A Survey on Semantic-based Friend Recommendation System for Social Networks

Bhavanasi Geethika¹, Ch. Raja Jacob²

¹M-Tech, Dept. CSE, Nova College of engineering and technology jangareddigudem, west Godavari
²Associate Professor Dept. CSE, Nova College of engineering and technology jangareddigudem, west Godavari

Abstract: Friend book is a novel semantic-based friend recommendation system for social networks, based on their life styles instead of social graphs which recommend friends to users. Existing social networking services recommend 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. User’s daily life is modeled as life documents, from which users life styles are extracted by using the Latent Dirichlet Allocation algorithm; Similarity metric to measure the similarity of life styles between users, user’s impact is calculated in terms of life styles with a friend-matching graph. In this paper, a social network is formally represented and taking text mining as a perspective, we have proposed a framework that will recommend friend using an efficient Algorithm. We find solution in proposed work, the factor of personal interest can make the recommend items to meet users' individualities based on keywords and location in searching. We suggest users while searching based on user’s search history. We process the search history based on keywords and location. Experimental results show the proposed approach outperforms the existing approaches.

Keywords: Friend recommendation; life style; social networks

1. INTRODUCTION

Existing social networking services recommend 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 Friendbook, a novel semantic-based friend recommendation system for social networks, which recommends friends to users based on their life styles instead of social graphs. By taking advantage of sensor-rich smartphones, Friendbook discovers life styles of users from user-centric sensor data, measures the similarity of life styles between users, and recommends friends to users if their life styles have high similarity. Inspired by text mining, we model a user’s daily life as life documents, from which his/her life styles are extracted by using the Latent Dirichlet Allocation algorithm. We further propose a similarity metric to measure the similarity of life styles between users, and calculate users’ impact in terms of life styles with a friend-matching graph. Upon receiving a request, Friendbook returns a list of people with highest recommendation scores to the query user. Finally, Friendbook integrates a feedback mechanism to further improve the recommendation accuracy. We have implemented Friendbook on the Android-based Smartphone, and evaluated its performance on both small-scale experiments and large-scale simulations. The results show that the recommendations accurately reflect the preferences of users in choosing friends.

As time passes, World Wide Web (WWW) goes on growing. Lots of information is available on WWW. All the information which we get is not
relevant, only few of them are relevant. When a user tries to search something on WWW s/he lands up with thousands of result. As a result, s/he will mess up with huge information. Hence fetching the actually required details becomes cumbersome and time consuming. This gives rise to data filtering system. In early days, for data filtering, Information Filtering (IF) was used. IF was basically developed for filtering documentation, articles, news etc. Looking to our era, e-commerce is growing explosively. Whenever a user makes a search for particular item on internet to buy, s/he will get many options. Looking at the options user gets confuse what to buy, and will not able to sort the item that is suitable to him/her. This problem gave rise to Recommendation System [RS]. A recommender system is a personalization system that helps users to find items of interest based on their preferences. Recommender systems are efficient tools that overcome the information overload problem by providing users with the most relevant contents [8].

The importance of contextual information has been recognized by researchers and practitioners in many disciplines including Ecommerce, personalized IR, ubiquitous and mobile computing, data mining, marketing and management. There are many existing e-commerce websites which have implemented recommendation systems successfully. We will discuss few website in our coming section that provides recommendation. Items are suggested by looking at the behavior of like-minded-users. Groups are formed of such users, and items preferred by such groups are recommended to the user, whose liking and behavior is similar to the group. In our model we have incorporated user preferences obtained from Social Networking Site. Social Networking sites have the largest data set of users. Each social networking site notes/records each and every activity of user (like: what user likes? what user is doing? what is user’s hobby? Etc). Social Networking site will prove to be largest domain in understanding the user behaviour. One of the best examples of social networking is FACEBOOK. According to current news FACEBOOK is trying to develop algorithm, to understand user behavior. Social Networking sites can help us in getting important information of users, such as age, gender, location, language, actives, likes etc. our model takes into account these parameters of the user to recommend books. Most of the friend suggestions mechanism relies on pre-existing user relationships to pick friend candidates. For example, Facebook relies on a social link analysis among those who already share common friends and recommend symmetrical users as potential friends. The rules to group people together include:

1) Habits or life style
2) Attitudes
3) Tastes
4) Moral standards
5) Economic level; and
6) People they already know.

