Poking Facebook: Characterization of OSN Applications

Minas Gjoka, Michael Sirivianos, Athina Markopoulou, Xiaowei Yang
University of California, Irvine
{mgjoka, msirivia, athina, xwy}@uci.edu

ABSTRACT

Facebook is one of the most popular Internet sites today. A key feature that arguably contributed to Facebook's unprecedented success is its application platform, which enables the development of third-party social-networking applications. Understanding how these applications are installed and used is important for the function and utility of web-based online social networks, e.g. to better engineer them and/or to design advertising campaigns.

In this paper, we characterize the popularity and user reach of Facebook applications. We analyze application usage data gathered over a period of six months from Facebook and Adonomics - a Facebook analytics service. We also crawl publicly accessible Facebook user profiles and obtain per-user application installation statistics, for approximately 300K users and 13.6K applications. Our findings include that (i) the popularity of Facebook applications has a highly skewed distribution; (ii) although the total number of application installations increases with time, the average user activity decreases; and (iii) users with more applications installed are more likely to install new applications.

Categories and Subject Descriptors

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

General Terms

Measurement

Keywords

Online Social Networks, Facebook, Applications, Characterization

1. INTRODUCTION

Web-based Online Social Networks (OSN), such as MySpace and Facebook (FB), are quickly emerging as a new Internet killer-application. We can view OSNs as natural extensions of Internet applications that establish relationships between users, such as email and IM. However, unlike those applications, OSNs not only facilitate direct communication between users but also allow them to post content that revolves around their profiles creating online personas that typically map to their real life personalities. In addition, OSNs explicitly expose a user's social contacts, enabling users to browse each other's social networks in search of common friends and interesting content. FB in particular, has approximately 67M users, while the total number of users in all other popular OSNs combined is around 270M.

In a successful attempt to enhance user experience and increase the site's appeal, in May 2007, FB made a key innovation: they opened their platform to third-party developers [3]. Developers are now able to create FB applications that augment FB's functionality or act as front-end to third party web-based services. The FB application paradigm is unique because the risk of development and promotion investment in third party applications is smaller than the risk of investing in stand-alone web applications. This is due to the simplicity of the FB application API and to the inherent capabilities of the platform. In particular, FB notifies its members about the applications their friends install and use, and applications themselves prompt the user to invite friends to install them.

In mid-February 2008, there were approximately 866M installations of 16.7K distinct FB applications, and 200K developers are utilizing the platform. As of today, more than 100 OSN application development companies have been founded and FB application based advertising campaigns have been surprisingly successful [1].

Motivated by this unprecedented success, we became interested in studying the popularity (both its distribution and change over time), and adoption dynamics (how applications are installed by users) of FB applications. An in-depth understanding of these characteristics is important both for engineering and marketing reasons. Understanding the popularity and adoption dynamics can assist advertisers and investors to form strategies for acquiring applications or purchasing advertising real-estate, so as to reach a large portion of the targeted user-base at a low cost. At the same time, determining which applications tend to attract more users, can help to better engineer the applications' user interface and features and to better provision the OSN system.

For this study, we collected and analyzed two data sets pertaining to FB applications. The first data set consists of data obtained from Adonomics [4], a service based on statistics reported by FB [2], for a period of 6 months from Sept. 2007 until Feb. 2008. It provides the number of installations for each application and the number of distinct users that engage each application at least once during a day, called *Daily Active Users (DAU)*. The second data set consists of a sample of publicly available FB user profiles, crawled for 1 week in Feb. 2008, pertaining to approximately 300K users and

13.6K applications. Based on the above data, we are interested in the following questions.

Aggregate Application Popularity. We first ask whether the continuous growth of FB applications' install base translates to increasing user engagement. We find that although the total number of installations increased linearly over the entire 6-month period, the total number of daily active users increased in the first three months but subsequently dropped and eventually stabilized.

