Estimating Active GitHub Users using Random Prefix Sampling

Jannik Haas
WPI Data Science
Worcester Polytechnic University
100 Institute Road, Worcester MA, 01609
jbhaas@wpi.edu

1 INTRODUCTION

With online social media networks (OSNs) continuing to grow, a common problem is figuring out just how many active users a platform has. In this paper we propose an unbiased estimator to estimate the number of active Github users using a sampling method similar to the Random Prefix Sampling [1]. We prove that the estimator is unbiased and perform validation on a small subspace of the Github users.

2 PROPOSAL

When a new Github user account is created it automatically receives an 'id'. These ids are assigned in an increasing fashion, starting from 1 for the first Github user ever. When a Github user deletes the account and is therefore no longer an active user, this id gets deleted from the Github API. Due to both time constraints and Github API [2] query limits, we are unable to crawl through the entire id space and count the number of inactive ids to get the exact number of active Github users. This is why we propose this estimator to quickly and accurately estimate the total number of active users. The query we can use to crawl the space returns the next 30 active user from the given starting id. For example when given the starting id of 1, the query returns the next 30 active users with the lowest id being 2 and the highest id being 47. This indicates that within the 45 ids there are 30 active user ids. Naturally, the density of active users will be much lower in the lower id space than in the upper id space. But this allows us to sample a larger bucket of say 100 or even 1000 and get the number of active users within that bucket.

The challenge in this space is the lack of consistent sampling intervals and a lack of predetermined subspaces or buckets of ids. We propose an estimator that artificially divides the Github user space into a certain number of buckets, from which we can then sample active users and calculate the number of active Github users within that particular bucket. The average across many samples is then calculated and extrapolated across the entire Github user space to estimate the total number of active Github users.

3 METHODOLOGY

We begin by collecting the highest possible id assigned currently by creating a new account and querying the id for the account. As of February 22nd, 2021 the highest user id and therefore the space size S is 79,471,337. The total number of buckets N_L = S/L where L is the size of the bucket (i.e. 30,100, or 1000). Sampling starting ids from the number of total buckets and multiplying them by the bucket size gives us the starting ids from each bucket. We then exhaustively search the bucket to get the total number of active users within the bucket. After taking m samples, we take the average across all the samples to get the average number of active users in each bucket. We can then multiply this by the total number of buckets in the id space to get our estimate of total active Github users.

For the results section we used a bucket size of 90 to allow for quicker sampling and allow us to take more samples within the query rate limit. The size of 90 was chosen since it would guarantee that each bucket would take at most 3 queries to exhaustively search. This gave us very consistent results especially as we increased the number of samples, however in further work we would like to explorer sampling larger buckets with fewer total samples and compare the results.

3.1 Proof of Unbiasedness

To prove that our estimator is unbiased we want to show $E(\widehat{N}) = N$ where $\widehat{N}$is the estimated number of active users and N is the actual number of active users. For our estimator $\widehat{N} = \frac{1}{mN_L}\Sigma_{j=1}^{m}X_j^L$ then $E(\widehat{N}) = E[\frac{1}{mN_L}\Sigma_{j=1}^{m}X_j^L]$ = $\frac{1}{mN_L}\Sigma_{j=1}^{m}X_j^L = \frac{1}{mN_L}\Sigma_{j=1}^{m}NN_L = N$.

4 RESULTS

For our results we first validated our approach on a subspace of user ids which we first exhaustively searched to find the true number of active users within that subspace and then tested our sampling method against this ground truth.

4.1 Validation Results

For the validation of our method we first exhaustively searched the first 140,000 user ids to get the exact number of active users within that subspace of 130,688. We then tested our approach using sample sizes of 10, 20, 30, 40, 50, 100, 200. The results can be seen in Figure 1. These results show the box plot of the results of doing each sample size 10 times.

