

# A Machine Learning Framework for Edge Computing to Improve Prediction Accuracy in Mobile Health Monitoring

Sigdel Shree Ram[1(✉)], Bernady Apduhan[1(✉)], and Norio Shiratori[2(✉)]

[1] Graduate School of Information Science, Kyushu Sangyo University, Fukuoka 813-8503, Japan
kl8gjk0l@st.kyusan-u.ac.jp, bob@is.kyusan-u.ac.jp
[2] Research and Development Initiative, Chuo University, Tokyo 112-8551, Japan
norio@riec.tohoku.ac.jp

**Abstract.** The great challenges in the aging society and the lack of human resources, especially in health care, remains a formidable task. The cloud centric computing paradigm offers a solution in processing IoT applications in health care. However, due to the large computing and communication overheads, an alternative solution is sought. Here, we consider machine learning in edge computing to detect and improve the predictability accuracy in mobile health monitoring of human activity. With multi-modal sensor data, we conducted pre-processing to sanitize the data and classify the activities in the dataset. We used and compare the processing performance using random forests and SVM machine learning algorithms to identify and classify the activities in the dataset. We achieved approximately 99% accuracy with random forest which was better than SVM, at 98%. We used confusion matrix to identify the majority of mis-matched data belonging to initial value of sensors while recording a particular activity, and also used visual representation of the data for better understanding. We extract the activity's ECG data and classify into four categories to provide more specific information from the person's activity data. The aforementioned experiments provided promising results and insights on the implementation to improve the prediction accuracy on the health status of people undergoing some activity.

**Keywords:** Edge computing/intelligence · Mobile health · Machine learning

## 1 Introduction

The popularity of IoT (Internet of Things) applications is gaining much interests in various fields in industry, manufacturing, agriculture, medical, and other sectors of society. The deployment of IoT devices (sensors, actuators, etc.) to detect and acquire data will generate a large volume of data which creates great challenges on how to store and process this large amount of collected data, as well as how to unleash and harness the wealth of knowledge hidden in the data.

© Springer Nature Switzerland AG 2019
S. Misra et al. (Eds.): ICCSA 2019, LNCS 11621, pp. 417–431, 2019.
https://doi.org/10.1007/978-3-030-24302-9_30

The data center in a cloud computing environment offers a cost-effective means of storing and processing the huge data collected by IoT devices. However, the large communication overhead incurred to transfer data from the IoT devices to/from the data center has raised crucial issues especially for applications which requires near-real time responses. Specifically, the problems of cloud centric computing, i.e., narrow bandwidth, long latency, intermittent connection, delay in decision making, etc., paves the way of edge computing as a new revolutionary way of correcting the aforementioned deficiencies. Edge computing is not a replacement of cloud computing, but rather as a supplement. That is, much of the processing maybe be done at the edge nodes which are in close proximity to the edge devices or sources. However, whenever the processing capacity of edge nodes is not sufficient at certain times, some part(s) of the application processing can be offloaded to the data center. The short data transfer time between the source (IoT devices) and edge node, can be considered as part of the solution to IoT applications requiring near-real time responses. However, the limited processing and storage capacities of edge nodes are seen as another hindrances to the processing requirements.

In this paper, considering the above-mentioned processing and storage limitations of edge nodes, we study the use and performance of machine learning techniques to augment the data classification accuracy, in an effort to establish our platform to develop a semantic-based mobile health monitoring system.

The rest of the paper is organized as follows. Section 2 described our research objectives, and Sect. 3 cites some related researches. In Sect. 4, we described our experiment environment and the datasets, whereas Sect. 5 described our system architecture. The experiment methods are explained in Sects. 6 and 7 describe the experiment results and observations. Section 8 describe the ECG classification prediction, whereas Sect. 9 discuss the activity data classification and ECG data categorization. Section 10 gives the summary and concluding remarks of the study. Last but not the least, Sect. 11 cites some of our future work.

## 2 Research Objective

The first objective of this research is to study and identify the most viable supervised machine learning algorithm [7] (Random Forests (RF), Support Vector Machines (SVM)) which will suit to our proposed semantic-based mobile health monitoring system. We will use datasets and conduct experiments to quantify the data classification accuracy of the activity sets using RF and SVM algorithms. Confusion matrix and graphs will be used for visual analysis of the results. To determine more specific conditions of the subject (patient), we extract the ECG values of the activity sets and categorize it accordingly. These experiments will provide us better understanding of machine learning algorithms and to gain insights to realize intelligent analytics in edge computing for personalized mobile health monitoring system. After comparing and contrasting between experiments expected outcome was positive.