Apparently, rule #3 and rule #6 are the mainstream factors considered by existing recommendation systems.

2. RELATED WORK

E. Miluzzo et al. [4] enables technology from mobile phone sensing to know about human behavior and context on mobile phones which uses
combination of collaborative sensing and classification techniques. This is the first system that applies distributed machine learning techniques and collaborative inference concepts to mobile phones. To achieve better interference accuracy we are using collaborative sensing. Machine learning techniques specifically designed to run directly on sensor enabled mobile phones. This paper is an automated approach to updating models over time such that the classifiers are robust to the variability in sensing conditions and settings common to mobile phones.

Advantages

- The classifiers are robust.
- The classifier methods are automated to update the models.
- Collaborative inference is implemented to achieve better accuracy.

K. Farrahi et al. [3] investigate probabilistic topic models as unsupervised machine learning tools for largescale socio-geographic activity mining. They propose a methodology based on Latent Dirichlet Algorithm (LDA) for the discovery of dominant location routines. In this paper they used two Probabilistic model namely Multi-Level Topic Model and Pair wise – Distance Topic Model. First they propose a Multi-Level Topic Model as a method to incorporate multiple time duration sequences into a probabilistic generative topic model. And then they propose the Pair wise-Distance Topic Model as an approach to address the problem of modeling long duration activities with topics. Overall, this thesis addresses investigations principled on mathematical models and multiple types of mobile phone sensor data are performed to mine real life human activities in large-scale scenarios.

Advantages

- Used for computing large scale data.
- Removes the need for coarse time-slot

T. Huynh et al. [2] develop an unsupervised methodology based on two differing probabilistic topic models and apply them to the daily life. They have proposed a method to represent location sequences, and incorporated this into the LDA and ATM topic models. The resulting distributions of words for latent topics, as well as topics given days, and topics given users, reveal hidden structure of routines which use to perform varying tasks, including finding users or groups of users that display given routines, and determining times.

Advantage

- Used to compute large amount of data easily

3. SYSTEM MODEL

![System architecture](Fig-1-System-architecture.png)

Life style modeling

Life styles and activities are reflections of daily lives at two different levels where daily lives can be treated as a mixture of life styles and life styles as a mixture of activities. This is analogous to the treatment of documents as ensemble of topics and topics as ensemble of words. By taking advantage of recent developments in the field of text mining, we model the daily lives of users as life
documents, the life styles as topics, and the activities as words.

![Activity Recognition Diagram](image)

**Fig 2: Bag of Activity**

**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. Here, we adopt the popular K-means clustering algorithm to group data into clusters, where each cluster represents an activity. Note that activity recognition is not the main concern of our paper. Other more complicated clustering algorithms can certainly be used. We choose K-means for its simplicity and effectiveness.

**Friend matching graph construction**

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. 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.

**User impact ranking**

The impact ranking means a user’s capability to establish friendships in the network. 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. 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. This can be achieved using Weighted Page Rank algorithm.

**4. PROPOSED WORK**

During the web development phase, the user data is recorded into our database. The user activity from the database is accessed. An algorithm for calculating dominating lifestyle vector of user is developed. LDA algorithm is away of automatically discovering topics that the sentences in document has, it finds the topic by calculating the probability of words in document. Similarly in case of Facebook we apply this method and find the dominant life style vector as below. The life style of users is extracted by the life style analysis module with the probabilistic topic model, and then the life style indexing module puts the life styles of users into the database in the
format of (life-style,user) The probabilistic topic model can be given as,

\[ P(W_i |d_k) = \sum_{j=1}^{Z} p(w_i |z_j) p(z_j |d_k) \]

Where, w - activity

Z - life style

D – set of document and in our case as we are implementing it in facebook, dk can be considered as 1 as we are able to fetch the topics directly by considering user activity as whole document.

The topics may be movies, books read, sports etc. and the count of these activities can be accessed based on the permission given by FaceBook developers and the people who logs in to our app and allow us to access the data to recommend them friend of similar interest among the people from our database. As we get the count values we calculate probability of each activity of user using above formula then we find dominating life style vector of user by specifying some assumed threshold value in our case we have considered it as 0.2. Let us define this threshold as \( \alpha \) (alpha) And after finding dominating life style vector of user we find similarity between the users this is done using the below formula,

\[ S = S_c(i,j).S_d(i,j). \]

where i & j are number of users \( S_c \) = is cosine similarity and \( S_d \) = is distance similarity.

