CSE 118/218 Final Report - Personality Insights

WEIYUAN WEN, University of California, San Diego, USA
CHENGLIN LIN, University of California, San Diego, USA
VINCENT LIM, University of California, San Diego, USA
DANIEL KOSTINKIY, University of California, San Diego, USA
CHAO WANG, University of California, San Diego, USA

With the advancements in research for context recognition in relation to gathering accurate context data using sensors available "in the wild"[9, 10], we wanted to apply this new technology to help us in the world. To do that, we looked at how this can improve the social aspects of our lives. In the modern age, social media is a prevalent tool that is used to help people stay connected and make connections. We introduce Personality Insights, a potential add-on to existing social media platforms. Personality Insights uses context data gathered through Extrasensory[9] in order to enhance existing connections and streamline the creation of new ones. In this paper, we will discuss how we came up with and designed Personality Insights, the value it provides, and how we see this technology evolving in the future.

Additional Key Words and Phrases: activity recognition, social media, competition, data visualization

ACM Reference Format:
Weiyuan Wen, Chenglin Lin, Vincent Lim, Daniel Kostinkiy, and Chao Wang. 2017. CSE 118/218 Final Report - Personality Insights. 1, 1 (December 2017), 23 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Over the course of 11 weeks, we studied ubiquitous computing systems related to context recognition, came up with a problem we wanted to solve, designed and developed a working prototype, and demonstrated this prototype along with our visions for what we hope this application will become in the future. We learned about ExtraSensory, an application developed by one of our TA's, Yonatan Vaizman. ExtraSensory essentially uses the sensors available on a mobile phone to detect and label a user's context. This context data is then stored on the user's phone in JSON format. We were tasked with making a new application utilizing this tool, so we first determined what problem we want to solve. That's when we looked into our personal and social lives and decided that we wanted to utilize this context data to help users understand their personalities and the personalities of those around them.

In the motivation and background section, we will explore why we created Personality Insights and what problems we hope it solves. We will take a quick look at possible use cases for Personality Insights and see how it can utilize in our everyday lives. Then we will talk about what Personality Insights is and what it is not.

Authors' addresses: Weiyuan Wen, University of California, San Diego, 9500 Gilman Dr. La Jolla, CA, 92093, USA, w5wen@ucsd.edu; Chenglin Lin, University of California, San Diego, 9500 Gilman Dr. La Jolla, CA, 92093, USA, chl734@ucsd.edu; Vincent Lim, University of California, San Diego, 9500 Gilman Dr. La Jolla, CA, 92093, USA, vincentorionlim@gmail.com; Daniel Kostinkiy, University of California, San Diego, 9500 Gilman Dr. La Jolla, CA, 92093, USA, Dkostins@ucsd.edu; Chao Wang, University of California, San Diego, 9500 Gilman Dr. La Jolla, CA, 92093, USA, c.wang.twenty@gmail.com.

XXXX-XXXX/2017/12-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

In the design section, we will look at specifics on how we designed Personality Insights, including but not limited to creating personas, storyboards, our design process, and different design decisions we made throughout the development of this application.

In the system development section, we will take a look at our toolkit and why we chose to use the APIs and services that we did. We will also explain our system architecture and give details about how it works and what features Personality Insights has.

In the testing and evaluation section, we will talk about user testing, some feedback we received, and the tests we did on our dataset. We will also reflect on our feedback, which we have taken into consideration when talking about our future works.

In the collaboration section, we will talk about our team. Specifically who contributed to what during these past few weeks. We will highlight some problems we had, how we solved those issues, the structure of our team, and our views on how collaboration worked between CSE 118 and CSE 218 was this quarter (FA 17) at the University of California, San Diego.

And finally, in conclusion, we will summarize what we achieved with Personality Insights thus far and explore how we hope to expand this application in the future.

2 MOTIVATION AND BACKGROUND

While coming up with Personality Insights, we considered many real-world problems that we wanted to solve, ranging from health to lifestyle choices. At first, we considered making an application to help individuals live in healthier lifestyles. But we realized there are already a lot of applications that tackle this problem and, although context data would have made our solution unique, we wanted to solve a more identified problem. We then went in the direction of creating an application that will help users make decisions, which can be done through recommendations. Past studies show that having mobile recommendations dramatically enriches an experience[11], and we believe the best way to do this is to strive for ubiquitously to avoid being ignored like pop-ups [1]. Before long, we thought of tools that detect mood in social media platforms by analyzing posts[5]. And that's when we thought of a personality classification application, which would help users understand their personality and their personalities of those around them.

Understanding personalities is a skill we use in our everyday lives to help us make informed decisions [3]. We use this skill to determine if someone is fit for a job, determine what gifts to get people, determine what to do with certain people, so on and so forth. As an example, imagine you had a friend named John Doe. You know John is very athletic and outgoing, and he loves doing things with others. You may consider buying him karate lessons, which would fit his athletic and extroverted personality.

The better we understand someone's personality, including our own, the better decisions we can make. There are other 'personality insight' tools that help individuals learn about their personalities, such as the Myer Briggs test and the Strengthfinders test. Our app, Personality Insights, accomplishes a similar result by using what an individual has done rather than what they would do. This minimizes the room for error when analyzing an individual's personality.

We believe that by analyzing an individual's context, we can infer their personality. In other words, we believe that we can predict an individual's personality based on what they do. So by sorting and displaying context data received from ExtraSensory, we hope that users will be able to infer their personalities as well as the personalities of their peers. We will mention in the future works section that we hope to have this inference step be automated by the application itself. But for now, Personality Insights simply augments the users' ability to understand personalities.

A recurring theme in this class was to make something that is potentially ubiquitous, and we believe that our application has this potential. We created a tool that is useful in day-to-day life. Therefore, we believe Personality

Insights will be used somewhat frequently. We hope that context recognition will become advanced enough to eliminate user interaction in terms of correcting and confirming labels. We believe the idea behind Personality Insights is flexible enough to expand upon and improve. We believe we came up with and created a working prototype of something that will be commonly used and blends into the background, becoming ubiquitous.

The goal of Personality Insights is to help users live better personal and social lives by helping us make informed decisions. And we believe that a data visualization based application is the best way for users to understand "personality data" so that they can begin to make those decisions. With that said, the main contributions of this work can be summarized as follows.

- To the best of our knowledge, Personality Insights is the first tool to utilize context data to analyze an individual's personality
- Personality Insights is more accurate than other personality tools because it is based on what has happened rather than hypotheticals
- We combine HCI with psychology in a new way, possibly opening a path for others in this direction

During our final demonstration, as well as throughout the development of Personality Insights, we realized that there might be a lot of misconception of what our application does at first glance.

- Some believe our application is a new social media platform, but we don't want that to be the case. We want Personality Insights to be either a stand-alone tool or an add-on to existing platforms.
- Setting goals and staying on track is not the main point of Personality Insights. Although the application can help you see what you do and how often, that is not the main goal.
- Personality Insights does promote friendly competition, but this is simply a side feature.
- Groups were created with the idea to mimic Facebook groups for possible future integrations with the social media platform, not to make Personality Insights its own social media
- Our main focus is personal and social life quality. Not health, which seems to be a possible side case of our application as it can help people live physically and mentally healthy lifestyles

Before continuing with this paper, we want to acknowledge the security and privacy concerns that Personality Insight poses. Some of these concerns are inherited from ExtraSensory, which we utilize to get our data. These concerns are related to how ExtraSensory gathers data on what you are doing nearly 24/7. Having an application be able to do that raises lots of questions about privacy and security. Personality Insights also takes the concerns a step further by sharing some of that gathered data.

Despite these concerns, we hope that research will continue in related fields. We believe the benefits outweigh the cost, and we also believe there will be a way to mitigate these issues. There's also the possibility that the next generation will not be worried about these issues. Afterall, we have a lot of our personal data out on the internet through social media as is. So we believe Personality Insights has the potential to continue developing and become a new tool used in our lives, and that future applications may also use context in a similar manner.

Personality Insights is a tool that augments our ability to understand personalities. Now that you understand some of our motivations, some considerations and reserves we thought about, and some background about our application, we will now discuss how we designed Personality Insights.

3 DESIGN

From the start, we knew that Personality Insights would be very much data-oriented, and the user interface and experience would have to revolve around that. Our design process was mostly standard for modern times. We created storyboards, personas, wireframes, and iterated through an agile implementation process.

