Framework for Monitoring and Recognition of the Activities for Elderly People from Accelerometer Sensor Data Using Apache Spark

Shubham Gaur[1], and Govind P. Gupta[2]
[1,2] Department of Information Technology,
[1,2] National Institute of Technology, Raipur (C.G.), India-492010
[1]shubh2495@gmail.com, [2]gpgupta.it@nitrr.ac.in

Abstract. Analysis of daily human activities becomes more open and prevalent with the advancement of sensors embedded in mobile devices. A number of applications such as fitness tracking, health analysis or user-adaptive systems, has been developed by tracking down detailed analysis of complex human activities. In this paper, a framework for the monitoring and recognition of the elderly people's activities using Apache-Spark Big-data processing tool is proposed. In the proposed framework, different classification techniques such as Logistic Regression, Decision Tree and Random Forest Classifier of ML Machine Learning library of Apache Spark are used to recognize human activities. In order to evaluate the proposed framework, two commonly known KAGGLE-UCI and WISDM Smartphone accelerometer datasets are used. Performance analysis of classification based human activities recognition schemesis evaluated in terms of training time, testing time, accuracy and F1-score. Results show that Logistic Regression classification after tuning it with Cross Fold Validation has provided better performance compared to remaining classification methods.

Keywords: Human Activity Recognition, Machine Learning, Apache-Spark, Classification.

1 Introduction

Recent advancement in microelectronics, sensors and low-cost and small size smartphone technologies, motivate the developments of the small size smart mobile devices. These devices are equipped with unmatched characteristics, exceptional computing power and high-end sensors for daily use. A number of activities are performed using smartphonesin our day to day life. One of the crucial factor that influences the popularity of these devices is the number of sensors these gadgets are equipped with, such as the camera, GPS, accelerometer, compass, touch sensor, proximity etc.

Human activities recognition using smartphone sensors have become a quest of great interest in various applications such as security, military and medical domain. For example, Obese people or heart patients are bound to take precautionary measures always. They follow a structured routine to keep themselves fit. Thus, recognizing

activities such as walking, cycling, or running becomes quite helpful to provide an assessment reportabout the patient's behaviourto the caregiver. Similarly, Old people normal activities can be diagnosed to keep a regular track of their health and provide real-time help with such continuous monitoring.

Detecting human activities and their behaviour using accelerometer sensors of mobile devices [1] is the fundamental research challenge in the field of the ambient-assisted living system. Human body performs complex activities. These activities need to be broken down to two or more simple activities. Thus, determining human activities require lots of mathematical computation. Detection and recognition of these activities play an important part to solve human-centric problems. A better way to achieve activity recognition can be using sensor data. Sensors enable the smartphone to collect data from and about its environment. This provides better control over the data and faster visualization. The data collected can be used to create a context for its functions.

In this work, a framework for human activity recognition using Apache-Spark Big-data processing tool is proposed. Accelerometer sensor data is used to detect and infer the different activities of the human. Proposed framework employed different classification techniques such as Logistic Regression, Decision Tree and Random Forest Classifier of the ML Machine Learning library of Apache Spark to recognize human activities. In order to evaluate the proposed framework, two commonly known Smartphone accelerometer datasets such as KAGGLE-UCI accelerometer dataset and WISDM accelerometer datasets are used. Performance analysis of classification based human activities recognition schemes is evaluated in terms of training time, testing time, accuracy and F1-score. Main contributions of this paper are as follows:

- The problem of perpetual enlargement of data collected from smart phone-sensors is discussed.
- Proposed framework focus on handling endlessly increasing smartphone accelerometer sensor data using Apache Spark Big data processing tool. Thus proving a scalable solution for Human activities recognition.
- Performance of the proposed framework using different classifiers is evaluated in terms of training time, testing time, F1-score and accuracy.

This paper is structured as follows: Section 2 presents a brief overview of the related works on human activity recognition. Section 3 proposed a framework for human activity recognition using accelerometer sensor data using Apache Spark tool. In section4, result analysis and its discussion are presented. Finally, the paper is concluded in Section5.

