Buzztraq: Predicting geographical access patterns of social cascades using social networks

Nishanth Sastry
University of Cambridge

Eiko Yoneki
University of Cambridge

Jon Crowcroft
University of Cambridge

ABSTRACT

Web 2.0 sites have made networked sharing of user generated content increasingly popular. Serving rich-media content with strict delivery constraints requires a distribution infrastructure. Traditional caching and distribution algorithms are optimised for globally popular content and will not be efficient for user generated content that often show a heavy-tailed popularity distribution. New algorithms are needed.

This paper shows that information encoded in social network structure can be used to predict access patterns which may be partly driven by viral information dissemination, termed as a social cascade. Specifically, knowledge about the number and location of *friends* of previous users is used to generate hints that enable placing replicas closer to future accesses.

1. INTRODUCTION

Requests for web content are known to follow a heavy-tailed distribution. For example, Yu et al. [11] find that the top 10% of the videos in a video-on-demand system account for approximately 60% of accesses, and the rest of the videos (the 90% in the tail) account for 40%.

This popularity pattern can make server provisioning difficult, especially for rich-media content such as streaming video, which have relatively strict delivery constraints. The problem becomes especially severe with the recent proliferation of rich-media User Generated Content (UGC) such as YouTube videos, whose popularity can vary dynamically, and often dramatically [3].

While global replication via content delivery networks (CDNs) is efficient for the most popular content, the majority of objects are in the tail and accessed too rarely for global replication to be practical. For instance, YouTube only uses CDNs for the most popular videos [5]. However, objects in the tail collectively account for a sizeable fraction of accesses.

The goal of this paper is to help mitigate the difficulty of serving content not yet popular enough to be globally replicated. Given a history of previous accesses, we wish to predict the geographies of the next few accesses, so that replicas may be intelligently provisioned to minimise future access times. Our predictions are based on the means by which a UGC object becomes known to potential users.

Knowledge of a UGC object can spread in two ways; broadcast highlights or viral propagation. The first happens when the UGC object is featured or highlighted on a central index page. Examples include being featured on the home page of the hosting sites (such as the featured videos list on YouTube); being promoted on an external social bookmarking site (e.g. if slashdotted, or featured on Digg, Reddit, Del.icio.us "hotlists", etc.); or ranking high on a google search. UGC objects in this class have to be popular according to the indexing algorithm used. Such high-visibility objects will likely be accessed many times and from all over the world, and are best served by replicating globally via CDNs.

The second possible means of propagation is by word-of-mouth, by sharing explicitly with a group of friends. This can happen through online social networks, emails, or out-of-band (or face-to-face) conversations. This kind of viral propagation has been termed as a *social cascade* and is considered to be an important reason for UGC information dissemination [4].

The links between friends on an online social network explicitly captures the *means* of propagation for social cascades. Furthermore, many social networking sites include approximate geography information. Thus, information about the friends of previous users and their geographical affiliations could be used to predict the geographical access patterns of future users.

In reality, content access is driven by a diverse mixture of random accesses and social cascade-based accesses. The content provider must place replicas to handle this access pattern. A user request from a region where the provider has a replica of the content object is counted as a local access. A user request from a region where there is no local replica is counted as a remote access. Remote access is costlier than local access. The goal of the provider is to minimise the cost of access by choosing the geographic regions in which to place a fixed number of replicas of the content.

Two replica placement strategies are considered. The first, *location based placement*, uses the geographical location of recent users[1] to place replicas. The second strategy, which we call *social cascade prediction*, places replicas in regions where the social cascade "epidemic" is densest, as determined by the average number of friends of previous users.

In the specific case where a fixed number, k, of replicas is chosen, location based placement amounts to placing the replicas in the top k regions ranked by number of recent users. Social cascade prediction ranks regions by the number of friends of previous users and places replicas in the top k regions.

Our main result is that social cascade prediction can decrease the cost of user access; i.e., more users are served by local replicas. The cost decrease is greatest when the cascade is responsible for most requests. Costs also decrease when cascades are responsible for fewer requests than random accesses.

Based on this, we have built a prototype system, Buzztraq, that provides hints for replica placement by using social cascade prediction. Intuitively, Buzztraq relies on the presence of a social cascade component, which makes the geographies of user requests non-random.

Location based placement predicts that future requests will come from the same geographies as those of past requests. If instead, the requests shift to a new region, it is slower to react – until enough requests come from the new region to displace one of the old top-k, replicas are not moved.

