Project Writeup

— An Application That Finds Your Crush

Lai Wei, Shirui Ye, Chang Gao

Abstract:

College students have crushes all around places. A lot of these students are confused of how to approach their crushes. We decide to solve this problem using technology: we want to develop an application that helps one-sided lovers to build confidence and acquire skills to approach their beloved ones.

Our application serves as a combination of an evaluation system and a GPS of relationships that provides the best possible paths for users to approach their crushes reflecting quantified parameters. The basic idea is similar to that of LinkedIn as it demonstrates the layers of connections that lead to the target. However, there are features that differentiate our application from LinkedIn and make it more powerful. Instead of being a static result, the path of connections suggested to our user is a dynamic optimization process. Our application is continuously fed by new data while people are updating their status on the social media and performing daily activities.

Introduction:

We conducted a survey on our friends and students at BU. The result is astonishing. We got 105 feedbacks from our online survey. 97 out of 105 students (that is, 92%) had crushes but did not act. Among these people who did not act, 95% of them regret that they did nothing but Facebook stalking. Our conjecture is that students rely so much on online social media that they are simply incapable of establishing relationship with a stranger. We decide to solve this problem using online social media itself.

The online social network applications have been growing rapidly over the past few years, and these social network applications can easily obtain the basic information of users from their APIs. In the social network, calculating the similarity and utilizing graph algorithm between users is an important issue.

After reading through over twenty papers, we selected three key parameters which are frequently mentioned for establishing relationships between human beings. We propose three methods to calculate the these parameters. The first one uses the semantics of location to capture the user's intention and interest. The second method calculate trust values using user's interactions on Facebook. The third method is to derive the compatibility/suitability based on the users' and their targets' interests, and the interactions between the targets' and their friends.

Related work:

There have been numerous efforts to calculate the similarities for different objectives. Perhaps the main question is that why a particular user similarity is important to solve real world problems. We read multiple psychological paper related to human relations and we found some crucial aspects that can affect relationship networks:

Sayed Atalla, a psychological professor from Maastricht University, states that trust is an inevitable factor among all kinds of relationship. He points out that we must be careful about who we contact and people need to choose friends that are trustworthy.

In their paper “ Friendship and Mobility: User Movement In Location-Based Social Networks”, Eunjoon Cho, Seth A.Myers and Jure Leskovec from stanford university make conclusion that location factors between people can explain about 10% to 30% of all social relationships.

In another academic paper, prof.Peter Godfrey-Smith and Prof.Manolo Martinez demonstrate that common interest is essential in a relationship. For example, they state in the paper that “if two people don’t have same interest, their conversations are cheap”. Thus they conclude that it is the best that people choose a relationship that shares a common interest.

As a result, we decide to make use of these factors and apply them to real world problems. Among multiple academic paper related to similarity calculation, we select three of them to help us achieve our goals. For trust algorithm, we implement an algorithm described by professor Mirijam Situm in his book “ Analysis of Algorithms for Determining Trust Among Friends on Social Networks”. For the location similarity, we propose an algorithm in the work of Lee, Min-Joong, and Chin-Wan Chung “A User Similarity Calculation Based on the Location for Social Network Services”. Finally, we calculate compatibility based on user interests using the approach introduced in paper “Finding Someone You Will Like and Who Won’t Reject You” written by Pizzato, Luiz Augusto, Tomek Rej, Kalina Yacef, Irena Koprinska, and Judy Kay.

Description of Data Collected:

We collect our data from Facebook Graph API, Foursquare API and Twitter API.

For the location similarity, the data we need to collect from the API is the location of the users. To be specific, we extract the location information from the API’s venuehistory field which returns a list of all venues visited by the specified user, along with how many visits and when they were last there. We also need the data from the Location Hierarchy field from the API which changes the exact locations to categories of locations.

For the trust value algorithm, the data we need to collect is the interactions between users such as tables of *PostTag*, *PostComment* and *PostLike*, which store tags, comments, and likes for every post. The details of the interactions between users collected will be discussed in the experiment part.

Finally, the data we need to collect for the compatibility algorithm are similar to the concept of recommendation system. We use Facebook API to collect the number interactions between persons, and AlchemyAPI to find the topics that a user concerns and the sentiments towards these topics.

