Friend Recommendation System for Social Networks: A Semantic and Profile based Approach

Fathima Mol[1*], Neetha B S[2*]
[1*]PG Scholar, Dept. of computer science and engineering
[2*]Asst. Professor, Dept. of computer science and engineering
TKM Institute of Technology
Kollam, India

***Abstract*— Social networking services which are currently on existence, recommend friends to users based on their social graphs. This graph is based on pre-existing user relationships. This approach does not reflect a user's preferences on friend selection in real life. A new semantic based friend recommendation system for social networks is proposed, where recommendation is based on their life styles. The recommendation system takes advantage of Smartphone sensors to discover life styles of users from user-centric sensor data. In order to improve the accuracy of the life style detected, the system uses similarity of the content extracted from the messages, the video and audio files stored in the Smartphone, and also the details of applications installed in the Smartphone. The proposed system measures the similarity of life styles between users, and recommends friends to users if their life styles have high similarity. The system is implemented on the Android-based Smartphones and its performance is evaluated.**

Index Terms*— Friend recommendation, social networks, life style, mobile sensor, messages, mobile applications.

I. INTRODUCTION

Recommending a good friend to a user is a major challenge with existing social networking services. Most existing social networking site depends on pre-existing user relationships to suggest friends. Some social networking site, for example Facebook depends on social link analysis between people who already share common friends. These recommend symmetrical users as potential friends. Recent sociology findings groups people together into many group based on habits or life styles, attitudes, tastes, moral standards, economic level and people they already know. The main factors considered by existing social networking sites are tastes and people they already know. Life style of a user is closely related to daily routines and activities of user. It is the most instinctive property but is not widely used because a user's life style is difficult to capture. It would be a unique approach, if we could collect user's daily routines and activities information, and recommend friends based on the similarity of life styles. This recommendation mechanism can be implemented on smartphones as a standalone app or can be added to existing social networking frameworks. The application helps users to find friends who have similar life style as the user.

Proposed recommendation system exploit a user's life style information discovered from smartphone sensors. In addition to the sensor data the system also uses the messages (sms and mail inbox), application installed video and audio logs in the smartphone to define the lifestyle. Natural language processing is used to extract information from the messages. The concept of text mining is used to model user's daily life as life documents. A user's life style is extracted by using Latent Dirichlet Allocation algorithm. A similarity metric is used to measure the similarity of life styles between users and user's impact is calculated in terms of life styles with a friend matching graph. On receiving a request the system returns a list of people with highest recommendation scores to the user. The system integrates a feedback mechanism to further improve the recommendation accuracy.

The rest of the paper is organized as follows Section II discusses related work. Section III discusses system architecture. The performance evaluation is shown in Section IV. Finally, the paper is concluded in Section V.

II. RELATED WORKS

There are many recommendation systems that recommend various items such as music, movie, books, etc. to the user. Amazon [11] recommends items based on the items previously visited by the user and which other users visit. Netflix [12] and Rotten Tomatoes [13] are recommendation system which recommends movies to the user based on previous user ratings and watching habits. With the fast advancement of social networking sites, friend recommendation in social networks has become a challenge. Existing social networks recommends friends based on social relationship between users, they have mutual friends.

Friendbook [1] is a friend recommendation system based on similarity of lifestyles between users. The system exploits smartphone sensors to discover lifestyles from user centric sensor data. MatchMaker [2] is a collaborative filtering friend recommendation system which is based on personality matching. In [3] Kwon and Kim proposed a friend recommendation method using physical and social context. In [4] Hsu et al. studied the problem of link recommendation in weblogs and similar social networks. In this an approach based on collaborative recommendation using link structure and content-based recommendation is using mutual declared interest is proposed. In [5] shows the demo of the Friendbook which recommends friends based on similarity of pictures taken by the users.

The paper [6] presented EasyTracker which uses GPS traces collected from smartphones installed on transit vehicles to determine served routes, infer schedules and locate routes. A work closely related to the proposed work is presented in [7] in which activity patterns are extracted using topic model from sensor data. In [8] Farrahi and Gatica-Perez presented a paper to discover daily location driven data from large-scale location data. Reddy et al. in [9] used a built in GPS and accelerometer on smartphone to detect transportation mode of an individual.

III. SYSTEM ARCHITECTURE

Figure 1 shows the system architecture of the system. It adopts a client-server mode.

