|  |
| --- |
|  |
| CIS700-Final Project Facebook Matchmaker |
| Amol Patil Ankit Junnarkar |
|  |
|  |
| **12/15/2013** |

|  |
| --- |
|  |

**1. INTRODUCTION**

Humankinds are easily influenced by the people around them, with whom they spend most of their time or people whom they like the most. There is a good possibility that such humankinds follow likes and interests of compatible people. Facebook activity is a good indication of a persons close social circle and can be used to find his/her compatibility with them.

We have developed a predictive application which analyzes the facebook profile of a person and predicts compatible people and interests for a given person based on his/her Facebook activity. To do so, the application would perform sophisticated analysis of person’s Facebook activity which includes analysis of the following:

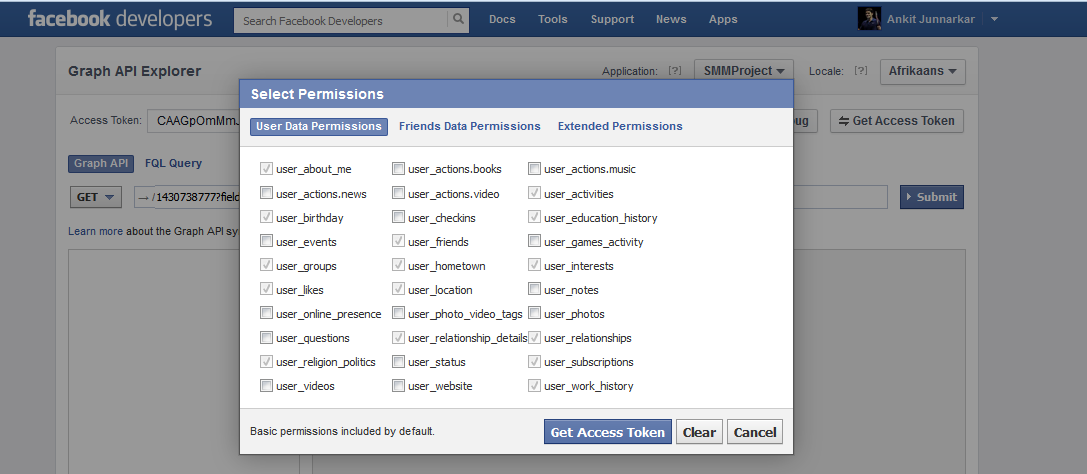
* Zodiac compatibility
* Name compatibility
* Likes
* Current location, hometown
* Languages known
* Educational history
* Age group
* Gender
* Photo tags and check-ins

Interests of compatible people found from the above analysis are suggested to the person. We plan to use the Facebook/Graph/User API which gives access to all the above mentioned attributes of the user. Algorithm for numerological and zodiac compatibility can be developed using the user’s birthdate. Name compatibility algorithm is based on the sequence of vowels and consonants in the name. For example, Aries and Taurus have strong zodiac compatibility, number 1 and 6 have strong numerological compatibility, name Adam and Eve have strong name compatibility (about 86%). Facebook page likes (including favorite music, athletes, sports etc.) can be determined using the facebook graph API. All the above mentioned attributes are given a score out of 10 and then all of them are added to give a final score out of 100. The combination of all the above attributes can give a strong compatibility analysis for the user.

This application is purely for entertainment purpose. This application would entertain the user by showing him/her the list of most compatible people and their interests, if not his/her close social circle. Apart from the entertainment value, knowing the interests of a person could be very useful when it comes to deciding gifts for him/her or shopping for special occasions. It would also help improve advertisement by suggesting the user, likes of his/her close friends. And more over there is no such application for Facebook yet. To filter the number of outputs, we have made a provision where the user sends the threshold compatibility value (in percentage) and only the people whose compatibility score is more than that value will be displayed to the user.

**2. DEVELOPMENT APPROACH**

As the name suggests, we have used the facebook graph API for the project. The Graph API is a low level HTTP-based API which is the primary way to get data in and out of Facebook’s social graph. By default not all the fields in a node or edge are returned when you make a query. So, you need to be signed in with an access token. The procedure to get an access token is very simple. You have to create an app on developers.facebook.com. You need to sign in with your facebook account. Once the access token is generated, you can determine which permissions are needed in this access token by looking at the Graph API reference for the node or edge that you wish to read as shown in figure 2.1. The permissions include all the data which is publicly available on the user profile. You can also access the public information of the user’s friends.

