

Application of Friend Recommendation System

Tofik R. Kacchi¹, Prof. Anil V. Deorankar²,

¹M.Tech, Dept. of Computer Science Engineering, GCOE, Amravati, Maharashtra, India

²Associate Professor, Dept. of Computer Science Engineering, GCOE, Amravati, Maharashtra, India

Abstract – Existing social networking services recommends friends to users based on their social graphs, which may not be the most appropriate to reflect a user's preferences on friend selection in real life. In this paper, we present a semantic based approach based on daily-activities of users and we deploy this approach to be used for health application. With the help of content features like blood-group and location, we are going to match the blood-group of the searched friend and this matching will lead to help the application search meticulously to stretch out to the desired person. At the end, we will give the mathematical model used in the system and provide its analysis with the help of bar-graph chart.

Key Words: applications of recommender system, cosine similarity relation, dot product, recommendation system.

1. INTRODUCTION

Recommender systems or recommendation systems (sometimes replacing "system" with a synonym such as platform or engine) are a subclass of information filtering system that seek to predict the 'rating' or 'preference' that a user would give to an item. Recommender systems have become extremely common in recent years, and are applied in a variety of applications. The most popular ones are probably movies, music, news, books, research articles, search queries, social tags, and products in general. However, there are also recommender systems for experts, collaborators, jokes, restaurants, financial services, life insurance, persons (online dating), and Twitter followers.

The emergence of social networks from the Internet sparked a major reform in information spread. We have observed that right from the data to search and from search to social interaction; users around the galaxy are now deeply involved with the Internet. With the help of social networks, user-generated record/document is comparatively more accessible than before. A powerful aspect of social networks is the customization of user experiences.

Recommendation systems constitute a large role in providing quality customized user experiences. The major difficulty in developing relevant friend recommendations is due to the non-static nature of humans' perception of friendship, which establishes a cause for heterogeneity in social networks. It is usual and common for humans to change their insight of relationship [3]. Further, this acuity varies from person to person in which a social network can

undergo recurrent and unforeseen change over time even without the starter of new nodes.

2. PROPOSED METHOD

In this system, five modules are designed to fulfill the task of friend recommendation. They are as follows:

1) Data Collection:

Data collection module collects the profile data of users from a file(I/p dataset) which is a semi-structured formatted file. The life styles of users are extracted by these module. The Screenshot of the implementation results of this module is as follows:

It uses hadoop technology because type of file is semi-structured and for fast processing, as tokenizing and extracting the data from a file is easy & optimal in hadoop using the Map/Reduce tool.

2) Data Analysis:

Life document of user is given as input to these module. This module processes the life doc using hadoop technology and prepares its Activity-Bag with the help of knowledge of probability distribution.

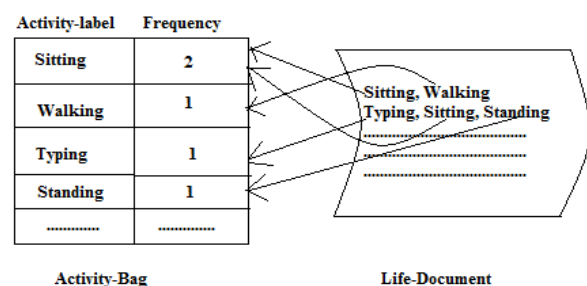


Fig -1: Activity-Bag Representation

Let $w = [w_1, w_2, \dots, w_W]$ be the set of 'W' Activities

$z = [z_1, z_2, \dots, z_Z]$ be the set of 'Z' Lifestyles

$d = [d_1, d_2, \dots, d_n]$ be the set of life-documents

Where $n =$ Number of users and

$p(w_i | d_k)$ is the Probability of activity w_i in a life-doc d_k

$p(w_i | z_j)$ is the Probability of how much the activity w_i contributes to the lifestyle z_j

$p(z_j | d_k)$ is the Probability of lifestyle z_j embedded in a life-document d_k

Let $f_k(w_i)$ be the frequency of occurrence of activity w_i in life-doc d_k . Therefore, we can write:

$$p(w_i | d_k) = f_k(w_i) / \sum_{i=1}^W f_k(w_i) \dots\dots\dots(1)$$

3) Indexing:

Then the life style indexing module puts the life styles of users into the database in the format of (life-style, user) instead of (user, lifestyle).

4) Friend-Matching Module:

A friend-matching module can be constructed accordingly with the help of recommendation algorithm presented here to represent the similarity relationship between users' life styles and suggest the list of similar users.

Algorithm Recommendation(L_i, L_j, λ)

{

Step 1: Design Activity-Bag modeling of the user

Step 2: Use the knowledge of probability distribution and calculate lifestyle vectors

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

Similarly we can construct L_j .

