Understanding Latent Interactions in Online Social Networks

JING JIANG, Peking University and State Key Laboratory of Software Development Environment Beihang University
CHRISTO WILSON, University of California, Santa Barbara
XIAO WANG, WENPENG SHA, PENG HUANG, and YAFEI DAI, Peking University
BEN Y. ZHAO, University of California, Santa Barbara

Popular online social networks (OSNs) like Facebook and Twitter are changing the way users communicate and interact with the Internet. A deep understanding of user interactions in OSNs can provide important insights into questions of human social behavior and into the design of social platforms and applications. However, recent studies have shown that a majority of user interactions on OSNs are *latent interactions*, that is, passive actions, such as profile browsing, that cannot be observed by traditional measurement techniques.

In this article, we seek a deeper understanding of both active and latent user interactions in OSNs. For quantifiable data on latent user interactions, we perform a detailed measurement study on Renren, the largest OSN in China with more than 220 million users to date. All friendship links in Renren are public, allowing us to exhaustively crawl a connected graph component of 42 million users and 1.66 billion social links in 2009. Renren also keeps detailed, publicly viewable visitor logs for each user profile. We capture detailed histories of profile visits over a period of 90 days for users in the Peking University Renren network and use statistics of profile visits to study issues of user profile popularity, reciprocity of profile visits, and the impact of content updates on user popularity. We find that latent interactions are much more prevalent and frequent than active events, are nonreciprocal in nature, and that profile popularity is correlated with page views of content rather than with quantity of content updates. Finally, we construct *latent interaction graphs* as models of user browsing behavior and compare their structural properties, evolution, community structure, and mixing times against those of both active interaction graphs and social graphs.

Categories and Subject Descriptors: J.4 [**Computer Applications**]: Social and Behavioral Sciences; H.3.5 [**Information Storage and Retrieval**]: Online Information Services

General Terms: Human Factors, Measurement, Performance

Additional Key Words and Phrases: Latent interaction, online social networks, measurement

---

This work is an extend and revised version of an article in *Proceedings of the Internet Measurement Conference* [Jiang et al. 2010].
This work is supported in part by the National Science Foundation of China under the National Basic Research Program of China grant No. 2011CB302305, the Project of the State Key Laboratory of Software Development Environment under grant No. SKLSDE-2013ZX-26, the National Natural Science Foundation of China under grant No. 61202423, and Fundamental Research Funds for the Central Universities under grant No. YWF-13-T-RSC-077. It is also supported by the NSF under IIS-0916307 and CNS-1224100, as well as DARPA GRAPHS BAA-12-01. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.
Authors' addresses: J. Jiang, Department of Computer Science, Peking University, Beijing 100871, China; and State Key Laboratory of Software Development Environment, Beihang University, Beijing 100191, China; email: jiangjing@buaa.edu.cn; C. Wilson (corresponding author) and B. Y. Zhao, Department of Computer Science, U. C. Santa Barbara, Santa Barbara, CA 93106; email: {bowlin, ravenben}@cs.ucsb.edu; X. Wang, W. Sha, P. Huang, and Y. Dai, Department of Computer Science, Peking University, Beijing 100871, China; email: {wangxiao, swp, huangpeng}@net.pku.edu.cn, dyf@pku.edu.cn.

DOI: http://dx.doi.org/10.1145/2517040

ACM Reference Format:
Jiang, J., Wilson, C., Wang, X., Sha, W., Huang, P., Dai, Y., and Zhao, B. Y. 2013. Understanding latent interactions in online social networks. ACM Trans. Web 7, 4, Article 18 (October 2013), 39 pages.
DOI: http://dx.doi.org/10.1145/2517040

1. INTRODUCTION

Not only are online social networks (OSNs) popular tools for interaction and communication, but they have the potential to alter the way users interact with the Internet. Today's social networks already count close to one billion members worldwide. Facebook, the most popular OSN, has more than one billion active users and has surpassed Google as the most visited site on the Internet [Yarow 2010]. Increasingly, Facebook and Twitter are replacing email and search engines as users' primary interfaces to the Internet [Gannes 2010; Kirkpatrick 2009]. This trend is likely to continue, as networks like Facebook seek to personalize the Web experience by giving sites access to information about their visitors and their friends through platforms such as OpenGraph.[1]

A deep understanding of user interactions in social networks can provide important insights into questions of human social behavior as well as the design of social platforms and applications. For example, gauging the level of reciprocity in social interactions could shed light on the factors that motivate interactions. In addition, understanding how interactions are distributed between friends could assist in tracking information dissemination in social networks, thus identifying "popular" or "influential" users to target in branding and ad campaigns [Chen et al. 2009; Gruhl et al. 2004; Kempe et al. 2003]. Finally, lessons from studying how users interact through different communication tools could guide the design of new, more engaging mechanisms for social interaction.

Initial measurement studies [Ahn et al. 2007; Mislove et al. 2007; Wilson et al. 2009] of OSNs focused on topological characteristics of the social graph, that is, the underlying structures of these services that captured explicit relationships between users. To better understand the true nature of relationships between OSN users, more recent work has shifted focus to measuring observable social interactions [Chun et al. 2008; Leskovec and Horvitz 2008; Viswanath et al. 2009; Wilson et al. 2009]. By examining records of interaction events across different links, the studies distinguish close-knit, active relationships from weak or dormant relationships and derive a more accurate predictive model for social behavior. Recently, two significant studies [Benevenuto et al. 2009; Schneider et al. 2009] used clickstream data at the network level to capture the behavior of OSN users and revealed that passive or *latent interactions*, such as profile browsing, often dominate user events in a social network [Benevenuto et al. 2009].

Unfortunately, these studies have been constrained by several limitations of clickstream data. First, the type of data captured in a clickstream is highly dependent on the time range of the clickstream. Captured events are also from the perspective of the current user, making it challenging to correlate events across time and users. Second, clickstream data is also highly dependent on the structure of the OSN site and can be extremely challenging to reduce large volumes of data to distinct user events. Finally, each application-level user event generates a large volume of clickstream data, and extremely large clickstreams are needed to capture a significant number of user events. These properties of verboseness and complexity mean that it is extremely difficult to gather enough clickstream data to study user interactions comprehensively at scale. However, a comprehensive and large study is necessary for answering many of the deeper questions about user behavior and interactions, such as are user interactions reciprocal, do latent interactions such as profile browsing reflect the same popularity

[1] `http://opengraphprotocol.org`.

distributions as active actions like user comments, what can users do to become "popular" and draw more visitors to their pages?

In this article, we seek to answer these and other questions in our search for a deeper understanding of user interactions in OSNs. To address the challenge of gathering data on latent interactions, we perform a large-scale, crawl-based measurement of the Renren social network,[2] the largest and most popular OSN in China. Functionally, it is essentially a clone of Facebook, with similar structure, layout, and features. Like Facebook, Renren also evolved from a university-based social network (a predecessor called Xiaonei). Unlike Facebook, Renren has two unique features that make it an attractive platform on which to study user interactions.

First, while Renren users have full privacy control over their private profiles, their friend lists were public and unprotected by privacy mechanisms (until additional privacy mechanisms were added in late 2010). This allowed us to crawl an exhaustive snapshot of Renren's largest connected component, producing an extremely large social graph with 42.1 million nodes and 1.66 billion edges. Second, and perhaps more importantly, Renren user profiles make a variety of statistics visible to both the profile owner and her visitors. Each user profile keeps a visible list of "recent visitors" who browse the profile, sorted in order, and updated in real time. Each photo and diary entry also has its own page with a count of visits by users other than the owner. These records are extremely valuable in that they expose latent browsing events to our crawlers, granting us a unique opportunity to gather and analyze large-scale statistics on latent browsing events.

Our Study. Our study of latent user interactions includes three significant components. First, we begin by characterizing properties of the large Renren social graph and compare them to known statistics of other OSNs, including Facebook, Cyworld, Orkut and Twitter. Our second component focuses on questions concerning latent interactions and constitutes the bulk of our study. We describe a log reconstruction algorithm that uses relative clocks to merge visitor logs from repeated crawls into a single sequential visitor stream. We repeatedly crawl users in the Peking University Renren network over a period of 90 days, extract profile visit history for 61K users, and examine issues of popularity, visitor composition, reciprocity, and latency of reciprocation. We define *popularity* as the number of views a user's profile receives. We compare user popularity distributions for latent and active interactions and use per-object visit counters to quantify the level of user engagement generated from user profiles, photos, and diary entries. We also study correlation of different types of user-generated content with a user's profile popularity using complete interaction records obtained directly from Renren. Finally, in our third component, we build *latent interaction graphs* from our visitor logs and compare their structure to those of social graphs and interaction graphs. This includes comparing topological graph properties, temporal dynamics, community structure, and mixing time. Our analysis finds that latent interaction graphs exhibit features that fall between the social graph and the active interaction graph. We revisit the issue of experimental validation for social applications and perform case studies of the impact of different graphs on evaluating information dissemination algorithms and social email whitelists.

Our study provides a number of insights into user behavior on online social networks.

—Users' profile popularity varies significantly across the population and closely follows a Zipf distribution.
—Profile visits have extremely low reciprocity, despite the fact that Renren users have full access to the list of recent visitors to their profile.

[2]`http://www.renren.com`.

—Compared to active interactions, latent profile browsing is far more prevalent and more evenly distributed across a user's friends. Profile visits are less likely to be repeated than active interactions but are more likely to generate active comments than other content, such as photos and diary entries.
—Users receive a significant part of visits from strangers. Social networks help people find and view strangers' profiles, but the effect varies greatly from person to person.
—For all users, regardless of their number of friends, profile popularity is not strongly correlated with frequency of new profile content.

2. METHODOLOGY AND INITIAL ANALYSIS

Before diving into detailed analysis of user interaction events, we begin by providing background information about the Renren social network and our measurement methodology. We then give more specifics on our techniques for reconstructing profile browsing histories from periodic crawls. Using a random subset of user profiles, we perform sampling experiments to quantify the expected errors introduced by our approach. We analyze characteristics of the Renren social graph and compare it to known graph properties of existing social graph measurements. Finally, we make a deep analysis of isolated users in campus network.

2.1. The Renren Social Network

Launched in 2005, Renren is the largest and oldest OSN in China. Renren can be best characterized as Facebook's Chinese twin, with most or all of Facebook's features, layout, and a similar user interface. Users maintain personal profiles, upload photos, write diary entries (blogs), and establish bidirectional social links with their friends. Renren users inform their friends about recent events with 140-character status updates, much like tweets on Twitter. Similar to the Facebook news feed, all user-generated updates and comments are tagged with the sender's name and a timestamp.

Renren organizes users into membership-based networks, much like Facebook used to. Networks represent schools, companies, or geographic regions. Membership in school and company networks require authentication. Students must offer an IP address, email address, or student credential from the associated university. Corporate email addresses are needed for users to join corporate networks. Renren's default privacy policy makes profiles of users in geographic networks private. This makes them difficult to crawl [Wilson et al. 2009]. Fortunately, profiles of users in authenticated networks are public by default to other members of the same network. This allowed us to access user profiles within the Peking University network, since we could create nearly unlimited authenticated accounts using our own block of IP addresses.

Like Facebook, a Renren user's homepage includes a number of friend recommendations that encourage formation of new friend relationships. Renren lists three users with the most number of mutual friends in the top-right corner of the page. In addition, Renren shows a list of eight "popular users" at the very bottom of the page. These popular users are randomly selected from the 100 users with the most friends in the university network.

User profiles on Renren are very similar to Facebook. Each profile includes a profile picture, personal information (name, age, education background, work experience, hobbies, etc.), and a subset of the user's friend list (since friend lists are often hundreds of users long). The body of each profile is a chronologically ordered "feed" of the user's actions: status updates, comments sent and received, photos uploaded and tagged, shared Web links, blog entries written, etc.

Unique features. Renren differs from Facebook in several significant ways. First, each Renren user profile includes a box that shows the total number of visitors to the

profile, along with names and links to the last nine visitors ordered from most to least recent. In addition, Renren also keeps on each individual photo and diary page a visible counter of visitors (not including the user himself). These lists and counters have the same privacy settings as the main profile. They have the unique property of making previously invisible events visible and are the basis for our detailed measurements on latent user interactions.

A second crucial feature is that friend lists in Renren were public in 2009 when we collected data for this study. Users had no way to hide them. This allowed us to perform an exhaustive crawl of the largest connected component in Renren (42.1 million users). This contrasts with other OSNs, where full social graph crawls are prevented by user privacy policies that hide friendship links from the public. The exception is Twitter, which behaves more like a public news medium than a traditional social network [Kwak et al. 2010]. Renren has since changed this policy: by default, friend lists are now only viewable by friends.

In addition, comments in Renren are threaded, that is, each new comment is always in response to one single other event or comment. For example, user A can respond to user B's comment on user C's profile, and only B is notified of the new message. Thus we can precisely distinguish the intended target of each comment. One final difference between Renren and Facebook is that each standard user is limited to a maximum of 1,000 friends. Users may pay a subscription fee to increase this limit to 2,000. From our measurements, we saw that very few users (0.3%) took advantage of this feature.

2.2. Data Collection and General Statistics

Like Facebook, Renren evolved from a social network in a university setting. Its predecessor was called Xiaonei, literally meaning "inside school." In September 2009, Renren merged with Kaixin, the second largest OSN in China, and absorbed all of Kaixin's user accounts.

Crawling the Renren Social Graph. We crawled the entire Renren network from April to June 2009, and again from September to November of 2009. We seed crawlers with the 30 most popular users' profiles and proceeded to perform a breadth-first traversal of the social graph. During the crawl, we collect unique user IDs, network affiliations, and friendship links to other users. For our study, we use data from our last crawl, which was an exhaustive snapshot that included 42,115,509 users and 1,657,273,875 friendship links. While this is significantly smaller than the 70 million users advertized by Renren in September 2009, we believe the discrepancy is due to Kaixin users who were still organized as a separate, disconnected subgraph. We describe properties of the social graph later in this section.

Crawling the PKU Network. We performed smaller, more detail-oriented crawls of the Peking University (PKU) network between September and November of 2009 (90 days) to collect information about user profiles and interaction patterns. This methodology works because the default privacy policy for authenticated networks is to make full profiles accessible to other members of the same network. Since we collected the network memberships of all users during our complete crawl, we were able to isolate the 100,973 members of the PKU network to seed our detailed crawl. Of these users, 61,405 users had the default, permissive privacy policy, enabling us to collect their detailed information. This covers the majority of users (60.8%) in the PKU network and provides overall network coverage similar to other studies that crawled OSN regional networks [Wilson et al. 2009].

As part of our PKU crawls, we gathered all comments generated by users in message board posts, diary entries, photos, and status updates. This data forms the basis of our experiments involving active interactions. Our dataset represents the record of public

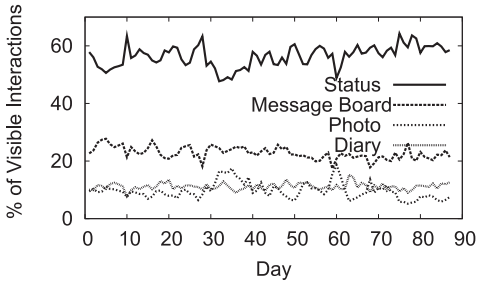


Fig. 1. Daily distribution of comments across applications.

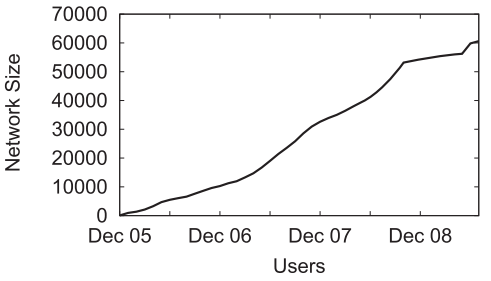


Fig. 2. Population growth of the PKU network over time.

