Geo-Social Profile Matching Algorithm for Dynamic Interests in Ad-Hoc Social Network

Nagender Aneja, Sapna Gambhir
Department of Computer Engineering, YMCA University of Science and Technology, Faridabad, India
Email: naneja@gmail.com, sapnagambhir@gmail.com

Received 20 August 2014; revised 25 September 2014; accepted 20 October 2014

Abstract
Among mobile users, ad-hoc social network (ASN) is becoming a popular platform to connect and share their interests anytime anywhere. Many researchers and computer scientists investigated ASN architecture, implementation, user experience, and different profile matching algorithms to provide better user experience in ad-hoc social network. We emphasize that strength of an ad-hoc social network depends on a good profile-matching algorithm that provides meaningful friend suggestions in proximity. Keeping browsing history is a good way to determine user's interest, however, interests change with location. This paper presents a novel profile-matching algorithm for automatically building a user profile based on dynamic GPS (Global Positioning System) location and browsing history of users. Building user profile based on GPS location of a user provides benefits to ASN users as this profile represents user’s dynamic interests that keep changing with location e.g. office, home, or some other location. Proposed profile-matching algorithm maintains multiple local profiles based on location of mobile device.

Keywords
Ad-Hoc Social Networks, User Profile, Dynamic Interests, Friends, Profile Matching, Search and Browsing History

1. Introduction
Ad-hoc social network (ASN) is a social network between users of mobile devices that are connected with an ad-hoc network. Today most of mobile phones are equipped with Bluetooth, Wi-Fi, and Cellular radio and so are capable of supporting ad-hoc communication mode. ASN is a social network where people of similar inter-
ests connect with each other using *ad-hoc* communication mode of mobile devices. Being connected using *ad-hoc* communication mode, ASN has its own advantages as it uses infrastructure-less network for communication.

2. Problem Definition

People are always looking for information and friends. Lee and Hong [1] observed a correlation in browsing web and user’s interests and proposed an algorithm to suggest friends by constructing a dynamic profile based on keywords extracted from urls accessed while browsing Internet. However, many times urls don’t include keywords or relevant keywords. Thus the profile based on keywords extracted from visited urls may not represent user’s interests rather keywords used to search the information represent their interests. Additionally, there is a correlation between search pattern (i.e. browsing history or interests) and location of searching information. Thus, the proposed algorithm follows a three prong approach where keywords used in searching, keywords extracted from accessed urls, and GPS location of the user used to search and browse are considered for constructing profile of the user. Further, building a single profile for a user and suggesting friends based on that single profile is not beneficial to the user. Since, a user has different search and browsing pattern at office, home, or at some other tourist place and its behavior changes significantly with respect to location of the user. For example, Person A while at work may usually accesses urls related to Java programming or urls related to job, while at home accesses urls related to movies, restaurants, parks, and while visiting other tourist places he accesses urls related to sharing taxi, locations to be visited, or other information within that tourist place. Therefore, suggesting social links based on a single global profile will not help to get a good experience of ASN. For example, at a tourist place, Person A may be interested in creating social links with persons having interests of sharing taxi to a particular place to share costs or persons having interests in Java Programming and along with interest of sharing taxi to a particular place so that he can share a taxi with a person with interests as he has. Therefore, there is a need to have different profiles for user that may automatically be set based on their GPS location. A user may always set a global profile that is a combination of some oral local profiles.

3. Related Work

Sarigol et al. [2] presented Ad Social that supports social network applications in an *ad-hoc* network and demonstrated on 10 - 15 Nokia N810 handhelds with a very low overhead. In a typical online social network (e.g. facebook) a user’s list of buddies consists of friends explicitly selected by the user, while in *ad-hoc* social networks, buddies are nearby users whose presence has been detected by Ad Social. Users can retrieve profile of any nearby buddy by right-clicking on the buddy’s icon and start a chat session with them. Alternatively, they can also search for buddies matching some specific interest. However, Ad Social matches interests using a simple string matching algorithm and requires user to enter his/her profile manually. Trieuand Pham [3] proposed a system called STARS, which is an *ad-hoc* network of smart phones. It is a network data sharing model where in users choose to share information with other people within a small group for limited amount of time. The system provides features to build social network and share interests like pictures, drawings, notes, comments, text etc. User registers an identifier on time line of *ad-hoc* network. The application broadcasts the identifier in the network. A decentralized application running on user’s device creates Interest-based network and perform security and privacy enforcement. Campbell et al. [4] proposed to add sensing capability into social networking applications. They presented a system called Cence Me which collects information about neighboring users and concise facts which can be used in many applications.