Step 3: Calculate cosine similarity between user 'i' and user 'j'.

$$Sc(i,j) = \cos(L_i, L_j) = L_i \cdot L_j / |L_i| * |L_j|$$

According to these similarity results, it will recommend all user 'j' to the user 'i' depending on the given threshold. This will provide us the list of similar users to a particular user.

Step 4: Sort L_i and L_j in descending order for determining dominant lifestyles.

$$D_i = \{ z_{i_1}, z_{i_2}, \dots, z_{i_{q_i}} \}$$

s.t. $q_i = \arg \min_q (\sum_{k=1}^q p(z_{i_k} | d_i)) \geq \lambda$
is the size of dominant lifestyles.

Similarly we will find D_j .

Step 5: Now $S_d(i, j) = 2 * | D_i \cap D_j | / | D_i | + | D_j |$
It is nothing but the dominant lifestyle score.

5) Recommendation Module:

The user query module takes a user's query and handover it to the recommendation module to send a list of potential friends to the user as response.

This module is divided into two categories:

1) Recommendation by Similarity of lifestyles

2) Recommendation by using Similarity of blood-group & location

Step 6: So our similarity score is determined as follows:

$$S(i, j) = S_c(i, j) \cdot S_d(i, j)$$

Step 7: If these $S(i, j) \geq S_{thr}$ then Recommend
Otherwise Does NOT Recommend.

}

The result of recommendation by matching required blood-group & location is as follows:

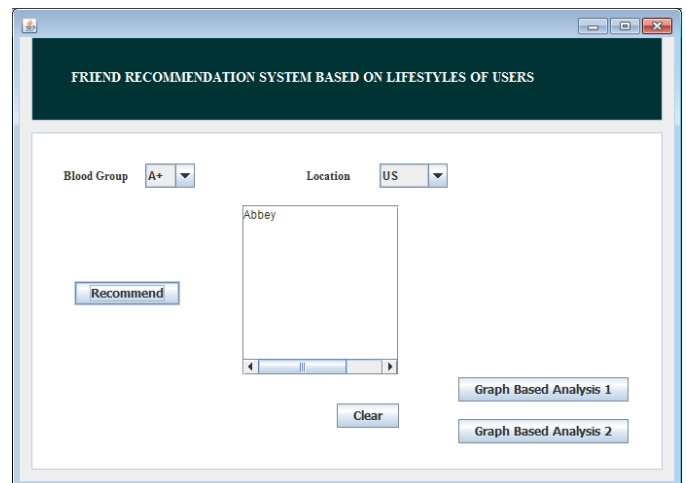


Fig -2: Recommendation by using Similarity of blood-group & location

3. MATHEMATICAL MODEL AND ANALYSIS

In this section we will present the mathematical model and will do the analysis by considering the lifestyle vectors of two users i.e. L_i and L_j . We will present the performance evaluation of friend recommendation system on small-scale field experiments.

Example:

$$\hat{L}_i = [0.3, 0.1, 0.2, 0.3, 0.1] \text{ // Life-style vector of user 'i'}$$

$$\hat{L}_j = [0.2, 0.1, 0.4, 0.01, 0.3] \text{ // Life-style vector of user 'j'}$$

$$\text{We have } Sc(i, j) = \cos(\hat{L}_i, \hat{L}_j) = \hat{L}_i \cdot \hat{L}_j / | \hat{L}_i | * | \hat{L}_j | = 0.18 / (0.4899) * (0.5477)$$

$$\text{Therefore, } Sc(i, j) = 0.6708 \geq 0.5$$

So, user 'j' is a similar user to 'i'.

Example:

$\vec{L}_i = [0.3, 0.1, 0.2, 0.3, 0.1]$ // Life-style vector of user 'i'

$\vec{L}_j = [0.2, 0.1, 0.4, 0.01, 0.3]$ // Life-style vector of user 'j'

$\lambda = 0.8$

$$\begin{aligned} \text{We have } Sc(i, j) &= \cos(\vec{L}_i, \vec{L}_j) \\ &= \vec{L}_i \cdot \vec{L}_j / (|\vec{L}_i| * |\vec{L}_j|) \\ &= 0.18 / (0.4899) * (0.5477) \end{aligned}$$

Therefore, $Sc(i, j) = 0.6708 >= 0.5$

So, user 'j' is a similar user to 'i'.

