Modeling/Predicting the Evolution Trend of OSN-based Applications

Han Liu
Department of Electrical & Computer Engineering
University of California-Davis
California, USA
bhgliu@ucdavis.edu

Atif Nazir
Department of Computer Science
University of California-Davis
California, USA
anazir@ucdavis.edu

Jinoo Joung
Department of Computer Science
Sangmyung University
Seoul, Korea
jjoung@smu.ac.kr

Chen-Nee Chuah
Department of Electrical & Computer Engineering
University of California-Davis, California, USA
chuah@ucdavis.edu

ABSTRACT

While various models have been proposed for generating social/friendship network graphs, the dynamics of user interactions through online social network (OSN) based applications remain largely unexplored. We previously developed a growth model to capture static weekly snapshots of user activity graphs (UAGs) using data from popular Facebook gifting applications. This paper presents a new *continuous* graph evolution model aimed to capture microscopic user-level behaviors that govern the growth of the UAG and collectively define the overall graph structure. We demonstrate the utility of our model by applying it to forecast the number of active users over time as the application transitions from initial *growth* to *peak/mature* and *decline/fatique* phase. Using empirical evaluations, we show that our model can accurately reproduce the evolution trend of active user population for gifting applications, or other OSN applications that employ similar growth mechanisms. We also demonstrate that the predictions from our model can guide the generation of synthetic graphs that accurately represent empirical UAG snapshots sampled at different evolution stages.

Categories and Subject Descriptors

C.2.0 [**Computer Communication Networks**]: General; H.4.3 [**Information Systems Applications**]: Communications Applications

General Terms

Measurement, Algorithms

Keywords

Online Social Networks, Social Games, Social Gifting, Facebook, Applications, Algorithms

1. INTRODUCTION

The growing popularity of online social networks (OSNs) such as Facebook has led to extensive research on OSN friendship graphs [1, 2, 3]. Arguably, however, the worth of an OSN resides in how much activity its users generate, rather than simply how connected its users are. Unlike online friendships that are mostly static, the amount of *activity* between user pairs varies over time [4, 5]. Citing this difference, many researchers have highlighted the importance of studying user activity as opposed to simple OSN friendship graphs [6, 7]. User activity data from OSN-based applications hence provide a gateway to study the nature of user dynamics on OSNs. In particular, a user activity graph (UAG) can be constructed for each application where a node represents a user and a directed edge represents an action initiated by one user, targeting another (e.g., user A sends a virtual gift to user B). Unlike friendship graphs, UAGs consist of directed and transient edges, which are not necessarily reciprocal.

We previously pioneered a large-scale measurement study of user interactions on selected Facebook applications with 8 million users [8]. We showed how application dynamics are pivotal in defining the structure of UAGs. Subsequently, we developed a generative model for static snapshots of UAGs [9] over short time-scales (one week), but it **did not** capture the continuous evolution of the UAG/application over time.

In this paper, we consider the problem of measuring and modeling the *long-term continuous* evolution of UAGs from OSN-based applications. Such an evolution model can provide important insights into patterns of user interactions and how they morph across different stages (e.g., initial growth, peak/mature, and decline) of OSN applications' life span. It forms a basis for investigating factors contributing to the viral growth of applications and cascading effects.

Although useful in many aspects, modeling the evolution of UAGs is a very challenging task given their dynamic nature and the lack of user activity data due to privacy concerns. To overcome those problems, we build our model based on the insights gained from a detailed large scale data set collected using the same methodology from our previous work [8]. This data set contains three top Facebook applications that collectively account for **more than 77 million users** and **5.3 billion** entries of user activities. All three applications: *iHeart*, *iSmile*, and *Hugged* belong to gifting applications, the second largest genre of Facebook applica-

WWW 2013, May 13–17, 2013, Rio de Janeiro, Brazil.
ACM 978-1-4503-2035-1/13/05.

tions. In such applications, when a gift is sent from User A to User B, who has not installed the application yet, the Facebook platform will deliver an Application Requests (ARs) to User B, and User B can either accept it (install the application) or ignore it. This AR based growth mechanism can boost the active user number free of cost [9], and hence is very popular among Facebook applications.

Our main contributions are summarized as the following:

• Using key insights gained from our longitudinal data, we develop a dynamic graph evolution model aimed to capture user-level behaviors that govern the macroscopic growth of the UAG and collectively define the overall graph structure.

• We demonstrate the practical relevance of our model by applying it to forecast the number of active users and predict the saturation peak of an OSN application (both are useful for developers). We evaluate our approach using both the activity data collected by ourselves, and publicly available data from application analytics site, AppData [10].

• We utilize the prediction results to generate synthetic snapshots of UAGs at different evolution phases of the applications. Our synthetic graphs mimic several important properties, including the graph size and degree distributions, of the real UAG snapshots, and are hence useful for studying how structural characteristics of UAGs change over time. The ability to generate representative UAGs is useful for the research community, especially since sharing real datasets is prohibitive due to their sheer size and privacy concerns.

Note that we do not seek to solve the problem of dynamic network modeling in general. Instead, our model is driven by the measurements of real UAGs from Facebook gifting applications, and we demonstrate that it is applicable to other applications that belong to the same genre or employ the same underlying growth mechanism (i.e., cost-free user recruitment by sending ARs). Our review of the top 250 Facebook applications suggests that ARs are widely used for expanding the user population, but some applications may use more than one growth mechanisms. Although our model does not fully capture the dynamics of applications that incorporate multiple mechanisms to drive the growth, it still provides us with an insight into the impact of ARs on the evolution of UAGs built on those applications. As a result, for applications incorporating multiple mechanisms to drive growth, our model can still help to investigate how the impact of the AR base growth mechanism takes place.

In the next section, we summarize the related work, and then provide an overview of our measurement methodology and data set in Section 3 . Section 4 presents a measurement-based characterization of individual node behavior across time. Section 5 presents the microscopic state transition model that captures user dynamics, which collectively determine the evolution of the resulting UAG and its structure. It also discusses how the model can be applied to predict number of active users and generate representative synthetic UAG snapshots. We evaluate our approach in Section 6. Section 7 concludes this paper with a discussion of the possible extension of our model.