## 3 Related Research

In Mahmut Taha Yazici et al. [1], their research used Random Forest, Support Vector Machine (SVM) and Multi-Layer Perceptron for testing data sets on Raspberry pi. They have compared accuracy, processing capacity, power consumption between three algorithms in Raspberry pi. They have concluded that when the size of dataset is small, SVM is slightly faster than Random Forest; but when the size of data is large, Random Forest was faster with higher accuracy. While we share similar objectives, our tests were conducted on a PC which provides more processing power and generate our envisioned model.

Whereas, Calvier et al. [2], proposed a method to bridge existing knowledge models with ad hoc taxonomies to address the problem of textual documents classification. This method allows the expert user to match their needs by optimizing text document classification. This technique is used on web based textual documents. In contrast, in our study, we have implemented the basic concept of semantization to improve the data pre-processing and in identifying the critical cases.

Furthermore, in Andreotti, Carr et al. [8], has classified the short segments of ECG into 4 groups using convolutional neural networks. The objective of this research was to detect arrhythmia. We used similar dataset in identifying the ECG status and comparing ECG status with the activity classifier's result.

## 4 Experiment Environment and Dataset

In our experiments, we used a Mac PC (macOS High Sierra version 10.13.6, 3.1 GHz Intel Core i5 CPU, and 8 GB RAM), Python 3.7 programming language and Anaconda navigator as the programming platform with Spyder 3.3.1 IDE.

### 4.1 Dataset

In this research two datasets are used. One dataset [10] consist of data of ECG value with four different classifications which is further more explained in Sect. 8 and another dataset [4–6] consist of data generated from multi-modal sensors placed on the subject's chest, right wrist, and left ankle. These three sensors record the body movement and turns of different human activity. In this experiment, 1,200,000 raw data were filtered down to 350,000 by removing null labeled data so that there will be less confusion while training the model. The activity and attribute are displayed in Tables 1 and 2, respectively.

**Table 1.** Activity labels and data size in terms of time and intervals

| S.N. | Activity | S.N. | Activity |
|---|---|---|---|
| 1 | Standing still (1 min) | 7 | Frontal elevation of arms (20x) |
| 2 | Sitting and relaxing (1 min) | 8 | Knees bending (crouching) (20x) |
| 3 | Lying down (1 min) | 9 | Cycling (1 min) |
| 4 | Walking (1 min) | 10 | Jogging (1 min) |
| 5 | Climbing stairs (1 min) | 11 | Running (1 min) |
| 6 | Waist bends forward (20x) | 12 | Jump front & back (20x) |

*In brackets are the number of repetitions (Nx) or the duration of the activity (min)

**Table 2.** Attribute information which was recorded from 3 sensors.

| Column | Attribute | Positional notation |
|---|---|---|
| 1 | Acceleration from the chest sensor | X axis |
| 2 | Acceleration from the chest sensor | Y axis |
| 3 | Acceleration from the chest sensor | Z axis |
| 4 | Electrocardiogram signal | Lead I |
| 5 | Electrocardiogram signal | Lead II |
| 6 | Acceleration from the left-ankle sensor | X axis |
| 7 | Acceleration from the left-ankle sensor | Y axis |
| 8 | Acceleration from the left-ankle sensor | Z axis |
| 9 | Gyro from the left-ankle sensor | X axis |
| 10 | Gyro from the left-ankle sensor | Y axis |
| 11 | Gyro from the left-ankle sensor | Z axis |
| 12 | Magnetometer from the left-ankle sensor | X axis |
| 13 | Magnetometer from the left-ankle sensor | Y axis |
| 14 | Magnetometer from the left-ankle sensor | Z axis |
| 15 | Acceleration from the right-lower arm sensor | X axis |
| 16 | Acceleration from the right-lower arm sensor | Y axis |
| 17 | Acceleration from the right-lower arm sensor | Z axis |
| 18 | Gyro from the right-lower-arm sensor | X axis |
| 19 | Gyro from the right-lower-arm sensor | Y axis |
| 20 | Gyro from the right-lower-arm sensor | Z axis |
| 21 | Magnetometer from the right-lower-arm sensor | X axis |
| 22 | Magnetometer from the right-lower-arm sensor | Y axis |
| 23 | Magnetometer from the right-lower-arm sensor | Z axis |
| 24 | Classification label | Activity |

*Units: Acceleration ($m/s^2$), gyroscope (deg/s), magnetic field (local), ecg (mV)