Table I. Types of Social Data on PKU User Interactions Received Directly from Renren in 2010

	Diary	Photo	Status
Number of unique visitors	√	√	
Number of "shares"	√		
Length (in bytes)	√		√
Number of comments from owner	√	√	√
Number of comments from others	√	√	√

active interactions between users in the PKU network. In total, 19,782,140 comments were collected with 1,218,911 of them originating in the September to November 2009 timeframe.

Figure 1 plots the percentage of comments in various applications each day. The most popular events commented on are status updates, which accounts for roughly 55% of all daily comments. Message boards cover 25%, while diary and photo each account for roughly 10%.

Figure 2 shows the growth of the PKU network over time. Although Renren does not disclose the account creation times of users, we can estimate each account's lifetime by looking at the oldest comment sent or received by that user [Wilson et al. 2009]. We observe a linear increase in PKU network size over time. This trend makes intuitive sense for an affiliation-based network, that is, there is a (roughly) constant number of new students admitted to PKU each year, a subset of whom create Renren accounts.

Privacy and Data Anonymization. Our study focuses on the structure of social graphs and interaction events between users. Since we do not need any actual content of comments, photos, or user profiles, we waited for crawls to complete, then went through our data to anonymize user IDs and strip any private data from our dataset to protect user privacy. In addition, all user IDs were hashed to random IDs, and all timestamps are replaced with relative sequence numbers. We note that our group has visited and held research meetings with technical teams at Renren, and they are aware of our ongoing research.

Complete Interaction Records. In November 2010, we contacted the provider of the Renren service and were given the anonymized information of 151,672 users in the PKU network. This data includes each user's popularity score, as well as complete records of diary entries, photos, and status updates. Table I shows the useful information associated with each piece of user data, including number of unique visitors, comments from the data owner, and comments from other users. Length refers to the number of bytes of text in diary entries and status updates. "Shares" refers to the number of times users have posted links to the data object in friends' news feeds. We use this additional

interaction data to analyze factors influencing latent interactions in Section 4. This dataset does not include the join-date of PKU users or timestamps of interactions (for privacy reasons).

Dynamic Interaction Records. In December 2011, we contacted Renren again and obtained the anonymized interactions and profile visits for the 61,405 PKU users from September 2009 to August 2010. These interaction records are the most complete dataset in our corpus: they include instant messages, message board posts, diary entries, photos, and status updates. Each interaction and profile visit includes a sender, a receiver, and a timestamp. In total, 532,326 interactions and 11,875,247 visits were given to us. We use this data to analyze time-varying interaction patterns in Section 5.4.

2.3. Measuring Latent User Interactions

In addition to active interactions generated by users in the PKU network, we also recorded the recent visitor records displayed on each user's profile. This data forms the basis of our study of latent interactions.

Reconstructing Visitor Histories. Crawling Renren for recent visitor records is complicated by two things. First, each user's profile only lists the last nine visitors. This means that our crawler must be constantly revisiting users in order to glean representative data, as new visitors will cause older visitors to fall off the list. Clearly we could not crawl every user continuously. Frequent crawls leave the ID of our crawler on the visitor log of profiles, which has generated unhappy feedback from profile owners. In addition, Renren imposes multiple rate limits on crawlers: first, each crawler account is only allowed to visit one profile per minute; second, each crawler account must solve a CAPTCHA if it visits 100 profiles in a short time. Otherwise, the crawler account is forbidden from viewing profiles for 2.5 hours. These rate limits slow our crawler significantly, despite our large number of crawler accounts. Thus, we designed our crawler to be self-adapting. This means that we track the popularity and level of dynamics in different user profiles and allocate most of our requests to heavily trafficked user profiles, while guaranteeing a minimum crawl rate (1/day) for low-traffic users. The individual lists from each crawl contain overlapping results which we integrate into a single history.

The second challenge of crawling recent visitor records is that each visitor is only shown in the list once, even if they visit multiple times. Repeat visits simply cause that user to return to the top of the list, erasing their old position. This makes identifying overlapping sets of visitors from the iterative crawls difficult.

To solve these two challenges, we use a log-integration algorithm to concatenate the individual recent visitor lists observed during each successive crawl. More specifically, some overlapping sets of visitors exist in successive crawl data, and our main task is to find new visitors and remove overlaps. There are two kinds of incoming visitors: new users who do not appear in the previous list, and repeat users who appear in the prior list at a different relative position. The first kind of incoming visitor is easily identified, since his record is completely new to the recent visitor list. New visitors provide a useful checkpoint for purposes of log-integration, since other users behind them in the list are also necessarily new incoming visitors. The second type of incoming visitor, repeat users, can be detected by looking for changes in sequence of the recent visitor list. If a user repeatedly visits the same profile in between two visits of other users, nothing changes in the recent visitor list. Therefore, consecutive repeat visits are ignored by our crawler.

Figure 3 demonstrates our integration algorithm. We observe that visitors `ABCDEFGHI` viewed a user's profile at some time before our first crawl. New users view the profile and are added to the recent visitor list by the second crawl at Times 2. We re-observe

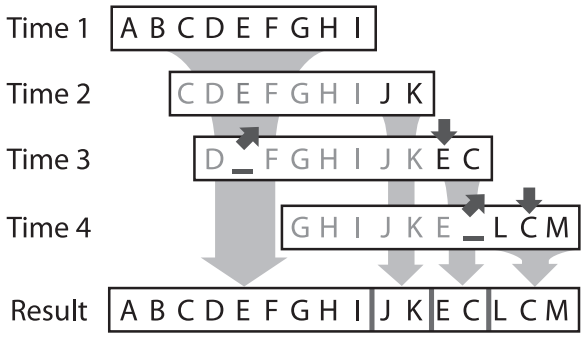


Fig. 3. Integrating multiple visitor lists captured by multiple crawls of the same profile into a single history.

the old sequence CDEFGHI and identify JK as new visitors, since JK do not exist in the previous visitor list. Next, we compare recent visitor lists at Times 2 and 3. We find that E is before K in the recent visitor list crawled at Time 2, but this order is changed at Time 3. This means that at some time before the third crawl, user E revisited the target and changed positions in the list. Thus we identify E as a new visitor. Since C is behind E at Time 3, C is also identified as a new visitor. Our integration algorithm also works correctly at Time 4. User L has not been observed before, and thus L, plus subsequent visitors C and M, are all classified as new visitors.

Overall, from the 61,405 user profiles we continuously crawled, we obtained a total of 8,034,664 total records of visits to user profiles in the PKU network. After integrating these raw results, we are left with 1,863,168 unique profile visit events. This high reduction (77%) is because most profiles receive few page views, thus overlaps between successively crawled results are very high. Although Renren does not show individual recent visitors of user diaries and photos, it does display the total number of visits, which we crawled as well.

Impact of Crawl Frequency. We are concerned that our crawls might not be frequent enough to capture all visit events to a given profile. To address this concern, we took a closer look at the impact of crawler frequency on missing visits. First, we take all of the profiles we crawled for visit histories and computed their average daily visit count between September and November 2009. We plot this as a CDF in Figure 4. Most users (99.3%) receive <=8 visits per day on average. Since Renren shows the nine latest visitors, crawling a profile once every day should be sufficient to capture all visits. While our crawler adapts to allocate more crawl requests to popular, frequently visited profiles, we guarantee that every profile is crawled at least once every 24 hours.

Next, we select 1,000 random PKU users and crawl their recent visitors every 15 minutes for two days. We use the data collected to simulate five frequencies for the crawling process: 15 minutes, 30 minutes, 1 hour, 12 hours, and 1 day. Then we use the log-integration algorithm to concatenate the individual recent visitor lists at different crawling frequencies. For every person, we compute the number of visits missed by the crawler when we reduce the frequency, beginning with visits every 15 minutes. We plot the CDF of these deviations in Figure 5. For 86% of users, there are no visits missed when we reduce the crawler rate from once every 15 minutes to once per day. The remaining 14% of users require more than one crawl per day to collect a full history of their visits.

Based on these observations, we engineered our crawler to allocate the bulk of crawl requests to high-popularity users. For each PKU user, the crawler determines how

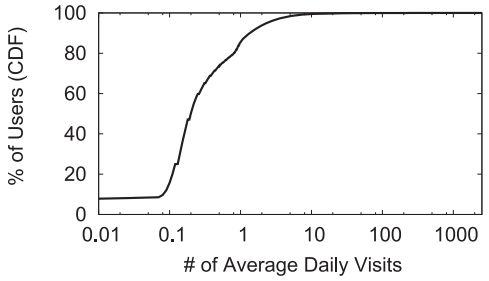


Fig. 4. Average daily visit counts of user profiles.

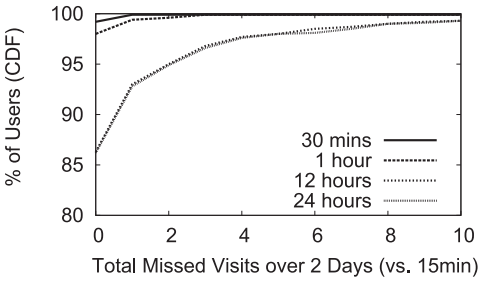


Fig. 5. Number of visits missed when we lower crawler frequency from a high of once every 15 minutes.

many times the user will be visited tomorrow by calculating $\lfloor v/9 \rfloor + 1$, where v is the number of times the user's profile was visited today. This formula ensures that all users are visited at least once, and users who are visited $\geq$9 times are crawled in proportion to their historical popularity.

2.4. Limitations

In the following sections, we quantify the graph structural properties of the Renren social graph and show that it is very similar to other large OSNs like Facebook and Orkut. However, in later sections when we analyze latent browsing behavior, it is more difficult to directly compare results to other OSNs. Although latent interactions have been studied before [Benevenuto et al. 2009; Schneider et al. 2009], the datasets used in these studies are not publicly available. It is possible that cultural and political issues in China affect the social behavior of Renren users. Therefore, we caution that latent interaction results from Renren may not generalize to all OSNs.

Furthermore, it is unknown how strongly the browsing behavior of Renren users is affected by the fact that browsing information is publicly visible. On one hand, it is possible that Renren users may browse more conservatively than users on other OSNs (e.g., Facebook) because they want to avoid the appearance of being a "stalker." However, as we show in Section 3.2, there are a significant number of nonfriend strangers that browse profiles, which indicates that users are not inhibited by social norms when they browse profiles on Renren.

On the other hand, users on Renren may browse more frequently than users on other OSNs because the visibility of latent interactions makes them a useful social signal. For example, a user could demonstrate closeness or concern for a friend by visiting their profile regularly. However, in Section 3.4, we find that latent interactions on Renren are not usually reciprocated, which indicates that users do not view latent interactions as strong social signals. Contrast this to active interactions, which are usually reciprocated due to social norms that dictate how to conduct polite conversation.

The News-Feed. When users log in to Renren, they are greeted with a news feed that displays a list of all their friends' recent activity. Clearly, the news feed reduces the number of latent interactions on Renren, since users no longer have to visit their friends' profiles to catch up on recent activity. However, all modern OSNs implement news feed functionality, including Facebook, Twitter, LinkedIn, Google+, etc. Thus, although the news feed reduces latent interactions, results derived from our Renren data should be consistent with latent interactions on other OSNs with respect to the impact of the news feed.

Table II. Topology Properties of OSNs

Network	Users Crawled	Links	Avg. Degree	C. Coef.	Assort.	Avg. Path Len.
Renren	42,115K	1,657,273K	78.70	0.063	0.15	5.38
Facebook[1]	10,697K	408,265K	76.33	0.164	0.17	4.8
Cyworld[2]	12,048K	190,589K	31.64	0.16	−0.13	3.2
Orkut[3]	3,072K	223,534K	145.53	0.171	0.072	4.25
Twitter[4]	88K	829K	18.84	0.106	0.59	N/A

Note: OSN data from [1][Wilson et al. 2009], [2][Ahn et al. 2007], [3][Mislove et al. 2007], and [4][Java et al. 2007].

2.5. Social Graph Analysis

In this section, we analyze the topological properties of the entire Renren social graph by focusing on salient graph measures. Table II shows some general properties of Renren, such as average degree, clustering coefficient, assortativity, and average path length, as compared to other social networks. Our Renren dataset is larger than most previously studied OSN datasets, the exceptions being recent measurements of the Twitter network [Cha et al. 2010; Kwak et al. 2010]. However, as shown in Table II, our dataset shares similar properties with prior studies [Ahn et al. 2007; Mislove et al. 2007; Wilson et al. 2009]. This confirms that Renren is a representative social network and that the behavior of its users is likely to be indicative of users in other OSNs like Facebook.

Degree Distribution. Figure 6 plots the complementary cumulative distribution function (CCDF) of social degrees on Renren. The bump in the distribution at degree 1,000 is due to the maximum friend limit on Renren: only users who pay may have >1,000 friends, and such users are rare. To obtain a power-law fit for the Renren degree distribution, we used the method from Clauset et al. [2007], as well as the slightly modified method used in Mislove et al. [2007]. In both cases, the fitting error was unacceptably high, even if the users with degree >1,000 were filtered out. Thus, we believe that the Renren degree distribution does not exhibit power-law scaling.

The original version of this article reported a power-law exponent of 3.5 for Renren, but this result is erroneous [Jiang et al. 2010]. The power-law fitting code from Clauset et al. [2007] has hard coded limits that restrict it to calculating exponents in the range [1.5, 3.5]. Originally, we were unaware of this restriction. Once we adjusted these limits, it became clear that the earlier result was due to the bounds of script, rather than the intrinsic characteristics of the Renren data.

Clustering Coefficient. Clustering coefficient quantifies the level of local connectivity between nodes in a graph. In undirected graphs, the clustering coefficient of a person is defined as the ratio of the number of links over all possible connections between one's friends. The clustering coefficient of the entire network is defined by the average of all individual clustering coefficients. Renren's average clustering coefficient is only 0.063, demonstrating that Renren friend relationships are more loosely connected than the other social networks studied in Table II (e.g., Renren users have many friends that are not mutual friends themselves).

Figure 7 displays the distribution of clustering coefficient versus node degree. As expected for a social network, users with lower social degrees have higher clustering coefficients, thus demonstrating high levels of clustering at the edge of the social graph.

Although Renren has lower average clustering than Facebook, this fact is unlikely to have significant impact on our later analysis. Comparing the clustering coefficient distribution of Renren to that of Facebook [Wilson et al. 2009] reveals that users with degree <100 have significantly higher clustering on Renren than on Facebook. Conversely, users with degree ≥100 have lower clustering on Renren than on Facebook.

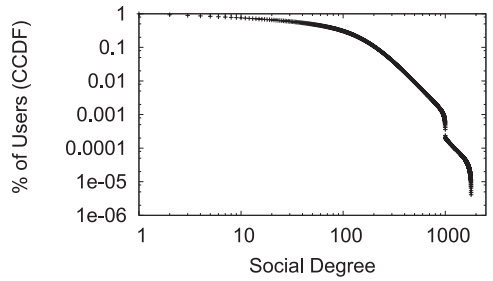


Fig. 6. Node degree distribution in the Renren network.

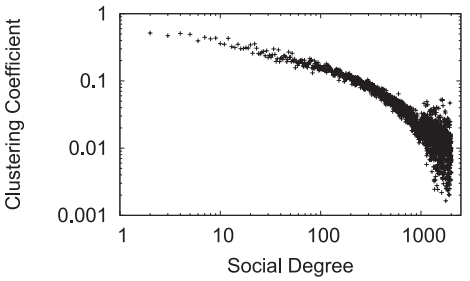


Fig. 7. Renren clustering coefficient distribution.

This comparison shows that Renren's low average clustering is due entirely to high-degree users, not the tightly-clustered fringe. Since ≈70% of Renren users have degree <100, a significant majority of users exhibit strong clustering.