Now after sorting, $\vec{L}_i = [0.3, 0.3, 0.2, 0.1, 0.1]$

$\vec{L}_j = [0.4, 0.3, 0.2, 0.1, 0.01]$

Therefore, $Di = \{ z_1, z_4, z_3 \}$ as $0.3+0.3+0.2=0.8 >= \lambda$

$Dj = \{ z_3, z_5, z_1 \}$ as $0.4+0.3+0.2=0.9 >= \lambda$

We have

$$Sd(i, j) = 2 * |Di \cap Dj| / (|Di| + |Dj|)$$

Therefore, $Sd(i, j) = 2 * |\{z_1, z_3\}| / (|\{z_1, z_4, z_3\}| + |\{z_3, z_5, z_1\}|)$

$$= 2 * (2) / (3) + (3)$$

$$= 4 / 6$$

So $Sd(i, j) = 0.67$

Sc is calculated as $= 0.6708$

Sd is evaluated as $= 0.67$

$$\begin{aligned} \text{So, } S(i, j) &= Sc(i, j) * Sd(i, j) \\ &= (0.6708) * (0.67) \\ &= 0.45 >= 0.4 \end{aligned}$$

So user 'j' is recommended as a friend to user 'i'. In this way, we can recommend number of friends to a user using lifestyles of users.

There are three free parameters used to generate the friend recommendation results, including the similarity threshold for friend-matching module S_{thr} , the threshold ' λ ' that controls the number of dominant life styles, and the number of life styles. In our practical experiments, we have used the following values as default through empirical studies and trial and error experiments, i.e., the similarity threshold S_{thr} is set to 0.5, the threshold λ is set to 0.8, and the number of life styles is set to 7.

Below Chart 1 demonstrates the distribution of 34 activities. Most of activities have the probability of about 3 percent, some have around 9 percent and three activities have the probability of larger than 32 percent, and the remaining activities have the probability amidst of these.

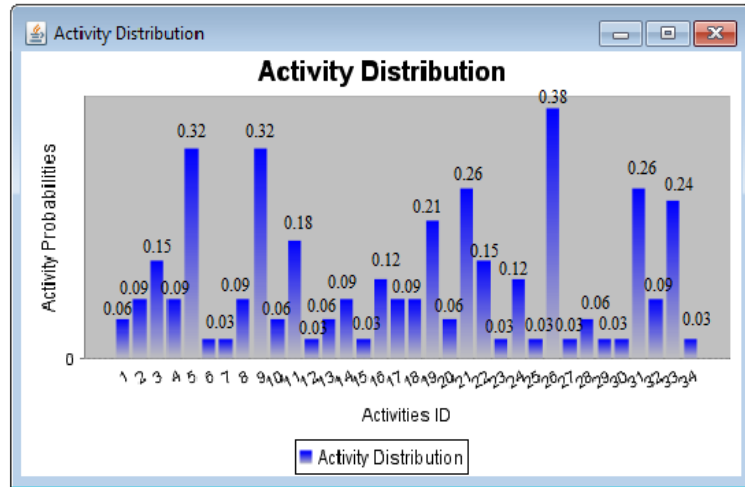


Chart -1: Activity Distribution

Corresponding to each activity, a centroid feature vector is calculated by using the K-means clustering algorithm. These 34 centroids are distributed to each smartphone so that it can perform real-time onboard activity recognition.

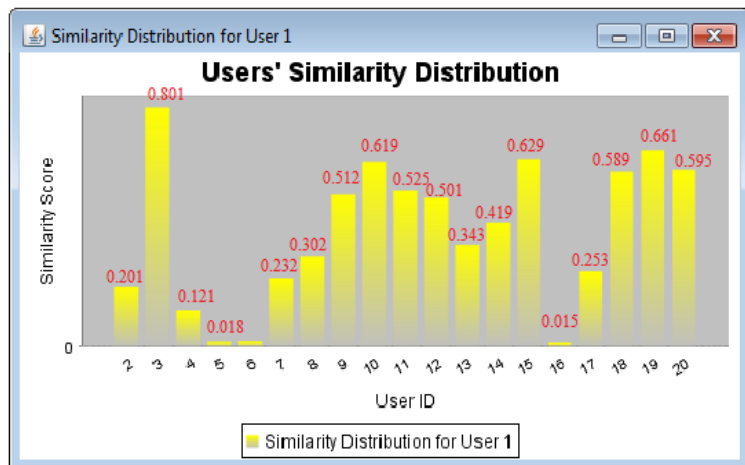


Chart -2: Similarity Distribution for User 1 with only first 20 users

Above chart 2 illustrates the image representation of the similarity distribution for user 1 with nineteen users. The bars in the chart 2 that has the higher peaks (score greater than 0.5) are the similar users to the user 1. As shown in figure of chart 2, user 1 has strong relationship with user 3, user 19, user 15 and user 10, while user 5, user 6 and user 16 have comparatively no relationship with user 1 at all. The result is consistent with the ground truth of professions present in the dataset considered because people have the same profession usually have the same life styles.

