Machine Learning Engineer Nanodegree

Capstone Proposal

by Sarthak Dargan
November 8, 2017

Proposal

Domain Background

The average office worker sits about 10 hours a day. There are all those hours in front of the computer, plowing through emails, making calls or writing proposals and eating lunch. And then there are hours sitting in front of the TV or surfing the web at home. Medical researchers have long warned that **prolonged sitting is dangerous**, associated with a significantly higher risk of heart disease, diabetes, obesity, cancer and depression, as well as muscle and joint problems.

I recently joined an IT firm and I could really see myself falling into this trap directing into significantly higher risk of diseases, this motivated me to actually work on this matter.

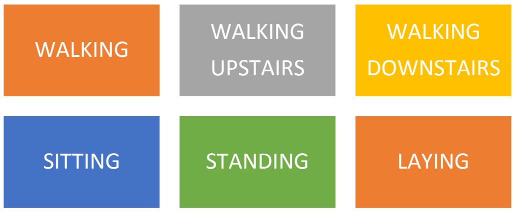
The aim of this project is to build a use case that take sensor information from our mobile phones and use that information to recognized and classify human activity like sitting running etc. and to analyze that information to give productive notifications, suggestions and recommendations.

Problem Statement

The Human Activity Recognition database was built from the recordings of 30 study participants performing activities of daily living (ADL) while carrying a waist-mounted smartphone with embedded inertial sensors.

The goal is to predict the likelihood that a human activity is from a certain class from the provided classes, thus making it a multi-class classification problem in machine learning terms and thereby analyzing it.

Six target classes are provided in this dataset:



The goal is to train our model to classify human activity into these six classes and give constructive feedback, notifications and analysis day routine.

Datasets and Inputs:

The experiments have been carried out with a group of 30 volunteers within an age bracket of 19-48 years. Each person performed six activities wearing a smartphone (Samsung Galaxy S II) on the waist. Using its embedded accelerometer and gyroscope, we captured 3-axial linear acceleration and 3-axial angular velocity at a constant rate of 50Hz. 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.

For each record in the dataset the following is provided:

- Triaxial acceleration from the accelerometer (total acceleration).
- Triaxial Angular velocity from the gyroscope.
- A 561-feature vector with time and frequency domain variables.
- Its activity label.
- An identifier of the subject who carried out the experiment.

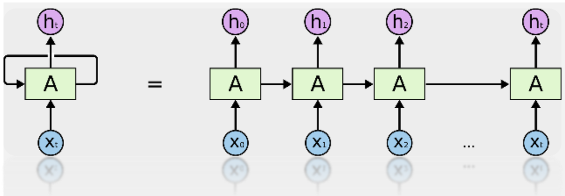
Solution Statement

I will be using a Long Short Term Memory network a recurrent neural network on the data to learn and to recognize the type of activity that the user is doing**.** I would be using many to one RNN architecture taking series of input vectors and output vectors with probabilities to help in classification or recognizing as one of the 6 classes.

Benchmark Model

Recurrent Neural Networks -

When we read some text we understand the text based on the previous words, our thought has persistence, normal Neural Networks cannot do this; so there we come with RNN, they are the networks with loop in them.



A recurrent neural network can be thought of as multiple copies of the same network, each passing a message to a successor

Unfortunately, as that gap grows, RNNs become unable to learn to connect the information. Hochreiter and Bengio found out some fundamental reasons why it might be difficult resulting in LSTM.

Long Short Term Memory network

LSTM is a special kind of RNN designed to solve long-term dependency problem. LSTM also have repeating module but with different architecture instead of having a single neural network layer, there are four, interacting in a very special way.

Evaluation Metrics

These evaluation metrics is to be considered while evaluating the final model.

- Accuracy
- Precision
- Confusion Matrix
- F1_score
- Recall

Project Design

- **Programming language**: Python 2.7+
- **Libraries :** Keras, Tensorflow, Scikit-learn, Opencv
- **Workflow:**
  - Download the dataset and segment in training and testing datasets.
  - Study the datasets and configure additional parameters.
  - Configure and Add a LSTM (RNN) artificial neural stacking LSTM cells which helps in adding depth to neural network.
  - Build the neural network.
  - Train neural network
  - Plot insights graphs for better visualization on training dataset.
  - Evaluate model on test datasets.

    - Use Evaluation matrix and confusion matrix model credibility.
    - Plot insights graphs for better visualization on test dataset.

References:

- Dataset Link -The dataset can be found on the UCI Machine Learning Repository.
- Kaggle Link - https://www.kaggle.com/uciml/human-activity-recognition-with-smartphones