ISSN:0975-9646

# A Semantic-Based Friend Recommendation System Based On Lifestyle Matching

Prof. Vijayalaxmi Kanade , Pratibha Uphade, Priti Thite, Omkar Bandal, Neeraj Datar

*Department of Computer Engineering*
*Pune Vidhyarthi Griha's ,COET*
*Pune, India*

**Abstract: Most of the social networking systems or friend recommendation systems used nowadays recommend friends based on their social connectivity or on a mutual basis. But this is a syntactic approach as there is no metric used and so it may not be that appropriate approach for friend recommendation. In this paper, a semantic-based friend recommendation system is proposed which recommends friends to users based on their lifestyles. Similarity of lifestyles between users is measured by a similarity metric and the system recommends those friends to the users that share high similarity among their lifestyles. Impact of user is calculated and a friend-matching graph is generated accordingly. Finally, the system returns a list of users with highest recommendation scores to the user. The system also includes a feedback mechanism which allows users to rate the friend recommendation.**

**Keywords: Friend recommendation, lifestyle, social networking, similarity metric, friend matching graph, impact ranking.**

## INTRODUCTION

Nowadays, the use of social digital media has increased to a huge extent which has led to social networks gaining popularity along with the emergence of social computing. Social networks are nothing but entities that have interdependence between them in the form of interests in common, friendship, kinship, financial exchange. Generating social graphs and forecasting using social network mining, searching groups of friends which are

popular in social network are well known. Earlier people used to make friends within their geographical area itself which we call as G-Friends. But later as technology increased drastically, various friend recommendation systems came into existence which revolutionalized the way of making friends. According to various studies, people can be grouped together based on lifestyles/habits,likings, standards or behaviour, people they already know. Lifestyles of people are connected to their daily chores and activities. Thus, we can collect information about these and recommend friends to users based on their lifestyles. It is frequently for humans to change their view of friendship. This view varies from user to user in which a social network can undergo frequent and instant change over time even without the proem of new nodes. Recommendation systems help users to identify their curiosity and sets of choices by predicting the usefulness degree of an item or group of items.

## RELATED WORK

Recommendation systems nowadays have gained popularity due to changing lifestyles of people. Amazon recommends items to users based on their previous preferences. Netflix [3] and Rotten Tomatoes [4] recommend movies to a user based on the user's previous ratings and watching habits. Matchmaker[2], which was invented by Bian and Holtzman was a recommendation based on personality matching. [5]Kwon and Kim proposed a friend recommendation method based on social and physical context , although they did not tell anything about the context and the way to obtain information. Yu et al.[8] recommended geographically related friends in social network by combining GPS information as well as social network structure. Hsu et al. [7] studied the problem of link recommendation in weblogs and similar social networks, and proposed an approach based on collective recommendation using the link structure of a social network and content based recommendation using mutual declared interests. Kwon and Gou et al. [6] introduced a visual system, SFViz, to support users to explore and search friends interactively under the context of interest, and studied the system to explore the recommendation of friends based on people's tagging behaviors in any community. These existing friend recommendation systems, but, are significantly different from present work, as we exploit recent sociology findings to recommend friends based on their similar life styles instead of social relations.

Existing system for this paper is Friendbook which is a semantic-based friend recommendation system based on lifestyle matching. It is basically a single user system in which only one-to-one communication is possible between users. In this system, in the friend matching graph, two users A and B can be made friends if and only if similarity between them is greater than or equal to a predefined similarity threshold. This threshold is the same for all users which means it is static. To support performance optimization at runtime, a feedback control mechanism has been integrated into Friendbook.

## PROPOSED SYSTEM

The proposed system is a semantic-based friend recommendation system based on lifestyle matching. It is a system consisting of a client, server and a database. Client could be on Android based smartphones and server on desktop machine and both these can be connected through internet .Client app has lifestyle entering facility along with user querying and feedback facility.Server performs various tasks including taking all data together, taking lifestyles from client,putting them into a specific format,generating graphs and ranking users, and finally taking feedback from users.

It is basically a multi user system in which group communication is possible between users. Group-chatting facility has been proposed in our system which makes it more user-friendly. The similarity threshold which was static in the existing system has been converted to dynamic in the proposed system which means it will vary for every two users based on the similarity extent in their lifestyles. The feedback mechanism is also being integrated in our proposed system which allows users to rate the friend recommendation on star-rating basis. This will help in making improvements in friend recommendation.

### A. *System S is defined as {I, O, Fu, Su, F}*

where ,

**Input** ,I={lifestyles of users}, where lifestyles can be jogging, swimming, singing, playing cricket, dancing, cooking, drawing, painting, sketching , reading etc.