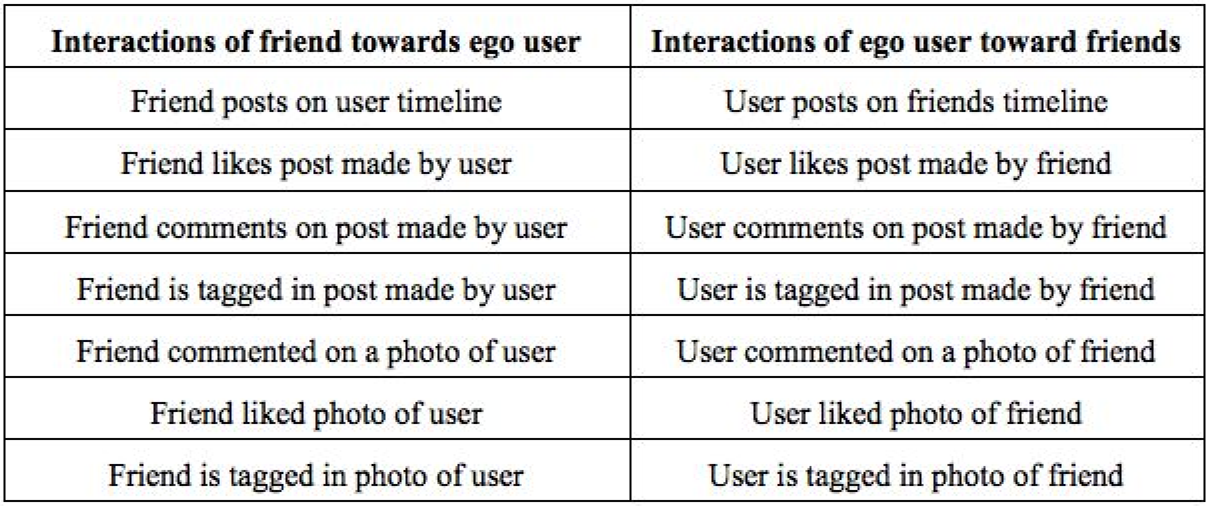
Methodology:

Trust Value Algorithm

We use trust value as another factor to determine who are proper for the users to contact with. Just like the psychology article written by Dr. Sayed Atalla says, “There is a lots of relationship in our life and in this world like - a relationship between husband and wife, friends, parents and children, classmates, workmates and even neighbors. A relationship is easy to make but it is so hard to make that relationship stronger and lasting. A long lasting relationship is very easy to establish if we put the spirit of trust.” A long lasting relationship is our goal, because steady relationships are more valuable and last longer. Our application intends to lead users to their crushes and make a steady relationship. In this case, we would like to calculate how much a person is worthy to trust. In short, we want to calculate people’s trust values, so we choose an algorithm which is called Tidal Trust Algorithm to calculate trust values. Tidal trust is based on people’s social media states. For example, we use Facebook as one of our APIs, we calculate trust values according to people’s corresponding Facebook picture likes, Facebook comments, etc. We will discuss how we will use this algorithm in our application precisely in experiment part.

Our application intends to building the connections between the users and their targets, so the potential trust between these people can be crucial. The more people trust each other, the more likely they can build a successful relationship. The algorithms to measure trust between users are described in *Analysis of Algorithms for Determining Trust Among Friends on Social Networks*, written by Šitum et al.

Interaction based algorithms calculate trust between user and his/her friends on Facebook based on their interactions in social network. These interactions can be divided into two groups:



Interactions of friend towards ego user	Interactions of ego user toward friends
Friend posts on user timeline	User posts on friends timeline
Friend likes post made by user	User likes post made by friend
Friend comments on post made by user	User comments on post made by friend
Friend is tagged in post made by user	User is tagged in post made by friend
Friend commented on a photo of user	User commented on a photo of friend
Friend liked photo of user	User liked photo of friend
Friend is tagged in photo of user	User is tagged in photo of friend

Trust value between ego user and his friend A is computed by this expression:

$$trust(ego\ user, friend_A) = \frac{\sum_{i \in I} i(A) * w_i}{\sum_{i \in I} w_i}$$

where I is the set of interactions of friend A towards ego user for engagement algorithm, and set of interactions of ego user towards friend A for popularity algorithm. i(A) is the number of interactions i towards ego user from friend A for engagement algorithm and number of interaction towards friend A form ego user for popularity algorithm. *wi* is the weight of the interaction i. Weights are different for every action and are calibrated in the later chapters of this work.

