**.NET Project On Friendbook: A Semantic-based Friend Recommendation System for Social Networks**

**ABSTRACT:**

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 smartphones, 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.

**EXISTING SYSTEM:**

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 recommends symmetrical users as potential friends. The rules to group people together include:

* Habits or life style
* Attitudes
* Tastes
* Moral standards
* Economic level; and
* People they already know.

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

**DISADVANTAGES OF EXISTING SYSTEM:**

* 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

**PROPOSED SYSTEM:**

* 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.
* 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.
* 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.
* We integrate a linear feedback mechanism that exploits the user’s feedback to improve recommendation accuracy.

**ADVANTAGES OF PROPOSED SYSTEM:**

* Recommend potential friends to users if they share similar life styles.
* The feedback mechanism allows us to measure the satisfaction of users, by providing a user interface that allows the user to rate the friend list

**MODULES:**

* Life Style Modeling
* Activity Recognition
* Friend-matching Graph Construction
* User Impact Ranking

**MODULES DESCRIPTION:**

**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. Given “documents”, the probabilistic topic model could discover the probabilities of underlying “topics”. Therefore, we adopt the probabilistic topic model to discover the probabilities of hidden “life styles” from the “life documents”. Our objective is to discover the life style vector for each user given the life documents of all users.

**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.

**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. 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.

**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.

**SYSTEM REQUIREMENTS:**

**HARDWARE REQUIREMENTS:**

* System :         Pentium IV 2.4 GHz.
* Hard Disk :         40 GB.
* Floppy Drive : 44 Mb.
* Monitor : 15 VGA Colour.
* Mouse :
* Ram : 512 Mb.

**SOFTWARE REQUIREMENTS:**

* Operating system : Windows XP/7.
* Coding Language : JAVA/J2EE, .net
* IDE : Netbeans 7.4
* Database : MYSQL