

Assignment 3: Activity Recognition (20P)

Hand in in groups of two.

Goals

You are familiar with the basic principles of machine learning. You know about the advantages and disadvantages of machine learning in the field of human-computer interaction. You can implement a simple machine learning classifier to detect activities based on real-time sensor data.

1 Read Up on Machine Learning 5P

In GRIPS, you can find the papers “Avoiding Pitfalls When Using Machine Learning in HCI Studies” (Kostakos et al., 2017) and “Machine Learning: The High-Interest Credit Card of Technical Debt” (Sculley et al., 2014). Split the papers among your team and read them. Read appropriate further literature as necessary.

Discuss the following questions with your partner and write down your findings briefly (about one page per team).

- What are some common use cases for machine learning in practical applications or research prototypes?
- Which problems of machine learning do the authors of the papers identify?
- What are the credentials of the authors with regard to machine learning? Have they published research on machine learning (or using machine-learning techniques) previously?

Score

(2P) Good answer to the first question.

(2P) Good answer to the second question.

(1P) Good answer to the third question.

2 Gathering Training Data 6P

Recommended Packages: DIPPID, pandas

Create a program called *gather-data.py* which captures sensor data (e.g. accelerometer, gyroscope) from the DIPPID device and saves it to an appropriately named and structured CSV file. This sensor data will be used to train a machine learning classifier later. It is recommended to include labels and timestamps in the log file. Furthermore, it should be possible to use the DIPPID device to start capturing data and stop capturing when a fixed time has passed or a certain amount of data has been gathered.

Select at least three types of activities (e.g. waving, shaking, standing, lying, running, jumping, punching, ...). Record at least five data sets per team member for each of your selected activities.

Score

- (2P) data is logged correctly
- (2P) log files are named and structured appropriately
- (1P) logging can be started with the DIPPID device
- (1P) enough data sets captured

3 Activity Recognition

9P

Recommended Packages: *pandas, numpy, pyglet, sklearn*

Create a program called *activity-recognizer.py* which predicts a user's activity based on sensor data from a DIPPID device. When the program is started, it should read training data from the CSV files with your training data and train a machine learning classifier. Most certainly, it is required to pre-process training data before training (e.g. filtering, normalization, transformation into frequency domain, ...).

After the training is finished, the program should predict activities based on sensor data from the DIPPID device. You can make the user press a button to start prediction, but you get an extra point if prediction runs continuously without requiring further intervention by the user.

Visualize recognized activities nicely using pyglet.

Score

- (1P) the program correctly loads training data
- (2P) training data is pre-processed appropriately
- (1P) a classifier is trained with this training data when the program is started
- (3P) the classifier predicts activities correctly
- (1P) prediction works continuously without requiring intervention by the user
- (1P) nice visualization

Code Quality: bad code quality and missing documentation can lead to loss of points.