**HUMAN ACTIVITY RECOGNITION**

R KUMARI SIVA PARVATHI

Intern at Terra.ai

**Abstract:**

**Human activity recognition plays a significant role in human-to-human interaction and interpersonal relations. To enhance the study in the field of human interaction we have classified the human activity performed using Machine Learning and Deep Learning. As ‘Deep Learning’ is the idea of this decade, we have implemented one of the first advances in deep learning for object detection i.e. Long Short-Term Memory network(LSTM) a special kind of Recurrent Neural Network(RNN), capable of learning long-term dependencies.**

**The research is performed on data collected from accelerometer and gyroscope sensors built from the recordings of 30 participants performing activities of daily living while carrying a waist-mounted smartphone with embedded inertial sensors. The objective is to classify activities into one of the six activities performed WALKING, SITTING, STANDING,WALKING\_UPSTAIRS,WALKING\_DOWNSTAIRS,LAYING.**

**Introduction**

The human ability to recognize another person’s activities is one of the main subjects of study of the scientific areas of computer vision and machine learning. Because of this research, many applications, including video surveillance systems, human-computer interaction, and robotics for human behavior characterization, require a multiple activity recognition system.

With applications in various fields, we collected dataset where 30 volunteers within an age bracket of 19-48 years were asked to perform sixactivities(WALKING,WALKING\_UPSTAIRS,WALKING\_DOWNSTAIRS,SITTING, STANDING, LAYING) wearing a smartphone (Samsung Galaxy S II) on the waist. Using its embedded accelerometer and gyroscope, 3-axial linear acceleration and 3-axial angular velocity at a constant rate of 50Hz was captured. The experiments have been video-recorded to label the data manually.

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). The sensor acceleration signal, which has gravitational and body motion components, was separated using a Butterworth low-pass filter into body acceleration and gravity. The gravitational force is assumed to have only low frequency components, therefore a filter with 0.3 Hz cutoff frequency was used. From each window, a vector of features was obtained by calculating variables from the time and frequency domain.

**Data Collection & Preparation**

We used the data provided by Human Activity Recognition research project, which built this database from the recordings of 30 subjects performing activities of daily living (ADL) while carrying a waist-mounted smartphone with embedded inertial sensors. The complete data can be accessed from Terra-AI team.

Data was collected for 30 volunteers whose age was between 19-48 years. Each record in the data represents information about features like acceleration along x,y,z axes, velocity along a,y,z axes, 561 attributes derived from these basic measurements, identifier variable for the user & the activity being performed.

There are 6 categories of activities being performed:

1. Standing

2. Sitting

3. Lying

4. Walking

5. Walking Upstairs

6. Walking Downstairs

The raw data has separate text files for most of the variable groups & we have used the dataset that was saved as RData file. In this dataset, a single column(‘subject’) is used to identify a user and the last column(‘activity’) was used to identify the activity being performed when the measurements were taken. All other attributes are available in the same column oriented data format. This is important to know, because, the values in the dataset have been normalized.

**Exploratory Analysis**

**1.High dimensionality**:

The dataset contains 561 features and we started out by exploring how these are related to each other & whether there are some which can be safely ignored for our problem.

**2.Correlation Check:**

We built a correlation matrix for all 561 variables in one got to identify any apparent patterns in the relationships. We see that most of these features are highly correlated with each other and it’s a good decision to drop most of these highly correlated features since we can get the same information from some other feature with high correlation to a group of them.

**3.Variance Check:**

We checked our variable for zero or low variance so that they can be removed before running any analysis. Variables which do not change have low variance and will eventually have smaller impact on the classification model itself.

