Soc. Netw. Anal. Min. (2011) 1:143–158
DOI 10.1007/s13278-010-0006-4

ORIGINAL ARTICLE

Analysis of user keyword similarity in online social networks

Prantik Bhattacharyya · Ankush Garg · Shyhtsun Felix Wu

Received: 13 March 2010 / Accepted: 3 June 2010 / Published online: 6 October 2010

Abstract How do two people become friends? What role does homophily play in bringing two people closer to help them forge friendship? Is the similarity between two friends different from the similarity between any two people? How does the similarity between a friend of a friend compare to similarity between direct friends? In this work, our goal is to answer these questions. We study the relationship between semantic similarity of user profile entries and the social network topology. A user profile in an on-line social network is characterized by its profile entries. The entries are termed as user keywords. We develop a model to relate keywords based on their semantic relationship and define similarity functions to quantify the similarity between a pair of users. First, we present a 'forest model' to categorize keywords across multiple categorization trees and define the notion of distance between keywords. Second, we use the keyword distance to define similarity functions between a pair of users. Third, we analyze a set of Facebook data according to the model to determine the effect of homophily in on-line social networks. Based on our evaluations, we conclude that direct friends are more similar than any other user pair. However, the more striking observation is that except for direct friends, similarities between users are approximately equal, irrespective of the topological distance between them.

P. Bhattacharyya (✉) · A. Garg · S. F. Wu
Department of Computer Science, University of California, Davis, One Shields Avenue, Davis, CA 95616, USA
e-mail: pbhattacharyya@ucdavis.edu

A. Garg
e-mail: garg@ucdavis.edu

S. F. Wu
e-mail: sfwu@ucdavis.edu

Keywords Online social network · User keywords · User similarity · Homophily measurement · Semantic analysis

1 Introduction

The famous experiment conducted by Travers and Milgram on the small world problem (Travers et al. 1969; Milgram 1967) tried to ascertain if people in society are linked by small chains. They asked people to forward letters to their friends who they thought were likely to know the target person. Thus, people implicitly made decisions based on their view of the geographical location and professional associations of their friends and the associated likelihood of a successful delivery of the letter through that friend. The results showed that people are able to find other individuals at even far off places fairly quickly and the path length connecting such a pair of individuals is small. These very interesting conclusions opened up the question about how individuals are connected amongst each other, in spite of living at far-off geographic locations. In other words, what brings a set of individuals together, even when they do not belong in the same geographic location? What role does homophily play here? Do people become friends when they share common interests and passions despite of living at different places?

On-line social networks (OSNs) help us study such problems using the set of rich data present about the users. A typical user profile in an on-line social network is characterized by its profile entries like location, hometown, activities, interests, favorite music, professional associations, etc. For example, in sites like Facebook[1] and Orkut,[2]

[1] Facebook is avilable at http://www.facebook.com.

[2] Orkut is available at http://www.orkut.com.

users establish friendships when they discover similar profile entries. In LinkedIn[3] people connect amongst each other to build professional networks and find career development opportunities. Using LinkedIn, employers look into the profile information of users to search for potential employees. Similarly, it helps employees find potential employers. Thus, when two people share a common professional field, they come closer, connect to each other and establish friendship.

In this work, our goal is to (1) understand the process of how people connect to each other, i.e., form friendships based on the intersections of their interests and passions, (2) study the similarity across different user profiles and (c) correlate user similarity with the network topology to understand the effect of homophily in on-line social networks.

Consider the scenario, where a newcomer in the city, say Bob, a soccer enthusiast friends other soccer enthusiasts. On his OSN profile, he enters 'football' as his entry in his interests field while his friends enter 'soccer'. When we try to analyze the similarity between Bob and his friends through a similarity analysis of their interests, we do not see any similarity based on a direct matching of the entries. But essentially, the friendship between Bob and his friends evolve because all of them are interested in a sporting activity and their interests match. In other words, homophily plays an important role in bridging friendship between Bob and his friends.

To understand the influence of homophily using the underlying semantic relationship of profile entries and to successfully extract relationship(s) from the diverse information present, we build models in this work. We term each of the individual profile entries of an user as *Keyword*. Our key contributions are summarized next. In this paper, we study the relationship between semantic similarity of user keywords and the social network topology. First, we define a model to categorize keywords based on the semantic relationship. The model consists of multiple categorization trees to aggregate similar keywords. We formally term the model of multiple categorization trees as the 'Forest Model'. Second, we define the notion of distance between keywords in the 'forest' and based on the keyword distance, we define functions to determine the similarity between a pair of users. Third, we analyze a set of Facebook data according to the model to determine the effect of homophily in on-line social networks.

Based on our evaluations, we conclude that direct friends are more similar than any other user pair. The most striking observation is that except for direct friends, similarity between users are approximately equal, irrespective of the topological distance between them. The similarity between users who are separated by two hops is nearly equal to the similarity between users placed at three, four or more hops away in the on-line social network. We also observe the effect of different ways in building the 'forest' in determining similarity between the users. Our analysis also shows that an increase in the number of friends and keywords for an individual user lowers the average similarity between the user and his friends.

In Sect. 2 we survey related work. We discuss the key challenges and present our findings on keyword usage patterns in Sect. 3. Next, we introduce the 'Forest Model' to categorize keywords and discuss its impact on analyzing user keywords in Sect. 4. We propose functions to quantify similarity between users in Sect. 5 and evaluate them in Sect. 6. We conclude in Sect. 7 with a discussion of future works.

2 Background

In this section, we review some of the related work. First, we discuss work related to the mathematics behind the small world problem and social networks in general. Next, we discuss works that address homophily in social networks and user similarity based on user characteristics.

Works in Kleinberg (2001), Sandberg (2007), and Kleinberg (2000) have developed mathematical models to show how users interact with one another and establish links to build a social network. The lattice model (Kleinberg 2000) is based on the geographical distance between users. The model defines a network model based on characteristics of user's to establish multiple short range friendships and few long range contacts. Based on the definitions, decentralized algorithms are developed to show that users can search for short paths to other users with high probability. The work in Sandberg (2007) also presents mathematical models to further the decentralized search algorithm to enable searches even when users are unaware of their own and other's positions in the network. In Kleinberg (2001), a hierarchical network model was developed. Users are arranged at the leaves of a hierarchical structure such that the least common ancestor of two nodes in the tree is the node at which they start differing in their attributes. Thus, the least common ancestor defines the similarity of two nodes or how likely they are to become friends. The closer the least common ancestor is to the two nodes, higher probability of the two nodes being friends. Based on this probability, the social network graph and the decentralized search algorithm are developed.

Homophily, or the more commonly known phrase of 'birds of a feather flock together' has constituted an important role in the study of social networks. Sociologists (Mcpherson et al. 2001) have tried to understand the

[3] Linkedin is available at http://www.linkedin.com/.

phenomenon using multiple characteristics like gender, race, ethnicity, age, educational level, etc. Similarity between users due to their association to same communities has been studied in Crandall et al. (2008). Community associations and user keywords have been used to model user communication in social networks in Banks et al. (2007, 2009). Information exchange between users takes place only when they share a social path and common keywords and community memberships. Decentralized search algorithms using combinations of homophily and node degree parameters have been developed in (Şimşek and Jensen 2005, 2008).

Similarity between users as a function of their topological distance was studied in Adamic et al. (2003). The work tried to find out the average fraction of similar users with a common characteristic like year in school, graduate status, etc. to track the number of similar users from a data set of Club Nexus. Their findings reported a gradual decay in similarity with increased topological distance in the social network. The work in Adamic and Adar (2001) developed functions to analyze similarity between users as a function of the frequency of a shared item.

Geographic ties between on-line social network users has been another property to understand homophily between users. Geographic location and friendship behaviors of bloggers was studied in Kumar et al. (2004). The work in Liben-Nowell et al. (2005) has also studied the relationship between geographic location of users and the relationships among them. The study showed that one-third of friendships in a social network are independent of geography. This is an interesting conclusion and raises the question of why people at far off locations become friends and what characteristics bring them together? Will understanding the other key interests or activities of users in on-line social networks explain why people become friends?

In this work we answer these questions by understanding the *interests* pattern of users in Facebook and how similarity between user interests influence friendship. We study the influence of user similarity in the network topology. We use the term network and social network interchangeably in our work to mean the set of all users and the links between them that represent the friendship between them. We also explain the patterns of characteristics associated with a user, i.e., a user's profile entries.

We classify the similarity between users through the semantic links between the keywords used by them. Methods like Latent Semantic Indexing (Deerwester et al. 1990) have previously explored the semantics between digital data to explore the relationship between them. Analysis of user similarity through relations between their profile characteristics can also help in furthering the link prediction problem (Liben-Nowell and Kleinberg 2007) in social networks to correctly identify pairs who are likely to forge friendship in future. In the next section, we discuss the keyword usage patterns of Facebook users.

3 Keyword usage patterns

To measure the similarity between keywords and understand the usage scope of keywords as entered by different users in their on-line social network profiles, we analyzed Facebook profiles. We considered keywords that are available in the English dictionary. For this purpose, we used the entries present in the *Interests* fields of a Facebook profile. Users list the activities they are passionate about or topics of which they are interested in this field. For example, an analysis of the data shows that a large portion of users list *Music* as their *Interests*.