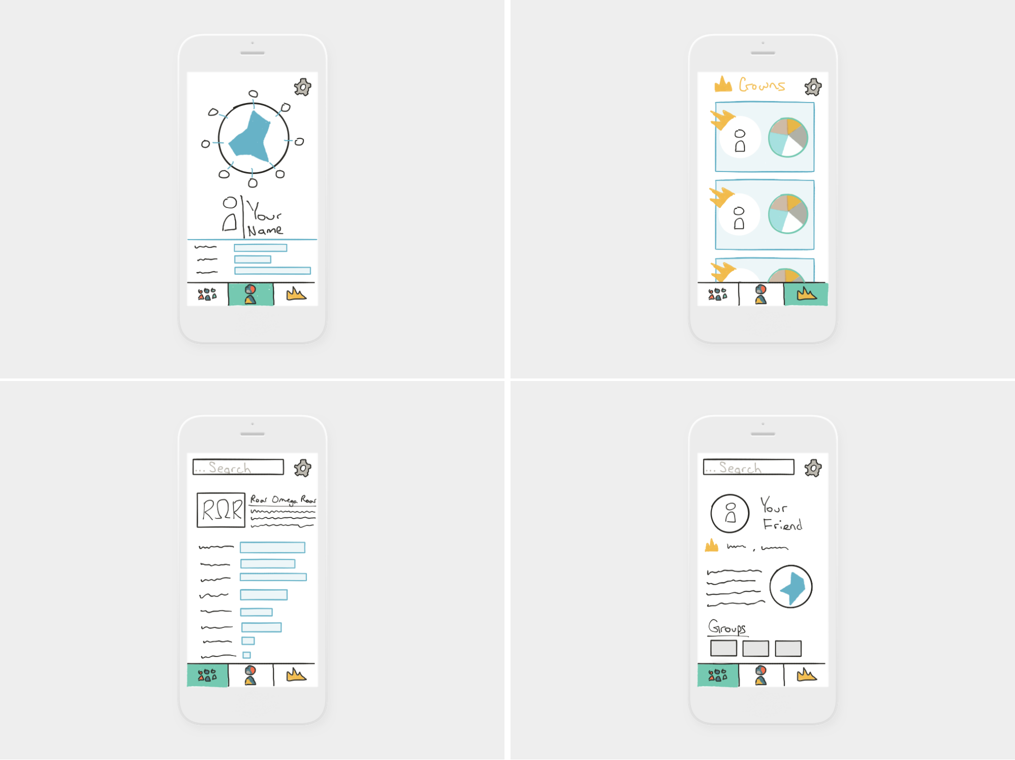


Fig. 1. Initial Wireframes for Personality Insights

3.1 Pre-Implementation

Before we implemented our idea, we create storyboards to see how Personality Insights can be used. This allowed us to see the 'before and after,' which enabled us to make a comparison and judge our application's usefulness and feasibility. We were able to create user stories from these storyboards, which were applied to personas we created.

We created two personas to demonstrate two example users of our application. We named these personas Jack and Jim. Jack and Jim were twins who had different personalities. They each wanted to make a decision based on their personalities and the personalities of those around them. Jack wanted to join a fraternity at UCSD but wasn't sure where he would fit in. With Personality Insights, Jack can see what the overall personality of a group is and make a decision based on that. Jim wanted to find someone to talk to about books and movies. He finds out that he has some friends nearby who spend a lot of time doing those activities, thanks to Personality Insights, and decides to reach out to them. As we can see, Personality Insights can help us make difficult decisions and give us the insight to allow us to accomplish certain things.

Our wireframes consisted of different kinds of graphs, which made sense because we wanted Personality Insights to be data-oriented. We believed the most efficient way to display the information to the user was to utilize data visualizations. We had radar charts, bar graphs, and pie charts throughout the UI we mocked up. Due to time constraints, we weren't able to completely follow our wireframes, which included personality classification based on label clustering. However, we were able to create our current prototype based on the wireframes we established at the beginning.

As a real quick note, we decided followed an agile-like structure for implementation because of the time constraint of the class. We wanted to make sure we had a working demo by the end of the course, and the best way to do so is to follow the agile methodology and have a working prototype after every iteration. Although we were only able to go through a few iterations, we believe we made significant progress in the little time we had.

3.2 During Implementation Design Decisions

With the wireframes in mind, we first started build up the foundation of the project by setting things up. So for the first two weeks, we were making sure everyone has their development environment ready. Our initial goal is to flesh out the three pages as soon as possible. To have a quick recap, the three pages are Profile for personal activity data, Crowns for friend ranking, and Group for group information and searching. At Week 3, November 18, we had a meeting with Yanaton, and many design decisions were made at the meeting. Mainly, we have decided how to pre-process the data from ExtraSensory and what exactly to show on the App. Here, we imagine users would like to have minute-per-day information to make sense about their daily routine, and the data will reset on Sunday midnight so that every week there might be a new top one on Crowns page. Later in Week 4, November 25, we added activity selection to allow users to pick the activity they want to see. In Week 5, besides creating Group page, we think it would make more sense to have a Friend page to show individual profile. Figure 2 is a timeline for the timing those decisions were made.

4 SYSTEM DEVELOPMENT

4.1 Architecture

We wanted data to be updated in real time, so we chose to use React Native to take advantage of reconciliation that takes care of real-time updates for us. Firebase also has a database so to store our data we chose to use Firebase Real-Time Storage, which allowed us to keep our data in a neat "folder" like data structure that can be extracted as JSON, and JSON is something easy to have your data as which is incredibly nice. We also needed a backend to handle some logic such as logging in, registering, and pushing/retrieving data from phones through the use of our application, so we used Firebase functions to make sure everything works properly. Figure 3 shows the architecture of our system and Figure 4 is our final schema of database. Finally, and most importantly we needed a way for context level data to be made available to us to be used for the core of our logic, so we incorporated Yonatan Vaizman's work called ExtraSensory. ExtraSensory provides the user with current contexts probabilities which we then used an average of to find what the most likely context a person would be in. Using ExtraSensory and Firebase functions we allowed ExtraSensory to work in the background, and then we grabbed and pushed this ExtraSensory data to our database every three hours. One of the things that we should update is our ExtraSensory pushing should be every minute instead of every three hours because as of right now the user has to wait three hours for the application to be useful for them, instead of one minute which would have been considerably better for the user.

4.2 Technology Used

- **React Native**
  React Native is a framework developed by Facebook for mobile development. A developer can use Javascript and React with some Native components to build a mobile application in a short development cycle. Developers just need to write the code once and React Native will be compiled into both IOS and Android versions.We used it to make our team work faster because it was easy to learn and is useful for a later iteration of this program because we can output it to both IOS and Android. Another advantage of React

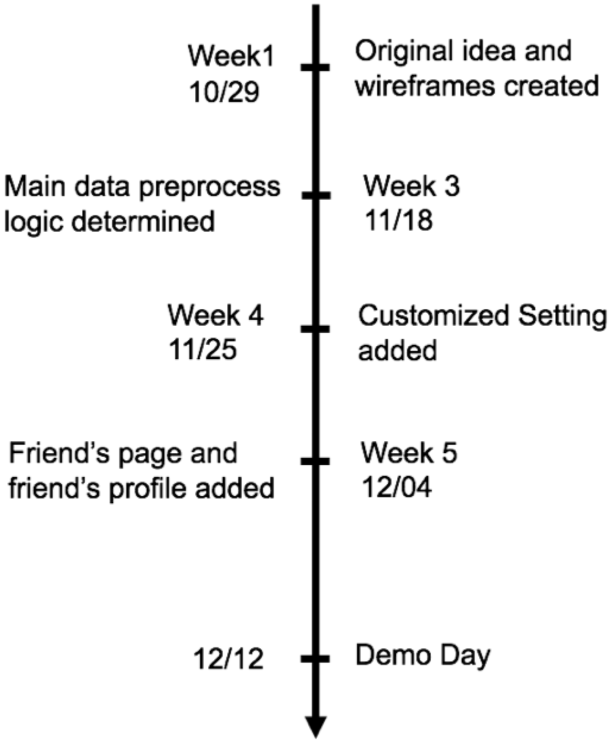

Week1
10/29
Original idea and wireframes created
Main data preprocess logic determined
Week 3
11/18
Week 4
11/25
Customized Setting added
Friend's page and friend's profile added
Week 5
12/04
12/12
Demo Day