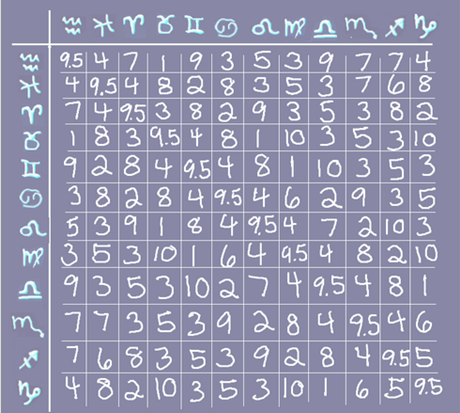


**Figure 2.1: Getting the access token and setting the permissions**

After creating the app, we now have access to the user’s public information (for which we have set permission). The facebook platform for python is available at <https://github.com/pythonforfacebook/facebook-sdk>. The next step is to calculate the compatibility attributes. They are discussed below:

***2.1 Zodiac Compatibility:***

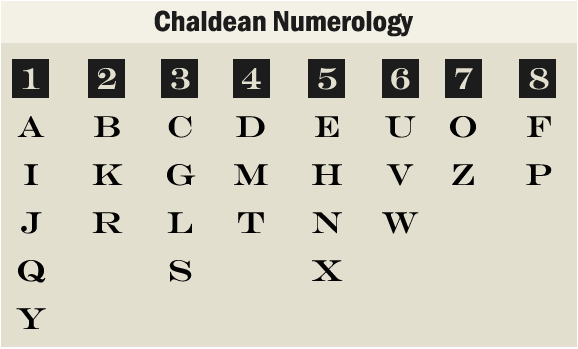
Zodiac compatibility calculation is based on the birthdate. There are many zodiac compatibility algorithms available online each having minor variations which are obvious because there cannot be a perfect algorithm for zodiac compatibility. We have used the algorithm from <http://autumncreek.deviantart.com/art/CoTZ-Zodiac-Compatibility-Chart-348529483> which is as shown in figure 2.2. Based on the value, a score is assigned out of 10 for zodiac compatibility.

****

**Figure 2.2: Zodiac Compatibility Chart**

***2.2 Name Compatibility:***

Name compatibility can be calculated based on the alphabets in a name. We referred the Chaldean numerology from the site <http://www.sevenreflections.com/more/chaldean-numerology/>. As shown in figure 2.3, each alphabet is assigned a particular number. All the corresponding numbers are added and assigned a score out of 10. We referred the site <http://the-oracle-answers.com/numerology-partner-number/> to assign relative scores out of 10.

****

**Figure 2.3: Name Compatibility according to Chaldean Numerology**

***2.3 Others (languages, education, location etc.)***

As already mentioned, once the access token is generated, you can determine which permissions are needed in this access token. The following calls can be used to access the user’s information:

graph = facebook.GraphAPI (oauth\_access\_token)

profile = graph.get\_object ("me")

This gives access to all the public information from which language, location, hometown and educational compatibility can be calculated. We also get the facebook likes of the user from which we can calculate compatibility based on his/her likes/interests like favourite athletes, sports, sports teams, music etc.

The compatibility is calculated based on the intersection of the two sets of the corresponding attributes of the user and his/her friend. The entries which are not in the intersection are displayed to the user as recommendations. This will suggest the user pages or activities he might be interested in.