After computing trust values between the users their direct friends, we will use Tidal Trust Algorithm to compute trust between two users that are not directly connected, that are not friends. Tidal Trust algorithm consists of two parts: finding a path from the source to the sink while rating nodes while on the way to the sink, and, after the sink is found, aggregating trust backwards towards the source. Every node in the graph, except the nodes in the final level of the graph that have direct trust values towards the sink, calculate their trust values and the best routes towards the sink by these expression from the paper:

$$t_{i,k} = \frac{\sum_{j \in adj(i) | t_{i,j} \geq max} t_{ij} * t_{jk}}{\sum_{j \in adj(i) | t_{i,j} \geq max} t_{ij}}$$

Where max is the trust threshold, ti,k is the trust between the nodes i and k , and j are all neighbors of node i. The more details of this algorithm are in the paper and we are not going into too much details here.

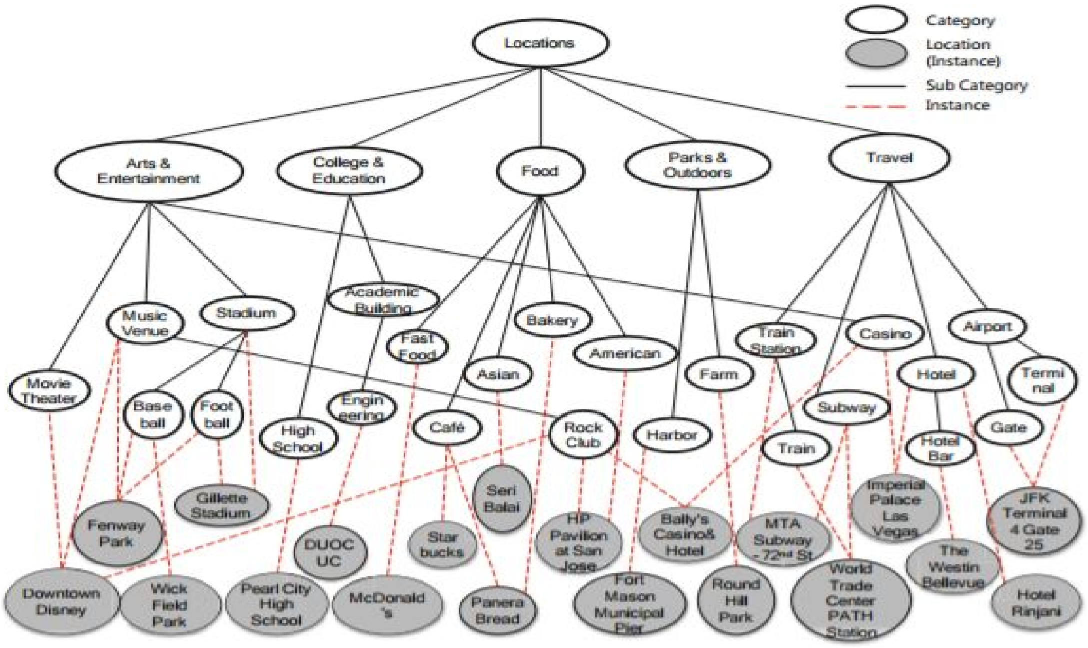
Location Based Similarity Algorithm

Geographical factor is also an important factor in determining whether two people are suitable for each other. In their paper “ Friendship and Mobility: User Movement In Location-Based Social Networks”, Eunjoon Cho, Seth A.Myers and Jure Leskovec from stanford university make an hypothesis that “ mobility may be shaped by our social relationships as we may be more likely to visit places that our friends and people similar to us visited in the past.” They further make the conclusion from their experiments that “ human movement can explain about 10% to 30% of all social relationships.” As a result, we propose an algorithm to calculate similarity between users based on their locations and utilizes it in our application.

