Human Activity Recognition using Smartphone Dataset

Bill Morrison Talla
Computer Science Department
Bishops's University
Sherbrooke, Canada
btalla18@ubishops.ca

Kadidia Sow Diallo
Computer Science Department
Bishop's University
Sherbrooke, Canada
Ksow18@ubishop's.ca

Philippe Alphe Atchori
Computer Science Department
Bishops's University
Sherbrooke, Canada
patchori18@ubishops.ca

***Abstract*—Over the years Deep Learning has kept increasing in interest in the IT world. Deep Learning is used anywhere from automating mundane tasks to offering intelligent insights, industries in every sector try to benefit from it. Human activity recognition is an important area of deep learning research as it has much utilization in different areas such as sports training, security, entertainment, ambient assisted living and many more. .**

Keywords*— *(Deep Learning, CNN algorithm, linear regression, sklearn, pre-processing, keras, research topics)

1. Introduction

Human activity recognition is a broad field of study concerned with identifying the specific movement or action of a person based on sensor data. Movements are often typical activities performed indoors, such as walking, talking, standing, and sitting. They may also be more focused activities such as those types of activities performed in a kitchen or on a factory floor. The sensor data may be remotely recorded, such as video, radar, or other wireless methods. Alternately, data may be recorded directly on the subject such as by carrying custom hardware or smart phones that have accelerometers and gyroscopes. To be more specific, we will train a convolutional neural network (CNN) to recognize the type of movement (Walking, Running, Jogging, etc.) based on a given set of accelerometer data from a mobile device carried around a person's waist. We will use a WISDM data set for our project.

.

2. Previous research works

a) Human Activity Recognition and Monitoring Using Smartphones

A group of students did implement an application to help people with their daily routine and motivating them to be more active. They used deep learning algorithms in their approach and were able to achieve 75 % for the ANN respectively on the overall five activities done by those users. The classification algorithms with artificial neural network (ANN) were applied to recognize user's activities.

b) HUMAN ACTIVITY RECOGNITION USING INERTIAL SENSORS IN A SMARTPHONE

Another group of students did a research on the human activity recognition with the help of sensors in a smartphone in order to guide and help developers to achieve a great work in that area. Their research proved to be helpful for many researchers and developers. They have presented a set of challenges and future research opportunities in the area of smartphone based HAR.

Bishop's University, UCI dataset, Computer science department.

3. Tools and Methods Used

The application was implemented using the following tools and methods:

1. **Python programming language**: Python is an interpreted, object-oriented, high-level programming language with dynamic semantics.

Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together

2. **Jupyter Notebook or Google Colab**: The Jupyter Notebook is an incredibly powerful tool for interactively developing and presenting data science projects. while Google Colab is a free cloud service, based on Jupyter Notebooks for machine-learning education and research. It provides a runtime fully configured for deep learning and free-of-charge access to a robust GPU.

3. **CNN:** one of the simplest and yet most useful Machine Learning structures. They can be used to solve both regression and classification problems.

Deep Learning: Deep learning is a set of automatic learning methods attempting to model with a high level of abstraction of data through articulated architectures of different nonlinear transformations

4. DATA SOURCE INFORMATIONS

The "WISDM Smartphone and Smartwatch Activity and Biometrics Dataset includes data collected from 51 subjects, each of whom were asked to perform 18 tasks for 3 minutes each. Each subject had a smartwatch placed on his/her dominant hand and a smartphone in their pocket. The data collection was controlled by a custom-made app that ran on the smartphone and smartwatch. The sensor data that was collected was from the accelerometer and gyroscope on both the smartphone and smartwatch, yielding four total sensors. The sensor data was collected at a rate of 20 Hz (i.e., every 50ms).

The dataset included the following files:

- Activitykey.txt – contain labels of activities performed identified by codes

TABLE 2
THE 18 ACTIVITIES REPRESENTED IN DATA SET

Activity	Code
Walking	A
Jogging	B
Stairs	C
Sitting	D
Standing	E
Typing	F
Brushing Teeth	G
Eating Soup	H
Eating Chips	I
Eating Pasta	J
Drinking from Cup	K
Eating Sandwich	L
Kicking (Soccer Ball)	M
Playing Catch w/Tennis Ball	O
Dribblinlg (Basketball)	P
Writing	Q
Clapping	R
Folding Clothes	S

- README.txt
- RAW folder containing Datasets from both smartphones and smartwatches