Sarigol et al. [5] presented a tuple space that abstracts underlying network as a common memory space in which nodes can store and look up key/value pairs (i.e., tuples). However, each application configures its own “shared memory” rather than all tuples residing in a single shared memory. Moreover, an application can control propagation of its tuples both in space and time. They used tuple space to implement a buddy presence service that allows users to view all buddies in their proximity as well as search for buddies with specific interests. Every user creates a profile that includes a list of interests. The profile is exchanged among users to filter friends. However, this method does not discuss about determining interests automatically to create a profile for the user. Yiu et al. [6] presented an application which locates friends in proximity based on given threshold Euclidean distance. They proposed that application tunes its proximity distance itself according to communication cost. Bottazzi et al. [7] proposed a middle ware for ASNs where in two components called Dependent Social Network Manager (PSNM) and Global Network Manager (GSNM) are presented for building user profile. PSNM pub-
N. Aneja, S. Gambhir

various research issues in establishing hierarchical model to infer user’s interest from urls of accessed web pages. User’s interests are continuously changing with respect to location. Nagender and Sapna [11] presented the need to investigate inter-based matching. They presented that entering interests or creating profile is not a good and the process of creating profile for ASN should be automated. The authors discussed that the solution to create automatic profile based on browsing history is good but may not be beneficial since users’ interests vary with movements and are based on location.
Therefore, current research in *ad-hoc* social networks demonstrated that AS neither can be an extension to on-line social network or an in dependent temporary social network formed for specific purpose. In order to provide better and satisfying experience of *ad-hoc* social network, the profile matching algorithm should provide meaningful friend suggestions. As mentioned, Lee and Hong [1] algorithm uses keywords extracted from urls to create profile of the user, thus there is a need of algorithm that not only uses keywords extracted from urls but also uses key words that have been used to search information and access the urls. Further, *ad-hoc* social network is create data specified location for some special purpose, there is also a need to have dynamic multiple *ad-hoc* profiles for user that changes with location of the user.

4. Algorithm

This paper presents a hierarchical model wherein first level children are GPS locations visited by a user and second and lower level children represent interests extracted from users search and browsing pattern. Further, a user can manually set the profile or can be automatically selected based on its GPS location. The proposed profile matching algorithm includes two phases 1) profile construction phase and 2) profile matching phase. Profile construction phase automatically creates multiple profiles wherein each profile is tagged with a GPS location and browsing history. Profile matching phase extracts meaningful keywords from browsing history and create a tree with root as GPS location and children as extracted keywords. The second phase *i.e.* profile matching phase uses cosine similarity to determine if the two profiles are similar enough to suggest social connection.

4.1. Profile Construction Phase

When a user searches information on a mobile device and accesses a web page through a web browser, system records keywords used in searching and visited urls along with its GPS location. GPS location is the location of the mobile device from where the user is searching some information. In order to infer user’s interest, the system extracts meaningful words from recorded urls. The algorithm uses a forest structure as mentioned in Figure 1.

**Algorithm 1** explains steps to build multiple local profiles. Current GPS location of the user is compared with GPS locations stored in the data structure list. If current GPS location is found in the list, then corresponding hierarchical structure’s address is extracted. Keywords used in searching and extracted from visited urls are compared with all of children nodes in the hierarchical structure. If the extracted word does not exist in the hierarchical structure, it is added to the structure and its interest level is initialized, otherwise the interest level of the word is increased. If GPS location is not present in the forest, then new tree with root node as GPS location is added in the forest along with keywords and interest level initialized as zero.

We also store GPS location in a form of meaningful names. For example GPS location can be University Campus (Office), 14 Street (Home), Tourist (may be a tourist place for a user). Therefore, a user has as many local profiles as number of trees in a forest. Active profile of a user would be based on its current GPS location. However, the user can select one or more active profiles. If a user selects multiple local profiles then interest level of the keywords is added to make a combined sub-profile. Social links would be created based on representative...
Node\textsubscript{A} - Node in ad-hoc social network
FA - Forest maintained by Node\textsubscript{A}
LA - List of GPS Locations of Node\textsubscript{A}
LA\textsubscript{t} - GPS Location of Node\textsubscript{A} at time \textit{t}
List\textsubscript{A}(LA\textsubscript{t}) - Hierarchical data structure of keywords for location LA\textsubscript{t}
K\textsubscript{i} - \textit{i}\textsuperscript{th} Keyword
I\textsubscript{Ki} - Interest level of \textit{i}\textsuperscript{th} Keyword
N - Total no of levels
m - Number of keywords at \textit{j}\textsuperscript{th} level
Trigger: Node\textsubscript{A} receives K\textsubscript{i} from search and browsing urls and receives LA\textsubscript{t} <\textgreater
\begin{align*}
\text{If} & (LA\textsubscript{t} \text{ exists in } FA) \text{ then} \\
& \quad \text{return } List\textsubscript{A}(LA\textsubscript{t}) \text{ from } FA \\
& \quad \text{updateInterests()} \\
\text{else} & \\
& \quad \text{empty tree } List\textsubscript{A}(LA\textsubscript{t}) \text{ added to } FA \\
& \quad \text{return } List\textsubscript{A}(LA\textsubscript{t}) \\
& \quad \text{updateInterests()} \\
\end{align*}
end if
Function:
updateInterests()
scan List\textsubscript{A}(LA\textsubscript{t})
if (K\textsubscript{i} found in scanning) then
\begin{align*}
& \quad I_{K_i} ++ \\
\text{else} & \\
& \quad \text{insert } K_i \text{ into } List\textsubscript{A}(LA\textsubscript{t}) \\
& \quad I_{K_i} = 0 \\
\end{align*}
end if