Hence, a cosine similarity can be calculated as below Between user 1 & user 2,

\[ S_c(U1,U2) = \cos(U1,U2) = \frac{a.b}{|a||b|} \]

Similarly with all the users it is calculated. And distance Similarity is calculated as below,

\[ S_d(U1,U2) = 2|D1 \cap D2|/|D1|+|D2|. \]

After calculating similarity value for all the user with every Other user we store those values in matrix form from which We recommend a friend to the user who is greater then Some specified threshold value, we have assumed Threshold value as 0.5 in our case and let’s consider this Recommending threshold as \( \beta \) (beta). In the proposed work, we have focused on four important phases as below Fig.

a. Creating a user interface application for login: Web applications that require authorization to access certain information. Your login page verifies a user’s name and password, places a cookie on the user’s computer so he can return later, and uses database queries to retrieve the personal information for the user.

b. Extracting user data and storing in database: We use Graph API tools for extracting data. The advantages of Graph API over previous work are the ability to learn highly accurate extraction rules, and then we store this user information like ‘name’, ‘email’, likes, in the database that we have created.

c. Finding dominant life style: Depending on the activities that user has done we get certain count of the activity, then we calculate probabilities of each life style and consider those values who are greater then some specified threshold value \( \alpha \) (alpha). In which the user interacts with the site through our application.

d. Recommending potential friend: We calculate the similarity between the users and recommend friends to the query user who are above certain threshold value \( \beta \) (beta).
5. RESULT MODULE

5.1 Bag of Activity of User1

```
<table>
<thead>
<tr>
<th>Active Label</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>class</td>
<td>108</td>
</tr>
<tr>
<td>cloud</td>
<td>274</td>
</tr>
<tr>
<td>computing</td>
<td>135</td>
</tr>
<tr>
<td>download</td>
<td>6</td>
</tr>
<tr>
<td>java</td>
<td>186</td>
</tr>
<tr>
<td>method</td>
<td>76</td>
</tr>
<tr>
<td>programming</td>
<td>10</td>
</tr>
<tr>
<td>quick</td>
<td>10</td>
</tr>
<tr>
<td>reference</td>
<td>10</td>
</tr>
<tr>
<td>resources</td>
<td>8</td>
</tr>
<tr>
<td>retrieved</td>
<td>149</td>
</tr>
<tr>
<td>this</td>
<td>12</td>
</tr>
<tr>
<td>tutorial</td>
<td>36</td>
</tr>
</tbody>
</table>
```

5.2 LDA Model

<table>
<thead>
<tr>
<th>No</th>
<th>Set name</th>
<th>Set of words</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Children</td>
<td>kids, toys, games, teddy.</td>
</tr>
<tr>
<td>2</td>
<td>cloud computing</td>
<td>cloud, online database, hybrid cloud, multi cloud, cloud technologies</td>
</tr>
<tr>
<td>3</td>
<td>Computer Language</td>
<td>c language, java, net, dot net, cobol, pascal, web technologies, java,</td>
</tr>
<tr>
<td>4</td>
<td>Finance</td>
<td>online Banking, internet banking, mobile banking, mini statement, ba rupees, banking, exchange, currency</td>
</tr>
<tr>
<td>5</td>
<td>Sports</td>
<td>cricket, football, hockey, soccer, rap-bee</td>
</tr>
</tbody>
</table>

5.3 User Life Style Index

```
User Life style
User1 cloud computing, Computer Language
User2 Computer Language
User3 cloud computing
```

5.4 User Reverse Index

<table>
<thead>
<tr>
<th>Life style</th>
<th>User's</th>
</tr>
</thead>
<tbody>
<tr>
<td>cloud computing</td>
<td>user1, user3,</td>
</tr>
<tr>
<td>Computer Language</td>
<td>user1, user2,</td>
</tr>
</tbody>
</table>
| Illustration of the reverse index table.

6. CONCLUSION

In our approach we presented the design and implementation of FriendBook, a semantic-based friend recommendation system for social networks. Different from the friend recommendation mechanisms relying on social graphs in existing social networking services, the results showed that the recommendations accurately reflect the preferences of users in choosing friends. Beyond the current prototype, the future work can be concentrated on implementing it on other social networking, and same can be used to build stand alone app and access the user activity through mobile sensors. FriendBook can utilize more information for life discovery, which should improve the recommendation experience in the future.

7. REFERENCES


