Counting YouTube Videos via Random Prefix Sampling

Jia Zhou, Yanhua Li, Vijay Kumar Adhikari, and Zhi-Li Zhang
Department of Computer Science and Engineering
University of Minnesota
Minneapolis, MN 55414, USA
{jzhou,yanhua,viadhi,zhzhang}@cs.umn.edu

ABSTRACT

Leveraging the characteristics of YouTube video *id* space and exploiting a unique property of YouTube search API, in this paper we develop a *random prefix sampling* method to estimate the *total* number of videos hosted by YouTube. Through theoretical modeling and analysis, we demonstrate that the estimator based on this method is *unbiased*, and provide bounds on its variance and confidence interval. These bounds enable us to judiciously select sample sizes to control estimation errors. We evaluate our sampling method and validate the sampling results using two distinct collections of YouTube video *id*'s (namely, treating each collection as if it were the "true" collection of YouTube videos). We then apply our sampling method to the *live* YouTube system, and estimate that there are a total of roughly 500 millions YouTube videos by May, 2011. Finally, using an *unbiased* collection of YouTube videos sampled by our method, we show that YouTube video view count statistics collected by prior methods (e.g., through crawling of related video links) are highly skewed, significantly under-estimating the number of videos with very small view counts (< 1000) ; we also shed lights on the bounds for the total storage YouTube must have and the network capacity needed to delivery YouTube videos.

Categories and Subject Descriptors

H.3.5 [**Information Storage and Retrieval**]: Online Information Services

General Terms

Measurement

Keywords

Online social networks, Sampling, YouTube

1. INTRODUCTION

As the world's largest video sharing website, YouTube hosts a large number of mostly user-generated videos that are viewed by millions of users each day. For example, based on its own counting [26], YouTube states that it serves a total of more than 2 billion views a day. According to a recent study [15], YouTube traffic contributes to a significant portion of inter-domain network traffic; some estimate [1] places it at 10% of the total Internet traffic. Estimating the total number of videos hosted by YouTube and other statistics associated with them, e.g., number of view counts per day or the number of users uploading videos, is of great interest and import from both technical and social perspectives. For instance, knowing the total number of videos and view counts per day can shed light on the total amount of storage as well as the network capacity needed to store and deliver YouTube videos.

Unfortunately, these statistics regarding YouTube videos are *not* made available publicly by YouTube. Obtaining such statistics through other means (e.g., sampling) is not an easy and straightforward task for a variety of reasons. For example, while each YouTube video is identified by a unique *11-character* identifier (thereafter referred to as YouTube video *id*) that is randomly generated, the video id space is extremely large, of the order $O(64^{11})$ (see Section 3 for details). Hence any *brute-force* survey of the entire YouTube video population will be too costly; nor will any direct application of *(uniform) random sampling* to the video id space, e.g., by querying randomly generated video id's *a la* [7, 24], be effective. Existing methods for *collecting* YouTube videos rely on crawling the YouTube website and following the "related videos" links embedded in the web pages via either breadth-first or depth-first search [4, 9, 17]. While these methods provide an effective way to collect a sample of YouTube videos (or video id's), they produce a biased sample. Estimating YouTube statistics (e.g., view counts) using such biased samples can produce very skewed results (see Section 3). More sophisticated (graph-based) sampling methods [11,14,16] to circumvent or correct the bias require that the underly graph be *undirected*, whereas the graph formed by YouTube related videos is *directed* (see Section 2 for further discussion on this point and other related work).

Leveraging the characteristics of YouTube video *id* space and exploiting a unique property of YouTube search API, in this paper we propose and develop a *random prefix sampling* method to estimate the *total* number of videos hosted by YouTube. YouTube provides an API to allow users to perform keyword search to find videos they are interested in. One unique property of YouTube search API that we accidentally stumble on is that when searching using a keyword string of the format "`watch?v=xy...z`" (including the quotes) where "`xy...z`" is a prefix (of length L, $1 \leq L \leq 11$) of a possible YouTube video id which does not contain the literal "-" in the prefix, YouTube will return a list of videos whose id's begin with this prefix followed by "-", if they exist. The search may also return some videos whose id's do not contain the prefix, but the title, description or other fields happen to contain the entire search string (including "watch?v="). When the prefix is short (e.g., 1 or

IMC'11, November 2–4, 2011, Berlin, Germany.

2), it is more likely that the returned search results may contain such "noisy" video ids; with longer prefix length, the probability that this happens becomes zero or extremely small. On the other hand, using a prefix that is too long may result in a no-hit, i.e., no video id's being returned. Hence when performing random prefix sampling, the prefix length needs to be carefully selected to balance this trade-off (see Section 4 for details).

Taking advantage of this unique property of YouTube search API that allows us to perform random prefix sampling, we develop a theoretical model to derive an *unbiased* estimator for estimating the total number of YouTube videos, and provide bounds on its variance and confidence interval. These bounds enable us to judiciously select sample sizes to control estimation errors. The model and theoretical analysis are presented in Section 5. In Section 6, we evaluate our sampling method and validate the sampling results using two distinct collections of YouTube video *id*'s (namely, treating each collection as if it were the "true" population of YouTube videos). We then apply our sampling method to the *live* YouTube system, and estimate that there are a total of roughly 500 millions YouTube videos by May, 2011. Further, using an *unbiased* collection of YouTube videos sampled by our method, in Section 3 we show that YouTube video view count statistics collected by prior methods (i.e., through crawling of related video links) are highly skewed, significantly under-estimating the number of videos with very small view counts (< 1000). Finally, we show the bounds for the total storage YouTube must have and the network capacity needed to delivery YouTube videos, which is important for us to understand the impacts of YouTube to the Internet.