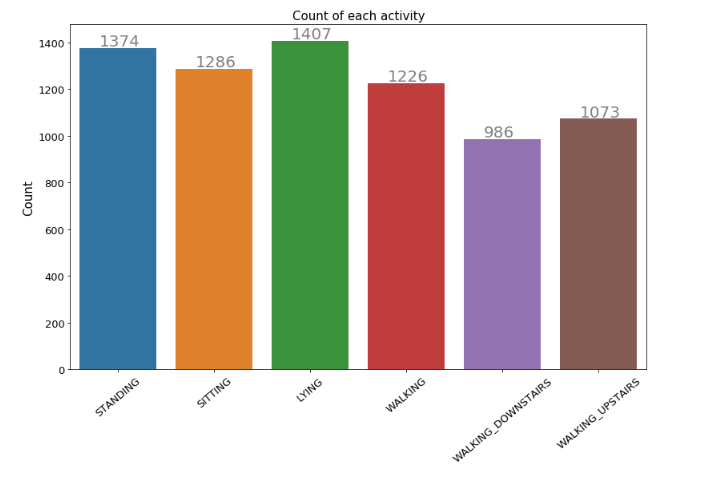
**4.Missing value Check:**

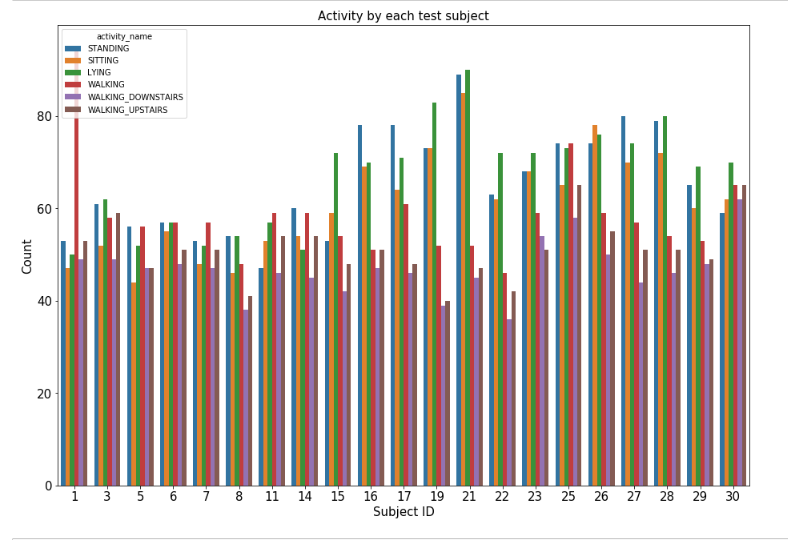
We checked for any missing values in our columns, which might lead to errors in any future analysis but didn’t find any and so proceeded with the complete dataset.

**5.Visual exploration:**

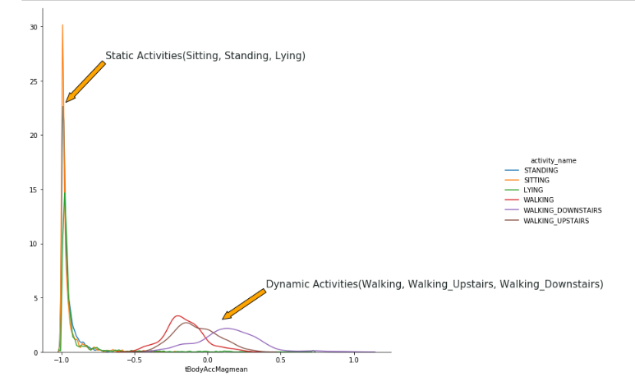
We also started out with basic visual exploration of the dataset by plotting distributions for the variables for each category, but given the large number involved, we dropped the idea. Though, in general there are two distinct major groups which we can see through the distributions. Some of the observations we have taken through these plots visualizations.

Below two count plots can infer that our classes are almost balanced.



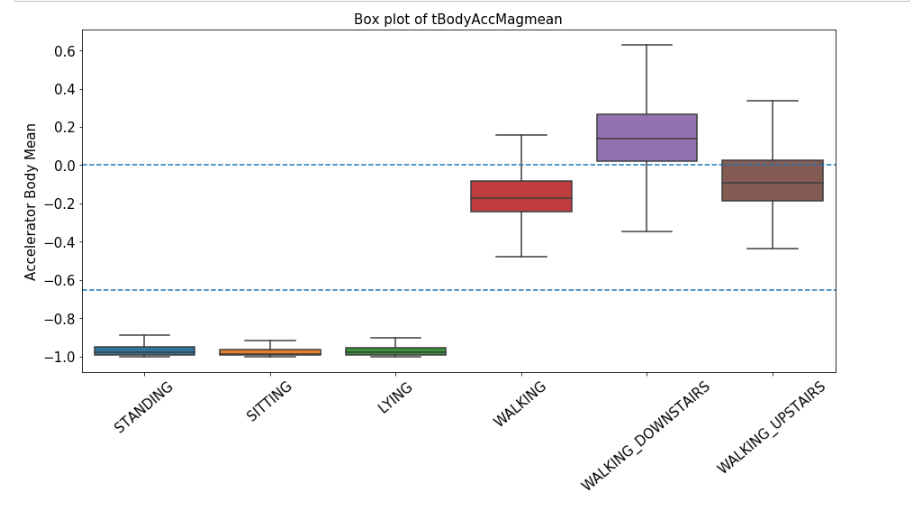


From facenet plot,we can clearly observe that how well "tBodyAccMagmean"--which is the magnitude of the mean of body acceleration in time-domain meaured by accelerometer--is able to separate static activity from dynamic activity. This shows that features are very carefully engineered.

