, stairs up, stairs down.
The names and models of the devices used in the HAR data set are LG-Nexus 4 with model of 'nexus4_1' and 'nexus4_2', Samsung Galaxy S3 with model of 's3_1' and 's3_2’, Samsung Galaxy S3 min with model of 's3mini_1' and 's3mini_2' and Samsung Galaxy S+ with model of 'samsungold_1' and 'samsungold_2'.
Due to the large size and complexity of the data set and also the lack of hardware of the computers, only the phone gyroscope values will be used in this study.

V. PREPROCESSING

In the most basic form of the data processing and evaluation of the results were made. The results were transferred in this section of the article. First, the preprocessing stages of the three-dimensional gyroscope data from mobile phones were applied. The reason for this is to increase the accuracy rate of the machine learning algorithms by making the data more quality.

Only index numbers were deleted on the data. Because the index number is a unique value defined for each instance. If we do not delete this column from our data, it will adversely affect the accuracy rates we will achieve in our studies. The dependent and independent variables were seperated. Our independent variables are 'Arrival_Time', 'Creation_Time', 'x', 'y', 'z', 'User', 'Model', 'Device'. Our dependent variable is 'gt' which contains 'Biking', 'Sitting', 'Standing', 'Walking', 'Stair Up' and 'Stair down' activities.

Then, for the discrete datas which are 'User', 'Model' and 'Device' label encoding process applied. Label Encoding refers to converting the labels into numeric form so as to convert it into the machine-readable form. Machine learning algorithms can then decide in a better way on how those labels must be operated. It is an important pre-processing step for the structured dataset in supervised learning.

Then for the datas which label encoding process applied, one hot encoding process applied. One hot encoding is a representation of categorical variables as binary vectors. This first requires that the categorical values be mapped to integer values. Then, each integer value is represented as a binary vector that is all zero values except the index of the integer, which is marked with a 1.

The encoded data was given the title of each feature. After that, for the datas imputer process applied.

Most frequent imputation replaces missing values with the most frequent value of that variables of feature.

We applied Min-Max scaling for each dependent and independent values because we tried Standart Scaling and other scaling methods and the best results were given by Min-Max scaling. In Min-Max scaling, the data is scaled to a fixed range - usually 0 to 1. The cost of having this bounded range - in contrast to standardization - is that we will end up with smaller standard deviations, which can suppress the effect of outliers. A Min-Max scaling is typically done via the following equation:

$$X_{sc} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

In this study, the correlation of the features analyzed. This process was analyzed by 2 methods. The first one is the correlation matrix and the other is the feature importance of random forest algorithm.

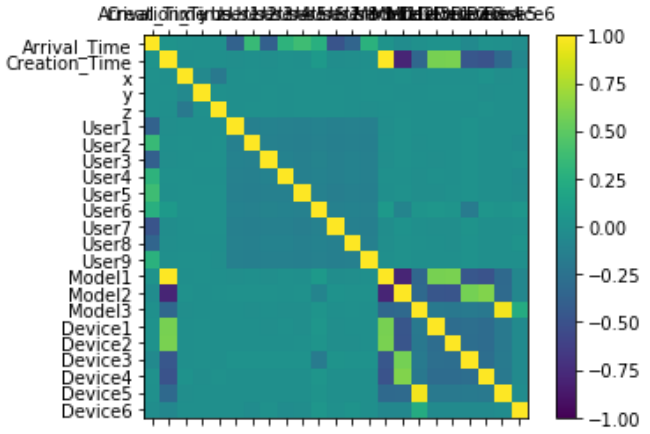

Figure 5.1. Correlation matrix

Correlation matrix results were analyzed in Figure 5.1. which shows the correlation of features. After the analysis, it was predicted to use arrival time, creation time and x, y, z features. But in order to consolidate this prediction, the results of feature importance were observed with random forest.