Assortativity. The assortativity coefficient measures the probability of users establishing links to other users of similar degree [Wilson et al. 2009]. It is calculated as the Pearson correlation coefficient of the degrees of node pairs for all links in a graph. A positive assortativity coefficient indicates that users tend to connect to other users of similar degree, and a negative value indicates the opposite trend. Renren's assortativity is 0.15, implying that connections between like-degree users are numerous. Similarly, Facebook's assortativity is 0.17 [Wilson et al. 2009].

k_{nn}. Figure 8 displays node degree correlation (k_{nn}) versus node degree. k_{nn} is a closely related metric to assortativity. The positive correlation starting around degree 100 demonstrates that higher-degree users tend to establish links with other high-degree users. More specifically, the Pearson correlation coefficient between degree and k_{nn} is 0.61 when the degree is bigger than 100. These chains of well-connected superusers form the backbone of the social network.

Average Path Length. Average path length is the average of all-pairs-shortest-paths in the social network. It is simply not tractable to compute shortest path for all node pairs, given the immense size of our social graph. Instead, we choose 1,000 random users in the network, perform Dijkstra to build a spanning tree for each user in the social graph, and compute the length of their shortest paths to all other users in the network. As shown in Table II, the average path length in Renren is 5.38, which agrees with the six degrees of separation hypothesis [Milgram 1967]. Average path length on Renren is similar to prior results from Facebook [Wilson et al. 2009], Cyworld [Ahn et al. 2007], and Orkut [Mislove et al. 2007].

Strongly Connected Component. Users' online friendship links often correspond closely with their offline relationships [Lampe et al. 2007]. Thus, it is natural to assume that college students would have many online friends in the same campus network. This behavior should manifest itself as a single, large, strongly connected component (SCC) that includes most users in the PKU network social graph. Surprising, we find that 23,430 (23.2%) of users in the PKU network have no friends in the PKU campus network and are therefore disconnected from the SCC. We refer to these as *isolated users*. To confirm these results, we measured the SCC of nine other large university networks and discovered similar numbers of isolated users.

Figure 9 shows social degrees and total number of profile visits for these isolated users. 83% of isolated users have social degrees fewer than ten. In addition, 70% of isolated users have fewer than 20 total profile visits, meaning their profiles are rarely

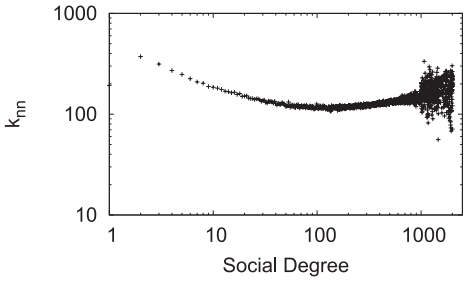


Fig. 8. Renren k_{nn} distribution.

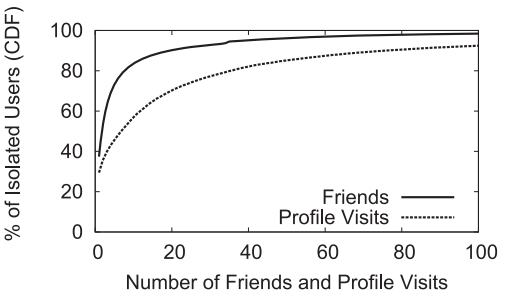


Fig. 9. Social degree and profile views for isolated users.

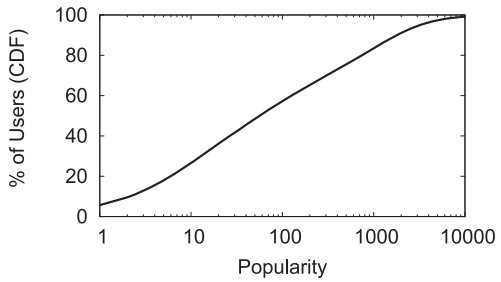


Fig. 10. CDF of user profile popularity defined by visits.

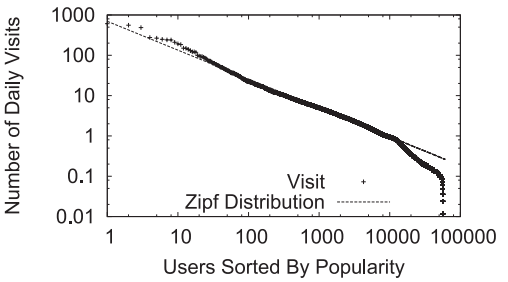


Fig. 11. Average number of visits per day per user.

browsed by others. Although it is not clear why isolated users do not have friends within the network, the vast majority of these users are both unpopular and low degree. Therefore, they have little impact on our overall results.

3. PROPERTIES OF INTERACTION EVENTS

Our work focuses on the analysis of latent interaction events and the role they play in OSNs. In our measurement of the Renren OSN, we use histories of visits to user profiles to capture latent interactions. In this section, we take a closer look at latent interactions and compare them with active interactions from a variety of perspectives.

3.1. Popularity and Consumption

We begin by analyzing the distribution of latent interactions across the Renren user base. Recall that we define popularity as the number of views a user's profile receives. Figure 10 shows the distribution of user popularity. As expected, popularity is not evenly spread across the population: only 518 people (1%) are popular enough to receive more than 10,000 views. Conversely, the majority of users (57%) exhibit very low popularity with fewer than 100 total profile views. Some users seldom publish any content to attract profile visits.

Figure 11 shows the average number of visits users receive on a daily basis. The distribution is fitted to a Zipf distribution of the form $\beta x^{-\alpha}$, where $\alpha = 0.71569687$ and $\beta = 697.4468225$. Popular users receive many more views per day: 141 users (0.2%) are viewed more than 20 times a day on average, with the most popular profile being viewed more than 600 times a day. Most users (85.5%) receive less than one visit per day on average. This reinforces our finding that latent interactions are highly skewed towards a very popular subset of the population.

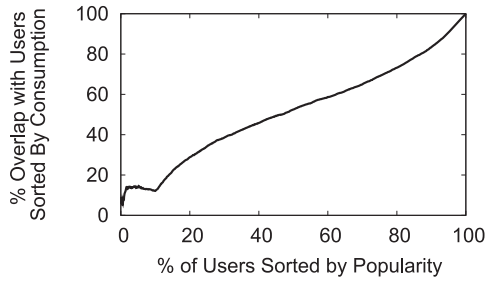


Fig. 12. Overlap between users sorted by popularity vs. sorted by consumption.

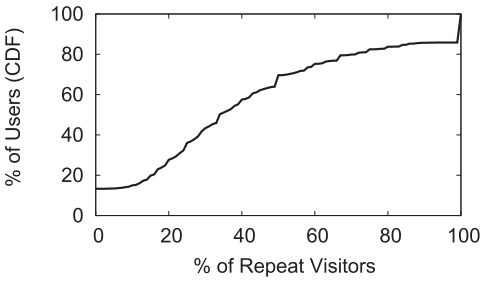


Fig. 13. Ratio of repeat visitors.

Finally, we examine whether the popularity of users corresponds to their profile viewing behavior. We define *consumption* as the number of other profiles a user views. Figure 12 plots the overlap between the top users sorted by popularity and top users sorted by consumption. The graph shows that the top 1% most popular users have 9% overlap with the top 1% biggest consumers. These users represent a hardcore contingent of social network users who are extremely active. For the most part however, users with high numbers of incoming latent interactions do not overlap with the people generating those interactions, for example, profiles of celebrities are viewed by many users, but they are inactive in viewing others' pages. This necessarily means that many (presumably average, low-degree) users actively visit others but are not visited in return. We examine the reciprocity of latent interactions in more detail in Section 3.4.

3.2. Composition of Visitors

Next, we want to figure out the composition of visitors to user profiles. We pose two questions: first, what portion of profile visitors are repeat visitors? Second, are visitors mostly friends of the profile owner, or are they unrelated strangers?

We begin by addressing the first question. We calculate the percentage of repeated visitors for each profile and report the distribution in Figure 13. Roughly 70% of users have fewer than 50% repeat visitors, meaning that the majority of visitors do not browse the same profile twice. This seems to indicate that the long tail of latent interactions is generated by users randomly browsing the social graph.

Next, we take a closer look at repeat profile visits. Figure 14 shows the probability density function (PDF) of the interval time between repeat visits. The graph peaks on day 0, meaning that users are most likely to return to a viewed profile on the same day. We will examine the causes for this behavior more closely in Section 4. The probability for repeated views decreases as the time delta expands, except for a noticeable peak at day 7. Interestingly, this shows that many users periodically check on their friends on a weekly basis. We confirmed that this feature is not an artifact introduced by our crawler or the use of RSS feeds by Renren users. Instead, we believe it may be due to the tendency for many users to browse their friends' profiles over the weekend.

We now move on to our second question: which users are generating latent interactions—friends of the profile owner or strangers. We define a stranger as any user who is not a direct friend of the target user. Renren's default privacy settings allow users in the same campus network to browse each other's profiles.

In order to answer our question, we calculate the percentage of visitors that are strangers and display the results in Figure 15. The results are fairly evenly divided: roughly 45% of users receive fewer than 50% of their profile visits from strangers. Or conversely, a slight majority of the population does receive a majority of their profile views from strangers.

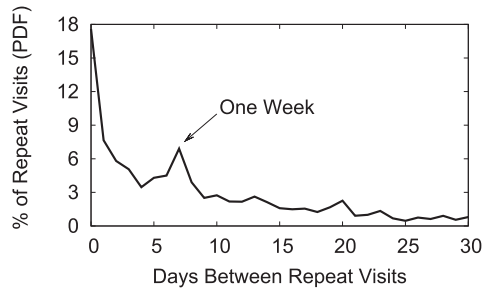


Fig. 14. PDF of interval time between repeat visits.

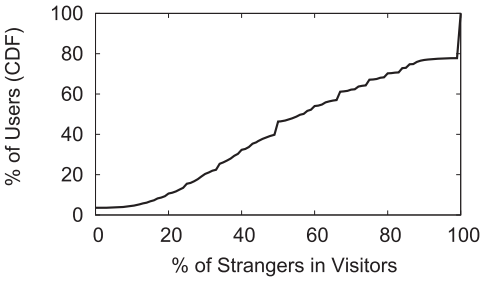


Fig. 15. Percentage of strangers in visitors.

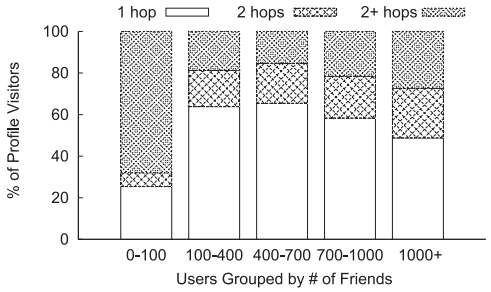


Fig. 16. Breakdown of visitors by owner's social degree.

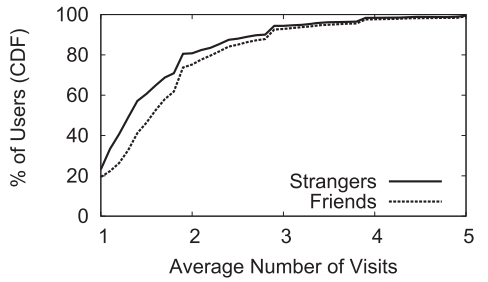


Fig. 17. Average number of visits for friends and strangers.

We want to take a closer look at what component of a profile's visitors are strangers, and how far are they from the profile owner in the social graph. In Figure 16, we group the owners of profiles together by their social degree and compute the average breakdown of their visitors into users who are friends (1-hop), friends of friends (2-hop), and other visitors (2+ hops). We see that for users with relatively few (<100) friends, the large majority of their visitors are complete strangers, with very few friends of friends visiting. For well-connected users with 100–1,000 friends, the majority of their visitors are direct friends and also a significant number of friends of friends. Finally, for extremely popular users with more than 1,000 friends, their notoriety is such that they start to attract more strangers to visit their profiles. These results confirm those from previous work that discovered many Orkut users browse profiles two or more hops away on the social graph [Benevenuto et al. 2009].

Unlike friends, strangers do not build long-term relationships with profile owners. Intuitively, this would seem to indicate that repeat profile viewing behavior should favor friends over strangers. To investigate this, we compute the average number of visits for strangers and friends for each profile and plot the distribution in Figure 17. Surprisingly, our results indicate that the repeat profile viewing behavior for friends and strangers is very similar, with friends only edging out strangers by a small margin. This result demonstrates that when considering information dissemination via latent interactions, the significance of nonfriend strangers should not be overlooked.

3.3. Visits to Strangers' Profiles

In Section 3.2, we observe that strangers account for a significant portion of profile visits. In this section, we examine the question how do people find and view nonfriends' profiles? There are several possible mechanisms that enable this behavior on Renren.

—*Featuring.* Renren automatically recommends the 100 most popular profiles in each network to other users in the same network.
—*Search.* Users may search for specific people within Renren. Personal attributes, such as name and university, are used by the Renren search engine to locate relevant users.
—*Social Links.* Users may visit a friends-of-friends' profile after seeing a link to it while browsing another profile. This is possible because Renren shows 24 random friends on each user's profile, along with wall posts and comments that also originate from friends. Each user's full list of friends is also accessible from their profile.

In this section, we focus on visits to strangers' profiles via social links. This analysis is possible because our crawled data includes each user's full friend list, as well as the approximate timestamp of all latent interactions. Because our dataset does not include click through statistics from Renren's featured links or search functionality, we are unable to directly examine the effects of these features on browsing behavior. However, the fact that visits from strangers are not confined to the top 100 most popular users in the PKU network indicates that profile featuring is not the primary driver behind stranger browsing behavior.

We use a time-based heuristic to infer when a Renren user is likely to have visited a stranger's profile via social links. The behavior we are looking for has the following structure.

—A views B's profile before A views C's profile.
—B and C are friends.
—A may or may not be friends with B and C.

Intuitively, the inference we draw from this situation is that A visits C's profile through a link on B's profile. Although without asking people directly we cannot say for sure that this is what happened, if the time between A's visits to B and C is small, then social links are the likely path a browser would have followed, especially if A and C are not friends.

In order to examine nonfriend profile browsing behavior, we create an ordered list of profiles visited by each member of the PKU network. Profile visits from non-PKU users are filtered out since we do not have complete access to those users' information. Each user's list of visits is ordered chronologically. Although Renren does not provide the exact timestamp for each profile visit, approximate timestamps can be inferred based on when the visit was first observed by the crawler. New visits to a given profile recorded by the crawler must have occurred in the time interval since the crawler previously visited the profile. We use these time intervals as approximate timestamps when chronologically ordering profile visit events. Returning to our example scenario: if the earliest possible time of A's visit to B is smaller than the latest possible time of A's visit to C, we assume that A views B's profile before A views C's profile. As a sanity check, we only infer correlation if the time delta between events is 24 hours or less, since this is the interval between periodic crawls used for most of the Renren population.

Using this approximate ordering methodology, we isolate views of stranger's profiles that immediately follow a visit to a friend of that stranger. Figure 18 shows the number of visits to strangers' profiles that fit our criteria, as a fraction of all visits to stranger's profiles. 62% of users do not visit any strangers' profiles via social links, indicating that they use other Renren functionality (search, featured profiles) when browsing. Conversely, 4% of users only visit strangers' profiles via social links.

Figure 18 also shows the percentage of visits to stranger's profiles that traverse through a mutual friend. This represents directly browsing from a friend's profile to a

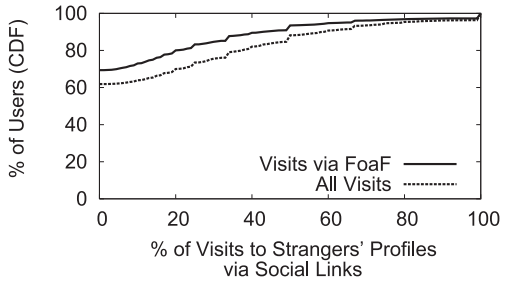


Fig. 18. Percentage of visits to strangers' profiles through social links, as a fraction of total visits to stranger's profiles.