Popularity of Individual Applications. A natural question is how skewed is the distribution of popularity across different applications. We examine the popularity of applications in terms of DAU and number of installations and we find that the distributions of both metrics are highly skewed. The next question is whether popularity depends on the type of application. To this end, we classify FB applications based on their functionality and identify the most popular categories and applications in terms of DAU.

User Coverage. The popularity of applications in itself does not tell us much about how applications are distributed among users. More detailed information is needed if user coverage is of interest, i.e. given a set of applications how many unique users have installed one or more applications from that set. For example, an advertiser may attempt to increase the reach of a campaign by acquiring two popular applications. However, this reach is diminished if there is significant overlap in the set of users that have installed these applications. To this end we use the crawled data, which essentially represent a bipartite graph of users and the applications they have installed. We derive statistics about this graph and user coverage thereof. We simulate and validate a "preferential installation" process, according to which the probability of a user installing a new application is proportional to a power of the number of applications she has already installed. The simulation takes as input the applications, their popularity, and the number of users. It outputs which applications each user has installed.

Our work makes the following contributions. We present the first study to characterize the statistical properties of OSN applications. Previous work focused on the characteristics of the social graph itself [5–7, 11, 12] or the popularity of user-generated content [8]. To the best of our knowledge, any formal characterization of FB's statistics, in general, has yet to be made available. In addition, we propose a simple and intuitive method to simulate the process with which users install applications. Using this method one can determine the user coverage from the popularity of applications, without detailed knowledge of how applications are distributed among users.

The rest of this paper is organized as follows. In Section 2, we describe our methodology for collecting FB application data and we summarize the data sets. In Section 3, we use the data to characterize important statistics regarding the popularity of FB applications. In Section 4, we provide a model of the application installation process and discuss its possible use as a user coverage computation tool.

2. DATA SETS AND COLLECTION

This paper is based on two data sets pertaining to third-party applications, which are summarized in Table 1. (Because we are interested in third-party applications, we exclude Facebook's in-house applications, such as "Groups", "Gifts", "Photos" etc.) In the rest of this section, we describe the collection processes we used for obtaining these data sets. But first let us give some brief background on how the FB platform works.

2.1 Background on the Facebook Platform

The FB Platform is a standards-based web service with methods

Data Set	Source	Period	Data Element
I	Adonomics (FB analytics)	08/29/07- 02/14/08	(date, application, # installations, # active users)
II	FB Public User Profiles	02/20/08- 02/27/08	(user, list of applications)

Table 1: Data Sets under Study

for accessing and contributing FB data [3]. It comprises the following parts: (i) the FB API, which enables developers to add social context to their application by utilizing data regarding the user's profile, its friends, its pictures, and events; (ii) the FB Query Language (FQL), which resembles an SQL interface to access the same data that one can access through the FB API; iii) Markup (FBML) and Java Script (FBJS), which allow developers to build applications that integrate into a user's FB experience through the Profile, Profile Actions, Canvas, News Feed and Mini-Feed.

2.2 Data Set I: Crawling Facebook Analytics

Data Set I consists of the daily number of installations and daily active users (DAU) for every application, for every day of a 170 day period. It was obtained by crawling the Adonomics [4] data sets. The rationale is as follows.

Collection Process. Facebook reports application statistics in its application directory [2]. It employs an application ranking system that is based on user engagement. From 12:00am to 11:59pm each day, they measure how many distinct users *engaged* the application at least once, i.e. performed one of the following actions: a) view its Canvas; b) clicked on FBML links; c) performed an AJAX form submission; and d) activated a click-to-play Flash. FB expresses DAU as an integer percentage of the total number of installations, which can be used to extrapolate the total number of installations of an application.