2 Related Works

The concept of activity recognition has been there for a long time now. It came into existence under the work proposed by the Neural Network House [2]. Their work laid the foundation to the concept of home automation, and a variety of location-based applications aimed to adapt systems to users' whereabouts [3-5]. Tapia et al. [6] de-

veloped a system for recognizing activities in the home setting using a set of small and simple state-change sensors. Their work displayed the possibility of recognizing human activities of medical professionals such as toileting, bathing etc. with detection accuracies ranging from 25% to 89% depending on the evaluation criteria used.

Most cited work that was done using multiple sensor data was by Bao et al. in [7]. This work concluded that thigh placement for the sensor provides the most accurate result for any activity performed. Their work was later used by Kwapisz et al. in [8] to perform Human Activity recognition using a single accelerometer sensor embedded with a smartphone. They used decision trees and multilayer perceptrons on their own hand-crafted features. However, both their classifiers were able to show better accuracy as compared to other data mining techniques but were not able to classify very similar activities like going upstairs or downstairs.

In 1999, research conducted by Foerster et al. [9] has demonstrated an accuracy of 95.6% in a controlled data acquisition experiment using tri-axial accelerometers to recognize locomotive activities. It was found that in natural conditions (i.e., outside of the laboratory), the accuracy dropped to 66%. In [10], authors have proposed activity recognition from acceleration data based on discrete cosine transform and SVM. This method has provided an accuracy of 92.25% and 97% using different methodologies. Khan et al. have proposed a hierarchical recognition model where accelerometer's Position Free Human Activity Recognition used. They used Artificial Neural Networks (ANN) based on the feed-forward back-propagation algorithm and gaining an accuracy of 95% and up to 98%, using two-level LDA recognition [11]. Work performed by Sharma et al. used neural networks (ANN) [12]. Later, Khan et al used decision trees based model to classify basic activities [13].

Wu et al. have proposed k-nearest neighbours (kNN) as the best classifier, but it too failed to recognize similar looking activities distinctly [14]. Anguita et al. [15] used 561 handcrafted features. They took the help of a multiclass support vector machine. Their classification involves six different activities. All of these works have derived their own set of hand-made features for classification and evaluation. These experiments lack precision and the ability to classify under a scaled environment. In this work, using accelerometer-sensor data of a smartphone, we show that Apache Spark pipelines are able to overcome these problems of current HAR systems.

3 Framework for Monitoring and Recognition of Activities for Elderly People Using Apache Spark

This section presents a detail discussion of the proposed framework for human activities recognition using Apache Spark [18]. Apache Spark is used for fast and efficient data processing due to the nature of big sensor data sets. Fig.1 shows different sub-modules of the proposed framework and their associations to achieve fast and efficient human activities recognitions using Smartphone's accelerometer sensor data. A brief discussion about different components of the proposed framework is presented as follows:

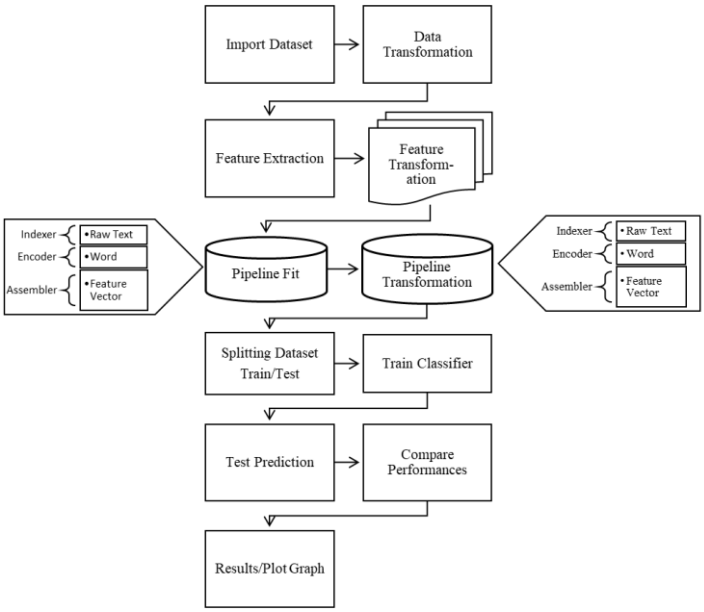

Fig. 1. Framework for Monitoring and Recognition of Activities for Elderly People Using Apache Spark.