To be specific, we use the algorithm described in paper “A User Similarity Calculation Based on the Location for social Network Services” written by Min-Joong

Lee and Chin-Wan Chung to calculate the location similarity between two users. The approach in paper uses the semantics of the location instead of the physical location.

In order to make the paper adapt our case, we extract the location category hierarchy from the foursquare API which gives access to world-class places database. Here is an example of location category hierarchy graph returned from foursquare Category Hierarchy API. It consists of two kinds of nodes, location nodes and category nodes. A location node represents the corresponding unique location such as Fenway Park or Mugar Library. A category node represents a location category such as park or library. Then we calculate the significant score of node n of user u which is equal to the number of visits at location node n of user u divided by the total number of visits of user u. Here is an example of an location category hierarchy graph from the paper:



The overall procedure of the algorithm can be described as follows:

1. Compute the significant score of each visited location of user u and user v.
2. Find top-k locations of user u and user v, and construct a top-k significant score table for each user.
3. Construct a location category hierarchy graph by using only visited location nodes of two user and visited location nodes' ancestor nodes.
4. Find the match nodes and its calculation order by using algorithm MatchN-odeOrder().
5. Calculate the user similarity between user u and v by using algorithm Similarity().

The formula of significant score, the algorithm for MatchN-odeOrder() and Similarity() are written in pseudocode in section 4.2 and section 4.3 in the work of Lee et al. As long as we have the significant scores and category hierarchy graph between two users, to successfully run this algorithm is guaranteed. As a result, we are not going into too much details of the algorithm.

The main question is that what are the advantages of this algorithm? In the real world, one specific location is related to many places such as coffee shop and a theater. This problem is worsened when calculating similarity if the users is in a building in a downtown because the exact place cannot be determined. With Foursquare's Category Hierarchy from its API and the Algorithm described in this paper, we can determine the exact place and capture the user's intention.For instance, if a user visits mugar library frequently, it is reasonable to infer that the user is a student or a faculty member of the university. However, a drawback of this approach is that users must have a Foursquare account. It's better to have users to link their Facebook or Twitter accounts to their Foursquare accounts so that the data collected will be more reliable.

The result of the this algorithm is meaningful for our application because the paper suggests that it collects more than 251000 visited locations over 591 users from foursquare database. The proposed method is 84% higher in precision, 61% in recall and 72% in f-measure than Jaccard index. The running time of this algorithm is O(n^2).

Compatibility

There is no doubt that the similarity of interests and tastes plays a significant role in a relationship between two human beings. Our application takes the likes, tagges and comments from Facebook and Twitter as signals of recognition to analyze the compatibility of two persons. As our application will show a path of the users to successfully approach their crushes, we will implement an algorithm combining Dijkstra's shortest path and an algorithm named RECON. The RECON algorithm is elaborated by Piazzato et al. in *Finding Someone You Will Like and Who Won't Reject You* and tested on the largest online dating website in Australia.

RECON is a content-based reciprocal recommender system for online dating. It uses positive user interactions to build the model used to generate compatibility parameters. RECON is a reciprocal recommender, meaning that it considers the preference models of both sides before a match is suggested to the user.

For instance, Alice has liked/tagged 10 students with the following characteristics: 7 CS major students, 3 music major students; 5 undergraduate student, 5 graduate students. The positive compatibility between Alice and a user Bob who is CS major and an undergraduate student is:

$$C^{+}(Alice, Bob) = \frac{7+5}{10 \times 2} = 0.6$$

The same approach can be used create a model based on the negative interactions between the users (i.e. indications that someone does not like someone else). Given a positive and negative compatibility, we can calculate the combined compatibility of a user A with a user B using A's positive and negative models of preference by subtracting the negative compatibility score from the positive compatibility score, with a normalisation step to obtain a compatibility score between 0 and 1. The formula is as follows:

$$C^{\pm}(A,B) = \frac{1 + C^{+}(A,B) - C^{-}(A,B)}{2}$$

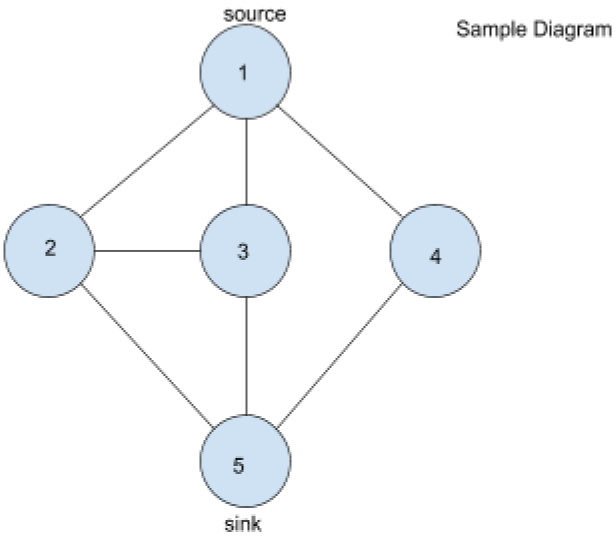
Similar to RECON, reciprocal recommendation can be created as the harmonic mean of the combined compatibility scores such that:

$$C^{\pm}_{rec}(A,B) = \frac{2}{\frac{1}{C^{\pm}(A,B)} + \frac{1}{C^{\pm}(B,A)}}$$

between all pairs of users A and B. We use harmonic mean because it is desirable to favour low compatibility scores over high scores when two users have distinctly different levels of compatibility. For instance, if Bob likes Alice a lot, and Alice does not like Bob at all, there is a very little chance that this reciprocal relationship will be successful; therefore, we want to have a reciprocal compatibility score more similar to Alice's score than to Bob's score.

To support this algorithm, we use User Like object in Facebook Graph API, which contains a parameter target_id that returns the pages that this user has liked. We also use AlchemyAPI to find out the topics the users concern and their attitudes towards these topics. By plugging the data into the formula described previously, the compatibilities will be calculated as weights on the edges of the network. Using shortest path algorithm, the application will be able to return a path reflecting the compatibility parameters.

Experiment:



Trust Value Algorithm:

The data below is randomly chosen from facebook API. We will use the 5 people(nodes) below to calculate the trust value between each other. We will present how tidal trust algorithm works in detail by calculating the trust value between nodes.

	Node 1	Node 2	Node 3	Node 4	Node 5
friend posts on user timeline	5	19	11	29	14
friend likes post made by user	20	45	23	10	61
friend comments on post made by user	3	11	19	15	23
friend is tagged in post made by user	4	19	14	21	32
friend is commented on a photo of user	8	89	35	17	67
friend liked photo of user	21	15	9	20	38
friend is tagged in photo of user	16	33	11	41	29

We suppose the factors above data have the same weights:

trust(Node 1, Node 2)=231
trust(Node 3, Node 5)=264
trust(Node 2, Node 3)=122
trust(Node 4, Node 5)=264

trust(Node 1, Node 3)=122
trust(Node 1, Node 4)=153
trust(Node 2, Node 5)=264

Path returned: 1 → 2 → 5

Location similarity:

We extract data from Foursquare users’ venuehistory field. It returns a list of all venues visited by the specified user, along with how many visits and when they were last there.

We record the number of visits of each place. Then we also sum the total visits of locations. Here is an sample dataset of the number of visits of locations of two users:

```
User1:                                   User2:

{ [ Panera bread : 27,             { [  UNO: 17,
    Fenway Park: 17,                    JFK Terminal Gates: 2,
    Bally’s Casino & hotel: 14,         Hotel Rinjiani : 5,
    McDonald’s : 5,                     Central Park : 7,
    Downtown Disney 3                   Starbucks : 15
] }                                ] }
```