Facebook does not report historical data on DAU and installations, only statistics for the current day. On the other hand, Adonomics [4], continuously processes the above statistics page to provide Facebook daily statistics and analysis over long periods of time. In order to determine the reliability of the statistics reported by Adonomics, we randomly sampled their statistics on DAU and cross-referenced it with what Facebook reported in its own application directory during February 2008. We confirmed that their application statistics were the same as in the original FB reports. Therefore, in the scope of this paper, and to allow for a rapid analysis over a longer period of time, we decided to scrape Adonomics instead of the raw FB Application Directory.

2.3 Data Set II: Crawling User Profiles

We also crawled a sample of publicly available FB user profiles for 1 week in Feb. 2008 and logged the applications that each user had installed. This constitutes our *Data Set II*.

Collection Process. Crawling Facebook is a difficult task because FB defends against automated data mining and users have a limited view of their social graph. As a result, we were severely constrained in the scope of our sampling. FB consists of many networks, each formed around a region, workplace, academic institution or high school. Each user can be a member of at most 2 FB networks at a time, and can change networks only two times every 60 days. With default privacy settings, a user can browse only her friends profiles as well as the profiles of other users in the same network.
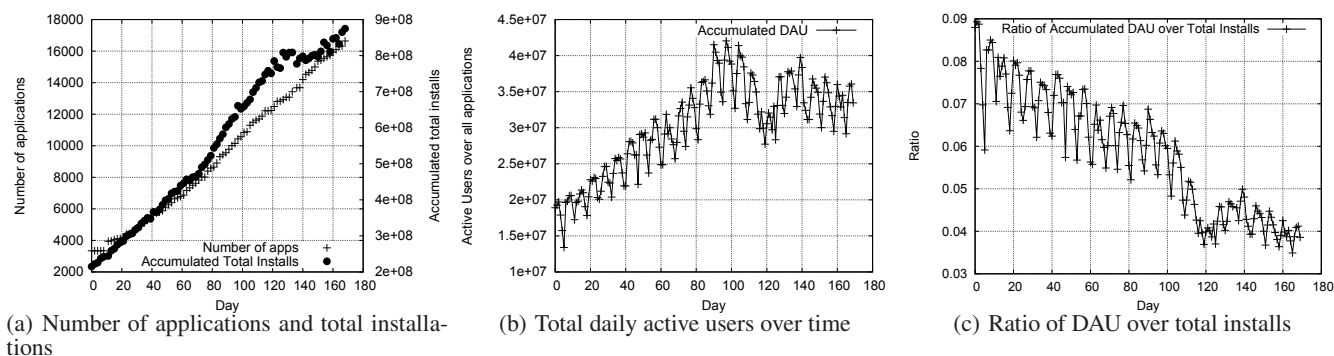
(a) Number of applications and total installations

(b) Total daily active users over time

(c) Ratio of DAU over total installs

Figure 1: Evolution of Facebook applications in aggregate.

To partly work around these constraints and efficiently obtain user-specific information, we created 20 FB user accounts. Each user joined a geographical network (US, Canada, UK, France, Greece, Mexico, India and Australia) and repeatedly used a feature provided by FB to "display 10 random people from network". By repeatedly requesting 10 users from a specific user account in a specific network, we were able to mine the profiles of 6K-7K distinct users at each network after approximately 2K-2.5K requests for "random" people. Furthermore, we crawled the profiles of the friends of those distinct users, acquiring 20K to 60K additional users at each network. We note that sampling FB by randomly generated IDs is not a plausible technique, because FB IDs are not correlated with user networks.

We automate all the above procedures using Python scripts, which we make available online [10]. We note that all our 20 accounts were eventually banned by FB due to excessive activity. In total, we crawled approximately 300K users that had installed 13.6K applications in total.