3.1 Description of Recognition Process

In order to evaluate the proposed framework, we have used two accelerometer sensor datasets such as WISDM [19] and KAGGLE-UCI [20] dataset. Firstly, we pre-process the data to make it fit for an import in Apache Spark. Pre-process involves data cleaning and removing wild entries from the dataset. Apache Spark uses two types of data structures. One is Resilient Distributed Datasets (RDD) and another is data frames for pipelines. Our whole work is focused on pipeline architecture as it provides better flexibility in handling data. Pipeline feature of Apache Spark makes it a great tool for solving machine learning problems. We import the dataset to perform the regular transformation. Data transformation involves normalization and labelling of data. Labelling of activities are being performed by the help of pandas (data manipulation tool in python). We have labelled all the activities with unique labels to distinctly recognize all the activities.

Feature Extraction is another important measure of any Human Activity Recognition Model. Handmade features are used within our recognition model. These features are cast by applying statistical mathematical functions on regular time-series data.

Feature transformation involves the conversion of features in require format so that the data can be fed in the pipeline. This not only makes the time series data fit for our operations within the pipeline but also reduces the load on the spark node by creating a feature vector rather than feeding them individually.

Pipeline fit and pipeline transformation check the compatibility of our data on the Apache Platform within our proposed framework. Usages of these functions are the root cause of the use of ML (machine learning) library of apache spark. Dealing with pipelines is much flexible as compared to resilient data frames supported by MLlib (another machine learning module present in Apache Spark). *Pipeline.fit()* and *Pipeline.transform()* use the same series of steps to generate a feature vector; only the key difference is that former make it fit on training set while the later transform according to the testing set.

We split the data into two sets i.e. one is training and another one is testing set. Later inthe process, analysis and prediction accuracy with multiple classifiers is tested on the testing dataset. In our case, there are three main classifiers such as Logistic Regression, Decision Tree and Random Forest. We further tuned our Logistic Regression with Cross-fold Validator to enhance the accuracy of the system.

3.2 Apache Spark

Apache Spark is a big-data processing tools based on the cluster computing framework [18]. Apache Spark enhances processing speed by performing in-memory cluster computing. It extends the Map-Reduce model to use it efficiently for many kinds of computations including stream processing and interactive queries. Spark SQL and Data Frames are used in the proposed framework where a large amount of accelerometer sensor data has been structured using Spark SQL. In the pre-processing part of the proposed framework, dataset further converted into data frames for classification and evaluation. The proposed framework uses the MLlibrary of Apache Spark due to its flexibility to work in pipelines. Machine learning classification algorithms such as Logistic Regression, Decision Tree and Random Forest has been evaluated provided under ML library of Apache Spark while handling pre-processed data in pipelines.

In machine learning paradigm, it is generic to run a series of machine learning algorithms to assimilate and learn from data. Data need to be fed within the pipeline in a particular format. These basically include three steps to make any dataset pipeline ready. These are:

- Indexer: Labeling Activity.
- Encoder: String to float conversion of feature data.
- Assembler: Creating a feature vector for encoded data.

A simple recognition process might involve several stages illustrated in above Fig. 1.