The standard entry method in the *Interests* fields of a Facebook profile is to input the keywords as a list of comma separated values. While some of these entries can consist of multiple words combined together, e.g., *computer science*, a lot of these entries are also made up by combining different forms of an English word, e.g., *listening to music*. Analyzing such keywords requires an understanding of word sense disambiguation (Spear et al. 2009). Instead, we extract information from single word entries. The data set we analyzed contains a set of 1,265 Facebook profiles of which, 765 profiles have one or more keywords in their profile entry. The keyword set obtained from these 765 profiles contains 1,301 unique keywords and the entire set consists of 4,787 keywords, for an average of approximately 6 keywords per user profile. This dataset is a subset of the data used in our preliminary work presented in (Bhattacharyya et al. 2009) and was updated to reduce noise levels in the data. Details about the data and

Table 1 Top ten keywords in keyword set along with respective occurrence frequency and percentage values

Rank	Keyword	Occurrence frequency	Occurrence percentage
1	Music	173	3.61
2	Movies	122	2.55
3	Sports	95	1.98
4	Reading	95	1.98
5	Traveling	87	1.82
6	Basketball	79	1.65
7	Soccer	77	1.61
8	Tennis	68	1.42
9	Football	66	1.38
10	Running	63	1.32

the data collection processes can be found in Spear et al. (2009). In Table 1, we listed the top ten keywords along with their frequency count. To analyze the distribution of keywords, we plotted the number of keywords for a given keyword frequency on a $\log_{10}$ scale in Fig. 1. We divided the keyword frequency in four categories to represent keywords with different frequencies and plot the number of keywords in each category in Fig. 2. Conclusions from the analysis are discussed in the rest of this section.

Table 1 contains the top ten most frequently used keywords. The top ten keywords collectively make up approximately 19% of the entire keyword set. This shows that for an average of 6 keywords per user profile, a user has a high chance of having any of these top ranked keywords. This result also opens the question on how the rest of keywords are distributed among the profiles. To inspect the frequency of keywords in the set, we plot the relationship between the number of keywords found for a given keyword frequency in Fig. 1. The plots show that there are 866 keywords (approximately 66.56% of the number of keywords) that occur only once, i.e., they occur with a frequency of only one. Based on keyword usage frequencies, we see a randomly picked user profile has a high chance of listing a keyword that none of the other users in the dataset have in their profile. These two results

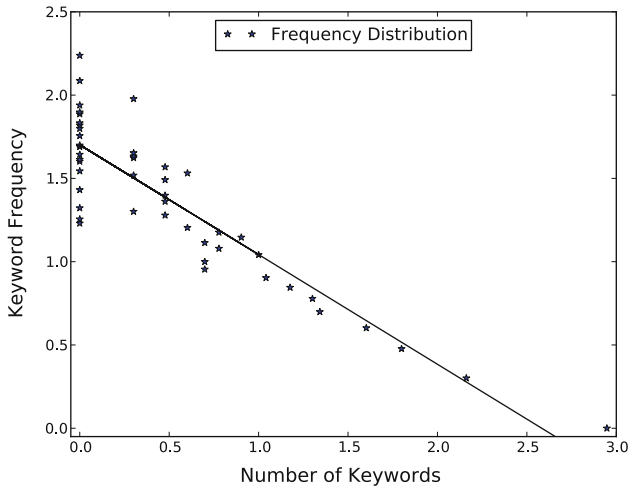

Fig. 1 Plot showing the number of keywords available for a frequency. The values have been plotted on a $\log_{10}$ scale

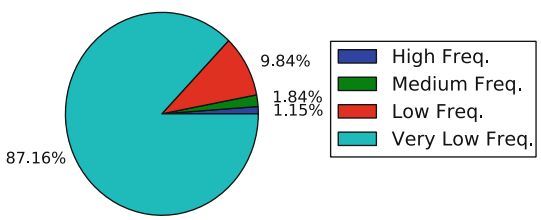

Fig. 2 Number of keywords distribution per frequency category

show the wide distribution in usage of keywords by users in their profiles.

The trend line (solid continuous line) in Fig. 1 shows an exponential drop in the number of distinct keywords as the keyword frequency increases. The distribution follows a power law as the number of distinct keywords decreases as the frequency of the keyword increases. The distribution also shows consistency with similar results on tag distribution over web applications (Xu et al. 2006).

We further substantiate this result by aggregating keyword frequencies into four categories based on the values. Keywords that occur more than 45 times are put into 'High Freq.' category. Keywords occurring more than 25 times are put into 'Medium Freq.' and more than 5 times are put into 'Low Freq.' categories. The rest of the keywords, i.e., those occurring less than 5 times are put into the 'Very Low Freq.'. These results are plotted in Fig. 2.

We observe here that approximately only 1.15% of the keywords belong to the high frequency category, while more than 87% of the entire dataset comes under the very low frequency category. With such wide distribution of keywords across user profiles, analyzing the similarity between two user profiles based on matching keywords leads to inconsistent and inconclusive results. The key questions now are, how can we aggregate different keywords based on their usage patterns to understand similarity between users? Can models be developed to match keyword pairs when they have semantic relationships? How can we explore the hidden relations and categorize them? For instance, from the previous example, if we can build models to understand the relationship between 'soccer' and 'football', we can analyze more deterministically the influence of homophily between Bob and his friends. In the next section, we introduce the 'Forest Model' to categorize and aggregate keywords effectively to understand the similarity between on-line social network users.

4 Forest model

In this section, we first describe the 'forest model' to categorize keywords. The model helps to define the data structure to utilize the underlying relationship amongst keywords. Second, we describe 'forest generation' process. Here, we also describe the heuristics we define to analyze the similarity between users in later sections. In the third subsection, we analyze the entire keyword set and present results of our evaluation.

4.1 Forest model

How do we relate two keywords? How do we keep two keywords separated when they can not be related? Our goal

here is to find a model that can clearly distinguish between related and unrelated keywords. We aim for a simple and intuitive model that helps us achieve this.

What is the underlying hidden relationship between any pair of keywords? Intuitively, keywords can share the same source of origin. This characteristic of keywords relating to each other is based on their source of origin and development. Linguists term this characteristic as etymology. For instance, in a language like English, words have a Latin or Greek root associated with them. Wordinfo[4] lists 61,362 English words that have either Latin or Greek roots. For example, the words 'equine' (horse), 'equestrian' (horse rider), 'equestrienne' (female horse rider) can be derived from the Latin root *'equus'*. *'Equus'* meaning a horse.

Alternatively, keywords can said to be related when they are semantically linked, e.g., when they share the same meaning. For example, keywords like 'football' and 'soccer' are related because they both are a type of sport. These keywords can also be related to the keyword 'sports' because of the relationships between their meanings. Thus, continuing Bob's example, now when we look at Bob's interest in 'football' and his new friend's interest in 'soccer', we can say how the two profiles match to each other from the fact that 'soccer' is a hyponym of 'football'. Thus, aided by relationships between keywords, we can match user profiles and analyze similarity between users, effects of homophily, and why friendship links are established.

Once we establish a relation between two keywords, the key requirements of a model is that it must keep unrelated keywords separated. This means, that while 'football' and 'soccer' are related through the model, keywords like 'soccer' and 'equine' from the previous examples are kept separated.

Next, we describe the model. Each keyword is considered as a node. Nodes are connected when relations exist between the keywords. These nodes are placed in a hierarchical order such that when a keyword is derived from another keyword, hierarchy helps in defining the relation between the keywords. The hierarchy thus gives the ability to detect distances and dissimilarity between keywords and prevents homogeneity between the nodes that can arise from the use of a flat data structure. Hierarchies, thus constructed to define the relationship between keywords leads to the definition of 'Trees'. To keep unrelated keywords separate from each other, multiple trees are defined. Such trees each contain set of keywords that are related to each other in the tree but are unrelated to any other keyword in any of the other trees. Formally, let a forest F be declared as a data structure consisting of t trees, $(T_1, T_2,...,T_t)$.

[4] Wordinfo is available at http://www.wordinfo.info and is copyrighted by Senior Scribe Publications.

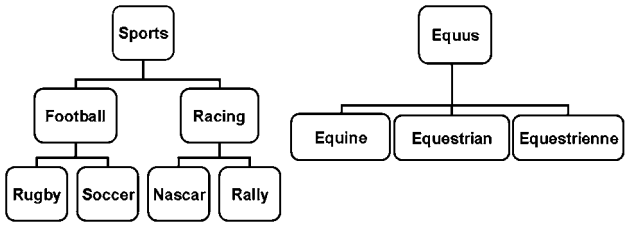

Fig. 3 Example 'Forest' with two component trees

An example 'forest' can be seen in Fig. 3. Two trees are built from the keywords discussed earlier. The root of the first tree is kept as 'sports' from where the other keywords like 'football' and 'racing' can be derived. Further in the subtree, 'rugby' and 'soccer' are placed as both are related to 'football'. Similarly, in the second tree, 'equine', 'equestrian' and 'equestrienne' are placed in the sub-tree, all placed directly under the root word 'equus'. Next, we describe how using the semantic relationship between the keywords, we generate the 'forest model'.

4.2 Forest generation

We used the underlying semantic relationship between keywords to built the 'forest'. We used WordNet (Fellbaum 1998) as the database of English words to build the forest structure. We will describe the features of WordNet next and then we will describe the heuristics we used during our evaluation process in Sect. 6.

WordNet is a large lexical definition of English words.[5] WordNet relates different words using their sets of cognitive synsets. The synsets further are linked together by means of conceptual-semantic and lexical relations. We used a Java API (Howe 2009) to look at the WordNet ontology. Definitions for some of the WordNet ontologies are taken from the information available at (Howe 2009). Based on the access to WordNet ontology, we define the following four heuristics to retrieve keywords related to a given keyword.