2. RELATED WORK

Most existing graph models for social networks are designed specifically for friendship graphs [11, 3, 2, 12]. Those studies generate edges between isolated nodes according to two structural properties of social networks (i.e., preferential attachment and triadic closure), and thereby create a graph topologically similar to real networks. In addition, when applying the preferential attachment and triadic closure, latest studies consider more information, such as social attributes [13] and user locations [14], in order to generate more representative synthetic graphs. Some recent work also allows new nodes to arrive at a given constant rate, and then adds edges between new nodes and existing nodes. This approach can partially mimic the expansion of friendship graphs [2, 12, 13, 14]. For the case of UAGs, however, since both nodes and edges are transient, these models can only be used to generate *static* snapshots, which represent user activities accumulated in a given time window, instead of the continuous evolution process. Moreover, for every snapshot, empirical parameters including the number of active nodes (or the nodes joining rate) are required as the inputs of these models.

The study of dynamic graphs [15, 16] and characterizing temporal networks through parametric, generative models remains an active research area. So far, two categories of modeling strategies have been proposed. The first category involves discretizing a temporal network by generating static snapshots of the network in consecutive time windows. The snapshots are then used to model how graph characteristics change with time [17, 18]. The second approach adopts a continuous process that predicts each node's activity based on its current and previous states [19, 20]. In order to model the temporal sequence of user interactions, some studies use small time granularities so as to capture only one user activity in each time slot [21]. The first approach is easier to apply and can leverage existing studies on static graphs, while the second is more accurate in capturing details of dynamic graph evolution. Unfortunately, current work adopting either approach is incapable of modeling temporal networks with both dynamic edges and nodes. In some recent studies, researchers assume that nodes neither join nor leave the graph to reduce the modeling complexity [22, 23]. Although such studies can be used in many scenarios, e.g., phone call communications [19] and face-to-face interactions of a static group of people [24], they do not suit our goal of modeling evolution of UAGs, since users may join or leave OSN applications (and the UAGs) freely.

To describe the transient of nodes, prior work on epidemic spread proposed the susceptible-infected-recovered (SIR) model, which represents the temporal graph of disease infection as a dynamic process on the static network of people's contacts. In the SIR model, in every time slot a susceptible node might become infected with certain probability if contacted by an already infected node, while the infected node can recover with another probability. The latest SIR studies consider cases where the underlying contact network involves heterogeneous [25], or even dynamic [26, 27] edges to better mimic real-world scenarios. However, the SIR model is still inadequate for our purpose due to its need for the knowledge on the underlying user contacts network, which is not available in our case. Through OSN platforms, user interactions (contacts) may take place between any pair of users (even between non-friends in many aplications), and are hence hard to be predicted or modeled as a previously known graph. Moreover, SIR model assumes the evolution of temporal graphs as a homogeneous stochastic process, i.e. the transition probabilities between "susceptible", "infected", and "recovered" for each node is invariant over time. Nevertheless, through our data analysis in Section 4, we demon-

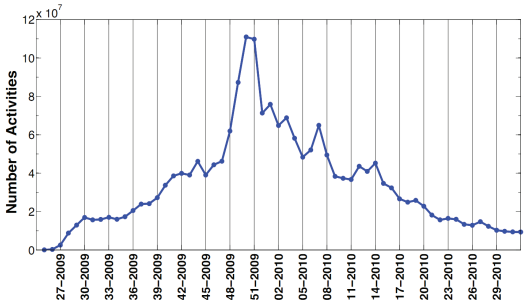


Figure 1: The evolution of graphic properties of the UAG built on iHeart.

strate the necessity of introducing a time-variant model in order to capture the time dependent nature of user behavior patterns in OSN applications.

Although the SIR model cannot be directly applied to model the long-term evolution of UAGs, it provides hints for characterizing a temporal network with dynamic edges and nodes. More specifically, the SIR model suggests that one could capture the behavioral pattern of individual nodes, and use this to explain the evolution of UAGs' size and topology. We employ this idea to design a state transition model to represent user behavior in OSN-based applications. Our model does not utilize any underlying network (i.e., a friendship graph), and recruits new nodes at a rate dependent on the current number of nodes.

3. MEASUREMENT METHODOLOGY AND DATASETS

The growth process of an UAG depends on the growth mechanism employed by the OSN application. In this paper, we first focus on gifting applications, and then evaluate the applicability of our model to other genres of applications that also employ the *AR-based growth mechanisms*. Our study is based on three categories of datasets described below.

The first category includes anonymized user activity data collected from three popular Facebook gifting applications, which were owned and operated by Manakki LLC during the period of our study. These applications are: *iHeart* (launched in June 2009, installed by 77 million users, 64 weeks' data), *iSmile* (launched in August 2008, installed by 43 million users, 85 weeks' data), and *Hugged* (launched in February 2008, installed by 28 million users, 140 weeks' data). For each recorded activity, this dataset includes the senders' and recipients' anonymized Facebook UIDs as well as timestamps at which the activities were initiated.

Our dataset spans more than a year of user activities on the three gifting applications, and therefore provide us with rich information on the patterns of long-term UAG evolution. In particular, the data from iHeart were collected immediately after it was launched, and hence covering the whole lifespan of the application (its growth, peak, and decline phases). Fig. 1 shows the number of user activities on iHeart, and indicates three distinct phases: the growth period (until week 46, 2009), the peak period (week 47, 2009 to week 5, 2010), and the decline period. In December 2009, iHeart was one of the top three Facebook applications by monthly active users (MAU). For each application, we construct the associated UAG $G = (V, E)$ by adding a temporal edge ($e \in E$) between two unique nodes A and B (where $A, B \in V$) when the User A performs an activity on the User B through the application. We also find that the clustering coefficient of the weekly UAGs remain low (about 0.03) and stable across weeks during the growth and peak phases, but gradually decreases in the decline phase (to about 0.011).

Table 1: Dataset Overview

Application	Launch Date	Records Since	Total Users	Peak WAU
Hugged	Feb 2008	Mar 2008	28M	550K
iSmile	Aug 2008	Jun 2009	43M	850K
iHeart	Jun 2009	Jun 2009	77M	5.2M
Send Gifts	Mar 2011	Mar 2011	N/A	25.5K
Birthday Cards	N/A	Nov 2009	N/A	16M
Free Gifts	N/A	Oct 2011	N/A	90K
Coco Girl	N/A	Oct 2011	N/A	550K
Truth about You	N/A	May 2012	N/A	4.2M
School Feed	Aug 2011	Aug 2011	N/A	7.8M

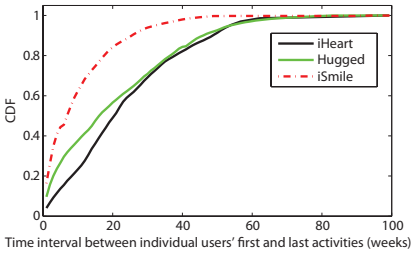


Figure 2: Length distribution of the interval between a user's first and last activity.

