Online Friend Recommendation System by Lifestyle Matching for Social Network

S. Saravanakumar[1], L. Somesh[2], K. Vijayaguru[3], I. Vasudevan[4]
[1, 2, 3] 8th sem / Department of CSE/V.S.B Engineering college/ Karur/TN/India.
[4] Assistant Professor/Department of CSE/V.S.B Engineering college/ Karur/TN/India

***Abstract*--- In existing social network system the friend recommendation for the user is given by considering their social graphs which means people they may know and taste of the user. This recommendation method does not suggest potential friends to user. The method that we are going to design is friend recommendation system to user by considering the life style and activities of user. This method will get the input by monitoring the daily activities, information submitted while creating the account and feedback information given by the user. It uses the Match-maker algorithm to recommend friends to the user by maintaining life document of the user. We use the similarity metric to calculate the similarity between user and friend matching- graph will be used to match the friend for the user based on their lifestyle. User impact ranking is given to individual user by considering the user capability to make friendship with other friends.**

keywords: Lifestyle, friend recommendation, user impact ranking.

I. INTRODUCTION

In social networking system user made friend with their neighbors, relation or colleagues. This recommendation is made through by considering their social graph of the user. The major drawback of this system is that user can be made friend with another user through the traditional fashion as G-friends, which stands for geographical location based friends because they are influenced by the geographical distances between each other. One major area where the social networks want to concentrate is to recommend good friends to user. The rules to be followed while grouping people together include: 1) habits or life style; 2) attitudes; 3) tastes; 4) moral standards; 5) economic level; and 6) people they already know. The system wants to recommend friend to the user by considering the life style and daily activities of the user. Lifestyle is expressed in both work and leisure behavior patterns and (on an individual basis) in activities, attitudes, interests, opinions, values, and allocation of income. The daily activities and life style of the user can be extracted by the text mining concept. Text mining is the process or practice of examining large collections of written resources in order to generate new information, typically using specialized computer software.

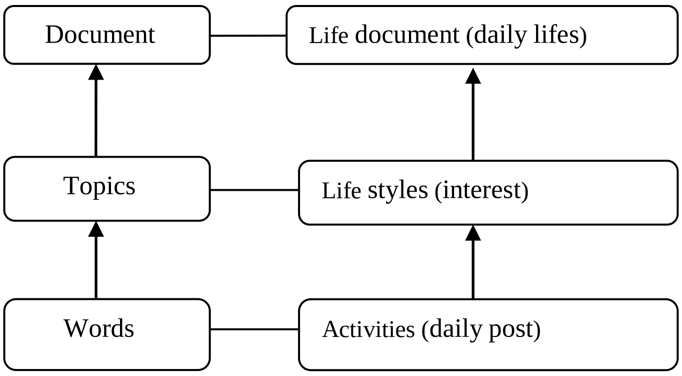

Fig. 1 Life document discovery

The similarity between users are identified using the similarity measure. Text mining has treated document as mixtures of topics, and topics as mixture of words. Similarly, we can treat our daily lives as a mixture of life styles and each life style as a mixture of activities. To the best of our knowledge, Friend-book is the first friend recommendation system exploiting[5] a user's life style information discovered from daily activities. Inspired by achievements in the field of text mining, we model the daily lives of users as life documents and use the probabilistic topic model to extract life style information of users. We integrate a linear feedback mechanism that exploits the user's feedback to improve recommendation accuracy.

II. SYSTEM OVERVIEW

We design high level friend recommendation system in this paper. Fig. 2 will so the proposed system architecture which completely explain about the client-server mode. Here the client is the web application and the server is the data centers. On the client side, the web application is designed using the advanced java. Advanced java will create the user interface for the user with the server. And the typical data generated each day is managed by the MYSQL as our low level data computation. On the server side, the data collected during the time of account creation and daily activities is taken as the input for recommending friends to the user. The life style modeling is used to extract the life style of each user. Activity recognition will recognize the activity and store the preprocessed data in the (life style, user) format. The friend matching graph will match the user with another user based on the life styles. Then the user impact ranking is used to calculate the impact of user on another users using the friend-matching graph.