1. Base: Here, we let the tree be composed of only the initial keyword. Thus, the tree is not allowed to grow its sub-tree. This heuristic thus constitutes the boundary condition where keywords match only if they are exactly similar to each other.
2. HM: In this heuristic, we grow the tree using the keyword's holonyms and meronyms. Consequently, we term this heuristic as 'HM'. The ontology 'Holonyms' is referred to mean a word that names the whole of which a given word is a part of. For example,'hat' is a holonym for 'brim'. The term 'Meronyms' is used to refer to a part/whole relationship. For example, paper is a meronym of book, since

[5] WordNet is available at http://wordnet.princeton.edu/.

paper is a part of a book. We also use the 'nominalizations' ontology of the WordNet to obtain the set of nominalized terms for all senses of the keyword, i.e., referring to the use of a verb or an adjective as a noun. For example, WordNet ontology returns the set of keyword 'happiness, felicity' for the keyword 'happy' and the set 'happy' for the keyword 'happiness'. Thus, depth of the tree is 2. The root word is the keyword itself and the rest of the terms returned by WordNet ontologies form the sub-tree.

3. SS: In this heuristic, we grow the tree using the keyword's 'similars' and 'synonyms' ontology available from WordNet and thus we term the heuristic as 'SS'. The keywords available from WordNet form the subtree and the depth of the tree thus formed is also 2. The WordNet ontology 'Similars' returns a similar-to list for the given keyword, e.g., it returns the set 'blessed, blissful, bright, golden, halcyon, prosperous' for the keyword 'happy'. These related keywords are obtained only for keywords that are adjectives. In the 'Synonyms' ontology, words that have similar meanings are obtained, e.g., 'glad' for the keyword 'happy'.
4. All: In this heuristic, we use 'all' the ontologies present in WordNet to obtain a list of all the related keywords available for a given keyword. The tree depth is 2 and the subtree is formed by keywords that are available using 'Nominlizations', 'Holonyms', 'Meronyms', 'Synonyms', 'Antonyms', 'Similars', 'Hypernyms', 'Hyponyms' and 'Derived Terms' ontologies. Hypnernymy refers to a hierarchical relationship between words. For example, furniture is a hypernym of chair since every chair is a piece of furniture (but not vice-versa). Hypnonymy is the opposite of hypernymy. Dog is a hyponym of canine since every dog is a canine. The derived-terms holds for adverbs and returns derived terms for all senses of a keyword. For example, the set of keywords 'jubilant, blithe, gay, mirthful, merry, happy' is returned for the keyword 'happily'. Thus, this heuristic makes for a boundary case where all related keywords are used to build the tree for evaluation purposes.

The motivation to use multiple heuristics as defined above comes from the observation that keywords can have more than one meaning or context, e.g., according to WordWeb,[6] the word 'stern' could mean 'severe' as an adjective and 'rear part of a ship' as a noun. Generating the forest with different heuristics helps us capture different scenarios where a keyword may be present in different trees due to varied usage and contextual scopes. Thus, the above mentioned heuristics not only capture different meanings of a keyword but also helps capture the similarity between keywords when they are used in different contexts or belong to different syntactic categories.

To build a 'forest', we adopted a more ad hoc approach, allowing each keyword of a keyword pair to build its own tree. For each of the above heuristics, related keywords were pulled from WordNet and aggregated together to form the individual tree. This process was recursively repeated to the desired tree *depth*. The initial keyword was placed as the root of the tree. For every keyword pair, thus two trees were formed. These two trees were checked for any common keyword. If a matching was found, keyword pairs were declared as related to each other. Otherwise, the keyword pair were termed as not similar to each other. In the next section, we analyze the effectiveness of the 'forest model' in matching keywords.

4.3 Analyzing the user keyword set

In this section, we will analyze the effectiveness of the 'forest model' in computing the similarity between user keywords. We will use a set of examples to demonstrate the advantages of the 'forest model'.

Let us look at four Facebook users with their keywords, for the *Interests* field (Table 2). All the users are interested in some type of sporting events. We compare the results of matching the keywords of the first three users with the user *Z* in Table 3. For simplicity, we demonstrate values only for the two cases, 'Base' and 'All'. For the other two heuristics, the number of matches increase similarly depending on how related keywords are available in the WordNet ontology.

It can be seen that for the 'Base' case, most of the trials to match keywords of both the users fail. Only since *B* and *Z* have 2 keywords in common across their profile that the similarity between *B* and *Z* come out to be a non-zero value. Now, when we look at the similarity between *B* and *Z* for the 'All' heuristic, we see that similarity values have risen to 25 and the fraction of keyword pairs now similar to each other stands at 62.5%. This is because their profiles match for keywords that can be derived from 'athletic sports' (e.g., pairs formed from running, soccer, tennis, etc.).

Table 2 Sample users with keywords

User	Interests
A	Wakeboarding, softball, fishing, jesus, god, learning, backpacking
B	Running, hiking, hurricanes, tornadoes
C	Basketball, dancing, shopping, pictures
Z	Running, soccer, tennis, football, hiking, knitting, art, tea, lime, pie

[6] WordWeb Software available at http://www.wordwebonline.com.

Table 3 Number of matches to keywords of user Z

User	Number of pairs	Base		All	
		Number of matches	Fractions match (%)	Number of matches	Fractions match (%)
A	70	0	0.0	27	38.57
B	40	2	5.0	25	62.50
C	40	0	0.0	29	72.50

It can also be seen that $s(Z, C)$ is maximum even though Z and C do not have more keyword pairs than between Z and A or between Z and B. This is because both are interested in arts (C has 'pictures' and Z has 'art') implying that Z has more common interests with C than with A or B. A and Z are least similar as A is mostly interested in water sports (and not athletic sports as Z) and does not share any other common interest with Z even though they both share a large number of keyword pairs. This shows the effectiveness of characterizing keywords using semantic relationships and that the content of keywords becomes more important than their number for finding similarity values. We also observe the effectiveness of the 'forest model', built using the semantic relations of keywords, in measuring the similarity of users keywords. Next, we describe similarity functions based upon the 'forest model' to measure the similarity between user profiles and understand the effect of homophily in social networks.

5 User similarity

With keywords present at different hierarchies in a tree, how do we measure the similarity between keywords and correspondingly the similarity between users? How do we differentiate the similarity between two users when all their keyword pairs belong to the same tree but the keywords are positioned at various different heights? In this section, we describe the formulations to answer some of these questions. First, we quantify the distance between two keywords in the 'forest'. Afterwards we describe two different similarity functions to quantify the similarity between users.

5.1 Keyword distance

Now we define the notion of distance between keywords based on the forest structure. Let there be t trees ($T_1, T_2,...,T_t$) in the forest F. Consider two keywords K_a and K_b such that both of them belong to the same tree. Let LCA be the least common ancestor of K_a and K_b. Also, assume $d(\text{LCA}, K_a)$ to be the depth of K_a from the LCA.

Definition 1 If K_1 and K_2 are two keywords, then the distance, $D(K_1, K_2)$, between them is given as:

$$D(K_1, K_2) = \begin{cases} d_{\text{LCA}}(K_1, K_2) & \text{if } K_1, K_2 \in T_i \\ \infty & \text{if no such } T_i \text{ exists} \end{cases}$$

where $d_{\text{LCA}}(K_1, K_2) = \max(d(\text{LCA}, K_1), d(\text{LCA}, K_2))$. If more than one such T_i exists, then the distance is set to the minimum of all the corresponding d_{LCA}'s.

If K_1 and K_2 do not have any relation then $D(K_1, K_2)$ is ∞. Also, the minimum of all d_{LCA}'s is used to account for multiple occurrences of keywords in F.

Thus, from Fig. 3, if $K_a = \textit{soccer}$ and $K_b = \textit{racing}$ then LCA = *sports* and $d(\text{LCA}, K_a) = 2$, $d(\text{LCA}, K_b) = 1$ and $D(K_a, K_b) = 2$. When $K_a = \textit{soccer}$ and $K_b = \textit{equine}$ then, as each of the keywords are present in different trees, no LCA exists and $D(K_a, K_b) = \infty$.

The separation of keywords into different trees and defining the distance between keywords as ∞ when they do not belong together in a tree makes the model robust enough to handle the aggregation of keywords and yet clearly separate keywords when they do not belong together. The hierarchy inside the trees helps determine the distance when the keywords belong to a single tree. This is an advantage over possible models where all keywords are put together in a single hierarchy, for example by generalizing the model of hierarchy presented in (Kleinberg 2001) to relate keywords.

It is also important to note that in the definition of $D(K_1, K_2)$ when keywords are aggregated together, the distance between keywords are captured from the generic point where an aggregation is possible. For example, in Figure 3, *soccer* and *racing* aggregate at *sports* and thus the distance between the keywords is defined as the farthest distance from this generic point. An alternate definition where distance between keywords is the summation of the distances of each keyword from the generic point (i.e., $D(K_1, K_2) = d(\text{LCA}, K_1) + d(\text{LCA}, K_2)$) fails to comprehend the importance of the distance from the LCA itself. Based on the definition of distance between keywords, next we describe the formulations to define the similarity between a pair of users.

5.2 Similarity functions

Assume that a social network user v has N_v keywords and let K_i^v ($1 \leq i \leq N_v$) be his/her keywords. Consider two users

Table 4 User similarity to user Z

User	$k(u, Z)$	$n(u, Z)$		$s(u, Z)$		$S(u, Z)$
		Base	All	Base	All	
A	70	0	27	0.00	0.386	0.142
B	40	2	25	0.05	0.625	0.262
C	40	0	29	0.00	0.725	0.267

u and v on the network. Let $k(u, v)$ $(N_u \times N_v)$ be the total number of keyword pairs that they have. Also, let $n(u, v)$ be the number of keyword pairs (K_i^u, K_j^v) such that K_i^u and K_j^v and K_j^v belong to the same tree in F. How do we measure the similarity between u and v? How will the similarity between u and v vary when the keyword pairs belong to the same tree compared to the similarity between u and v when keyword pairs also belong to the same tree but at different hierarchical levels? We define two similarity functions to address these questions. We describe these functions next.