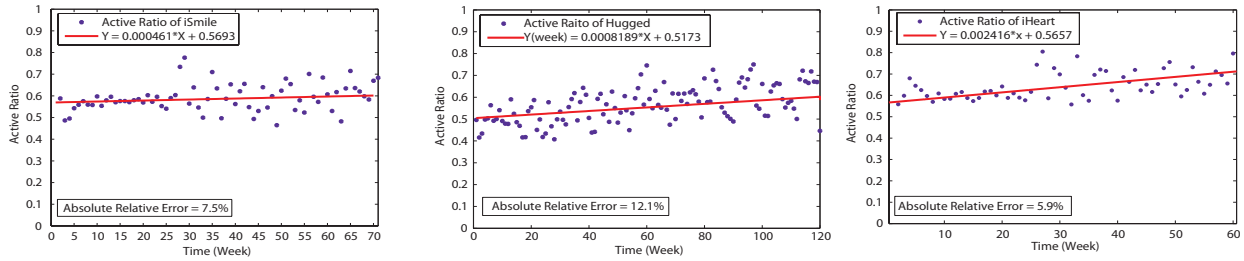
The second category of datasets contain the number of monthly, weekly, and daily active users on other Facebook applications, which is publicly available from the application analytics site AppData ([10]). We use these datasets to evaluate the effectiveness of our model on a larger basis. We do this by randomly picking three gifting applications from [10], namely *Send Gifts*, *Birthday Cards*, and *Free Gifts*.

Third, we pick several popular non-gifting applications, which also use ARs as the dominant mechanism for growth, to evaluate the applicability of our model beyond the genre of gifting.

In Section 6, we fit our model to the above datasets, and test whether it can reproduce the evolution of the number of active users in those applications. Table 1 summarizes the datasets analyzed in this paper (WAU refers to weakly active users).

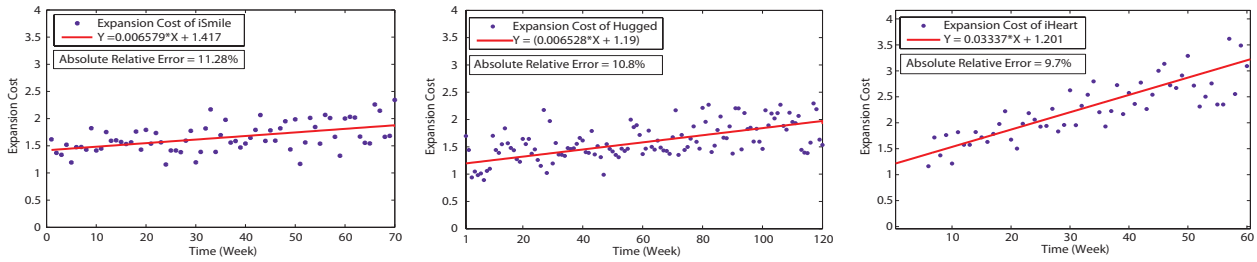
4. NODE BEHAVIOR CHARACTERISTICS

In this section, we attempt to characterize individual nodes' properties and behaviors based on analysis of data from iHeart, iSmile, and Hugged. We also briefly discuss the key empirical observations that provide guidelines for formulating the state transition probabilities in our model. The time granularity used in our model is one week, which we have previously shown to be long enough for the structural properties of UAG to stabilize [9] while sufficiently short to capture the details of UAG evolution.



(a) iSmile (b) Hugged (c) iHeart

Figure 4: The active ratio as a function of time.



(a) iSmile (b) Hugged (c) iHeart

Figure 5: The expansion cost as a function of time.

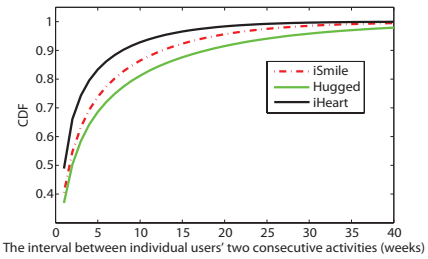


Figure 3: Length distribution of the interval between a user's two consecutive activities.

4.1 Node States Definition

In the UAGs built upon iHeart, iSmile, or Hugged, a temporal, directed edge is added between two nodes when a user sends an item/gift to another user. For the convenience of expression, we refer to this as an "action" of the node representing the gift sender.

Fig. 2 illustrates the distribution of the time intervals between users' first and last activities. Although an application can span over one hundred weeks, most users perform all of their activities within a much shorter time span (more than 50% of users were only active for less than 20 weeks). On the other hand, Fig. 3 illustrates the distribution of time intervals between two consecutive activities of the same user. We can see that the likelihood for the consecutive activity from the same user to take place within a week since his/her previous activity is less than 50%. This finding shows that, in a social application UAG, it is rare for a node to continuously generate activities through multiple weeks. The observations from Fig. 2 and Fig. 3 indicate the necessity of introducing an intermediate state, in which the probability for a node to generate an action in each week is larger than 0 but smaller than 1.

Based on our findings above, we define three node states:

Active. A node is in active state if it generates at least one action within a week.

Alive. A node enters the alive state since it joins the UAG (as a potential sender). In this state, every node has a positive probability smaller than 1 to be active in each week.

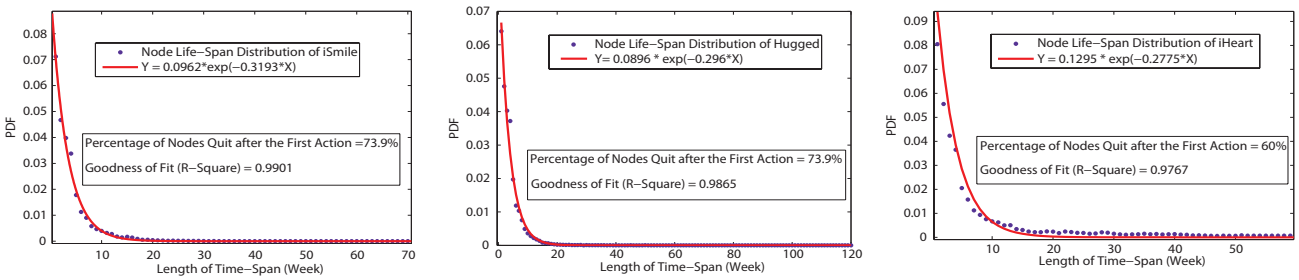
Quit. A node in quit state no longer generate any action and is assumed to have abandoned the application.[1]

In our empirical analysis, if a node has been inactive for a period longer than the preset threshold, it is assumed to have already left the graph (transition to *Quit*) **since its last action**. We use one month as the default value of this threshold. Slightly increasing or decreasing the threshold will not affect the main findings of this paper.