3.3 Classifiers used in Human Activities Recognition

In order to identify different activities of the human, we have used four different classifiers such as Logistic Regression, Decision Tree, Random Forest and Logistic Regression with Cross-Fold Validation. These classifiers are briefly described as follows:

Logistic Regression: Itis a method of classification which performs predictions for deciding the categorical response and predicts the probability of the outcomes. It's a linear method of classification with the logistic loss function given as:

$$L(w; x, y) := \log(1 + e^{-yw^T x}) \quad (1)$$

If *x* denotes the new data point, then predictions are made by the model by applying the following logistic function:

$$f(z) = \frac{1}{(1+e^{-z})} \quad (2)$$

where $z = w^T x$, if $f(w^T x) > 0.5$, the outcome is +ve, otherwise,the outcome is –ve

It has two variants- Binomial Logistic regression and Multinomial logistic regression (generalized binomial logistic regression). Binomial logistic regression works to predict a binary outcome whereas multiclass logistic regression predicts the multiclass outcome. Our work uses a Multiclass Classification. Multiclass outcome contains K-1 models of binary logistic regression regressed against first class. After running K-1 models, the class with the largest probability will be the predicted class. To choose between these variants one can select the family parameter or the spark can auto-infer the correct variant if left unset.

Decision Tree Classifier: It is a machine learning method for regression and classification in sequential decision problems which are easy to handle categorical features and easy to interpret. It supports both binary and multiclass classification problem. In apache spark, binary and multiclass classification and regression by decision trees are supported by *spark.ml.classification* package. The implementation allows distributed training with millions of instances and partitions the data by rows. Decision trees have three types of nodes - Root node, Splitting nodes and Terminal node. The process initiates from the root node, based on the decision done here, the splitting node gets selected, based on the decision done here the child split node is selected further. This goes on further until the terminal node is reached. The final outcome is the value at the terminal node. The Basic algorithm of decision trees implemented in apache spark is a greedy algorithm. It performs binary partitioning of feature space in a recursive manner. The same label is predicted for each leaf (bottom-most partition). Each partition is greedily chosen by the selection of best split from a set of possible splits.

Random Forest Classifier: Itis the machine learning method for classification and regression based on the decision trees. It is one of the ensemble methods supported by *spark.ml* for the decision trees. Random forest classifier also has two variants for classification i.e. binary classification and multiclass classification. SparkML supports both the variants.

Random forests can train a number of trees in parallel. It's less prone to overfitting because training various trees in parallel through this algorithm reduces the chances of overfitting. Since the performance in Random Forest classifier improves monotonically with a number of trees, so they are easier to tune. The basic algorithm injects randomness to create different decision trees. These components are– consideration of different random subsets of features for splitting at each node and sub-sampling the original dataset after each iteration in order to get a different training set. Prediction for

a new instance is made by aggregating the predictions from its decision tree set. This aggregation can be different for classification and regression. For classification, majority vote rule is used i.e. the label is predicted to be the class that has the majority of votes, whereas for regression averaging rule is used i.e. the average of all the tree predictions is predicted to be the label.

Logistic Regression with Cross-Fold Validation: It's a method for model selection in machine learning. In apache spark, *spark.ml.tuning* provides cross-validation [18] tool for it. Cross-fold validation splits the data into folds which can be used as separate training and test datasets. We have used K=5 cross-fold validation, resulted into 5(training, test) dataset pairs from which 4 parts out of 5 of the data are used as training-set and 1 out of 5 parts of the data is used as testingset. In general, K- cross-fold validation is a technique to train and test the data k times. This tool requires estimator, set of parameters and an evaluator.

The estimator is used fortuning the pipeline. Set of parameters are required to choose parameters from the parameter grid. In our experiment, Estimator used over Logistics Regression includes regularization parameter, elastic net parameter, number of features and number of iterations. The evaluator is used formeasuring or to evaluate how well a Model does on test data. We have used three evaluators: Regression Evaluator for regression problems, Binary Classification Evaluator for binary data and Multiclass Classification Evaluator for multiclass problems. We have tested our preprocessed dataset over both the classification techniques. However, Human Activity Recognition being a multiclass problem definitely outperformed the binary classification.

4 Result Evaluation and its analysis

This section presents results analysis of the proposed framework for the human activities recognition using two different accelerometer sensor datasets. Classification and Regression machine learning algorithms provided within the ML library of Apache Spark has been optimized to deal with the data feed within pipelines. Evaluation and performance metrics have been calculated on both the datasets. We kept the number of activities as high as 6 to 7 activities.