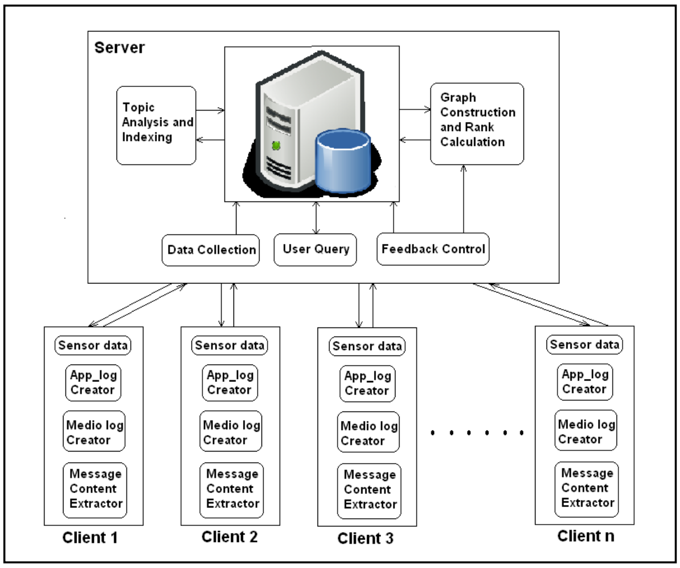

Fig. 1. System Architecture

Each client is a smartphone carried by the user and server is a data centre or a cloud.

On the client side, the data is collected from the smartphone and real time activity recognition is performed. The data collected are sensor data, messages, details of applications installed, and video and audio files stored in the smartphone. MySQL is chosen as the low level data storage platform. For real-time activity recognition on smartphone a suitable activity classifier is built in an offline data collection and training phase. These activity classifiers are distributed to each user's smartphone and real-time activity recognition is done. As the user uses the system continuously, more and more activities is accumulated in his/her life documents. Based on this life document the life style of user is discovered using probabilistic topic modelling.

The work done in the server side is divided into five phases. In the data collection phase the data is collected from the smartphone of the user. Life style of the user is extracted in the topic analysis and indexing phase by using the concept of probabilistic topic model and the lifestyle is stored in the database in the (lifestyle, user) format. The concept of Natural Language Processing (NLP) is used to extract information from the messages. The log details of the video and audio stored in the smartphone and the applications installed in the smartphone, the messages send from and received to the smartphone, and the sensor data are the data collected and these data are used as input in the topic analysis and indexing phase. To represent similarity relation between the lifestyle's of users a friend-matching graph is constructed in the friend-matching graph construction phase. A user's impact rank is calculated based on the friend-matching graph in the user impact ranking phase. Friend query from the user and friend suggestions to the user are made in the user query phase. A user can send feedback of the friend suggestions result, to improve the recommendation accuracy, in the feedback control phase.

A. *Client side*

The client side is implemented in android. The client is a smartphone carried by the user. The main function of the client is to record the activities and send to the server. The data considered by the client and send to the server side is as follows:

a) Sensor data
b) Messages
c) ID3 tag of MP3 file
d) Details of the installed applications.

B. *Topic Modelling*

An analogy is drawn between people's daily lives and documents to model daily lives properly. Previous researches on probabilistic topic models[10] treated document as a collection of topics and topic as a collection of words. Similarly our daily life (life document) is treated as a mixture of lifestyles and each life style as a mixture of activities. The probabilistic topic model (LDA)[10] is used to extract life style of user. The process of life style extraction

In order to extract information from the messages, the concept of topic modelling is used. Topic modelling is a type of text mining. In topic modelling a collection of text is grouped into topics. A topic model is a set of programs that extract topics from the text. A topic is a collection of words. A text can be any kind of unstructured text such as an email, a blog post, a book chapter, etc. It is called unstructured because there is no computer readable interpretation that explains the semantic meaning of the text. Mallet is used to extract topics in the proposed work. The lifestyles extracted by using the topic modelling is then stored in the (lifestyle, user) format in the database.

Privacy is an important consideration especially for users who are sensitive to information leakage. The new system provides two levels of privacy protection. First, the system protects the privacy of user at data level. This is provided by processing the raw data and uploading it, instead of uploading raw data to the servers. These processed data are classified into real time activities and these activities are labeled by integers. The physical meaning of the document is not known to the user because document contains integers. Second, protects users privacy at the life style level. The system shows only the list of recommended users, it does not show the similar life style.

C. Similarity Calculation

The value obtained from the sensor data, messages, details of application installed, and the ID3 tag of MP3 file, all of these account for the calculation of the similarity metric. Let $L_i$and L_j denote the vector representing the topic extracted using the sensor data for user i and user j, respectively.

$$L_i = [p(z_1|d_i), p(z_2|d_i), \dots \dots \dots p(z_z|d_i)\,]$$

Here z_i represent topic and d_i represent document. Similarly L_j is represented.