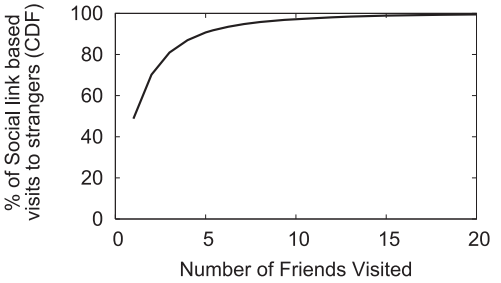


Fig. 19. Number of friends browsed before visiting a stranger's profile.

friend of a friend (FoaF). Browsing to FoaF profiles accounts for the majority of visits to stranger's profiles via social links. This indicates that when Renren users surf around profiles, they do not stray far from their immediate social circle, that is, users are likely to visit stranger's profiles if they share a mutual friend.

Figure 19 explores the number of friends a user visits before viewing a stranger's profile. 49% of visits to FoaF strangers occur after visiting only a single friend, indicating that most users browse directly from one-hop to two-hop neighbors. However, 10% of stranger browsing occurs after visiting $\geq$5 mutual friends. This captures cases where a user browses many friends and notices that they all share a mutual friend that the user herself is not friends with. These cases may indicate locations where edges are missing from the graph, or where the presence of high clustering leads to the creation of new social connections.

3.4. Reciprocity

Social norms compel users to reply to one another when contacted via active interactions. Prior work has shown that these interactions are largely reciprocal on OSNs [Wilson et al. 2009]. However, is this true of latent interactions? Since Renren users have full access to the list of recent visitors to their profile, it is possible for people to pay return visits to browse the profiles of their visitors. The question is, does visiting other user profiles actually trigger reciprocal visits?

As the first step towards looking at reciprocity of latent interactions, we construct the set of visitors who view each user profile and the set of people who are visited by each user. Then, we compute the intersection and union of these two sets for every user. Intuitively, intersections include people who view a given user profile and are also visited by that user, that is, the latent interactions are reciprocated. Unions contain all latent relationships for a given user, that is, all users who viewed them or whom they viewed. We calculate the Jaccard index for each user using their intersection and union set, then plot the results in Figure 20. The ratio represents the number of reciprocated latent interactions divided by the total number of latent relationships. For more than 93% of users, fewer than 10% of latent relationships are reciprocated. This demonstrates that incoming profile views have little influence on users' profile browsing behavior. This is surprising, especially considering the fact that users know that their visits to a profile are visible to its owner through the visitor history feature.

Next, we examine the time-varying characteristics of reciprocal profile visits for both strangers and friends. We compute the number of reciprocal visits that take place within t days after the initial visit. Figure 21 shows the results for threshold t values of one and five days plus the entire 90 days. As we look at increasingly larger window sizes, we see more profile visits being reciprocated. However, reciprocity remains low overall.

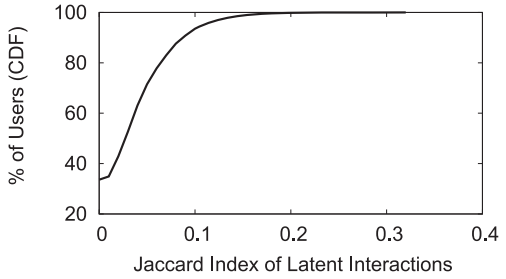


Fig. 20. Ratio of reciprocated latent interactions over total latent relationships.

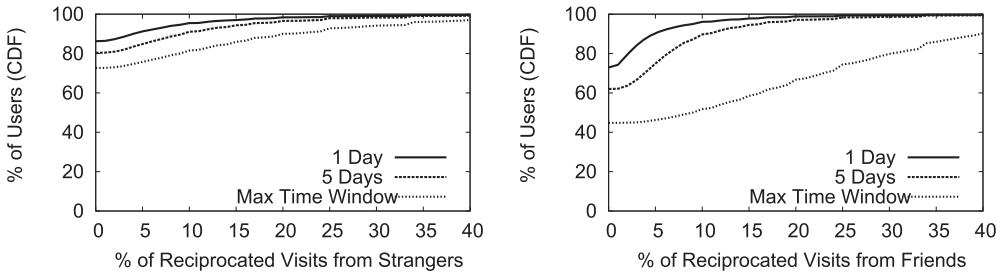


Fig. 21. Probability of reciprocated profile views over various time windows for both strangers and friends.

Even across the entire measurement period, 73% of users receive no reciprocal page views from strangers, and 45% of users obtain no reciprocal page views from friends. This demonstrates that even with Renren's visitor history feature, visiting other user profiles is not sufficient to generate reciprocal visits. Compared to strangers, friends have relatively higher probabilities of reciprocal visits.

We take a further step and quantify the lack of reciprocity for latent interactions. For a data set of n users, if user i visits user j, then $v_{ij} = 1$; otherwise $v_{ij} = 0$. The reciprocity coefficient [Chun et al. 2008] is defined as $\frac{\sum_{i \neq j}(v_{ij}-\bar{v})(v_{ji}-\bar{v})}{\sum_{i \neq j}(v_{ij}-\bar{v})^2}$, where $\bar{v} = \frac{\sum_{i \neq j} v_{ij}}{n(n-1)}$. The reciprocity coefficient is measured between -1 and 1, where positive values indicate reciprocity, and negative values anti-reciprocity. The reciprocity coefficient of profile visits on Renren is only 0.23. In contrast, reciprocity of active comments on Renren is 0.49, and the reciprocity of active interactions on Cyworld [Chun et al. 2008] is 0.78. Compared to these active interactions, latent interactions show much less reciprocity.

3.5. Latent vs. Active Interactions

In this section, we compare the characteristics of latent and active interactions. To understand the level of participation of different users (e.g., highly interactive users vs. more passive users) in both latent and active interactions, Figure 22 plots the contribution of different users to both kinds of interactions. The bulk of all active interactions can be attributed to a very small, highly interactive portion of the user base: the top 28% of users account for all such interactions. In contrast, latent interactions are more prevalent across the entire population, with more than 93% of all users contributing to latent interaction events. This confirms our original hypothesis that users view more profiles than leave comments. Given its widespread nature, this result also underscores the importance of understanding latent interactions as a way of propagating information across OSNs.

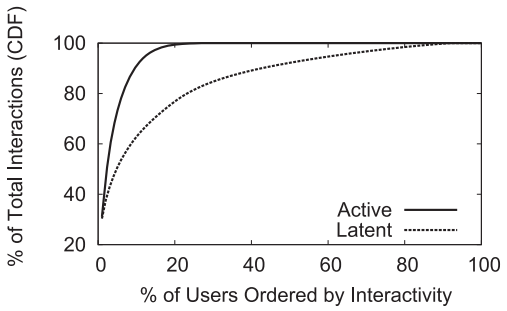


Fig. 22. Distribution of interactions, with users ordered from most to least interactive.

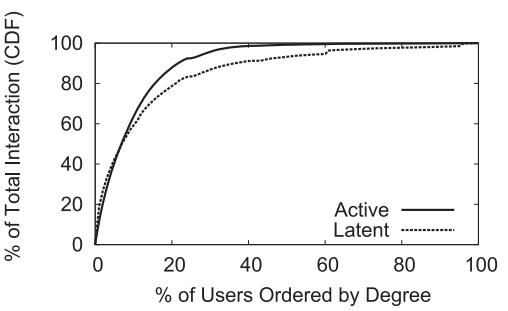


Fig. 23. Distribution of interactions, with users ordered by social degree.

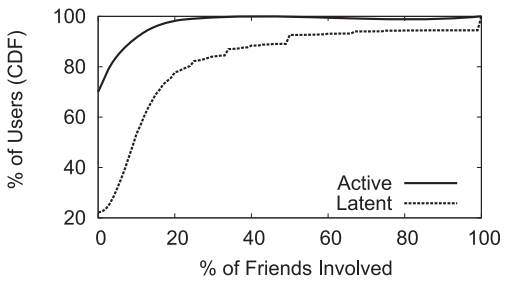


Fig. 24. Distribution of interactions among each user's friends.

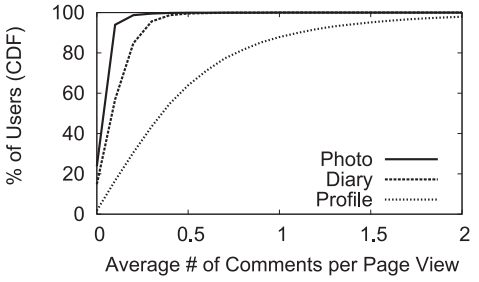


Fig. 25. Average number of comments per page view for different types of pages.

Figure 23 shows the distribution of interactions among users when ordered by degree. In contrast to Figure 22, active interactions are not as tightly concentrated amongst high-degree users. Instead, the weak coupling between social degree and interactivity spreads the active interactions more evenly throughout the population. However, the correlation between degree and active interactivity is still greater than that between degree and latent interactivity. Latent interactions have a significantly longer tail than active interactions on Renren.

Next, we compare latent and active interactions in coverage of friends. We compute for each user a distribution of their latent and active interactions across their social links. We then aggregate across all users the percentage of friends involved in these events and plot the results in Figure 24. We see that roughly 80% of users only interact visibly with 5% of their friends, and no users interact with more than 40% of their friends. In contrast, about 80% of users view 20% or more of their friends' profiles, and a small portion of the population views all of their friends' profiles regularly. Thus, although not all social links are equally active, latent interactions cover a wider range of friends than active interactions.

To get a sense of how many active comments are generated by latent interactions, we examine the average number of comments per page view for a variety of pages on Renren, including profiles, diary entries, and photos. Figure 25 plots the results. Recall that along with active comments, Renren keeps a visitor counter for each photo and diary entry. For diary entries and photos, the conversion rate is very low: 99% of users have less than one comment for every five photo views, and 85% people have less than one comment for every five diary views. This indicates that most users are passive information consumers: they view/read content and then move on without commenting themselves. In contrast, profile views have a higher conversion rate. Interestingly, 13%

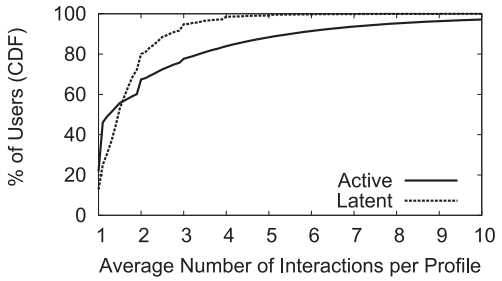


Fig. 26. Average number of interactions per profile.

of users have a view/comment ratio greater than one. This is because these users use profile comments as a form of instant messaging chat, leaving multiple responses and replies upon each visit.

Finally, we analyze the repeat activity frequency for latent and active interactions on Renren. In particular, we want to examine the likelihood that users will repeatedly interact with the same page once they have viewed or commented on it once. Figure 26 plots the average number of interactions each user has with profile pages. 80% of users view a given profile < 2 times. However, 80% of users leave 3.4 comments, almost twice the number of latent interactions. This result makes sense intuitively: for most types of data, users only need to view them once to consume the data. However, comments can stimulate flurries of dialog on a given page, resulting in many consecutive interactions.

4. FACTORS INFLUENCING LATENT INTERACTIONS

As shown in Section 3.1, not all users in Renren are the target of equal numbers of latent interactions. Put another way, not all users have equally popular profiles. Although the popularity of some Renren users can be explained by their real-world celebrity status, this is not true for all popular users on Renren. This leads to the following question: what factors cause certain profiles to receive more latent interactions and become popular?

In this section, we analyze factors that may encourage users to visit Renren profiles. Quantifying how the actions of a profile owner impact the number of views received by that profile is an important step in understanding how OSN accounts become popular. If there are strong correlations between popularity and particular user actions (e.g., posting photos, writing diary entries, etc.), then this provides a roadmap for individuals looking to accrue popularity and promote themselves via social media. On the other hand, if there are no correlations between user actions and popularity, then this would reveal that there is no simple formula for a user to gain popularity on social media.

We examine the following factors and correlate them with profile popularity (i.e., number of received latent interactions).

—*Number of Friends.* Does social degree correlate with popularity?
—*Lifetime.* Are long-lived accounts more likely to be popular than newer, less active accounts?
—*Shared Links.* Do users attract more visits if they frequently share links to other content?
—*Diary (Blog) Entries.* Are there correlations between diary update frequency or length and user popularity? Do diary entries generated by popular users receive more views and comments than those generated by less popular users?
—*Photos.* Does the popularity of a user's photos correlate with their popularity?

Table III. Number of Renren Users in Each Popularity Group

Popularity Groups	Group Sizes for Table IV	Group Sizes for Tables V, VI, and VII
0–100	35,154	104,254
100–1,000	16,047	27,651
1,000–10,000	9,686	18,468
> 10,000	518	1,299
Total:	61,405	151,672

Table IV. Average Value of Factors Associated with User Popularity

Popularity	Friend	Lifetime	Shared Links
0–100	16 (0.15)	35 (0.55)	1 (0.5)
100–1,000	131 (0.56)	423 (0.41)	43 (0.41)
1,000–10,000	401 (0.43)	792 (0.24)	155 (0.23)
>10,000	708 (0.02)	869 (0.02)	273 (−0.05)
All Users	112 (0.73)	263 (0.75)	39 (0.72)

Note: Spearman's ρ is shown in parentheses.

Table V. Diary's Factors Associated with User Popularity

Popularity	Amount	Visitors	Shared Links	Length	Owner's Comments	Others' Comments
0-100	1 (0.53)	10 (0.54)	1 (0.51)	826 (0.53)	1 (0.52)	1 (0.52)
100-1,000	7 (0.43)	220 (0.49)	4 (0.48)	17308 (0.42)	9 (0.51)	19 (0.52)
1,000-10,000	49 (0.35)	3759 (0.52)	59 (0.33)	83157 (0.31)	151 (0.44)	292 (0.46)
>10,000	142 (0)	35274 (0.17)	809 (0.08)	273399 (0)	668 (−0.01)	1347 (0.03)
All Users	9 (0.72)	807 (0.74)	15 (0.64)	16190 (0.72)	26 (0.7)	51 (0.72)

Note: Spearman's ρ is shown in parentheses.

—*Status Updates*. Does the quantity and length of user's status updates correlate with their popularity? Are status updates from popular people more likely to receive comments from others?

4.1. Methodology

In order to examine the correlations between each factor and popularity, we divide users into four groups based on their popularity. Table III shows the popularity score ranges for each group as well as the number of Renren users in each group. The analysis in Table IV uses the data from our original crawl of the PKU graph. The number of users grows for the analysis in Tables V, VI, and VII because they are based on the more complete dataset that Renren gave us in November 2010.

For each popularity group, we calculate the average value of each factor and display the results in Tables IV, V, VI, and VII. All factors increase along with popularity, that is, the most popular users also have the most friends, the oldest accounts, and generate the largest amounts of content/active interactions.

Given the size differences between popularity groups and the average nature of the values in Tables IV, V, VI, and VII, it is difficult to infer definite correlations between any one factor and popularity. To analyze these correlations more specifically, we leverage a technique from prior work [Cha et al. 2010] called Spearman's rank correlation coefficient (Spearman's ρ). Spearman's ρ is a nonparametric measure of the correlation between two variables that is closely related to Pearson's correlation coefficient [Lehmann and D'Abrera 1998]. It is defined as $\rho = 1 - \frac{6\sum(x_i - y_i)^2}{n(n^2-1)}$, where x_i and y_i are the ranks of two different features in a dataset of n users. $\rho > 0$ indicates

Table VI. Photo's Factors Associated with User Popularity

Popularity	Amount	Visitors	Owner's Comments	Others' Comments
0–100	2 (0.46)	8 (0.67)	1 (0.52)	1 (0.55)
100–1,000	38 (0.41)	822 (0.49)	9 (0.51)	18 (0.49)
1,000–10,000	205 (0.33)	12827 (0.51)	125 (0.45)	228 (0.48)
> 10,000	668 (0)	178094 (0.19)	575 (0.02)	1185 (0.09)
All Users	39 (0.73)	3242 (0.85)	22 (0.72)	41 (0.75)

Note: Spearman's ρ is shown in parentheses.