4.1 Description of Datasets

This section presents a brief description of the UCI-HAR [20] and WISDM-HAR [19] accelerometer Sensor Datasets

- **UCI-HAR[20] Accelerometer Sensor Dataset:**This database consists of observations of 30 study participants performing daily common activities. They perform all the activities with a smartphone mounted on a waist with embedded sensors such as an accelerometer, gyroscope etc. The objective is to classify activities into one of the six activities performed. All individuals within an age range of 19-48 years performed six activities using the smartphone such as WALKING, WALKING_UPSTAIRS, WALKING_DOWNSTAIRS, SITTING, STANDING, LAYING. 3-axial linear acceleration and 3-axial angular velocity

measures have been recorded using accelerometer and gyroscope sensor, at a constant rate of 50Hz. The experiments have been video-recorded and labelling of data is manually performed. The obtained dataset has been randomly partitioned into two sets, in multiple split ratios to perform recognition activities. The sensor signals (accelerometer and gyroscope) were pre-processed by applying noise filters and then sampled in fixed-width sliding windows of 2.56 sec and 50% overlap (128 readings/window).

- **WISDM-HAR[19] Accelerometer Sensor Dataset:**We have used transformed WISDM Human Activity Recognition dataset within this work for cross-referencing our results. Activities described within the dataset are WALKING, JOGGING, UPSTAIRS, DOWNSTAIRS, SITTING, and STANDING. There isa total of 5423 instances of all the activities such as2048-WALKING 1,626-JOGGING, 633-UPSTAIRS, 529-DOWNSTAIRS, 307-SITTING, and 247-STANDING. Our experiment only uses normalized features leaving out discrete time series data for constructing stricter recognition model using machine learning Apache Spark and reducing in-memory data operations.

4.2 Result Analysis

In the experiment, for the evaluation of the proposed framework, datasets are splits into three different ratios such as 70%-30%, 80%- 20% and 90%-10% for training and testing, respectively. Table 1 shows the results for each classifier for the WISDM dataset in terms of training time, testing time, F1-score and accuracy. It can be observed from Table 1 that Logistic Regression with Cross-fold Validation Model performs better than other classifiers. It is observed from Table 1 that proposed a framework for Logistic Regression with Cross-fold Validation Model takes maximum training time to compare to other classifiers and with Logistic Regression model takes minimum time for training.

Table 1. Training and Testing Time when data split in multiple ratios for WISDM-HAR[19] datasets

S.No.	Classifier	Split Ratio	Training Time	Testing Time
1.	Logistic Regression	70-30	11.482	0.053
		80-20	2.176	0.056
		90-10	2.612	0.04
2.	Decision Tree	70-30	13.187	0.076
		80-20	6.999	0.082
		90-10	7.016	0.071
3.	Random Forest	70-30	24.188	0.086
		80-20	21.276	0.075
		90-10	20.91	0.095
4.	Logistic Regression + Cross Fold Validaton Model	70-30	112.925	0.04
		80-20	121.234	0.024
		90-10	103.893	0.021

Table 2 listed the training and testing time taken by the proposed framework for KAGGLE UCI HAR dataset with different split ratio and different classifiers. It can

be observed from Table 2 that Testing time for the proposed framework with Logistic Regression Model takes minimum time to compare to the remaining classifiers. However, Logistic Regression with Cross-fold Validation Model takes maximum testing time.

Table 2. Training and Testing Time when data split in multiple Ratio

S.No.	Classifier	Split Ratio	Training Time	Testing Time
1.	Logistic Regression	70-30	12.506	0.282
		80-20	5.868	0.143
		90-10	8.049	0.191
2.	Decision Tree	70-30	17.362	0.198
		80-20	15.299	0.20
		90-10	15.299	0.50
3.	Random Forest	70-30	18.738	0.198
		80-20	19.548	0.209
		90-10	180675	0.204
4.	Logistic Regression + Cross Fold Validaton Model	70-30	271.196	0.203
		80-20	216.06	0.17
		90-10	220.98	0.36

Fig 2 illustrates the performance evaluation of the proposed framework in terms of accuracy by varying the application of different classifiers for activities recognitions. I can be seen from Fig. 2that the proposed framework with Logistic Regression with Cross-fold Validation Model achieves accuracy 91.9% for KAGGLE UCI HAR dataset and 73.3% for WISDM dataset. Fig. 3 shows performance of the proposed framework in terms of F1-score by varying the application of different classifiers for activities recognitions. The proposed framework with Logistic Regression with Cross-fold Validation Model achieves F1-score 91.9% for KAGGLE UCI HAR dataset.

