Profile Attribute Matching in Mobile Social Networks

#**Arukala Radhika**[1], *M.Tech Computer Science& Engineering, E mail:* radhijka.cse0512@gmail.com
#**Mohd. Fasi Ahmed Parvez**[2], Assoc. prof. and HOD, *Department of CSE, E mail:* parvez40509@gmail.com
#**Vijaykumar Janga**[3], Assist. prof., *Department of CSE, E mail:* vijaykumar.janga@gmail.com
*# **Balaji Institute Of Engineering &Sciences, Warangal, Telangana, India***

Abstract:
Mobile social networks (MSNs) are precise types of social media which unite the ability of omnipresent connection for mobile devices to share user-centric data objects among interested users. In online social networks (OSNs), user connections can be represented as a graph. The network formed has distinct properties that distinguish it from other graph topologies. E.g., it has high average node degree, high value of clustering and displays small world properties. For accurate results, these properties need to be considered during analysis of OSNs. For instance, when a user searches for other users (or user characteristics), the network's search scheme may have to search and sort through a large set of results even at small topological distances. Additionally, users consider search results relevant based on their position in the network rather than globally relevant results. This paper makes two key contributions. First, we develop an information flow model to disseminate keyword information when users add keywords as their profile attributes. To address privacy concerns as outlined in works on future OSN architectures, we restrict the identity of users during information flow to only their direct friends. Beyond that, even though information about keywords is allowed to flow, the identity of users is not propagated. Second, we design and develop a search algorithm based on keyword information. The search problem is broadly defined as the scenario when a user queries with a set of keywords to contact other users (termed as targets) who have all those keywords as their profile attributes.

Index Terms: Mobile social networks, disseminate keyword information, online social networks.

1. Introduction

When knowledge about users and their keyword information is available to each individual node, aided by a centralized system (like Facebook, Orkut, etc.), then the search problem reduces to sorting through a list of targets who match the set of keywords in the query and construct the good result set. Here, we broaden the scope to consider future OSN architectures with a decentralized architecture where no node has access to complete information about all users. For these decentralized settings, searching for targets becomes a challenging problem. Further, the unstructured nature of the network increases the difficulty. Simplest would be to broadcast the network with queries till all possible targets are found, but this is inefficient and unscalable. Other decentralized search techniques based on either breadth first search or random walks are also not good candidates as they don't utilize the level of information available, in the form of user keywords and their policies, in a keyword based social network. The search algorithm we present utilizes the information available in the keyword based social network as it looks for targets. It uses a linear combination of two primary metrics: a distance metric to find topologically closer targets and, a trust metric to find paths with high trust values between the querying node and the target. The increase in the number of mobile devices has enabled users to be ubiquitously connected through wireless and mobile communications technologies. However, unlike conventional mobile ad hoc networks, persistent connectivity is not a necessity in every type of network. This has led to a totally progressive kind of social network called mobile social networks (MSNs). MSNs can be viewed as modern kinds of delay-tolerant networks (DTNs) in which mobile users interact with each other to share user-centric data objects among interested observers.

In real life, individuals become friends when they share common interests or passions. Sociologists have termed this tendency of human beings as `homophily'. Similarly, on online social networks (OSNs), like Facebook or Orkut, users establish friendships when they discover similar profile characteristics. The growth of LinkedIn [3], a social networking website, demonstrates the impact of profile information very well. Its purpose is to help people build professional networks and find career development opportunities. Using LinkedIn, employers can look into the profile information of users to search for potential employees.

Similarly, it helps employees look for potential employers. We feel that categorizing profile information and correlating it with network topology constitutes an important step towards the study of OSNs.