Table VII. Status's Factors Associated with User Popularity

Popularity	Amount	Length	Owner's Comments	Others' Comments
0–100	1 (0.53)	9 (0.53)	1 (0.52)	1 (0.52)
100–1,000	24 (0.46)	550 (0.45)	25 (0.49)	39 (0.49)
1,000–10,000	164 (0.35)	4078 (0.34)	283 (0.35)	451 (0.36)
> 10,000	420 (0)	11796 (0)	870 (−0.02)	1468 (0)
All Users	28 (0.74)	704 (0.74)	46 (0.7)	75 (0.72)

Note: Spearman's ρ is shown in parentheses.

positive correlation, while <0 indicates negative correlation. Spearman's ρ is shown in parenthesis beside the average value for each entry in Tables IV, V, VI, and VII.

The popularity correlation analysis conducted in the original version of this article was constrained to only examining scalar values that could be collected by our crawler from user profiles (e.g., number of friends, number of diary entries, number of photos, etc.) [Jiang et al. 2010]. In order to conduct a more comprehensive examination, we contacted Renren in November of 2010 and obtained complete anonymized information for all PKU network users, as detailed in Section 2.2. This new data (shown in Table I) enables us to perform additional correlation analysis on diary entries, photos, status updates, and user comments that were previously impossible.

4.2. User Account Characteristics

Table IV shows the average number of friends, account lifetime, and number of shared links for our four popularity groups. Shared Links refers to URLs shared by PKU users, not received from friends. Lifetime is measured as the number of days in between a user joining and leaving Renren. Neither of these pieces of information is provided by Renren, and thus must be estimated. Join date can be approximated by the timestamp of the first comment received by a user, since the comment is likely to be a welcome message from a friend greeting the new user [Wilson et al. 2009]. Because abandoned accounts can still receive comments, the best estimate of departure time is the timestamp of the last comment left by a user.

Although all factors in Table IV exhibit high correlation with the low-popularity and All Users categories, this is an artifact of the tied ranks among the (numerous) low-activity users. All of these users exhibit very low interactivity and social degree, thus leading to high levels of correlation. Previous work has observed similar artifacts when analyzing all users in a large OSN dataset [Cha et al. 2010].

For the two median-popularity groups (100–1,000 and 1,000–10,000) in Table IV, number of friends has the highest correlation with popularity. Users in these categories can be broadly defined as normal social network users. They are not celebrities; they simply use the OSN for its intended purpose of sharing information with friends. Account lifetime is a less important factor for users in the 1,000–10,000 popularity range, given the ease with which users can quickly amass hundreds of friends on OSNs.

Table VIII. Correlation between Factors of Diary and Photo

	Diary Entries				Photos	
	Shared Links	Length	Owner's Cmnts	Others' Cmnts	Owner's Cmnts	Others' Cmnts
Visitor	0.46	−0.02	0.51	0.68	0.55	0.59
Shared Links		0.5	0.42	0.41	N/A	N/A
Length			0.06	0	N/A	N/A
Owner's Cmnts				0.82		0.9

No factor has strong correlation with popularity for users in the high-popularity group in Table IV. This is an important finding, as it shows popularity is not trivially gained simply by having lots of friends.

4.3. Diary Entries

Table V shows the average value of various metrics associated with users' diary entries, as well as Spearman's ρ for each metric. The Amount column lists the number of diary entries per user, Visitors is the number of unique visitors to each diary entry, and Length is the number of characters in each diary entry. Shared Links lists the number of times users have shared a link to a diary entry with friends. The two comment columns are the number of comments each diary entry receives from the entry's owner and from other people. Intuitively, many of these metrics are intrinsically linked, for example, a diary that is shared many times is also likely to receive many visitors, which can also result in many comments.

For users with less than 10,000 popularity, all factors have high correlation with the popularity. However, for the high-popularity group only the counter of visitors has obvious correlation with user popularity. One explanation for this correlation is that when people view a diary, they are also likely to visit the owner's profile, thus boosting the user's popularity. For popular users, no correlation exists between popularity and the number of diary entries or their length. Thus, producing copious or expansive diary entries is not enough to attract profile visits.

4.4. Photos, Status Updates, and Comments

Tables VI and VII shows Spearman's ρ for users' photos and status updates. All columns are defined the same as for diary entries. Similar to diary, all factors show strong correlation with popularity in low- and median-popularity groups. Also similarly, only the visitor counter for photos has significant correlation for high-popularity users. Again, this demonstrates that popularity is not simply gained by producing copious amounts of user-generated content (photos or status updates, in this case).

Previous work observes that in Flickr, a photo's visitor counter does not have high correlation with the number of comments or shares associated with that photo. However, the number of comments is strongly correlated with the number of shares [Cha et al. 2009]. We perform similar cross-correlation on our Renren data for diary entries and photos and show the results in Table VIII.

In contrast to Flickr [Cha et al. 2009], the number of visitors has high correlation with the number of shared links and comments (both from the owner and other users) for diary entries and photos on Renren. This may be due to the more social nature of Renren as compared to Flickr, that is, all Renren users belong to the social network, and the average number of friends is high, versus Flickr, where not all users leverage the website's social capabilities. Similar to Flickr, shares on Renren positively correlate with comments.

Unsurprisingly, the highest correlations occur between comments from the owner and other people, stemming from the use of comment areas to hold bidirectional conversations. As shown in Table IX, this trend holds across all Renren features, over all popularity groups.

Table IX. Correlation between Owner's Comments and Others' Comments

Popularity	Diary	Photo	Status
0–100	0.98	0.89	0.99
100–1,000	0.93	0.89	0.96
1,000–10,000	0.97	0.96	0.99
>10,000	0.95	0.92	0.97
All Users	0.97	0.91	0.98

4.5. Limitations

There may be other factors outside the scope of our measurements that contribute to user popularity. One possibility is that real-world celebrity status is the most important determining factor of online popularity. Unfortunately, we cannot quantify these factors at present. Recall that 100 of the most popular users in the university network are recommended to users by Renren. These 100 users account for fewer than 7.7% of the total users in the high-popularity group, so the recommendation mechanism has limited impact on the high-popularity group results.

5. LATENT INTERACTION GRAPHS

Previous studies have demonstrated that taking active interactions into account has important implications for applications that leverage social graphs [Wilson et al. 2009]. These changes can be modeled by *interaction graphs*, which are constructed by connecting users from the social graph who have visibly interacted one or more times.

We have already demonstrated significant differences between latent and active interaction patterns on Renren. To summarize these key differences briefly, latent interactions are more numerous, nonreciprocal, and often connect nonfriend strangers. These results are also likely to have profound implications on applications that leverage social graphs and thus warrant the construction of a new model to capture the properties of latent interactions. We call this new model *latent interaction graphs*. In this section, we formally define latent interaction graphs, analyze their salient properties, and compare them to the Renren social and active interaction graphs.

5.1. Building Latent Interaction Graphs

A latent interaction graph is defined as a set of users (nodes) that are connected via edges representing latent interaction events between them. Unlike the social graph and active interaction graph, we have shown that latent interaction is nonreciprocal (Section 3.4). Thus, we use directed edges to represent user's page views, unlike the social and active interaction graphs, which are both undirected. The set of users (61,405 total) remains unchanged between the social and interaction graphs. We define *latent interaction in-degree* of a node as the number of visitors who have visited that user's profile; while *out-degree* is the number of profiles that user has visited.

We construct latent interaction graphs from our Renren data using profile views as the latent interactions. We use user comments as the active interaction data to construct active interaction graphs for Renren. In this article we restrict our social, latent, and active interaction graphs to only contain users from the PKU network, since these are the only users for which we have complete interaction records. Note that we only consider interactions that occur between users in the PKU network, as it is possible for interactions to originate from or target users outside the network for whom we have limited information. Also note that because nonfriend strangers can view user's profiles, the latent interaction graph will contain edges between users who are not friends in the social graph.

Our formulation of interaction graphs uses an unweighted graph. We do not attempt to derive a weight scheme for interaction graphs analyzed in this article, but leave exploration of this facet of latent interaction graphs to future work.

Table X. Topology Measurements for Latent Interaction, Active Interaction, and Social Graphs

Network	Nodes	Links	Avg. Deg.	C. Coef.	Assort.	PathLen.
Social Graph	61,405	753,297	12.3	0.18	0.23	3.64
Active Interaction Graph	61,405	27,347	0.89	0.05	0.05	5.43
Latent Interaction Graph	61,405	240,408	7.83	0.03	−0.06	4.02

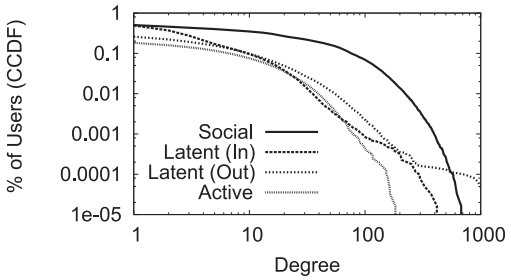


Fig. 27. CCDF of node degree for latent interaction graph, active interaction graph, and social graph.

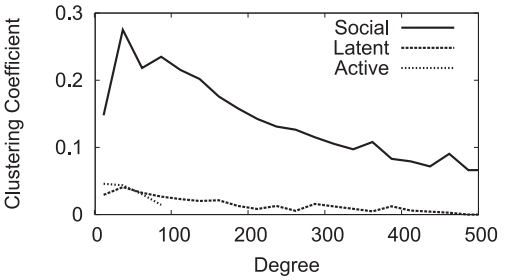


Fig. 28. Clustering coefficient distribution for different graph types.

5.2. Comparing Social, Active Interaction, and Latent Interaction Graphs

In this section, we compare the salient characteristics of the Renren social, active, and latent interaction graphs using common graph metrics. We use data from the PKU regional network, since we do not have active and latent interaction events for all Renren users. Thus, results for the social graph in this section are not directly comparable to the results for the entire social graph presented in Section 2.

Degree Distribution. Figure 27 plots the CCDFs of node degree for the three types of graphs. Since the latent interaction graph is directed, we plot both in-degree and out-degree. In Section 3.5, we show that latent interactions are more prevalent than active interactions. This is reflected in the relative number of edges in the two interaction graphs, as shown in Table X. This also leads to nodes in the latent graph having a noticeably higher degree of distribution in Figure 27. However, neither of the interaction graphs have as many edges as the raw social graph, which leads to the social graph having the highest degree distribution. Interestingly, because a small number of Renren users are frequent profile browsers, that is, they like to visit a large number of profiles (far greater than their circle of friends), the distribution of latent out-degrees flattens out at the tail end and never approaches 0%.

Clustering Coefficient. Table X shows that the average clustering coefficient is 0.03 for the latent interaction graph and 0.05 for the active interaction graph, which are both much less than that of the social graph. This is because not all social links are accurate indicators of active social relationships, and these links with no interactions are removed in interaction graphs. This produces loose connections between neighbors and low clustering coefficients in these graphs. A portion of the latent interactions to a profile are from nonfriend strangers who randomly browse the network. Thus, links between visitors in the latent interaction graph are less intensive than friends exchanging messages, which further lowers the clustering coefficients in latent interaction graphs.

Figure 28 further explores the distribution of clustering coefficients in our three graphs. As previously noted, the sparsity of edges in the latent and active graphs results in much less clustering when compared to the social graph. In fact, the line for the active graph stops at 87, because that is the maximum node degree in the entire graph. In addition to the overall sparseness of the latent and active graphs,

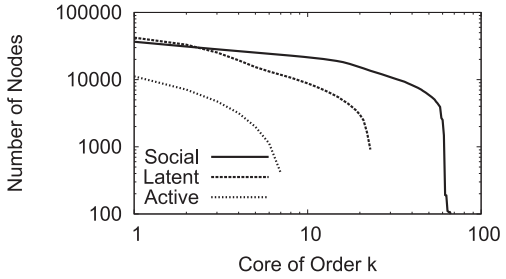


Fig. 29. k-core analysis of different graph types.

these graphs also lose the tightly clustered fringe exhibited by the social graph. This indicates that the interaction graphs are comparatively less "small-world" than the social graph.

Assortativity. Table X shows that the Renren latent interaction graph is slightly disassortative. This makes sense intuitively, as latent interactions are highly skewed towards a small subset of extremely popular users. In contrast, the other two graphs are both assortative, with the social graph being more so. This result contrasts with previous studies in which the interaction graph was more assortative than the social graph [Wilson et al. 2009].

Average Path Length. The average path length of the latent interaction graph is between that of the active interaction graph and the social graph. As the average number of links per node and the number of high-degree super-nodes decreases, the overall level of connectivity in the graph drops. This causes average path lengths to rise, especially in the active interaction graph. This further corroborates the weakening of small-world properties previously evinced with regards to the clustering coefficient.

k-core. k-cores are used to study the strongly connected core of graphs. The k-core is a subgraph in which all nodes have at least k edges to neighbor nodes in the subgraph. k-cores are computed by iteratively removing nodes with degree $< k$ from a graph until all remaining nodes have degree $\geq k$. Note that the removal of a node modifies the degree of its neighbors, who may also then drop below the acceptable threshold and need to be removed. The end result of this process is one or more subgraphs called k-cores.

Figure 29 shows the k-core size of social, latent interaction, and active interaction graphs. In the social graph, the core size remains relatively stable until $k > 60$, at which point the number of nodes drops rapidly. This threshold separates the fringe of the network from the strongly connected core. The Renren social graph exhibits a larger fringe than other large social graphs, such as Cyworld ($k = 38$) [Chun et al. 2008] and MSN Messenger ($k = 20$) [Leskovec and Horvitz 2008].

In contrast to the social graph, the core size for interaction graphs rapidly declines as k increases. Neither interaction graph exhibits a well-defined inflection point between strong and weak core connectivity. The difference between social and interaction graph k-core results stems from the lack of high-degree supernodes in the interaction graphs.

5.3. Community Structure of Social, Active Interaction, and Latent Interaction Graphs

In this section, we compare and contrast the community structure of the Renren social, active interaction, and latent interaction graphs. Our goal is to investigate the quantitative differences between communities in different types of graphs. Our expectation

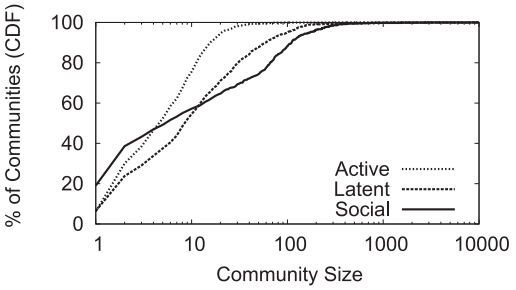


Fig. 30. The distribution of community sizes.

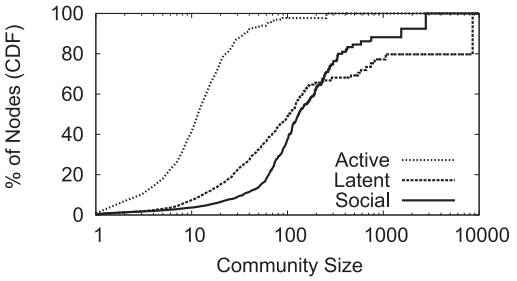


Fig. 31. The percentage of nodes in communities of different sizes.

is that the deviations in structural properties exemplified in Section 5.2 will translate to different observed community structures.