**Note: We have assigned a score of 3 out of 10 in the cases where some user information is not accessible or not available. For example, if user’s or any of his friend’s birthdays is not public, zodiac compatibility between them cannot be calculated.**

**3. CODE ARCHITECTURE:**

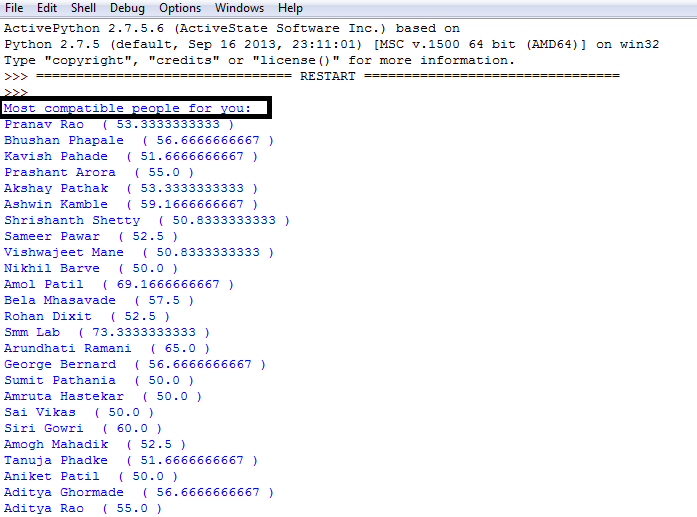
We have given the entire code a modular structure consisting of 6 python modules:

* **Main** – Main module from where the execution of the program begins and methods from other modules are being called.
* **Algorithms –** This is the module where all the algorithms are defined i.e. zodiac sign, name, educational etc.
* **Connector –** This is the module which communicates with the facebook REST API through the app authentication key. It has methods which extract the user information using the facebook Graph API.
* **Model –** Module handles app exceptions and defines the facebook user model for this app by initializing user attributes which are considered for compatibility analysis to none.
* **Helper –** Helper module where the helper functions are defined like zodiac sign calculator, converting a particular id in to the app user model (which is defined in Model module) and fetching the user friends.
* **Suggestions –** Module essentially encapsulates the app logic by calculating the compatibility score and returning the best matches based on a threshold value and providing suggestions based on compatible people’s likes/interests.

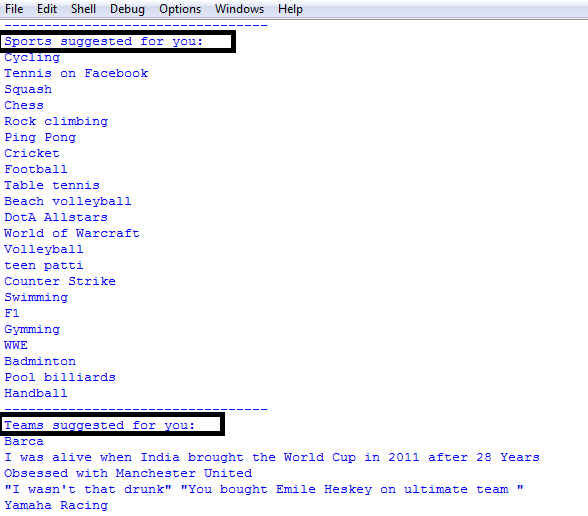
**4. OUTPUT:**

The output will be a list of compatible people whose compatibility score is above the threshold value. The threshold value is currently set to 50. It can be changed in the main module to filter out the results.

The second part of the output will be suggestions for the user. As already mentioned, compatibility score will be calculated on the basis of intersection of the two sets of corresponding attributes for the user and his/her friend list. Suggestions will be the elements which are not in the intersection set. A sample screenshot of the output is as follows:



**Figure 4.1: Output Screenshot 1**



**Figure 4.2: Output Screenshot 1**

****

**Figure 4.3: Output Screenshot 3**

**5. CONCLUSION AND FUTURE SCOPE:**

The app is purely for entertainment purpose only. It entertains the user by showing list of most compatible people from his/her friend list. Currently, the app is not gender and age group specific. So, it will show list of friends whose compatibility is above the threshold score irrespective of the gender and age group. This is a future modification which can be added to give a more sophisticated and dedicated compatibility analysis. We also haven’t implemented the compatibility based on photo tags and check-ins. But the above two features can be added easily. The procedure is the same as for the other attributes. We just have to select the appropriate user data permissions while getting the access token and add them to the app user model in model.py. The compatibility calculations will be the same as for other attributes. In the future, we would like to add these two attributes to give a better compatibility score and more suggestions based on interests in music, books, movies etc.

We have uploaded the project on github as well. You can download it from <https://github.com/amolpatil8187/facebook_mining> . We have kept it open source. Whatever changes we will do in the future will be committed to the github repository. You can always download the updated version from the link.