Social networks are a widely researched area. Thus, people implicitly made decisions based on their view of the geographical location or professional links of their friends and the associated likelihood of successful delivery of the letter. Lattice Model uses geographical distance, a user trait, to model social networks. Models based on interest and hierarchy has also been proposed to model the friendship behavior of people. In Davis Social Links (DSL), the social map is defined on the basis of keywords that are set by social peers as their profile attributes. Information transfer takes place only when a social path exists between the end users. Thus, it seems that keywords will play an important role in the development of future OSNs. A typical user profile on an OSN is characterized by its profile entries (key-words) like location, hometown, activities, interests, music, etc. It is important to understand the use of keywords and how they can be used effectively to classify content in OSNs. Consider the scenario, where a newcomer in the city, say Bob, would like to find people interested in soccer. As he doesn't know anyone yet, he tries his OSN profile to search for soccer enthusiasts in the city but uses the word `football' for the query. Though, both the words `soccer' and `football' can refer to the same sport, Bob's query returns no successful results because traditional residents use the word `soccer' for the game. The system fails to understand the underlying semantic relationship between the keyword entered by Bob and profile entries of other users. This shows the importance of extracting relationship(s) from the diverse information provided by users.

2. Related Work

2.1 How to Categorize Keywords?

A typical profile on any OSN consists of numerous sections (e.g. Orkut has Social, Professional and Personal sections; Facebook has Basic, Education & Work and Personal Information sections) that characterize the user. These sections are further sub-divided into various fields, e.g. the Personal section on Facebook has Interests, Activities, Favorite Movies, Books, etc. as fields. We call the entries in these fields Keyword(s) as they represent user attributes. To understand the keyword usage patterns we analyzed 1265 unique Facebook1 user profiles. Most of the fields contained proper nouns (e.g. movie names, albums, etc.) as entries, hence, for all evaluation purposes, we restricted ourselves to keywords found in the Interests (which contained words mostly from an English dictionary) field. We looked at the magnitude of information given by users and how keywords can be processed to extract meaningful knowledge. On average, each user provided keywords for the Interests field and the keyword set contained 1573 unique keywords. To analyze the distribution of keywords, we plotted the number of distinct keywords for a given keyword frequency on a log-log scale. We divided the keyword frequency into four categories to represent keywords with different frequencies.

We looked at the magnitude of information given by users and how keywords can be processed to extract meaningful knowledge. On average, each user provided keywords for the Interests field and the keyword set contained 1573 unique keywords. To analyze the distribution of keywords, we plotted the number of distinct keywords for a given keyword frequency on a log-log scale. We divided the keyword frequency into four categories to represent keywords with different frequencies. The trend line (solid continuous line) for the graph. The distribution shows consistency with similar results on tag distribution over web applications. It follows the Zipf's Law because the occurrence frequency of a keyword increases as its popularity increases in the frequency list. Thus, we can infer that most of the keywords entered by users are distinct. This means that only a fraction of keywords are repeatedly used by different users and a large percentage of keywords (44% of the total) occur with very low frequency.

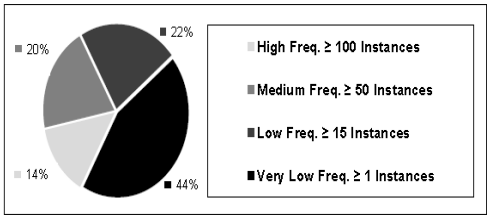

Figure 2.1: Keyword Frequency Distribution

Two important conclusions can be drawn using the above discussion. First, as the number of unique keywords is large there may be some relationship between these different keywords. This observation uses the fact that different topics in which users are interested can be generalized to a small number as there are limited `community' categories in OSNs. The large

number of unique keywords must also be related to these limited categories. Second, it is possible that the very low frequency keywords (which constitute almost half of the total) aren't very dissimilar either with each other or to the other 56% of keywords due to the extensive usage scope of English words. Thus, to develop a social network model based on keywords, there is a need to explore the hidden relations among keywords and to categorize them. For instance, in Bob's case, if the OSN could understand the relationship between `soccer' and `football' it might give better results for Bob's query.