4.2 Active Ratio

We define the term "active ratio" as the portion of alive nodes that are active in each week. This metric provides us hints on how to model the weekly behavior of nodes in the 'Alive' state. Fig. 4 shows how the values of active ratio change through time for all our three applications. As shown by the data fitting results shown in Fig. 4, this metric is approximately linear with respect to time. The absolute relative error of the fitted linear model to the real data are 5.9% for iHeart, 12.1% for Hugged, and 7.5% for iSmile.

[1]A small fraction of nodes that quit may generate actions after extended period of absence, and we treat them as new nodes in our model.



(a) iSmile (b) Hugged (c) iHeart

Figure 6: The distribution of node life-span (the time interval in which a node remains in the alive state).

4.3 Expansion Cost

To model the evolution of UAGs, it is also important to infer the number of new nodes joining the graph in each week. In applications that employ ARs as the major growth mechanism, a large portion of new users are recruited by the ARs sent to them from current users. Alternatively, users may also install an application when they see it on the application list of the OSN platform, or discover that their friends are using it. Not surprisingly, the rate of new nodes joining is highly dependent on the population of existing users. Fig. 5 plots the ratio of current number of active nodes to the number of new nodes joining in the following week. We define this ratio as the "expansion cost", i.e., how many existing users it takes to recruit a new user. Similar to the active ratio, our results show that the expansion cost is also approximately linear with respect to time (absolute relative error: 9.7% for iHeart, 10.8% for Hugged, and 11.28% for iSmile), and monotonically increases after the application is launched. This finding indicates that, as time passes by, an application needs more and more active users to recruit the same amount of new users, and hence will gradually saturates. When the rate of nodes leaving exceeds joining/staying, the application will begin to shrink and eventually die out.

4.4 Life-span Distribution

In addition to the expansion cost that characterizes the recruitment of new nodes, we use the distribution of nodes' life-span, the time interval in which a node remains in the *Alive* state, to predict when nodes will leave the graph. As shown in Fig. 6, a large portion of nodes transition into the *Quit* state immediately after their first action, and the life-span distribution of the remaining nodes can be perfectly fitted into an exponential function.The goodness of fit (R-Square) are 0.9767, 0.9865, and 0.9901 for iHeart, Hugged, and iSmile, respectively.

5. MODELING THE EVOLUTION OF UAGS

5.1 State Transition Model

This section presents our node state transition model that describes the dynamics of microscopic user behavior through time, which in turn govern the evolution of the UAGs. Our model can be explained using the state transition graph shown in Fig. 7 and the notations used are listed in Table 2. Our model only contains state transitions for *action initiators*. Nodes that only *receive* and never generate actions are not considered here.

All the new nodes that join the graph after generating their first actions are considered *Active* in their first week according to our definition. However, as shown in Section 4.4, a large proportion of new nodes quit the graph immediately after the first week and, as a result, significantly differ from the remaining nodes whose life-span follows the exponential distribution. Therefore we introduce a separate state "new joining", which transitions into the *Quit* state with a probability P_{IQ}, to model the behavior of this type of new nodes.

After surviving the first week, nodes will remain in *Alive* state. As described in Section 4.1, an *Alive* node can either be inactive (noted as *Alive & inactive* in the transition graph) or generate at least one action (noted as *Active*) in each week, and will switch between these two states with certain probabilities.

Most transition probabilities in our model are time dependent. There are two time notations used in Fig. 7, t and t'. t denotes the number of weeks since the beginning of the application, and t' denotes the number of weeks after a particular node joins. All the transition probabilities are functions of three fundamental metrics: (1) P_{IQ}, (2) $P_Q(t')$, the probability of transitioning from *Alive* to *Quit*, and (3) $P_A(t)$, the probability of *Alive* nodes becoming *Active*. These metrics are also the input required by our model. Moreover, in order to describe the size change of an UAG, the expansion cost (noted as $E_P(t)$) is required as well.

Based on the observations presented in Section 4, we formulate $P_A(t)$ and $E_P(t)$ as a linear function of t, and $P_Q(t')$ as a function of the nodes' exponential lifespan distribution. i.e.

$$P_A(t) = a * t + b \tag{1}$$

$$E_P(t) = c * t + d \tag{2}$$

$$P_Q(t') = \frac{e * \exp(f * t')}{1 - \sum_{t''=1}^{t'-1} e * \exp(f * t'')} \tag{3}$$

where $t' \geq 1$. Since $P_Q(t')$ and P_{IQ} describe the probability density function (PDF) of nodes' life-span distribution, the summation of P_{IQ} and the integral of $e * \exp(f * t')$ over $t' \in [1, +\infty)$ should be 1. As a result, the value of e is dependent of f, and only f is required by our algorism. In other words, our proposed evolution model for UAG requires six input parameters $\{a, b, c, d, f, P_{IQ}\}$.

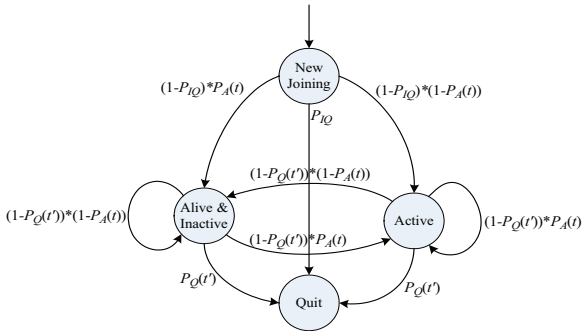


Figure 7: State transition model that describes the behavior of individual nodes.

Table 2: Notations of the state transition model

New Joining	The node joins the UAG and takes its first action in the current week
Alive & Inactive	The node is alive, but takes no action in the current week
Active	The node takes at leat one action in the current week
P_{IQ}	The probability that a new node quits the graph immediately after its first action
$P_Q(t')$	The probability for a node to quit in the current week
$P_A(t)$	The probability for alive nodes to be active in the current week

5.2 Modeling and Predicting the Number of Active Users

A very practical use of our UAG evolution model is to predict the number of active users in an OSN application in future weeks based on initial empirical observations. Knowing the number of active users, the source of all dynamics, is a key step for studying many other structural properties of UAGs.