In this way, for each user considered, we can easily find the similarity relationship between any numbers of users.

4. APPLICATIONS

In this paper, we are using the parameters blood-group and location to search for the person with desired matching blood and nearby location so that the required blood from the desired person should be quickly and easily available. This is one of the major health application of the friend recommendation system.

We can add content features like DNA of a person, other ID's, etc. So with little modification, we can easily find the person with matching DNA sequences or ID's by searching on this system. This can be helpful for the intelligence bureau/CID/police in case of accidents or murder of a person who is unidentifiable.

5. CONCLUSIONS & FUTURE SCOPE

We have observed that almost ninety percent of the young generation people (especially students) are using the social networking sites like Facebook, Twitter, LinkedIn, etc. That means these are the fastest and best medium to reach to maximum people. So if we deploy our recommendation system on these social services then it can be very helpful for all.

In this paper, we have designed and implemented friend recommendation system that allows users with similar interests to be quickly identified and recommended, to get one's interest without one's specification, to quickly find person according to matching blood group and nearby location.

This application can be deployed on Android based smartphones for mobile app by using the sensor-rich smartphones having the sensors like GPS(motion sensors), accelerometer, gyroscope, etc. and tracking user activities using these. Beyond the current prototype, the future work can be four fold.

- We would like to evaluate our system on large-scale field experiments.
- To construct the system to be scalable
- The similarity threshold used for the friend-matching module is fixed in our proposed work. It would be interesting to explore the adaption of the threshold for each edge and see whether it can better represent the similarity relationship on the friend-matching module.
- Actually, we expect to incorporate these into existing social services (e.g., Facebook, Twitter, LinkedIn) so that recommendation system can utilize more information for life discovery, which should improve the recommendation experience in the future.

REFERENCES

- [1] Zhibo Wang, Hairong Qi, "Friendbook: A Semantic-Based Friend Recommendation System for Social Networks," IEEE Transactions on Mobile Computing, Vol. 14, No. 3, MARCH 2015.
- [2] Pankaj L. Pingate, S. M. Rokade, "A Survey of Friendbook recommendation Services ", International Journal of Science and Research (IJSR), ISSN (Online): 2319-7064, Vol 3, Issue 11, November 2014.
- [3] S.Adsure, A.Arane, A.Chavhan, R.Jagdhane, A.Pardeshi, "A Survey on Friendbook Using Semantic based Friend Recommendation System," IJARIE-ISSN(O)-2395-4396, Vol. 1, Issue-4, 2015.
- [4] B. Bahmani, A. Chowdhury, and A. Goel, "Fast incremental and personalized pagerank," Proc. VLDB Endowment, vol. 4, pp. 173– 184, 2010.
- [5] 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.
- [6] Z. Wang, C. E. Taylor, Q. Cao, H. Qi, and Z. Wang, "Demo: Friendbook: Privacy preserving friend matching based on shared interests," in Proc. 9th ACM Conf. Embedded Netw. Sens. Syst., 2011, pp. 397–398.
- [7] Tom White, "Hadoop: The Definitive Guide," Vol. No. 2, OCTOBER, 2010, pp. 15-40.
- [8] L. Page, S. Brin, R. Motwani and T. Winograd, "The Pagerank citation ranking: Bringing order to the web ", Stanford InfoLab, Stanford, CA, USA, Tech. Rep. 1999-66, 999
- [9] T. Huynh, M. Fritz, and B. Schiel, "Discovery of activity patterns using topic model," in Proc. Int. Conf. Ubiquitous Comput., 2008, pp. 10-19.
- [10] Amazon. (2014). [Online]. Available: <http://www.amazon.com/>
- [11] Facebook statistics. (2011). [Online]. Available: <http://www.digitalbuzzblog.com/facebook-statistics-stats-facts-2011/>
- [12] Netflix. (2014). [Online]. Available: <https://signup.netflix.com/>
- [13] Rotten tomatoes. (2014). [Online]. Available: <http://www.rottentomatoes.com/>
- [14] J. Biagioni, T. Gerlich, T. Merrifield, and J. Eriksson, "EasyTracker: Automatic transit tracking, mapping, and arrival time prediction using Smartphones," in Proc. 9th ACM Conf. Embedded Netw. Sensor Syst., 2011, pp. 68–81.
- [15] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," J. Mach. Learn. Res., vol. 3, pp. 993–1022, 2003.