**Algorithm 1. Algorithm to create profiles based on GPS location.**

keywords from the combination of profiles.
Suppose Person A spends most of the time at office or at home but at weekend he goes to different places say parks, shopping malls etc. At such location he may still want to publish his global profile instead of default local profile based on current location. We propose that a global profile is built by adding weights of interest level and for all locations. Let FA is the Forest of Node\textsubscript{A} with various locations. A set of keywords are formed by taking union of keywords in all the trees at various location. For each common keyword in more than one tree, weight is computed by adding its interest level in each tree.

4.2. Profile Matching Phase

**Figure 2** presents profile matching algorithm to match profile of a node with profile of nodes in its range. Node extracts keywords and their corresponding interest levels based on current profile that may be local profile or global profile depending on user’s choice.

Suppose for a Node\textsubscript{A} set of keywords is \{k\textsubscript{1}, k\textsubscript{2}, \ldots, k\textsubscript{p}\} with corresponding set of interest levels or weights (w\textsubscript{1}, w\textsubscript{2}, \ldots, w\textsubscript{p}). Node\textsubscript{B} has a set of keywords \{l\textsubscript{1}, l\textsubscript{2}, \ldots, l\textsubscript{q}\} with interest levels or weights (u\textsubscript{1}, u\textsubscript{2}, \ldots, u\textsubscript{q}). In order to match interests, individual profiles are modified in relation to received profile of other user.

4.2.2. Profile to Be Matched

In order to match, a set of keywords is formed by taking union of Node\textsubscript{A}’s and Node\textsubscript{B}’s keyword sets. For keyword which is present in profile of one node set but not in profile of other node set, its weight is taken as zero. Now suppose (w\textsubscript{1}, w\textsubscript{2}, \ldots, w\textsubscript{r}) are weights for Node\textsubscript{A} and (u\textsubscript{1}, u\textsubscript{2}, \ldots, u\textsubscript{r}) are weights for Node\textsubscript{B}.

4.2.2. Cosine Similarity

Cosine similarity is used to calculate the similarity between two nodes. Cosine similarity between two vectors...
(or two profiles) is a measure that calculates the cosine of the angle between two profiles and provides a measure to know similarity.

\[
\text{Similarity}(\text{Node}_A, \text{Node}_B) = \frac{\text{Node}_A \cdot \text{Node}_B}{\|\text{Node}_A\| \|\text{Node}_B\|} = \frac{\sum (w_i \cdot u_i)}{\sqrt{\sum w_i^2} \cdot \sqrt{\sum u_i^2}}
\]

If similarity is greater than a pre-defined threshold value then neighboring node is added to friend list or suggested as a friend. Cosine Similarity of 0 indicates that profiles are independent and 1 means the profiles are exactly same, and in-between values indicating intermediate similarity or dissimilarity. The threshold value that is defined to measure relationship can be provided by the user. For simplicity, we assumed 0.75 as default threshold value.

5. Example

Consider two Persons A and B browsing Internet on their individual interested topics. Assume that both Persons A and B searched some information related to sports, entertain, politics, and foreign. These are the words that were either used to search the information or extracted from website urls accessed by A and B. Let us also assume two GPS locations location X and location Y at time \(t_1\) and \(t_2\) respectively. Both persons searched for sports and entertain at Location X at time \(t_1\) and \(t_2\) respectively. Both persons searched for politics and foreign at Location Y at time \(t_2\). Assume number of times the terms sports and entertain appear are 2 and 4 for Person A and are 24 and 22 for person B at Location X. Similarly, assume number of times the terms politics, and foreign appear are 13 and 16 for Person A and are 12 and 12 for Person B at Location Y.