Limitations. Our crawling technique has some limitations that stem mainly from the relatively restrictive FB data access policies. First, our sampling methodology currently misses profiles of users that restrict access by other users in the same FB network. Second, we do not capture privacy-conscious users that choose not to place any or all applications in their profile. Third and most important, it is unclear whether our currently crawled sample is representative of the whole Facebook. This is due to the fact that the inner-workings of the "10 random users" feature are unknown. In addition, by sampling the friends of the "randomly" returned users, it is possible to skew the distribution of application installations; e.g. some applications may be more popular among a group of friends than in the entire facebook. We address this concern in Section 3.4, where we provide evidence that Data Set II is sufficiently representative, at least for the application properties of interest in this paper. We are currently working on addressing these limitations, by crawling an order of magnitude larger data set and by reverse engineering the "10 random users" feature.

3. DATA ANALYSIS

In this section, we analyze the two data sets and provide statistics about the popularity of FB applications.

3.1 Aggregate FB Application Statistics

We start by looking at all third-party Facebook applications together as an aggregate. Fig. 1(a) shows that the total number of applications and the total number of installations increases almost linearly over the 170 days of Data Set I.

We then look at how many of these installed applications are actually active. Fig. 1(b) plots the number of active users over time: initially this number follows the growth of total installations, indicating that most users engage their installed applications. However, after day 104 the number of active users saturates at 42M and subsequently drops and stays at around 35M. In Fig. 1(c), we look at the ratio of total daily active users (from Fig. 1(b)) over the total installations per day (Fig.1(a)). This ratio continuously decreases from 9% down to 4%. This indicates that an increasing number of applications competes for user attention that has plateaued.

Another observation from Fig. 1(b) and (c) is the weekly usage pattern. Although not clearly visible in these plots, our data analysis reveals that Tuesday and Wednesday were typically the most active and Saturday and Sunday were the least active days of the week.

3.2 Popularity of Individual Applications

We now turn our attention from FB as an aggregate to individual applications. We find that the popularity distribution is highly skewed; this is expected since popular applications tend to be more visible and solicit more invitations. Notice however that, as discussed above, the results can be quite different depending on whether we consider the number of *installations* or the number of *daily active users (DAU)* as the measure of popularity.

Fig. 2 shows the distribution of number of installations per application as found in Data Sets I and II. 10% of the top ranked applications account for 98% of total installations. The distribution of installations per application is approximated by a log-normal distribution with mean $\mu = 7.13$ and standard deviation $\sigma = 2.63$.

The distribution of DAU per application is shown in Fig. 3. We use the techniques presented in [9] to fit a power-law distribution to the data. In Fig. 3, a power-law with parameter $\alpha = 1.57$ seems to best approximate the distribution among other known distributions we tried (including exponential, Weibull, and log-normal).[1]

The main observation is that the popularity distribution is highly skewed, with regards to both metrics of popularity. However, un-

---

[1] We use the discrete maximum likelihood estimator to compute the fitting power-law scaling parameter α, along with the Kolmogorov-Smirnov-based approach to estimate the lower cutoff x_{min} for the scaling region. We also use the Kolmogorov-Smirnov statistic D to compute the goodness-of-fit and the Kolmogorov-Smirnov test to compute a p-value for the estimated power-law fit. The estimated fit gives parameters $\alpha = 1.57$, $x_{min} = 514$, $D = 0.0418$. We find that excluding the applications with less than $x_{min} = 514$ DAU, 20% of the top ranked applications account for 89.6% of total user engagements. However, we find that $p = 0$, which means that the power-law hypothesis is rejected in the strict statistical sense.

derstanding the underlying process that leads to this property and finding the appropriate distribution fit is part of ongoing work.
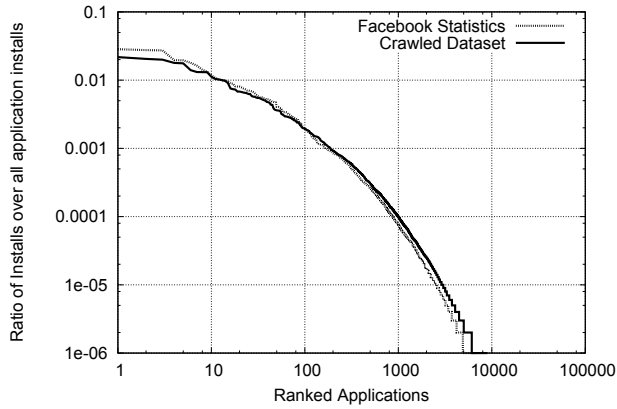