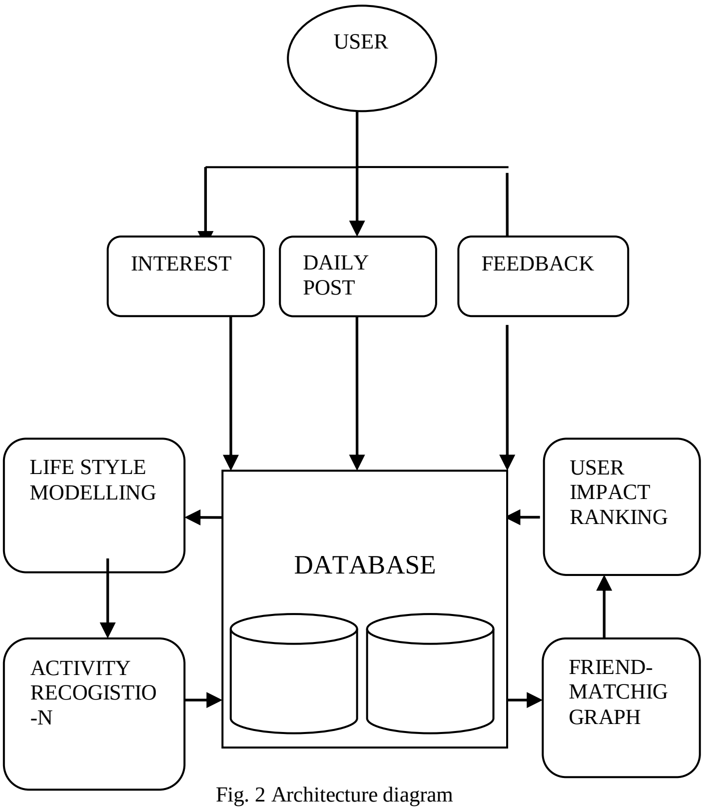

Fig. 2 Architecture diagram

III. PROPOSED SYSTEM

A. *LIFE STYLE MODELLING*

As in the system the life styles and activities are consider as the following model, the life style of the user is treated as the mixture of interest and the interest as the mixture of activities. We adopt the probabilistic topic model to identify the probabilities of hidden interest of the user from the life documents. For the above observation we introduce the " bag-of-activity" model to replace the original sequences of activities recognized based on the raw data with their probability distribution.

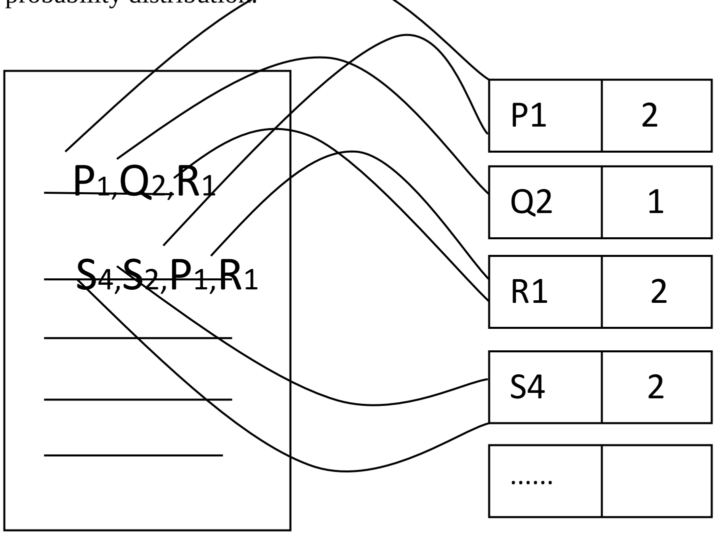

Fig. 3 Bag of activity

Let a={ a_1, $a_{2,\ldots,}$ a_A} denote a set of activities, where a_i is the i^{th} activity and A is the total number of activities. Let b={b_1, $b_{2,}$…., b_B} denote a set of life styles, where b_i is the i^{th} life style and B is the total no. of life styles. Let c={c_1,c_2,…, c_n} denote a set life documents, where c_i is the set of life documents and n is the total no. of users.

$$p(a_i /d_k)= \sum p(a_i/b_j)p(b_j/c_k) \rightarrow 1$$

Observe that $p(a_i /d_k)$ can be easily calculatedby using the bag of activity representation for the life document d_k

$$p(a_i /d_k)=f_k(a_i)/\sum f_k(a_i) \rightarrow 2$$

B. *ACTIVITY RECOGNITION*

We need to first classify or recognize the activities of users. Life styles are usually reflected as a mixture of motion activities with different occurrence probability. Generally speaking, there are two mainstream approaches: supervised learning and unsupervised learning. For both approaches, mature techniques have been developed and tested. In practice, the number of activities involved in the analysis is unpredictable and it is difficult to collect a large set of ground truth data for each activity, which makes supervised learning algorithms unsuitable for our system. Therefore, we use unsupervised learning approaches to recognize activities. The supervised learning is done through the k-means clustering algorithm[6] to group the data into the clusters, where each cluster represents an activity. Other more complicated clustering algorithm can be used, here the K-means for its simplicity and effectiveness.

