

Classification of human activities

Deep Neural Network



November 24, 2021

Table of Contents

[Abstract 2](#_Toc88486225)

[Introduction 3](#_Toc88486226)

[Aim 3](#_Toc88486227)

[Problem Statement 3](#_Toc88486228)

[Social and Ethical Aspects 3](#_Toc88486229)

[Structure 4](#_Toc88486230)

[Method 5](#_Toc88486231)

[Dataset 5](#_Toc88486232)

[Data Exploration 5](#_Toc88486233)

[Correlation between all the Features 14](#_Toc88486234)

[Pre-processing 15](#_Toc88486235)

[Deep Neural Network 16](#_Toc88486236)

[Model Selection 17](#_Toc88486237)

[Training and Inference 17](#_Toc88486238)

[Analysis and Discussion 20](#_Toc88486239)

[Conclusions and Future Work 21](#_Toc88486240)

# Abstract

Human Activity Recognition is the problem of predicting what a person is doing based on a trace of their movement using sensors. It aims to understand human behavior and integrate users and their social context with computer systems.

In 2012, Davide Anguita, from the University of Genova, made available the dataset [A Public Domain Dataset for Human Activity Recognition Using Smartphones](https://upcommons.upc.edu/handle/2117/20897).”. A rich dataset, containing human body motion and their activities, it was collected based on the smartphones to gather context information about people actions.

In this project, we will predict the human activity based on 4 information collected by the sensors of the smartphone: Acceleration, Orientation, Marginal and Velocity.

With the advancements in Deep Learning, Deep Neural Networks (DNN) are becoming very increasingly effective on this type of task. Therefore, this report aims to investigate the viability of Deep Neural Networks using human motions data to predict human activity.

We were very successful on the first experiment to achieve high performance, with accuracy 1 on the validation dataset, we will present in the next parts the dataset used in the experiment, the pre-processing, and the successful model.

**Keywords:** Human Activity Recognition, Deep Learning, Prediction of human activity, Sensors.

# Introduction

## Aim

This project aims to study and investigate the applicability of Deep Neural Networks to predict human activity based on human motions, more precisely the features that has been used are Acceleration, orientation, marginal and velocity.

## Problem Statement

To solve the task of human motion recognition, there are a lot of methods that can be successful to do so: like video processing, and the previous machine learning models like decision tree.

The available dataset that will conduct the following study contains information about Acceleration, orientation, marginal, and velocity.

* **Which DNN is superior in the context of predicting human motions?**

## Social and Ethical Aspects

The dataset is completely based on the perception of the individual (the subject of the study) and the environment. Therefore, it is very hard to have a general model that works for everyone. For example, the human motion of an injured person will be completely different than a normal person. Another example will be the difference based on gender; women have q center gravity completely different than men.

There is a lot of factors that play key role in this aspect like: age, gender, normal/injured….

This project can have more humanistic and positive applications, in our world, like for example the democratization of the analysis of physical efforts, so everyone can analyze his movements during a workout and his performance without spending too much money on medical tests.

## Structure

This report is divided into 3 Chapters. Chapter 2: “Method” discusses the details regarding the dataset along with parameter selection for the model. Chapter 3 will present different results obtained during the experiment. Chapter 4 will analyze the previous results. And finally, the conclusion and the future work.

# Method

## Dataset

### Data Exploration

The dataset contains 5 files:

* **Acc.csv**
* **Labels.csv**
* **Orin.csv**
* **Mag.csv**
* **Velo.csv**

For all the files, we transformed the timestamp to seconds, and we took the first time as our reference '2019-10-11 19:05:56.381000'.

##### Acc.csv

* Columns of the file

Table

Description automatically generated

* Data type of the columns of the file

Table

Description automatically generated with medium confidence

* Missing values

Table

Description automatically generated with medium confidence

* Plot 2d of the values Acc\_X, Acc\_Y and Acc\_Z of over time

Chart

Description automatically generated

* Plot 3d scatter cloud points

Chart, scatter chart

Description automatically generated

##### Velocity.csv

* Columns of the dataset

Table

Description automatically generated

* Datatype of the dataset

Table

Description automatically generated with medium confidence

* Missing values

Text

Description automatically generated

* Plot 2d of the values AngV\_X, AngV\_Y and AngV\_Z of over time

Chart

Description automatically generated

* Plot 3d scatter cloud points

Chart, scatter chart

Description automatically generated

##### Orin.csv

* Columns of the dataset

Table

Description automatically generated

* Datatype of the dataset

A picture containing text

Description automatically generated

* Missing Values

Graphical user interface, text, application

Description automatically generated

* Plot 2d of the values Orin\_X, Orin\_Y and Orin\_Z of over time

Diagram

Description automatically generated

* Plot 3d scatter cloud points

Chart

Description automatically generated

##### Mag.csv

* Columns of the dataset

Table

Description automatically generated

* Datatype of the dataset

Table

Description automatically generated with medium confidence

* Missing Values

Text

Description automatically generated

* Plot 2d of the values Orin\_X, Orin\_Y and Orin\_Z of over time

Chart, histogram

Description automatically generated

* Plot 3d scatter cloud points

Chart, scatter chart

Description automatically generated

##### Final Table

After doing a left join to all tables on Timestamp, we obtain the final table.