2. RELATED WORK

There are a number of recent studies on estimating the size and other properties of on-line social networks. In [20], Rejaie *et al.* estimate the number of users for MySpace and Twitter, where the key technique used is based on the observation that user id's are generated sequentially in an increasing order. This method is not applicable to YouTube, as video id's are randomly generated from a large id space. The authors of [25] propose a method to estimate the number of nodes in a given connected graph, and applies to a YouTube *related video* (sub)graph obtained using a sample YouTube video dataset from [17]. This method cannot be used to estimate the *total* number of YouTube videos. Graph-based methods such as snowball sampling [23] or random walks [19, 21, 22] have been widely used for collecting a *sample* of a large online social network, and this sample is then used to estimate other properties (e.g., degree distribution) of the social networks. To ensure this sample is unbiased (with respect to statistics of interest) or to correct the bias inherent in the sample, several variations of variations of random walk sampling methods such as Metropolis-Hastings random walk [11] and reweighting random walk [16] have been proposed. These methods cannot directly be used to estimate the *total size* of the underlying network. In a more recent work [14], the authors develop a novel random-walk based method to estimate the total number of users in an online social networks. This as well as the previously cited graph-based sampling methods all assume that the underlying network is *undirected*. Unfortunately, we have tested several YouTube datasets and found that the commonly used YouTube related video network is highly *asymmetrical*: for a given video v, on average more than 50% of its related videos do not list v as their related videos. The study in [8] propose several sampling methods via a search engine API to generate a "near-uniform" sample of documents (under certain plausible assumptions about the search engine). These methods, however, do not provide an estimate of the total size of the underlying document space.

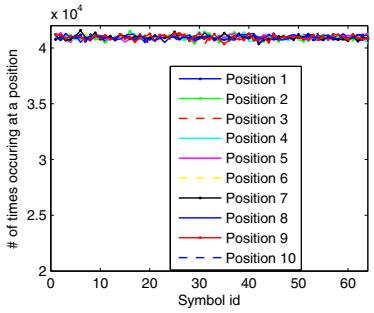

Figure 1: Frequencies at a given position of a video ID

3. YOUTUBE VIDEO ID SPACE

In this section, we present some characteristics of YouTube ids, which motivates us to propose the random prefix sampling method, that can retrieve uniform YouTube video samples.

Each YouTube id consists of 11 characters denoted by $v = [v_1, \cdots, v_{11}]$. The first 10 characters of a valid id contain any of the characters in $S = \{0-9, \_, -, A-Z, a-z\}$, i.e., $v_i \in S$ $(i = 1, \cdots, 10)$. The last (11-th) character v_{11} only comes from $T = \{0, 4, 8, A, E, I, M, Q, U, Y, c, g, k, o, s, w\}$, namely, $v_{11} \in T$[1]. The size of video id space is, therefore, $10^{64} \times 16$. We also observe that YouTube video ids are not generated in any sequence. Instead, YouTube picks an unused random id from this pool for each new video that is uploaded.

To show that YouTube video is randomly generated from the id space, we show that each valid character shows up at i-th position of an id with the same probability. We use a set of 2 million video ids collected via breadth first search using related video links. The result is shown in Fig. 1. In this figure, X-axis represents different positions of an id, and Y-axis shows the number of times a specific character shows up at that position. For any given position, we can see that all the characters are chosen with nearly equal probability. Moreover, if we fix the character(s) in one or multiple positions and count the number of appearances of characters in other positions, we can also see that all the characters are chosen with nearly equal probability for the rest positions. So the chosen of characters in different positions are also independent.

4. RANDOM PREFIX SAMPLING VIA YOUTUBE SEARCH API

In this section, we discuss how we use the YouTube API to perform a random prefix search on the YouTube video id space.

4.1 Random Prefix Search

One unique property of YouTube search API we find is that when searching using a keyword string of the format "`watch?v=xy...z`" (including the quotes) where "`xy...`" is a prefix (of length L, $1 \leq L \leq 11$) of a possible YouTube video id which does not contain the literal "-" in the prefix, YouTube will return a list of videos whose id's begin with this prefix followed by "-", if they exist. The search may also return some videos whose id's do not contain the prefix, but the title, description or other fields happen to contain the entire search string (including "watch?v="). When the prefix is short (e.g., 1 or 2), it is more likely that the returned search results may contain such "noisy" video ids; also, the short prefix may match a large number of videos and YouTube API can only return

[1]Video ids with other characters in 11th position represent copies of other videos (e.g. xG0wi1m-89p is just a copy of xG0wi1m-89o)

some of them in the result, as YouTube limits the number of returned results for any query. In contrast, if the prefix is too long (e.g., 6 or 7), no result may be returned by the search engine.

Note that YouTube search is not case sensitive, so both 'abcd' and 'AbcD' will return the same set of results. If we include a "-" in the prefix or try to search use some other format, the results returned by YouTube may contain many unrelated videos (not beginning with the prefix expected) and hard to explain. We find that querying prefixes with a prefix length of four (with all returned ids having a "-" in the fifth place) provide a big enough result set so that each prefix returns some results and small enough to never reach the result limit set by the API.