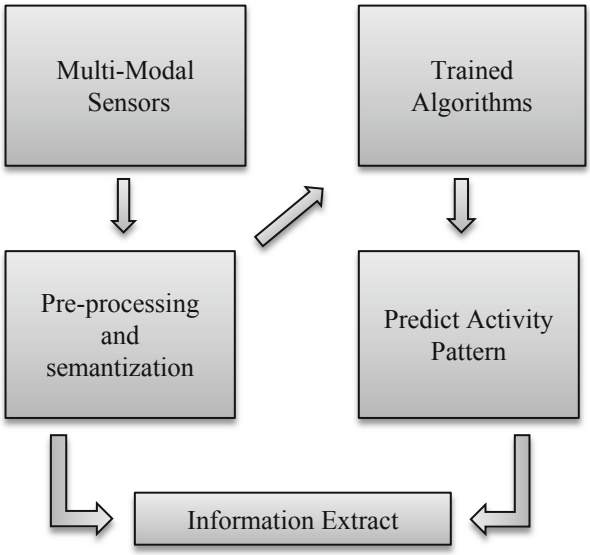


**Fig. 1.** Block diagram of a personalized health monitoring system

## 5 System Architecture

Our proposed personalized health monitoring system consist of multi-modal sensors to record the human body activities and vital signs of the body, i.e. Electrocardiogram values, motions, etc., as shown in Fig. 1. To outline the scope of semantization, we conducted preliminary experiments to find out the critical values. By extracting such information, we can be more accurate to predict as well as to react immediately on critical situations. The sensors data is pre-processed and the output data is passed to the trained models to predict the activity. Furthermore, after the prediction, semantization can be applied to augment the prediction accuracy of the subject's activity.

**Multi-modal Sensors**
Multi- modal sensors are such sensors which can record multiple instances at a same time. The data used in this experiments are recorded from 3 sensors which has 23 attributes.

**Pre-processing and Semantization**
Those data are pre-processed to maximize the prediction accuracy and narrow down the data composition. The detailed pre-processing is mentioned in Sect. 6.1. Semantization of data like labeling or extracting information took place. In Sect. 9, the basic concept of semantization (extracting information/giving meaning) is explained.

**Trained Algorithms**
Generally, in this experiment the choice of algorithms has been narrow down to two algorithms, i.e., the support vector machine and random forest after reviewing some related articles and considering the platform, i.e. edge computing.

Random forest (RF) algorithm was used because of its simplicity and high accuracy. RF creates random forest from assembly of decision trees. And the classification is achieved by having most votes overall in the forest.

Whereas, Support Vector Machine (SVM) looks at the extreme value of the datasets and creates the decision line, known as hyperplane. Although this algorithm takes a bit more time during the training process when the training dataset is huge. But one of the major advantage of SVM is it provides higher precision and very good with large number of features. These both algorithms were used to train and test the experiments.

**Predict Activity and Information Extraction**

The model predicts the activity according the algorithm used and the nature of the input data. Those output were compared and contrast between two models output which provides a generic level of semantization. This flow of data and output is explained with experiment in Sects. 8 and 9.

## 6 Experiment Methods

### 6.1 Data Pre-processing

To make the input data easier to work with machine learning algorithms, data pre-processing was conducted, as follows:

- **Importing Libraries and datasets**—Also includes adding labels and additional information as well as finding out the boundary of the datasets and removing the unnecessary data which leads to confusion while training the model. (Null class data were removed)
- **Missing data**—Although the missing value were very few in numbers, to get a better quality output, mean value calculation was used. Missing value was replaced by calculating the mean of the existing observation for that variable.
- **Splitting datasets**—We have implemented a 70:30, i.e., 70% of datasets is used for training purposes and the rest of the data for testing purposes.
- **Scaling data**—Dataset consists of different kinds of data with huge differences. So to optimize, we need to do the scaling of the data. Here, we used standardization to scale the data which converts the different variety of data into the same type considering the same basic features for SVM whereas for random forest scaling was not necessary because of the tree based features.

### 6.2 Training and Testing

After pre-processing the data, we train and test the algorithms one at time. In the experiments, we used Support Vector Machines and Random Forest supervised machine learning algorithms.

### 6.3 Data Analysis

During the analysis process, we concentrate on the accuracy check with respect to the processing speed. Accuracy check between real output of the testing data and output of predicted data. Furthermore, to understand the concentration of the predicted data and real data we have used confusion matrix.

## 7 Experiments and Observations

Experiments were conducted to understand the implementation of machine learning intelligence on edge devices while considering the data classification accuracy as the major measuring component. We calculate and compare the outcomes of RF and SVM ML-algorithms, to find out the best algorithm for this kind of datasets. The following measures were performed to analyze the outcome.