**Outpu**t ,O={ranked list of recommended friends }, where ranked list denotes that users in the generated list have non-increasing order of similarity from top to bottom i.e. user with highest matched similarity is at the top of the list and the user with the least matched similaritry is at the bottom of the list.

**Functions**,$F_u$={RegFragment(),LoginFragment(),ButtonLifestylesOnclick(),calculate(),storedImpactRanking(),sendFriendRequest(),sendMessage(),msg()}

RegFragment() -user needs to enter the appropriate details for registration.

- LoginFragment()-user must enter the valid username and password which he/she has registered for login. only registered user's can login successfully.
- ButtonLifestylesOnclick()-user specifies his own lifestyles.
- calculate()-when new user enter his lifestyles calculate function compared the lifestyles of registered users present in the database.
- storedImpactRanking()-this function stored the user's efficiency to promote friendship in the network.
- sendFriendRequst()-depending upon the impact ranking sever will send friend request from one user to other user.
- sendMessage()-after the user become friends they can communicate one to one.
- msg()-users who are friends they can communicate in group or chat in a group.

**Success** ,$S_u$={ranked list of recommended friends} , where ranked list denotes that users in the generated list have non-increasing order of similarity from top to bottom i.e. user with highest matched similarity is at the top of the list and the user with the least matched similarity is at the bottom of the list.

**Failure** ,F={if lifestyles are incorrectly spelled then inappropriate results will be generated}, which means for example there are two users A and B having their lifestyle vectors as, A={swimming, dancing}and B={swiming, dancing},

In this case users A and B have both lifestyles matching practically. But as the word "swimming" is spelled incorrectly in case of B, so system will not match this lifestyle of both users as it will detect that "swimming and "swiming" are different lifestyles. So words must be correctly spelled.

## SYSTEM ARCHITECTURE

The system architecture consists of client and server. Client app could be implemented on Android based smartphone and server side could be implemented on desktop machine. Client and server will be connected via Internet.

On the client side, each smartphone takes user's lifestyles(data) and stores them into the server. These lifestyles are used in lifestyle analysis, lifestyle indexing and to generate friend matching graph.

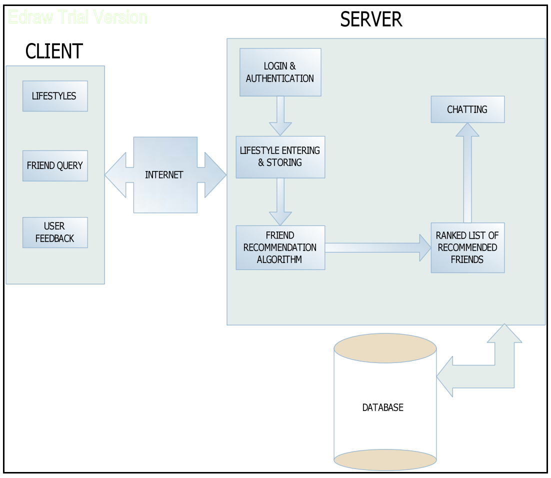

Figure1. friend recommendation system

### *I. Client :Client consist of three parts*

- Life Styles: Life styles are nothing but activities, hobbies, daily chores of people. They can be anything like jogging, swimming, singing, playing cricket, dancing, cooking, drawing, painting, sketching, reading etc.
- Friend Query: The user who wants to make friends makes the friend query. Friend query is nothing but request sent by user to get friend recommendations.
- User Feedback: User will get friend recommendations on querying the system. User feedback is a facility which will allow him/her to give his views about the friends recommended in the form of star-rating basis.

### *II. Server : It consists of five parts*

- Login and Authentication- This section has two parts. Firstly, the user has to enter his username which is nothing but his email id . Then, he will have to enter his password. Then he/she will be a registered user in the system.
- Lifestyle entering and Storing- After becoming a registered user , the system will ask the user to enter

his/her lifestyles. Lifestyles can be anything like jogging, swimming, singing, playing cricket, dancing, cooking, drawing, painting, sketching, reading etc. These lifestyles will be then stored onto server.

- Friend recommendation algorithm- This is the algorithm to recommend friends to the query user. This algorithm will take lifestyles of query user as input.
- Ranked list of recommended friends- This is the output of the friend recommendation algorithm, where ranked list denotes that users in the generated list have non-increasing order of similarity from top to bottom i.e. user with highest matched similarity is at the top of the list and the user with the least matched similarity is at the bottom of the list.
- Chatting- Once the ranked list of recommended friends is generated, then user can chat with his/her friends. Also, user can create groups among his/her recommended friends for various purposes. This makes the system more user-friendly.

