Human Activity Recognition Using Smartphone Sensor

Pranshu Agrawal
IIIT Delhi
pranshu18170@iiitd.ac.in

Raghav Agarwal
IIIT Delhi
raghav18177@iiitd.ac.in

Ratna Sai Kiran
IIIT Delhi
ratna18179@iiitd.ac.in

Abstract

Human Activity Recognition is the classification of the activity a human is performing by evaluating the data obtained using sensors that are affected from human movement. Accelerometer and gyroscope are two such sensors which are commonly available in smartphones, from which the data obtained can be used for human activity recognition. The activities that a person performs over a time period can be analyzed to track his/her health and recommend him/her suitable actions to maintain good health. Such an analysis can be helpful in preventing lifestyle related disorders, due to Work from home and online classes' culture, by alerting the user in case of inactivity for a long time.

We use the UCI HAR public dataset to classify the human movements into 6 different activities. We apply different data reduction techniques such as PCA, SVD and statistical feature selection. Different classification techniques: Support Vector Machines, Random Forest Classifiers, K nearest neighbors, Gaussian Naive Bayes, Logistic Regression and Decision Trees.

Github Repository Link:
https://github.com/agarwalraghav2012/Human-Activity-Recognition-ML

1. Introduction

Human Activity Recognition has been used for several healthcare applications for a long time. Traditionally, resource intensive methods such as computer vision or installation of dedicated sensors on the user's body have been used, however due to advancement in electronic devices, sensors like accelerometer and gyroscope can fit easily in small devices like smartphones. Accelerometer measures the change in speed along the 3 different axes over a period of time, while Gyroscope uses gravity to measure orientation. Data from these devices can be evaluated for solving the Human Activity Recognition problem. We propose to use these sensors to classify the human activities into different classes namely: sitting, standing, walking, walking upstairs, laying and walking downstairs.

The data obtained from the smartphone sensors is processed over a sliding time interval of 2.56 seconds to obtain 561 feature vectors. These features are then reduced using data reduction techniques: SVD, PCA, tSNE and statistical feature selection. We obtain our best results in terms of accuracy, precision, recall and F1-score by using the statistical feature selection and training the Random Forest model on it.

2. Literature Survey

Different studies have been conducted in the past for solving this problem using data from smartphone sensors. The data obtained from these sensors is continuous in time domain, so various preprocessing strategies such as time domain averaging, peak value, Frequency domain analysis etc. have been used for this type of data. Since the dataset consists of a large number of features and data points, various feature reduction techniques such as statistic feature selection, feature level fusion and score level fusion have been used on the data.

3. Dataset

3.1. The UCI HAR dataset

The publicly available UCI HAR dataset's data has been taken from an experiment carried out with a group of 30 volunteers within an age bracket of 19-48 years. Each person performed six activities (1.WALKING, 2.WALKING_UPSTAIRS,3.WALKING_DOWNSTAIRS, 4.SITTING, 5.STANDING, 6.LAYING) wearing a smartphone on the waist. Using its embedded accelerometer and gyroscope, data was captured as 3-axial linear acceleration and 3-axial angular velocity at a constant rate of 50Hz. The obtained dataset has been randomly partitioned into two sets, 70% Train & 30% Test. Some data preprocessing techniques like Noise Filtering were applied to the raw data to come out with 561 features like mean, median, correlation, energy, entropy etc. removing effects of g, overlapping.

In total the training data is of the dimensions 7352x561, while testing data has dimensions 2947x561.