### 7.1 Data Classification Accuracy

The observation results were different with various parameters of machine learning algorithms. The maximum accuracy that was achieved with Random Forest was approximately 99%, whereas with SVM, it was approximately 98%. The result was highly influenced by the size of dataset. While considering larger dataset, the processing speed of Random Forest was faster than SVM.

### 7.2 Confusion Matrix

Confusion Matrix is a matrix which compares the real output and output obtained using an algorithm on a test dataset. In other words, a confusion matrix is a technique for summarizing the performance of a classification algorithm. Although we were able to achieve 99% accuracy, but classification accuracy alone can be a misleading metric for accuracy. Classification accuracy only provides a numerical value of whole datasets. To carefully analyze and understand every activity and its accuracy, confusion matrix is necessary.

The label and datasets are organized into a tabular form, or as a matrix, as shown in Figs. 2 and 3. Each row on the table corresponds to a predicted activity, whereas each column of the table corresponds to an actual activity. The diagonal concentrated values are the correctly predicted values. But if we compare the data between random forest and SVM in activity 2 (sit n relax), 3 (Lying Down) and 4 (walking), random forest is able to classify the data properly than support vector machine.

### 7.3 Visual Analysis

To perform visual analysis, we plotted the data of one particular data attributes of the predicted and actual data values, i.e., X-axis Acceleration from chest sensor. First, we plot the test results and then the predicted results, and later on combined them both to visualize the difference. There is not much difference between the graphs as shown in Figs. 2 and 3.