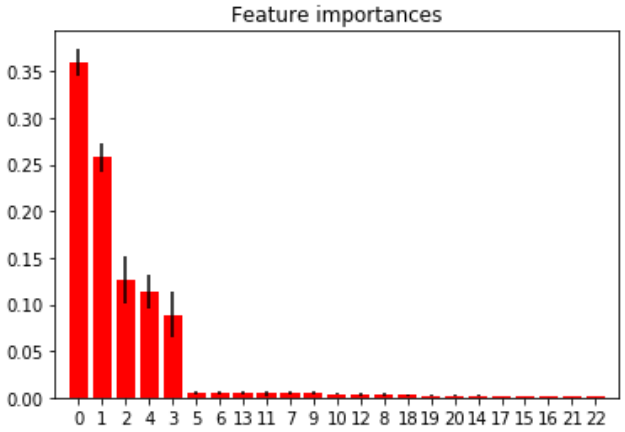

Figure 5.2. Feature importance

As shown in Figure 5.2., the highest importance values are foreseen as arrival time, creation time and x, y, z. Following these results, model name, device name and user name features were dropped.

walk	2350429
sit	2218501
stand	2024206
bike	1911730
stairsup	1884306
stairsdown	1673833

Figure 5.3. Sample number of labels

Then, the balanced and imbalanced status of the data was observed. When the number of labels is displayed, the distributions of the labels are obtained as shown in Figure 5.3. The data were considered to be balanced because the number of labels in this distribution were close to each other.

Then for the datas cross validation process applied, 67 percent of train and 33 percent of the test set rate of train-test was made because we wanted to seperate our train and test data in a balanced way. Cross validation is a model evaluation method that is better than residuals. The problem with residual evaluations is that they do not give an indication of how well the learner will do when it is asked to make new predictions for data it has not already seen. One way to overcome this problem is to not use the entire data set when training a learner. Some of the data is removed before training begins. Then when training is done, the data that was removed can be used to test the performance of the learned model on ``new'' data. This is the basic idea for a whole class of model evaluation methods called cross validation. Thus, the data became runnable.

VI. RESULTS

After the preparation of the data, the model was first trained with logistic regression algorithm. The hyperparameters of the logistic regression model are “random_state = 1234”, “class_weight = ’balanced’”. Accuracy, recall and precision values were obtained from the test data and validation. When the obtained results were evaluated, accuracy was 16 percent and precision was 14 percent for both test and validation parts. It is concluded that logistic regression algorithm is not suitable for raw data. The same results were observed in naive bayes algorithm with same accuracy, precision and recall particularly. Hyperparameters are not used on the naive bayes model.

After logistic regression and naive bayes algorithms have failed badly, the model has been trained by using k-nearest neighbors algorithm which is another classification algorithm. The hyperparameters of the knn model are “n_neighbors = 120”, “metric='minkowski'”. The hyperparameters were selected with trial and error. Validation accuracy value was 93 percent and precision value was taken as 88 percent. In contrast, the accuracy of the test set was 87 percent and the precision value was 77 percent. Although the success of the model was high with these values, it was observed that it tended to overfit.

The study continued with the decision tree algorithm. The results obtained were more successful than other algorithms. The hyperparameters of the decision tree model are “random_state = 1234”, “criterion = 'gini'”, “splitter='best'”, “class_weight ='balanced'”. The hyperparameters were selected with trial and error. Validation accuracy and precision values were taken as 99 percent. In contrast, the accuracy of the test set was 95 percent and the precision was 91 percent.

Random forest algorithm has been tested since it is considered that this success will be likely to increase with the decision tree algorithm performs high. The hyperparameters of the random forest model are “random_state = 1234”, “criterion = 'gini'”, “n_estimators = 22”, “class_weight ='balanced'”. The hyperparameters were selected with trial and error. According to the result, the validation accuracy and precision value was 99 percent and the accuracy of the test set was 96 percent and the precision value was 93 percent. As expected, better results were obtained than the decision tree with the random forest algorithm.

In the five classification algorithms the best results were given with random forest algorithm. Values of all algorithms; accuracy, precision and recall values of the test set given in table 6.1; validation set accuracy, precision and recall values are given in table 6.2.