4.2 Completeness of the Returned Results

In this section, we show that our prefix based search is nearly complete. We use prefix length four for this validation. Since, by design, our queries only return videos where the fifth character is a "-", we need to show that any video id that has its first "-" in the fifth position can be found by querying YouTube API using our method and using the first 4 characters as the query string. In fact, a "-" is generally used as a separator in URLs for Google search [2] [3].

We use three real datasets to validate these results. The first one is generated using breadth first search (BFS) method based on related YouTube videos [6]. The second one provides all the videos in Entertainment category by December 21, 2006 [9]. The third one with video ids is obtained by searching different keywords and their combinations from a dictionary. In each of these cases, we see whether any video id with "-" in its fifth character observed in these dataset can be also obtained by our method. Table 1 summarizes our findings. We see that in the worst case, there are not more than 0.3% of the ids that are seen in other dataset but cannot be found using our method. More than 99.6% of the video ids with "-" in fifth position can be obtained using our method. After we carefully checked each of those missing videos, we found that those videos are missing all due to the following reasons: a) it is a very new video; b) the video is blocked due to copyright, violence, and sexual issues; c) it is already deleted or configured to be a private video by the uploader. All in all, the prefix search is indeed able to retrieve a complete video id set.

Note that, searching the same prefix in different geographical locations, in general, may return results in different order based upon local popularity. However, since we select our query string carefully to have each query match only a small set of videos, the ordering does not matter much. Moreover, we find that we see the same set of results (albeit sometimes in a different order) when we queried the YouTube API using our method from a large number of Planetlab [18] nodes.

5. RANDOM PREFIX SAMPLING: THEORETICAL ANALYSIS

An estimator is a function of a set of samples which produces an estimate of an unknown characteristic. In this section, we introduce our estimator of the total number of YouTube videos. In addition, we also analyze the variance for the proposed estimator, and develop its confidence interval. Note that the proposed methodology can also be applied to other online social systems as long as those systems satisfy: a) a new generated ID is uniformly selected from ID space; b) entries in those systems can be enumerated by ID prefix searching.

5.1 Estimator of the total number of videos

Using the unbiased random prefix sampling method, now we propose an estimator $\hat{N}$ for the total number of YouTube videos.

Table 1: Prefix search via YouTube API can return a complete set of YouTube videos.

.	Size	$N1$	$N2$	$N3$
Dataset #1 [6]	932763	13864	41	0.30%
Dataset #2 [9]	1687506	24576	66	0.27%
Dataset #3	6692429	99887	122	0.12%
$N1$	Number of IDs with "-" at the 5-th position			
$N2$	Number of IDs with "-" at the 5-th position, but are not in search result			
$N3$	Percentage of IDs with "-" at the 5-th position, but are not in search result			

The observations presented in previous section indicate that 1) the entire YouTube id space can be represented as $\mathcal{S} = S^{10} \times T$, where $S = \{0-9, \_, -, A-Z, a-z\}$ and $T = \{0, 4, 8, A, E, I, M, Q, U, Y, c, g, k, o, s, w\}$; 2) when a YouTube video is uploaded, the probability that a randomly generated id matches a given L-length prefix is a constant, $1 \leq L \leq 11$. Let p_L denote this probability and we have

$$p_L = \begin{cases} \frac{1}{|S|^L} & \text{if } L = 1, \cdots, 10 \\ \frac{1}{|S|^{10}|T|} & \text{if } L = 11 \end{cases} \quad (1)$$

(2)

To estimate N, we randomly generate m prefixes with length L, $1 \leq L \leq 11$, and query them using YouTube API. Each query returns a sample value X_i^L, $1 \leq i \leq m$, representing the total number of YouTube videos with that particular prefix. Then, the total number of YouTube videos can be estimated using these m samples, as stated in Theorem 1.

THEOREM 1 (Estimator of the Total Number of Videos). *Given m samples X_i^L, ($1 \leq i \leq m$) by querying randomly generated prefixes of the same length $1 \leq L \leq 11$, we have the* unbiased *estimator*

$$\hat{N} = \frac{\bar{X}^L}{p_L} = \frac{1}{mp_L} \sum_{i=1}^{m} X_i^L. \quad (3)$$

for the total number of YouTube videos, where p_L is defined as above.

PROOF. We consider the process of how all N YouTube video ids are generated.

Based on our observations discussed in previous section, each YouTube video is generated uniformly and independently from the id space $\mathcal{S}$. Given a prefix of length L, let I_k^L ($1 \leq k \leq N$) be an indicator variable for the k-th YouTube video id, where $I_k^L = 1$ if the k-th ID belongs to that prefix, and $I_k^L = 0$ otherwise. Clearly, $I_k^L (k = 1, \cdots, N)$ are all independent and they all follow the same Bernoulli distribution with successful probability as p_L. Then, the random variable

$$X^L = \sum_{k=1}^{N} I_k^L \quad (4)$$

captures the number of videos with a prefix of length L, and satisfies binomial distribution $Binomial(N, p_L)$ (since it is a sum of N random variables with the same Bernoulli distribution).

The random variable X^L can be sampled by querying randomly generated prefixes with length $1 \leq L \leq 11$. Each outcome from a query is the total number of videos with a particular prefix of length L. If we take m ($1 \leq m \ll 1/p_L$) queries, we have m samples as

$$X_i^L \sim Binomial(N, p_L), \qquad i = 1, \cdots, m. \quad (5)$$

Then each of them has expectation value $E[X_i^L] = Np_L$.

