User Physical Activity Identification

Vitali Baranov, Samuel Hemond, Bing Yi Liu
Sponsor: Jonathan Andree, Galvion
Department of Computer Science, University of New Hampshire, Durham, NH

Introduction

The purpose of our project is to create a neural network model that can recognize a soldier's current physical activity, which will be delivered in the form of a library. Our work will be added to Galvion's larger Heads-up-displays(HUD) ecosystem, where it will be used to inform the system what information to display.

Android Smart Helmet

Heads-Up-Display (HUD)

Goals

- Identify the following actions with 95% accuracy: Standing, Sitting, Lying down, Walking, Running, Driving, Shooting Prone, Hiding / Crouching, Climbing
- Complete tests and model of our program using an android based program
- Create a library using our model for Galvion to use in their main app

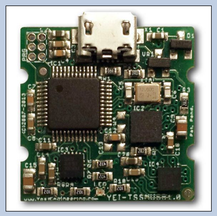
Reserach Question

Our research was based around the question:

What data do we need to collect to achieve 95% accuracy?

Hardware

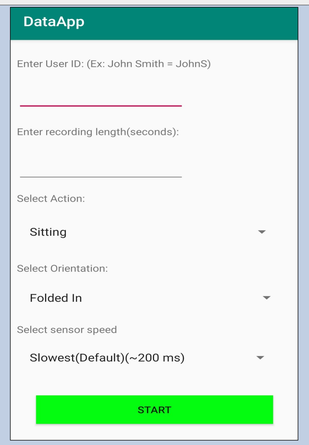
- Integrated accelerometer within an android phone mounted on a chest rig
- Yost 3 space sensor mounted on the back of a helmet



Yost 3 Sensor

Data Collection App

1. Prompt subject for settings and action to perform
2. Countdown three seconds
3. Record Accelerometer data from Android device and sensor for set amount of time
4. Add recorded data to table stored in Google Sheets



Data Collection App UI

Research

Throughout our research, we experimented with different data to include in our final model, and how to collect this data. For example, we had to find the optimal time to record each action. We had different options, such as recording for a set amount of time, or using custom times for each action. We used Google AutoML Tables to assist in conducting our trial runs, and based our trials around maximizing the accuracy of the mode.

Tools

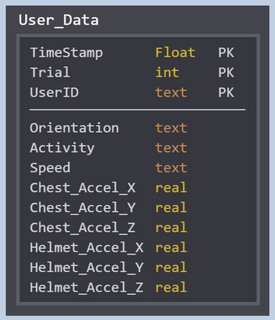
Google AutoML - AutoML Tables allows us to automatically build and deploy our models very efficiently. The platform will automatically select the best type of model, and train it end-to-end for us. To start, we input our data in a CSV file. We then need to define our schema, identifying what each column in our dataset is for, and what column we want our model to predict. Next, AutoML tables trains, evaluates, and tests our model for us. In addition, we are provided with useful information like the model's precision, recall, false-positive rate, and a confusion matrix.

Building Final Model

Originally, our plan was to use AutoML Tables to export the model to TensorFlow Lite, but we found out that this action wasn't supported with our model. As a result, we had to build and train the model in Python using the Keras library. Our final model is an LSTM(Long short-term memory) network. An LSTM is a neural network that can process sequences of data, which is why we chose it for this project. We then, exported the model to Protobuf format so that it could be used in Android.

Results & Future Plans

The Schema to the right shows the final variables we decided on recording. We weren't able to collect the final set of data, but we believe this is the data necessary to achieve 95% accuracy.
Galvion will assign this project to its intern team this summer where they will collect the final set of data and complete the implementation of the library.



User_Data

TimeStamp	Float	PK
Trial	int	PK
UserID	text	PK
Orientation	text	
Activity	text	
Speed	text	
Chest_Accel_X	real	
Chest_Accel_Y	real	
Chest_Accel_Z	real	
Helmet_Accel_X	real	
Helmet_Accel_Y	real	
Helmet_Accel_Z	real	

Final Schema