Figure 2: PDF of total installations per application in Data Set I (FB statistics) and II (crawled dataset).
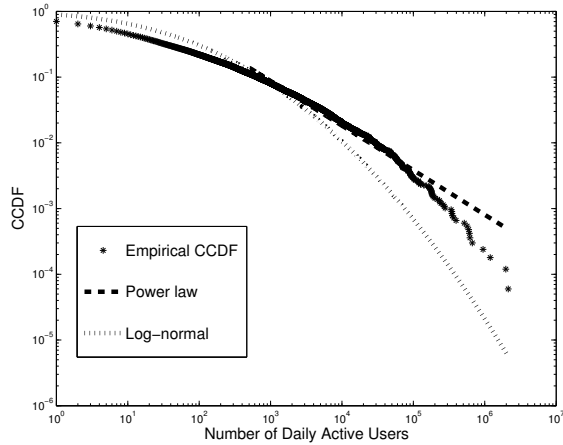

Figure 3: CCDF of DAU per application.

3.3 The Effect of Application Category

There are several factors that may affect the popularity of an application. One such factor seems to be the type/category of application. In this section, we classify applications in thematic categories and look closer at the statistics and evolution of particular categories.

Due to space limitations, we list only the 7 most popular categories. We call the first category "Friend Comparison": it includes applications that allow users to declare best friends and compare friend traits. The second category is "Casual Communication", which includes applications that allow users to exchange messages and write on each other's wall. We call the third category "Rating, Taste Matching and Recommendations": it enables users to review, compare and recommend items spanning from music to restaurants. The fourth category is "Gestures": it includes applications that allow users to perform virtual gestures (poke, bite to convert to a zombie etc). The fifth category is called "Self Expression" and enables users to express moods, political opinions etc. The sixth category is "Gifting", enabling users to exchange virtual gifts. The
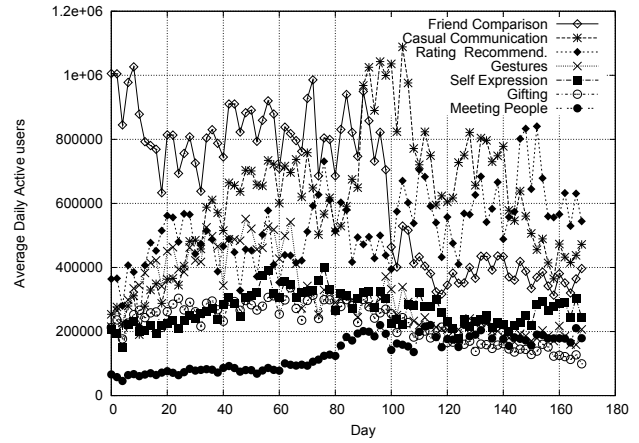

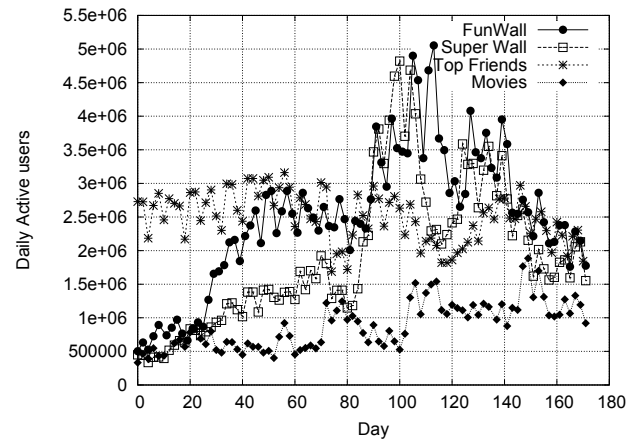**Figure 4: Average DAU per application category. Categories are listed in the order of their total DAU rank.**

Figure 5: Daily active users of top 4 applications.

seventh category is "Meeting People", which is of particular interest to online dating services.We are now ready to look closer at the average popularity of application categories and at the popularity of individual applications. In particular, we are interested in the most popular among them.