Community Detection Methodology. Community detection is a well-studied area, and there are many different algorithms for locating communities [Fortunato 2010]. Our goal is not to evaluate different community detection algorithms or propose new ones. Instead, in this article, we use the local community detection approach proposed in Viswanath et al. [2010]. At a high level, this algorithm starts with a single seed node then iteratively, greedily adds new nodes that minimize the community's *conductance*. This process terminates when adding new nodes no longer decreases the community's *conductance*. We chose this algorithm because it has been shown to work well on data from large OSNs, and it has practical applications in areas like Sybil detection [Viswanath et al. 2010].

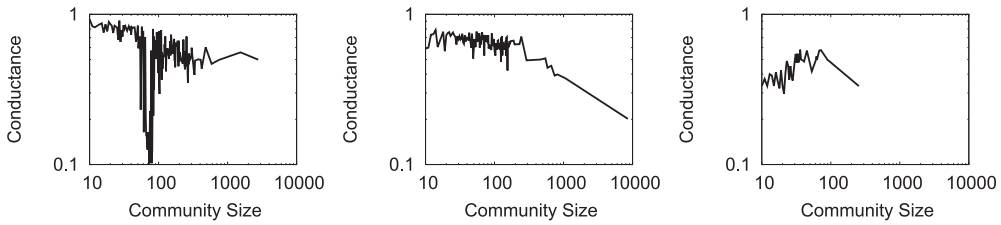
Conductance is a metric of community quality defined in Leskovec et al. [2010]. Formally, conductance $\phi(S)$ of the set S is $\phi(S) = \frac{c_S}{min(Vol(S),Vol(V \setminus S)}$, where $c_S = |\{(u, v) : u \in S, v \notin S\}|$, $Vol(S) = \sum_{u \in S} d(u)$, and $d(u)$ is the degree of node u. Conductance of a community ranges from 0 to 1, with lower conductance indicating better internal connectivity than external connectivity.

In our experiments, we select a random seed node and execute the community detector. The nodes in the resulting community are marked as unavailable, a new random seed is selected from the remaining nodes, and the community detector is run again. This process continues until all nodes are placed in communities.

Community Analysis. Figure 30 shows the distribution of community size obtained after running the community detector on the Renren social, latent interaction, and active interaction graphs. In each case, the majority of communities are small: 50% of communities have $\leq$6 nodes in the social graph, $\leq$9 nodes in the latent graph, and $\leq$5 nodes in the active graph. The active interaction graph produces the smallest communities overall, which makes sense given that it is the smallest and sparsest of the three graphs.

Although Figure 30 reveals that the social graph produces the greatest percentage of large communities, the latent interaction graph actually produces the largest single community of the three graphs. Figure 31 shows the percentage of nodes contained in communities of different sizes. The largest community in the social graph is 2,779 nodes, while the largest in the latent interaction graph is 8,555 nodes (which are 20% of the total nodes in the graph).

The giant community in the latent interaction graph arises because this graph is disassortative (see Table X). Although the latent graph is sparser than the social graph, it still contains many high-degree supernodes, that is, real-life celebrities whose profiles



(a) Social graph. (b) Latent interaction graph. (c) Active interaction graph.

Fig. 32. Conductance versus community size. Lower scores are better.

are viewed by many people. The importance of these nodes is actually magnified in the latent graph, because the fringe of the network is no longer tightly clustered (thus causing assortativity to become negative). Thus, the community detection algorithm builds a single large community with supernodes at its core.

Figure 31 reinforces the size disparity between communities in the social and latent interaction graph versus the active interaction graph. In the active interaction graph, 98% of nodes belong to communities with size $\leq$100. In contrast, only 38% of nodes in social graph and 49% of nodes in the latent graph belong to communities with $\leq$100 nodes. This result indicates that communities in the active interaction graph may be the most "natural," that is, correspond to actual real-world communities, as opposed to being artifacts of the chosen community detection algorithm. Unfortunately, we are unable to investigate this hypothesis more deeply because we do not have the detailed profile information of PKU users, for example, employer, hometown, favorite music, etc. Without this additional data, it is impossible to say whether the detected communities correspond to groups of users that share similar traits.

Conductance. Figure 32 plots the *conductance distribution function* of Renren communities versus their size. The conductance distribution function is defined as $\Psi(k) = min_{|S|=k}\phi(S)$, where $\phi(S)$ is conductance of the community with size k. Thus, $\Psi(k)$ is the smallest conductance from all communities of size k [Leskovec et al. 2010]. Intuitively, this function characterizes the quality of communities of a given size, with lower values denoting stronger communities. As seen in Figure 32, conductance tends to improve as community size grows. This trend is independent of the graphs we are using, because the conductance metric favors larger communities in general. For example, a community that encompasses an entire graph would have zero conductance. We ignore communities with < 10 nodes, as they tend to always have zero conductance [Leskovec et al. 2010], and the results are not useful.

Surprisingly, communities from the active interaction graph have the lowest overall conductance. This again indicates that active interaction graphs are the most useful graph formulation for locating strong communities. This occurs because the vast majority of inactive social edges are pruned from the active graph. The conductance of communities from the latent interaction graph are also lower than from the social graph, but only by a small margin. The large 8,555-node community from the latent graph performs particularly well, but this result is due to the inherent bias of conductance towards large communities.

5.4. Evolution of Active and Latent Interaction Graphs

One key difference between friendship links and implicit, interaction links is that interaction events are inherently temporal. Friendship links have a creation time, but after that, they are static (except in rare cases where users unfriend each other).

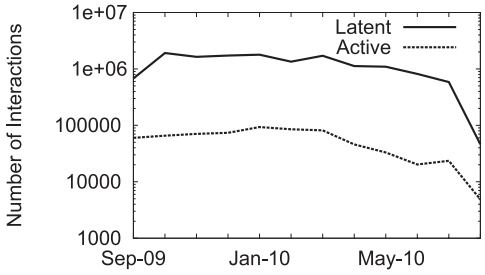


Fig. 33. Interactions per month in the Renren PKU network.

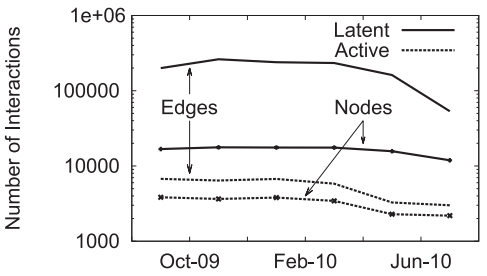


Fig. 34. Nodes and edges per interaction graph snapshot.

However, interactions between users are ephemeral: the set of friends a given user interacts with changes over time as relationships wax and wane.

If we assume that interaction links have a half-life after which they are no longer relevant and expire, this means that interaction graphs may have significant temporal dynamics. Prior work observes that links in Facebook-derived active interaction graphs come and go rapidly over time [Viswanath et al. 2009], confirming our hypothesis. However, to date, no studies have examined the time-varying nature of latent interactions.

Understanding the temporal nature of latent interactions has important implications for researchers studying social networks, as well as companies designing social applications. The design of algorithms and applications is often motivated by snapshots of real data from social networks. However, if there are significant temporal variations in latent interactivity, data snapshots may not be representative, which may lead to incorrect results.

Dataset and Experimental Setup. In this section, we explore the evolution of interaction graphs on Renren. As described in Section 2.2, Renren gave us the complete, anonymized set of active and latent interactions for the 61,405 users in the PKU network. These interactions occurred between September 2009 and August 2010. Figure 33 shows the number of latent and active interactions per month in our dataset. As expected, latent interactions outnumber active ones by an order of magnitude. The data also exhibits a noticeable seasonal trend: during summer break in August, when PKU is not in session, user activity drops significantly.

Using this data, we construct time-varying active and latent interaction graphs. We divide the dataset into six snapshots, each of which contains two months' worth of interactions. We then construct active and latent interaction graphs using the same methodology given in Section 5.1. As before, latent graphs are directed, while active graphs are undirected. Intuitively, these time-based interaction graphs reflect users' changing communication patterns. In order for a given edge to appear in multiple snapshots, the users in question must communicate at least once every two months. The set of 61,405 PKU users examined in this section remains the same as in Section 5.2.

Figure 34 shows the number of nodes and edges per snapshot. Solid lines denote number of edges over time, while lines with points denote nodes over time. The number of nodes in each snapshot reveals how many users generated active and latent interactions during each two-month period. Around 17% of PKU users generate at least one latent interaction per month, while only $\approx$4% of PKU users generate active interactions per month. The number of edges in each graph is around one order of magnitude less than the total number of interactions seen in Figure 33, since many interactions occur between duplicate pairs of users. The seasonal fall-off in interactions

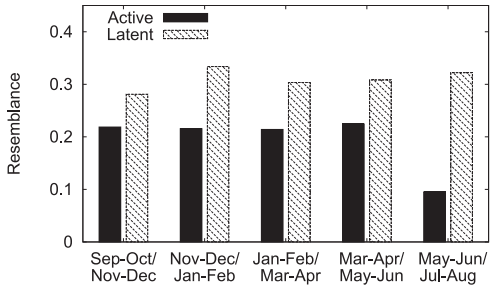


Fig. 35. Resemblance of evolving active and latent interaction graphs.

heading into summer break translates directly into fewer nodes and edges in the last two snapshots.

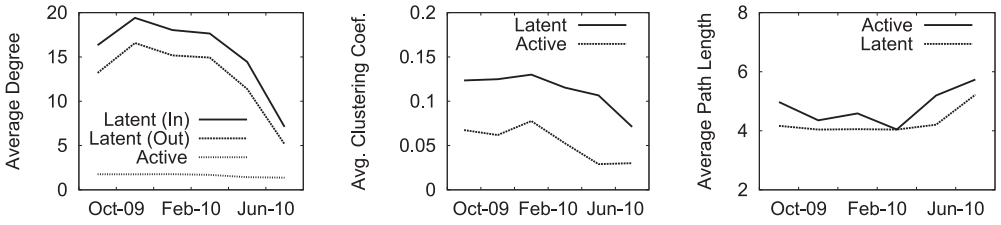
Resemblance. One important measure of time-varying graphs is *resemblance*: the fraction of links which remain unchanged from one snapshot to the next. Resemblance is defined as $r_t = \frac{|S_t \cap S_{t+1}|}{|S_t|}$, where S_t is the set of links in the snapshot t [Viswanath et al. 2009]. The value of r_t is between 0 and 1. If $r_t = 1$, all links in snapshot t exist in snapshot $t + 1$. If $r_t = 0$, none of links in the snapshot t persist in snapshot $t + 1$.

We plot the resemblance of the evolving active and latent Renren interaction graphs in Figure 35. In general, latent interaction graphs have more resemblance than the active graphs, indicating that users' browsing behaviors are more stable than their communication patterns. However, this greater stability is relative; both types of graphs have resemblance of <0.5, meaning that the majority of interacting pairs change every two months. Most of the active interaction graphs have similar resemblance (≈ 0.21), with only the final snapshots having low resemblance (0.1). The resemblance of the latent interaction graphs is stable (≈ 0.3). This reveals an interesting seasonal trend: PKU students visibly interact with different people when they are away on summer break, possibly friends from home. However, they still use Renren to browse the profiles of their friends from college, presumably to keep up-to-date on their summertime activities.

Structural Properties. We now examine how the structural properties of evolving interaction graphs change over time. Figure 36 shows the average degree, average clustering coefficient, and average path length for active and latent interaction graph snapshots over time. The average degree results in Figure 36(a) exactly track the seasonal fluctuations in edges seen in Figure 34, although this is difficult to see, since Figure 34 is in log-scale. Figures 36(b) and 36(c) also exhibit the same seasonal patterns, with clustering reducing and path lengths rising as the graphs become sparser.

Note that the results in this section are not directly comperable to those in Section 5.2, since different datasets are used in each section. This section leverages complete interaction records from September 2009 to August 2010, whereas Section 5.2 leverages our crawled dataset.

Our results from Renren indicate that seasonal factors can drastically affect the properties of interaction graphs. These results contrast with the findings from prior work, which showed that the properties of active interaction graphs on Facebook were stable over time [Viswanath et al. 2009]. Viswanath et al.'s results differ from ours because they focus on a regional network, while we are focused on an academic network. Students exhibit strong trends that are tied to the school year, while an entire American city is not as strongly synchronized seasonally.



(a) Average degree. (b) Average clustering coefficient. (c) Average path length.

Fig. 36. Structural properties over time for active and latent interaction graphs.

6. IMPACT ON SOCIAL APPLICATIONS

Researchers have proposed many algorithms and applications that leverage social graphs to perform useful tasks. For example, researchers have leveraged social networks to augment email spam filters [Garriss et al. 2006], as well as perform decentralized detection of Sybils (fake accounts) [Yu et al. 2006, 2008]. Similarly, several algorithms have been developed that can identify influential users on social graphs in order to maximize the potential for information dissemination [Chen et al. 2009; Kempe et al. 2003].

These social algorithms and applications are often validated using static social graph topologies. However, static graph topologies are not necessarily the most realistic model to use when evaluating social applications. Consider the example of information dissemination: existing work assumes that for each user, information is equally likely to disseminate across each of their edges. Our results from Section 3 cast doubt on this simplistic model: in reality, information does not disseminate unless there are active or latent interactions across edges. Thus, we believe that a more realistic model for building and evaluating social algorithms and applications should take interactions into account.

To understand how the underlying choice of model impacts social applications, we reevaluate three social applications from the literature on the social, active interaction, and latent interaction graphs presented in Section 5. The first application we evaluate is SybilGuard [Yu et al. 2006], a decentralized system for detecting Sybils on social graphs. In order to function, SybilGuard assumes that there is strong pairwise trust between social friends. However, the static social graph is likely to overestimate how many edges are actually trusted by users. To understand the performance of SybilGuard under more realistic circumstances, we evaluate the algorithm on active and latent interaction graphs. We observe that SybilGuard's performance drops significantly on active and latent interaction graphs. We examine why this occurs by calculating the *mixing time* of our Renren graphs, since the performance of SybilGuard is closely tied to this metric.

The second application we evaluate is the MixedGreedWC algorithm for calculating influence maximization (i.e., how to maximize information dissemination on a graph) [Chen et al. 2009]. Our results show that information does not disseminate as widely when interaction graphs are used. This result demonstrates that prior evaluations of this algorithm based on static social graphs are likely to be overly optimistic, since real-world information dissemination relies on latent and active interactions.

Finally, the third application we evaluate is the Reliable Email [Garriss et al. 2006] whitelisting system. RE augments existing spam filters by assuming mail from social friends should be whitelisted, which reduces the computational load on spam filters. However, RE assumes that all social friends are equally trusted. This assumption is likely to overestimate the trust users place on some of their social contacts. A more

realistic model is to evaluate RE using interaction graphs, that is, only assume that users trust friends that they have previously interacted with. In this case, our evaluation shows that using interaction graphs improves the performance of RE by reducing the amount of spam received by simulated users. Thus, in contrast to the prior examples, in this case, using interaction graphs improves the performance of prior work.

Note that we are not advocating for social network providers to begin revealing active and latent interactions to third parties. Obviously, users' interactions and browsing behavior are privacy sensitive information. Our purpose in this section is simply to point out that many current social applications implicitly rely on user interactions, and that evaluating these applications on models derived from static graph topologies may lead to results that do not conform to reality.

6.1. SybilGuard

The Sybil attack is one of the most well-known and powerful attacks against OSNs. In a Sybil attack, an attacker creates multiple fake accounts that work together to increase the power of the attacker. Sybils are responsible for generating much of the spam on today's OSNs [Gao et al. 2010; Grier et al. 2010; Yang et al. 2011; Thomas et al. 2011; Wang et al. 2012].

