**EMOTION DETECTING SENSOR**

**Bandaru Sai Kishore**

**Degala Laxman Dutt**

**Arapally Ebenezer Anand**

**Kapa Avani**

**Pasala SaiKiran**

**Increment 4**

**Project Goals and Objectives:**

**Motivation:**

In certain situations a person may not be able to find the right choice or some way to revive him/her self. So this being the prime motive, we came up with a solution to handle those kind of situations with a mobile application “What to Do?” which is an emotion sensor recognizer. This application helps you to find the right option to revive yourself.

**Significance:**

The application is basically designed for the android mobile users and the important feature of this application emotion recognition. These emotions will be detected based on your hand movements. And based on them we will be detecting your current mood and recommend you with certain options which can uplift your mood further.

**Objectives:**

The main goal of this application is to detect your emotion at the current moment and suggest you places or the things you can do to revive yourself or to uplift your mood. Here we use the basic hand movements for the motion recognition and detect the emotion. For example if you make a hand movement which resembles like stomp then it indicates that your current mood is “irritated” and likewise we will be able to detect five different motions using our sensor based application. Here we use the sensor tag to detect the hand movements. In addition to this activity recognition we also make some recommendations to revive yourself by suggesting some places to visit. Here we split the users further as kids, students and professions and suggest the appropriate places.

**System Features:**

The system is basically developed as a native mobile application. We use the sensor tag for tracking the hand gestures. We also use the google charts for designing he reports for the data that is being collected and also make some sample introduction to the recommendation system we provide in the reporting tool. Initially we shall train the data and store the sequence files in the android mobile. Then when we record the hand gestures we shall again generate a sequence file for the hand gesture and using the K-NN algorithm we shall find out the matching activity for the hand gesture using the trained sequence files and the test sequence file. Thereby depending on the activity we shall recommend the places to visit for the users.

**Software Specification:**

Tools: Android Development Kit, R, Sensor Tag (External Tool), Google Charts

Operating System: Android

Development Operating System: Windows 8

Programming Language: Java 7.0

Databases: HBase

Here is the flowchart of the system operations:

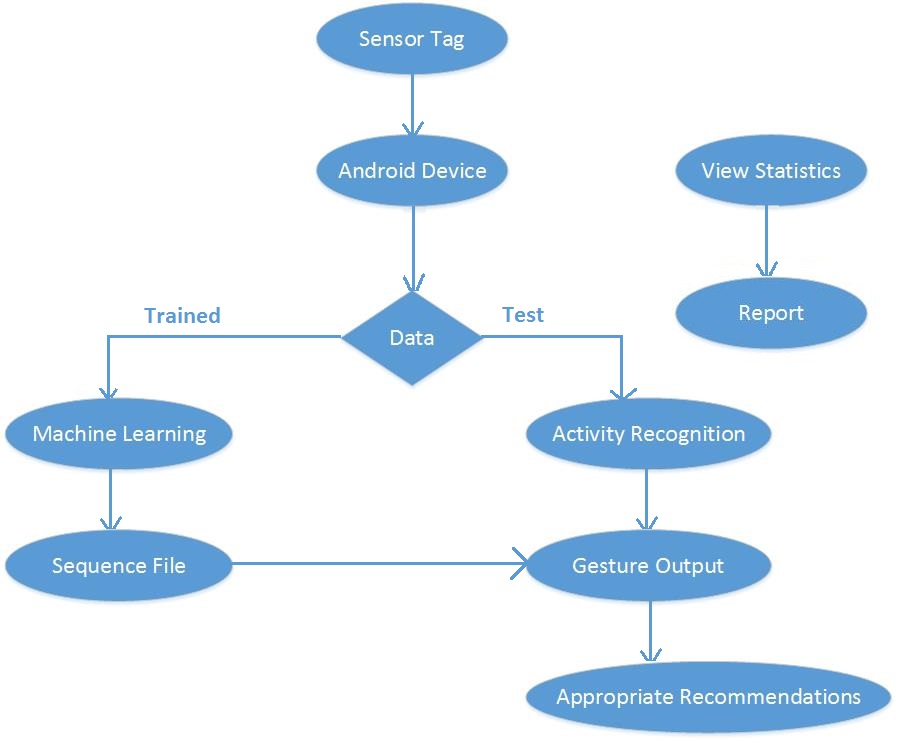


Figure Flowchart

**Online Application: Application by Motion**

**Devices/Sensors:**

Devices which we have used for our emotion detection application are:

* Android mobile device
* Sensor Tag

The sensor tag here we used is a product of Texas Instruments.

**Motion Models:**

We basically have five hand gestures to detect. Since we already have sequence files of the trained data. Now the new gesture data will be collected using the sensor tag and then a sequence file will be generated for these data. Now these sequence file will be compared with the trained sequence files and the appropriate gesture is detected. The five gestures which we use for emotion detection are:

1. Facepalm Hand gesture is represented as worried
2. Weeping Hand gesture is represented as Sad
3. Stomp hand gesture is represented as Irritated
4. Punch hand gesture is represented as Angry.
5. High Five hand gesture is represented as Happy.