Let M_i and M_j denote the vector representing the topic extracted from message content from the smartphone of user i and user j respectively.

$$M_i = \left[p(w_1|m_i), p(w_2|m_i), \dots \dots \dots p\left(w_y|m_i\right)\right]$$

Here w_y represent topic or keyword extracted and m_i represent message text. Similarly M_j is represented.

Let A_i and A_j denote the vector representing the details of the applications installed the smartphone of user i and user j respectively.

$$A_i = [p(a_1|a_i), p(a_2|a_i), \dots \dots \dots p(a_n|a_i)\,]$$

Here $p(a_1|a_i)$ represent the probability of the number of similar applications installed by user i given the same by user j. Similarly A_j is represented.

Let I_i and I_j denote the vector representing the details extracted from the ID3 tag of MP3 files stored in the smartphone of user i and user j respectively.

$$I_i = [p(f_1|f_i), p(f_2|f_i), \dots \dots \dots p(f_n|f_i)\,]$$

Here $p(f_1|f_i)$ represent the probability of the number of similar MP3 files stored in the smartphone of user i given the same by user j. Similarly A_j is represented.

The similarity of life styles between user i and user j, denoted by $S(i,j)$ is defined as follows:

$$S(i,j) = \beta \sum \left(L_i \times L_j\right) + \gamma \sum \left(M_i \times M_j\right) + \delta \sum \left(A_i \times A_j\right) + \mu \sum \left(I_i \times I_j\right)$$

where β, γ, δ and μ are constants. These actually show the importance of the data. They are given the values $\beta = 0.5$, $\gamma = 0.5$, $\delta = 0.2$, and $\mu = 0.2$. Sensor data and the message content are given more priority.

Algorithm 1 Computing similarity metric

Input: The query user i, the vectors L, M, A, and I for all n users, each representing sensor data, message content, installed applications, and ID3 tag of MP3 files, the constants β, γ, δ *and* μ.

Output: Similarity metric $S(i,j)$.

```
for each user $j = 1\ to\ n$ do
    $L(j) = \left(L_i \times L_j\right)$
    $M(j) = \left(M_i \times M_j\right)$
    $A(j) = \left(A_i \times A_j\right)$
    $I(j) = \left(I_i \times I_j\right)$
end for
$S(i,j) = \beta \sum L(j) + \gamma \sum M(j) + \delta \sum A(j) + \mu \sum I(j)$
return $S(i,j)$.
```

D. Graph Construction and Rank Calculation

To show the relationship between users a friend graph is constructed using the similarity metric. This graph represents the similarity between lifestyles and how they influence the other people in the graph. The weight on the link between two users represents similarity of life styles. And from this graph we can calculate the impact rank. Impact rank of the user is affinity of the user to establish friendship with the user.

Friend graph is a graph $G = (V, E, W)$, where V is the set of vertices which denotes the users, E is the set of links between users, and W denotes the set of weights of edges. There is an edge $e(i,j)$ between two users if their similarity $S(i,j) \geq S_{thr}$, where S_{thr} is a predefined similarity threshold. The weight on the edge is represented by similarity, i.e. , $w(i,j) = S(i,j)$.

The rank is calculated from the friend graph..Ranking depends on how the edges are connected and the weight on each edge. Higher the rank of a user easier he/she can be made friend with other users. We use the same procedure as in [1] for calculating the rank of the user.

E. User Query and Feedback Control

Before any user can initiate a query, he/she should have accumulated enough activities in his or her life documents for efficient lifestyle analysis. The minimum period for collecting data is usually one day. To get more satisfied friend recommendation results longer time is expected. A system will extract user's life style vector on receiving a user's request and based on this vector recommend friends to the user.

The recommendation results are highly dependent on the user's preference. Some prefer the system to recommend friend with high impact where as some others prefer users with the most similar life styles. Some prefer system to recommend users having high impact and also similar life styles to them. To better categorize the requirement a metric called recommendation score of user [1].

A feedback mechanism is integrated to support performance optimization at runtime. After a reply is generated by the server in response to the query, a user interface is provided to rate the friend list. The feedback mechanism allows us to measure the satisfaction of the user.