Classification Algorithm	**Test Accuracy**	**Test Recall**	**Test Precision**
Logistic Regression	0.16	[0. 0. 0. 0. 0. 0. 1.]	0.14
Naive Bayes	0.16	[0. 0. 0. 0. 0. 0. 1.]	0.14
KNN	0.87	[0.82 0.89 0.90 0.87 0.87 0.83 0.88]	0.77
Decision Tree	0.95	[0.99 0.93 0.98 0.88 0.90 0.98 0.97]	0.91
Random Forest	0.96	[0.99 0.93 0.98 0.92 0.91 0.99 0.98]	0.93

Table 6.1. Results of test set

Classification Algorithm	**Validation Accuracy**	**Validation Recall**	**Validation Precision**
Logistic Regression	0.16	[0. 0. 0. 0. 0. 0. 1.]	0.14
Naive Bayes	0.16	[0. 0. 0. 0. 0. 0. 1.]	0.14
KNN	0.93	[0.91 0.94 0.95 0.93 0.93 0.91 0.94]	0.88
Decision Tree	0.99	[0.99 0.99 0.99 1. 0.99 0.99 1.]	0.99
Random Forest	0.99	[0.99 0.99 0.99 0.99 0.99 0.99 0.99]	0.99

Table 6.2. Results of validation set

In addition, after the model has been trained with classical classification algorithms, the result has been observed with an artificial neural network, which has recently increased in popularity and gave very successful results. After the feature one hot encoding process, 23 features have been obtained and the input layer of the model was determined as 23 neurons. Because of the large number of samples in our data set, the complexity of the artificial neural network model

should be high. However, due to technological deficiencies, the complexity of the model was not very high. In addition, hidden layer has 4 layers and first hidden layer has 1024 neurons, second hidden layer has 512 neurons, third hidden layer has 512 neurons, last hidden layer has 256 neurons. It has been proven that the Relu function as an activation function is more successful than other activation functions [23]. In this study, Relu function is used as activation function. And also we tried Sigmoid activation function too but the Relu activation function is increased accuracy of the models we tried. Finally, 6 neurons were identified, which were indexed to the output layer label number. And for optimizer we used Adam optimizer because with the trail and error it was given the best results. And also we used categorical cross entropy. Because we have 6 class. That's why the function of the output layer was softmax. Softmax gives a value between 0 and 1 for all classes. And it use argmax for select the highest value in the each given values of classes. As a result of the model training with these values, the test accuracy was 87%, the loss value was 0.31 and the train accuracy was 88%, the loss value was 0.28.

VII. CONCLUSION

In this study, the data from the mobile phone gyroscope sensor was used to estimate the activity of a person. Before this process was started, the data were preprocessed. At these stages, the index numbers were deleted, followed by the label and one hot encoding. Then the missing datas were completed by using imputer. We performed these operations, then we performed min-max scaling. Then we reduced the size of data set by feature selection and cross-validation process. Their performances were evaluated one by one using different classification algorithms. In addition, the current method has been trained with artificial neural networks and the results were observed. When we examine the results, we have determined that Random Forest algorithm is the most successful algorithm for this data set since the train accuracy ratio is %99 and the test accuracy ratio is %96. As a result of the study, the situation of a person was estimated by taking datas from the gyroscope sensor which is on the mobile phones used by everyone nowadays. This model can be used in future technologies about healthcare and can contribute to the industry 4.0.

VIII. REFERENCES

[1] S.Bhagya Rekha, M. Venkateswara Rao, "Methodical Activity Recognition and Monitoring of a Person through Smart Phone and Wireless Sensors," in *IEEEE Internatioanal Conference, Signals and Instrumentation Engineering(ICPCSI)*, 2017.

[2] Mohammed Mobark, Suriayati Chuprat, Teddy Mantoro, "Improving the Accuracy of Complex Activities Recognition Using Accelerometer-Embedded Mobile Phone Classifiers".

[3] Dengpan Tian and Xiaowei Xu, Ye Tao, Xiaodong Wang, "An Improved Activity Recognition Method Base," in *IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)*, 2017.

[4] Lili Xie, Jun Tian, Genming Ding, Qian Zhao, "Human Activity Recognition Method based on Inertial Sensor and Barometer".