The timestamp column corresponds to the transformation of the timestamp to seconds, using the first time as a reference '2019-10-11 19:05:56.381000'.

Table

Description automatically generated

### Correlation between all the Features

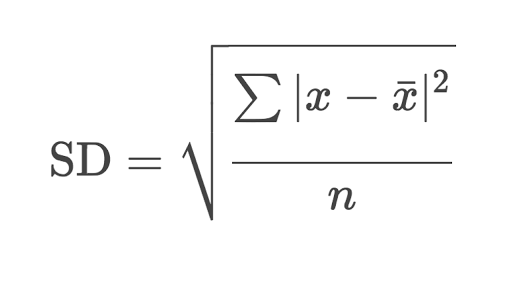
Chart, treemap chart

Description automatically generated

### Pre-processing

#### Standardization

We transform the 12 columns [ 'Acc\_X', 'Acc\_Y', 'Acc\_Z', 'AngV\_X', 'AngV\_Y', 'AngV\_Z', 'Mag\_X', 'Mag\_Y', 'Mag\_Z', 'Orin\_X', 'Orin\_Y', 'Orin\_Z' ] using the standardization technique.



#### Label Encoding

As an instruction on the beginning of this project, we have to choose only 4 activities, our choice was : ['Walking', 'Vibration','Standing','Sitting'].

We did a Label Encoding to the column Activity on the final table, we assigned:

**Sitting : 0**

**Standing : 1**

**Vibration : 2**

**Walking : 3**

The final table becomes:

A picture containing table

Description automatically generated

#### Split Train/ Test / Validation

We split the dataset into Train/ Test/ Validation with the proportion 60/20/20.

## Deep Neural Network

Deep Neural Network (DNN) is a collection of connected processing units known as neurons, which are modeled to reproduce the flexibility and power of a biological brain. A neuron is a generalization of the idea of perceptron, which was developed by Frank Rosenblatt. A perceptron takes multiple binary inputs to produce a single binary output. To compute the output, the weights are introduced to represent the significance of the input a perceptron.

A neuron produces a graded value between 0 and 1. This graded value at the output allows the network to learn the weights with the help of backpropagation. Further, the output values are biased to generate extreme values towards 0 or 1. A DNN is a complex network of these neurons which take various input values to generate decisions that has the potential to mimic human thinking. Since its creation, DNN has been widely used to solve complex tasks.

### Model Selection

Diagram

Description automatically generatedFor our model, we choose the following architecture:

1. In the beginning, a Dense Layer with 100 hidden units.
2. Followed by a DropOut Layer with 50% activation, for regularization.
3. Followed by a Dense Layer with 100 hidden units.
4. Followed by the final Dense with 4 hidden units, and softmax activation.
5. The model takes as input the 12 features [ 'Acc\_X', 'Acc\_Y', 'Acc\_Z', 'AngV\_X', 'AngV\_Y', 'AngV\_Z', 'Mag\_X', 'Mag\_Y', 'Mag\_Z', 'Orin\_X', 'Orin\_Y', 'Orin\_Z' ] and predict the probability of one of the four human activity ['Walking', 'Vibration','Standing','Sitting'].
6. Table

   Description automatically generated

### Training and Inference

We choose as initial parameters for parameters:

* Batch size = 32
* Epochs = 128
* Early Stopping in call back section: we stop the training of the model after 3 iterations with no improvements.

We compile the model with sparse categorical cross-entropy function.

Then, we launch the training.

#### Loss Function

Chart, line chart

Description automatically generated

#### A picture containing graphical user interface Description automatically generatedAccuracy

#### Evaluation on Validation set

##### Loss and accuracy

##### 

##### Graphical user interface, application Description automatically generatedConfusion Matrix

#### 

# Analysis and Discussion

* The accuracy of the model is 1, on train set, test set and validation set. We were successful to obtain this result without the need to launch a lot of experiments to find the best hyperparameters for the model.
* The model achieved the aim goal. Based on the information of the sensors, our model can predict the activity of the body.
* The accuracy and loss function are performing a little bit better on the test set compared to the train set. This is completely normal behavior of the layer Dropout, that has the functionality to disactivate a portion of the number of neurons and activate them on the inference, which makes the model much smoother and more performing.

# Conclusions and Future Work

To conclude, I gained a lot of experiences doing this project, I have got more familiar with Matlab Mobile, and I can use it to analyze my physical performances. Also, I have gained more knowledge of the framework Tensorflow and Keras, and it opened my sights to be able to see what can be done in the field of AI. Because, it will takes too much time, to program everything from the theoretical perspective.