The Freshman Handbook: A Hint for the Server Placement of Social Networks

Ying Zhang
Ericsson Research
ying.zhang@ericsson.com

Mallik Tatipamula
Ericsson Research
mallik.tatipamula@ericsson.com

ABSTRACT

There has been a recent unprecedented increase in the use of Online Social Networks (OSNs) to expand our social life, exchange information and share common interests. Many popular OSNs today attract hundreds of millions of users who share tremendous amount of data on it such as Facebook, Twitter, and Buzz. Given the huge business opportunities OSNs may bring, more and more new social applications has emerged on the Internet. For these newcomers in the social network business, one of the first key decisions to make is to where to deploy the computational resources to best accommodate future client requests. In this work, we aim at providing useful suggests to the new born social network providers (freshman) on the intelligent server placement, by exploring available public information from existing social network communities. In this work, we first propose three scalable server placement strategies for OSNs. Our solution can scalably select server locations among all the possible locations, at the same time reducing the cost for inter-user data sharing.

Categories and Subject Descriptors

C.2.4 [**Distributed applications**]: Social network

General Terms

Performance

Keywords

Server placement, Social network

1. INTRODUCTION

A social network is a set of people connected by a set of social relationships, such as friendship, co-working or information exchange. There has been a recent unprecedented increase in the use of Online Social Networks (OSNs) to expand our social life, such as finding others of a common interest. The OSNs have become a large scale distributed system providing services to hundreds of millions of users and delivering messages at very high rate, *e.g.,* Facebook and Twitter. Besides the traditional client-to-server requests, OSNs need to handle highly interconnected data due to the strong community structure and human relationships among their end users, which often result in complex data sharing among users. Given the tremendous user population and frequent data access, effective resource planning and provisioning strategies are of extreme importance to the performance and revenue of an OSN. In particular, selecting the most suitable locations to deploy server farms is one of the key steps in resource management.

WWW 2011, March 28–April 1, 2011, Hyderabad, India.
ACM 978-1-4503-0637-9/11/03.

Most proposals on the server placement problem rely on extracting clients' requests from history traces collected on the web servers, and then search for the best placement given the particular client and load distribution [1, 2, 3, 4, 5, 6, 7]. In this paper, we aim at solving a slightly different problem, *i.e.,* how shall a new born Internet application service make the decision on where to deploy servers? For instance, where would Mark Zuckerberg deploy the server farms or data centers to host Facebook content when it was first launched in 2004? At the first glance it is unsolvable as the future communication is unpredictable. The best one can do is to choose large cities and metropolitan areas according to population density. Such statement may be true for general web applications. However, in this work, we propose a hypothesis that *by best utilizing public information from existing established Online Social Networks(OSN), we can obtain sufficient information to provide intelligent server placement suggestions to the new born social network applications*. We provide useful hints for the freshmen in the OSN community to solve the server placement problem.

In this paper we first devised three scalable server placement algorithms optimizing for different objectives. Using these algorithms, we make a qualitative analysis of 4 popular Internet social networks of quite different applications and thus may exhibit different distribution trajectories. We conduct complete analysis on the server placement problem for four different OSNs, *i.e.,* Amazon review system, Buzz, Twitter, and Flickr. Despite the detailed difference in application types, we identify significant commonality between OSNs in the selected set of deployment locations. We further conduct joint analysis by combining information from all OSNs to provide a single suggestion for newcomers. Though this study is motivated by providing placement hints to freshman, the methodology is general for all OSNs.

First, we propose a scalable server placement algorithm based on graph partitioning. Our key idea is to employ clustering techniques that partition the whole client space into non-overlapping groups, where all the IP addresses in a group are topologically close to the centroid, would indicate the best suitable locations to minimize user to server latency. Second, we develop an effective server selection algorithm taken inter-user sharing into consideration. We are the first to comprehensively study the commonality between the server deployment solutions among popular OSNs. And finally we are the first to explore and demonstrate the methods of utilizing publicly available social network data to assist resource provisioning issues for future businesses.

2. APPROACH

We sketch the components of our work in Figure 1. It consists of data pre-processing process, placement engine, and a post-analysis module. The input is a set of user profiles and the list of friends for each user. This work use all the user profiles gathered by crawl-