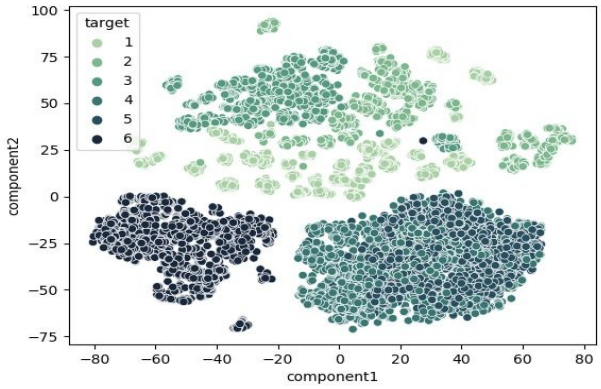
3.2. Feature Reduction

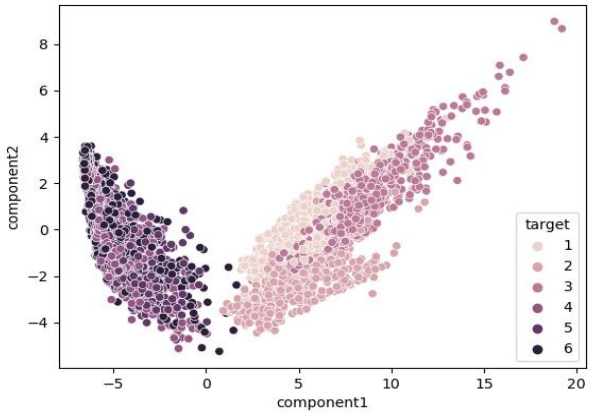
We have reduced these features through various data reduction and data selection strategies. This lowers the requirement of resources for training the model, and also prevents overfitting. The data reduction techniques we used are:-

PCA - Principal Component Analysis is used to reduce the dimensions of a given matrix using Linear Algebra techniques without losing the essence of the data.

Statistic Feature Selection - Only the features corresponding to the mean and standard deviation would be considered, which are commonly used features in statistics.

tSNE (t distributed Stochastic Neighbor Embedding)- was used to visualize the data reduced through the PCA method.





4. Methodology

After reducing the dimensions of the Training data, we feed the data into different models. Training our model for appropriate number of epochs and avoiding overfitting of our model, K-fold CV can be used here as well.

Once we've trained our model, we will preprocess our testing data the same way we did the training data and then predict the labels for the testing data.

Since it is a multi- class Classification problem. We used the following Machine Learning Algorithms:-

4.1. KNN

K-Nearest Neighbour is one of the ML Algorithms based on supervised learning techniques which can be used to solve classification and regression problems. It works even for non-linearly separable data. This paper has used the KNN classifier from Sklearn library for multi-class classification on different Datasets. KNN is very fast and gives comparable accuracy.

4.2. Logistic Regression

Logistic regression is a supervised machine learning algorithm which can be used for prediction of probability for classification. This model is for binary class classification but it can be used for multiclass classification as well. This paper uses a logistic function to classify into binary labels, but can be used to classify multi class problems.

4.3. SVM

Support vector Classifier is used to fit the data we provide. It returns a hyper plane that divides or categorize our data. SkLearn library have SVC, NuSVC and LinearSVC classifiers which can be used to perform multi-class classification on different type of datasets (sparse or dense). This paper has used the SVC classifier from sklearn library for the classification of our Datasets. After pre-processing of the data SVC model has been used. First it's fitted using a train dataset and then used for classification. Finally Accuracy score and confusion matrix has been plotted.

4.4. Decision Tree

Decision Tree is a flowchart-like tree structure where internal nodes represent features, the branch tells us the decision rule and leaf nodes give us the outcome. The root node of the entire tree learns to partition by anticipating the best predictor. To get the right branch at every node, approval condition is recursively applied. DT Classifier from sklearn was used to perform the classification. We trained the model to get the accuracy scores.

4.5. Random Forest

This is an ensemble technique. Many decision trees (uncorrelated) are trained on various subsets of data and classification is based upon output of all these decision trees using averaging etc.

4.6. Gaussian Naïve Bayes

GNB uses Naive Bayes Algorithm to correctly classify models. It assumes independence between the given features. The sample is given label having maximum given Probability. This algorithm is based on Maximum A Posteriori (MAP).

5. Results & Analysis

After using PCA we reduced the features to 80 ,and applied the above mentioned algorithms to the data.