Therefore, we have following data:

Person A:

\(L_A = \{X, Y\}\)
\(L_A(t_1) = \{X\}\) and \(L_A(t_2) = \{Y\}\)
List_{A}(X) = \{K_1, K_2\} with K_1 = \text{sports} and K_2 = \text{entertain}
List_{A}(Y) = \{K_3, K_4\} with K_3 = \text{politics} and K_4 = \text{foreign}
F_A = \{\text{List}_{A}(X), \text{List}_{A}(Y)\}

\begin{align*}
I_{K_1} &= 2; \ I_{K_2} = 4; \ I_{K_3} = 13; \ I_{K_4} = 16
\end{align*}

Person B:
List_{B}(X) = \{K_1, K_2\} with K_1 = \text{sports} and K_2 = \text{entertain}
List_{B}(Y) = \{K_3, K_4\} with K_3 = \text{politics} and K_4 = \text{foreign}
F_B = \{\text{List}_{B}(X), \text{List}_{B}(Y)\}

\begin{align*}
I_{K_1} &= 24; \ I_{K_2} = 22; \ I_{K_3} = 12; \ I_{K_4} = 12
\end{align*}

Now, if we use Lee and Hong [1] algorithm then Global Profile for Person A will be \{\text{sports, entertain, politics, foreign}\} with weights \{2, 4, 13, 16\}. Similarly, global profile for Person B will be \{\text{sports, entertain, politics, foreign}\} with weights \{24, 22, 12, 12\}. Global profile is basically combination of all profiles based on different GPS location. In case two profiles have different keywords, profiles are merged.

Now,\[\text{Similarity}(A, B) = \frac{\sum (w_A \cdot w_B)}{\sqrt{\sum w_A^2 \cdot \sum w_B^2}}; \text{ where } w_A = \{2, 4, 13, 16\} \text{ and } w_B = \{24, 22, 12, 12\}\]

Therefore\[\text{Similarity}(A, B) = \frac{2 \times 24 + 4 \times 22 + 13 \times 12 + 16 \times 12}{\sqrt{2^2 + 4^2 + 13^2 + 16^2} \times \sqrt{24^2 + 22^2 + 12^2 + 12^2}} = \frac{484}{\sqrt{445 \times 1348}} = 0.62\]

Therefore, cosine similarity between Persons A and B is 0.62 which is less than predefined threshold that is 0.75. Thus, according to Lee and Hong [1] algorithm ad-hoc social network cannot be established between Persons A and B even though they have similar interests at the same location.

Now let us calculate cosine similarity according to the proposed algorithm.
Let us compute cosine similarity at Location X at time t_1 for their local profiles:
Person A:
List_{A}(X) = \{K_1, K_2\} with K_1 = \text{sports} and K_2 = \text{entertain}
\begin{align*}
I_{K_1} &= 2; \ I_{K_2} = 4
\end{align*}

Person B:
List_{B}(X) = \{K_1, K_2\} with K_1 = \text{sports} and K_2 = \text{entertain}
\begin{align*}
I_{K_1} &= 24; \ I_{K_2} = 22
\end{align*}

\[\text{Similarity}(A, B) = \frac{2.24 + 4.22}{\sqrt{2^2 + 4^2} \times \sqrt{24^2 + 22^2}} = \frac{136}{\sqrt{20 \times 1060}} = 0.93\]

Now, let us compute cosine similarity at Location Y at time t_2:
Person A:
List_{A}(Y) = \{K_3, K_4\} with K_3 = \text{politics} and K_4 = \text{foreign}
\begin{align*}
I_{K_3} &= 13; \ I_{K_4} = 16
\end{align*}

Person B:
List_{B}(Y) = \{K_3, K_4\} with K_3 = \text{politics} and K_4 = \text{foreign}
\begin{align*}
I_{K_3} &= 12; \ I_{K_4} = 12
\end{align*}

\[\text{Similarity}(A, B) = \frac{13 \times 12 + 16 \times 12}{\sqrt{13^2 + 16^2} \times \sqrt{12^2 + 12^2}} = \frac{348}{\sqrt{425 \times 288}} = 0.99\]
Therefore, we can see the cosine similarity when calculated for Location X and Y is higher than predefined threshold and ad-hoc social network will be created between Persons A and B at both locations. This explains benefits of having multiple local profiles for suggesting friends.

6. Conclusion

This paper presented a geo-social profile matching algorithm that constructed profiles dynamically based on user’s search and browsing history and showed that local profile based on GPS location of users were more meaningful and provided good friend suggestions.

References


Scientific Research Publishing (SCIRP) is one of the largest Open Access journal publishers. It is currently publishing more than 200 open access, online, peer-reviewed journals covering a wide range of academic disciplines. SCIRP serves the worldwide academic communities and contributes to the progress and application of science with its publication.

Other selected journals from SCIRP are listed as below. Submit your manuscript to us via either submit@scirp.org or Online Submission Portal.