Let $A(t)$, $AL(t)$, and $N(t)$ denote the total number of *active*, *alive*, and *new joining nodes* in week t. Let $AL^{(t)}(t')$ denote the number of nodes that join the graph in week $(t-t')$ and are still alive in week t. From state transitions illustrated in Fig. 7, we have

$$A(t) = N(t) + AL(t) * P_A(t) \tag{4}$$

$$N(t) = A(t-1)/E_P(t-1) \tag{5}$$

$$AL(t) = \sum_{t'=1}^{t-1} AL^{(t)}(t') \tag{6}$$

$$AL^{(t)}(t') = N(t-t') * (1-P_{IQ}) * (1 - \sum_{t''=1}^{t'-1} e * \exp(f * t'')) \tag{7}$$

Using Equation (4)-(7), we can get $A(t)$, $AL(t)$, and $N(t)$ based on information from weeks no later than $(t-1)$, and the results can then be used to calculate $A(t+1)$, $AL(t+1)$, and $N(t+1)$. Thus, by recursively utilizing Equation (4)-(7), we will be able to generate the whole process of how the active user population in an OSN UAG evolves through time.

In practice, we can estimate $\{a, b, c, d, f, P_{IQ}\}$ by fitting the data collected in a short initial monitoring period into Equation (1)-(3) (the time window for data collection in all of our experiments is 20 weeks), and input the fitted parameters into our model to forecast the number of active users in the future. We now discuss how to address two practical challenges in employing our model as a prediction tool. The first challenge is to accurately estimate the number of existing nodes that are *Alive* when we first monitor an OSN application. One option is to pre-monitor the application for a month, and count all the nodes that are active at least once. Then, using this as a basis, we can start collecting data and estimating other input parameters after this period. Secondly, the exact time when the application was launched may be unknown. Hence, it may be difficult to determine the value of t, which is relative to the launching date. To address this issue, we assume the week when data collection starts as the first week of the application (t =1), so the $P_A(t)$ and $E_P(t)$ we get are the original functions with a time-shift. Since both functions are linear with respect to time, the final results of the prediction would not be affected.

5.3 Generating Representative UAG Snapshots

In this subsection, we demonstrate how the results of active user number prediction can be used to guide the generation of synthetic weekly snapshots of the UAG. Note that these snapshots are static graphs, and a directed edge is added if its source end takes at least one action on its destination end within the time window over which the snapshot is generated. We follow the approach proposed in [9] to generate the synthetic weekly UAG, while taking advantage of the prediction results to enhance the modeling accuracy. We include the detailed description of the mechanism (Algorithm 1) here for completion.

This mechanism requires the following additional parameters: (1)$d(x)$, the distribution of the number of days an active node is activated (i.e. generates at least one action) in a week, (2) $m(y)$, the distribution of the number of actions from a node, if activated, in a day, (3) s_l, the limitation of actions a node is allowed to generate in a day (this constraint comes from the upper-bound of daily send activities set by some OSN platforms, such as Facebook), (4) β, the percentage of inactive nodes, whose out-degree is 0, in an UAG snapshot.

Based on our characterization of UAGs generated upon real data, between 70% to 75% of active users in a given week are only activated for a day, and this percentage decays approximately as a power law with number of days activated (e.g.,only between 1.5% and 2% users are active for all seven days). Our measurements also show that the distribution of actions initiated by a activated user per day follows a power law [9]. As a result, we formulate $d(x)$ and $m(y)$ as

$$d(x) = g * x^{-\gamma_d} \tag{8}$$

where $x \geq 1$,

$$m(y) = h * y^{-\gamma_m} \tag{9}$$

where $y \geq 1$. Similar to e and f, only γ_d and γ_m are required to describe these two power-law distributions. Therefore, the additional inputs required by the synthetic graph generation mechanism is actually a four-tuple $\{\gamma_d, \gamma_m, s_l, \beta\}$.

Among these parameters, s_l is an external input parameter that is set by the OSN platform. For all other pa-

rameters, we found that their values are relatively stable throughout the life-span. Take iHeart as an example, the mean values of $\{\gamma_d, \gamma_m, \beta\}$ across all the weeks in the data sets are {1.8233, 0.8467, 0.8852}, while their standard deviations are {0.0780, 0.0546, 0.015}. Since the deviations are quite small compared with the means, we can compute the means of parameters based on empirical data collected during the initial monitoring period (during which the parameters for active user number prediction are estimated), and use the results as the input of our mechanism to generate synthetic graphs for all other weeks.

At the first step of our mechanism, $A(t) * \frac{1}{(1-\beta)}$ isolated nodes are placed in the graph, in which $A(t)$ nodes are active and the rest are inactive ($A(t)$ is determined by our active user number prediction model). The whole generation process is divided into 7 time slots, each representing one day. In each time slot, an active node will be activated, and how many actions it is going to take are determined by Equation (8) and Equation (9). Note that all nodes determined to be activated for more than 7 time slots will only be active for 7 time slots. Similarly, nodes determined to take more than s_l actions in one day will stop after the $s_l{}^{th}$ action.

When a node generates an action, the target will be chosen according to the principle of preferential attachment [28], i.e. the probability for a node to be picked is proportional to its current in-degree. If currently there is no edge from the action initiator to the target, a new edge will be added between them. Preferential attachment has been reported as an effective way to model the in-degree distribution of social network graphs by many recent studies [2]. However, there are still some practical issues remaining to be addressed. First, every inactive node should have an in-degree no less than 1. Because a node that is neither the source nor the destination of any action would not have been recorded in the data, and hence may not appear in the graph. As a result, if there are still inactive nodes without any incoming edge in the graph, the destination of a new added edge should be randomly chosen among them. Second, after all inactive nodes are provided with one incoming edge, the in-degree of all active nodes is still 0. Therefore, if we adopt the preferential attachment model, the destinations of edges will all be assigned as inactive nodes. To address this problem, we modify the assignment principle as: an active node is chosen to be the edge destination with a probability proportional to its current in-degree plus 1.

After 7 time slots, the synthetic snapshot is produced. Algorithm 1 shows the pseudocode for our synthetic snapshot generation mechanism. In the pseudocode, $IPL(\gamma_d, 7)$ and $RPL(\gamma_m, s_l)$ are integer- and real-valued random variables with distributions following Equation (8) and (9). The exponents of the two power law distributions are cut-off at 7 and s_l, respectively. Moreover, the variable act_x is the number of time slots a node has previously been active. It works with another intermediate variable dif_x to adjust the error caused by discretization (line 10-14 in Algorithm 1).