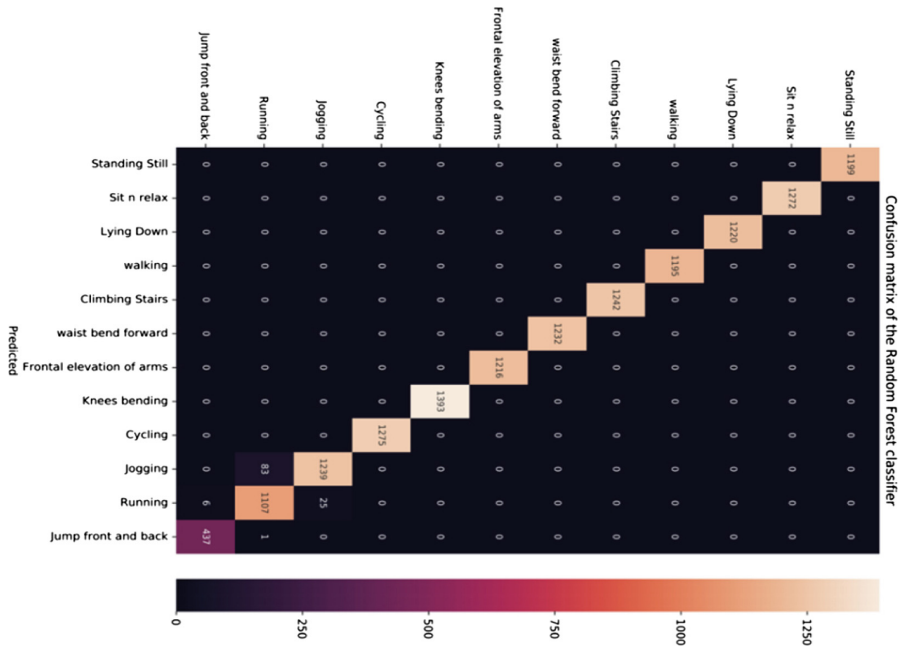


**Fig. 2.** Confusion matrix after using random forest algorithm

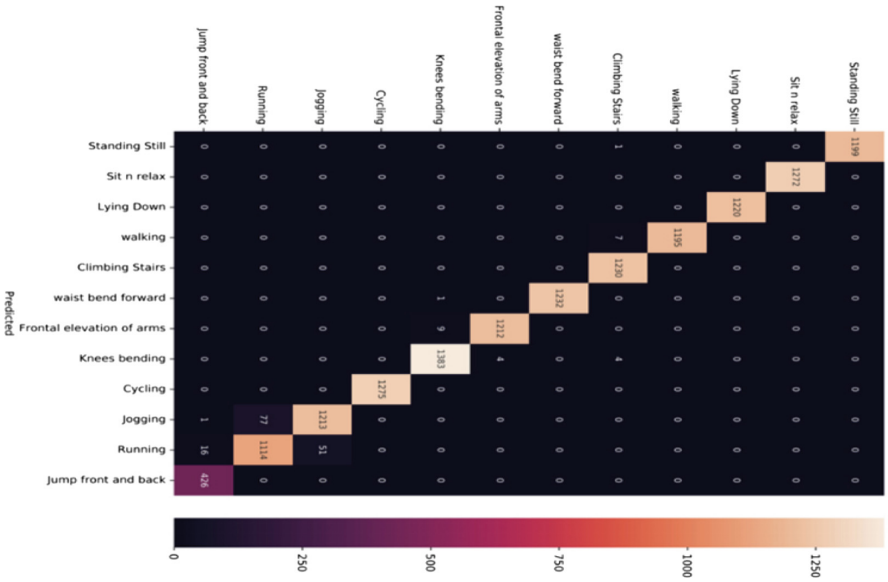


**Fig. 3.** Confusion matrix after using SVM algorithm

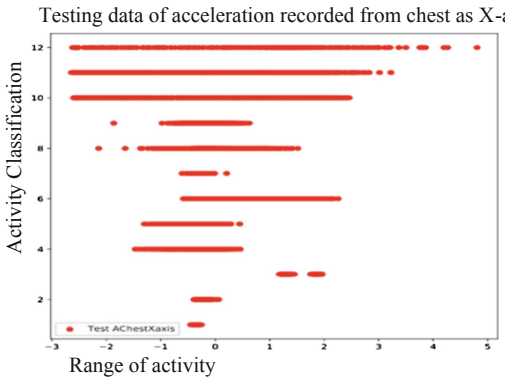


**Fig. 4.** Visual representation of testing data: X-axis acceleration recorded from chest sensors data using RF algorithm

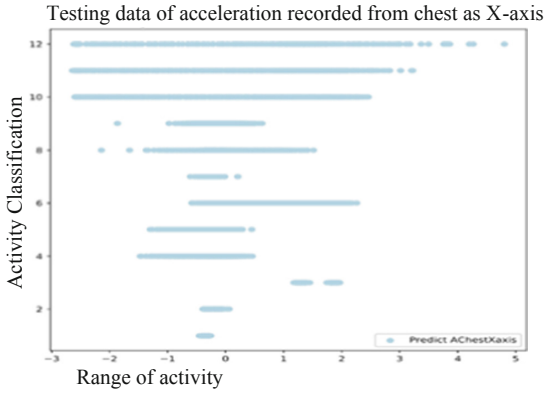


**Fig. 5.** Visual representation of predicted data: X-axis acceleration recorded from chest sensors data using RF algorithm

Figures 4 and 7 represents the testing data activity set of random forest and SVM. In both algorithms, the same data set was used, so there is not much difference. But in Figs. 5 and 8, we see the changes occurred on testing data result while using support vector machine and random forest. Then we combined both the visual representation of the activity sets and observed the differences between the real activity and predicted activity (Fig. 6 and 9).

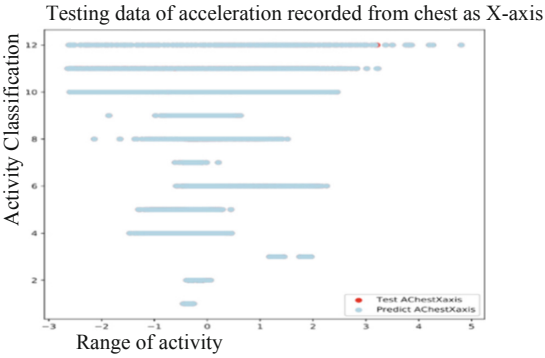


**Fig. 6.** Visual representation of testing and predicted data combined: X-axis acceleration recorded from chest sensors data using RF algorithm

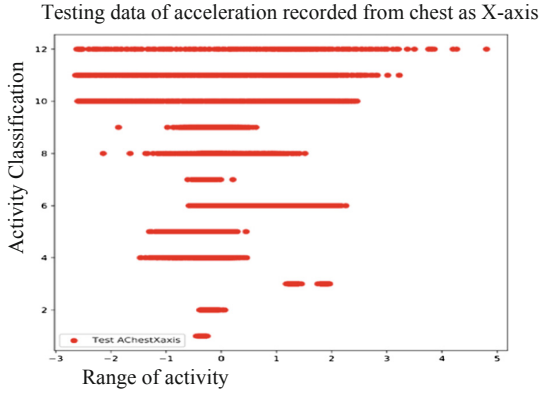


**Fig. 7.** Visual representation of testing data: X-axis acceleration recorded from chest sensors data using SVM algorithm