Define the variable $\bar{X}^L = \frac{1}{m}\sum_{i=1}^{m} X_i^L$, which indicates the sample mean. The expectation of $\bar{X}^L$ satisfies

$$E[\bar{X}^L] = E[\frac{1}{m}\sum_{i=1}^{m} X_i^L] = \frac{1}{m}\sum_{i=1}^{m} E[X_i^L] = Np_L$$

Hence the estimator $\hat{N}$ can be derived as follows

$$\hat{N} = \frac{\bar{X}^L}{p_L} = \frac{1}{mp_L}\sum_{i=1}^{m} X_i^L. \quad (6)$$

Then from eq. (6), the expectation of $\hat{N}$ satisfies

$$E[\hat{N}] = E[\frac{\overline{X^L}}{p_L}] = N, \quad (7)$$

which proves that $\hat{N}$ defined in eq. (3) is unbiased. □

5.2 Variance Analysis and Confidence Interval

In this part, we provide some analytical results for our estimator, i.e., its variance and the confidence interval. According to eq. (3), the variance of $\hat{N}$ is

$$\begin{aligned} Var[\hat{N}] &= Var[\frac{\overline{X^L}}{p_L}] = \frac{1}{p_L^2 m^2} Nmp_L(1-mp_L) \\ &= N(\frac{1}{mp_L} - 1), \end{aligned} \quad (8)$$

which indicates that two key factors, i.e., the prefix length L and the number of samples m determine the variance (or the accuracy) of the estimator $\hat{N}$. When m increases, the variance decreases linearly, and when L increases, the variance increases exponentially. In the next subsection, we will discuss how to choose the parameters m and L to minimize the variance.

Now we switch to find the confidence interval for our estimator, where the following Theorem 2 states the result.

THEOREM 2 (Confidence Interval of the Estimator $\hat{N}$). *Given any $0 < \epsilon \ll 1$ and $0 < \alpha \leq 1$, the confidence interval for our estimator $\hat{N}$ as below*

$$Pr[N(1-\epsilon) \leq \hat{N} \leq N(1+\epsilon)] = 1 - \alpha,$$

can be guaranteed when

$$m \geq \frac{z_{\alpha/2}^2}{p_L(\epsilon^2 N + z_{\alpha/2}^2)},$$

where $z_{\alpha/2}$ is the $100(1-z_{\alpha/2})$-th percentile of the standard normal distribution $\mathcal{N}(0,1)$.

PROOF. Since each random variable X_i^L follows the same binomial distribution, their sum $\sum_{i=1}^{m} X_i^L$ also follows a binomial distribution $Binomial(N, mp_L)$. When both Nmp_L and $N(1-mp_L)$ are larger than 10, $\sum_{i=1}^{m} X_i^L$ can be well approximated by a normal distribution [13]. Then, we have the random variable

$$Y_m^L = \frac{\frac{1}{m}\sum_{i=1}^{m} X_i^L - Np_L}{\sqrt{\frac{Np_L}{m}(1-mp_L)}} \quad (9)$$

approximate to the standard normal distribution $\mathcal{N}(0,1)$. Then for a given confidence level $1-\alpha$, $0 < \alpha < 1$, we have

$$Pr(-z_{\alpha/2} \leq Y_m^L \leq z_{\alpha/2}) = 1 - \alpha, \quad (10)$$

where $z_{\alpha/2}$ is the $100(1-z_{\alpha/2})$-th percentile of the standard normal distribution. Then, from eq. (9) and eq. (10), we have

$$N(1-\epsilon) \leq \hat{N} \leq N(1+\epsilon) \quad (11)$$

$$\epsilon = \frac{z_{\alpha/2}}{p_L}\sqrt{\frac{p_L(1-mp_L)}{mN}} \quad (12)$$

where from eq. (12), we know that eq. (11) holds true when the sample size m satisfies

$$m \geq \frac{z_{\alpha/2}^2}{p_L(\epsilon^2 N + z_{\alpha/2}^2)}.$$

□

5.3 How to Choose the Prefix Length L and Sample Size m

Now we are in a position to show how to choose m and L in practice. We use the relative root mean square error (RRMSE) as a metric to quantify the accuracy of the estimation, which is defined as below

$$RRMSE(\hat{N}) = \sqrt{E[(\frac{\hat{N}-N}{N})^2]}. \quad (13)$$

Table 2: Minimal number of samples for different RRMSE and L. Each entry represents corresponding minimum m value for a particular tuple (RRMSE, L)

RRMSE \ L	1	2	3	4	5	6	7
0.05	1	1	1	7	430	27488	1759218
0.10	1	1	1	2	108	6872	439805
0.15	1	1	1	1	48	3055	195469
0.20	1	1	1	1	27	1718	109952

In Table 2, we can see that as L increases, RRMSE increases exponentially. Hence, we will choose L as small as possible to minimize the variance. However, in practice, a prefix of length $L < 5$ contains usually more than one hundred results, and YouTube API can only return at most 30 ids for each prefix query. On the other hand, based on our experimental results, a prefix with length $L = 5$ always contains less than 10 valid ids. Therefore, a prefix length of 5 is a good choice in practice. With this prefix length, from Table 2 we can see that to achieve an RRMSE of 0.05, 0.10, 0.15 and 0.20, we need to have at least $m = 430, 108, 48, 27$ samples, respectively.