TABLE 4
DISTRIBUTION OF RAW SENSOR DATA

Activity	Phone		Watch		Total	Class %
	Accel	Gyro	Accel	Gyro		
Walking	279,817	203,919	210,495	192,531	886,762	5.7%
Jogging	268,409	200,252	205,787	187,833	862,281	5.5%
Stairs	255,645	197,857	207,312	180,416	841,230	5.4%
Sitting	264,592	202,370	213,018	195,050	875,030	5.6%
Standing	269,604	202,351	216,529	194,103	882,587	5.6%
Typing	246,356	194,540	205,137	187,175	833,208	5.3%
Brush Teeth	269,609	202,622	208,720	190,759	871,710	5.6%
Eat Soup	270,756	202,408	209,483	187,057	869,704	5.6%
Eat Chips	261,360	197,905	210,048	192,085	861,398	5.5%
Eat Pasta	249,793	197,844	203,112	189,609	840,358	5.4%
Drinking	285,190	202,395	215,879	197,917	901,381	5.8%
Eat Sandwich	265,781	197,915	203,684	190,191	857,571	5.5%
Kicking	278,766	202,625	209,491	191,535	882,417	5.6%
Catch	272,219	198,756	210,107	187,684	868,766	5.6%
Dribblinlg	272,730	202,331	212,810	194,845	882,716	5.6%
Writing	260,497	197,894	215,365	197,403	871,159	5.6%
Clapping	268,065	202,330	208,734	190,776	869,905	5.6%
Fold Clothes	265,214	202,321	211,335	193,373	872,243	5.6%
Total	4,804,403	3,608,635	3,777,046	3,440,342	15,630,426	100%

- WISDM.pdf providing more information on the dataset
- Arff_files folder another version of the raw dataset containing datasets from both smartphones and smartwatches
- Arffmagic-master another folder containing files.

Summary of the dataset

SUMMARY INFORMATION FOR THE DATASETS

Number of subjects	51
Number of activities	18
Minutes collected per activity	3
Sensor polling rate	20Hz
Smartphone used	Google Nexus 5/5x or Samsung Galaxy S5
Smartwatch used	LG G Watch
Number raw measurements	15,630,426

DEFINITION OF ELEMENTS IN RAW DATA MEASUREMENTS

Field name	Description
Subject-id	*Type: Symbolic numeric identifier.* Uniquely identifyies the subject. Range: 1600-1650.
Activity code	*Type: Symbolic single letter.* Identifies a specific activity as listed in Table 2. Range: A-S (no "N" value)
Timestamp	*Type: Integer.* Linux time
x	*Type Numeric: real.* Sensor value for x axis. May be positive or negative.
y	Same as x but for y axis
z	Same as x but for z axis

5. OUR APPROACH/ SOLUTION

The Human Activity recognition (HAR) is an important aspects of today's modern trends in the IT industry. The goal of this project is to build a classification model that can precisely identify human fitness activities. In order to do that we had to design an application with the help of deep learning algorithms to perform on the dataset and help resolve and classify the activities. Here are the steps by steps procedure we did during the implementation of the project.

a. Step 1: Load accelerometer data from the WISDM data set

b. Step 2: Convert and reformat accelerometer data into a time-sliced representation

c. Step 3: Visualize the accelerometer data

d. Step 4: Reshape the multi-dimensional tabular data so that it is accepted by Keras

e. Step 5: Split up the data set into training, validation, and test set

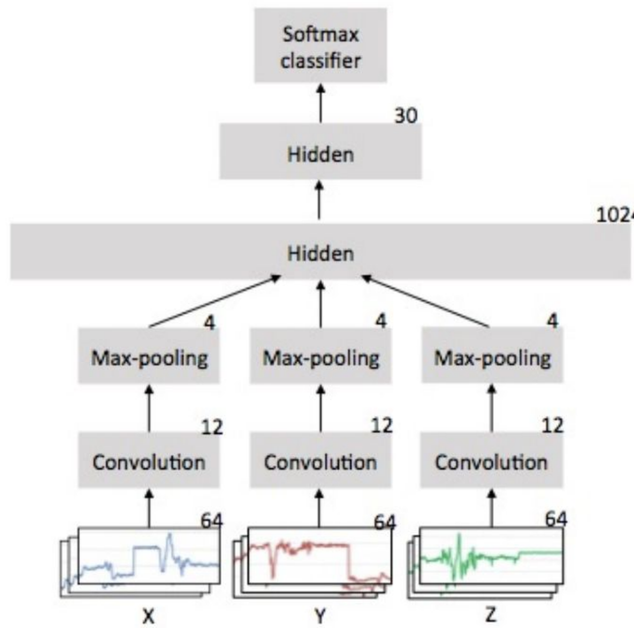
f. Step 6: Define a convolutional neural network model in Keras