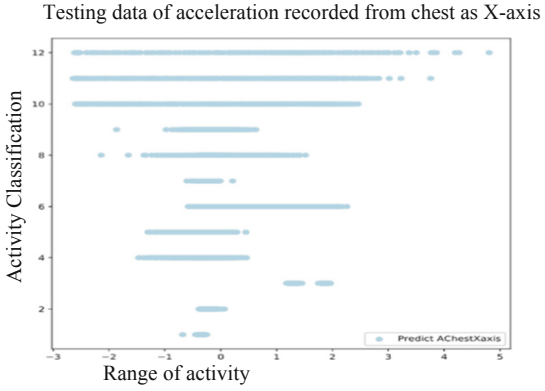


**Fig. 8.** Visual representation of predicted data: X-axis acceleration recorded from chest sensors data using SVM algorithm

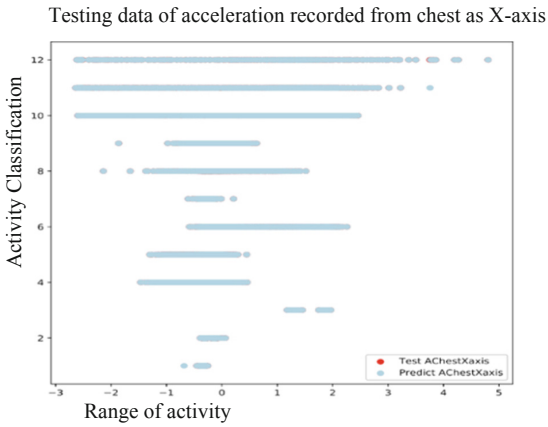


**Fig. 9.** Visual representation of testing and predicted data combined: X-axis acceleration recorded from chest sensors data using SVM algorithm

### 7.4 ECG Values Analysis with Respect to Different Activity Sets

In Fig. 10, the Y-axis represents the scaled ECG value, whereas the X-axis represents the number of recordings. Blue line represents the ECG notation whereas orange line represents the activity label. To understand the characteristic of the datasets and data concentrations, we have sorted the data values with respect to activity sets. Which can be observed in following diagram.

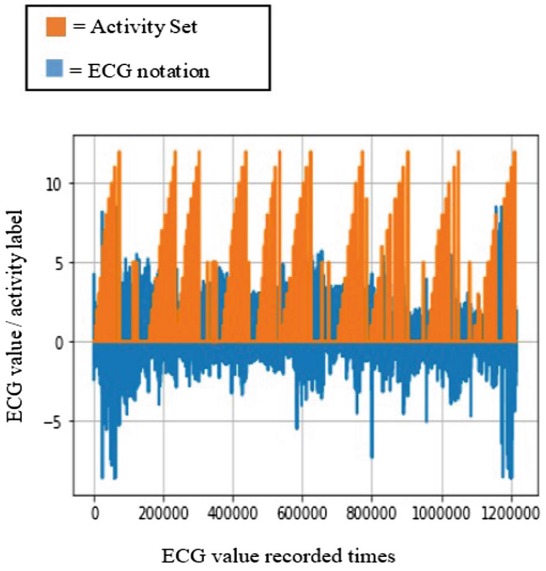


**Fig. 10.** Showing the ECG value with respect to activity sets before sorting.

In the above graph (Fig. 11), we compared the ECG values with respect to activity sets. We can observe that the datasets overall value has decreased drastically than previous graph, i.e. Fig. 10, because of the removal of the large number of ECG values with no activities. Although there is no recorded activity, but the ECG value fluctuates. Therefore, we can say that even while recording and classifying the data, we can still make a huge improvement on increasing the prediction accuracy and consistency of data by acquiring the activity data only.

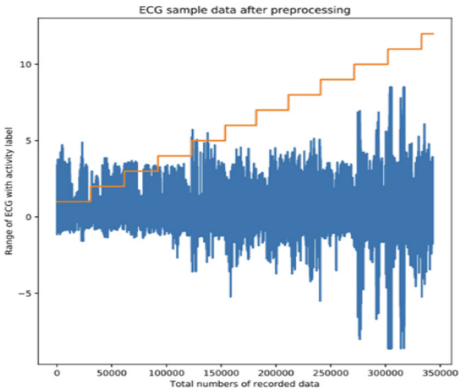


**Fig. 11.** Showing the ECG value with respect to activity sets after sorting.

From this ECG value, we identify the category classification so that we can implement the semantization to identify the critical situation. With semantization, a lot of information can be extracted from these data, which will further help to be more specific and accurate on analysis. For example, if the ECG value and activity value evaluation is on same track then the classification accuracy can be higher (more explanation on experiment results are given in Sect. 9).