Fig. 2. Timeline of idea evolution for Personality Insights

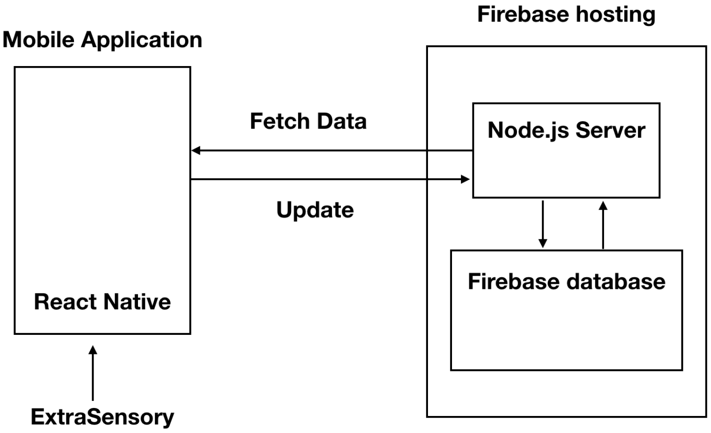
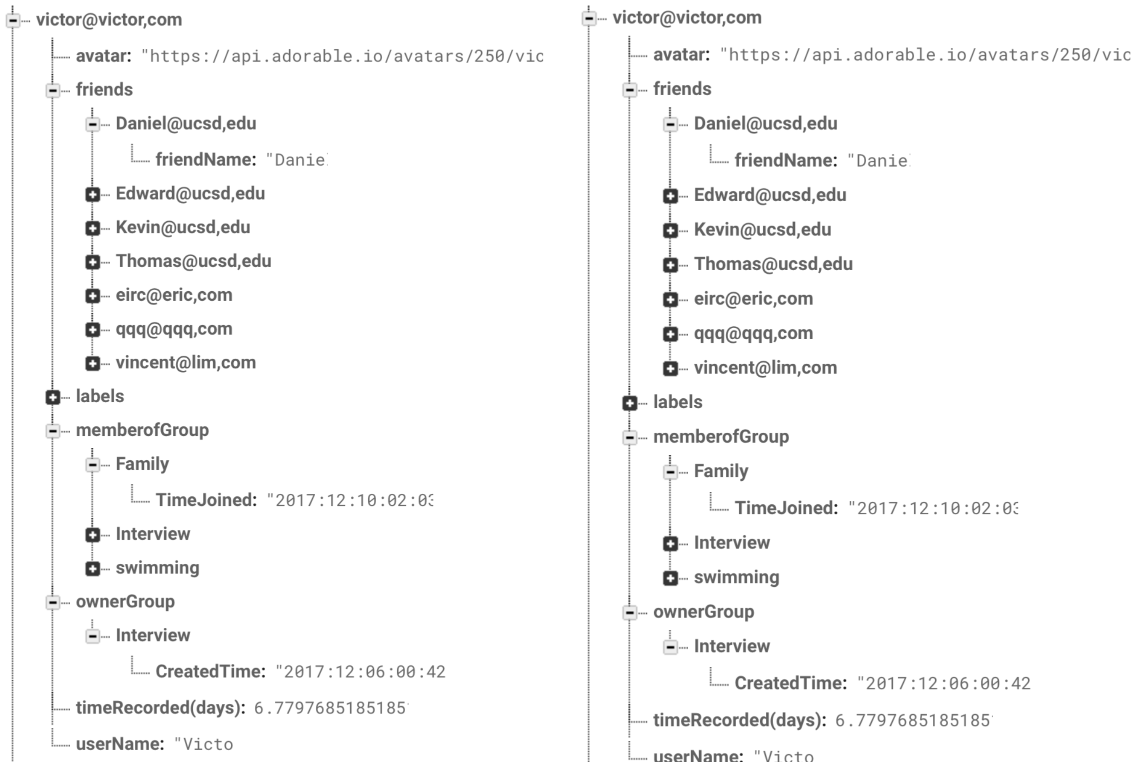


Fig. 3. System architecture of Personality Insights.



```
victor@victor,com
    avatar: "https://api.adorable.io/avatars/250/vic
    friends
        Daniel@ucsd,edu
            friendName: "Danie
        Edward@ucsd,edu
        Kevin@ucsd,edu
        Thomas@ucsd,edu
        eirc@eric,com
        qqq@qqq,com
        vincent@lim,com
    labels
    memberofGroup
        Family
            TimeJoined: "2017:12:10:02:0
        Interview
        swimming
    ownerGroup
        Interview
            CreatedTime: "2017:12:06:00:42
    timeRecorded(days): 6.7797685185185
    userName: "Victo
```

```
victor@victor,com
    avatar: "https://api.adorable.io/avatars/250/vic
    friends
        Daniel@ucsd,edu
            friendName: "Danie
        Edward@ucsd,edu
        Kevin@ucsd,edu
        Thomas@ucsd,edu
        eirc@eric,com
        qqq@qqq,com
        vincent@lim,com
    labels
    memberofGroup
        Family
            TimeJoined: "2017:12:10:02:0
        Interview
        swimming
    ownerGroup
        Interview
            CreatedTime: "2017:12:06:00:42
    timeRecorded(days): 6.7797685185185
    userName: "Victo
```

Fig. 4. Schema of groups and users of Personality Insights database.

Native is that the community of React Native is enormous, highly active and supportive. The fact that the community is large, we were able to use many packages other developers had released. They were also properly documented, and hence we don't have to reinvent the wheel and can easily expand the functionalities of our app within a short period of time. Tech-wise, we don't have to rebuild the application every time, its hot reload feature makes it easy and fast to test out the application when developing. We also took advantage of reconciliation of React Native, which gives us the flexibility to update at any moment during the life cycle of a component.

React Native allowed for the quick development of sophisticated features, and because it is JavaScript, we were able to take advantage of JavaScript APIs and libraries available, including D3 based visualization tools and node modules. Once again touching on reconciliation, we also put React's life cycle model to use to keep an instance of the application up to date when an update was made to the database. If given time, we would have been able to optimize our renders by utilizing React Native's partial render logic. Oh, and did we mention that we all wanted to get experience with React Native because React and React Native are becoming widely used in industry?

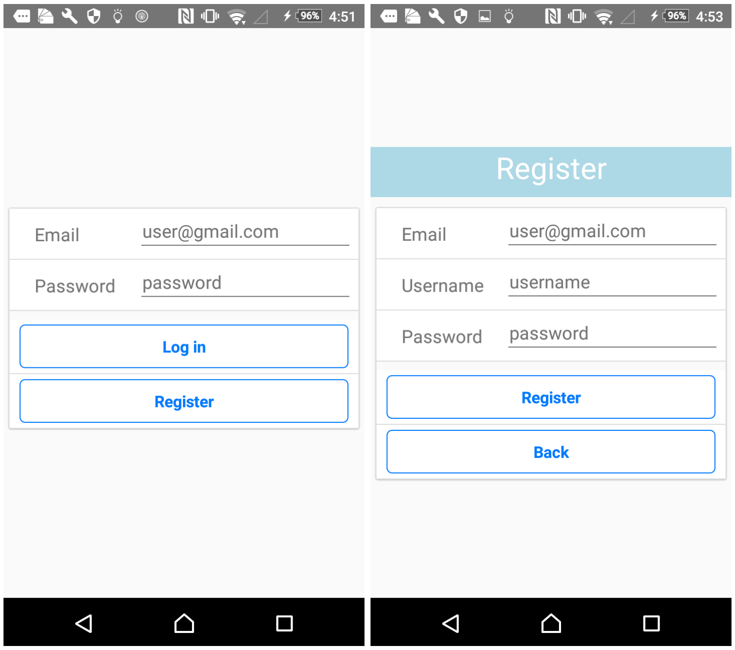
- **Node.js**

  Node.js is a well known and opensource programming language for Javascript on the server side. To improve the efficiency, it is designed to be event-driven, non-blocking and asynchronous, which can optimize the amount and the scale of data transferred between application and server. We also used Express.js, a light but yet agile Node.js Web application framework, to accelerate our developing process, with Express.js, we can easily write route handlers using simple and readable syntax. One of the biggest benefits of Node.js though, is its ability to be scalable. It even has its built-in thread concurrency that doesn't use Operating System threads which compared to the standard today is much faster and easier to use. In fact, no I/O directly happens in Node.js, and there are no worries for blocking since locks don't exist in Node.js. This made Node.js the primary tool that we could have possibly used for our project.

- **Firebase**
  Firebase is a great and easy-to-use platform for many different things; biggest benefit is it helps to create lightweight applications easily and quickly. In Firebase's Real Time Storage, all data is stored in the format of JavaScript Object Notation (JSON) which provides an easy to examine "folder" way of seeing the data. The fact that it uses JSON is a huge benefit to us, because if we ever want to create an API, JSON happens to be one of the most popular systems for storing and exchanging data through web services, which would make it much easier for the user of our new API to be able to grasp how to use it. Also, since JSON is inherently easy to understand, it was much easier for us to fetch the data and create data structures that have key mappings. Most importantly is that Firebase has an elegant user interface to let us inspect the database structure and any dynamical changes which made it easier to make sure there are no bugs in our database functionality. Finally, due to the nature of Firebase, it is much easier to make sure the data is secure since most other types of databases use either PHP or SQL which is known to be prone to data attacks since these languages are not incredibly secure while Firebase doesn't use a specific language directly. Firebase also has many different other services like hosting, but we wanted to use React-Native since it also has a considerable amount of benefits and hence these other Firebase options were not used.
- **Extrasensory**
  Extrasensory is a context-aware application developed by Yonatan Vaizman; a UCSD Ph.D. It senses user behavior and creates a JSON file indicating the 51 labels with corresponding probabilities of what the current user is doing every minute. These contexts are the heart of our application. In Personality Insights, for every JSON file generated from Extrasensory, we chose to update the top five activities with 5 highest probabilities. These top 5 activities, were then added to the user's information about their contexts for a week which is updated every three hours and reset weekly. Using this information, the user could keep direct track of what there are contexts are in our app using radar and bar graphs. We presented the average minutes that user spends on each activity sensed by Extrasensory starting from every Sunday midnight per day per week. We also used extrasensory as a way to demo our code, because we simply did not have a big enough userbase to show a good demo.
- **Adorable Avatars**
  Adorable Avatars is an API that Personality Insights used to create unique avatars for each user. It has a string template like https://api.adorable.io/avatars/285/SOME_STRING.png. When a user registers with there email, we insert the email, which is required to be unique in Firebase, into the template at the position of SOME_STRING. This string will be translated by the API into an avatar, and since the string is unique so are the avatars for each profile. The color or face of avatar generated by the API does not necessarily reflect the emotion of a user. It is just served as a unique image for the corresponding user. But, in our future work, we do want to reflect the emotion of the user with this picture. In next iterations of this program, we want to assign key values about a person's behavior using their contexts, when that is completed the avatars used for each person will not be unique anymore but will be representative of the person's mood which is more useful use for the avatars then now.