The results were as follows:-

SVM	Logistic Regr	Decision Tree	Random Forest	Gaussian NB	KNN
77.12	75.18	68.32	75.32	74.84	74.23

%	%	%	%	%	%

Table1: Accuracy for different models

Analysing the confusion matrix for the best model i.e SVM with an accuracy of 77.12%

	WALKING	Walking Upstairs	Walking downstairs	Sitting	Standing	Laying
Walking	414	7	75	0	0	0
Walking Upstairs	12	393	66	0	0	0
Walking Downstairs	165	62	193	0	0	0
Sitting	0	2	0	321	162	6
Standing	0	1	0	113	417	0
Laying	0	0	0	3	0	534

Table2: Confusion matrix for SVM

We can say that our model is classifying nicely between static and dynamic activities but when it comes to classifying in dynamic activities it is not performing that good i.e it is misclassifying between walking upstairs with walking downstairs and walking downstairs with walking. Also it is misclassifying between sitting and standing.
We need to tweek our model such that it performs well in the above case.
Statistical feature selection generated a much better model on various aspects. When used to train different models, using just the mean and Standard Deviation gave the following metrics on the test data: -

	Precision	Recall	F1-score	Accuracy
Gaussian NB	0.84	0.82	0.81	0.82
SVM	0.91	0.91	0.91	0.91
Logistic Regression	0.93	0.92	0.92	0.92
Decision Tree	0.84	0.84	0.84	0.84
Random Forest Classifier	**0.93**	**0.93**	**0.93**	**0.93**
K nearest neighbors	0.90	0.89	0.89	0.89

Table3: Evaluation matrix for Statistic Feature Selection

All models have a significant gain over training using PCA reduced data in terms of all the above metrics. The Random Forest Classifier performed the best with an accuracy of 0.93. The logistic regression, while being significantly less complex and less resource intensive over other algorithms, performed only slightly worse than the Random Forest Classifier. This might be due to overfitting of the other models. By Occam's Razor, Logistic Regression performs almost as well as the best model, while being much less resource intensive. Hence both Random Forest Classifier and Logistic Regression would be our go-to choices. The Confusion matrix for Random Forest Classifier is as given below, obtained on the test set: -

	WALKING	Walking Upstairs	Walking downstairs	Sitting	Standing	Laying
Walking	477	8	11	0	0	0
Walking Upstairs	22	441	8	0	0	0
Walking Downstairs	25	52	343	0	0	0

Sitting	0	0	0	443	48	0
Standing	0	0	0	25	506	0
Laying	0	0	0	0	0	537

Table4: *Confusion matrix for Random Forest Classifier*

The Confusion matrix for Logistic Regression is given as below, obtained on the test set: -

	WALKING	Walking Upstairs	Walking downstairs	Sitting	Standing	Laying
Walking	354	18	124	0	0	0
Walking Upstairs	26	391	54	0	0	0
Walking Downstairs	135	48	237	0	0	0
Sitting	0	4	0	238	241	8
Standing	0	0	0	240	283	0
Layi	1	0	0	0	0	536
ng						

Table5: Confusion matrix for Logistic Regression

From the above confusion matrices, we observe that the logistic regression model correctly segregates Walking, Walking_Upstairs and Walking_Downstairs classes, but is unable to differentiate between Sitting, Standing and Laying very precisely.
On the other hand, the Random Forest Classifier segregates the Standing, Laying and Sitting classes effectively, but is prone to slightly more incorrect predictions between the Walking, Walking_Upstairs and Walking_Downstairs classes.

6. Conclusion

By using different techniques we reduced the dimensions of data and applied different ML models on them to obtain the results.Our best Model was obtained by providing the features obtained through Statistical Feature Selection on the Random Forest Classifier and had an accuracy of 93% on unseen test data. In future, our work can be expanded to other more complex tasks such as playing particular sports like swimming and boxing.

5.1. References

[1] E. Bulbul, A. Cetin and I. A. Dogru, 2018, 'Human Activity Recognition Using Smartphones', ISMSIT IEEE , Ankara, 2018

[2] B. Lavanya and G. S. Gayathri,2017,'Exploration and Deduction of Sensor-Based Human Activity Recognition System of Smart-Phone Data', ICCIC IEEE , Coimbatore, 2017

[3] "A. Jain and V. Kanhangad, 2018, 'Human Activity Classification in Smartphones Using Accelerometer and Gyroscope Sensors', IEEE Sensors Journal, vol. 18 (no. 3), pp. 1169-1177

[4] https://archive.ics.uci.edu/ml/datasets/human+activity+recognition+using+smartphones