C. *MATCH MAKER ALGORITHM*

In natural language processing, Match-Maker is a generative model that allows sets of observations to be explained by unobserved groups that explain why some parts of the data are similar. We describe Match-Maker a generative probabilistic model for collections of discrete data such as text corpora. Match-Maker is a three-level hierarchical Bayesian model, in which each item of a collection is modeled as a finite mixture over an underlying set of topics. Each topic is, in turn, modeled as an infinite mixture over an underlying set of topic probabilities. In the context off text modeling, the topic probabilities provide an explicit representation of a document. We present efficient approximate inference techniques based on variational methods and an EM algorithm for empirical bayes parameter estimation. We report results in document modeling, text classification and collaborative filtering, comparing to a mixture of unigrams model and the probabilistic LSI model. We present Match-Maker, a collaborative filtering friend recommendation system based on personality matching. The goal of Match-Maker is to leverage the social information and mutual understanding among people in existing social network connections, and produce friend recommendations based on rich contextual data from people's physical world interactions

D. FRIEND-MATCHING GRAPH

To characterize relations among users, in this section, we propose the friend-matching graph to represent the similarity between their life styles and how they influence other people in the graph. In particular, we use the link weight between two users to represent the similarity of their life styles. Based on the friend-matching graph, we can obtain a user's affinity reflecting how likely this user will be chosen as another user's friend in the network. We define a new similarity metric to measure the similarity between two life style vectors. Based on the similarity metric, we model the relations between users in real life as a friend-matching graph. The friend-matching graph has been constructed to reflect life style relations among users.

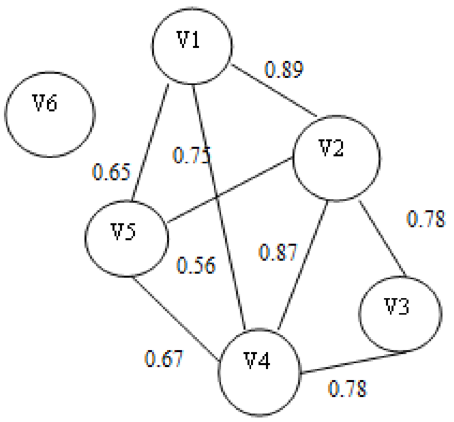

Fig. 4 Friend matching graph

E. USER IMPACT RANKING

The impact ranking means a user's capability to establish friendships in the network. In other words, the higher the ranking, the easier the user can be made friends with, because he/she shares broader life styles with others. Once the ranking of a user is obtained, it provides guidelines to those who receive the recommendation list on how to choose friends. The ranking itself, however, should be independent from the query user. In other words, the ranking depends only on the graph structure of the friend-matching graph, which contains two aspects: 1) how the edges are connected; 2) how much weight there is on every edge. Moreover, the ranking should be used together with the similarity scores between the query user and the potential friend candidates, so that the recommended friends are those who not only share sufficient similarity with the query user, and are also popular ones through whom the query user can increase their own impact rankings

5.1 Algorithm

Input : Friend Matching Graph
Output: Impact Ranking Vector

Step1: for i=1 to n do

Step2: $r_0(i)=1/n$

Step3: end for

Step4: $\mu=\infty$

Step5: $\beta=e^{-9}$

Step6: while $\mu>\beta$ do

Step7:for i=1 to n do

Step8:$r_{k+1}(i)=\sum_j 1-\alpha/n\ r_k(j)+\alpha\sum_j a(i,j).r_k(j)/ \sum_j a(i,j)$

Step9: end for

Step10: $\mu=\sum|r_{k+1}(i)-r_k(i)|$

Step11: end while

Step12: return r

Accordingly to the incremental computation of Page-Rank [7], [8] and the distributed computation of Page-Rank [1], the iterative matrix- vector multiplication method.

IV. CONCLUSION

Our system primarily focuses on creating a social network with strong recommendation system for the user by considering life style, daily activities and feedback from the user. This system will make users to experience the high qualified friend relation with him which helps in gathering information from others.

V. REFERENCE

[1] A.D. Sarma, A.R. Molla, G. Pandurangan, and E. Upal, Fast Distributed Pagerank Computation. Berlin, Germany: Springer, pp. 26-11-13.

[2] J. Kwon and S. Kin ,"Friend Recommendation Method Using Phsical and Social Context", Int. joint conput. Sci. netw. Security, vol. 10, no. 11, pp. 116-120, 2010.

[3] L.Bian and H. Holtzman, "Online Friend Recommendation Through Personality Matching and Collaborative Filtering", in Proc. 5th Int. Conf. Mobile Ubiquitous Comput., Syst., Services Technol., 2011,pp. 230-235.

[4] Hajime Hotta,Masafumi Hagiwara, "User Profiling System Using Social Networks for Recommendation", 8th International Symposium on Advanced Intelligent Systems (ISIS 2007), 2007.

[5] L. Gou, F. You, J. Guo, L. Wu, and X. L. Zhang, "Sfviz: Interest-based friendexploration and recommendation in social networks," in Proc. Visual Inform. Commun.-Int. Symp., 2011, p. 15.

[6] C. M. Bishop, Pattern Recognition and Machine Learning. New York, NY, USA: Springer, 2006.

[7] B. Bahmani, A. Chowdhury, and A. Goel, "Fast incremental and personalized pagerank," Proc. VLDB Endowment, vol. 4, pp. 173–184, 2010.

[8] P. Desikan, N. Pathak, and J. Srivastava, V. Kumar, "Incremental page rank computation on evolving graphs," in Proc. Special Interest Tracks Posters 14th Int. Conf. World Wide Web, 2005, pp. 1094– 1095.