### *A. FRIEND RECOMMENDATION ALGORITHM:*

This is the algorithm to recommend friends to the query user. This algorithm will take lifestyles of query user as input. Then. it will return ranked list of friends as output.

Here ,

$U_q$ is the lifestyle vector of query user,

$F_i$ is the list of recommended friends which is null at the beginning ,

$U_i$ is the lifestyle vector of $i^{th}$ user,

S(i,j) is the similarity metric between users i and j,

$U_q$.length is the number of lifestyles in lifestyle vector of $q^{th}$ user, $U_i$.rank is the rank of $i^{th}$ user.

```
INPUT :Query user $U_q$ and his lifestyles
OUTPUT: Friend list $F_i$
1: $F_i \leftarrow \emptyset$, rank $\leftarrow 0$
2:Take query user's lifestyles from database db.
3:Take all users' lifestyles from database db.
4:Initialize query user's lifestyle vector $U_q$ .
5: for i in range 1 to number of users
      Initialize $user_i$ 's lifestyle vector $U_i$
      for j in range 1 to $U_q$.length
        for k in range 1 to $U_i$.length
          if $U_q[j] == U_i[k]$
            if $U_i \in F_i$
              $S(U_q,U_i)$ +=0.1
            else
              $S(U_q,U_i) = 0.1$
              $F_i$.insert($U_i$)
          end if
        end if
      end for
    end for
  end for
6:Sort all users $U_i$ in $F_i$ in non-increasing order according to $S(U_q,U_i)$.
7: for all users $U_i$ in $F_i$
    $U_i$.rank=$U_{(i-1)}$.rank+1
```

## IV. CONCLUSION

A semantic-based friend recommendation system for social networks, which recommends friends to users based on their life styles instead of social graphs has been proposed. Group chatting facility is proposed which will allow communication among various users and their friends, thus making our system more user-friendly. The introduction of dynamic similarity threshold in the system has increased the efficiency of friend recommendation.

## REFERENCES

[1] Zhibo Wang, Jilong Liao, Qing Cao, Hairong Qi and Zhi Wang, " Friendbook: ASemantic-Based FriendRecommendationSystemforSocialNetworks ,"*IEEE TRANSACTIONS ON MOBILE COMPUTING, VOL. 14, NO. 3, MARCH 2015*

[2] L. Bian and H. Holtzman, "Online friend recommendationthrough personality matching and collaborative filtering," *in Proc.5th Int.Conf. Mobile Ubiquitous Comput., Syst., Services Technol.,2011, pp. 230–235.*

[3] Netflix. (2014). [Online]. Available: https://signup.netflix.com/

[4] Rotten tomatoes. (2014). [Online]. Available: http://www.rottentomatoes.com/

[5] J.Kwon and S. Kim, "Friend recommendation method using physical and social context," *Int. J.*

[6] L. Gou, F. You, J. Guo, L. Wu, and X. L. Zhang, "Sfviz: Interestbased friends exploration and recommendation in social networks," *in Proc. Visual Inform. Commun.-Int. Symp., 2011, p. 15.*

[7] W. H. Hsu, A. King, M. Paradesi, T. Pydimarri,and T. Weninger,"Collaborative and structural recommendation of friends using weblog-based social network analysis," *in Proc. AAAI SpringSymp. Ser., 2006, pp. 55–60.*

[8] X. Yu, A. Pan, L.-A. Tang, Z. Li, and J. Han, "Geo-friends recommendationin GPS-based cyber-physical social network," *in Proc.Int. Conf. Adv.Social Netw. Anal. Mining, 2011, pp. 361–368 Comput. Sci. Netw. Security,vol. 10, no. 11, pp*

**PRATIBHA UPHADE:** currently pursuing BE Computer from PUNE VIDHYARTHI GRIHA'S ,Maharashtra. Currently working project is A semantic-based friend recommendation system based on lifestyle matching.

**NEERAJ DATAR:** currently pursuing BE Computer from PUNE VIDHYARTHI GRIHA'S ,Maharashtra. Currently working project is A semantic-based friend recommendation system based on lifestyle matching.

**PRITI THITE:** currently pursuing BE Computer from PUNE VIDHYARTHI GRIHA'S ,Maharashtra. Currently working project is A semantic-based friend recommendation system based on lifestyle matching.

**OMKAR BANDAL:** currently pursuing BE Computer from PUNE VIDHYARTHI GRIHA'S ,Maharashtra. Currently working project is A semantic-based friend recommendation system based on lifestyle matching.