First, we ask whether popularity depends on the application category. We select the 100 most active applications on 03/05/2008 and we classify them in the aforementioned categories. We compute the average DAU of the applications in each category for every day in a 170-day period. In Fig. 5 we include the 7 top categories ranked according to the sum of their average DAU over all days. It turns out that the "Friend Comparison" and "Casual Communication" categories are the most popular. This indicates that Facebook best serves the need of users to declare how they feel about their friends and the need to exchange messages. Interestingly, the fourth most popular category, "Gestures", was initially very popular but was subsequently matched by the "Self Expression" and "Meeting People" categories. A possible explanation is that the initial craze with applications of the likes of "Vampires", eventually turned into annoyance, while the applications in the other categories became more useful. We also observe that, in general, the user activity of the top ranked categories follows the same trends as the total user activity shown in Fig. 1(b): activity peaks at around the same time (95th-105th) day and drops afterwards.

Second, we looked at the 5 most popular individual applications every day in the 170-day period. Interestingly, there were only 17 unique applications among them, which is much smaller than the number $5 \cdot 170 = 850$ that would correspond to a different top-5 popular applications every day. This indicates that the most popular applications remain popular throughout the entire period. Furthermore, we observed that 16 out of these 17 applications were present from the beginning of the period. Fig 5 shows the DAU evolution over time for the four most popular applications. We observe that all four applications belong in the three most popular categories. It is notable that the two most popular applications, which belong in the "Casual Communications" category, exhibited impressive viral growth. E.g. "Super Wall" grew from ~1.1M to ~4.8M DAU in 20 days.

3.4 Data Set I as a Sample of Data Set II

On one hand, Data Set II contains more detailed information (users and their installed applications) than Data Set I (popularity of applications). On the other hand, Data Set II is based on a small sample of crawled users with public profiles compared to Set I. An interesting question is whether II is a representative sample of I.

We have three indications that this is indeed the case. First, the distribution of application installations follows the same distribution as in the complete network, as shown in Fig. 2. This allows us to answer questions regarding the user coverage of applications in FB (in Section 4), based on a smaller sample of the user base. Second, Data Set II has ~13.6K distinct applications, which matches the applications with $\frac{DAU}{\#Installations} > 0\%$ in all Facebook from Dataset I. Third, we found that the top 50 most installed applications are common in both sets.

4. USER COVERAGE SIMULATION

In this section, we develop a simulation model that generates the bipartite graph between the users and the applications installed. The input to the simulator is the list of applications, the number of installations per application and the number of users. The output of the simulator is the bipartite graph (which we cannot obtain without crawling). Based on this graph we can compute several metrics of interest such as: the distribution of number of applications installed per user, the number of applications needed to cover all users, etc.

Such a simulator would be useful to those interested in reaching users via applications, such as advertisers. For example, an advertiser might be interested in which applications she should purchase to cover a certain set of users; other constraints, such as cost, could also be taken into account in these optimization problems. User coverage strategies can be studied if the bipartite graph is available. However, this is not a trivial task in practice. No Facebook analytics service offers statistics that can be directly used to infer the coverage of more than one application. In addition, privacy-conscious FB application operators are likely not to release information on which users have installed their applications. On the other hand, developers and Facebook already release statistics about application usage and user demographics. This can be then used as input to our simulator, in the absence of detailed crawled data.

4.1 Preferential Installation

The skewed distribution of application popularity motivated us to investigate whether rich-get-richer types of mechanisms can apply to the process according to which individual users install applications. At the heart of our simulator lies a preferential installation process, according to which users that have already several applications installed are more likely to install even more new applications.