The following hand movements are represented in the below snapshot:



Figure Hand Gestures

**Application Features:**

The primary feature of our application is to tap your hand gestures and detect your current emotion. Now based on you emotion we shall some places to visit to revive yourself from your current mood. The recommendations will not be same for all the users but varies from user to user like kids, students and professionals. These application is basically designed for the Android users and the support will be provided starting from SDK version 18 till SDK version 19. The high value for the minimum SDK version is to provide support for the detection of the sensor tag. Here we also need to use the BLE Sensor Tag application in order to detect the sensor tag on your mobile device.

**Recognition/ Analysis Approaches:**

As mentioned earlier in our application we have used the K-NN logic and also HMM model for the detection of the emotions based on the hand gestures. On the background we will already prepare the sequence files for the trained data. Now in our application on the foreground we will first detect the hand gesture. This will be done by using two buttons namely “Start recognition” and “End recognition” which indicates the start and end of hand gesture respectively. Once the user is done with providing the hand gesture, on clicking the “detect emotion” we shall first take the values i.e. the sequence file values and using the algorithms of K-NNand HMM model we shall detect the emotion. So the detect emotion has two important methods which handle the motion detection. They are:

* Train()
* Test()

In the train function first we shall take the trained sequence files and train the application. Now once we have the trained results, which will be stored internally in the cache by the K-NN algorithm, we shall take the currently tapped sequence file data and map them to the trained results so as to detect the emotion. The following is the class diagram, which depicts the important functions and activities performed ine ach phase which are prior mentioned:



Figure Class Diagram

**System Features and Android App GUI:**

The following is the starting page of the android application which has four buttons:

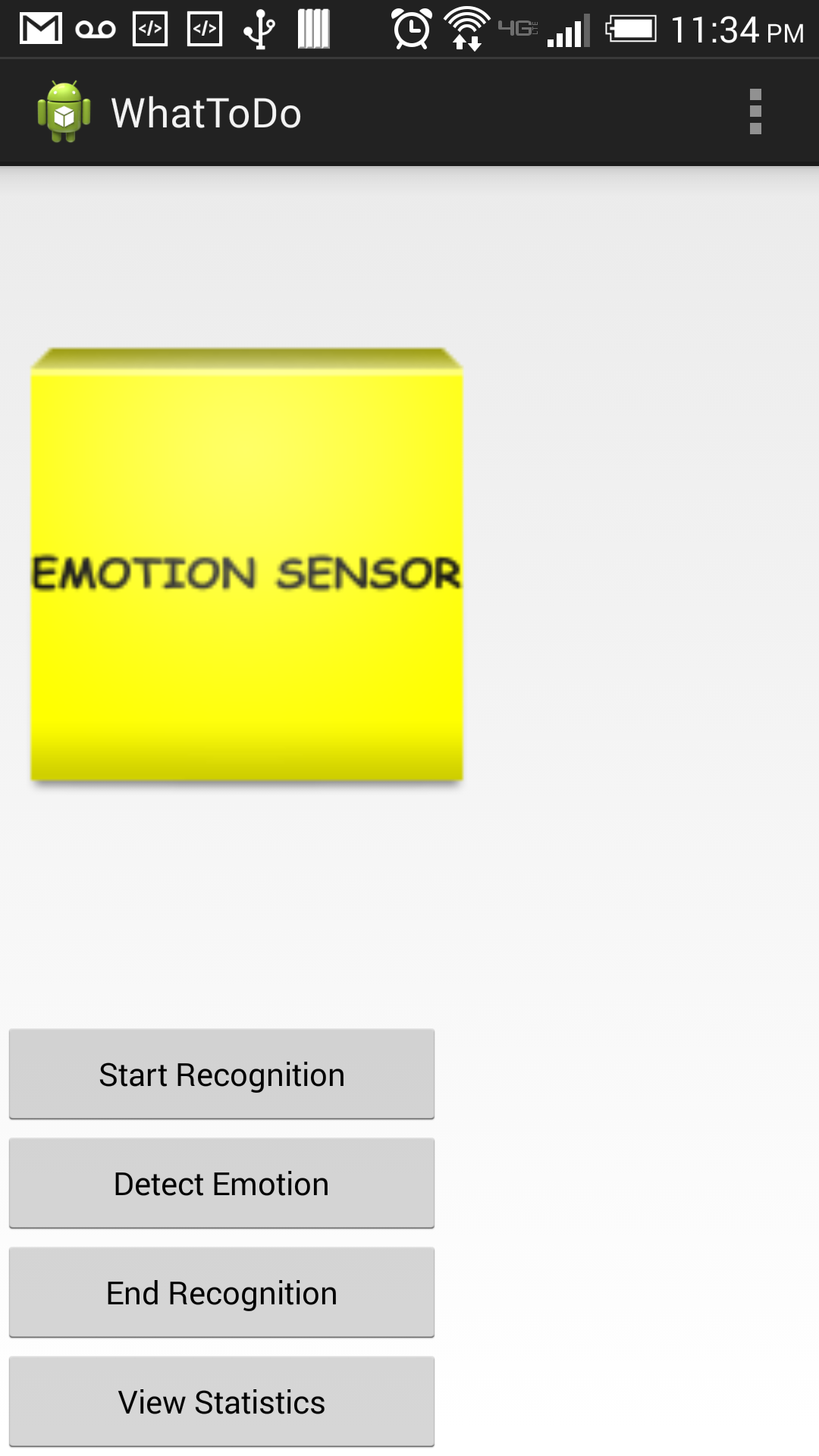


Figure Home Page

Now click on the start recognition button and start making your hand gesture. Once you are done with the gesture making click on the end recognition button. The following is the snapshot of facepalm hand gesture:



Figure Facepalm Gesture

Now click on the Detect Emotion button to detect your mood. The following snapshot represents the detected emotion for the facepalm hand gesture:

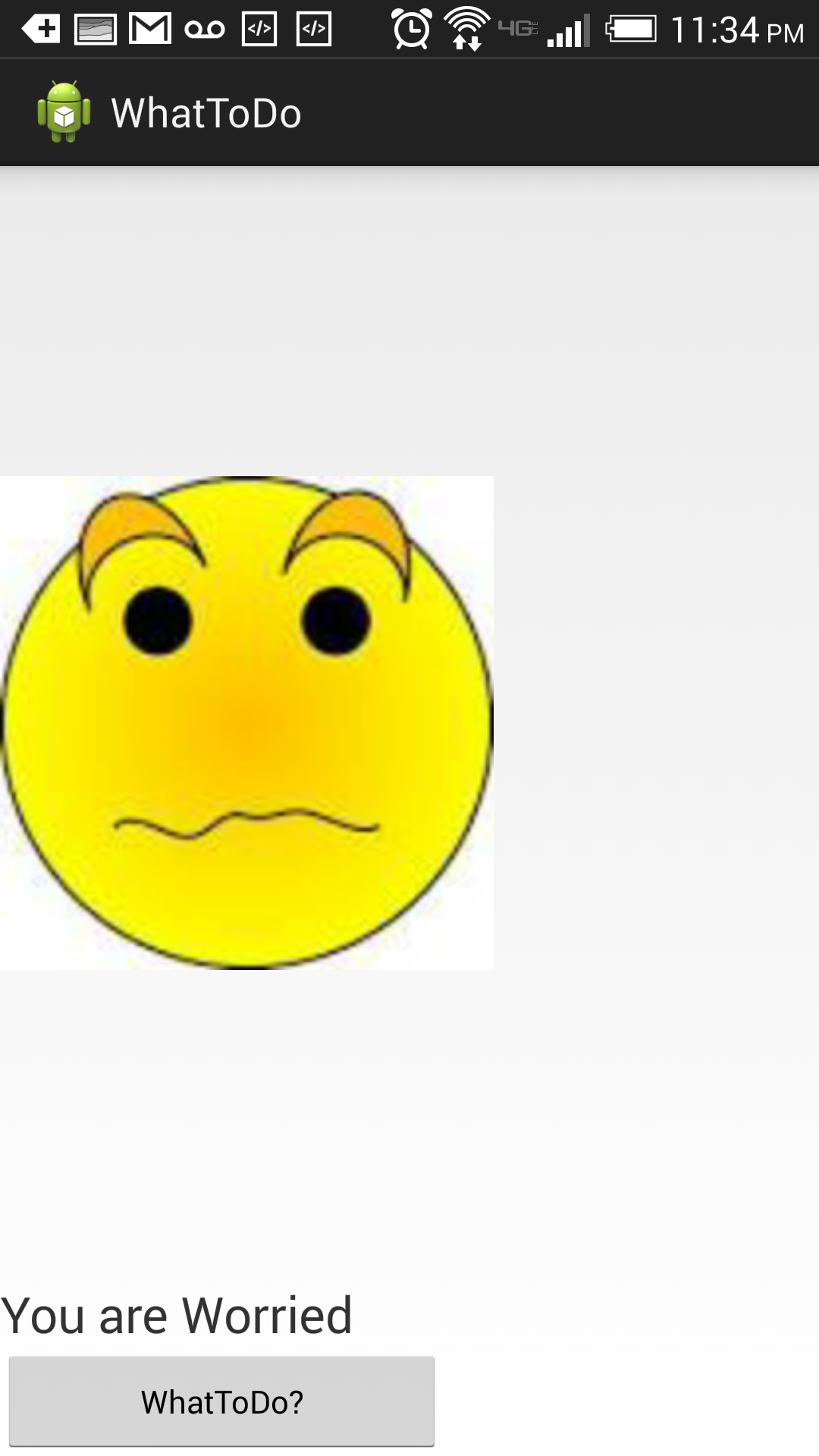


Figure Emotion Detection

Now to revive you form present mood click on “WhatToDo?” button. On clicking the “WhatToDo?” button you will prompted with a page containing three options and will ask to select on appropriate option:

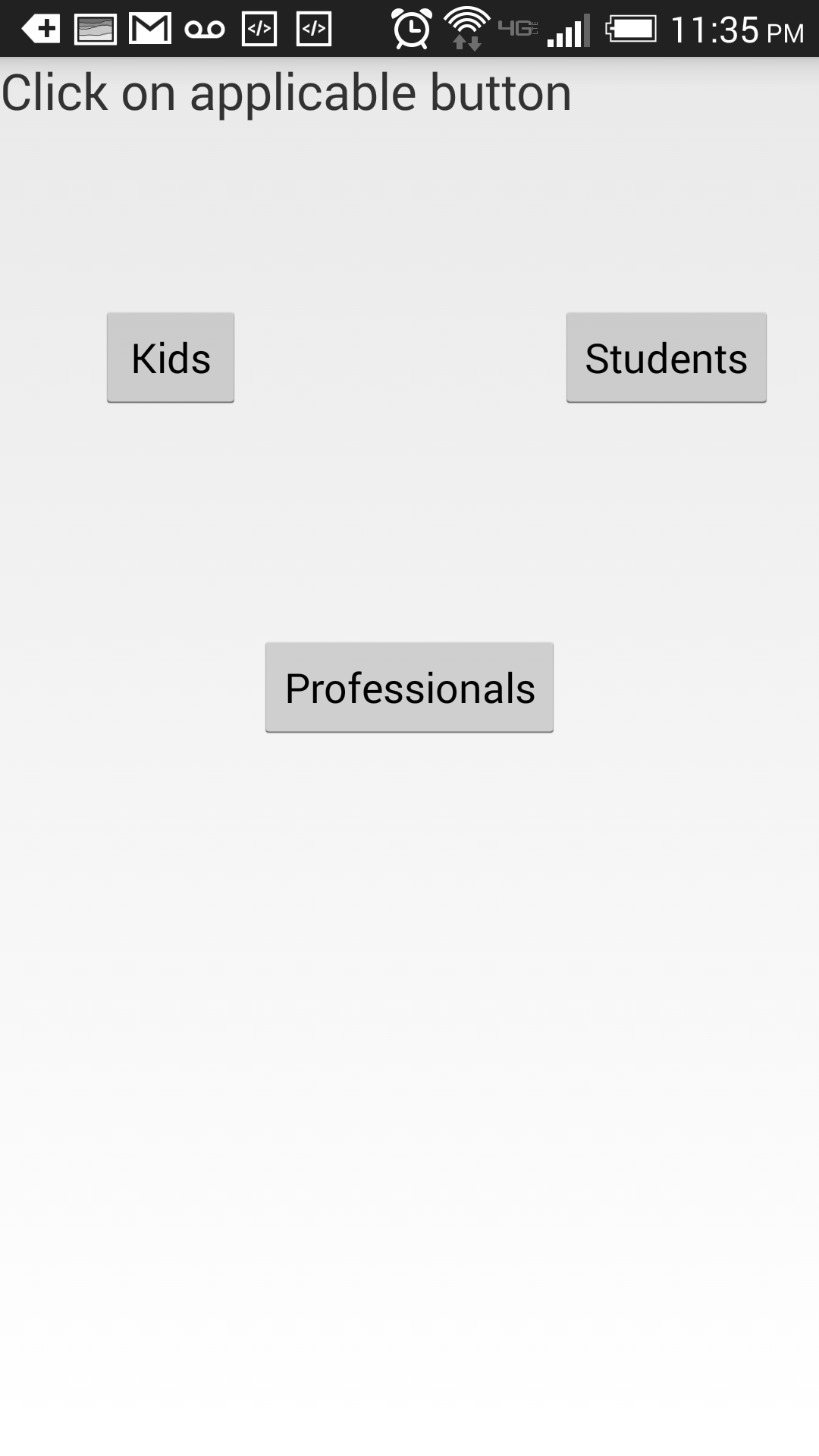


Figure User identification

Now on clicking a button say student the following page will be displayed:

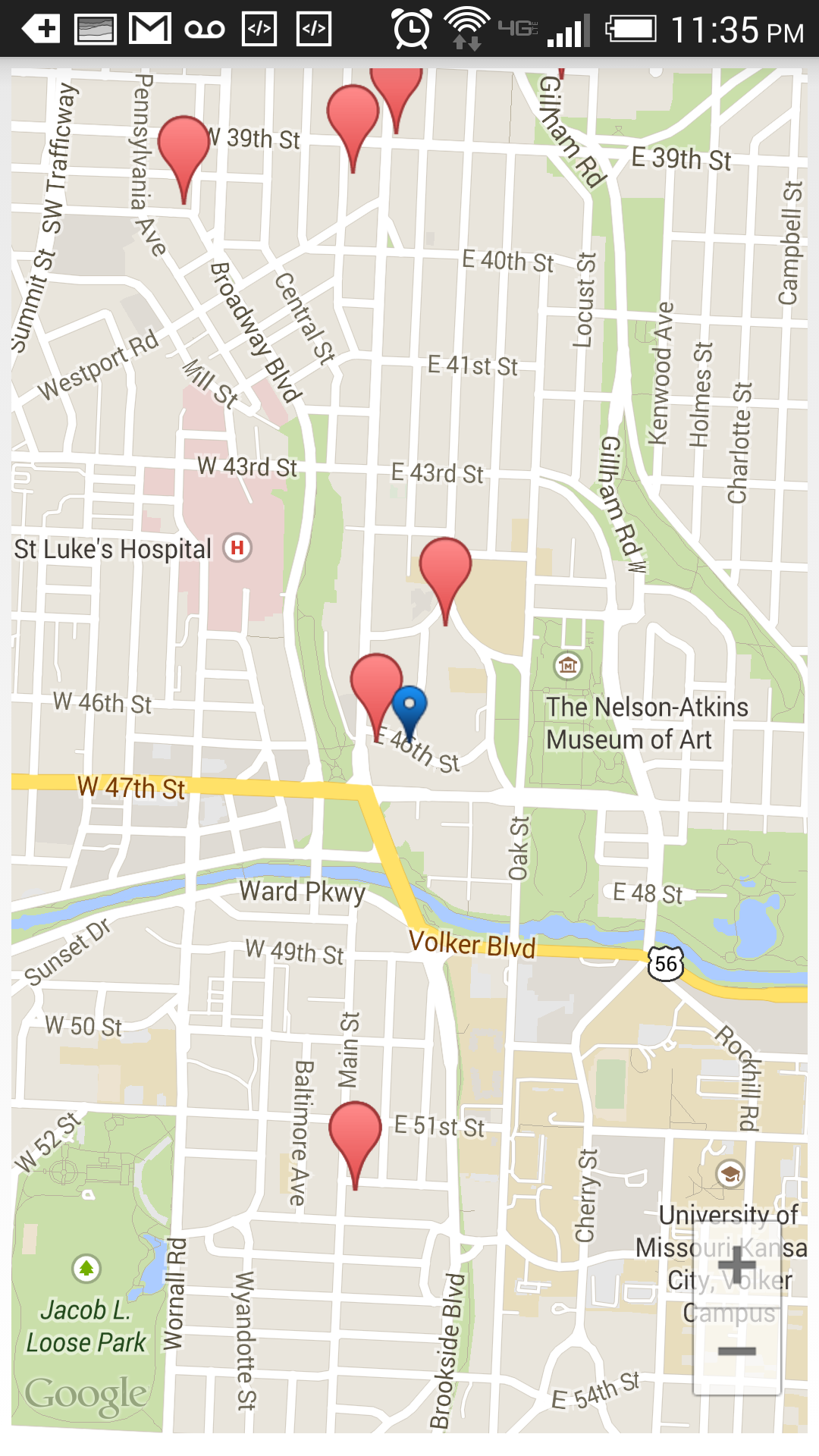


Figure Maps with places to visit

Here the blue label indicates your current location. Now on clicking on any of the red button will pop up the place name.



Figure Location Name

Since the person we suggest and user being student we suggest the user to visit some holy places or parks to revive.

**Offline Application: Activity Report**

**Devices/Sensors:**

Devices which we have used for our emotion detection application are:

* Android mobile device
* Sensor Tag

The sensor tag here we used is a product of Texas Instruments.

**Activity Models:**

As mentioned earlier the activities here represent the user emotions whether he is worried, sad, happy, angry or irritated. Based on the hand activity we are going to recognize the emotions. And the other activity we have in our application is recommendation system based on the detected emotion. Now our primary in the offline part is to generate sequence files for the trained data. Here each hand gesture is trained with twenty samples from five different users. Now these data has been trained and the respective sequence files have been generated for each and every hand gesture.

**Features:**

The main features of the activity recognition is to detect five different gestures i.e. facepalm, weep, stomp, high five, and punch. Now each hand gesture has the respective human emotion. Now we have trained this gestures in such a way that the trained data has both the left hand and right hand gestures with five different users to achieve accuracy. We have used the K-NN technique and certain mathematical probabilities to detect the start and end of gestures.

**Recognition/ Analysis Approaches:**

The analysis of the values is being done based on five different users with each user making a gesture four times. This made the process a bit efficient and the accuracy values and precision values are further discussed. Initially we thought of using the R and Hbase for analysis and storage of data respectively but due to compatibility issues we have not included in the recognition or analysis process. Now to analyze the recorded data we have used google charts to represent these data. The data model of the HBase we have initially designed was in the following manner for storing the data:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| RowKey | TimeStamp | Motion1 | | | Motion2 | | | ……. |
|  |  | X | y | Z | X | y | Z | ….. |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

Detector

**System Features:**

As mentioned prior we have designed these application for Native mobile users with operating system Android and SDK version support from 18 to 19. The following is the system architecture of our application:



Figure System Architecture

Initially in activity report we collect data from sensor tag for all the five hand gestures. Now these data consists of the x, y, and z co-ordinates. These data will be now sent as input the machine learning algorithm (here we used k-nn) and the algorithm returns a set of values in the sequence file as output. Now these data will be used for the activity recognition whenever the user uses the application.