6. EXPERIMENTAL RESULTS

In this section, we examine how correctly and efficiently our method works with actual data. Since we do not know the actual video counts for YouTube, we use a "synthetic data" approach. We take a subset of YouTube video ids as "ground truth" and try to see how correctly our method can estimate it. In particular, we study the error rates, and confidence intervals as they change when we change the sample size etc. To justify that the estimated results are unbiased and accurate, we perform cross validations in "synthetic dataset" and in real YouTube. We next apply our method to estimate the total number of actual YouTube videos. Finally, we also present results that show how we can construct a more realistic picture of the view-count distribution of YouTube videos using our unbiased samples.

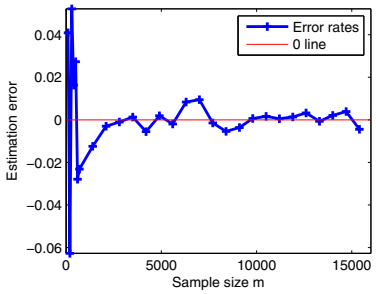

Figure 2: Error rate over Ground Truth, when $L = 3$

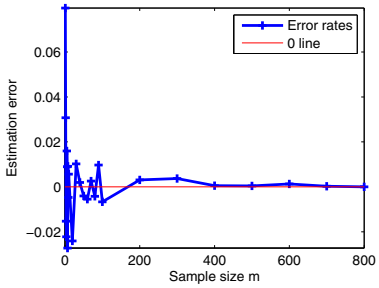

Figure 3: Error rate over Ground Truth, when $L = 4$

6.1 Validating the Theory

We first evaluate our sampling method and validate the theoretical results using two distinct collection of YouTube video id's (Datasets I and II in Table 1), treating each collection as if it were the "true" collection of YouTube videos. Using Dataset II and with $L = 3$ and $L = 4$ respectively, Fig. 2 and Fig. 3 show the estimation errors as a function of the sample size m. We see that in both cases, the estimation error converges quickly to 0, when the sampling size m increases. Further, for smaller prefix length, e.g., $L = 3$, more samples are needed to reach the same level of error rate as that of $L = 4$. These results confirm our theoretical analysis. Fix the prefix length as $L = 3$, Fig. 4 shows the estimation accuracy as a function of the sample size with different confidence interval levels, $\alpha = 5\%, 25\%, 75\%, 95\%$. We see that as we increases the sample size, the confidence interval narrows as expected.

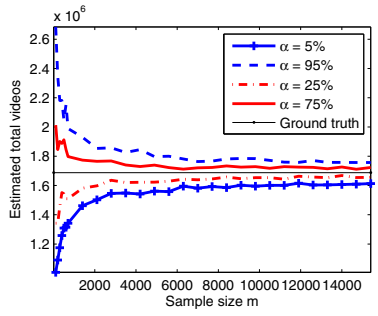

Figure 4: Confidence interval

6.2 Estimating the Total Number of YouTube Videos

We now apply our method to the *live* YouTube website and estimate the total number of YouTube videos. We set $L = 5$ and randomly generate prefixes of length 5 to perform random prefix sampling using the YouTube search API. In Fig. 5, for each $x = 0, 1, \ldots$, we plot the number of (randomly generated) prefixes (the y-axis) that the YouTube search yields exactly x video id's matching the prefix – this plot provides an approximation to the distribution of X^L. Clearly, the curve is bell-shaped, resembling a Gaussian distribution, as predicted by our theoretical model: for large N, the binomial distribution of X^L (see eq.(5) can be approximated by a Gaussian distribution with the same mean and variance.

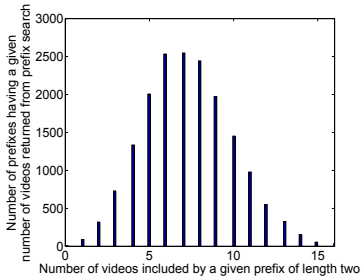

Figure 5: The distribution of the prefixes of length $L = 5$, over number of returned videos

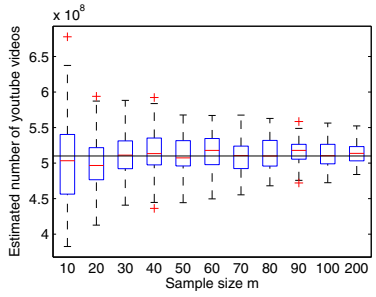

Figure 6: Estimated number of YouTube videos by 05/12/2011

For a fixed sample size m, we can estimate the total number of YouTube videos using the estimator in eq.(7), and bound the variance of the estimation using eq.(8). Fig. 6 plots the estimation results as we vary the sampling size m. We see that as the sampling size m increases, the estimated total number of YouTube videos converges to 5.02×10^8 (502 millions).

Since we do not know the *ground truth* about the total number of YouTube videos, we perform the following simple cross-validation using the sample YouTube video datasets listed in Table 1. We take $N = 5.02 \times 10^8$ as if it were the ground truth, and for a fixed L, generate the *theoretical* distribution of X^L using eq.(5). We then use the sample YouTube datasets to generate the *empirical* distribution of X^L. For $L = 2$, Fig. 7 compares the theoretical and empirical distributions of X^L using Dataset I. The two distributions match surprisingly well, indicating that our estimated total number of YouTube videos is likely within the ballpark of the real ground truth.