In contrast, Buzztraq's social cascade prediction strategy starts counting friends of previous users who are in the new region even before a request originates from the region. Furthermore, the number of local friends of users grows faster than the actual number of users from the new region. Thus, Buzztraq's strategy is faster to shift replicas and incurs fewer remote accesses.

The paper proceeds as follows. Section 2 studies the geographic spread of social cascade. Section 3 describes how we obtain the inputs used by Buzztraq. Section 4 discusses the mechanics of Buzztraq and strategies for replica placement. Section 5 evaluates these strategies. Section 6 discusses related work. Section 7 discusses some limitations. Section 8 discusses our next research steps and concludes.

2. SOCIAL CASCADE AS AN EPIDEMIC

When users access a UGC object influenced by their friends, it can be modeled as if infected by such friend's opinion. We envision that many ideas, messages, and products could be spread rapidly through our population as social epidemics.

A recent example is the use of the hashtag "uksnow"[2] on Twitter messages sent across the UK on Feb 2, 2009. Although there was no prior agreement on using this string, it quickly spread amongst Twitter users, and became the most popular hashtag. At its height, between 3pm and 5pm, nearly 2000 Twitter posts used the tag, making it the most popular hashtag of the day.

This section investigates how information can spread across geographies as an epidemic. We take an empirical approach, using friend lists from an online social network (details in Section 5.1) to emulate a social epidemic. We select a single user as an initial infectious user and propagate the infection process to her friends. This process is repeated over n rounds, with infection spreading from the initial seed to nodes n hops away.

Fig. 1 shows two possible geographic distributions of infected users. Fig. 1(a) depicts a rapidly shifting epidemic. The infected population and the regional spread of the users changes from the third round (left) to the fifth round (right). On the other hand, Fig. 1(b) shows the infection can also proceed without much change in geographic locations.

The history of past locations can trivially predict the future when the epidemic is localised. The rest of the paper discusses how to predict regions of future infection when the epidemic is shifting.

3. INPUTS TO BUZZTRAQ

Buzztraq takes users' declared social links and geographic affiliations and produces hints on where to place replicas. This section discusses how Buzztraq obtains the declared social links and geographic affiliations.

3.1 Social network information

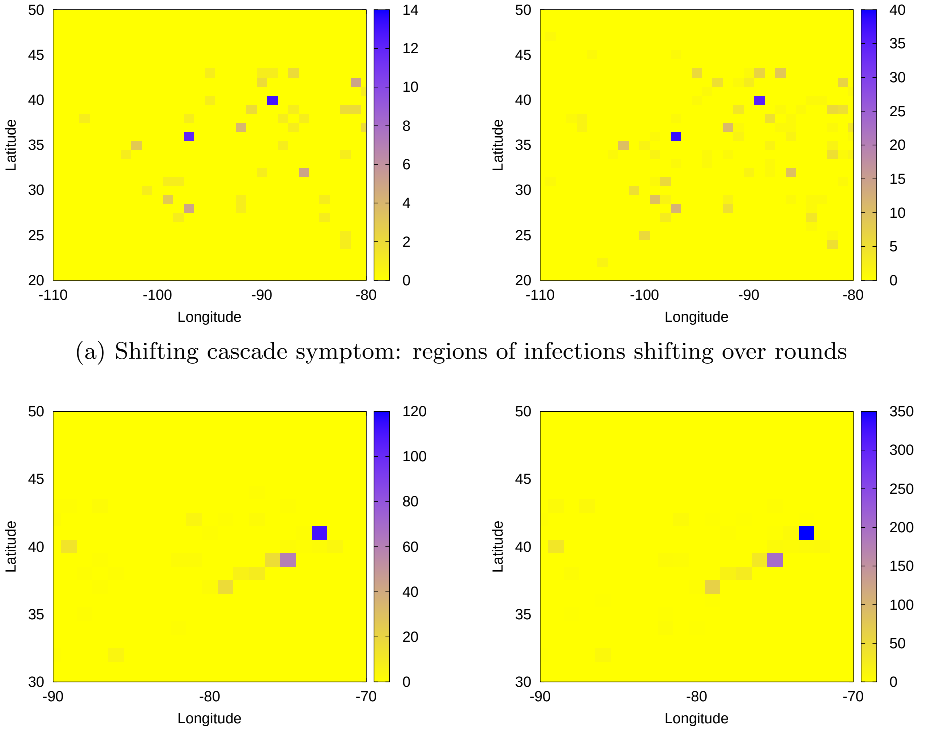
Buzztraq needs the declared social links of users. Previously this information was confined to social networking sites. New APIs such as Facebook Connect[3] and MySpace Data Availability[4] are starting to make this data available to external web sites. These new APIs allow a user to login to external web sites using their identity on the corresponding social network. The external web site is authorised to retrieve and add related information about the user. Buzztraq uses the Facebook Platform API to retrieve each user's friends, and their publicly available affiliation information.