2.2 Social Network Architecture

We want a data structure that can help define distance between keywords by capturing hidden relations between them. It must employ methods to clearly distinguish between related and unrelated keywords. A single hierarchical structure will be insufficient as it will fail to capture important characteristics of keywords. First, it is not always possible to relate all the words, e.g. `earthquake' and `soccer', in a single structure. The distance between such unrelated words must come out to be relatively larger than that between related words. Second, the data structure must capture all meanings of a word as it can be used in different contexts (or in different syntactic categories). E.g., according to Word Web, the word `stern' could mean `severe' as an adjective and `rear part of a ship' as a noun. We propose a forest structure to store keywords where each tree in the forest contains related keywords. As a keyword could have more than one meaning it could occur in different trees. This way, we use multiple hierarchical trees (i.e. a forest) to measure distance between keywords. Now, we discuss the methods used to evaluate the above social network model. We considered two networks and compared the similarity values to observe the effectiveness of the `forest' structure in correlating profile keywords with network topology. One network represented a realistic scenario while the other was generated through simulation of our social network model.

Here, we focus on unstructured keyword based social networks where the user topology is represented by a set of nodes and edges. Let the social network be an undirected graph G = (V;E), where V represents the set of nodes in the social network and E is the set of edges between the vertices in V . A link exists between nodes i and j if both the nodes want to be friends with each other. Each edge has a trust value associated with it to represent the mutual relationship between the corresponding nodes. The concept of trust can be generalized to include different models that characterize different friendship levels, such as the frequency or quality of interactions, privacy settings during information sharing, etc. The value of trust lies between 0.0 and 1.0 with a higher value representing more trust.

2.3 Information Flow Model

Once a user joins the network and adds keywords as his Profile Attribute(s), information needs to be propagated in the network so that other users can search and contact him. The information flow model primarily needs to satisfy three conditions while spreading the profile attributes. First, it must propagate the information properly using minimal resources. Second, it must address the privacy concerns of users. Finally, it must ensure that nodes maintain the most recent information i.e. information about changes in friendship(s) or keyword policies must be propagated to nodes quickly.

To respect privacy requirements, any social network model and corresponding applications must be designed so that user privacy is protected as information flows within the graph. Our design of message propagation removes the identity of users as their information flows deeper in the graph, i.e. at distances further away from their direct friends. The keywords, propagation data and PID's are stored but they are insufficient to reveal any relevant information about the people who are not direct friends of the user. As the identity of only the direct friends is stored, the network has the capability of supporting both anonymous and identified messages (depending on the application) without compromising the privacy of users making the message propagation model more general and suitable for wider use.

As friendships change, the topological structure of the graph changes. Policies of individual keywords may also be changed by users. Such updates should also be reacted in the information that is stored by users. Thus, the information flow process must notify nodes about such updates so that each user has access to the latest information. To account for such scenarios, we introduce the concept of timely updates for propagated keywords that have owed into the network. Each node will send beacons (which could be the (keyword, PID) pair) after a certain time to tell other nodes that the corresponding keyword still exists in the network.

3. Querying with Keywords

Now, we describe the querying process that a user uses to search for targets. Nodes consider two important factors as they route query messages through the network: a) Value of Friends b) Threshold Function. The first parameter helps a node to determine the value of direct friends while the second parameter dynamically sets threshold values so as to reduce the number of edges the algorithm needs to inspect.

The evaluation methodology consists of four steps: generation of graphs with properties of social networks (high average values of node degree and clustering) as well as small world properties (low diameter), distribution of trust among the edges, assignment of keywords to user nodes with corresponding policies and their propagation and finally, issuing queries from a set of nodes to see how the algorithm performs. We assumed that the time taken by the graph topology to change or for a user to update keywords and/or its policies is much greater than the time taken to search for a query. Thus, for simulation we used a static graph environment.

Information Propagation: Keywords:

We set our information propagation process into two environments by varying the depth in the policy to see how the search algorithm performs in each of the environments.

Restrictive Policy: In this case, all users uniformly set the policy associated with each keyword. When a keyword is added by a user, the maximum depth value (D) is set to 2, i.e. a keyword can travel from its originator to a maximum of two hops. The choice of two hops for restrictive policy is significant as it represents the `friend of a friend' radius. This policy represents the case when users are very restrictive about their information being propagated in the network.