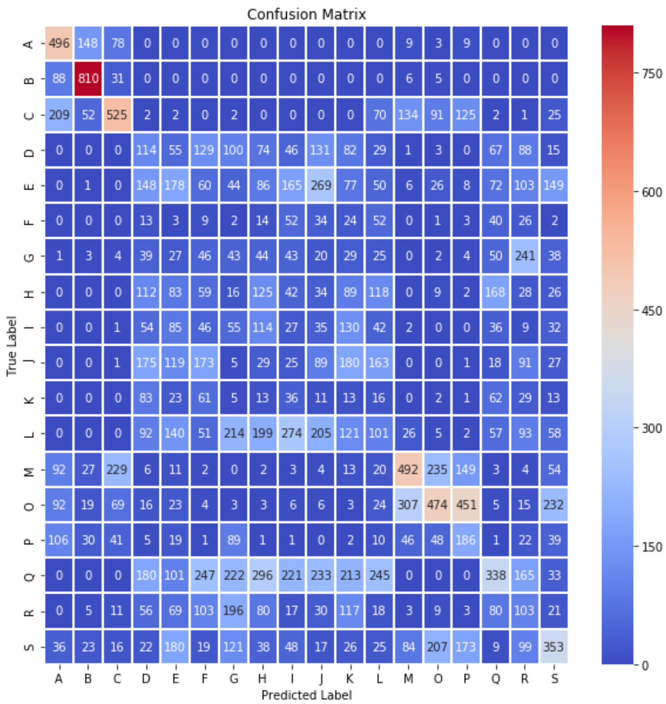
g. Step 7 Train the convolutional neural network for human activity recognition data

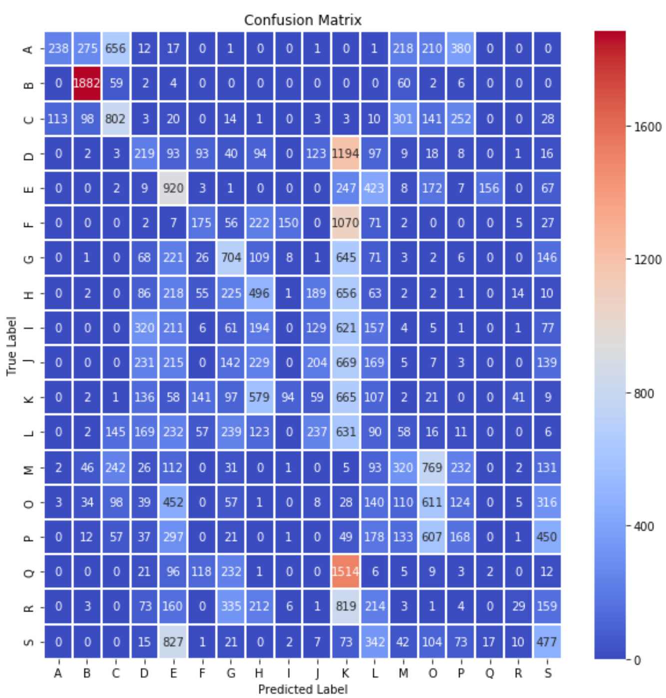
h. Step 8: Validate the performance of the trained CNN against the test data using learning curve and confusion matrix

i. Step 9: Export the trained Keras CNN.

j. Step 10: Create a playground in Xcode and import the already trained Keras model







Then the confusion matrix is displayed for both dataset

A. *Development phases*

In the development phase, we faced many problems such as choosing which algorithm to implement. All the algorithms looked essentials to us, but we had to decide on which one to select.

Selection of the environment: we had to choose the best python editor that was going to help us achieve our goal efficiently.

Group Meetup we had to plan and agree on what time to meet and work as a group.

Debugging the code: we had multiple errors where we had to used google to find solutions to those errors.