4.3 Features

Personality Insights has a lot of features, including data visualization of contexts, competition to change habits between friends, freedom of choosing which contexts to show, and basic social media elements such as friend lists and groups between users. The most prominent reason why we created our app the way we did is to make it fun and useful. Due to lack of time, we were not able to construct what the main point of our project was supposed to be which is showing behavior and competing to change them with other people,instead a user has to use their ability to see what kind of behaviors they have and they need to themselves structure what they should change to better themselves,but it does utilize a way to compete to create change in user's life through competition.



Email user@gmail.com
Password password

(a) Login page

Register

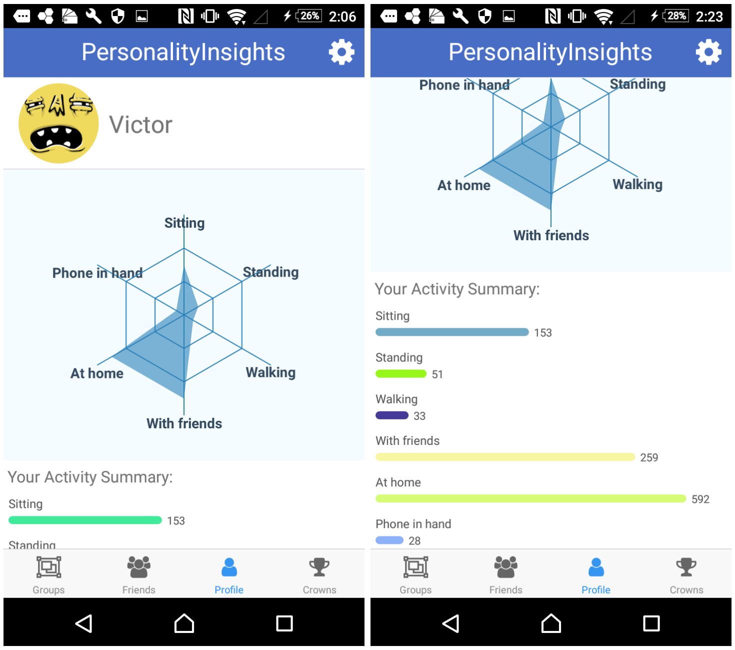
Email user@gmail.com
Username username
Password password

(b) Register page

Fig. 5. Membership

- **Membership**
  We built our membership system by utilizing the authentication supported by Firebase which is both supported on IOS and Android, so we have a few steps done if we ever want to switch to IOS which would be useful to be able to get customers from both platforms instead of just Android. Users can register and login to our application using their unique email address. Users may use any username they are fond of. Currently, each user will be assigned an avatar corresponding to his registered email. In further iterations of this app, security would be changed to protect membership in our program. As of the writing of this report, we simply don't encrypt our user's information. In fact, the way our code is setup up it is straightforward for an admin to find the password of any user using our program, which means our application is incredibly insecure. The things for membership security we are going to add would be hashing our passwords instead of no encryption what's so ever; this should lead to a much secure database we will then discuss more way to make sure attacks on our system are hard to do.

- **Data Visualization**
  Data visualization is the most important feature of Personality Insights, since the goal of the application is to provide the statistics of time consumption in every user's activities.
  On the main page of our application, a radar chart and bar chart is provided to show a default configuration of activities of the user. The user can know the statistic of his activities in a short time and figure which activities they should spend more time on. Pie charts are another important visualization feature of Personality Insights. The user can have a general idea of how he or she performs compared to their friends and also the user can just manually check exactly the activities and context their friends are spending time on. In further iterations of our program, we will try to improve the general "look and feel" of our application. Currently, our application is very simple visually and could be altered to become something considerably better. For instance, the pie charts could be pressed to zoom in and hence would allow a better way to check data a lot more precisely then it is currently being used.



Your Activity Summary:

Sitting 153

Standing

Your Activity Summary:

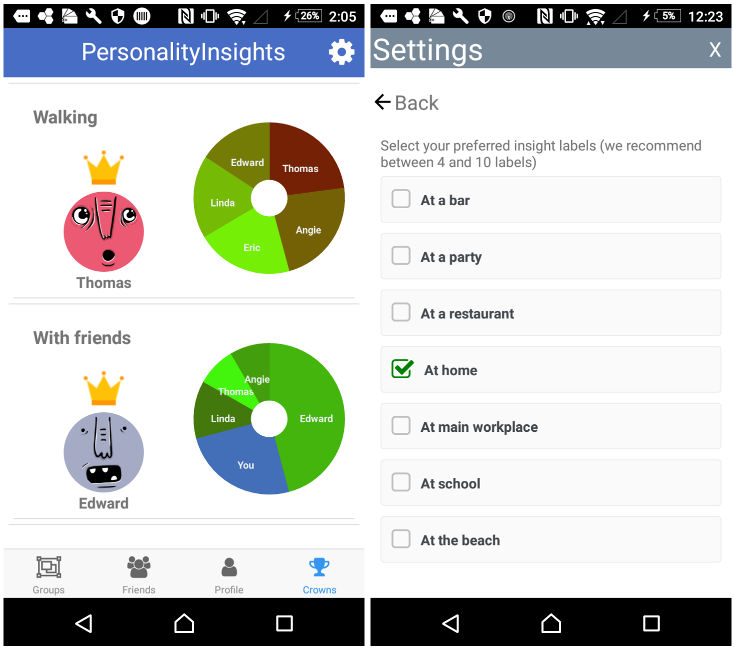
Sitting	153
Standing	51
Walking	33
With friends	259
At home	592
Phone in hand	28

(a) Radar chart (b) Bar graph

Fig. 6. Profile page

- **Profile page**
  Personality Insights aims to give users insights into their personality through graph visualizations of data about what they do in every week gathered via ExtraSensory. Users can see a radar chart and a bar graph that is updated every three hours and then reset at the start of the week which gives the user a way to see of how they have spent their time. This is currently also the "main page" of the user when they log in. From this page, and every other page user has access to any and all pages. This application is always set to vertical view which for now provides limited space for a user to see their information in the forms of the graphs on this main page. One of the big things we can do to make it considerably better is to allow

horizontal view via the phone. If we have a horizontal view, it is pretty easy to see that both graphs would fit without the need for a person to scroll down which would make it easier to look at.



Select your preferred insight labels (we recommend between 4 and 10 labels)

□ At a bar

□ At a party

□ At a restaurant

☑ At home

□ At main workplace

□ At school

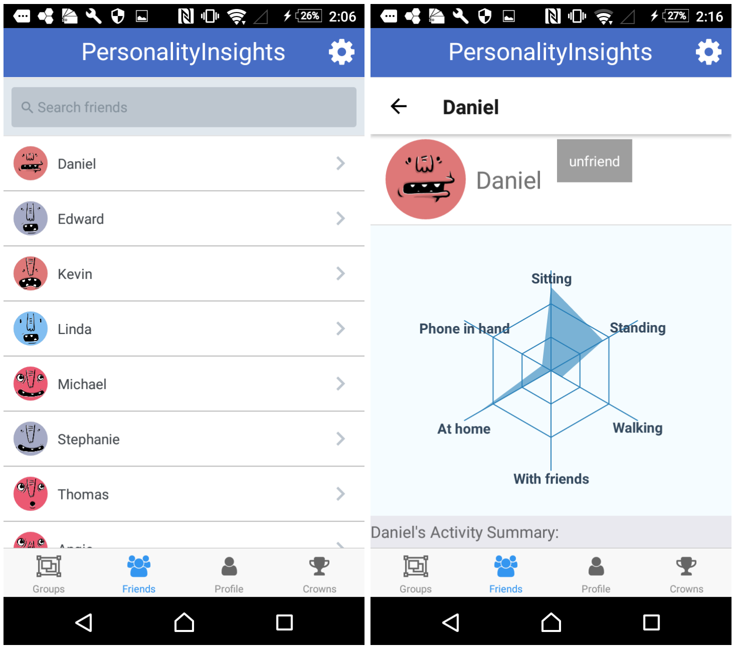
□ At the beach

(a) Crowns (b) Settings

Fig. 7. Crown page and Settings page

- **Freedom of configuration**
  In the setting module, the user can choose the activities that he or she is interested in to show on their profile page which adds a freedom of choice and scope to our program.For instance, if the user is interested in who has "won" any activity currently aka. The crown holder. These users can choose which activities from settings to see which will show them also who has won the activity as of right now. In order to get the best-visualized result, we at Butter Croissant recommend choosing the minimum of four to the maximum of ten activities out of 51 activities, this is because the way our data is shown to the user it is not optimal to have more than ten activities out of those 51 shown as well it is not optimal to have four activities either. When we work on this application again, as said earlier, we will change the look and feel of our application which then it is likely probable that our application will be able to support both less than four and greater than ten activities.
- **Crowns page**
  The crowns page is showing the leadership scores depending on the activity via pie chart and user's "crowned" avatars. Users can see insights about their friends, allowing for activity comparisons- " oh I wonder how well I have been doing compared to my friend Isabelle, maybe I'm finally caught up with her? Oh wow! She has been running a lot more then I have, it appears I need to step up my game if I want to be as cool as she is. ", this is an example of the user reaction of using the crowns page. This gives users motivation to appear on the leaderboards, and it allows users to see which of their friends share similar

interests and since these are the top users per activity also shows that their friends do say activity a lot. Not only does the crowns page," crown" the person that has been doing an activity the most, the pie chart corresponding to an activity also shows the top five. This means a user might even become embarrassed to the point it changes their behavior due to the fact they might not even be in the top five, of course, if you have less then five friends this is not possible.