[1] This can be determined from the IP address block of the user. Commercial CDNs may employ similar strategies [9].

[2] http://www.hashtags.org/tag/uksnow

[3] http://developers.facebook.com/connect.php

[4] http://developer.myspace.com/community/myspace/dataAvailability.aspx



(a) Shifting cascade symptom: regions of infections shifting over rounds

(b) Stable growth of cascade: Infection regions stay the same over rounds

Figure 1: Geographical nature of social cascades

3.2 Obtaining geographic information

We attempt to deduce a geographic location for each retrieved affiliation using Google's geocoding API[5]. Section 5.1 shows that the affiliations of the users are largely geographical in nature. The geocoding API translated 71.3% of user affiliation strings into latitude-longitude coordinates.

The final goal is to design a replica placement policy. For that, geographic decisions must be made. Locations are treated as points and are clustered into regions using the k-means algorithm [8]. To decide cluster membership, Vincenty's formula is used to calculate geodesic distances between points [10]. The clusters define fixed regions across the world, and the UGC provider can place replicas in any of the identified regions.

Although social networks typically contain information about users' current geographic locations, this is not the right granularity for our purpose. The geographical spread of a user's influence is not limited to their current location. Often, information about new objects may be forwarded by out-of-band channels such as emails to old friends in the user's previous locations. In practice, we also find that the current location information is not entered by a vast majority of users. Therefore, we use all the declared affiliations of the user in predicting the location of next access.

By giving equal weight to all locations, we are ignoring the complexities involved in word-of-mouth propagation, and may end up introducing false positives. For instance, depending on the nature of the content, a user may be much more likely to spread information about it in only a subset of her affiliations/communities.

One mitigating factor is that Buzztraq hints are restricted to regions of the world. If the geographic affiliations of a user all belong to a single region, then the hints will be correct. Furthermore, to the extent that the user has more friends in the geographic region where she is most likely to spread a social cascade, Buzztraq hints will still be correct.

4. PREDICTING FUTURE ACCESSES

The UGC provider specifies a single UGC object or a collection of related content by a content-id. Buzztraq keeps note of users accessing content identified by each content-id. Using information about these users' friends and affiliations, hints are generated on where to place replicas of the content.

For purposes of exposition, we discuss how this is done in the context of a possible UGC provider architecture. Note that the basic concepts underlying Buzztraq do not rely on this specific model.

[5] `http://code.google.com/apis/maps/documentation/geocoding/index.html`

4.1 A basic UGC provider model

We assume that users first log in at a central site and are redirected to one of a fixed number of replica sites to access the requested rich media UGC. Redirection to a replica site local to the user is considered to be preferable. For instance, this may enable better delay and jitter guarantees.

The central site runs Buzztraq, which obtains the user's declared friends and geographical affiliations as detailed in Section 3. After each user access, Buzztraq uses this information to predict the top-k regions from which future accesses to the requested UGC are likely to originate. The UGC provider can use these hints to decide where to locate each UGC object. Specifically, in our evaluation, we look at the case where each content object is placed in a fixed number of replicas.

Buzztraq predictions are to be treated as *hints* for replica placement. While hints are generated after each access, the provider is not required to reconfigure replica locations each time. This is not critical since the set of top-k regions is not expected to change frequently.

There may be regimes where it is practical to reconfigure replicas after each user. For instance, suppose each region contains all the UGC objects hosted by the provider, but only those most likely to be accessed are kept in main memory. Buzztraq hints can be used to decide which k regions will keep a given object in main memory. Changing this set is less expensive than shipping the content over the network, and could potentially be done after each user access.

4.2 Generating replica placement hints

Without social network links, the best a UGC provider can do is **location based placement**. This strategy keeps per-region histories of user accesses and places replicas in regions which have historically contributed the maximum number of users. Typically, only one region can be found, by reverse-mapping the IP address block of the user to a geography/ISP. The evaluation below uses the social network affiliation information and updates the UGC provider's history for the all the regions the user is affiliated with.