## 8 ECG Classification Prediction with Random Forest

Having known that random forest provides more accurate activity classification on the used dataset, we studied the four categories of ECG data so that it can provide more specific information on the patient's status.

To make the model more informative and descriptive, ECG data is treated as an important data which outlines the objects vital signs in four categories (0- Classification Not Available, 1- Normal, 2- Atrial Fibrillation, 3- other Rhythm). To understand ECG classification and activity classification simultaneously, first ECG classification prediction model is trained and tested using random forest algorithm, as below (Fig. 13) The approach for training and testing of ECG datasets is same as of activity classification (Fig. 12).

*Note: Atrial Fibrillation – is an irregular, rapid heart rate which can increase the risk of heart stroke, heart failures or other heart related issues.

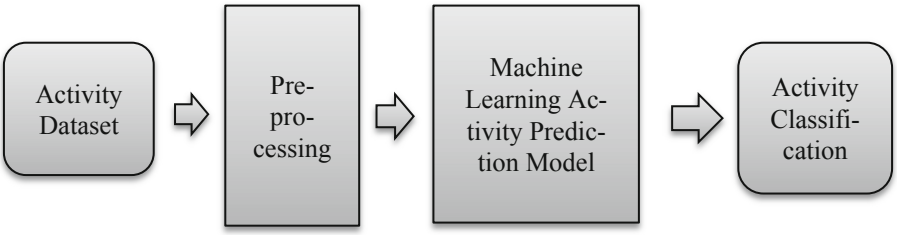


**Fig. 12.** Block diagram of training and testing of activity classification model

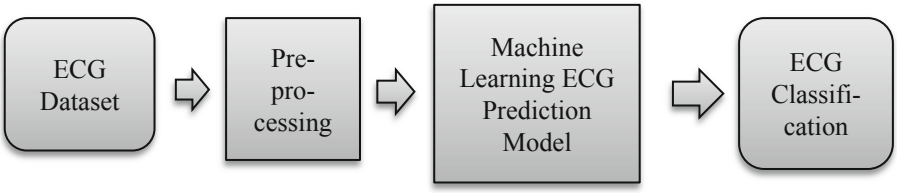


**Fig. 13.** Block diagram of training and testing of ECG classification model.

## 9 Activity Data Classification and ECG Data Categorization

After the training and testing of the models. Both models are used for same activity dataset which performs two task: activity classification and ECG classification, as in Fig. 14. Activity classification processes the whole activity datasets to predict the

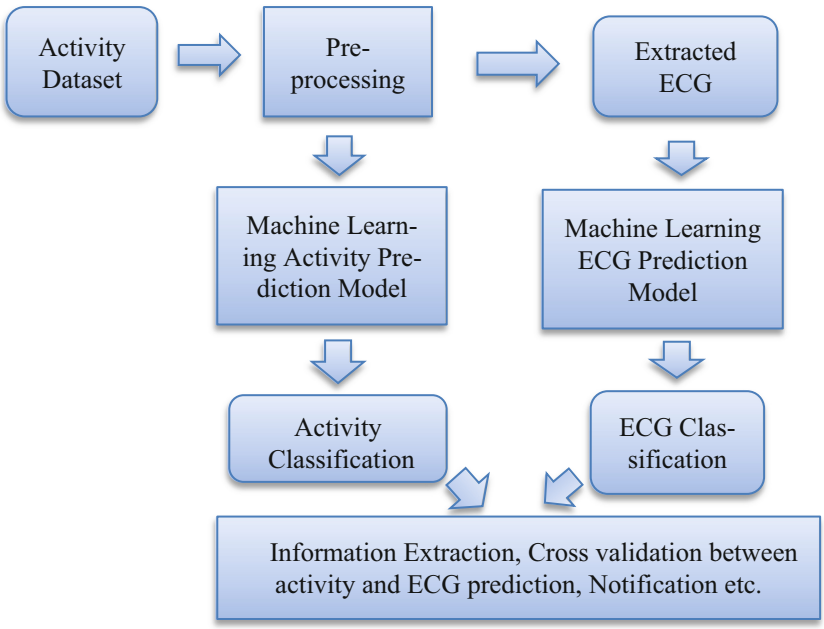


**Fig. 14.** Block diagram of activity classification and ECG classification simultaneously.

activity classification using random forest algorithm. In case of ECG classification, ECG data is extracted from whole activity dataset after pre-processing of datasets and processed through the random forest prediction model for ECG classification. After the classification process is finished, individual classification as well as comparing and contrasting the classification can be performed as required. By this approach the output becomes more meaningful than just activity classification.