In particular, our simulation proceeds as follows. Consider the
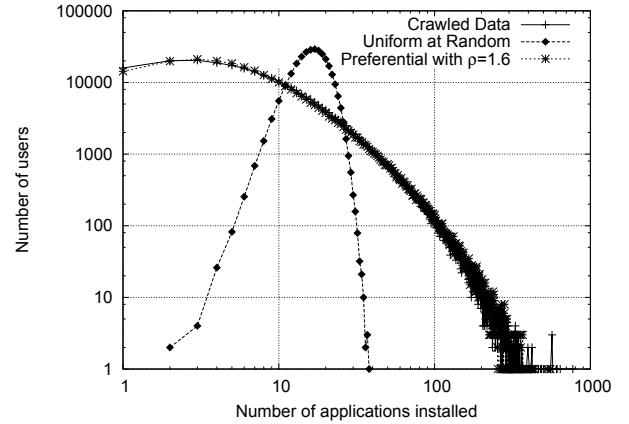
Figure 6: Distribution of per-user number of installed applications in the crawled data (Data Set II) and in the simulation-generated data.

users as bins and the applications as balls of different colors. The number of installations of an application are considered as balls of the same corresponding color. At each step of the simulation, a ball is selected, starting from the color (application) with the most balls (installations), and proceeding to the next color once all balls of the current color are exhausted. Once a ball is selected, it must be assigned with a certain probability to one of the bins that does not contain a ball of the same color (assuming that a user installs a certain application only once).

The probability that a ball chooses a certain bin specifies the behavior of the installation model and eventually the statistics we will observe. For example, a ball could choose uniformly at random among the bins; this turned out to be a very bad model for our data, as shown in Fig. 6. We then explored preferential installation models that assign more probability to bins that have already several balls; this captures the intuition that a user that has already several applications installed is more prone to install new applications as well. In particular, the probability of a ball (application) to be installed at a bin (user) i is calculated as

$$P_{bin}(i) = \frac{balls(i)^\rho + init}{\Sigma_{j \in B}(balls(j)^\rho + init)} \qquad (1)$$

where $balls(i)$ is the number of balls that bin i contains prior to this installation, B is the set of bins without a ball of the same color and ρ is an exponent that can magnify the effect of preferential attachment. The parameter $init$ defines the initial probability $P_{bin}(i)$ of a bin without any installations, and controls the significance of the number of balls in a bin in the early steps of the simulation. In the 1st iteration of the simulation, all bins are equally likely to be chosen w.p. $1/|B|$; in the 2nd iteration the ones with already 1 ball have $\frac{init+1}{init}$ times higher probability than the ones with 0 balls.

After careful tuning of the parameters ($\rho = 1.6$ and $init = 5$) this process results in the same statistics as those found in Data Set II. Fig. 6 shows the distribution of the number of installations per user as found in Data Set II. It also shows that a very similar distribution is generated by our simulation model with the above parameters and random order of application installations. We also ran simulations in which applications are installed in increasing or decreasing order of popularity and observed similar results.

Limitations. Our simulator captures accurately the preferential installation of applications to users with already installed applications. However, it does not currently capture other factors that may
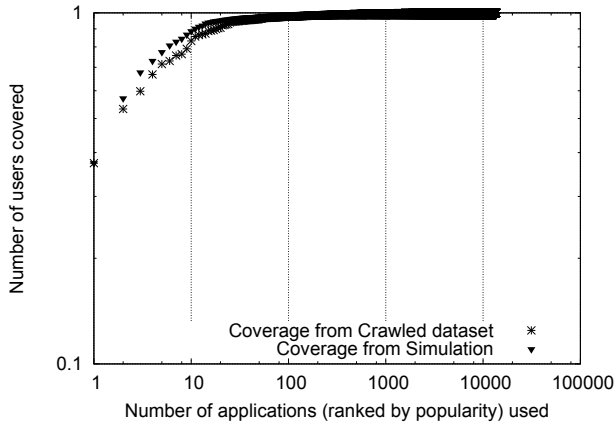
Figure 7: User coverage. The x axis is fraction of applications (ordered in decreasing popularity). The y axis is the fraction of users that are covered.