Several prior works have developed algorithms for performing decentralized detection of Sybils on social graphs. The original and most well-known algorithm in this space is SybilGuard [Yu et al. 2006]. SybilGuard works by initiating n specialized random walks of length w from nodes u and v on a social graph. Node u is known as the *verifier*, while v is the *suspect*. If the majority of the random walks intersect, then u accepts that v is not a Sybil. However, if too few or none of the walks intersect, then u classifies v as a Sybil. There are many works that all propose similar algorithms to SybilGuard [Yu et al. 2008; Danezis and Mittal 2009; Cao et al. 2012; Tran et al. 2009], and it has been shown that all of these techniques generalize to community detection Viswanath et al. 2010.

The intuition behind the SybilGuard algorithm is that it is easy for Sybils to friend each other but difficult for Sybils to form friendships with honest users. Thus, Sybils tend to form tightly-knit communities that have few connections to the honest region of the graph. SybilGuard's random walk process is designed to detect the small quotient-cut that separates the Sybil and honest regions of the graph, that is, random walks that begin in the Sybil region are likely to get trapped there and thus will not intersect walks that begin in the honest region. This property enables honest nodes to detect Sybils.

Analyzing SybilGuard. In order to function, SybilGuard assumes that users will exchange cryptographic keys with their friends. The authors of SybilGuard evaluate their algorithm and present results based on a complete social graph, that is, they assume that all edges are signed. However, as we have shown in Section 3.5, most edges on the Renren social graph are not interactive. Thus, it seems unlikely that users will exchange keys with all their friends. In this case, active and latent interaction graphs are a more realistic model on which to evaluate the effectiveness of SybilGuard.

We implemented the SybilGuard algorithm and executed it on the Renren social, active interaction, and latent interaction graphs to observe if and how its performance might change. Figure 37 shows the percentage of random walks that intersect on each type of graph as the length of the walks is increased. Intuitively, longer walks are more likely to intersect, but they also enable more Sybils to be erroneously accepted. Thus, shorter walk lengths offer better security. For each walk length, we executed 2,500 random walks between randomly chosen pairs of verifier/suspicious nodes.

Figure 37 demonstrates that SybilGuard is less effective when run on the latent and active interaction graphs. When the walk length is 300, 45% of walks intersect on the

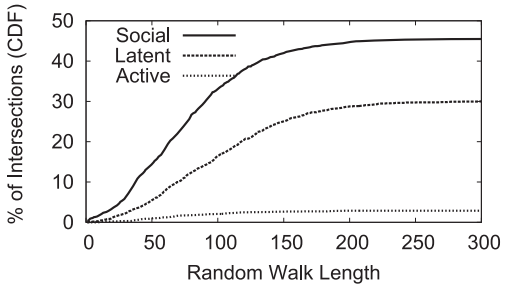


Fig. 37. Random walk intersections as walk length is varied on different graph topologies.

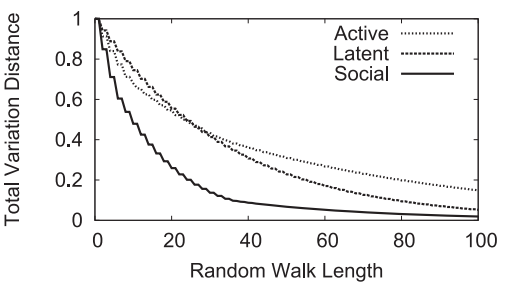


Fig. 38. Mixing time for social, latent interaction, and active interaction graphs.

full social graph versus 30% and 3% for the latent and active graphs, respectively. Note that many walks never intersect, and thus the CDF totals never reach 100%. These result confirm our suspicion that the performance of SybilGuard may be reduced when executed under realistic circumstances.

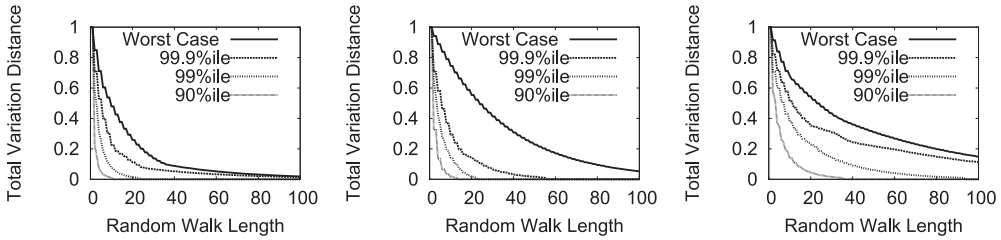
Mixing Time. Next, we endeavor to understand why the performance of SybilGuard is reduced on the latent and active interaction graphs. In particular, we focus our analysis on the *mixing time* of our Renren graphs. The authors of SybilGuard state that the performance of the algorithm is linked to the mixing time of the underlying graph, that is, *fast-mixing* graphs will perform better than *slow-mixing* graphs [Yu et al. 2006]. However, prior work has shown that interaction graphs are slower mixing than full social graphs [Mohaisen et al. 2010]. Thus, it is likely that the reduced performance of SybilGuard seen in Figure 37 is linked to the mixing time of the Renren graphs.

Mixing time is defined as the number of steps a random walk on a graph must take before the probability that the walk has reached any given node is equal to the *stationary distribution* for the graph. The stationary distribution for a graph is simply a probability vector π such that $\pi = \pi P$, where P is the transition matrix of the graph. More specifically, for an undirected graph $G = (V, E)$, $\pi = [\pi_{v_i}]$, where $v_i \in V$, the degree of a node is $deg(v_i)$, and $\pi_{v_i} = deg(v_i)/(2 * |E|)$. Intuitively, the stationary distribution is just the limit of a Markov process on the graph as the number of steps in the process approaches infinity. Mixing time is simply a more precise measure of the length of the process: rather than continue for infinite steps, what is the finite number of steps necessary to reach the stationary distribution?

The speed of mixing on a graph is a measure of how quickly random walks reach the stationary distribution relative to the size of the graph. A graph is considered to be fast mixing if the mixing time is on the order $O(log|N|)$. If mixing time is greater than this quantity, then the graph is considered to be slow mixing. Intuitively, if a graph is fast mixing, this tells us that the graph is very well connected: a random walk that begins at any node can reach any other node in the graph in just a few steps, regardless of how large the graph is. Conversely, slow-mixing graphs may have internal structural impediments that prevent random walks from being able to easily traverse the graph.

Analyzing Mixing Time. To compute the mixing time for our Renren graphs, we conduct brute-force random walks from each node in each graph. At each step of each walk, we compute the *total variation distance* of the walk from the stationary distribution. Walks terminate when the variation distance approaches zero. The mixing time for a given graph is the maximum random walk length (e.g., the worst case) necessary to reach the stationary distribution. Note that we use the same methodology used in Mohaisen et al. [2010].

Figure 38 shows the variation distance of the worst-case random walk for social, latent interaction, and active interaction graphs. As walk length increases the variation



(a) Social graph. (b) Latent interaction graph. (c) Active interaction graph.

Fig. 39. Variation distance for different percentiles of nodes (sorted from shortest walk length to longest).

distance decreases, since the walk progressively approaches the stationary distribution. The social graph exhibits the fastest mixing, that is, the walk converges to zero the fastest. Interestingly, although the walk on the latent graph starts off slowly, by step 100, it almost catches up to the social graph. This indicates the presence of poorly connected outliers in the latent graph. Local exploration is slow due to the nodes poor connectivity, but once the densely connected core is reached, the walk speeds up. The sparsely connected active interaction graph is the slowest mixing of the three, which agrees with prior Facebook mixing results [Mohaisen et al. 2010].

The results in Figure 38 reinforce the findings in Figure 37. SybilGuard performs best on the social graph, which is also the fastest-mixing graph by a large margin. Conversely, SybilGuard performs worst on the active interaction graph, which is also the slowest-mixing graph. However, care should be taken not to overgeneralize the worst-case mixing time results: SybilGuard performs significantly better on the latent graph than the active interaction graph, but the two interaction graphs have similar worst-case mixing characteristics.

To gain a deeper understanding of why SybilGuard's performance varies so widely between the latent and active interaction graphs, we plot the variation distance for different percentiles of nodes (sorted from shortest to longest walks) in Figure 39. For example, the worst-case variation distance is 0.554 for a 20-step walk on the latent interaction graph. However, variation distance for the 99.9th percentile node is only 0.139, and variation distance for the 90th percentile node is 0. This means that 90% of random walks on the latent graph reach the stationary distribution within 20 steps.

There are two takeaways from Figure 39. First, we observe that worst-case mixing times on Renren graphs are not representative for the majority of nodes. In each case, 90% of nodes have mixing times $\leq$10, which is quite fast. Although the worst-case mixing time on the latent graph diverges from the worst-case on the social graph, the results for the 99.9th percentile and lower are quite similar. This reinforces our assertion that a few poorly connected outliers significantly impact the worst-case mixing time on the latent interaction graph.

Second, Figure 39 explains why SybilGuard performs better on the latent interaction graph than the active interaction graph. Although both graphs have similar worst-case mixing times, the latent graph has much faster mixing times in general, that is, the 90th, 99th, and 99.9th percentile mixing times are similar to those exhibited by the full social graph. Conversely, the 99th and 99.9th percentile mixing times are similar to the worst-case mixing times on the active interaction graph. Thus, many more nodes experience slow-mixing times on the active interaction graph.

6.2. Efficient Influence Maximization

As OSNs become increasingly popular worldwide, they also become more critical platforms for information dissemination and marketing. Understanding how to fully utilize OSNs as marketing and information dissemination platforms is a significant challenge.

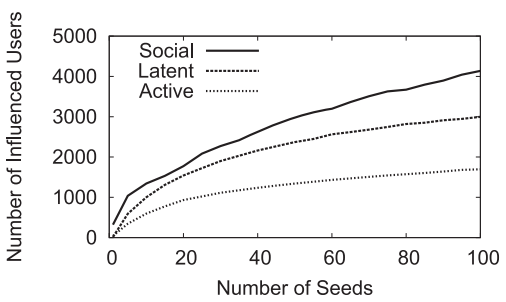


Fig. 40. Influence spread using the MixedGreedyWC algorithm on different graph topologies.

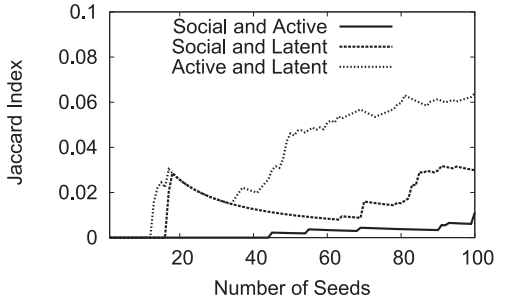


Fig. 41. Overlap in influential seeds selected by MixedGreedyWC on different graph topologies.

The influence maximization problem seeks to determine the most influential individuals who will maximize the spread of information in an OSN. This can be framed as an optimization problem: choose S *seed nodes* that maximize the diffusion of information on a target graph given some rules about how information propagates across social links.

Given the lack of publicly available social influence datasets, previous work [Chen et al. 2009; Kempe et al. 2003] builds statistical models based on raw social graph topologies and designs algorithms to address influence maximization problems within these models. One prominent model is the *weighted cascade model* [Kempe et al. 2003]. In this model, each user that receives information has a single chance of activating (i.e., spreading information to) each currently inactive neighbor. The activation probability is related to a node's degree: if a person w has d_w neighbors, it is activated by neighbors with probability $1/d_w$.

Unfortunately, selecting optimal seeds in the weighted cascade model is NP-hard, and thus solutions must be approximated. Chen et al. instantiate the MixedGreedyWC algorithm to implement a fast, accurate approximation of the weighted cascade model [2009].

Goals and Methodology. In this section, our aim is to evaluate the MixedGreedyWC algorithm on all three of our graphs: full social, interaction, and latent. Intuitively, this experiment evaluates how information spreads in social graphs under different assumptions about how users consume information. If all social links are equally important to dissemination, then the full social graph is the best graph to use when modeling. However, a more realistic assumption is that only active social links are useful for information dissemination. In this case, latent or active interaction graphs are better choices when modeling information dissemination.

Note that the results presented for the MixedGreedyWC algorithm have changed from those presented in the original conference version of this article [Jiang et al. 2010]. In the original, article diffusion was calculated using a different methodology for each graph type (social, active, latent). This made results for each graph difficult to compare directly. The results presented in this article use one methodology for calculating diffusion on all three graphs, and thus the results are clearer and easier to compare.

For our experiments, we use the MixedGreedyWC algorithm [Chen et al. 2009] to find the most influential individuals in each of our three graphs, and then compute the number of people influenced. We vary the set of seed users to the MixedGreedyWC algorithm from 1 to 100 in our tests and observe the effects on influence spread.

Experiments and Analysis. Figure 40 shows influence spread versus seed set size for the three test graphs. Influence spreads fastest via the full social graph, since it has

the highest average node degrees (as shown in Table X). However, Figure 24 illustrates that information is not disseminated equally through all social links. For example, user profiles are usually only viewed by and receive comments from a small portion of friends. Thus, modeling information dissemination using the full social graph is an optimistic upper bound on how information would spread in reality, since not all social links correspond to active relationships.

Interaction graphs provide a more accurate base for modeling information dissemination, since each edge indicates user engagement (either browsing or exchanging messages). Figure 40 shows that fewer users are influenced when using interaction graphs versus the full social graph. The latent interaction graph results in greater dissemination than the active interaction graph because average node degrees are higher in the former.

In addition to understanding the properties of information dissemination, we also want to examine whether the same seeds are optimal on different types of social graphs. To understand this, we plot the overlap in seeds between different types of social graph in Figure 41. For example, the Social and Active line in Figure 41 shows the Jaccard index (overlap) between the set of seeds chosen by MixedGreedyWC on the Renren social and active interaction graphs. A Jaccard index of 1 denotes total overlap, that is, MixedGreedyWC selects exactly the same seed nodes from both graphs. 0 denotes that no seeds were selected in common. The x-axis denotes the total number of seeds (i.e., the size of the seed set) chosen by MixedGreedyWC.

Figure 41 shows that the optimal seed nodes are very different on different graph topologies. The full social graph and the active interaction graph are almost totally dissimilar; the maximum Jaccard index between these two graphs is $\approx$0.01, meaning that, on average, 1 out of 100 seeds overlaps. The full social graph and the latent interaction graph show relatively more overlap, but the scalar values are still small: $\approx$3 out of 100 seeds. Finally, the active and latent interaction graphs have the most similar seed sets, sharing $\approx$6 out of 100 seeds.

The conclusion from Figure 41 is that the optimal seeds for disseminating information vary widely depending on the graph model used. Practically, this means users of algorithms like MixedGreedyWC need to be careful that they choose the correct representation of the social graph before running the algorithm to select seeds. For example, seeds chosen from the full social graph are unlikely to be successful if information actually disseminates along the active or latent edges.

Discussion. The takeaway from these results does not suggest that information dissemination algorithms should be evaluated on static social graph because they produce the "best performance." Instead, our aim is to point out that results from prior work that base performance claims on static social graphs may be overly optimistic. In reality, information disseminates when users forward it to friends (active interactions) or consume it from friends' profiles (latent interactions). Interaction graphs provide a more realistic model for evaluating information dissemination algorithms and thus generate evaluation results that are more indicative of the real-world performance of information dissemination algorithms.

6.3. RE: Reliable Email

RE [Garriss et al. 2006] is a whitelisting system for email that securely marks emails from a user's friends and friends of friends as nonspam messages, allowing them to bypass spam filters. Socially-connected users provide secure attestations for each other's email messages, while keeping users' contacts private.