Buzztraq uses an alternate strategy, **social cascade prediction**, which predicts the next accesses by taking social cascade into account. If user accesses are being driven by word-of-mouth propagation, we expect that some of the future accesses will be made by friends of previous users. Thus, our strategy is to place the replicas in the top-k regions where the number of potential future users, as measured by the number of friends of previous users, is highest.

Unlike location based placement, which only counts the number of *previous* users in each region, social cascade prediction additionally attributes non-local friends to their appropriate regions as potential *future* users.

If the cumulative number of friends of previous users ranks a new region in the top-k, Buzztraq predicts that more accesses will originate from this region, owing to social cascade. Location based placement will not rank this new region in the top-k until the new region generates enough requests to displace one of the previous top-k. During this transition period, location based placement will cause non-local replica access for users from the new region, leading to higher costs.

If a user's friends are local to her region, then both social cascade prediction and location based placement will recommend placing replicas in the same regions.

The approach of counting friends of previous users is similar to the concept of the reproductive number R in epidemiology, which measures the average number of secondary cases caused by each case of an infectious disease [2]. If $R > 1$, then the infection will be sustained in the region. In this language, we are counting the number of potential secondary accesses that could be caused by a previous infected user. Buzztraq's output of top-k regions gives the regions where the intensity of infection is highest. Since each access generates new hints, only the current infection intensity is counted. We do not normalise to predict whether the infection will be sustained.

5. EVALUATION

This section evaluates the relative costs of location based placement and social cascade replication using a synthetic workload. Social links and geographical affiliations are derived from a small subset of Facebook users. We generate a workload with user requests coming as a mixture of social cascade and random accesses, and compare the relative costs of the two different strategies for replica placement. The simulations find that social cascade prediction can help place replicas closer, on average, than location based placement.

5.1 User characteristics

Users for our workload are drawn from 20,740 facebook profiles from the Harvard network with profile IDs $< 36,000$. There are 2.1 million links between them, with a mean degree of 63 and a maximum degree 911.

The users have 1,660 distinct affiliations, of which 1,181 could be mapped to geographic locations, all over the globe. Using k-means clustering, we classified these into 10 regions. Our algorithm found separate clusters for North Africa, South and Central Africa, Europe and the Middle East, Australia and the Far East, South America, and the Indian Sub-continent. Predictably, there were multiple (4) regions within the United States.

5.2 Workload

Evaluation is driven by a simple workload. It is not intended to capture the all the complexities of user request arrivals. User accesses from across the globe are assumed to arrive at the central site in some serialisable fashion. Only the sequence of requests matters; there is no notion of real time. We also assume that user accesses are generated either by a social cascade or by a random process. Additionally, each user performs at most one access.

The main goal of the workload is to have a tunable amount of social cascade-based user accesses. User requests are assumed to arrive because of social cascade with probability p_s, or as a result of a random access, with probability $(1 - p_s)$. Thus, with probability p_s, the next user is chosen to be a friend of a previous user; with probability $1 - p_s$, the next user is a random user. We incorporate a notion of recency in the social cascade process – only friends of the last TTL users are chosen for non-random accesses.

Given this workload, the UGC provider has to place replicas so that access cost is minimised. If the provider has a replica in the region of the next user, it is deemed to be a local access; otherwise it is a remote access. The cumulative cost is measured by a cost function which is arbitrarily defined so that a remote access is $c_r = 20$ times costlier than a local access. The provider's goal is to minimise the total cost of all user accesses. Note that any value of $c_r > 1$ will capture the relative difference in the long term costs of two replica placement strategies. Using larger c_r allows us to see the difference after fewer simulated user requests.

The UGC provider is allowed a fixed number of replicas ($k = 3$ in our experiments), and there are 10 regions in the world where the replicas can be placed. The replica placement strategy basically amounts to a strategy of choosing the top regions predicted for future accesses. The UGC provider is allowed to reconfigure its replicas after each user access.

Our dataset contains more declared affiliations for places within USA than any other country. Thus a safe strategy would be to concentrate all replicas in US regions. However, note that USA also contains four regions. By restricting the number of allowed replicas to three, any placement strategy is forced to choose at least one US region to serve remotely. This counteracts any inherent geographical bias and brings out the relative difference in the costs of the two strategies.

5.3 Relative cost of social cascade prediction

In effect, location based placement uses the history of previous accesses to predict future accesses. Social cascade prediction explicitly captures a user's friends as potential future users. Thus, social cascade prediction should be expected to work better if there is a strong social cascade component driving the user accesses.