Once the emotions of the users are captured based on their gestures (for example Happy, Sad, Irritated), we have planned to create a report on daily basis to allow user know how his emotions were on that particular day. For simplicity sake, we have taken a 3 hour interval starts from 6am in the morning till 9pm in the night.

We have used Google Charts. Based on our data file, we came to a conclusion that Bar Chart would suit better for our project. For simplicity we have considered only 3 emotions Happy, Sad and irritated when it comes to plotting the graph. This can be upgraded for any number of different emotions. X-axis has the time and Y-axis has the count. And it’s a Daily Activity Report.

A person can have various different emotions in this interval of 3 hrs. We count the number of times he is happy or sad or irritated etc. based on his different gestures in that 3 hours and plot the graph. Based on these graphs, user tries to decrease the curve under the emotion 'Sad', this helps him lead a happy life.

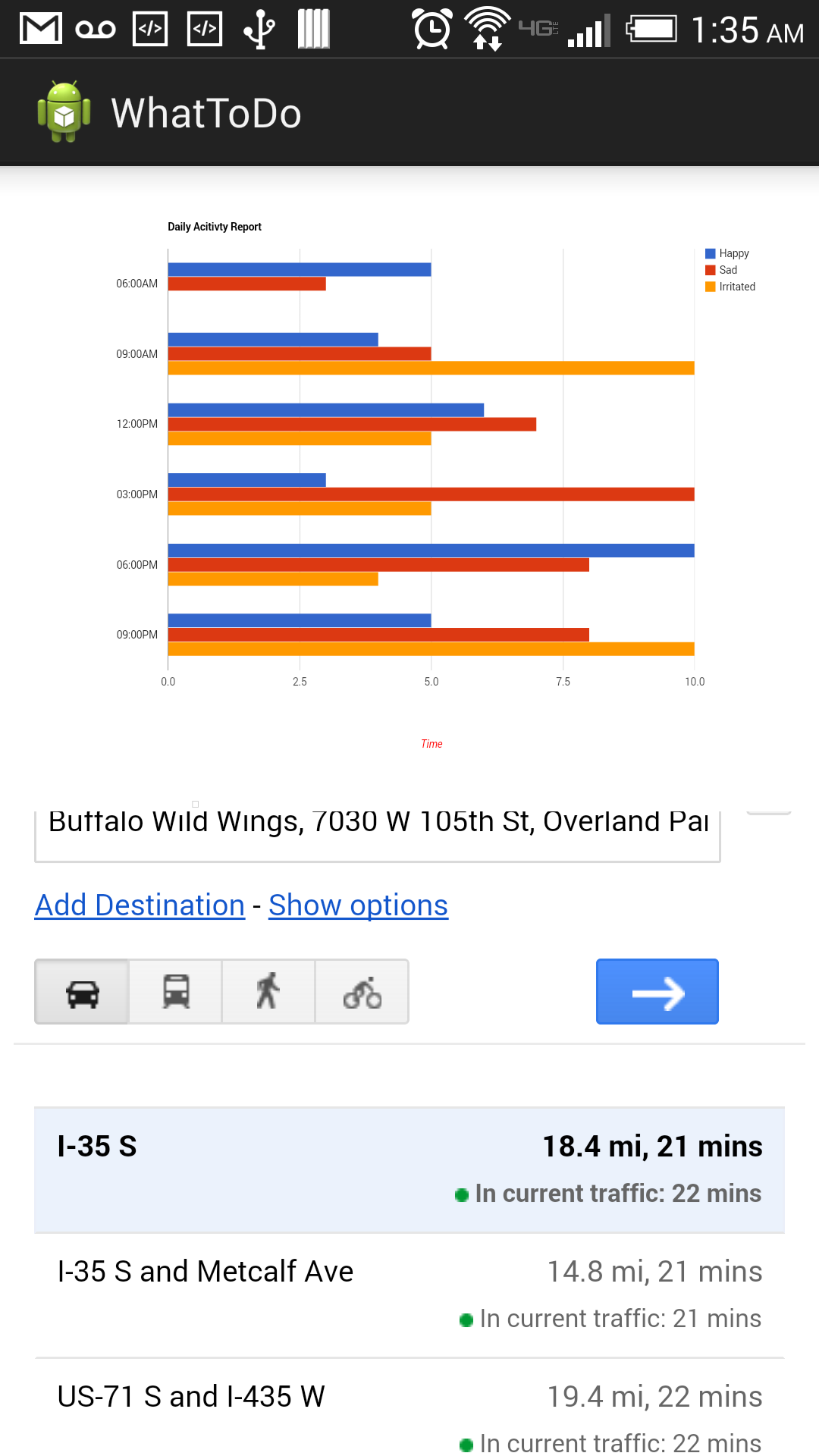


Figure Reporting

**Evaluation: Motion/Activity Recognition**

**Number of Users:**

The data has been collected from five users and each user will be making five samples of actions for each gesture.