Weak similarity: This function defines the similarity between users when keyword pairs belong to the same tree. Thus, for two users, u, v with keywords K_1 and K_2 respectively, whenever $D(K_1, K_2) \neq \infty$, $n(u, v)$ is incremented by 1. Formally, it is defined as follows.

Definition 2 For two users u and v in the social network, the 'weak similarity', $s(u,v)$, between them is defined as:

$$s(u, v) = \frac{n(u, v)}{k(u, v)} \tag{1}$$

The position of the keywords inside the tree is not taken into account, i.e., keywords with distinct distance values will contribute equally towards the weak similarity. The word 'weak' is used to define the function because conceptually the definition ignores the position of the keywords and only tries to capture the fact whether two keywords belong to the same tree. In order to measure similarity between users with due consideration to position of the keywords we next define 'strong similarity'.

Strong similarity: We utilize the definition of keyword distance to define this function. We use exponential function for the definition because it has finite values at the boundary conditions of $D(K_i^u, K_j^v)$ (as $e^{-0} = 1$ and $e^{-\infty} = 0$ for $D(K_i^u, K_j^v) = 0$ and $D(K_i^u, K_j^v) = \infty$, respectively). Formally, it is defined as follows.

Definition 3 For two users u and v in the social network, the 'strong similarity', $S(u,v)$, between them is defined as:

$$S(u, v) = \frac{\sum_{\substack{1 \le i \le N_u \\ 1 \le j \le N_v}} e^{-D(K_i^u, K_j^v)}}{k(u, v)} \tag{2}$$

The function S is called 'strong similarity', as it considers the relative position of the keywords in the tree. It may happen that strong similarity is numerically smaller than the weak similarity but still it is a relatively stronger definition as it captures more information. Using this definition, keywords at a greater distance contribute less towards the similarity value. The value of $S(u, v)$ decreases as the distance between the keywords increases implying that u and v share lesser interests or attributes.

We use examples from Table 2 and the heuristics 'Base' and 'All' to analyze the similarity functions. The results are presented in Table 4. In Table 2, we saw user C was significantly similar to user Z than user B was to user Z (72.5% compared to 62.5%). This happened despite the fact that B and Z shared 2 similar keywords among them. But, when we see the values of 'strong similarity' between the respective users in Table 4, we see the difference in values have lowered (0.267 compared to 0.262). This is because now the distant keywords contributed less towards the similarity and the two similar keywords played a more dominant role. From here we see how 'weak similarity' can capture the general similarity between users and how 'strong similarity' is successful in capturing the similarity between users more broadly. Next we talk about the results obtained by analzying Facebook profiles.

6 Results and discussion

In this section, we describe the results of analyzing the Facebook profiles for similarity according to the 'forest model' and the similarity functions. First, we present results from the analysis on the number of keyword pairs the forest model was successful in matching. Second, we present results describing the variations in number of matches between keyword pairs and the variations in weak similarity and strong similarity for different number of keyword pairs between two users. Finally, we present results showing the variation in weak similarity and strong similarity based on different node degree of users and their individual number of keywords.

User pairs across the available network data is divided in the following three categories. Friend Pairs: When a user pair is formed such that the users are direct friends in the network. Friend^2 Pairs: When a user pair is formed such the participating user pairs share a common friend and are separated in the network by 1 hop. All Pairs: In this category, we consider all users pairs irrespective of the topological distance between them in the network. The 'All Pairs' category helps us to compare entries of more than half a million user pairs. Now, we describe the results obtained by comparing keywords of user pairs belonging to each of the categories. We compare the keywords according to each of the heuristics defined in Sect. 4.2.

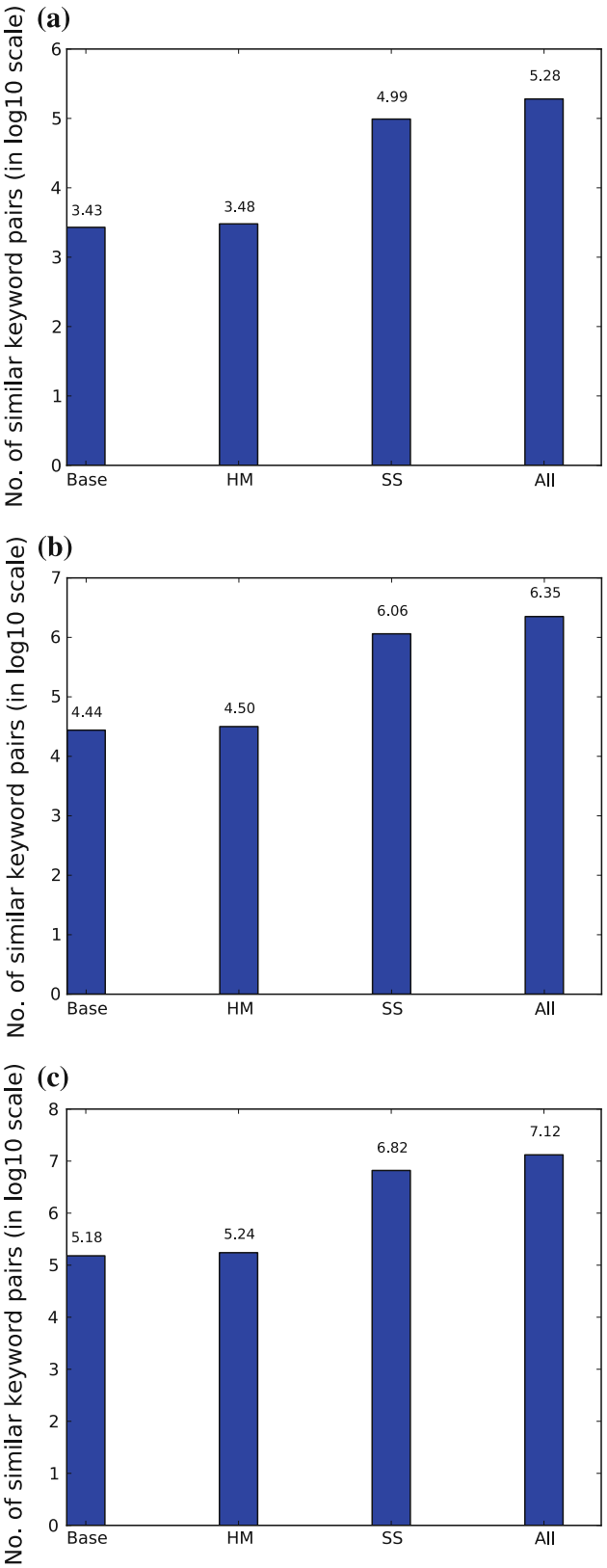

Fig. 4 Total number of keyword pairs similar to each other such that both keywords belong to the same tree for each of the different heuristics. The values are presented in a $\log_{10}$ scale. **a** Friend Pairs, **b** Friend^2 Pairs, **c** All Pairs

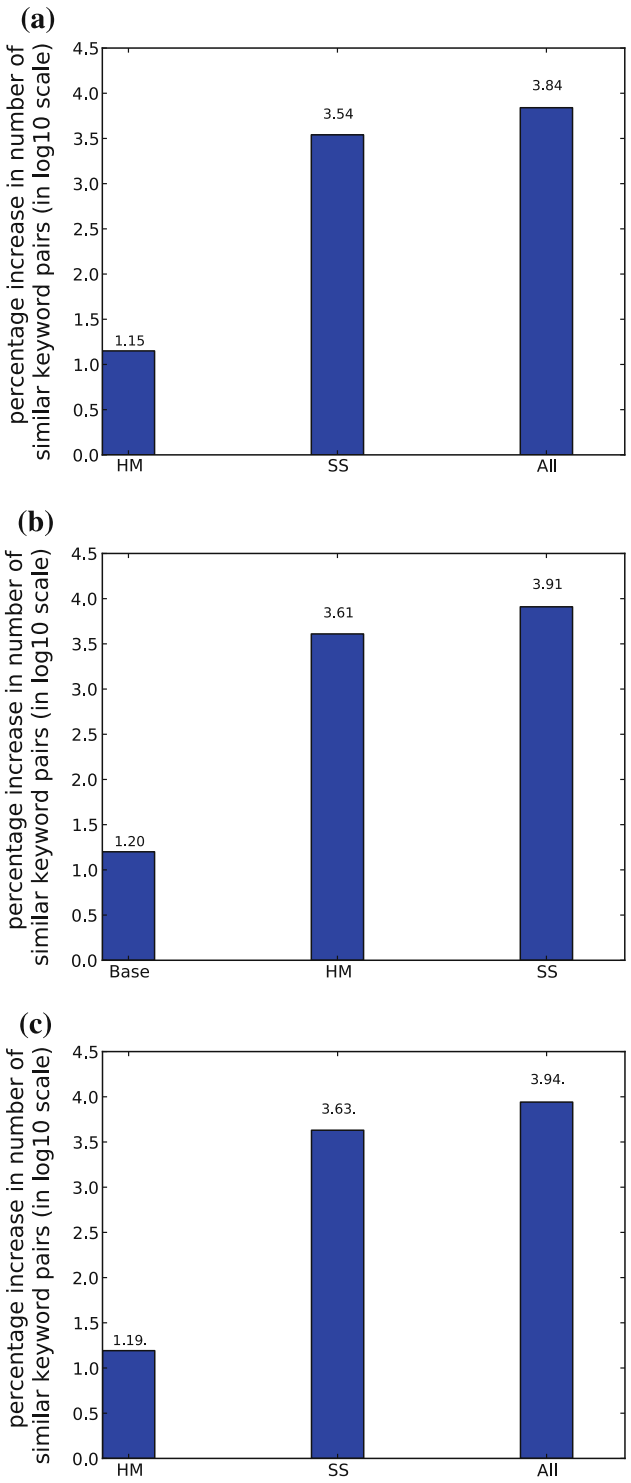

Fig. 5 Increase in number of keyword pairs similar to each other from the 'Base' heuristic such that both keywords belong to the same tree for each of the different heuristics. The values are presented in a $\log_{10}$ scale. **a** Friend Pairs, **b** Friend^2 Pairs, **c** All Pairs