Our experiment is to examine the level of potential impact on RE users if accounts in the social network are compromised using phishing attacks. We randomly choose a

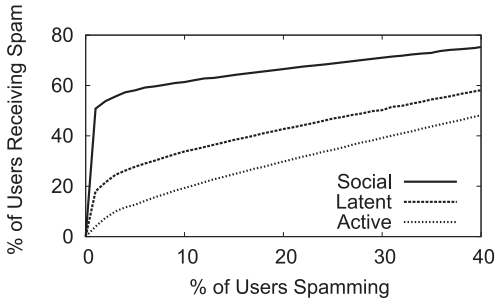


Fig. 42. The percent of users receiving spam as the number of spammers increases on different graph topologies.

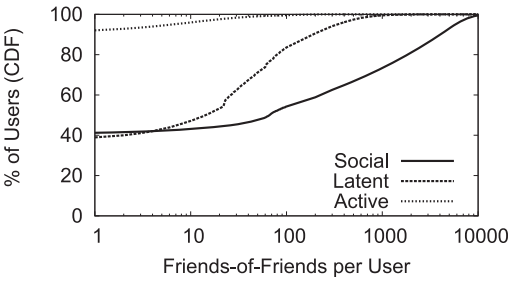


Fig. 43. Number of friends of friends per user on different graph topologies.

percentage of users as spammers and calculate the number of users who are one or two hops away from the spammers on the graph. These users would receive spam due to RE's whitelisting. We include the spammers themselves in the set of users receiving spam. We perform these experiments on Renren's social, active interaction, and latent interaction graph. All experiments are repeated ten times, and the results are averaged.

Experimental results are shown in Figure 42, which plots the proportion of users receiving spam versus the percentage of compromised users sending spam. Although each point in Figure 42 is an average from ten experiments, the standard deviations are all $<0.4\%$, so we omit error bars in the figure. On the social graphs, spam penetration quickly reaches $>50\%$ of the user base when only 1% of users are spamming. Conversely, spam penetration is greatly reduced in active and latent interaction graphs. Even when 40% of people spam, $<50\%$ of users receive spam. This is because both interaction graphs have lower average node degrees, compared to the full social graph. This property reduces the number of users receiving spam. The latent interaction graph allows for slightly greater spam penetration than the active graph because of its higher average node degrees.

To gain a deeper understanding of why spam levels differ so greatly across the three graphs, we plot Figure 43, which shows the number of friends of friends per user in each graph. Intuitively, RE whitelists email from friend of friends; if a given graph is denser, then users will have more friends of friends and thus be more vulnerable to spammers infiltrating the graph. Figure 43 confirms our intuition: users on the Renren social graph have the largest friend-of-friend sets, and the smallest sets on the active interaction graph. This corresponds exactly with the spam penetration results observed in Figure 42.

Unlike in the previous case, in this example, the performance of RE improves when the algorithm is run on interaction graphs. However, this does come at a cost: by shrinking the size of each user's whitelist, more emails will need to be evaluated by traditional spam filters. Fortunately, our evaluation results provide a roadmap for managing the trade-off between resiliency to spam from compromised accounts versus higher load on the spam filters. By choosing a different underlying graph model, administrators looking to deploy RE can choose a point in the spectrum that suits their individual needs.

7. RELATED WORK

Much effort has been put into understanding the structure of large-scale online social networks [Fu et al. 2008]. Ahn et al. analyze topological characteristics of Cyworld, MySpace, and Orkut [Ahn et al. 2007]. Mislove et al. measure the structure of Flickr,

YouTube, LiveJournal, and Orkut [Mislove et al. 2007] and observe the growth of the Flickr social network [Mislove et al. 2008]. Java et al. study the topological and geographical properties of Twitter [2007]. Huang and Xia measure user prestige and visible interaction preference in Renren [2009]. To the best of our knowledge, our measurement of the Renren network provides the largest non-Twitter social graph to date, with 42,115,509 users and 1,657,273,875 friendship links. Most of Renren's topological properties are similar to those of other OSNs, including power-law degree distribution and small-world properties.

Researchers have also studied the visible interaction network (or active interaction network, in our terminology). Leskovec and Horrite analyze the instant messaging network which contains the largest amount of user conversations ever published [2008]. Valafar et al. characterize indirect fan-owner interactions via photos among users in Flickr [2009]. Chun et al. observe that visible interactions are almost bidirectional in Cyworld [2008]. Wilson et al. show the structure of the interaction graph differs significantly from the social network in Facebook [2009]. Viswanath et al. observe that social links in the activity network tend to come and go quickly over time [2009]. Burke et al. investigate the role of visible interaction between pairs [2010]. Finally, a recent study from Northwestern and UC Santa Barbara quantified the role of spam and phishing attacks in Facebook wall posts [Gao et al. 2010].

Benevenuto et al. collect detailed click-stream data from a Brazilian social network aggregator, and measure silent activities, like browsing [2009]. Schneider et al. extract clickstreams from passively monitored network traffic and make similar measurements [2009]. We analyze latent interactions from a different perspective than these existing works by leveraging data that is intrinsic to the OSN and not inferred from a third party. We would like to make a comparison between our dataset a and clickstream dataset. Unfortunately, the sensitive nature of these datasets make their distribution challenging, we are currently unaware of any publicly available clickstream dataset.

Some researchers have performed initial studies on information propagation and user influence in OSNs. Cha et al. present a detailed analysis of popularity and dissemination of photographs on Flickr [2009]. They find that popular users with high in-degree are not necessarily influential in terms of spawning subsequent, viral interactions in the form of retweets or mentions on Twitter. Our Renren data confirms these results, as we show that factors like number of friends and amount of user-generated content produced are not strongly correlated with popularity.

8. CONCLUSIONS

Latent user interactions make up the large majority of user activity events on OSNs. In this article, we present a comprehensive study of both active and latent user interactions in the Renren OSN. Our data includes detailed visit histories to the profiles of 61,405 Renren users over a 90-day period (September to November 2009). We compute a single visitor history for each profile by using a novel technique to merge visitor logs from multiple consecutive crawls. We analyze profile visit histories to study questions of user popularity and reciprocity for profile browsing behavior and the link between latent profile browsing and active comments.

Our analysis reveals interesting insights into the nature of user popularity in OSNs. We observe that user behavior changes for latent interactions: more users participate, users do not feel the need to reciprocate visits, and visits by nonfriends make up a significant portion of views to most user profiles. We also see that visits to user profiles generate more active interactions (comments) than visits to photos or diary pages. Social networks help people to find and view strangers' profiles, but the effect varies greatly from person to person. Using profile browsing events, we construct latent interaction graphs as a more accurate representation of meaningful peer interactions.

Analysis of the latent interaction graph derived from our Renren data reveal characteristics that fall between active interaction graphs and social graphs. This holds true for simple metrics, like degree distribution and average path length, as well as for more complicated measures, like community structure and mixing time. This confirms the intuition that latent interactions are less limited by constraints, such as time and energy, but more meaningful (and thus sparser) than the social graph.

Finally, our measurement study also includes an exhaustive crawl of the largest connected component in the Renren social graph. The resulting graph is one of the biggest of its kind, with more than 42 million nodes and 1.6 billion edges. Other than the proprietary Cyworld dataset, this is the only social graph we know of that covers 100% of a large social graph component. Given its size and comprehensiveness, we are currently investigating different options for sharing this dataset with the research community.

REFERENCES

Ahn, Y.-Y., Han, S., Kwak, H., Moon, S. B., and Jeong, H. 2007. Analysis of topological characteristics of huge online social networking services. In *Proceedings of the World Wide Web Conference*.

Benevenuto, F., Rodrigues, T., Cha, M., and Almeida, V. 2009. Characterizing user behavior in online social networks. In *Proceedings of the ACM Internet Measurement Conference*.

Burke, M., Marlow, C., and Lento, T. 2010. Social network activity and social well-being. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.*

Cao, Q., Sirivianos, M., Yang, X., and Pregueiro, T. 2012. Aiding the detection of fake accounts in large scale social online services. In *Proceedings of the 9th USENIX Conference on Networked Systems Design and Implementation.*

Cha, M., Haddadi, H., Benevenuto, F., and Gummadi, K. 2010. Measuring user influence in twitter: The million follower fallacy. In *Proceedings of the International AAAI Conference on Weblogs and Social Media (ICWSM).*

Cha, M., Mislove, A., and Gummadi, K. 2009. A measurement-driven analysis of information propagation in the flickr social network. In *Proceedings of the World Wide Web Conference*.

Chen, W., Wang, Y., and Yang, S. 2009. Efficient influence maximization in social networks. In *Proceedings of the ACM Conference on Knowledge Discovery and Data Mining*.

Chun, H., Kwak, H., Eom, Y. H., Ahn, Y.-Y., Moon, S. B., and Jeong, H. 2008. Comparison of online social relations in volume vs interaction: A case study of cyworld. In *Proceedings of the ACM Internet Measurement Conference*.

Clauset, A., Shalizi, C. R., and Newman, M. E. J. 2007. Power-law distributions in empirical data. *J. Comput.-Mediated Commun.*

Danezis, G. and Mittal, P. 2009. Sybilinfer: Detecting sybil nodes using social networks. Tech. rep. MSR-TR-2009-6. Microsoft.

Fortunato, S. 2010. Community detection in graphs. *Physics Rep. 486*, 75–174.

Fu, F., Liu, L., and Wang, L. 2008. Empirical analysis of online social networks in the age of Web 2.0. *Physica A 387,* 2–3, 675–684.

Gannes, L. 2010. When social replaces search, what can you do to monetize? GigaOM. http://gigaom.com/2010/03/24/when-social-replaces-search-what-can-you-do-to-mouetize/.

Gao, H., Hu, J., Wilson, C., Li, Z., Chen, Y., and Zhao, B. Y. 2010. Detecting and characterizing social spam campaigns. In *Proceedings of the ACM Internet Measurement Conference*.

Garriss, S., Kaminsky, M., Freedman, M. J., Karp, B., Mazires, D., and Yu, H. 2006. Re: Reliable email. In *Proceedings of the USENIX Symposium on Networked Systems Design and Implementation*.

Grier, C., Thomas, K., Paxson, V., and Zhang, M. 2010. @spam: The underground on 140 characters or less. In *Proceedings of the 17th ACM Conference on Computer and Communications Security.* 27–37.

Gruhl, D., Guha, R., Liben-Nowell, D., and Tomkins, A. 2004. Information diffusion through blogspace. In *Proceedings of the World Wide Web Conference*.

Huang, L. and Xia, Z. 2009. Measuring user prestige and interaction preference on social network site. In *Proceedings of the 8th IEEE/ACIS International Conference on Computer and Information Science (ACIS-ICIS).*

Java, A., Song, X., Finin, T., and Tseng, B. L. 2007. Why we twitter: Understanding microblogging usage and communities. In *Proceedings of the 9th Web KDD and 1st SNA-KDD Workshop on Web Mining and Social Network Analysis (WebKDD/SNA-KDD'07).*

JIANG, J., WILSON, C., WANG, X., HUANG, P., SHA, W., DAI, Y., AND ZHAO, B. Y. 2010. Understanding latent interactions in online social networks. In *Proceedings of the Internet Measurement Conference*.

KEMPE, D., KLEINBERG, J. M., AND TARDOS, E. 2003. Maximizing the spread of influence through a social network. In *Proceedings of the ACM Conference on Knowledge Discovery and Data Mining*.

KIRKPATRICK, M. 2009. Social networking now more popular than email, report finds. ReadWrite. http://readwrite.com/2009/03/09/socia_networking_now_more_popular_than_email#awesm-~ogyunfdzg+Ovjb.

KWAK, H., LEE, C., PARK, H., AND MOON, S. B. 2010. What is Twitter, a social network or a news media? In *Proceedings of the World Wide Web Conference*.

LAMPE, C., ELLISON, N., AND STEINFIELD, C. 2007. A familiar face(book): Profile elements as signals in an online social network. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*.

LEHMANN, E. L. AND D'ABRERA, H. J. M. 1998. *Nonparametrics: Statistical Methods Based on Ranks*. Prentice-Hall. Upper Saddle River, NJ.

LESKOVEC, J. AND HORVITZ, E. 2008. Planetary-scale views on a large instant-messaging network. In *Proceedings of the World Wide Web Conference*.

LESKOVEC, J., LANG, K. J., AND MAHONEY, M. W. 2010. Empirical comparison of algorithms for network community detection. In *Proceedings of the World Wide Web Conference*.

MILGRAM, S. 1967. The small world problem. *Psychol. Today 1*.

MISLOVE, A., KOPPULA, H. S., GUMMADI, K. P., DRUSCHEL, P., AND BHATTACHARJEE, B. 2008. Growth of the flickr social network. In *Proceedings of the 2nd ACM Workshop on Online Social Networking (WOSN)*.

MISLOVE, A., MARCON, M., GUMMADI, P. K., DRUSCHEL, P., AND BHATTACHARJEE, B. 2007. Measurement and analysis of online social networks. In *Proceedings of the ACM Internet Measurement Conference*.

MOHAISEN, A., YUN, A., AND KIM, Y. 2010. Measuring the mixing time of social graphs. In *Proceedings of the Internet Measurement Conference*.

SCHNEIDER, F., FELDMANN, A., KRISHNAMURTHY, B., AND WILLINGER, W. 2009. Understanding online social network usage from a network perspective. In *Proceedings of the ACM Internet Measurement Conference*.

THOMAS, K., GRIER, C., PAXSON, V., AND SONG, D. 2011. Suspended accounts in retrospect: An analysis of Twitter spam. In *Proceedings of the ACM SIGCOMM Conference on Internet Measurement Conference*.

TRAN, N., MIN, B., LI, J., AND SUBRAMANIAN, L. 2009. Sybil-resilient online content voting. In *Proceedings of the 6th USENIX Symposium on Networked Design and Implementation*. 15–28.

VALAFAR, M., REJAIE, R., AND WILLINGER, W. 2009. Beyond friendship graphs: A study of user interactions in flickr. In *Proceedings of the 2nd ACM Workshop on Online Social Networking (WOSN)*.

VISWANATH, B., MISLOVE, A., CHA, M., AND GUMMADI, K. P. 2009. On the evolution of user interaction in facebook. In *Proceedings of the 2nd ACM Workshop on Online Social Networking (WOSN)*.

VISWANATH, B., POST, A., GUMMADI, K. P., AND MISLOVE, A. 2010. An analysis of social network-based Sybil defenses. In *Proceedings of the ACM SIGCOMM Conference on Application, Technologies, Architectures, and Protocols for Computer Communications*. 363–374.

WANG, G., WILSON, C., ZHAO, X., ZHU, Y., MOHANLAL, M., ZHENG, H., AND ZHAO, B. Y. 2012. Serf and turf: Crowdturfing for fun and profit. In *Proceedings of the World Wide Wed Conference (WWW)*.

WILSON, C., BOE, B., SALA, A., PUTTASWAMY, K. P. N., AND ZHAO, B. Y. 2009. User interactions in social networks and their implications. In *Proceedings of the 4th ACM European Conference on Computer Systems (EuroSys)*.

YANG, Z., WILSON, C., WANG, X., GAO, T., ZHAO, B. Y., AND DAI, Y. 2011. Uncovering social network Sybils in the wild. In *Proceedings of the ACM SIGCOMM Conference on Internet Measurement Conference*.

YAROW, J. 2010. Facebook was more popular in the U.S. than Google last week. BusinessInsider.com. http://www.businessinsider.com/facebook-was-more-popular-in-the-us-than-google-last-week-2010-3.

YU, H., GIBBONS, P. B., KAMINSKY, M., AND XIAO, F. 2008. SybilLimit: A near-optimal social network defense against Sybil attacks. In *Proceedings of the IEEE Symposium on Security and Privacy*. 3–7.

YU, H., KAMINSKY, M., GIBBONS, P. B., AND FLAXMAN, A. 2006. Sybilguard: Defending against Sybil attacks via social networks. In *Proceedings of the ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM)*.

Received June 2011; revised March, October 2012, April, July 2013; accepted July 2013