Detection and Prediction of KeyStrokes Through SmartWatch Motion Sensors

-Anuj Kumar, 30517967

This project will be a solo undertaking by me. This project is also not shared with any other class, and will be started anew. Hence, no work on it has been done previously, with very few exceptions.

The problem being investigated is if we can classify and predict keystrokes using the sensor (accelerometer and gyroscope, but other sensors may be explored) data from smartwatches, and use it to recover what a user is typing. The (ambitious) aim of this project is to learn the "hand model" of a single person and train a robust classifier such that it can predict what a user intends to type even if he is typing on an imaginary keyboard(wooden desk). I plan to use two smartwatches, one worn on each wrist, to try and achieve this.

Previous related work on this can be found in the following:

- Anindya Maiti, Oscar Armbruster, Murtuza Jadliwala, and Jibo He, "Smartwatch-Based Keystroke Inference Attacks and Context-Aware Protection Mechanisms", *ASIA CCS ’16, May 30-June 03, 2016*
- He Wang, Ted Tsung-Te Lai, Romit Roy Choudhury, "MoLe: Motion Leaks through Smartwatch Sensors", *2015 ACM*
- Allen Sarkisyan, Ryan Debbiny, Ani Nahapetian, "WristSnoop: Smartphone PINs Prediction Using Smartwatch Motion Sensors", *2015 IEEE*
- Tony Beltramelli, "Deep-Spying: Spying using Smartwatch and Deep Learning", *Master Thesis 2015*

In all of the works above, the experimentation is on security issues arising from analysis on these sensors. For example, one paper combines sensor data of keystrokes and the eavesdropped noise of keystrokes to build a high accuracy model of what is being typed. The other papers try predicting whole words using just one smartwatch data and depict security issues such as accurate prediction of phone PINs or even whole words(with restrictions and not-too-high accuracy). I wish to go a step further and predict at the level of the keystrokes, so that a user can simulate typing without needing a keyboard. Even though the idea of two smartwatches feels weird, it isn't because accelerometer and gyroscope are sensors that can be embedded in any body wearables, and with the advent of new technology, we may find another wrist-wearable with different functions.

Essentially this project is divided into two parts. The first stage will be to predict individual keystrokes from the sensor data from both smartwatches. Once a decent accuracy is achieved on this, I can give a shot at implementing RNNs to better my accuracy using predictive models on what the user is about to type. For example, if a user has pressed 'Q', and then he presses some other keystroke, my model should classify this next keystroke as a 'U' with a much higher probability due to the learned RNN model, as compared to other keystrokes near 'U' (since 'u' usually entails 'q').

Since I intend to build a personal model for keyboard typing, all the data will be collected by me. In other words, all of the data will consist of my typing. I intend to use only alphanumeric characters and a few other keys on a QWERTY keyboard (such as Space, BackSpace, Enter...the list might change later). The data will be collected using a modified version of the *SensorDataLogger* app (available on PlayStore) such that the app gives me csv files at 100Hz for selected sensors of smartwatch. The ground truth will be collected through a modified version of the keylogger software *logkeys* (available on Ubuntu Software Center) such that it gives me timestamps of individual keystrokes. Since the CSV files of smartwatch data also come with timestamps, I will assume that the time on my laptop and my watch are synchronized (internet time?) and use these as my labeled data.

I intend to try various algorithms to achieve the initial classification pipeline. In my previous experience with smartwatches, I have noticed that the Fourier feature space works well in classification of different activities, hence I'll try predicting keystrokes using this feature space. If it doesn't work well, I will implement the algorithm from the related papers mentioned above. Of course I will have to modify their algorithms to use two smartwatches, and build point cloud for right hand keystrokes also. The second part of this project will be more involved, where I try to predict keystrokes and up my accuracy using that feedback model. I intend to achieve this using feedback RNNs. In the event the above ideas don't work or the theory proves too much for little time, I can use a CNN just to learn my feature space from the raw sensor data, and use a RNN to classify the stream of keystrokes, and analyze its performance.

The performance analysis of this project is straightforward. It will be the classification accuracy of the model to correctly predict the keystroke against the ground truth.