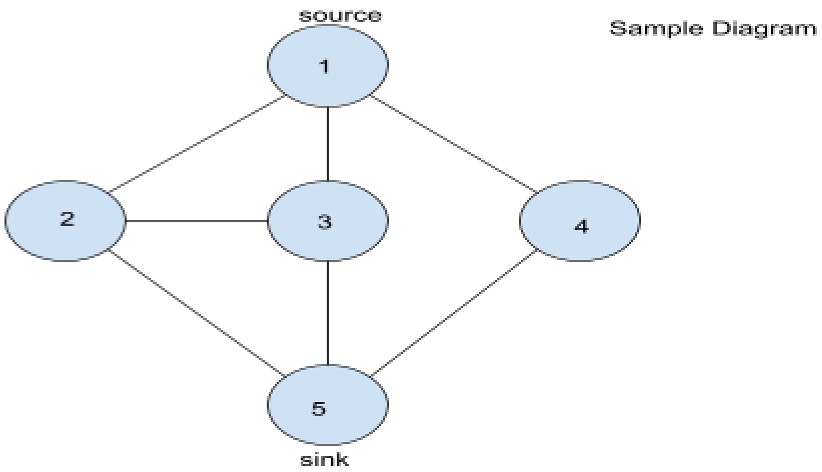
We then calculate the significant score of each locations which is equal to the number of visits of each location divided by the total number of visits. The significant scores are listed below:

```
User1:                                   User2:

{ [Panera bread :0.409,              { [  UNO: 0.369,
   Fenway Park: 0.258,                    JFK Terminal Gates: 0.043,
   Bally’s Casino & hotel: 0.212,         Hotel Rinjiani : 0.106,
   McDonald’s : 0.076,                    Central Park : 0.152,
   Downtown Disney :0.045,                Starbucks : 0.326
] }                                  ] }
```

The paper experimentally shows that the visits to top k locations take up great part of total visits in Section 5.4. As a result, to avoid a time-consuming process, we consider only top k locations of a user to calculate the similarity. For the purpose of demonstrating concepts we assume the above 5 locations are the top k locations that user1 and user2 visited. With the significant score calculated and the location hierarchy(change exact locations to location categories) extracted from from the Foursquare database we are able to carry out the algorithms described in the paper. The sample results are demonstrated in the result part. The similarity scores are the weight of the edges and we will find the shortest path based on these weights.

Compatibility:



Because we need to return a directed path, we do not consider any compatibility between two nodes that are not neighbors, and we do not compatibilities that are in the reverse direction of the edge (For example, we consider only C+(1, 3) but not C+(3, 1), because the there is an edge (1, 3) but there is no edge (3, 1)). The Recon+ and Recon- are randomly generated to demonstrate the algorithm. The shortest path will be calculated using weights of C+-rec, which are negative correlated to compatibility.

Recon+	node1	node2	node3	node4	node5
node1	0	0.6	0.5	0.3	0
node2	0	0	0.2	0	0.8
node3	0	0	0	0	0.6
node4	0	0	0	0	0.7
node5	0	0	0	0	0

Recon-	node1	node2	node3	node4	node5
node1	0	0.3	0.7	0.2	0
node2	0	0	0.1	0	0.7
node3	0	0	0	0	0.3
node4	0	0	0	0	0.5

node5	0	0	0	0	0

Compatibility	node1	node2	node3	node4	node5
node1	0	0.65	0.4	0.55	0
node2	0	0	0.55	0	0.55
node3	0	0	0	0	0.65
node4	0	0	0	0	0.5
node5	0	0	0	0	0

C+-rec	node1	node2	node3	node4	node5
node1	0	3.077	5	3.636	0
node2	0	0	3.636	0	3.636
node3	0	0	0	0	3.077
node4	0	0	0	0	5
node5	0	0	0	0	0

Path returned: 1 → 2 → 5

Results:

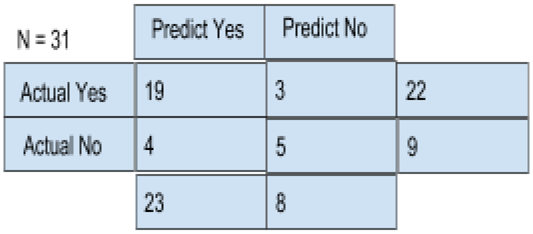
	Edge:1 to 2	Edge:1 to 3	Edge:1 to 4	Edge:2 to 3	Edge:2 to 5	Edge:3 to 5	Edge:4 to 5	Path
1.Trust Value	231	122	153	122	264	264	264	1 → 2 → 5
2.Location Similarity	0.327	0.431	0.225	0.341	0.103	0.534	0.030	1 → 3 → 5
3.Compatibility	0.65	0.4	0.55	0.55	0.55	0.65	0.5	1 → 2 → 5

Evaluation:

We conducted research on 31 BU students and we created three confusion matrix to evaluate our approach. To be specific, “Actual Yes” means that the user actually agrees with the suggested paths and thinks they are efficient. The predictions we made are based on the evaluation of each algorithms in paper.