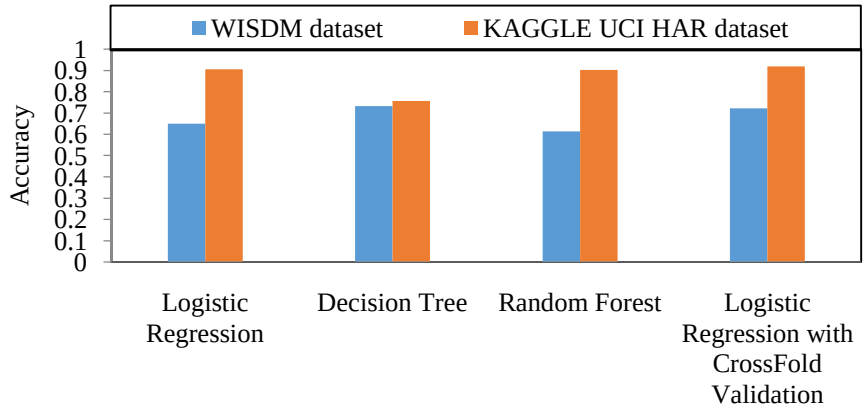

Fig. 2. Accuracy vs. Classification on WISDM and Kaggle-UCI HAR datasets when data split in 70-30% train-test ratio

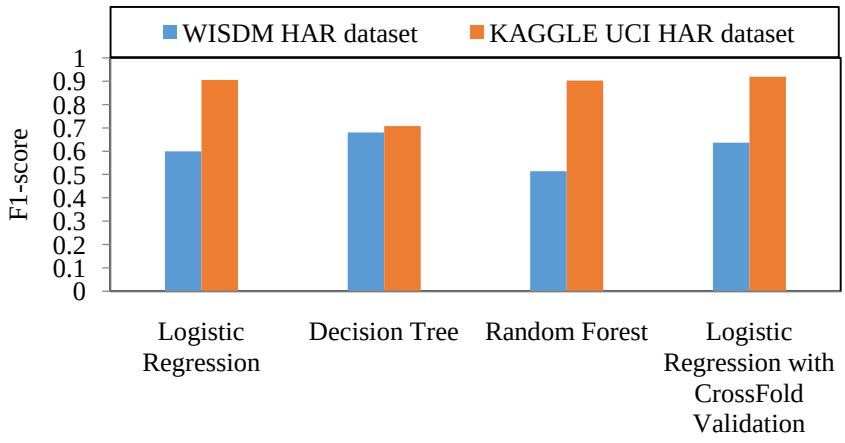

Fig. 3. F1-score vs Classification on WISDM and Kaggle-UCI HAR datasets when data split in 70-30% train-test ratio.

5 Conclusion

In this paper, a framework for human activity recognition using Apache Spark big data processing tool is presented. For the pre-processing of the accelerometer dataset such as Data transformation, normalization and labelling of the human activities, is being performed by the help of the Pandas, a data manipulation tool in python. For the recognition of human activities for different classifiers are used such as Logistic Regression, Decision Tree, Random Forest and Logistic Regression with Cross-fold Validation Model. In the experiments, it is observed that 70-30 Split Ratio i.e. 70% of the dataset for training and 30% of the dataset for testing, as the most convenient setup to deal with a variety of data. This avoids overtraining of the dataset but enough trained for testing purposes. Result analysis shows that Logistic Regression with Cross Fold Validated over 5 fold, have performed better than other classifiers in terms of accuracy and F1-score. We have achieved an accuracy of 72.10% when tested over WISDM Human Activity Recognition Accelerometer-Sensor time series data. It provided the maximum accuracy of 91.02% when tested over KAGGLE-UCI Human Activity Recognition Accelerometer-Sensor time series data.

