Efficient Activity Recognition for Individual Fitness Tracking

Name: Bharath Kumar Avusherla and Sagar Sharma

UID: U00790636, U00774498

Goals

Fitness devices have become a common trend amongst health conscious individuals. The implications of data collected by such devices are of interest to not only the individual users but health providers, employers, and insurance companies. Many applications have been designed for tracking daily activities and overall health monitoring. Being able to identify users' activities to assess and predict an individual's overall health or fitness status requires complex feature extraction and Machine Learning algorithms. The ultimate goal of these algorithms is to produce a personalized health or fitness report to an individual, and health practitioners with information such as active minutes and calories burned. This project will look at few successful machine learning algorithms in the area of activity recognition and give a comparative analysis. It will delve into the complexity involved, performance, and accuracy for each algorithm using established performance measuring metrics.

Objective

- Demonstrate and apply data processing techniques to extract relevant features from an activity tracking device.
- Keep into consideration the notions of training, testing, and cross-validation while designing the activity recognition model.
- Efficiently classify data collected by fitness tracker into several activities such as walking, running, and standing using the models we construct.
- Compare advantages and disadvantages of different machine learning algorithms in terms of complexity, performance, and accuracy ,and possibly look into methods of optimization for better results.

Approach

- Apply various signal processing techniques such as SMA (Signal Magnitude Area) to extract the features from the tracking device.
- Use different supervised machine learning techniques such as Random Forest and K-Nearest Neighbors algorithms to classify the activities.
- Experiment with different proportions of training, testing and validation data to optimize the model.
- Measure the implementations' performance using metrics such as F-measure, Confusion matrix, and Area Under Curve.

- **Technology :** Implementation in Python or MATLAB. Use preexisting libraries in Weka tool.
- **Data :** UCI Machine Learning Repository: Activity Recognition from single Chest-Mounted Accelerometer. https://archive.ics.uci.edu/ml/datasets/Activity+Recognition+from+Single+Chest-Mounted+Accelerometer

References

Carus, Juan Luis, Victor Palaez, Gloria Lopez, and Vanesa Lobato. "Result Filters." *National Center for Biotechnology Information*. U.S. National Library of Medicine, n.d. Web. 26 Oct. 2015.

Casale, Pierluigi, Oriol Pujol, and Petia Radeva. "Human Activity Recognition from Accelerometer Data Using a Wearable Device." *ResearchGate*. N.p., n.d. Web. 26 Oct. 2015.

Karantonis, Dean M., and Nigel H. Lovell. "Implementation of a Real-time Human Movement Classifier Using a Triaxial Accelerometer for Ambulatory Monitoring." *IEEE Xplore*. N.p., n.d. Web. 26 Oct. 2015.

Li, Aiguang, Lianying Ji, Shaofeng Wang, and Jiankang Wu. "Physical Activity Classification Using a Single Triaxial Accelerometer Based on HMM." *IEEE Xplore*. N.p., n.d. Web. 26 Oct. 2015.

Meyer, Francois G. "Physical Activity Classification Using a Single Triaxial Accelerometer Based on HMM." *IEEE Xplore*. N.p., n.d. Web. 26 Oct. 2015.
Penhaker, Marek, Petr Gajdos, and Pavel Dohnalek. "Comparison of Classification Algorithms for Physical Activity Recognit." *Ion*. N.p., n.d. Web. 26 Oct. 2015.

Tapia, Emanuel Munguia, Stephen S. Intille, and Kent Larson. "Multiple People Activity Recognition Using Simple Sensors." *Proceedings of the 1st International Conference on Pervasive and Embedded Computing and Communication Systems* (2011): n. pag. *Http://web.media.mit.edu/~intille/papers-files/TapiaIntilleLarson04.pdf*. Web.

Appendix:

Schedule and Milestones

9th week: Proposal submission, data interpretation, and understand feature extraction. Begin working on the report in a parallel manner.

10th week: Identify potential Machine Learning algorithms, start implementation

11th week: Complete the implementation. Test and cross-validation.

12th week: Performance Metrics, and Fine Tuning

13th week: Complete the report, Fix the problems if any

14th week: Presentation