Figure 4 shows the total number of keyword pairs that matched. We observe from the figure how the ontologies present in the WordNet influence the number of matching

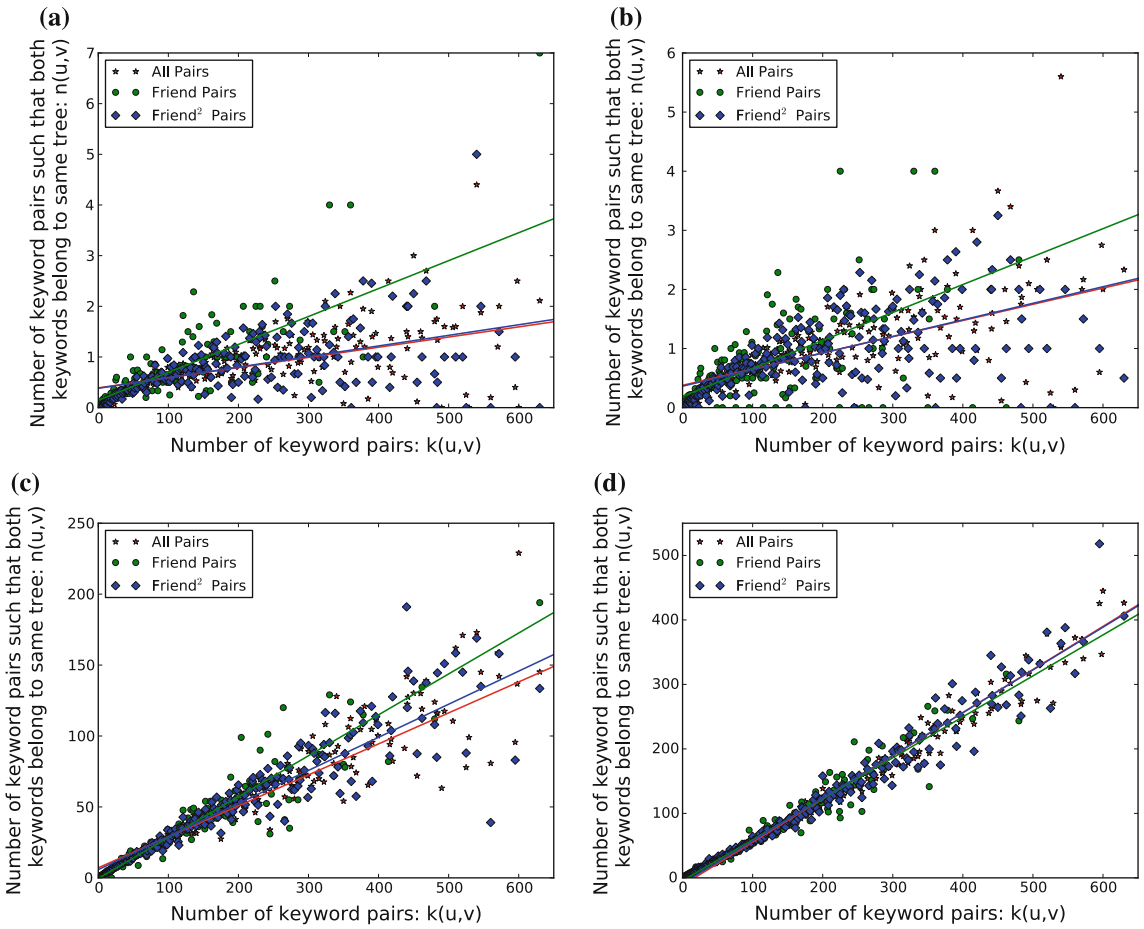

Fig. 6 Variation in number of keyword pairs such that both keyword belong to the same tree for increasing number of keyword pairs between users. Values plotted for each different heuristic. **a** Base, **b** HM, **c** SS, **d** All

keywords. For 'Base' and 'HM' heuristics, the numbers are approximately equal for each of the different user pairs. But the numbers rise significantly for 'SS' and 'All'. The increase in numbers compared to the 'Base' case are presented in Fig. 5. The results confirm the intuition that keywords are related to each other and the 'forest model' can successfully capture the relations to measure the similarity between users. Next, we discuss how keyword pairs match for users for different number of keyword pairs between the users.

In Fig. 6, we present results on the variation in number of keyword pairs belonging to the same tree for different number of keyword pairs between the users. The solid line in all the figures shows the trend line for the values. The numbers gradually increase in all the figures with an increasing number of keyword pairs. This is primarily due to the property of the respective heuristics as more related keywords are available in the WordNet ontology for a keyword for the 'SS' category than in the 'HM' category. Figure 6a shows the numbers for the 'Base' case. The values show a relatively low increase even though the number of keyword pairs keep increasing. The maximum number of matches is seen for pairs that are direct friends. The most important factor to notice here is how the values are almost equal for pairs that belong to 'Friend^2 Pairs' category and for the 'All Pairs' category. Similar trends are visible in Fig. 6b, c.

In Fig. 6d, we observe different results. Since in this heuristic, WordNet ontology provides a large number of keywords related to the inspected keyword, thus increasing the chances of a matching, we observe that values for all three different categories of user pairs are approximately equal to each other. Moreover, the values for 'All Pair' even surpass the values for 'Friend Pairs'. In Fig. 7, we plot the values for each of the three different categories of user pairs for different heuristic parameters. The influence of WordNet ontology is also reflected in the values here as the number of keyword belonging in same trees keep growing up. Next, we will see how these values effect the similarity values between the users.

Figure 8 contains the plots showing the variation of weak similarity. The trend lines in Fig.8 leads to the following interesting observations. First, in Fig. 8a, we see that

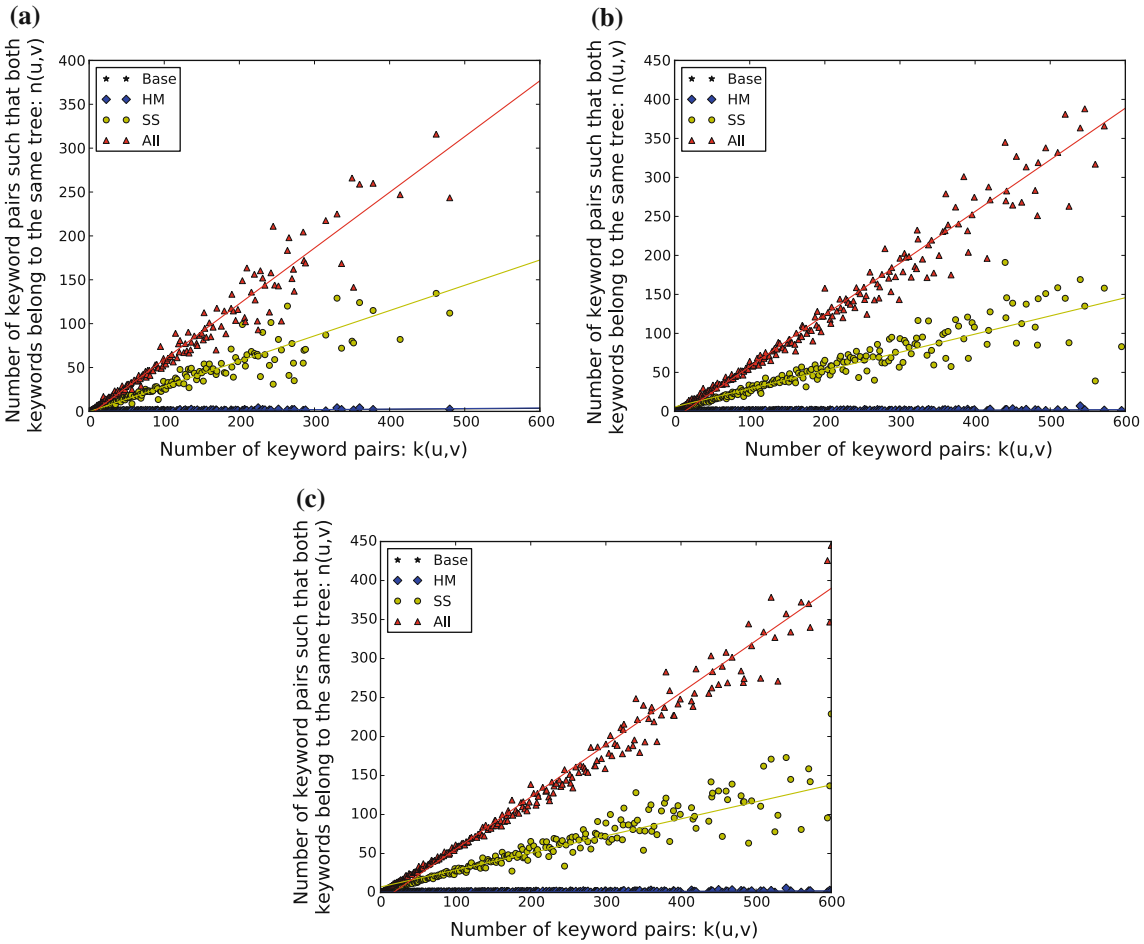

Fig. 7 Variation in number of keyword pairs such that both keyword belong to the same tree for increasing number of keyword pairs between users. Values plotted on individual plots across all heuristics. **a** Friend Pairs, **b** Friend^2 Pairs and **c** All Pairs

similarity values for 'Friend Pairs' are higher as compared to the values of other two pairs for all the heuristics. This observation is true for the entire plots of Fig. 8a–d till the number of keyword pairs between the users are less than 300. From here we conclude that friend pairs are more similar than any other pair in the social network. Second, for the first three heuristics, i.e., in Fig. 8a–c, the similarity values fall as number of keyword pairs between two users increases. This trend is reversed in the final heuristic and the similarity values for either of the pair categories increase from 0.60 to 0.65. This is because the wide distribution in usage of keywords are closed down due to the characteristic of the heuristic. The heuristic this time is successful to relate and match keywords inside a pair.