6.3 Impact on View Counts

Our random prefix sampling method not only enables us to estimate the total number of YouTube videos; it also produces an *unbiased* sample of YouTube videos. To illustrate the importance of such an *unbiased* sample in estimating other important YouTube video statistics, we take the total *view count* distribution as an example. Fig. 8 plots the total view count distributions obtained from

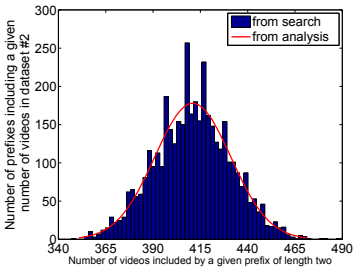

Figure 7: The distribution of the prefixes of length $L = 2$, over number of returned videos

Datasets I and II (two "biased" collections via crawling related videos via breadth-first search) as well as the sample video collection using our random prefix search. In this figure, the x-axis represents the percentage of the videos and the y-axis represents the total view counts. It is clear that the two biased datasets overestimate the total view counts of YouTube videos. For example, our dataset indicates that only 14% of videos have a total view count of more than 1000 (the straight line), whereas 89% and 52% percentage of videos in Datasets I and II have at least 1000 view counts. So Datasets I and II significantly underestimate the number of videos with extremely small total view counts (< 1000). As a result, Datasets I and II significantly inflate the average view counts: the average view count from Dataset I, Dataset II and our dataset are 32046, 9928, and 3898 respectively.

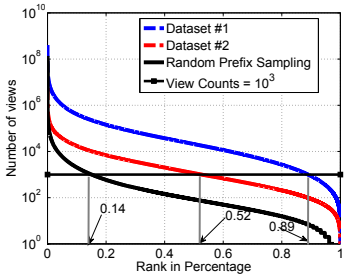

Figure 8: Views counts by different data sets

7. ESTIMATING THE TOTAL STORAGE AND NETWORK CAPACITY REQUIREMENTS OF YOUTUBE

In this section, using an *unbiased* collection of YouTube videos sampled by our method, we provide an estimation of the minimal total amount of storage needed to store YouTube and the total network capacity needed per day to delivery YouTube videos.

To begin with, we first calculate the average file size of YouTube videos. To do this, we analyze the length fields of HTTP responses for the corresponding sampled videos. The average size we obtained is 9.87MB, which is similar to that in [12]. Multiplying this average size with the total number of YouTube videos by May 2011 , we obtain an estimate of the minimal total storage needed to store all YouTube videos by then, which is around 5 Petabytes (PBs) $\approx$ 10MB $\times 0.5 \times 10^9$. Note that for each video, YouTube in general stores multiple (at least 4 [5]) copies and several different formats [10]. So the actual storage capacity needed is likely far bigger; perhaps multiplying a factor of 10 would give us a closer estimate to the real order of magnitude.

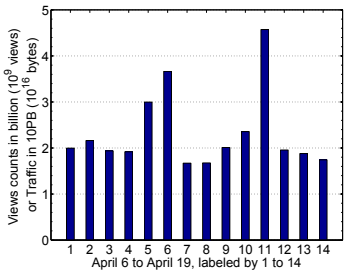

Figure 9: Views counts by different data sets

Further, using an unbiased sample of YouTube videos and examining their view counts over time, we calculate an estimation of the total YouTube video counts per day, and by multiplying it by the average video file size (10 MBs), we obtain an estimated amount of total network capacity needed to carry YouTube videos over the Internet each day. For a two-week period from April 5th to April 18th, the results are shown in Fig.9. From Fig.9, we see that the total view counts range from around 1.7 billion/day to 4.6 billion/day, and the resulting traffic (network capacity) ranges from 17 PBs/day to 46 PBs/day. Note that users may not download and watch the entire video during each viewing and YouTube has different video formats (e.g., formats used by mobile players on smart phones); on the other hand, there are also "wasted" network capacity in YouTube delivery [10]. The above simple "rule-of-thumb" estimation does not capture these effects. Nonetheless, we believe that our results provide a "ballpark estimate" of the actual amount of total YouTube traffic per day.

8. CONCLUSION

In this paper, we introduce a random prefix sampling method via YouTube API, that can uniformly collect YouTube ids, which enables us to design an unbiased estimator of the total number of YouTube videos, as well as in depth analysis on its variance and confidence interval in terms of the sample size and prefix length. Extensive experimental results demonstrate that our sampling and estimation methods provide unbiased estimation of total number of YouTube videos, and total view counts, which discloses a high inherent bias in the results obtained by existing biased sampling methods. We also shed lights on the bounds for the total storage YouTube must have and the network capacity needed to delivery YouTube videos. The proposed random prefix sampling provides a way to unbiasedly study characteristics of YouTube Videos.

As part of our future work, we are interested in using this method to further study how the statistics, such as the total number of YouTube videos, view counts evolve over time, which would give us a dynamic view of traffic by YouTube.

9. ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers and our shepherd, Ratul Mahajan, for their helpful feedbacks on this paper.

We gratefully acknowledge the support of our sponsors. The work is supported in part by the NSF grants CNS-0905037, CNS-1017092 and CNS-1017647.

10. REFERENCES

[1] Ellacoya data shows web traffic overtakes peer-to-peers as largest percentage of bandwidth on the network. http://www.ellacoya.com/news/pdf/2007/NXTcommEllacoyaMediaAlert.pdf.

[2] Google test : hyphen and underscore. http://www.prweaver.com/blog/2004/08/26/2-hyphen-and-underscore/.

[3] Word separators used by google. http://www.internetofficer.com/seo/google-word-separator/.

[4] V. K. Adhikari, S. Jain, Y. Chen, and Z.-L. Zhang. Reverse Engineering the YouTube Video Delivery Cloud. In *HotMD'11*.

[5] V. K. Adhikari, S. Jain, Y. Chen, and Z.-L. Zhang. Vivisecting youtube: An active measurement study. Technical report. http://www.cs.umn.edu/research/technical_reports.php?page=report&report_id=11-012.