To verify this, we simulate the same workload (with $p_s = 0.5$) on two UGC objects which are placed using following social cascade prediction and location based strategies, respectively. We measure the cumulative cost of serving the first n requests, as n increases.

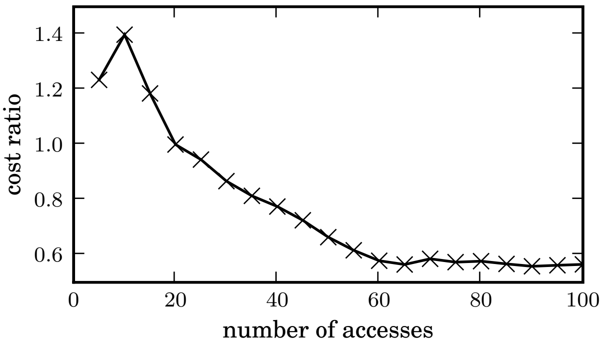


Figure 2: Cost comparison of social cascade prediction to location based placement. $p_s = 0.5$. When cost ratio is less than 1, social cascade prediction is cheaper.

If the UGC provider is able to serve more users local to regions where it has placed replicas, its cost is lower. Fig. 2 plots the result. The x-axis shows n and the y-axis plots the ratio of the cumulative costs of serving the first n requests using the social cascade prediction strategy to the cumulative costs using location based placement. Initially, when there is no discernable social cascade, location based placement outperforms. However, as the number of accesses increases, social cascade prediction becomes the more efficient strategy.

Fig. 3 examines the relative efficiency of the two strategies for different values of p_s, the probability that the next user accesses because of a social cascade. A sequence of 100 requests are performed, and the relative cumulative cost of serving the last ten requests is measured, for different values of p_s. The cost ratio remains less than 1 (i.e. social cascade prediction is cheaper) for all the p_s values we measure. As the probability of a social cascade choice increases, the cost ratio drops, showing that the social cascade prediction does detect the underlying process generating the user inputs.

6. RELATED WORK

Buzztraq is motivated by a recent result [4] confirming anecdotal evidence that social cascades are an important factor in information spreading about user generated content, specifically photos on Flickr.

We emphasise that this system is intended mainly for UGC objects that are not popular globally. For globally popular content, commercial CDNs such as Akamai[6] can be a better fit. On the other hand, Akamai and other CDNs use DNS resolution to direct users to the

[6] http://www.akamai.com

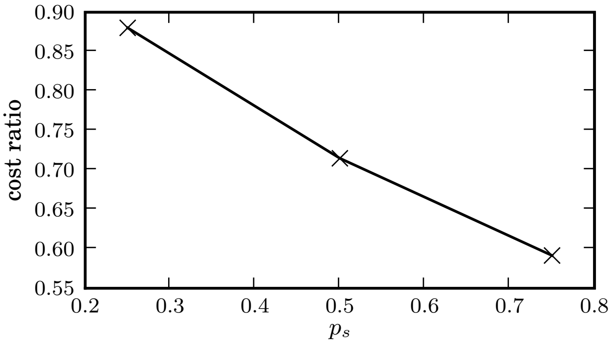


Figure 3: Average cost ratio for different values of p_s. As p_s, the probability that social cascade drives user access, increases, Buzztraq's strategy becomes more efficient

nearest replica [1]. This can be incorporated into our system as a useful complementary behaviour.

The expected utility of Buzztraq hints depends on the *collective* popularity of long-tail content. In one study [11], the 90% of the videos that comprise the tail account for 40% of accesses, whereas another [3] reports that the tail 90% accounts for around 20% of the views, at least in the limited datasets studied. Buzztraq will clearly be more useful in the former case.

The system attaches a geographic profile to users by utilising geographic affiliations on their online social network profile. We believe this is a novel application. Several previous systems, including online advertisement systems have previously tracked users, but most of them use the IP address of the user to glean geography. One illustrative example is Cluster Maps[7], which pinpoints visitors to blogs and draws a world map showing visitor locations.

7. SCOPE AND LIMITATIONS OF SOCIAL CASCADE PREDICTION

We showed how to use social cascade prediction to mitigate the cost of storing and serving the long-tail of UGC objects that are not (yet) popular enough to replicate worldwide, using standard mechanisms such as CDNs. Any content that is disseminated virally can potentially benefit from social cascade prediction; it is not specific to serving user generated content.