IV. PERFORMANCE EVALUATION

The proposed system improves the accuracy of the life style detection as compared to semantic based recommendation system discussed in [1], because it considers both profile and semantic data. As the size of the dataset increases, the accuracy of the recommendation system increases. The rate of the increase of accuracy is more in the case of semantic and profile based friend recommendation system. Accuracy is calculated by taking the average of the weight of edges on the friend graph.

$$\alpha = \frac{\sum_{i,j=1}^{n} w(i,j)}{|E|}$$

where α denotes the accuracy, $w(i,j)$ denote the weight of edge in the friend graph, $|E|$ denotes the total number of edges in the friend graph.

Figure 2 shows comparison of the accuracy of the existing and proposed friend recommendation systems.

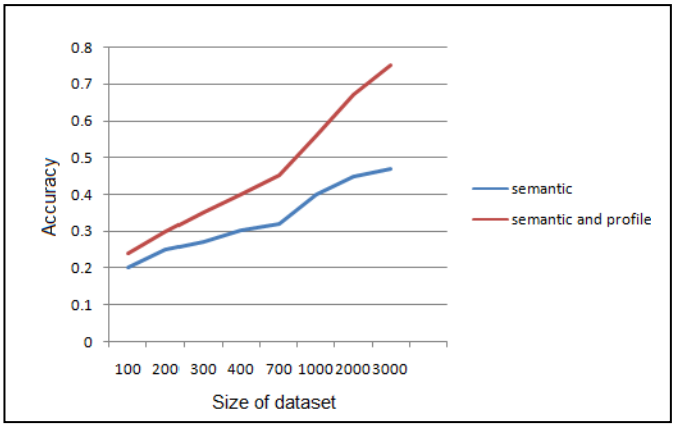

Fig. 2. Accuracy of recommendation system

The semantic based recommendation system considers only the sensor data from the smartphone. Whereas in the semantic and profile based friend recommendation system in addition to the sensor data the system also considers the content of the messages stored in the smartphone, the applications installed in the smartphone and the ID3 tag of the smartphone, also to extract the lifestyles of users. As more data are considered in the proposed method the accuracy of the system is more as compared to the existing system.

V. Conclusion

In this paper, the design and implementation of a semantic and profile based friend recommendation system for social networks is presented. Existing friend recommendation mechanisms rely on social graph; different from these the proposed system extracts the life style of user. In order to extract life style we consider sensor data, messages, application installed and MP3 files stored in the smartphone. The system recommend potential friends if they share similar life styles. The system is implemented on the Android-Based smartphone. The recommendation results show that it accurately reflects the preferences of user for choosing friend.

References

[1]. Z. Wang, C. E. Taylor, Q. Cao, H. Qi, and Z. Wang. Friendbook: A semantic based friend recommendation system for social networks. IEEE Transactions on Mobile Computing, Page(s): 1, 2014.

[2]. L. Bian and H. Holtzman. Online friend recommendation through personality matching and collaborative filtering. Proc. of UBICOMM, pages 230-235, 2011

[3]. J. Kwon and S. Kim. Friend recommendation method using physical and social context. International Journal of Computer Scienceand Network Security, 10(11):116-120, 2010.

[4]. W. H. Hsu, A. King, M. Paradesi, T. Pydimarri, and T. Weninger. Collaborative and structural recommendation of friends using weblog-based social network analysis. Proc. Of AAAI Spring Symposium Series, 2006.

[5]. Z. Wang, C. E. Taylor, Q. Cao, H. Qi, and Z. Wang. Demo: Friendbook: Privacy Preserving Friend Matching based on Shared Interests. Proc. of ACM SenSys, pages 397-398, 2011.

[6]. J. Biagioni, T. Gerlich, T. Merrifield, and J. Eriksson. EasyTracker: Automatic Transit Tracking, Mapping, and Arrival Time Prediction Using Smartphones. Proc. of SenSys, pages 68-81, 2011.

[7]. T. Huynh, M. Fritz, and B. Schiel. Discovery of Activity Patterns using Topic Models. Proc. of UbiComp, 2008.

[8]. K. Farrahi and D. Gatica-Perez. Discovering Routines from Largescale Human Locations using Probabilistic Topic Models. ACM Transactions on Intelligent Systems and Technology (TIST), 2(1),2011.

[9]. S. Reddy, M. Mun, J. Burke, D. Estrin, M. Hansen, and M. Srivastava. Using Mobile Phones to Determine Transportation Modes. ACM Transactions on Sensor Networks (TOSN), 6(2):13, 2010.

[10]. D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet Allocation. Journal of Machine Learning Research, 3:993-1022, 2003.

[11]. Amazon. http://www.amazon.com/.

[12]. Netfix. https://signup.netflix.com/.

[13]. Rotten tomatoes. http://www.rottentomatoes.com/.