Algorithm 1 Generating Representative UAG Snapshots

Require: γ_d, γ_m, s_l, β, $A(t)$;
1: Initialize a graph $G = (V, E)$ (V is the node set, E is the edge set). Divide V into three subsets, V_a, V_i, V_{in} ;
2: Add $A(t) * \frac{1}{(1-\beta)}$ isolated nodes into V. Assign $A(t)$ of them into V_a, rest into V_i;
3: **for all** x in V_a **do**
4: Activate x;
5: $d_x = IPL(\gamma_d, 7); m_x = RPL(\gamma_m, s_l); act_x = 0$;
6: $dif_x = \lfloor d_x * m_x \rfloor - d_x * \lfloor m_x \rfloor$;
7: **end for**
8: **for** each i in $[1, 7]$ **do**
9: **for all** activated x in V_a **do**
10: **if** $dif_x \geq act_x$ **then**
11: $dailyactions = \lfloor m_x \rfloor$;
12: **else**
13: $dailyactions = \lfloor m_x \rfloor + 1$;
14: **end if**
15: **for** each j in $[1, dailyactions]$ **do**
16: **if** V_{in} is not empty **then**
17: Randomly pick a node y from V_{in};
18: Add an edge (x, y) to E;
19: Move y from V_{in} to V_i;
20: **else**
21: Pick up a node y from V, with probabilities proportional to out-degrees for nodes in V, and proportional to out-degree +1 for nodes in V_a;
22: **if** $(x, y) \notin E$ **then**
23: Add an edge (x, y) to E;
24: **end if**
25: **end if**
26: **end for**;
27: $d_x = d_x - 1; act_x = act_x + 1$;
28: **if** d_x=0 **then**
29: Deactivate x;
30: **end if**
31: **end for**
32: **end for**
33: **return** G

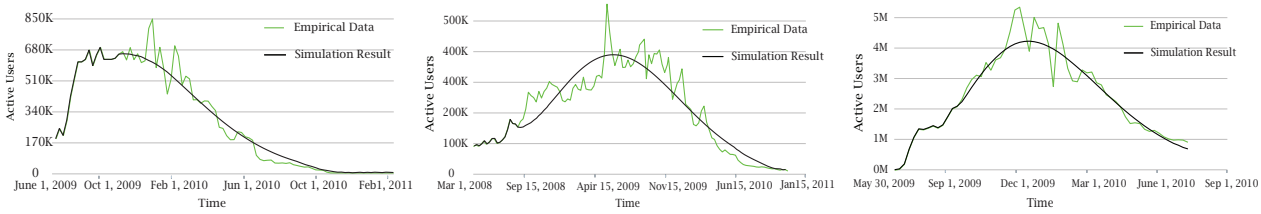
6. EVALUATION

6.1 Active User Number Prediction

We first evaluate the performance of our model in predicting the number of active users in an UAG using the data of iHeart, iSmile, and Hugged. For each of the applications, we estimate all the six parameters $\{a, b, c, d, f, P_{IQ}\}$ based on 20 weeks of data, and conduct the prediction for all the future weeks. Our records for iHeart can be traced back to when the application was launched (64 weeks in total), while the data for iSmile and Hugged start at a more mature phase (85 and 140 weeks, respectively).[2] As stated in Section 5.2, our model is capable of predicting active user population regardless of when the measurement begins relative to the launching date of the application. The results also demonstrate that the predictions for the three applications are all fairly accurate. We use the average value of absolute relative errors to quantify the performance of our prediction and the results are shown in Fig. 8. The average errors for iSmile, Hugged, and iHeart are 13.8%, 14.6%, and 10.7%, respectively.

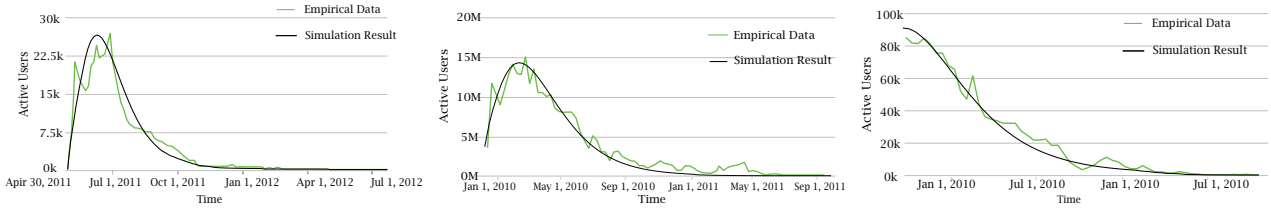
Note that the major errors of prediction occur during holiday weeks (e.g. Christmases and Valentine's days for iHeart and iSmile, and the Mothers' day for Hugged), where more users are active due to exogeneous effects, which leads to

[2]iSmile dies out in October 2010, after which only a small amount of users are active each week. Therefore, our performance evaluation only uses data prior to this date.



(a) iSmile (b) Hugged (c) iHeart

Figure 8: Comparison between the predicted number of active users and real data records.



(a) Send Gifts (b) Birthday Cards (c) Free Gifts

Figure 9: Reproduce the evolution process for applications from appdata.com.

sub-linear expansion cost. Notwithstanding, the holiday factor does not affect the prediction for the subsequent weeks. This observation shows that users are more likely to use the social applications on holidays, but those transient extra activities do not help significantly in attracting new users. In our future work, we will consider using an external variable to describe the impact of holidays on the active user population.

6.2 Universality

Next, we examine whether our model can reproduce the evolution process of other social applications besides iHeart, iSmile and Hugged. Note that the data from AppData [10] only contains the number of active users in every week. The lack of sufficient information makes the user number prediction a very difficult task. Therefore, in this group of experiments we estimate the model parameters based on all the available data, instead of only the first 20 weeks, and try to find out a set of input $\{a, b, c, d, f, P_{IQ}\}$ that enables our model to reproduce weekly active node numbers that are close to the real data. We adopt an iterative search method to guess a proper set of parameters for each application. In all of our experiments, the target set of $\{a, b, c, d, f, P_{IQ}\}$ can always be found. This indicates that our model is able to explain the UAG evolution of all applications we have tested.