**Types of Motion/Activities:**

We have typically used five gestures in our application to detect five different emotions. The five gestures which we use for emotion detection are:

1. Facepalm Hand gesture is represented as worried
2. Weeping Hand gesture is represented as Sad
3. Stomp hand gesture is represented as Irritated
4. Punch hand gesture is represented as Angry.
5. High Five hand gesture is represented as Happy.

**Size of Data/Number of Gestures:**

As mentioned prior we have five different gestures. And the size of data for each trained gesture is found to be around 20-25 KB based on the gesture. But each training file has a huge number of values which are found be good enough for detecting the activities.

**Data Preparation:**

The data has been collected using the sensor tags. And after recording of each gesture we will be using these files for machine learning to generate the sequence files. The following are the snapshots of the respective data collected for one kind of gesture i.e. facepalm and its respective sequence files:

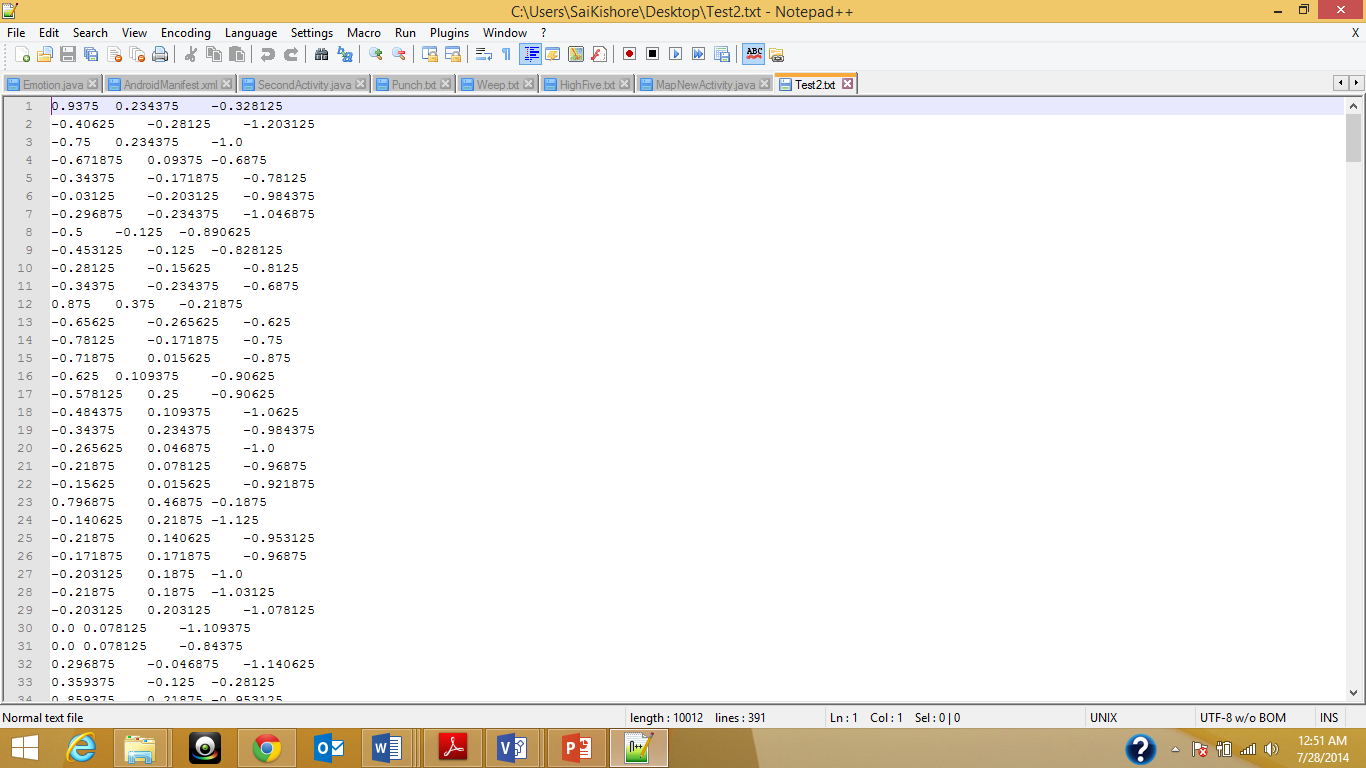


Figure Trained Data

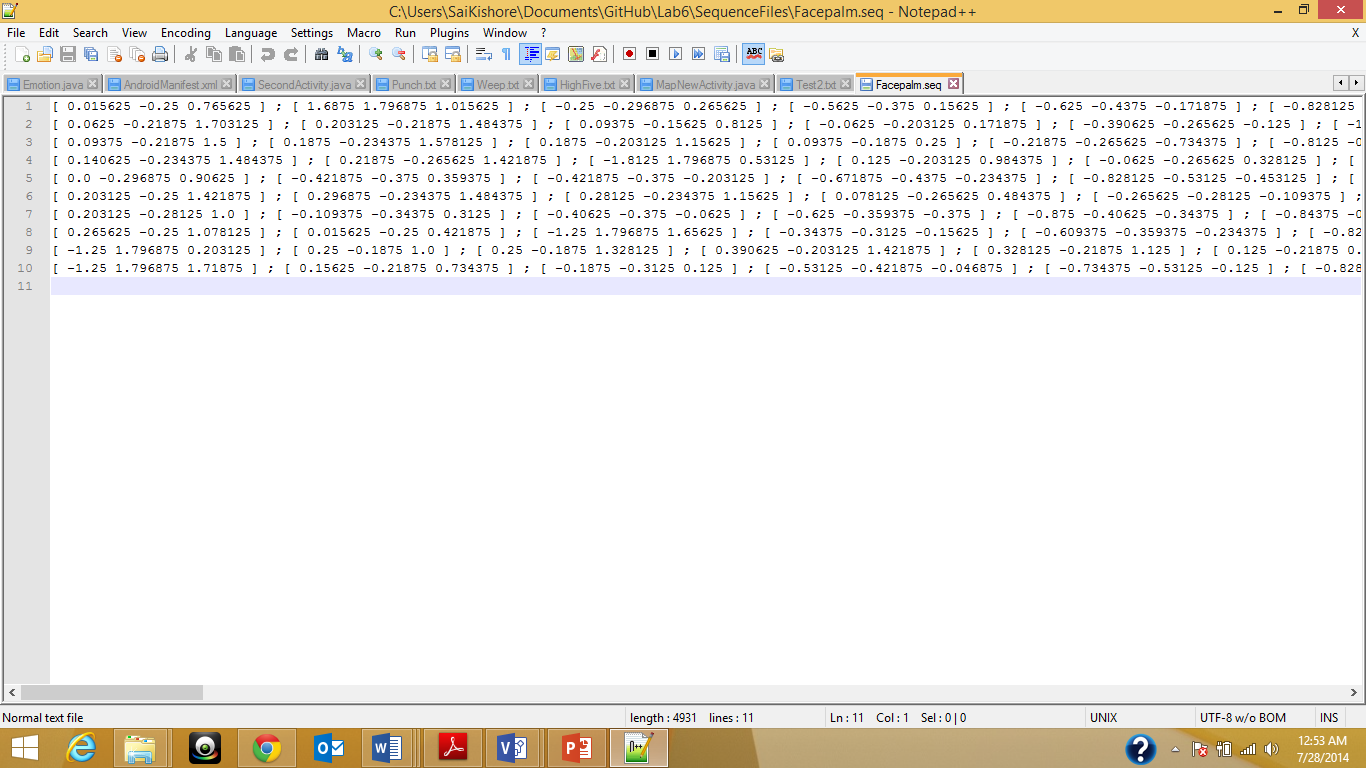


Figure Sequence file

**Accuracy – Precision/Recall/F-Measure:**

One observation we made while the project development was that the more is the trained the more efficient sequence files are generated which will help in high accuracy. The efficient sequence files include sufficient number of values in each row which helps in k-nn for detecting the emotions properly. The following are some of the mathematical observations we made for our project:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Happy | Sad | Angry | Irritated | Worried |