Application Popularity Rank	# Installations	Coverage Real(%)	Coverage Simulat.(%)
5	87609	30.2	30.2
15	45396	41.6	39.8
46	19504	43.9	42.6
99	9685	44.9	43.6
12	50825	51.5	51.1

Table 2: User coverage statistics. The first column corresponds to the rank of (five randomly chosen) applications according their number of installations in Data Set II.

affect the probability of installing an application such as: (i) demographics the user belongs to; (ii) previously installed applications, e.g. with similar function; (iii) installations of the user's friends. Furthermore, our current model is based on the number of installations not DAU. We plan to extend and refine this model in future work to include these considerations.

4.2 User Coverage Analysis

For validation, we compare user coverage statistics obtained by the model and the crawled data set (we have determined in Section 3.4 that the crawled data set is sufficiently representative of Facebook). The simulation can be used to predict user coverage of a target demographic given the number of installations per application. Fig. 7 shows a close match between user coverage in the crawled data and in the output of our simulator when run for the same popularity distribution. (The PDF is shown in Fig. 2: it is the same for the sample, Data Set II, and the full Data Set I and can be modeled well as log-normal.) Table 2 shows another example of how our simulator can be used to study user coverage. We randomly selected 5 applications and looked at their user coverage, which was 51.5% of all users in Data Set II and 51.1% in the simulation-generated data. Such information would be particularly useful to an advertiser. For example, the two applications that rank 1 and 2 may cover 50% of the users in the sample but they may cost a lot more to acquire than all 5 listed applications combined. We repeated this process 50 times, randomly selecting 5-20 applications and found that the sample coverage was within $\pm 4\%$ of the simulated coverage.

5. CONCLUSIONS

We have presented the first measurement-based characterization of the popularity and usage of third-party Facebook applications. We plan to extend this work with additional datasets, improved models, and study of more dynamic aspects such as application virality on the social graph.

6. REFERENCES

[1] Business Week. Building a Brand with Widgets. http://www.businessweek.com/technology/content/feb2008/tc20080303_000743.htm.

[2] Facebook application directory. http://www.facebook.com/apps.

[3] Facebook platform. http://developers.facebook.com/.

[4] Adonomics. http://www.adonomics.com, 2008.

[5] L. A. Adamic, O. Buyukkokten, and E. Adar. A Social Network Caught in the Web. In *First Monday*, 2003.

[6] Y. Ahn, S. Han, H. Kwak, S. Moon, and H. Jeong. Analysis of Topological Characteristics of Huge Online Social Networking Services. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, 2007.

[7] L. Backstrom, D. Huttenlocher, J. Kleinberg, and X. Lan. Group formation in large social networks: Membership, growth, and evolution. In *KDD '06: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2006.

[8] M. Cha, H. Kwak, P. Rodriguez, Y.-Y. Ahn, and S. Moon. I Tube, You Tube, Everybody Tubes: Analyzing the World's Largest User Generated Content Video System. In *IMC '07: Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, 2007.

[9] A. Clauset, C. R. Shalizi, and M. E. J. Newman. Power-law Distributions in Empirical Data. In http://arxiv.org/abs/0706.1062v1, Jun 2007.

[10] M. Gjoka. Scripts for Crawling Facebook. http://www.ics.uci.edu/~mgjoka/facebook/.

[11] R. Kumar, J. Novak, and A. Tomkins. Structure and evolution of online social networks. In *KDD '06: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2006.

[12] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and S. Bhattacharjee. Measurement and Analysis of Online Social Networks. In *IMC '07: Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, 2007.