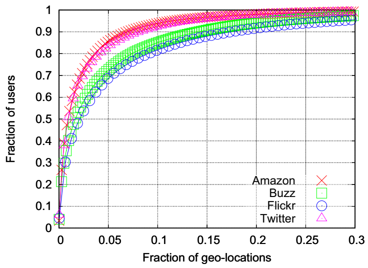

Figure 2: User distribution in different geographic locations

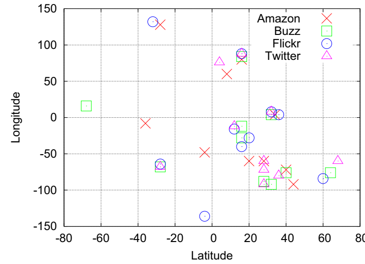

Figure 3: 10 server locations

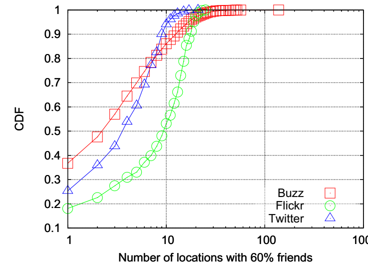

Figure 4: Number of locations contributing 60% friends

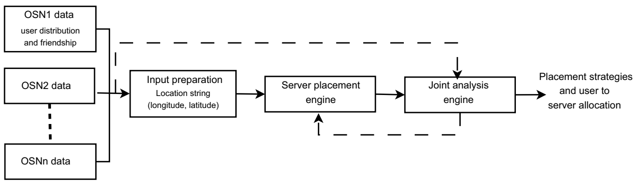

Figure 1: Architecture

ing the OSNs. However, our methodology can also work for the sampled data with representative distributions. The next step is to construct a latency map between any pair of users. The ideal data would be the round-trip delay collected by all users using measurement tools such as ping or traceroute. But due to the limited access to end hosts as well as privacy concerns, it is unlikely to obtain real delay in such a large scale. Existing network measurement platforms, *e.g.,* Plantlab is far from the scale required. Thus, we decide to use hypothetical direct link latency to approximate the latency between two users. Computing such hypothetical link latency requires the knowledge of geographic locations. We extract the location information from user profiles, pre-processes the data to correct typos, eliminate ambiguity, and combine the same locations with different expressions together, *e.g.,* California and CA. We use the Yahoo Geocoding API [8] to map the geographic string to the coordinates of longitude and latitude. The latter can be used to directly compute the distance and for grouping. The third module is a placement engine which generates multiple selection solutions depending on different input data and parameter configuration. For instance, the placement engine can generate results with cost function f defined above, or g, or both. Given the solutions from multiple OSNs, we conduct joint analysis to provide the best solution for the new OSN. The joint analysis combines data from multiple OSNs together. We study two approaches: performing joint analysis after the server selection for individual OSNs or incorporating the inputs from multiple OSNs first and then perform selection based on the joint input.

3. EXPERIMENT RESULTS

In the following we present the results from data analysis and server selection algorithms.

Minimizing user to server latencies

We first present the results on selecting servers to minimize client to server latency. Before diving into the selection, at the first step, we conduct analysis on the user distribution data to understand the user concentration according to geographic locations. We group the users based on its longitude and latitude to groups of every 2 degree groups. For instance, a user of geographic location (11.5,21.5) is grouped together with point (10,20). Figure 2 shows the number of users in each group for the four OSNs. As expected, it shows that a small fraction of locations contain most users. For instance, 70% of users reside in the 5% of locations in Flickr, possibly large cities and metropolitan areas. This result conveys two important pieces of messages: 1) there exist locality of user distribution; 2) there is strong similarity amongst four OSNs.

Now we directly run the server selection *OSN-KMEANS* algorithm to select k partitions for minimizing latency. Figure 3 shows the location of servers when $k = 10$. We can clearly observe 8 out of 10 number of locations are very close between Buzz and Twitter. For the spreading of the locations, Amazon has a slightly different distribution compared to the other three, which is likely caused by its slightly different application characteristics.

Minimizing inter-DC communications

Next, we study the impact of inter-user data sharing on server selections. Similar to the previous analysis, before finding the actual server selection solutions, we first conduct simple analysis on the friendship distribution to find if any common property exists. On the other hand, in order to understand the existence of interconnection between clients could be quite distant, we begin by characterizing the geographic diversity of clients in the data. Figure 4 shows that for any user, the number of locations that contain 60% of his friends. Intuitively, a person's friends may mostly reside in a few places such as his living or working place. Buzz and Twitter has similar trends, *i.e.,* most users have friends locating in only a few locations. Flickr has a much more spread friendship distribution. This is because flickr is a social platform for people who are interested in photography to share their work. Thus, people connected by such common interests are likely to locate in distant places in the world.

4. REFERENCES

[1] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee, "Measurement and analysis of online social networks," in *IMC '07: Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, pp. 29–42, 2007.

[2] Y.-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*, pp. 835–844, 2007.

[3] 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*, pp. 611–617, 2006.

[4] B. Li, X. Deng, M. J. Golin, and K. Sohraby, "On the optimal placement of web proxies in the internet: The linear topology," in *Proc. of the 8th IF IP Conference on High Performance Networking*, (Veinna), 1998.

[5] L. Qiu, N. V. Padmanabhan, and M. G. Voelker, "On the placement of web server replicas," in *Proc. of IEEE INFOCOM*, 2001.

[6] S. Jamin, C. Jin, A. R. Kurc, D. Raz, and Y. Shavitt, "Constrained mirror placement on the internet," in *Proc. of IEEE INFOCOM*, pp. 31–40, 2001.

[7] N. Sastry, E. Yoneki, and J. Crowcroft, "Buzztraq: predicting geographical access patterns of social cascades using social networks," in *SNS '09: Proceedings of the Second ACM EuroSys Workshop on Social Network Systems*, pp. 39–45, 2009.

[8] "Yahoo GeoCode." `http://developer.yahoo.com/maps/rest/V1/geocode.html`.