6. Algorithm Selected

In this challenge we had to choose any algorithms that will work perfectly on the dataset for the classification and produce the best classification and accuracy as well as the correct prediction. Here are the list algorithms we could choose from below:

Algorithms
Deep Feedforward Networks
Convolutional Neural Networks
Sequence Modeling: Recurrent and Recursive Networks
Dense prediction
Generative adversarial networks
Deep Q-learning

We decided to select Convolutional Neural Network CNN first to do this project. Convolutional Neural Network models (CNN) are a type of deep neural network.

They have proven very effective on challenging computer vision problems when trained at scale for tasks such as identifying and localizing objects in images and automatically describing the content of images.

They are models that are comprised of two main types of elements: convolutional layers and pooling layers.

Convolutional layers read an input, a 1D signal, using a kernel that reads in small segments at a time and steps across the entire input field. Each read results in an input that is projected onto a filter map and represents an internal interpretation of the input.

When we applied CNNs to human activity recognition data,

the CNN model learns to map a given window of signal data to an activity where the model reads across each window of data and prepares an internal representation of the window.

Pooling layers take the feature map projections and distill them to the most essential elements, such as using a signal averaging or signal maximizing process.

The convolution and pooling layers can be repeated at depth, providing multiple layers of abstraction of the input signals.

The output of these networks is often one or more fully connected layers that interpret what has been read and map this internal representation to a class value.

7. Compare with other Algorithms

When applied to time series classification like human activity recognition, RNN is better suited to for the classification because we are dealing with sequential datasets. It has the highest classification because it is better suited for the dataset provided compared to other techniques.

8. Group members participation

Each Members took part equally in the project development here are the group members participation:.

Members	Participation
Bill	CNN, RNN and DFN algorithms, debugging
Kadidia	Testing, Data visualization and project report
Philippe	Loading of the dataset and cleaning, Splitting of the datasets

9. Conclusion and future work

In conclusion we can clearly notice that the RNN algorithm works well with the temporal datasets providing the best possible accuracy of 90% and 92% respectively on both accel and gyro data.

We plan next time using different algorithms to compare with the decision tree algorithm. While doing that it will help in a true comparison between all of the deep learning model algorithms

10. Acknowledgment

A huge thanks to the Professor Ayoub for teaching an important trending aspect of the computer science which is the Deep Learning. With the knowledge acquired in this course we can collaborate in the AI world and computer science world at large.

11. References

[1] Alsheikh, M.A., Selim, A., Niyato, D., Doyle, L., Lin, S., Tan, H.P.: Deep activity recognition models with triaxial accelerometers. CoRR abs/1511.04664 (2015)

[2] Burns, A., Greene, B.R., McGrath, M.J., O'Shea, T.J., Kuris, B., Ayer, S.M., Stroiescu, F., Cionca, V.: ShimmerTM a wireless sensor platform for noninvasive biomedical research. IEEE Sensors Journal 10(9), 1527 – 1534 (2010).

[3] Zeng, M., Nguyen, L.T., Yu, B., Mengshoel, O.J., Zhu, J., Wu, P., Zhang, J.:Convolutional neural networks for human activity recognition using mobile sensors.In: 6th International Conference on Mobile Computing, Applications and Services.pp. 197–205 (Nov 2014).

[4] Wang, J., Chen, Y., Hao, S., Peng, X., Hu, L.: Deep learning for sensor-based activity recognition: A survey. CoRR abs/1707.03502 (2017).

[5] Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. CoRR abs/1412.6980 (2014).

[6] Robertas Lamaseries, Mindaugas Vasiljevic, Justas SalkeviIius, Marcin Wofniak, Human Activity Recognition in AAL Environments Using Random Projections, Computational and Mathematical Methods in Medicine, Volume 2016, Article ID 4073584, 17 pages

[7] Lichman, M. (2013). UCI Machine Learning Repository Irvine, CA: University of California, School of Information and Computer Science.

[8] Jose Daniel Pereira Ribeiro Filho, Francisco Jose da Silva eSilva, Luciano Reis Coutinho, Berto de Tácio Pereira Gomes, Markus Endler, A Movement Activity Recognition Pervasive System for Patient Monitoring in Ambient Assisted Living, SAC 2016, April 04 - 08, 2016, Pisa, Italy

[9] Cook, D., Feuz, K.D., Krishnan, N.C.: Transfer learning for activity recogni-tion: a survey. Knowledge and Information Systems 36(3), 537–556 (Sep 2013).

[10] Godfrey, A., Conway, R., Meagher, D., Laighin, G.: Direct measurement of human movement by accelerometry 30, 1364–86 (01 2009)

.