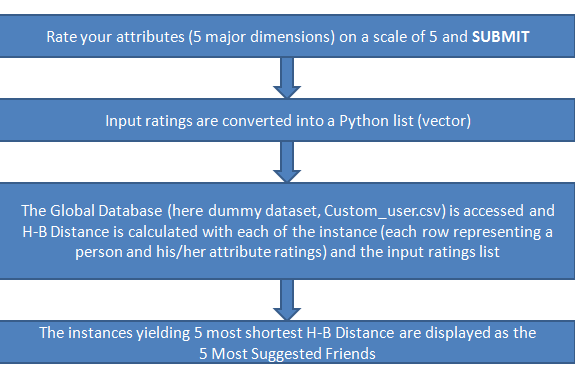
Problem Statement: Friend Affinity Finder

The Idea

Novelty: According to the aforementioned problem statement, list of people having great probability of becoming friends based on possible affinity score depending upon 5 major dimensions (attributes): Agreeableness, Conscientiousness, Extraversion, Emotional range and Openness. If (hypothetical scenario) people rate their attributes themselves in different social platforms and their data (ratings) are stored in a database, then for new users searching for friends online, can get a set of suggested friends having high affinity scores based on the similarity between the set of attribute ratings of the different people in the database and the set of ratings entered by the user using the web application. In social platforms, the database maintaining the attribute ratings of different people is subject to change and the web application should fetch the updated dataset every time it is used to find suggested friends.

Proposed Solution with Architectural Flow: The solution developed for finding suggested friends based on affinity score is done by deploying the concept of Hellinger-Bhattacharya Distance (H-B Distance):

where X and Y are points on n-dimensional space and X = (x1, x2, x3,...., xn) & Y = (y1, y2, y3, ....., yn) with HB(X,Y) as Hellinger-Bhattacharya Distance between X and Y

The user (the one using the Web Application) is required to rate himself/herself on a scale of 5 (>0 and <=5) on the 5 major dimensions and the web app takes the input and visualizes it as a point in a 5-dimensional co-ordinate space. Similarly, the database containing ratings of other people on the 5 dimensions are also visualized as points on the 5-dimensional space. So, the H-B Distance is calculated with the input point with each of the other points present in the database. 5 points (each representing a person with a set of ratings of the 5 dimensions) are selected that have the smallest H-B Distance in descending order to be the 5 most suggested friends. The Architectural Flow of the solution is given below:

The maximum H-B Distance between any 2 points in a 5 dimensional space, given the maximum value representing a dimension is 5 (rating is done on a scale of 5), is 3.54 as when X = (0,0,0,0,0) & Y = (5,5,5,5,5). Hence, the similarity/affinity score in % is calculated by the formula:

where H is the H-B Distance

Technologies Used: Hyper Text Markup Language (HTML), Cascading Style Sheets (CSS), JavaScript (JS), jQuery and Python Programming using Common Gateway Interface (CGI)

Impact of the proposed solution on Business: The proposed solution can be deployed in any platform and does not require custom libraries or external Python packages for its deployment. This can add to a huge benefit on the implementation point of view and hence making the overall web application compact and light-weight. The proposed solution only requires a Global People Database that contains self-ratings of the 5 major dimensions (in scale of 5) from which it can fetch the details of the people and can suggest & find the friends among which, high affinity may exist. All-in-all it takes almost no expenses for the deployment and can be socially beneficial to people round the globe.

Scope of the work: A Global People Database is required for implementing the solution stated above in real time. Here, a dummy data-set is created by us, named “Custom\_user.csv”. After the database is present physically, data can be fetched from that and hence, the remaining steps of the solution can be performed to arrive at the result (friend suggestion or friend finder based on affinity). The remaining modules are already implemented by us.