References

1. Timo Sztyler, H. S.,"Position-aware activity recognition with wearable devices. Pervasive and Mobile Computing", 38(2), 281-295. doi:https://doi.org/10.1016/j.pmcj.2017.01.008, Jan 30,2017.

2. Mozer, M.C. “The Neural Network House: An Environment that Adapts to its Inhabitants”,2002.
3. U. Leonhardt and J. Magee, “Multi-sensor location tracking,” in Proc. 4th ACM/IEEE Int. Conf. Mobile Comput. Netw., pp. 203–214, 1998.
4. A. R. Golding and N. Lesh, “Indoor navigation using a diverse set of cheap, wearable sensors,” in Proc. 3rd Int. Symp. Wearable Comput., pp. 29–36, Oct. 1999.
5. A.Wardand,A.Hopper,“A new location technique for the active office,” IEEE Personal Commun.in Oct. 1997, vol. 4, no. 5, pp. 42–47.
6. Tapia E.M., Intille S.S., Larson K. “Activity Recognition in the Home Using Simple and Ubiquitous Sensors” in Oct. 1997
7. Bao, L. Intille, S. (2004). Activity recognition from user-annotated acceleration data. In Pervasive computing, Lecture notes in computer science: Vol. 3001 (pp. 1–17). Springer
8. Kwapisz, J. Weiss, G. Moore, S. (2010). Activity recognition using cell phone accelerometers. SIGKDD Explorations, 12 (2), 74–82
9. F. Foerster, M. Smeja, and J. Fahrenberg, “Detection of posture and motion by accelerometry: a validation study inambulatory monitoring,” Computers in Human Behavior, vol. 15, no. 5, pp. 571–583, 1999
10. Z. He and L.-W. Jin, “Activity recognition from acceleration data using ar model representation and SVM,” in International Conference on Machine Learning and Cybernetics, vol. 4, pp. 2245–2250, 2008.
11. A. Khan, Y. Lee, and S. Lee, “Accelerometer’s position free human activity recognition using a hierarchical recognition model,” in IEEE International Conference on e-Health Networking Applications and Services (Healthcom), pp. 296–301, 2010.
12. Sharma, A. Lee, Y.-D. Chung, W.-Y.,“High accuracy human activity monitoring using neural network” in proceedings of international conference on convergence and hybrid information technology, (pp. 430–435), 2008.
13. Khan, A. M.“Recognizing physical activities using Wii remote. International Journal of Information and Education Technology”, 3 (1), 60–62
14. Wu, W., Dasgupta, S., Ramirez, E. E., Peterson, C., Norman, G. J. Classifica tion accuracies of physical activities using smartphone motion sensors. Journal of Medical Internet Research, 14 (5). doi: 10.2196/jmir.2208, 2012.
15. Anguita, D. Ghio, A., Oneto, L. Parra, X. Reyes-Ortiz, J. L. Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine. In Proceedings of international conference on ambient assisted living and home care (IWAAL) (pp. 216–223),2012.
16. Gupta GP, Kulariya M. A framework for fast and efficient cyber security network intrusion detection using Apache Spark. Procedia Comput Sci. 2016;93:824–31.
17. Kulariya M., Saraf P., Ranjan R. and Gupta G. P., “Performance analysis of network intrusion detection schemes using Apache Spark,” 2016 International Conference on Communication and Signal Processing (ICCSP), Melmaruvathur, 2016, pp. 1973-1977.
18. Spark ML Programming Guide. (n.d.). Retrieved from Apache Spark: https://spark.apache.org/docs/1.2.2/ml-guide.html
19. WISDM Human Activity Recognition Dataset, W. A. (n.d.). Retrieved from http://www.cis.fordham.edu/wisdm/dataset.php
20. UCI Machine Learning. (n.d.). Human Activity Recognition with Smartphones. Retrieved from Kaggle: https://www.kaggle.com/uciml/human-activity-recognition-with-smartphones/home