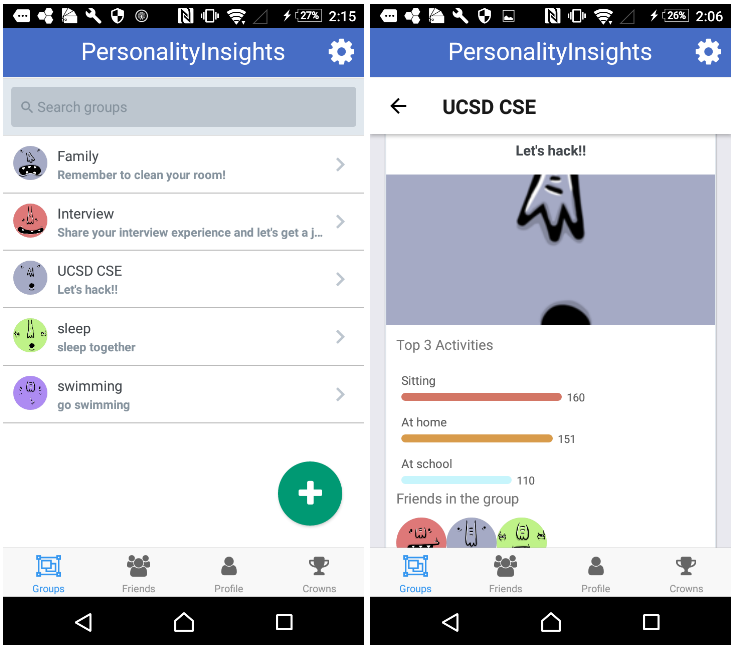
(a) Friends page

(b) A friend's profile

Fig. 8. Friends page and friend's profile

- **Friends page**
  Our app is meant to be a Social Media application, for instance, we see it as an add-on to Facebook or an add-on to a dating application. So friend's page is another essential feature of Personality Insights. By using this page, users can connect with other users to acquire the ability to be able to see their friend's profiles; which facilities our idea of motivating people to change their habits by comparing their habits with their friend's habits. In this page, users can search other users by writing down specific search queries for a username. This search gives the user back; a list of people who have a similar username to what has been typed in the search bar, if they are a stranger or a friend, and all the specific cute avatars per person. Then the user can see a specific user's profiles by clicking one of the rows that were returned by the search query. By clicking on these specific rows, the user can see what the possible new friend is interested in and the groups this new possible friend is a member of, then the user can decide if there is a point to add this new friend by clicking the add friend button. Similarly, we can also use the search option to find a person who is already a friend of the user. In this specific case, this person can again see the user's profile but this time also has the option to delete this said person from the friend's list. One of the big things that we are missing for this page is the ability to be able to message each other. Without that, our friend's list does not

add much to the functionality of our program; so probably one of the most important things we need to look into for next time is adding chat to our program. Also, we are reiterating, look and feel of this page can be improved for instance the unfriend and friend buttons, they are made with the intention to look nice.



(a) Groups page

(b) Detail of a group

Fig. 9. Groups page and group's information

- **Groups page**
  Groups is another social media element we have which as was stated is incredibly vital since our application is supposed to be a social media one. Just like on the friend's page, searching has very similar inner workings in the group's page. Users can search groups with the name given to it by whatever some other user has created it in the first place. The search option again returns rows with the name of the groups. As of right now, this group pages is nearly identical to the friend's page search. We even use the same avatars in the same way we used for friend's profiles but now for groups.Once one of the search query rows has been clicked, the user can see groups' objective and statistics which is a sum of data from all the users in the group. For now we chose to show the top 3 activities, but we, of course, can add a lot more functionality to this page, for instance just like with radar charts of a person's profile we can figure out how to show a lot more than just the top three, now it simply isn't possible because it would look very messy . To do this, we need to somehow to incorporate a better style to support the ability to see more then three top activities or this page will look silly if we allow more then three as of right now. Furthermore, Users are also able to see their friends that also joined those groups; then afterward they can decide whether they want to join or not. Users can create their group. Since one can join groups, he also has the option to leave any group that they decided to join . Since friend and group pages are similar in what they do probably one of the biggest

things we can work on it besides the need to change how it looks is adding chat functionality. In fact, chat functionality is probably the biggest thing in this whole application that is missing.

5 TESTING AND EVALUATION

5.1 ExtraSensory Dataset Analysis

5.1.1 The ExtraSensory Dataset.

The ExtraSensory dataset [9, 10] is mainly provided by Dr. Yonatan, the creator of ExtraSensory Application, which contains data from 60 testers. Each tester has a unique identifier(UUID) for us to identify them. From every user, it contains examples for a whole week.(Some of the testers may only have data for several days) Every example is a JSON file include 51 labels and their corresponding probabilities which indicate the confidences of the activities, typically taken in the time-frame of 1 minute with some time gaps.

Though the ExtraSensory dataset contains much more detail information, such as the typical sensors, the self-labeled information. In this paper, we only use the 51 labels and their corresponding probabilities, and furthermore, generating the minutes per day information for each tester. In this case, we avoided the problem of time gaps between data and overcame the issue of some lazy testers which only have several days' data.

5.1.2 Motivation.

Our motivation is clear, to get the features of the testers, and put those testers with similar features into the same group. As described in section 5.1.1, we have proceeded the ExtraSensory dataset into 60 testers with their minutes/day information for 51 labels(activities). For the "feature" we have mentioned above, we interpret it as the character of a person, more specifically, the habit of a tester. For example, if a person has more than 20 hours running data recorded for a week, we can tell that this user may like running a lot and has a habit of physical exercises. By putting those testers with similar features into the same group, we mean creating a group that the members all like running a lot.

5.1.3 Methods.

In this section, we will introduce our method by introducing some certain problems we met in processing this task, and also describing how we overcame these issues.

- **Neural Network or Clustering**
  To be honest, the intuitive method we came up with is using Neural Network tools to train a classifier that can classify testers into different groups. However, there are many problems came along with it. Firstly, we can not use supervised learning, which is a much more convenience tools than unsupervised one, since we don't have labels for each user. By using unsupervised learning, we find out that our dataset is insufficient and there is a huge probability that we may get over-fitting while we increase the iteration number. Secondly, comparing to another method such as clustering, the neural network is hard to update in real-time, which we may have a big trouble when we began to have a huge dataset.
  However, we may still use neural network method in the future, for instance, if we get bigger groups which users manually add into. Thus, we can use machine learning method to learn the features of those certain groups. Thus, we chose to cluster to classify the dataset.
- **Data Characteristics**
  After we processed the original dataset and got our user data, which is 60 testers with 51 labels each, and each label contains minutes/day information for that label, we find out that our data have two significant characteristics: imbalanced distribution and high dimension, which we will discuss their pros and cons in the following content.

(1) Imbalanced Distribution
We can tell from the Figure 10 below that the dataset has lots of Zeros and several labels with huge values. This feature has both advantages and disadvantages. For the pros, it is easier to separate since we

have significant value for each tester, which means we can easily differentiate different habits. For the cons, which is a bit hard to observe that the label with huge values will overwhelm the minor difference, for example, if we have a tester with activities that have portions as 98, 1, 0.1, 0.01, we can only get the information of that 98. However, the difference between 1, 0.1 and 0.01 is also important in some case. By using the data directly, we can only obtain a coarse feature of each tester and lose a lot of information in the meanwhile.

Plus, there is another problem with the imbalanced dataset that different tester has different labels with huge values, which will also make it harder to extract the same feature among testers. There is one solution to this problem, normalization, getting the portions of activities for each person. However, this solution will cause a scaling problem that we presume all the users spend the same time on the ExtraSensory Application, which is not. Thus, for instance, if there is a tester A does 2 hours running which takes 10% of his/her time and a tester B who also does 2 hours running but only takes 2% of his/her time. In this case, we won't regard their running as the shared feature, because we do clustering on the portions after the normalization.