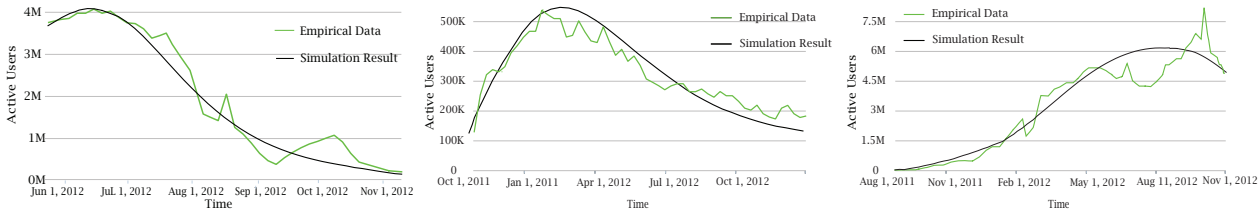
Similar to Section 6.1, we consider an application (Send Gifts) that has data records since its inception, and two other applications (Birthday Cards, Free Gifts) of which we only have data traces for the post peak (mature) phase. We calculate the absolute relative errors for the modeling of Send Gifts and Free Gifts before the applications die out (January 2012 for Send Gifts, May 2010 for Free Gifts). The average errors are 15.2% and 14.3% for those two applications, respectively (Fig. 9).

While analyzing data from Birthday Cards, we found that the number of active users sharply increases in October 2010, and the evolution of active user population greatly deviates from its original trend after that date. It is highly possible that certain external factors, e.g. paid advertising, have been introduced that result in this abrupt change. Unfortunately, our current model does not consider the impact of such external factors. As a result, we only calculate the average error of the modeling results on Birthday Cards upon data collected earlier than October 2010. The resulting average error is 13.2%. We will try address the issue of external factors affecting the evolution of UAGs in our future work.

Lastly, Fig. 10 illustrates the error when our model is applied to reproduce the evolution of UAGs from applications other than gifting. All the three applications are chosen from the top 100 most popular Facebook applications. Truths about You is an interactive quizzing platform that allows friends to ask and answer questions about themselves. School Feed is an online classmate network integrated with Facebook. While Coco girl is a virtual shopping tool enabling users to purchase digital items and show them to friends. The average absolute relative errors for those three applications are 9.7%, 7.2%, and 13.7% respectively, indicating that our model can also be used to predict the evolution of non-gifting applications as long as they employ the AR-based growth mechanism as the major tool to recruit new users. Most of the applications that our model fail to capture (such as social games) incorporate multiple growth mechanisms, including paid advertising or direct email contact, which are not considered in our model.

6.3 Representation of Structural Properties

We now use the data traces of iHeart to evaluate the performance of the synthetic snapshots generation mechanism proposed in Section 5.3. We estimate all the required parameters $\{a, b, c, d, f, P_{IQ}, \gamma_d, \gamma_m, \beta\}$ based on 20 weeks of data, and set s_l to be the same as announced by Facebook. Using the estimated parameters, we predict the UAG snapshots for all future weeks.



(a) Truths about You (b) Coco Girl (c) School Feed

Figure 10: Reproduce the evolution process for applications in genres other than gifting.

Table 3: Evaluation of the node number, edge number, and largest connected component size of synthetic graphs

	Node Number		Edge Number		Size of LCC	
Week	Emp	Syn	Emp	Syn	Emp	Syn
34, 2009	9.8M	10.7M	11.8M	12.3M	8.7M	9.5M
45, 2009	22.3M	26.3M	29.6M	31.5M	19.1M	23.5M
50, 2009	44.7M	39.3M	110.0M	103.2M	41.4M	38.1M
01, 2010	31.4M	33.0M	43.8M	48.2M	30.0M	30.5M
06, 2010	30.9M	35.3M	47.6M	52.0M	28.5M	30.9M
14, 2010	19.0M	20.2M	23.7M	24.2M	17.9M	18.4M
19, 2010	11.6M	11.2M	13.0M	12.8M	10.2M	10.8M
26, 2010	9.9M	8.3M	11.1M	10.0M	8.5M	7.8M
30, 2010	6.5M	5.7M	6.9M	5.8M	5.9M	5.4M

The comparisons of number of nodes, number of edges, and the size of largest connected component (LCC) between the empirical graphs and synthetic graphs are shown in Table 3. These results demonstrate that the synthetic graphs can mimic the original snapshots with high accuracy. Particularly, for different time periods, the largest connected components of the synthetic graphs consist of more than 95% of all the nodes, and the second largest components are of negligible size (less than 20 nodes). This is consistent with what we have observed in the empirical data. Moreover, although the clustering coefficient (CC) of the synthetic graphs is not precisely the same as the empirical data, the synthetic graphs all have very small CC (less than 0.025). This result accurately represents the trend that UAGs from gifting applications all tend to have much smaller CC than many other applications, such as social games.

In addition, the synthetic graphs also have similar degree distribution as the original graphs. Due to space limitations, we only show our results for week 2009-34, 2009-45, and 2010-14 in Fig. 11. The goodness of fit (R-squre) for in-degree and out-degree distributions are above 0.98 and 0.88 in all of the three weeks, indicating good matches.

7. DISCUSSION AND FUTURE WORK

Using insights gained from unique longitudinal (64-140 weeks) user activity data from three popular Facebook gifting applications, we study and model microscopic user behaviors that give rise to the long-term evolution of user activity graphs (UAGs) for OSN-based applications. We demonstrate that our model can be used to predict the number of active user population in both gifting and non-gifting OSN applications that employ the same underlying growth mechanism (i.e., cost-free user recruitment by sending ARs)for recruiting new users. We evaluate our model using Facebook application data shared by application developers as well as publicly available statistics from application analytic web site, AppData. Lastly, we also show the utility of our model in estimating some of the required input parameters for generating synthetic snapshots of the UAG evolution process. Our results show that the synthetic weekly graphs can mimic several structural properties of UAGs derived from our empirical data.

While our work provides a big step forward compared to prior work that can only model *static snapshots* of UAG graphs, our current growth model has some limitations that are worth exploring further. First, our synthetic graphs only represent the graph size, connected component, clustering coefficient, and degree distribution of the empirical UAG snapshots. Existing studies [2, 8] have pointed out that a proper suite of metrics for social network analysis should also contain properties such as joint degree distributions, and community characteristics. In order to mimic these properties, we need a more sophisticated mechanism to determine the destination of each new edge when it is generated in our model. Second, our current model does not account for several external factors that might influence the evolution of UAGs, such as seasonal effects (e.g., holidays). Lastly, we are unable to model applications that employs multiple growth mechanisms, such as the popular social gaming applications. Modeling social games, however, is a more challenging task due to the high variability in game design and mechanics' complexity [8]. Nevertheless, we believe our work provides a general methodology for modeling UAGs from other genre of applications by considering the most popular growth mechanism (i.e., the cost-free invitation that is used in all applications), and will motivate further research into modeling evolution of UAGs on OSN platforms.