Third, it is also interesting to see how the trend lines between the different categories of friends behave for different heuristics. For example, in 'Base' and 'HM', the similarity between 'All Pairs' crosses the trend line for similarity of 'Friend Pairs' at increasing count of keyword pairs. This trend reverses in the later two heuristic cases as the gap in similarity values between 'Friend Pairs' and other pairs keep increase as the number of keyword pairs increases. We can see from here how the relations between the keywords play a role in determining the similarity between any user pairs and how a model like the 'forest model' is crucial to homophily analysis in social networks. Fourth, it is also interesting to note how the similarity values between 'Friend^2 Pairs' are always so close to the values of 'All Pairs' for each of the heuristics. We conclude from these observation that similarity between friends of friends is almost equal to the similarity between any pair of users, i.e., topological distance between users does not significantly effect the similarity between users after the first hop. In other words, friend pairs are relatively high in similarity but beyond that, any user is almost similar to every one another, irrespective of the topological distance.

In Fig. 9, we see the values broken down for each pair categories for all the heuristics together. The figures corroborates the set of observations we made from Fig. 8. Next, we will talk about the 'strong similarity' between users and how they vary with increasing number of keyword pairs between users.

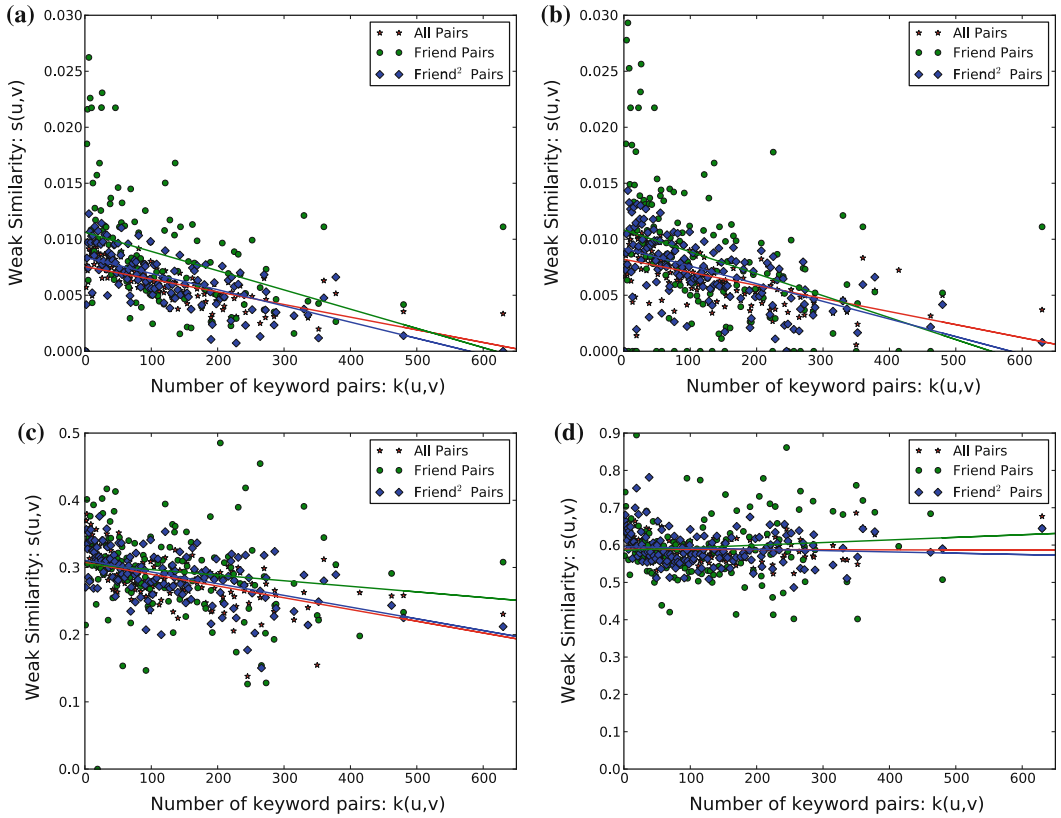

Fig. 8 Weak similarity versus number of keyword pairs. Values plotted for each different heuristic. **a** Base, **b** HM, **c** SS, **d** All

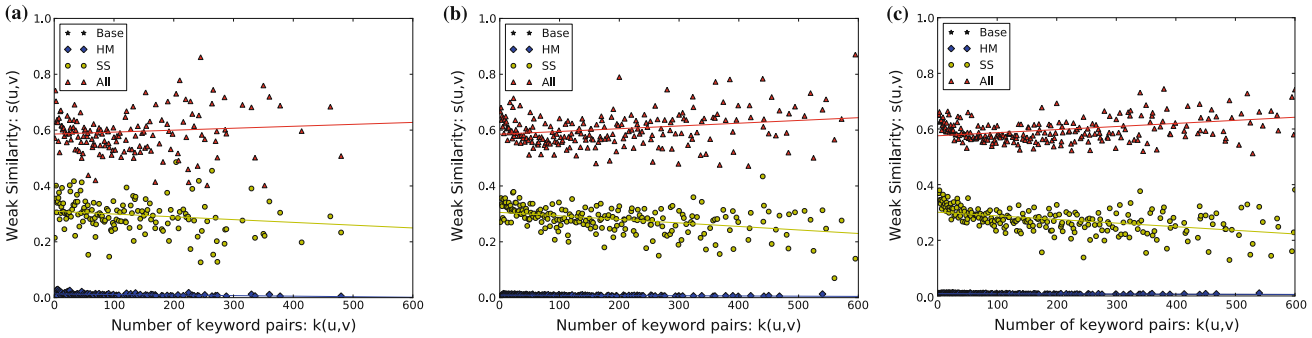

Fig. 9 Weak similarity versus number of keyword pairs. Values plotted on individual plots across all heuristics. **a** Friend Pairs, **b** Friend^2 Pairs and **c** All Pairs

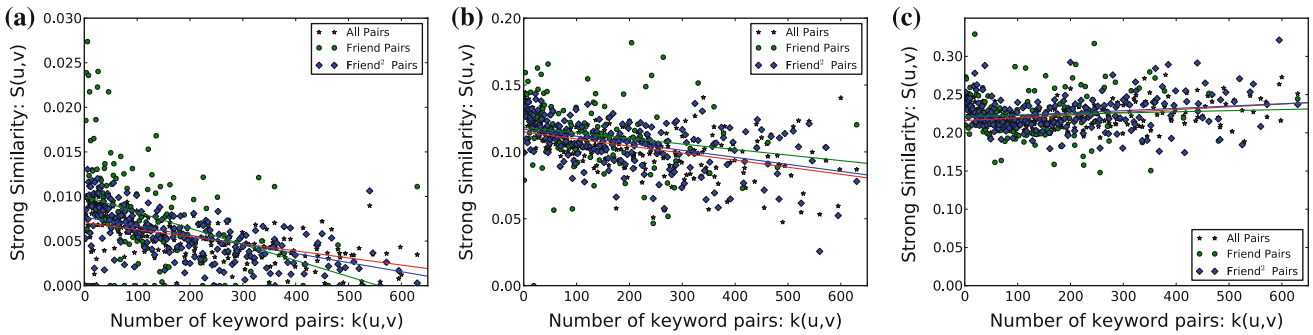

Fig. 10 Strong similarity versus number of keyword pairs. Values plotted for each different heuristic. **a** HM, **b** SS, **c** All

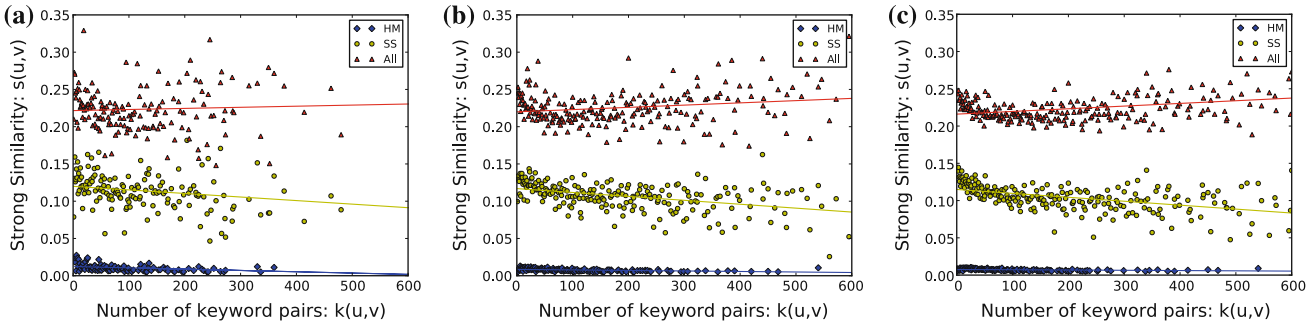

Fig. 11 Strong similarity versus number of keyword pairs. Values plotted on individual plots across all heuristics. **a** Friend Pairs, **b** Friend^2 Pairs and **c** All Pairs