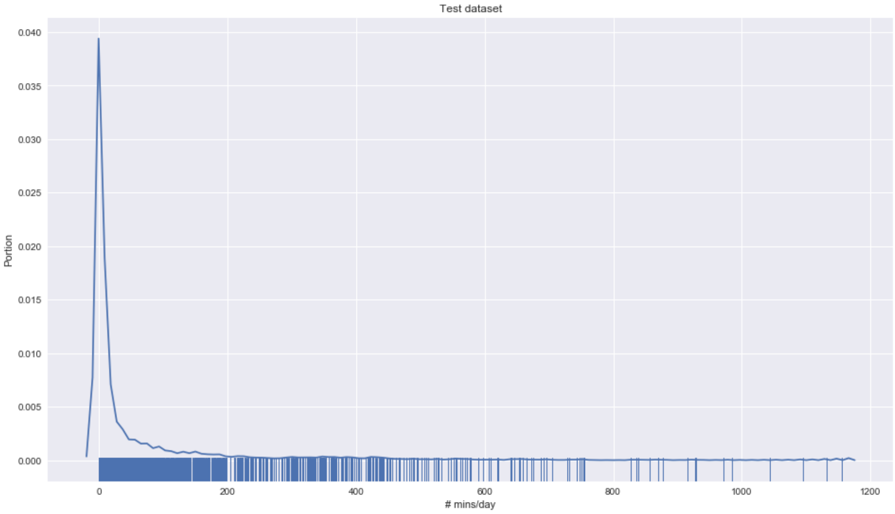


Fig. 10. Test dataset overview

(2) High Dimension

As we can see in Figure 11 and the dataset described above that we have a high dimension dataset which contains lots of zeros. Thus, the intuitive thinking is to delete some labels with all zeros to boost up the performance of clustering. However, after we generate the figure for non-zero labels (See Figure 12, here we regard activities that have more than half an hour per week as non-zero activities), we find out that there is no label that has all zero values, in another word, all types of activities has someone's data on it. As a result, we can not use this method. To solve this problem, we chose to use PCA (Principle Component Analysis) on the dataset, which is described in the next section.

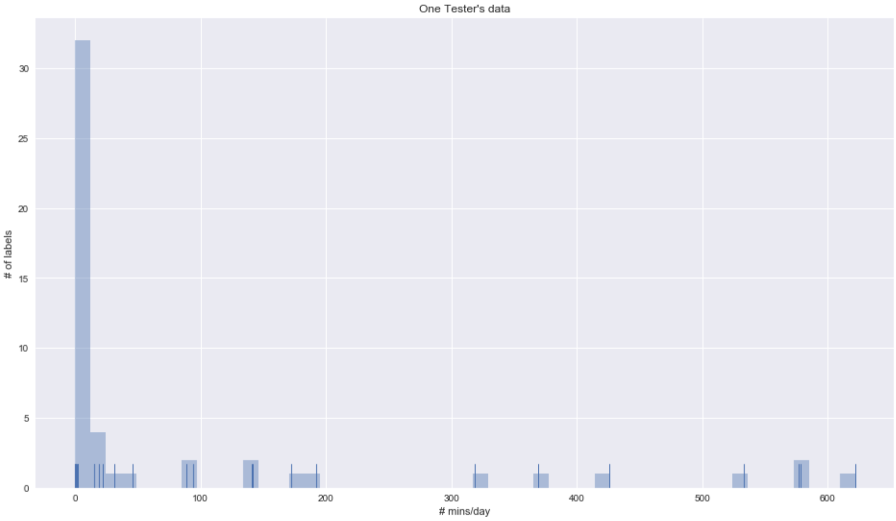


Fig. 11. A specific tester's data

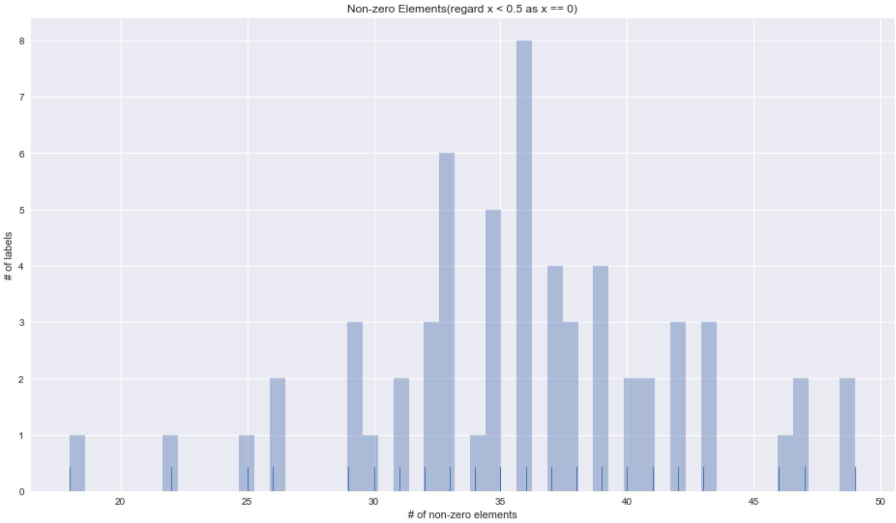
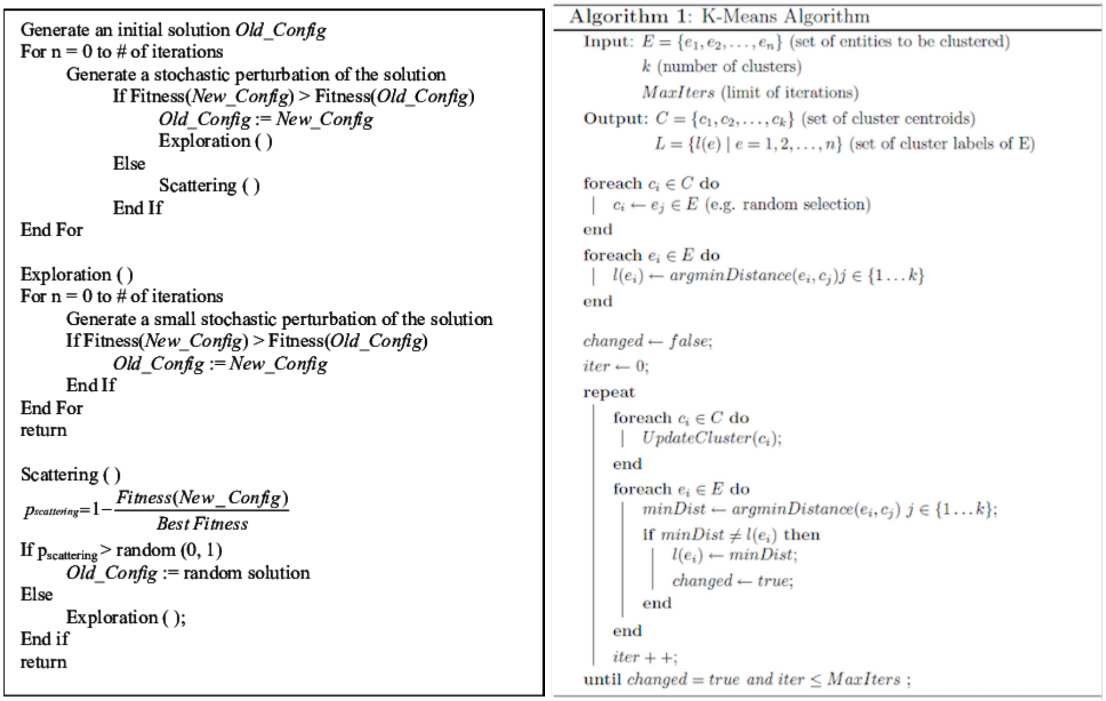


Fig. 12. NonZero labels

- **Principal Component Analysis**
  As mentioned above, we chose PCA to reduce the dimension of the dataset. [8] This method also came with the pros and cons. For the benefit, we can tell from the dataset that it has inherent feature that some labels are highly correlated and some are exclusive from others. For example, cooking at home, sitting in a car, etc. And more, you can not run in a car, nor can you talking while sleeping, or barely. Those are all the labels provided by the ExtraSensory dataset. For the defect, it is obvious that the PCA will change the

meaning of each axis and lose information. And with the PCA fan-out become smaller, the information mixed in the axis will be vaguer that makes the clustering more difficult, in another word, makes the result worse. Thus the fan-out dimension of PCA is important, but for the convenience of visualization, we use the fan-out dimension of 2 in PCA (See Figure 14). The algorithm of PCA is shown in Figure 13a.[6] And it is interesting that there is neural network method to do PCA, which explain again that neural network may perform better than the cluster, however, since we do not have enough data, we only implement clustering here.



```
Generate an initial solution Old_Config
For n = 0 to # of iterations
        Generate a stochastic perturbation of the solution
                If Fitness(New_Config) > Fitness(Old_Config)
                        Old_Config := New_Config
                        Exploration ( )
                Else
                        Scattering ( )
                End If
End For

Exploration ( )
For n = 0 to # of iterations
        Generate a small stochastic perturbation of the solution
        If Fitness(New_Config) > Fitness(Old_Config)
                Old_Config := New_Config
        End If
End For
return

Scattering ( )
$p_{scattering} = 1 - \frac{Fitness(New\_Config)}{Best\ Fitness}$
If $p_{scattering}$ > random (0, 1)
        Old_Config := random solution
Else
        Exploration ( );
End if
return
```

(a) PCA Algorithm[6]

```
Algorithm 1: K-Means Algorithm
Input: $E = \{e_1, e_2, \ldots, e_n\}$ (set of entities to be clustered)
       $k$ (number of clusters)
       $MaxIters$ (limit of iterations)
Output: $C = \{c_1, c_2, \ldots, c_k\}$ (set of cluster centroids)
        $L = \{l(e) \mid e = 1, 2, \ldots, n\}$ (set of cluster labels of E)

foreach $c_i \in C$ do
    $c_i \leftarrow e_j \in E$ (e.g. random selection)
end
foreach $e_i \in E$ do
    $l(e_i) \leftarrow argminDistance(e_i, c_j) j \in \{1 \ldots k\}$
end

$changed \leftarrow false$;
$iter \leftarrow 0$;
repeat
    foreach $c_i \in C$ do
        $UpdateCluster(c_i)$;
    end
    foreach $e_i \in E$ do
        $minDist \leftarrow argminDistance(e_i, c_j)\ j \in \{1 \ldots k\}$;
        if $minDist \neq l(e_i)$ then
            $l(e_i) \leftarrow minDist$;
            $changed \leftarrow true$;
        end
    end
    $iter++$;
until $changed = true$ and $iter \leq MaxIters$ ;
```

(b) K-means Clustering Algorithm [2]

Fig. 13. Algorithm for PCA and Clustering

- **K-Means Clustering**
  Here, we use K-means clustering to classify the PCA. [2] For the convenience of visualization, we chose dimension of 2 here to present the clustering result, which is in Figure 15. Here, different color patches mean different clusters, and each point is a tester, the blue crosses within the patches are the cluster centroid.