Liberal Policy: When users are allowed to set any policy for keywords. Here, when a keyword is added by a user, a random integer between zero and diameter of the graph is set as the depth value in the policy. This helps us to observe the performance of the algorithm in a more realistic situation where some people are private in nature and concerned about who gets to access any information about them (and thus, set the depth value to 1 where only direct friends can access the information) whereas other people let their information is accessible from many hops away in the network.

We next assigned to each keyword the weighted average of the trust values (that came out to be 0.4828) of the edges, as its policy, so that it uniformly propagates throughout the network. We randomly selected 100 nodes from the graph and initiated the process of keyword addition to these nodes and then propagated those keywords using Algorithm. Since, the search algorithm dynamically sets the value of the threshold by looking at the node degree and by applying the pruning function, query nodes must be selected so they are representative of the various node degrees in the network. Thus, we sorted the nodes according to their node degree and picked 100 nodes with their degrees ranging from the minimum node degree to the maximum node degree of the generated graph. For each of the query nodes, we started the search algorithm and analyzed its behavior.

4. Conclusion

This chapter modeled the flow of information in keyword based social networks. We developed a search algorithm for the given information flow settings that showed improvement in orders of magnitude when compared to BFS. The algorithm concentrated on finding a subset of results that have good characteristics. We do so with special focus on decentralization and privacy as proposed in future social net work architectures. We believe that the algorithms presented can be adapted for network applications that may show graph properties similar to an unstructured social network. As future work, we would like to explore this work in multiple directions. First, we want to evaluate the search strategy for real social network graphs as we presented a limited evaluation on a synthetic graph here. Second, we are interested in further modeling the threshold function by incorporating the distribution of `value' of friends between the minimum and maximum values. Third, we would like to extend the definition of trust to bidirectional trust for an edge and model accordingly. Finally, a broader expansion of the search algorithm using semantics of query keywords will form the next stages.

References

[1] J. R. Douceur, "The Sybil attack," in *Proc. 1st IPTPS*, 2002, pp. 251–260.

[2] N. Kayastha, D. Niyato, P.Wang, and E. Hossain, "Applications, architectures, and protocol design issues for mobile social networks: A survey," *Proc. IEEE*, vol. 99, no. 12, pp. 2130–2158, Dec. 2011.
[3] K. Fall, "A delay-tolerant network architecture for challenged Internets," in *Proc. SIGCOMM*, 2003, pp. 27–34.
[4] M. Khabbaz, C. M. Assi, and W. Fawaz, "Disruption-tolerant networking: A comprehensive survey on recent developments and persisting challenges," *IEEE Commun. Surveys Tuts.*, vol. 14, no. 2, pp. 607–640, 2nd Qtr. 2012.
[5] A. Kapadia, D. Kotz, and N. Triandopoulos, "Opportunistic sensing: Security challenges for the new paradigm," in *Proc. COMSNETS*, 2009, pp. 127–136.
[6] A. Shikfa, "Security challenges in opportunistic communication," in *Proc. IEEE Conf. Exhib. GCC*, 2011, pp. 425–428.
[7] A. Beach, M. Gartrell, and R. Han, "Solutions to security and privacy issues in mobile social networking," in *Proc. Int. Conf. CSE*, 2009, pp. 1036–1042.

Arukala Radhika

M.Tech in Computer Science Engineering from JNTU Hyderabad.His research areas includes Programming Languages, Data Base Management Systems, Mobile Applications, Data Mining.

Vijaykumar Janga :obtained Master's in Computers with the specialization SoftwareEngineering and Pursuing PhD from JNTU Hyderabad. Research interest includes Bigdata, Stream Outlier Mining, Information Retrieval systems. Currently Working as Assistant Professor in Computer Science and Engineering with Balaji Institute of Technology and Sciences Narsampet.

Fasi Ahmed Parvez : working as Associate professor and HOD BALAJI INSTITUTE OF ENGINEERING SCIENCES-NARSAMPET, with 12+ years of Experience. Completed M.Tech from JNTU Hyderabad in2010. SUBJECT INTERESTED are programming languages, database management system and data warehouse & data mining.