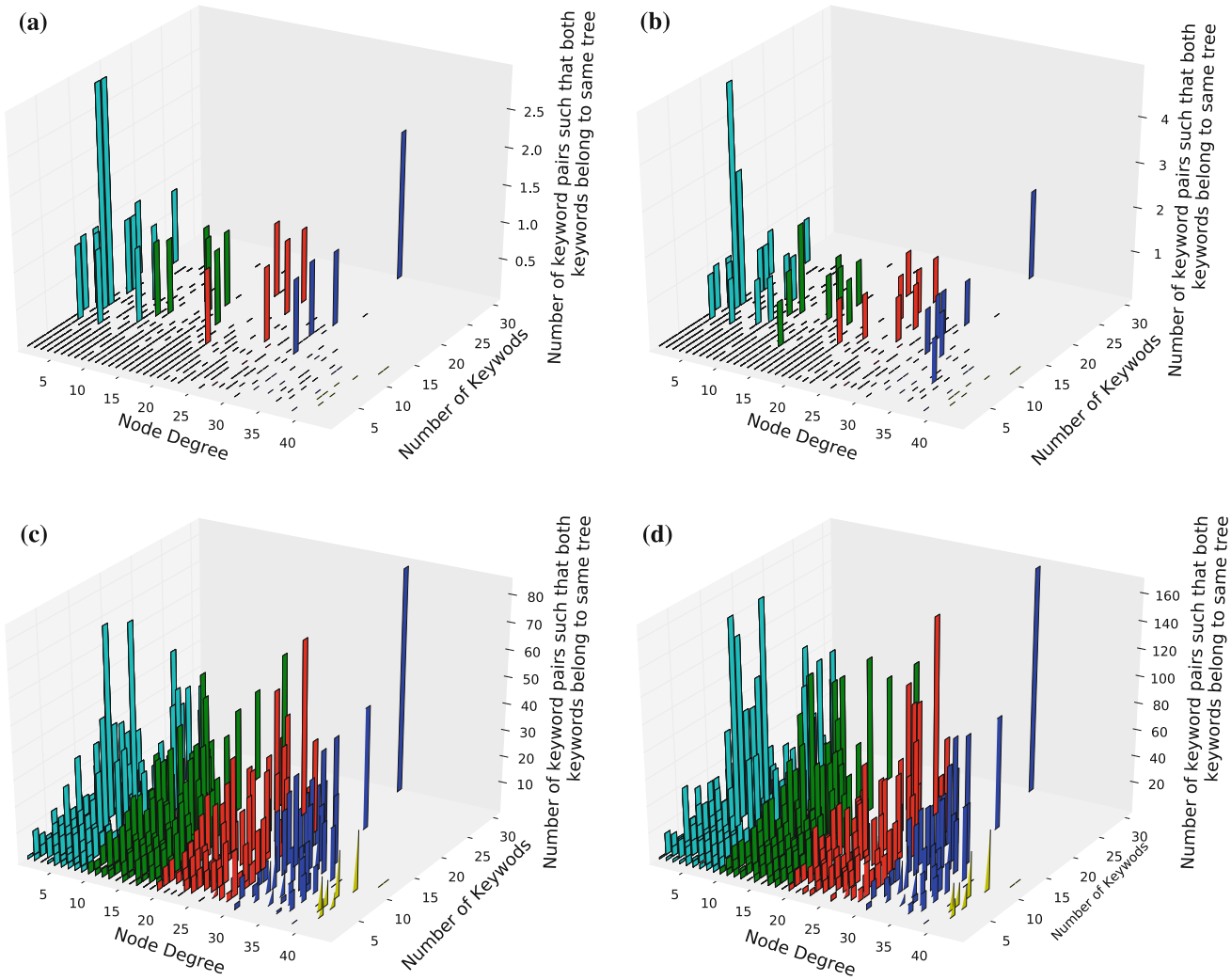

Fig. 12 Number of keyword pairs such that both keyword belong to same tree versus node degree and number of keywords. Values plotted for each different heuristic. **a** Base, **b** HM, **c** SS, **d** All

Figures 10 and 11 plot the 'strong similarity' between user pairs versus the number of keyword pairs between the users. The initial set of observations here are the repetition of the previous observations where we saw the friend pairs are more similar than the other set of pairs. This observation can be seen again in Fig. 10a, b. The next most interesting observation here is how the trend lines for the values of similarity seem to coincide for either of the three categories of pairs in Fig. 10c with very little difference in values for increasing number of keyword pairs between

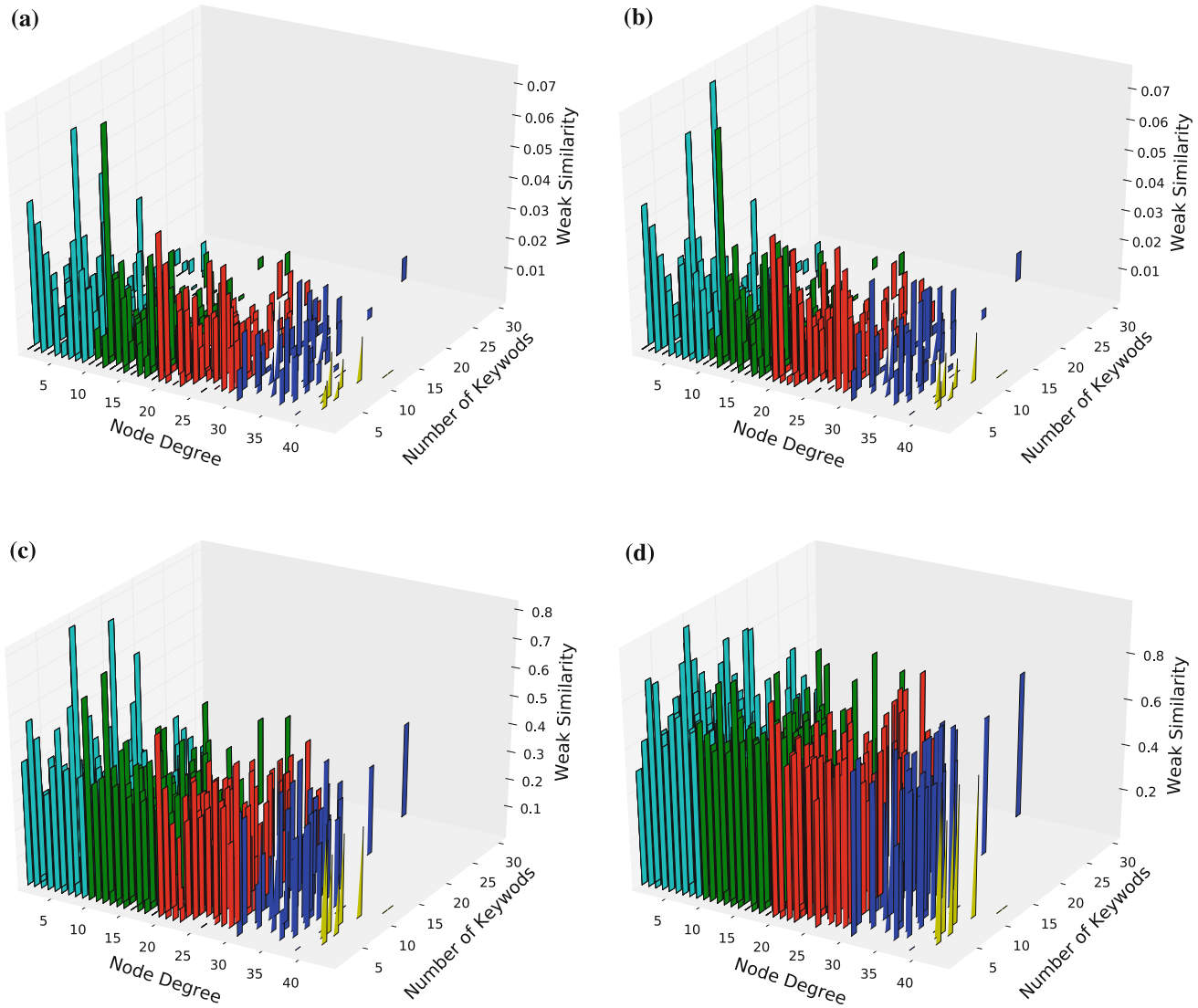

Fig. 13 Weak similarity versus node degree and number of keywords. Values plotted for each different heuristic. **a** Base, **b** HM, **c** SS, **d** All

users. We conclude from here that this happens because keywords are closely tied in their usage and when similarity is measured by giving weight-age to keyword positions in the hierarchy, similarity values between users come so close.