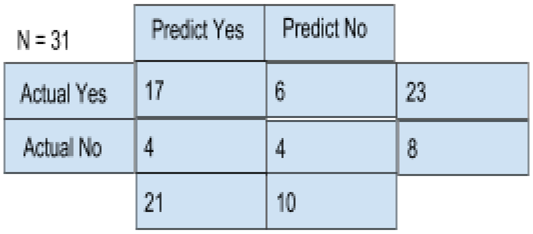
However, establishing relationships between users are a long time process and we do not have much time keep tracking them. We will try to gather more information afterwards to make our results more concrete. Below are the confusion matrix based on the data we collected:

Tidal Trust Algorithm:



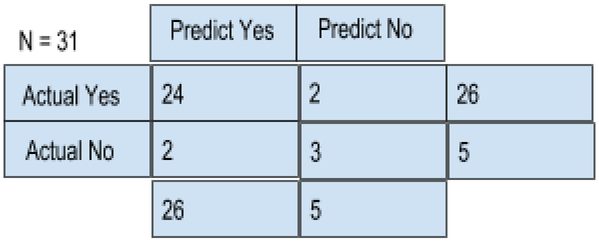
N = 31	Predict Yes	Predict No	
Actual Yes	19	3	22
Actual No	4	5	9
	23	8	

Location Similarity Algorithm:



N = 31	Predict Yes	Predict No	
Actual Yes	17	6	23
Actual No	4	4	8
	21	10	

Compatibility Algorithm:



N = 31	Predict Yes	Predict No	
Actual Yes	24	2	26
Actual No	2	3	5
	26	5	

The results from the confusion matrix are optimistic: the percentages of users agree and likes the paths proved by Tidal Trust, location similarity, and compatibility are 72%, 68%, 77% respectively.

Future Work:

For the next stage, our plan is to develop an artificial neural network that can be trained using the data collected from social media, so that the application will learn the grammars, vocabularies and tones of the targets. In addition, as our goal is to motivate and help users to interact with people in real-life, the application should also estimate the targets' responses towards the users' speech based on the models developed and social courtesies, and it should also provide advice to users to adjust their speeches and tones. In this way, the application servers as a personal coach that help users practice social skills and build confidence.

In addition, as facial expressions and body languages convey 70% of information in most human interaction (Joe Navarro, 2008), for the third stage, we will integrate facial expression analysis tools such as EmoVu to analyze pictures on social media. Our application can also be incorporated into domestic robots such as JIBO, that help autism patients.

Bibliography:

Cho, Eunjoon, Seth A. Myers, and Jure Leskovec. "Friendship and Mobility." *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '11* (2011): n. pag. Web.

Lee, Min-Joong, and Chin-Wan Chung. "A User Similarity Calculation Based on the Location for Social Network Services." *Database Systems for Advanced Applications Lecture Notes in Computer Science* (2011): 38-52. Web.

Pizzato, Luiz Augusto, Tomek Rej, Kalina Yacef, Irena Koprinska, and Judy Kay. "Finding Someone You Will Like and Who Won't Reject You." *User Modeling, Adaption and Personalization Lecture Notes in Computer Science* (2011): 269-80. Web.

Šitum, Mirjam. *ANALYSIS OF ALGORITHMS FOR DETERMINING TRUST AMONG FRIENDS ON SOCIAL NETWORKS*. Vienna: n.p., 2014. Print.

Navarro, Joe, and Marvin Karlins. *What every BODY is saying: an ex-FBI agent's guide to speed-reading people*. New York, NY: Collins Living, 2008. Print.

Sayed Atalla, "Love Trust and Respect". https://www.linkedin.com/pulse/importance-trust-love-any-successful-relationship-atalla. Publisher, Sayed Atalla, May 30 2015 published

Peter Godfrey-Smith and Manolo Martinez, "Communication and Common interest". https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3820505/
Editor Carl T. Bergstrom, Nov 7 2013 published