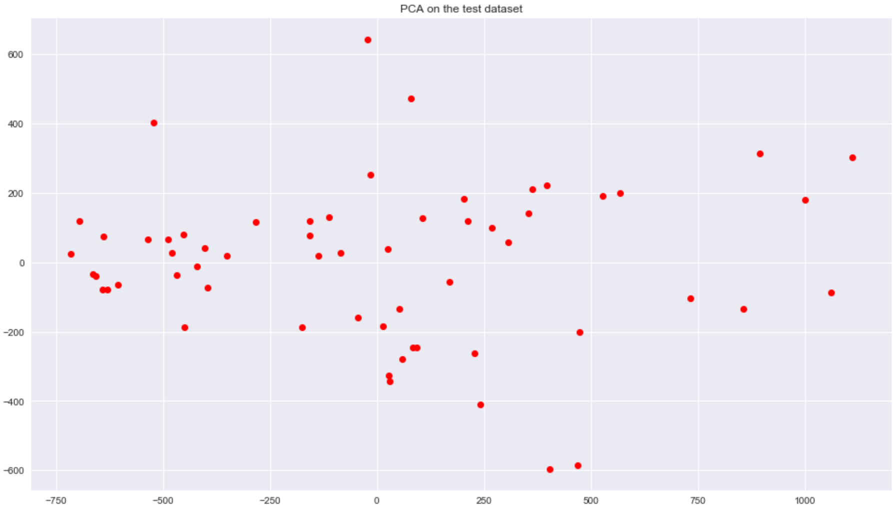


Fig. 14. using PCA to reduce the dimension into 2

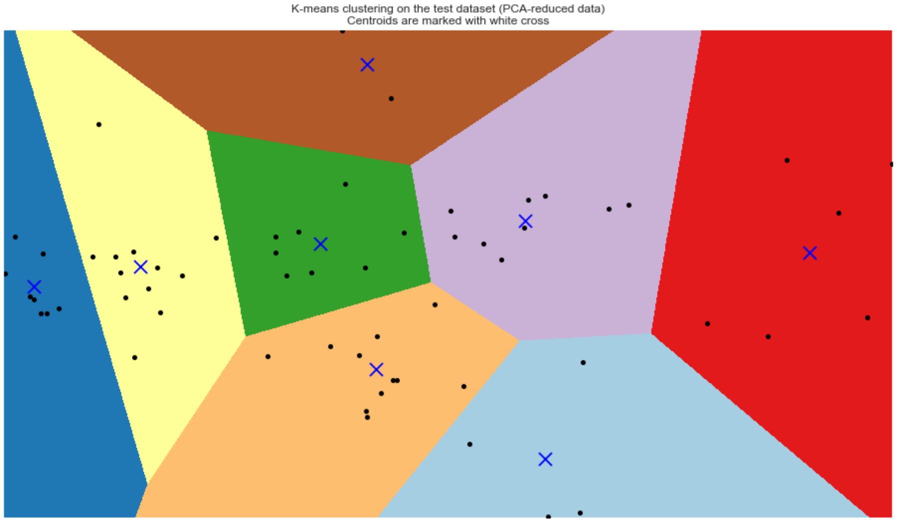


Fig. 15. The result after K-means clustering (dim=8)

5.1.4 Evaluation.

Since we demonstrate our method in the dimension of 2, the result is not the best one. However, we can tell from the Figure 15, the clustering result is not bad. And we can get cluster features in high dimension PCA and clustering, such as the group of testers like running a lot, and another group is always sitting, etc.

5.1.5 Problems.

There are several problem still remains in our method. Firstly, we use one day as the time frame, however, usually testers did not open the ExtraSensory the whole day. As a result, the features we extracted may be incorrect. For example, only a few tester use ExtraSensory while they are sleeping, and the sleeping could take a huge portion of the whole activities of that person, which will result in the scaling problem we mentioned before. Secondly, in our method, by doing clustering, one person can only belong to one group, since clustering exclusively separate them. But in real situation, one person can belong to different groups. Neural Networks won't have this problem.

5.1.6 *Inspiration.*

By clustering the data, we separate the testers into different groups based on their features. This idea inspires us that we can use it to do group suggestion, suggesting groups based on user's features. Besides, we can also compare the features of a user to the features of the group he/she is enrolled in, to see if he/she is belonging to that group and tell the user how far he/she is to that certain group. Furthermore, in each day, we can get a user's daily activities, and we can show the user his/her is shifting from one group to another group based on the group's features. This is like a point in Figure 15 is shifting from one patch to another patch, which is interesting.

5.2 User Testing

We conducted one round of user study, where each user had their hands on Personality Insights for about 5 minutes with a 5-minute tutorial and 5-minute follow-up interviews.

5.2.1 *User Base.*

We recruited three people, two male, and one female, aging from 20 to 25, who are all UCSD students and are excited to test our App. They all have substantial experience with Smartphones and are familiar with and active in nowadays social media.

5.2.2 *Interview Questions.*

Here are the questions we are required to ask users:

- What do you like about this App?
- What do you not like about this App?
- In what application would this App be useful?
- What do you wish was added to this App?
- How do you think this App can change your life?

5.2.3 *Feedback.*

Here we will talk about some feedback from these interviews, and how we interpret the comments, and what we can thus change for better.

"It is cool to see what activities other people are doing." We think this is because people nowadays are used to the interaction with others on social media, and our App provides useful information on others they care about.

"I would like to use this App the find out what gift I should give to my friend for their birthday." We believe our App provides a clear picture of a person's interest through activity data visualization, giving not just the person, but also their friend more details about someone's personality.

"I wish I can chat with other people in the App." This is an interesting suggestion, and as we mentioned, Personality Insights can be an add-on to the existing platform, such as Facebook, which already has a messenger to allow chatting directly with friends. Besides, this feedback reflects the fact that users would also want to use this App to connect with others when they see others' activity data, which is one of our intentions.

"There are a lot of social media exist, so there is no need for a new one." We do agree with this comment to some

degree, and again we are glad to see Personality Insights be part of a popular social media so that we will not re-invent the wheel.

6 COLLABORATION

6.1 Team Structure

Our team consists of two graduate students and three undergraduate students, who are all enrolled in Computer Science and Engineering Department in UC San Diego. The two graduate student, Weiyuan, and Chenglin rotate their role as the scrum master for the team. Our team is structured into two small groups, one is front-end development, with Vincent and Chenglin, while the other is back-end development, with Weiyuan, Daniel, and Chao. Each week, we will have a sync-up meeting to communicate what's finished and what should be implemented in the next sprint.

6.2 Team Member Contribution on PI

Weiyuan Wen:
For the application, Weiyuan brainstormed ideas with other teammates. He set up Profile page and Crowns page UI with basic dummy data being displayed from the backend server. He fleshed out the basic login and register page along with Firebase authentication implemented. Read Extrasensory data and send to the server. He accessed Android filesystem to fetch the ExtraSensory data, parsed them in the server and updated the current user's consuming the time of top 5 activities in the database. He found the Adorable Avatars API and used it to assign a random avatar to each user and group that newly created. He implemented the basic UI base for group page and friend page along with all server-side route handlers, which includes finding user's friends in each group and send structured data back to render. He created stack navigator of friends and groups page with Daniel and connected features such as add/unfriend user and joined/leave group implemented by Daniel. He implemented the searching feature for both friends and groups so that the user can search other users and groups. He set up friend's or stranger's profile like user's profile with a feature to show which groups did a user join in. He set up the page to show details of a group, including objective, top 3 activities, and friends in a group.
Daniel Kostinkiy:
Daniel is the Firebase go-to person in our team and who planned most meetings and helped with the brainstorming. He set up Firebase and created folders during the first week. He then implemented the API to add/remove friends and join/leave the group, or most of the backend required for the rest of project. For most of the rest of the project, Daniel collaborated immensely with Weiyuan Wen. With Weiyuan, Daniel got up to date with how React-Native and together with Weiyuan on one crazy all-nighter in the middle of the quarter hacked out friend's page and some of the details and aspects of Group. He then helped to figure out some of the bugs in those pages and fixed them for the team. For the most part, Daniel helped on the back-end with Firebase and tried to make sure that everyone is up to date with what they need to be doing by being also the guy who helped with all of the ideas. Finally, even on this report, I worked with Weiyuan for the System's development section which includes system architecture, the technology used and features that Personality Insights has.
Chao Wang:
Chao is a member of back-end group, and he did most thing around the ExtraSensory App and the provided dataset. He tested the ExtraSensory App and extracted some real data. Furthermore, he extracted the information from the test dataset and convert it into data form we used in our database, then updated the database, created the test users. Moreover, he then analyzed the test dataset with his knowledge, extracting the feature of each tester and put those testers with similar features into the same groups, by this way, creating groups with certain features.
Vincent Lim:

Vincent is the React Native expert. He single-handedly set up the entire development environments (React Native and Firebase Hosting). He then implemented front-end routing, login/registration, navigation, settings, radar chart, with nice taste on UI design. He assisted to searched and fixed bugs related to the front-end and assistant in the implementation of other parts of the front-end as well. He is also the main presenter during demo day!
Chenglin Lin:
Chenglin serves as the secondary front-end developer and worked closely with Vincent. He develops the piechart view in Crown page and barchart view in Profile page. Wenyuan and Chenglin take the roles as the scrum master to monitor the progress of each team member. They set up conference rooms for weekly sync-up meetings, and led the discussion.

6.3 Team Member Contribution on Report

Weiyuan Wen:
System Development - Architecture, Technology used, Features
Daniel Kostinkiy:
System Development - Architecture, Technology used, Features
Chao Wang:
Testing and Evaluation - ExtraSensory Dataset Analysis
Vincent Lim:
Introduction, Motivation and Background, first half of Design, and Future Work
Chenglin Lin:
the second half of Design, Testing, and Evaluation - User Testing, Collaboration, Conclusion

6.4 Collaboration Methods

In the project development, we tried to adopt the principles of Agile software development. We have set up Trello task boards, showing weekly sprint goals and progress. Weekly meetings were scheduled based on everyone's availability, and in the meeting, we would start by going through what we have or have not done from last week. We then moved forward to this week's milestones and adjusted them through reflection on last week. If something unexpected happened, e.g., broken deployment, we could call out to a Facebook group chat we created to call out and resolve the problem immediately.

6.5 Collaboration Issues

- Difficulty Adopting Agile: We had a hard time keeping up with Agile methodology, mainly because each of us could not spend full-time on this project. We have to simplify many procedures, like stand up meeting and the retro meeting was combined into our weekly meeting. Some of us could not finish the milestone of some weeks, and thus we simply had to drag most of the items to next week, which was not an ideal situation for a sprint. So, it would be up to the scrum master to give a lift or follow up with some team members to ensure progress.
- Task Assignments: Since each of us has different expertise on the tools, we mostly assigned tasks to what the team member is comfortable with so that we can have a working prototype in five weeks. However, one of us wasn't familiar with any part of the tech stack we used. It became a trouble for us to assign an appropriate task for him. It is not fair not to assign him anything, but it is almost impossible to for him to finish anything even with our help, for some reason. The problem was not resolved even until now. The other four of us were patient enough to include him in our discussion to let him participate. The efforts are precious to the rest of us.

7 CONCLUSION AND FUTURE WORK

Although we weren't able to get too far with our implementation, we believe we have started a new idea that can be expanded upon.

In terms of Personality Insights, we hope that development of the application continues. Some features we would like to add to it include are:

- Integration of data from social media platforms.
- A personality-classification algorithm to cluster users into personality types for better, more valuable insights. This would come with descriptions and characteristics of the different types.
- Integration of tests such as the Myer Briggs test and the Strengthfinders test to set a baseline personality that context data can enhance.
- Integration of mood detection to allow us to display avatars that match a user's mood.
- A suggestion algorithm based on a user's personality, which can include groups, jobs, shopping items, and activities.

Initially, we considered Personality Insights to be a possible add-on to existing social media platforms. That is still a possibility. As mentioned above, we can take data from existing platforms and incorporate them into Personality Insight's personality-classification algorithm. There are already tools that can predict moods and personalities based on tweets[4] and Facebook posts[7]. And it won't be long before we can pull some context data from pictures using image recognition as well.

We also thought about altering the idea behind Personality Insights. Keeping it the way it is now and not including a personality-classification component to the application, we essentially have a behavior data visualization application. This could potentially be called Behavioral Insights, which has it's own use cases that mostly pertain to health and social networking.

Another possible direction for Personality Insights is to follow the social game route. By having users compete against each other, this can give users motivation to do certain activities more. This would probably be the lifestyle equivalent of the popular application 'Swarm' by FoureSquare, which provides incentives for checking-in, adding photos, and leaving reviews of places you visit or pass by.

We also considered making Personality Insights a dating application. It seems to have already implications of helping users find people similar to them, which could directly translate to finding a compatible partner. We had a lot of feedback relating to this topic, but we do not think that we, ourselves, would take Personality Insights in this direction. However, we are open to others doing so on top of what we have created thus far.

We believe Personality Insights classifies a new realm of ideas: context recognition from a social perspective. Other possible outcomes of this field include motivational tools and games that lead users to healthier lifestyles. With the emergence of context recognition tools and the advancement of sensors on our phones and in the environment, applications like Personality Insights may become the norm in the future.

Imagine a world where personality information was available to a protected extent. Jack would be able to find the perfect fraternity to join, and Jim would be able to find what gifts to get his book-worm friends. People would be able to understand why they excel at certain tasks and build their strengths based on their personalities. Jobs would be able to show their overall personality which could attract potential employees. Applications like tinder would be able to use personality information as another layer to add to their matching algorithm. Understanding personalities is a skill that helps us in many ways that we might not even realize. Personality Insights augments this skill, allowing us to focus not on analyzing personalities but making decisions that affect our personal and social lives.

In conclusion, Personality Insights is the start of an idea, a potential use case for context data in the social realm. We believe that Personality Insights, if further developed, can become a useful augmentation of our psychological skill to understand personalities. We believe this tool, Personality Insights, can be handy in our everyday lives and can even become ubiquitous. Although we had a limited amount of time to work on this application, we learned a lot and have been inspired to continue learning about how context data can help us. Our prototype of Personality Insights thus far is nowhere near production ready, but we believe the idea to be novel and the benefits to be significant. We hope you think so too and we look forward to seeing what new technologies will build off of what we have started.

ACKNOWLEDGMENTS

The authors would like to thank the teaching staff of CSE 118 at the University of California, San Diego, during Fall Quarter 2017. We had a great time this quarter and learned a great deal about the future of context recognition technologies.

REFERENCES

[1] G Susanne Bahr and Richard A Ford. 2011. How and why pop-ups don't work: Pop-up prompted eye movements, user affect and decision making. *Computers in Human Behavior* 27, 2 (2011), 776–783.

[2] Lucas Cunha. 2009. Data Mining Algorithms In R/Clustering/K-Means. (Dec 2009). https://en.wikibooks.org/wiki/Data_Mining_Algorithms_In_R/Clustering/K-Means Online; accessed 15 Dec 2017.

[3] Robert Ward Leeper and Peter Madison. 1959. Toward understanding human personalities. (1959).

[4] Michael J Paul and Mark Dredze. 2011. You are what you Tweet: Analyzing Twitter for public health. *Icwsm* 20 (2011), 265–272.

[5] Daniel Riff, Stephen Lacy, and Frederick Fico. 2014. *Analyzing media messages: Using quantitative content analysis in research.* Routledge.

[6] Wagner Sacco, Hermes Filho, and Pereira Claudio. 2007. COST-BASED OPTIMIZATION OF A NUCLEAR REACTOR CORE DESIGN: A PRELIMINARY MODEL. (01 2007).

[7] H Andrew Schwartz, Johannes C Eichstaedt, Lukasz Dziurzynski, Margaret L Kern, Eduardo Blanco, Michal Kosinski, David Stillwell, Martin EP Seligman, and Lyle H Ungar. 2013. Toward Personality Insights from Language Exploration in Social Media.. In *AAAI Spring Symposium: Analyzing Microtext.* 72–79.

[8] Jonathon Shlens. 2014. A Tutorial on Principal Component Analysis. *CoRR* abs/1404.1100 (2014). arXiv:1404.1100 http://arxiv.org/abs/1404.1100

[9] Yonatan Vaizman, Katherine Ellie, and Gert Lanckriet. 2016. Recognizing Detailed Human Context In-the-Wild from Smartphones and Smartwatches. *arXiv preprint arXiv:1609.06354* (2016).

[10] Yonatan Vaizman, Katherine Ellie, Gert Lanckriet, and Nadir Weibel. 2018. Data Collection In-the-Wild with the ExtraSensory App: Rich User Interface to Self-Report Behavior.

[11] Rainer Wasinger, James Wallbank, Luiz Pizzato, Judy Kay, Bob Kummerfeld, Matthias Böhmer, and Antonio Krüger. 2013. Scrutable user models and personalised item recommendation in mobile lifestyle applications. In *International Conference on User Modeling, Adaptation, and Personalization.* Springer, 77–88.