Social cascade prediction predicts the geographic location of social cascades by utilising friendship and geographic information in social networks. Lacking accurate and complete geographic affiliation records in current online social networks, we use users' network affiliations and attach geographic locations to them. Success naturally depends on accuracy of geocoding systems - while the current crop of geocoding APIs are very good at parsing, there are limitations. (e.g. “MIT”, “BYU” etc. were parsed to latitude-longitude coordinates, but “SUNY Buffalo Graduate Center” proved to be too complex). Also, we are conflating geographical closeness between server replica and user, with good network connectivity. This may not neccessarily be a correct assumption in all cases.

Buzztraq uses the logical OR of a user's geographical affiliations on Facebook. On the one hand, this is beneficial because it captures information not in the social network about means for social cascade (e.g. a user might spread information to someone not on their Facebook profile but in their geographical affiliation region). On the other hand, this could introduce noise – old and inaccurate affiliations might cause our system to predict the next few accesses from a foreign location when it is not called for.

Even with the above caveats, we believe that our strawman implementation of Buzztraq is the beginning of a system that can efficiently handle the long-tail of UGC that is not yet popular for expensive worldwide delivery by CDNs.

8. CONCLUSION AND FUTURE WORK

We conclude by emphasizing that identifying the geographical locations of potential next users is only half the problem. The other half of the problem is actually provisioning a server or servers such that the service time is minimised. This is a complex problem in itself, and this paper does not address all the details. Instead, we simplify the problem and find the best regions in the globe in which to place a given number of replicas.

Furthermore, this paper only considered placement strategies. In other words, social cascade prediction has been used to answer the question of *where* to place a UGC object. This works well when the UGC object is being replicated on spare storage available at the replicas using spare bandwidth that the UGC provider is already contractually obligated to consume. Our next steps to a more complete system will require resolving the question of *which* objects to replicate when replicas have limited storage and bandwidth, as well as possible strategies for the replicated videos *re*placing other videos at the replica site.

Finally, our early prototype captures social cascades using a very simple model.Considerably more sophisticated models have been proposed [6, 7]. Incorporating these could lead to better geographical access pattern predictions with Buzztraq.

9. ACKNOWLEDGEMENTS

We thank Joe Bonneau and Jonathan Anderson for providing us the data from facebook users, Steve Hand and Derek Murray for valuable suggestions and dis-

[7] http://www.clustrmaps.com

cussions that helped improve and clarify earlier drafts, and our shepherd Tao Stein, for editing help. This research is funded in part by the EU grants for the Haggle project, IST-4-027918, and the SOCIALNETS project, 217141, and a St. John's Benefactor's Scholarship to NS.

10. REFERENCES

[1] Akamai Technologies. Fast internet content delivery with freeflow, April 2000.

[2] Anderson, R. M., and May, R. M. Population biology of infectious diseases: Part i. *Nature* (1979), 361—67.

[3] Cha, M., Kwak, H., Rodriguez, P., Ahn, Y.-Y., and Moon, S. I tube, you tube, everybody tubes: analyzing the world's largest user generated content video system. In *IMC '07: Proc. of the 7th ACM SIGCOMM conference on Internet measurement* (2007).

[4] Cha, M., Mislove, A., Adams, B., and Gummadi, K. P. Characterizing social cascades in flickr. In *Proceedings of the first ACM Workshop on Online social networks (WOSN)* (2008).

[5] Hoff, T. Youtube architecture, Mar 2008. `http://highscalability.com/youtube-architecture`. Notes on google video talk by Cuong Do.

[6] Kempe, D., Kleinberg, J., and Tardos, E. Maximizing the spread of influence through a social network. In *KDD '03: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining* (New York, NY, USA, 2003), ACM, pp. 137–146.

[7] Leskovec, J., Adamic, L. A., and Huberman, B. A. The dynamics of viral marketing. *ACM Trans. Web 1*, 1 (2007), 5.

[8] MacQueen, J. B. Some methods for classification and analysis of multivariate observations. In *Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability* (1967).

[9] Mahajan, R. How akamai works. `http://research.microsoft.com/en-us/um/people/ratul/akamai.html`, 2001.

[10] Vincenty, T. Direct and inverse solutions of geodesics on the ellipsoid with application of nested equations. *Survey Review XXIII*, 176 (April 1975).

[11] Yu, H., Zheng, D., Zhao, B. Y., and Zheng, W. Understanding user behavior in large-scale video-on-demand systems. *SIGOPS Oper. Syst. Rev. 40*, 4 (2006), 333–344.