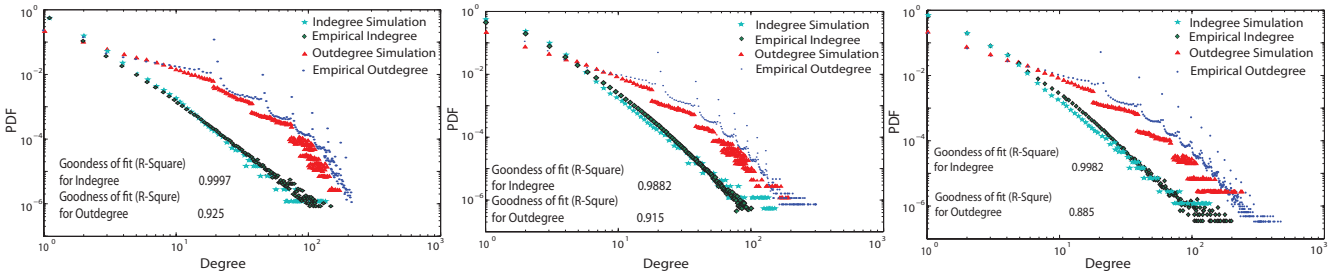
8. ACKNOWLEDGMENTS

This work was supported in part by Google Research Award. In addition, the authors would like to thank Stratis Ioannidis for his suggestions on the state transition model.

9. REFERENCES

[1] J. Ugander, B. Karrer, L. Backstrom, and C. Marlow, "The anatomy of the facebook social graph," *Arxiv preprint arXiv:1111.4503*, 2011.

[2] A. Sala, L. Cao, C. Wilson, R. Zablit, H. Zheng, and B. Zhao, "Measurement-calibrated graph models for



(a) Week 34, 2009 (b) Week 45, 2009 (c) Week 14, 2010

Figure 11: Evaluation of the degree distribution of synthetic graphs.

social network experiments," in *Proceedings of WWW*, pp. 861–870, ACM, 2010.

[3] A. Vázquez, "Growing networks with local rules: Preferential attachment, clustering hierarchy and degree correlations," *Arxiv preprint cond-mat/0211528*, 2002.

[4] M. Torkjazi, R. Rejaie, and W. Willinger, "Hot today, gone tomorrow: on the migration of myspace users," in *Proceedings of WOSN*, pp. 43–48, ACM, 2009.

[5] C. Wilson, B. Boe, A. Sala, K. Puttaswamy, and B. Zhao, "User interactions in social networks and their implications," in *Proceedings of ACM ECCS*, pp. 205–218, Acm, 2009.

[6] B. Viswanath, A. Mislove, M. Cha, and K. Gummadi, "On the evolution of user interaction in facebook," in *Proceedings of WOSN*, pp. 37–42, ACM, 2009.

[7] M. Valafar, R. Rejaie, and W. Willinger, "Beyond friendship graphs: a study of user interactions in flickr," in *Proceedings of WOSN*, pp. 25–30, ACM, 2009.

[8] A. Nazir, S. Raza, and C. Chuah, "Unveiling facebook: a measurement study of social network based applications," in *Proceedings of IMC*.

[9] A. Nazir, A. Waagen, V. Vijayaraghavan, C. Chuah, R. Souza, and B. Krishnamurthy, "Beyond friendship: Modling user activity graphs on social network based gifting applications," in *Proceedings of IMC*, 2012.

[10] "appdata.com." `http://www.appdata.com`.

[11] M. Newman, "The structure and function of complex networks," *SIAM review*, pp. 167–256, 2003.

[12] J. Leskovec, J. Kleinberg, and C. Faloutsos, "Graphs over time: densification laws, shrinking diameters and possible explanations," in *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pp. 177–187, ACM, 2005.

[13] N. Gong, W. Xu, L. Huang, P. Mittal, E. Stefanov, V. Sekar, and D. Song, "Evolution of attribute-augmented social networks: Measurements, modeling, and implications using google+," in *Proceedings of IMC*, 2012.

[14] M. Allamanis, S. Scellato, and C. Mascolo, "Evolution of a location-based online social network: Analysis and models," in *Proceedings of IMC*, 2012.

[15] D. Braha and Y. Bar-Yam, "Time-dependent complex networks: Dynamic centrality, dynamic motifs, and cycles of social interactions," *Adaptive Networks*, pp. 39–50, 2009.

[16] P. Holme and J. Saramäki, "Temporal networks," *Arxiv preprint arXiv:1108.1780*, 2011.

[17] R. Breiger, K. Carley, and P. Pattison, *Dynamic social network modeling and analysis: Workshop summary and papers*. Natl Academy Pr, 2003.

[18] S. Hanneke and E. Xing, "Discrete temporal models of social networks," *Statistical network analysis: Models, issues, and new directions*, pp. 115–125, 2007.

[19] H. Jo, R. Pan, and K. Kaski, "Emergence of bursts and communities in evolving weighted networks," *PloS one*, vol. 6, no. 8, p. e22687, 2011.

[20] S. Bansal, J. Read, B. Pourbohloul, and L. Meyers, "The dynamic nature of contact networks in infectious disease epidemiology," *Journal of Biological Dynamics*, vol. 4, no. 5, pp. 478–489, 2010.

[21] M. Karsai, M. Kivelä, R. Pan, K. Kaski, J. Kertész, A. Barabási, and J. Saramäki, "Small but slow world: How network topology and burstiness slow down spreading," *Arxiv preprint arXiv:1006.2125*, 2010.

[22] F. Guo, S. Hanneke, W. Fu, and E. Xing, "Recovering temporally rewiring networks: A model-based approach," in *Proceedings of ICML*, pp. 321–328, ACM, 2007.

[23] M. Morris and M. Kretzschmar, "Concurrent partnerships and transmission dynamics in networks," *Social Networks*, vol. 17, no. 3-4, pp. 299–318, 1995.

[24] K. Zhao, J. Stehlé, G. Bianconi, and A. Barrat, "Social network dynamics of face-to-face interactions," *Physical Review E*, vol. 83, no. 5, p. 056109, 2011.

[25] E. Volz, "Sir dynamics in structured populations with heterogeneous connectivity," *Arxiv preprint physics/0508160*, 2005.

[26] E. Volz and L. Meyers, "Susceptible-infected-recovered epidemics in dynamic contact networks," *Proceedings of the Royal Society B: Biological Sciences*, vol. 274, no. 1628, pp. 2925–2934, 2007.

[27] C. Kamp, "Untangling the interplay between epidemic spread and transmission network dynamics," *PLoS Computational Biology*.

[28] M. Newman, "Power laws, pareto distributions and zipf's law," *Contemporary physics*, vol. 46, no. 5, pp. 323–351, 2005.