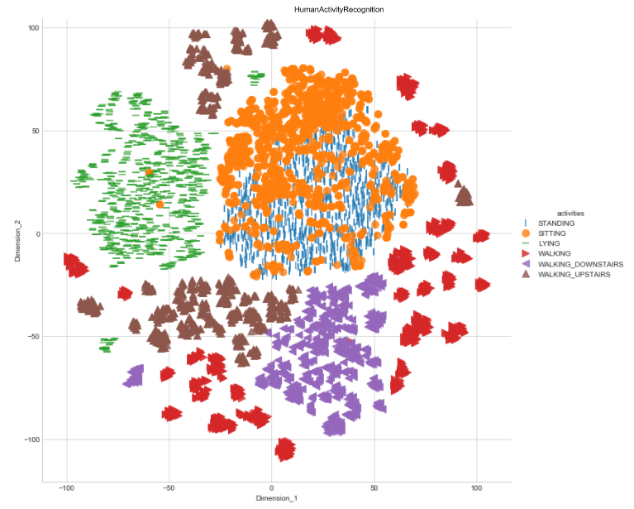


From Box plot,we observed that

* If tAccMean is < -0.8 then the Activities are either Standing or Sitting or Lying.
* If tAccMean is > -0.6 then the Activities are either Walking or Walking\_Downstairs or Walking\_Upstairs.
* If tAccMean > 0.0 then the Activity is Walking\_Downstairs.
* We can classify 75% the Acitivity labels with some errors.



From TSNE plots, we can observe that,except **STANDING** and **SITTING**, all other activities are separated fairly well.



**6.Methods:**

The first step was to create a train & test set. We split our data into two sets in 7:3 ratios by random sampling without replacement. This ensures that our train & test sets are representative of the complete dataset. Another approach to do it would be to do this sampling for each output class. In our case, the result wasn’t significantly different.

For modelling, we used the following techniques on our training set:

* Logistic Regression
* Linear SVM
* RBF SVM
* Decision Trees
* Random Forest
* Gradient Boosted Trees
* LSTM

To determine stability of the model being used, we use OOB score calculated during model building phase as representative of the validation set & optimized our model to increase this score. For determining true performance, we used a separate test set which was not included in any of our variable selection, model training or validation phases. A high accuracy on this independent test set is proof that the model is not overfitting our training data & hence, should generalize well.

We started out with all 561 variables & reduced the total features to 5 in our final model. The focus of our process was to follow algorithmic approach instead of a domain knowledge based model building process & hence we relied on oob score & variable importance to determine the optimal number of

features, trees to be used & which features to use.

**Step-by-step process**

1.Split data into two sets:

i)train(70%)

ii)test(30%)

2.In all the machine learning models, we have given hyper parameters from 0.001 to 10\*\*3.In each model,we got the best parameter value.

3.We have also set up cross validation to 3(cv=3) in all the algorithms we have used.

4.In Decision tree we have taken max-depth value 2 to 8 and in Random forest we have taken n-estimators [50,100,200,400,800]

5.In Gradient Boosted tree we have set up our estimators to [50,100] and max depth [1,3]

6.In deep learning model,we have used LSTM and set epochs to 8 and batch size to 32.

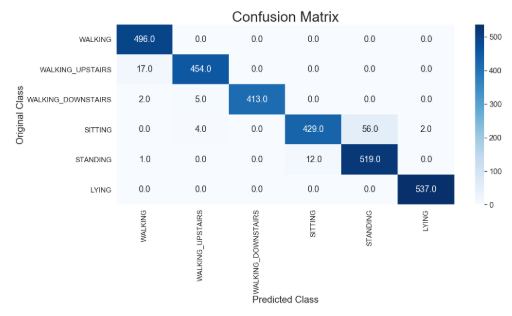
**Analysis and Results**

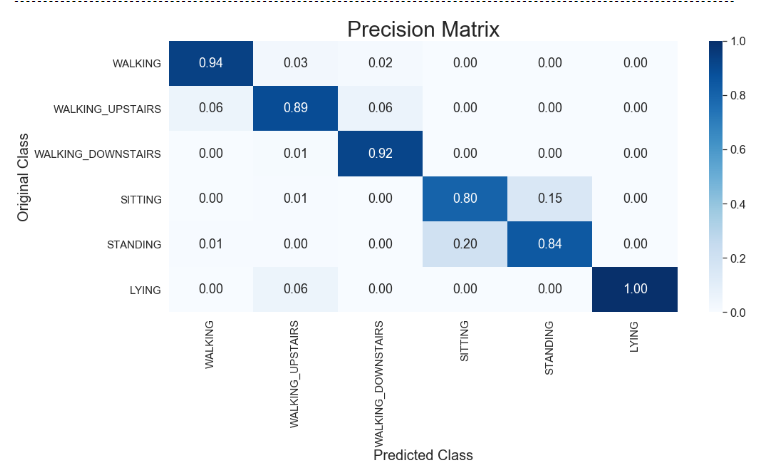
1.Important Features:

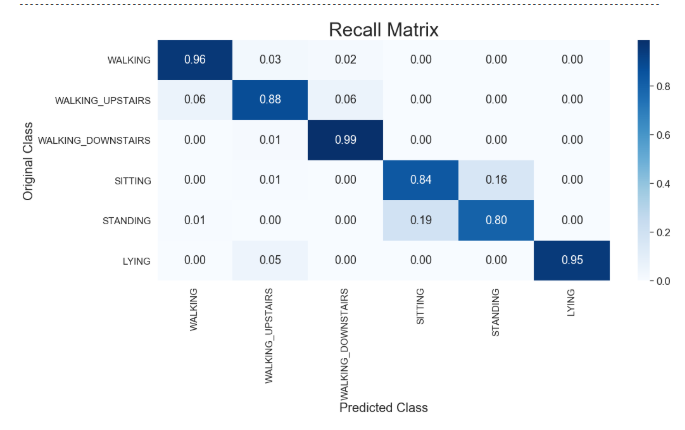
Using the previously described feature selection, we determined that the following features were important for building our classification model.

1. angle(X,gravityMean)
2. tGravityAcc-mean()-Y
3. tGravityAcc-min()-X
4. tGravityAcc-max()-X
5. tBodyAcc-mad()-X

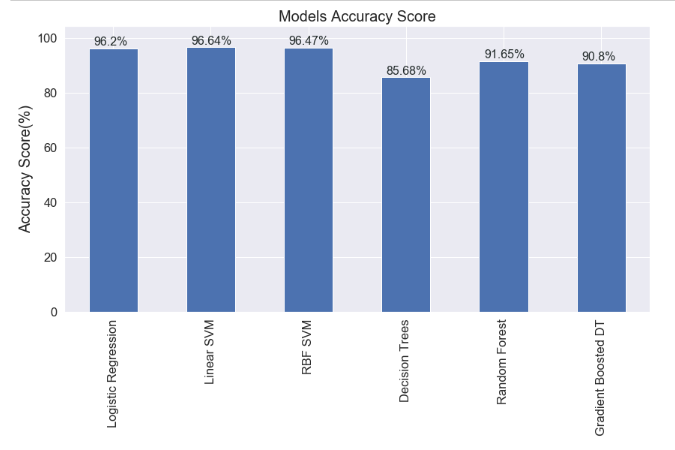
2. The final model of Linear SVM with confusion matrix, precision matrix and recall matrix.

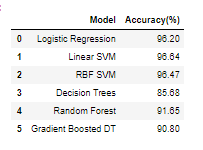






3. The final model with accuracy scores.





**Conclusion**

Overall, we relied heavily on Logistic Regression,Linear SVM,RBF SVM,Decision Trees,RandomForest,Gradient Boosted Trees and LSTM. By Simple two layered LSTM, we got a good accuracy of 89.82%. In short, Deep Learning help us to built models even when we don't have domain expert engineered features.

LSTM model can be further improved by running it for more epochs and more evaluations while tuning hyper-parameter.

**References**

**Human activity recognition**

[**https://machinelearningmastery.com/how-to-load-and-explore-a-standard-human-activity-recognition-problem/**](https://machinelearningmastery.com/how-to-load-and-explore-a-standard-human-activity-recognition-problem/)

**Logistic Regression**

**https://en.wikipedia.org/wiki/Logistic\_regression**

[**https://towardsdatascience.com/logistic-regression-detailed-overview-46c4da4303bc**](https://towardsdatascience.com/logistic-regression-detailed-overview-46c4da4303bc)

**SVM**

**https://en.wikipedia.org/wiki/Support\_vector\_machine**

**https://towardsdatascience.com/https-medium-com-pupalerushikesh-svm-f4b42800e989**

**RBF SVM**

**https://en.wikipedia.org/wiki/Radial\_basis\_function\_kernel**

**Decision Trees**

**https://towardsdatascience.com/decision-trees-in-machine-learning-641b9c4e8052%20%20%20%20%20%20%20**

[**https://en.wikipedia.org/wiki/Decision\_tree**](https://en.wikipedia.org/wiki/Decision_tree)

**Gradient Boosted Decision Trees**

**https://en.wikipedia.org/wiki/Gradient\_boosting%20%20%20%20%20%20**

**https://towardsdatascience.com/understanding-gradient-boosting-machines-9be756fe76ab**

**Human activity recognition with Deep learning**

**https://machinelearningmastery.com/deep-learning-models-for-human-activity-recognition/**

**https://towardsdatascience.com/human-activity-recognition-har-tutorial-with-keras-and-core-ml-part-1-8c05e365dfa0**

[**https://machinelearningmastery.com/how-to-load-and-explore-a-standard-human-activity-recognition-problem/**](https://machinelearningmastery.com/how-to-load-and-explore-a-standard-human-activity-recognition-problem/)

**Github Page**

<https://github.com/Kushi95/Human-activity-recognition>