[5] Hui Huang, Xian Li,Ye Sun, "A Triboelectric Motion Sensor in Wearable Body Sensor Network for Human Activity Recognition," *IEEE.*

[6] Sebastian Scheurer,Salvatore Tedesco, Kenneth N. Brown,Brendan O'Flynn, "Sensor and Feature Selection for an Emergency First Responders Activity Recognition System," *IEEE.*

[7] Song-Mi Lee, Sang Min Yoon, Heeryon Cho, "Human Activity Recognition From Accelerometer Data Using Convolutional Neural Network," *IEEE,* 2017.

[8] Yu-Chuan Huang, Chih-Wei Yi, Wen-Chih Peng, "A Study on Multiple Wearable Sensors for Activity Recognition," *IEEE,* 2017.

[9] Duc Ngoc Tran, Duy Dinh Phan, "Human Activities Recognition in Android Smartphone Using Support Vector Machine," in *7th International Conference on Intelligent Systems, Modelling and Simulation*, 2016.

[10] ZhenyuHe, "Activity Recognition from Accelerometer Signals Based on Wavelet-AR Model," *IEEE,* 2010.

[11] Saleha Khatun, Bashir I. Morshed, "Fully-Automated Human Activity Recognition with Transition Awareness from Wearable Sensor Data for mHealth," *IEEE,* 2018.

[12] Walid Mohamed Aly, Khaled Eskaf, Alyaa Aly, "Aggregated Activity Recognition Using Smart Devices," *IEEE,* 2016.

[13] Guangyu He, Xinze Luan, Junyi Wang, Xiaoting Wang, "A novel feature map for human activity recognition," in *Second International Conference on Mechanical, Control and Computer Engineering*, 2017.

[14] Tahmina Zebin, Patricia J Scully, Krikor B. Ozanyan, "Human Activity Recognition with Inertial Sensors using a Deep Learning Approach".

[15] Tahmina Zebin, Patricia J. Scully, Krikor B. Ozanyan, "Evaluation of Supervised Classification Algorithms for Human Activity Recognition with Inertial Sensors," *IEEE,* 2017.

[16] A. Mukherjee, Sudip Misra, Muttukrishnan Rajarajan, Yogachandran Rahulamathavan, "SmartARM: A Smartphone-based Group Activity Recognition and Monitoring Scheme for Military Applications".

[17] Luis F. Mejia-Ricart, Paul Helling, Aspen Olmsted, "Evaluate," in *The 12th International Conference for Internet Technology and Secured Transactions (ICITST)*, 2017.

[18] Pinky Paul, Thomas George, "An Effective Approach for Human Activity Recognition on Smartphone," in *IEEE International Conference on Engineering and Technology (ICETECH)* , Coimbatore, TN, India, 2015.

[19] Murat Cihan Sorkun, Ahmet Emre Danışman, Özlem Durmaz İncel, "AKILLI TELEFON ALGILAYICILARI İLE AKTİVİTE TANIMA: ALGILAYICILARIN VE PENCERE BOYUNUN ETKİSİ HUMAN ACTIVITY RECOGNITION WITH MOBILE PHONE SENSORS: IMPACT OF SENSORS AND WINDOW SIZE," *IEEE,* 2018.

[20] Tuba KAYTARAN, Levent BAYINDIR, "İçinde Photon Geliştirme Kartı ve İvmeölçer Bulunan Bileklik Kullanarak Aktivite Tanıma Activity Recognition with Wrist found in Photon Development Board and Accelerometer," *IEEE,* 2018.

[21] Burak Çatalbas¸, Bahadır Çatalbas¸, Ömer Morgül, "Farklı Yapay Sinir Ag˘ı Temelli Sınıflandırıcılar ile İnsan Hareketi Tanımlama Human Activity Recognition with Different Artificial Neural Network Based Classifiers," *IEEE,* 2017.

[22] "https://archive.ics.uci.edu/ml/datasets/Heterogeneity+Activity+Recognition," [Online].

[23] N. A. Cenk Bircanoglu, “A Comparison of Activation Functions in Artificial”, *IEEE,* 2018.