In Figs. 12, 13 and 14, we plot the values of number of keywords matching in a single tree, weak similarity and strong similarity versus the node degree of each user and the number of keywords an individual user has, respectively. The goal here is to understand the cumulative effect of increased degree, i.e., number of friends per individual user and the number of keywords on the average similarity across for each user pair. The plots are varied in color after every increment of 10 degrees for easy understanding. We ignore a few scattered values for node degrees greater than 45 also for ease in viewing the plots and to obtain meaningful conclusions. The result in Fig. 12 is quite intuitive and shows that as the number of keyword increase for an individual user, the number of keywords that match with its friend's keyword also go up. In Figs. 13 and 14, we observe that as the node degree and the number of keywords increases for the users, the average similarity a user has along with its friends come down. This observation is significant because it shows that users become more divergent in their interests to form new friendships, resulting in a decrease of similarity. activities. Intuitively, this conclusion is similar to what we expect in real life where a user interested in many different topics may have a large social network but his average similarity to others will be lower compared to someone who mixes in a small circle of friends and is only interested in a certain few topics. e as node degree and number of keywords of a user increases is same. This concludes our discussion on evaluation of the Facebook

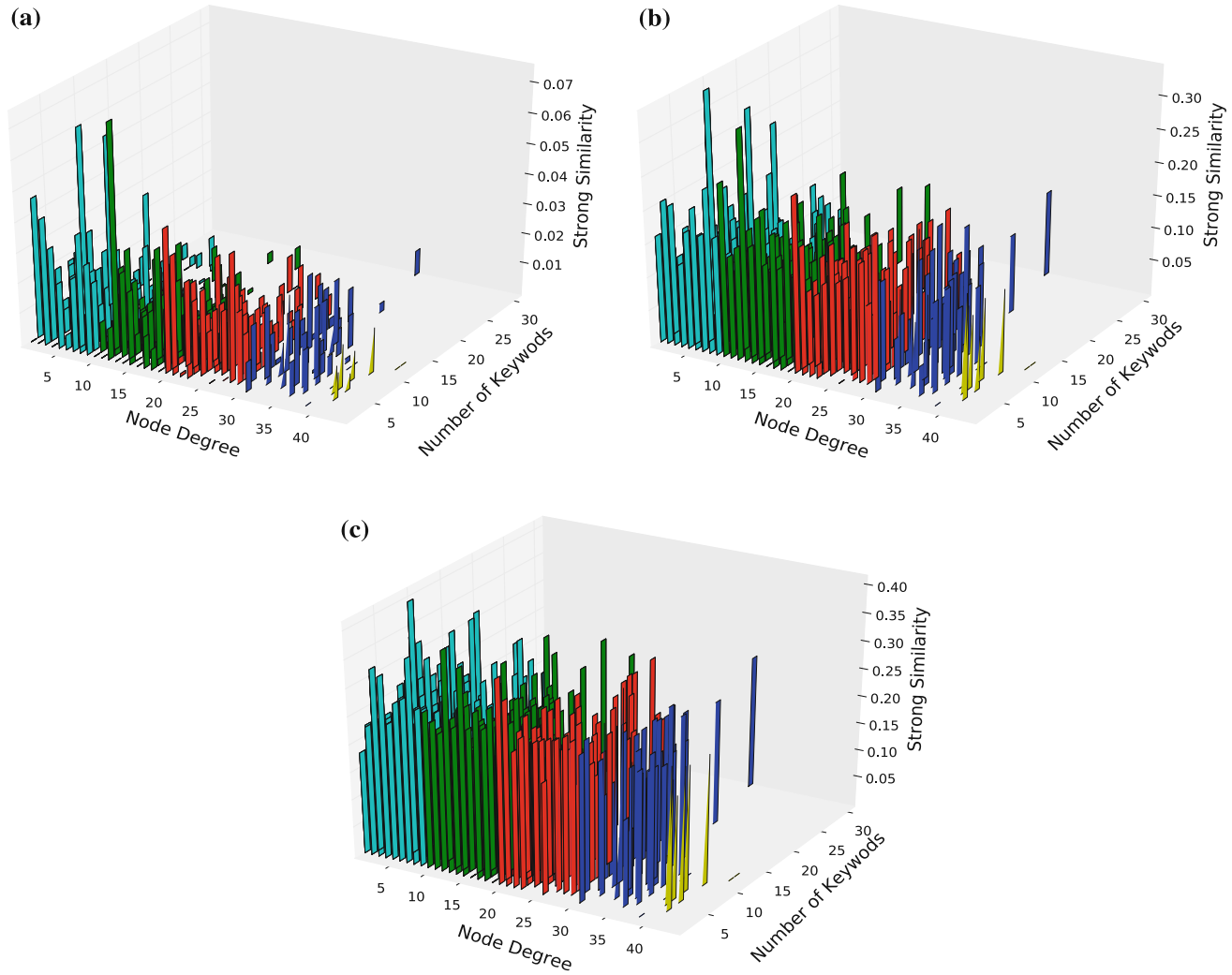

Fig. 14 Strong similarity versus node degree and number of keywords. Values plotted for each different heuristic. **a** HM, **b** SS, **c** All

profiles. In the next section, we conclude our work and present a discussion on the future works.

7 Concluding remarks

In this paper, we studied the similarity between users in an online social network. We based our studies on user similarity by evaluating the similarity between user keywords. First, we studied the distribution of user keywords in online social networks. Next, we defined a 'forest model' to link related keywords. The model links keywords based on the semantic relations. We showed how the model is able to quantify the similarity between seemingly unrelated user profile information available in social networks. Based on the model, we defined two different types of functions to quantify the similarity between users. Next, we evaluated a dataset containing Facebook user profiles for similarity between the users using the 'forest model' and the similarity functions.

We saw that user keywords can be aggregated effectively, based on the heuristic used to generate the 'forest', to evaluate user similarity. Based on our evaluations, we conclude that direct friends are more similar than any other user pair in the social network. The similarity between users remains approximately the same, irrespective of the topological distance between them. Finally, we also observed that with an increase in the node degree and number of keywords for a user, the average similarity a node has with its friends comes down.

Future research would augment the social network model based upon user similarity functions that we proposed in our earlier work (Bhattacharyya et al. 2009). The motivation is to generate an online social network model based upon a user's similarity with other users and establish links when certain levels of similarity are observed.

Another direction is to develop social search query models by comparing the similarity among friends.

Acknowledgments We are thankful to Matthew Spear who provided us the Facebook data. We also thank Lerone Banks for his help during the manuscript preparation. We thank the anonymous reviewers for their insightful comments. This work was supported by the National Science Foundation FIND (Future Internet Design) program under Grant No. 0832202, MURI under ARO (Army Research Office) and Network Science CTA under ARL (Army Research Laboratory).

References

Adamic LA, Adar E (2001) Friends and neighbors on the web. Soc Netw 25:211–230

Adamic LA, Buyukkokten O, Adar E (2003) A social network caught in the web. First Monday 8(6)

Banks L, Ye S, Huang Y, Wu SF (2007) Davis social links: integrating social networks with internet routing. In: LSAD '07: Proceedings of the 2007 workshop on large scale attack defense ACM Press, New York, pp 121–128

Banks L, Bhattacharyya P, Spear M, Wu SF (2009) Davis social links: Leveraging social networks for future internet communication. Ninth annual international symposium on applications and the internet, pp 165–168

Bhattacharyya P, Garg A, Wu SF (2009) Social network model based on keyword categorization. International conference on advances in social network analysis and mining (ASONAM'09), pp 170–175

Crandall D, Cosley D, Huttenlocher D, Kleinberg J, Suri S (2008) Feedback effects between similarity and social influence in online communities. In: KDD '08: Proceeding of the 14th ACM SIGKDD international conference on knowledge discovery and data mining. ACM Press, New York, pp 160–168

Deerwester S, Dumais ST, Furnas GW, Landauer TK, Harshman R (1990) Indexing by latent semantic analysis. J Am Soc Inform Sci 41:391–407

Fellbaum C (1998) Wordnet: an electronic lexical database. Bradford Books, Bradford

Howe DC (2009) Rita wordnet Java based API to access Wordnet. http://www.rednoise.org/rita. Accessed 26 October

Kleinberg J (2000) The small-world phenomenon: an algorithm perspective. In: STOC '00: Proceedings of the 32nd annual ACM symposium on theory of computing

Kleinberg J (2001) Small-world phenomena and the dynamics of information. In: Advances in neural information processing systems. MIT Press, Cambridge, pp 431–438

Kumar R, Novak J, Raghavan P, Tomkins A (2004) Structure and evolution of blogspace. Commun ACM 47(12):35–39

Liben-Nowell D, Kleinberg J (2007) The link-prediction problem for social networks. J Am Soc Inform Sci Technol 58(7):1019–1031

Liben-Nowell D, Novak J, Kumar R, Raghavan P, Tomkins A (2005) Geographic routing in social networks. PNAS 102(33):11623–11628

Mcpherson M, Lovin LS, Cook JM (2001) Birds of a feather: homophily in social networks. Annu Rev Sociol 27(1):415–444

Milgram S (1967) The small world problem. Psychol Today 61:60–67

Sandberg O (2007) The structure and dynamics of navigable networks. PhD thesis, Chalmers University

Şimşek Ö, Jensen D (2005) Decentralized search in networks using homophily and degree disparity. In: Nineteenth international joint conference on artificial intelligence (IJCAI 2005)

Şimşek Ö, Jensen D (2008) Navigating networks by using homophily and degree. Proc Natl Acad Sci 105(35):12758–12762

Spear M, Lu X, Matloff NS, Wu SF (2009) Inter-profile similarity (ips): a method for semantic analysis of online social networks. In: Complex '09: Proceedings of the first international conference on complex sciences: theory and applications

Travers J, Milgram S, Travers J, Milgram S (1969) An experimental study of the small world problem. Sociometry 32:425–443

Xu Z, Fu Y, Mao J, Su D (2006) Towards the semantic web: collaborative tag suggestions. In: WWW'06: Proceedings of the collaborative web tagging workshop