| Happy | 6 | 0 | 0 | 2 | 0 |
| Sad | 0 | 5 | 0 | 0 | 3 |
| Angry | 3 | 0 | 5 | 0 | 0 |
| Irritated | 0 | 0 | 0 | 8 | 0 |
| Worried | 0 | 2 | 0 | 0 | 6 |

Data Observations

|  |  |  |  |
| --- | --- | --- | --- |
|  | Precision | Recall | F-Measure |
| Happy | 0.667 | 0.75 | 0.706 |
| Sad | 0.71 | 0.625 | 0.664 |
| Angry | 1 | 0.625 | 0.77 |
| Irritated | 0.8 | 1 | 0.889 |
| Worried | 0.67 | 0.75 | 0.702 |

Accuracy Calculations

**Limitations:**

The project is in working mode now and has all operations working properly. There is certain issues in detecting the irritating motion as it is mapping with happy gesture. But this was rarely observed. The application doesn’t support any inbuilt sensors and is applied only for the sensor tag designed by texas Instruments.

**Project Video:**

The following is the URL for the video presentation on the application usage and its brief insight:

* <https://www.youtube.com/watch?v=sjL12Td0vFI>

**Project Planning:**

The project tasks were completed on time with desirable accuracy rate. However certain enhancements can be made for the project in regard with application GUI by including some catchy backgrounds and emotions. All the tasks and their increments are being updated in the scrumdo.

* [https://www.scrumdo.com/organization/umkc95/dashboard#](https://www.scrumdo.com/organization/umkc95/dashboard)​

**Bibliography:**

* Developing Android Applications Using Sensors -- <http://developer.android.com/guide/topics/sensors/sensors_overview.html>
* Microsoft research Projects -- <http://research.microsoft.com/en-us/projects/imanexpert/>
* Developing and testing recommendation Systems: <http://cran.r-project.org/web/packages/recommenderlab/vignettes/recommenderlab.pdf>