[6] V. K. Adhikari, S. Jain, and Z.-L. Zhang. Youtube traffic dynamics and its interplay with a tier-1 isp: an isp perspective. In *Proceedings of the 10th annual conference on Internet measurement*, IMC '10, pages 431–443, New York, NY, USA, 2010. ACM.

[7] Y. Ahn, S. Han, H. Kwak, S. Moon, and H. Jeong. Analysis of topological characteristics of huge online social networking services. In *Proceedings of the 16th International Conference on World Wide Web*, Alberta, Canada, 2007.

[8] Z. Bar-yossef and M. Gurevich. Random sampling from a search engine's index. In *WWW '06: World Wide Web Conference*, pages 367–376.

[9] 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 *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, IMC '07, pages 1–14, New York, NY, USA, 2007. ACM.

[10] A. Finamore, M. Mellia, M. Munafo, R. Torres, and S. R. Rao. YouTube Everywhere: Impact of Device and Infrastructure Synergies on User Experience. In *IMC '11*.

[11] W. R. Gilks, S. Richardson, and D. J. Spiegelhalter. Markov chain monte carlo in practice. In *Operations Research*, 1996.

[12] P. Gill, M. Arlitt, Z. Li, and A. Mahanti. Youtube traffic characterization: a view from the edge. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, IMC '07, pages 15–28, New York, NY, USA, 2007. ACM.

[13] I. Hacking. *An Introduction to Probability and Inductive Logic*. Cambridge University Press, 2001.

[14] L. Katzir, E. Liberty, and O. Somekh. Estimating sizes of social networks via biased sampling. In *Proceedings of the 20th international conference on World wide web*, WWW '11, pages 597–606, New York, NY, USA, 2011. ACM.

[15] C. Labovitz et al. Internet inter-domain traffic. In *SIGCOMM'10*.

[16] L. Lovász. Random walks on graphs: A survey. *Combinatorics Paul Erdos is Eighty*, 2(1):1–46, 1993.

[17] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee. Measurement and analysis of online social networks. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, IMC '07, pages 29–42, New York, NY, USA, 2007. ACM.

[18] PlanetLab. An open platform for developing, deploying, and accessing planetary-scale services. http://www.planet-lab.org.

[19] A. Rasti, M. Torkjazi, R. Rejaie, N. Duffield, W. Willinger, and D. Stutzbach. Respondent-driven sampling for characterizing unstructured overlays. In *Proceedings of IEEE INFOCOM 2009 Mini-Conference*, April 2009.

[20] R. Rejaie, M. Torkjazi, M. Valafar, and W. Willinger. Sizing up online social networks. *Network, IEEE*, 24(5):32 –37, 2010.

[21] B. Ribeiro and D. Towsley. Estimating and sampling graphs with multidimensional random walks. In *Proceedings of the 10th annual conference on Internet measurement*, IMC '10, pages 390–403, New York, NY, USA, 2010. ACM.

[22] D. Stutzbach, R. Rejaie, N. Duffield, S. Sen, and W. Willinger. On unbiased sampling for unstructured peer-to-peer networks. In *Proceedings of the 6th ACM SIGCOMM conference on Internet measurement*, IMC '06, pages 27–40, New York, NY, USA, 2006. ACM.

[23] S. Wasserman and K. Faust. *Social Network Analysis. Methods and Applications.* 1994.

[24] C. Wilson, B. Boe, A. Sala, K. Puttaswamy, and B. Y. Zhao. User interactions in social networks and their implications. In *EuroSys*, pages 205–218, 2009.

[25] S. Ye and F. Wu. Estimating the size of online social networks. In *Social Computing (SocialCom), 2010 IEEE Second International Conference on*, pages 169 –176, 2010.

[26] YouTube. YouTube statistics. www.youtube.com/t/press_statistics.

Summary Review Documentation for

"Counting YouTube Videos via Random Prefix Sampling"

Authors: J. Zhou, Y. Li, V. Adhikari, Z. Zhang

Reviewer #1

Strengths: Nice paper that exploits some interesting practical observations. Well written with very careful and thorough analysis.

Weaknesses: Estimating total number of videos in Youtube appears to be a relatively narrow problem. The approach itself is quite simple and did not particularly strike me as a fundamentally.

Comments to Authors: I like this paper's methodology, careful experimentation and attention to detail. I was however underwhelmed by the goal of just counting YouTube Videos. To me, that was too narrow a goal.

I guess things like view count distributions are interesting to estimate, but that has so little real estate devoted in this paper. I wish I had seen a bit more treatment of such more detailed analysis where the power of the proposed scheme may be a bit more relevant and important.

The authors argue vehemently in the intro that existing approaches offer skewed and biased results. I am not sure I saw in the paper what the total video count results are using those method compared to the proposed method.

I could not completely follow the view count distribution results. The authors mention they use Datasets I and II obtained by BFS crawling of YouTube as the total collection. Then they compare Random Prefix Sampling with other biased methods that again use BFS. So, this is BFS over BFS crawled data, which makes it a bit hard to digest what the implications are.

Why not plot the actual distribution in Figure 8 instead of showing one point alone ?

Reviewer #2

Strengths: A nice, short paper that provides both a way to randomly sample youtube videos and shows the bias in current datasets.

Weaknesses: The results may be short-lived. Perhaps youtube will change their id assignment mechanism tomorrow.