### 9.1 Experiments with ECG Dataset

In this experiment, first of all, we extract the value from activity dataset. This data was a kind of relaxing and sitting represented as 2. This data was passed through the system architecture from which both activity classification prediction and ECG classification prediction was done. Activity classification prediction was achieved as 2 which means sitting and relaxing. And similarly, from 2 lead ECG value both result was 1, i.e. normal. So, generally a person who is relaxing or sitting should have normal heartbeat unless some other external factors like fear or heart problems.

### 9.2 Experiment Results and Observations

Highest training and testing accuracy achieved with random forest is 37.93%. The major reason behind low accuracy is because of similar kind of datasets which can be seen in Fig. 15. From the following confusion matrix grid we can observe the classification of dataset with higher confusion.

| | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| 0 | 185 | 874 | 85 | 615 |
| 1 | 136 | 2019 | 117 | 1312 |
| 2 | 55 | 775 | 133 | 812 |
| 3 | 149 | 1642 | 131 | 1760 |

**Fig. 15.** ECG training data confusion matrix.

*(0- Classification Not Available, 1- Normal, 2- Atrial Fibrillation, 3- other Rhythm)

From the above confusion matrix, the nature of data can be explained as highly related data. Because of very marginal differences of ECG value there is a lot of confusion between 4 different ECG categorical values. As diagonal values being correctly predicted values. Furthermore, to improve the accuracy of prediction ECG classifier output is compared with activity classifiers output.

## 10 Summary and Concluding Remarks

In this paper, we conducted experiments with mobile health data to determine which particular machine learning algorithm is best suited to our proposed personalized mobile health monitoring system. Experiment results depicts the characteristics of Support Vector Machines (SVM) and Random Forests (RF) machine learning algorithms with respect to our test dataset. The RF achieved a 99% data classification accuracy compared to 98% using SVM. Moreover, the RF exhibited faster processing speed with larger datasets. Confusion matrix was used to determine the behavior of the dataset with respect to the algorithm used. Furthermore, we classify and categorize the activity ECG datasets to derive more specific conditions of the subject. The aforementioned results provided us with clearer understanding of the algorithms and insights on realizing an intelligent edge analytics for mobile health monitoring.

### 10.1 Future Work

This study is still in its preliminary stage and much work has still to be done. We plan to study deeper on the algorithms while developing the framework on the data semantization for intelligent edge analytics.

## References

1. Rodriguez, N.D., Cuellar, M.P., Lilius, J., Calvo-Flores, M.D.: A survey on ontologies for human behaviour recognition. ACM Comput. Surv. **46**(4), 33 (2014). https://doi.org/10.1145/2523819. Article 43
2. Calvier, F.-É., Plantie, M., Dray, G., Ranwez, S.: Ontology based machine learning for semantic multiclass classification. In: TOTH: Terminologie Ontologie: Théories et Applications, Chambery, France, p. 100 (2013)
3. Bajaj, G., Agarwal, R., Singh, P., Georgantas, N., Issarny, V.: A study of existing ontologies in the IoT-domain. In: Elsevier JWS SI on Web Semantics for the Internet/Web of Things (2017)
4. Banos, O., et al.: Design, implementation and validation of a novel open framework for agile development of mobile health applications. BioMed. Eng. OnLine **14**(S2:S6), 1–20 (2015)
5. Banos, O., et al.: mHealthDroid: a novel framework for agile development of mobile health applications. In: Pecchia, L., Chen, L.L., Nugent, C., Bravo, J. (eds.) IWAAL 2014. LNCS, vol. 8868, pp. 91–98. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-13105-4_14
6. Dua, D., Taniskidou, E.K.: UCI machine learning repository. University of California, School of Information and Computer Science, Irvine (2017). http://archive.ics.uci.edu/ml
7. Machine learning. https://machinelearningmastery.com/. Accessed 07 May 2019
8. Andreotti, F., Carr, O., Pimentel, M.A.F., Mahdi, A., De Vos, M.: Comparing feature-based classifiers and convolutional neural networks to detect arrhythmia from short segments of ECG. In: Computing in Cardiology, vol. 44, pp. 1–4 (2017)
9. Goldberger, A.L., et al.: PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals. Circulation **101**(23): e215–e220
10. Physionet. https://www.physionet.org/. Accessed 07 May 2019