Inufficient evidence that the id assignment mechanism has been reverse engineered correctly.

Comments to Authors: Your entire paper hinges on you having correctly reverse-engineered the video id assignment mechanism. Yet, it wasn't clear to me that the evidence you present of its correctness in the paper is sufficient. Your evidence is essentially the experiment behind Figure 1, which shows that each symbol is equally likely to occur at each position.

Aren't there other id assignment mechanisms that lead to the same distribution (e.g., some multi-letter codes)? It appears that you need a more detailed experiment that shows that each position is filled independently.

Why do you think '-' is treated as a special character in the search? Is that an attempt to accommodate for accidental hyphenation (e.g., by email readers) or something else is going on?

Reviewer #3

Strengths: A piece of work with a clear purpose and a significant technical contribution that provides new insight into YT.

Weaknesses: The work and proposed methodology is YT-specific and unlikely to apply to other online social media systems. There is also little room for follow-up work.

Comments to Authors: This is a nice short paper. while it is exclusively focused on YT, the work does provide new insight into YT as a system and how it is being used.

While theorems 1 and 2 are rather standard, it's nice to see a careful treatment of the proposed estimator before it is applied in practice.

Some discussion about YT-like systems where the proposed methodology may also applicable (what key ingredients are required?) would be helpful.

Reviewer #4

Strengths: The analysis for any sampling bias is the paper's main strength.

Weaknesses: There is little point to the analysis for sampling bias; it is not clear why one should expect there to be significant bias, it is not clear how (much more difficult) measurements of social network size via crawling are precedent for this relatively straightforward count of objects in a uniform ID space.

Comments to Authors: The main problem in this paper is lack of motivation. Why does it matter how many videos are in youtube? Why is your technique interesting for networking researchers in general? Now that I know there are 500 million youtube videos, what am I going to do with that number? Your introduction states (without support) "estimating the total number of videos hosted by youtube and other statistics associated with them ... is of great interest and import from both technical and social perspectives". I don't see it. Sure, you can guess about how much total storage youtube must have, but not really understand the network capacity needed, since the number of videos is not representative of the popularity of videos. And what's the "import" from a social perspective?

At a lower level, again the problem is motivation; assuming I care about how many youtube videos exist, why do I need all the

analysis of how a hashed ID space uniformly distributes objects so that counting the number of objects in a portion of the uniform space estimates the occupancy of the whole? Why not do something more interesting? Why not answer the questions posed in the introduction of how much total storage youtube must have or how much network capacity is needed? It would seem to be simple arithmetic when youtube is happy to provide view counts and file sizes.

Thus, the beginning of section 2 (the topic sentence of the only paragraph in the section) seems unnecessary. I can understand what challenges may befall a project that wishes to estimate the size of a graph based on the properties of a random walk. I cannot understand what those challenges have to do with estimating the size of a bag based on the properties of regions of ID space.

The work appears very well done, which leads to my relatively high rating of the paper. Yet, this exhaustive analysis of a relatively simple trick (that is likely to be disabled if youtube wants to keep the total number of videos a secret) seems more appropriate for a journal than for IMC.

Reviewer #5

Strengths: An unexpected application of prefix sampling.

Weaknesses: The problem domain is narrow and it is not clear how important to count YouTube videos. I'd like to see more interesting applications besides this one.

Comments to Authors: The paper shows an unexpected application of random sampling technique to count YouTube videos. However, the problem domain is too narrow and not well motivated. Some of the proposed techniques are tailored this context that YouTube happened to return its videos that begin with a given prefixes. Such interfaces are bound to change, and the exact ID assignment are also subject to change, which renders the approach no longer work.

I'd like to see other applications of your approach beyond this problem or give me some interesting analyses that you can perform with the knowledge of video counts.

Response from the Authors

We are grateful to the anonymous reviewers and our shepherd, Ratul Mahajan, for their helpful feedback on this paper. We have carefully revised our paper with respect to these comments and we think that as a result, the quality of the paper has been considerably improved. We highlight the major changes as follows.

1. One common concern among reviewers is that it is a relatively narrow goal if this paper only considers estimating the total number of YouTube videos. One of the major goals of this paper is to understand how much storage and network capacity YouTube needs to store and deliver its videos. Therefore, as a response to this concern, we add section 7 discussing the total storage and the network capacity needed to delivery YouTube videos. Specifically, after an estimation of average file size and the total number of YouTube videos, we compute a lower bound for the storage YouTube must have. Additionally, after a sampling and estimating of view counts per day, we compute the total traffic generated by YouTube video delivery. This new section sheds lights on the direct impacts of YouTube to the Internet traffic, demonstrating an important application of counting YouTube videos.
2. Another common concern is that a change of YouTube will render the methodology no longer work. We admit that the proposed method needs certain interfaces to work. However, the insights gained by our method, such as uncovering the possible biases introduced by previous BFS sampling and the impacts of YouTube videos to the Internet traffic, is still important.
3. A third concern is that the proposed methodology may not work for other online social systems. We add a discussion pointing out that the proposed methodology can also be applied to other online social systems as long as those systems satisfy: a) a new generated id is uniformly select from id space; b) entries in those systems can be enumerated by id prefix searching.
4. Finally, for other specific technical concerns, we point out each id position is filled independently. We validate this by fixing the symbol(s) in one or multiple positions and counting the number of appearances of symbols in other positions. For the symbol "-", we provide references showing that it is generally used as a separator in URLs for Google search and we also carefully validate the YouTube case using Datasets I, II and III in Section 4.