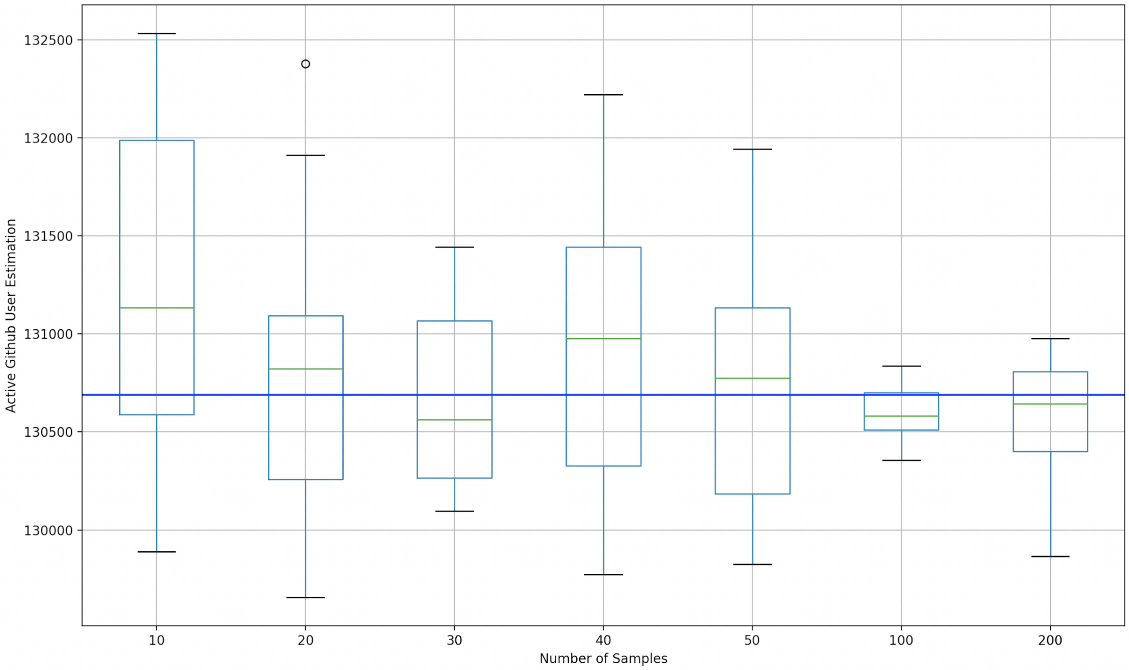

Figure 1: Github Active User Estimation of Validation Subspace

As you can see as m increases, the results approach the ground truth of 130,688. This shows that our estimator is unbiased.

4.2 Total Active Github Users Results

After validating our approach we tested the estimator on the entire Github user id space. Due to time constraints and API query limits we cannot obtain the ground truth of active users, however our validation test showed that our estimator functions as designed and can therefore trust the results from the entire subspace. You can see the results in Figure 2 with the same number of samples as the validation. The results are converging to about 76,840,000 with the actual mean of all the samples of size 200 being 76,840,394. The convergence also shows the unbiasedness of the estimator.

Even a single sample of a bucket size of 90 as you can see in Figure 3 where we performed a single sample 10 times and the results are still consistent with the larger sample sizes.

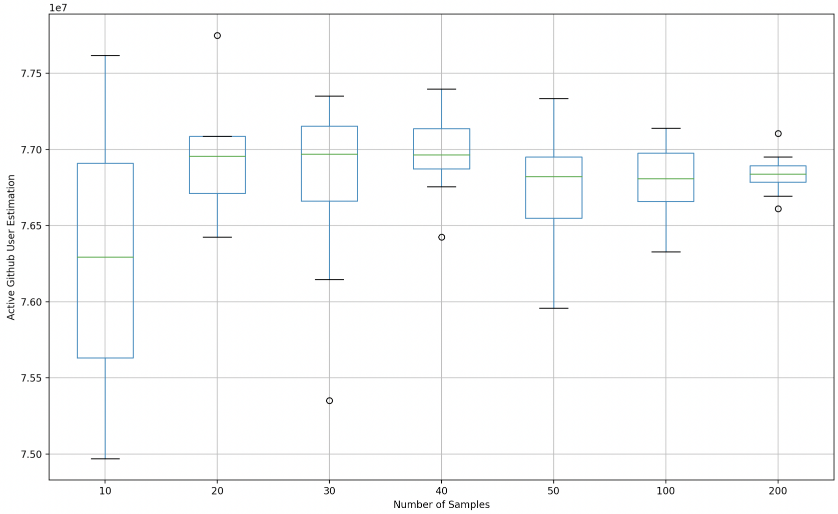

Figure 2: Github Active User Estimation of Entire ID space

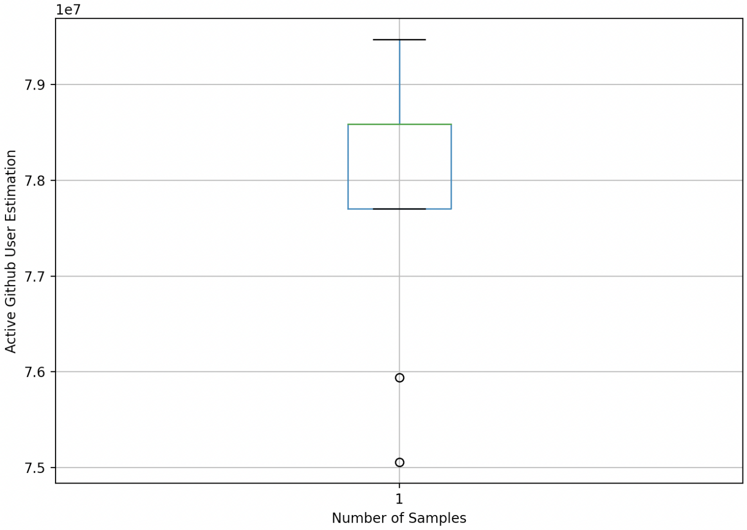

Figure 3: Github Active User Estimation of Entire ID space with only a single sample

5 CONCLUSION

The random sampling estimator performed very well when testing it against the validation subspace and showed similar results when testing it on the entire id space even though we are unable to compare it to the ground truth. A sample of 200 buckets of size 90 will still be much faster than exhaustively searching the entire id space to get the actual number of active users and if a fast answer is needed we can see that even 10 samples are able to estimate the total number of active github users well.

REFERENCES

[1] “GitHub REST API.” *GitHub REST API - GitHub Docs*, docs.github.com/en/rest.

[2] Zhou, Jia, et al. “Counting YouTube Videos via Random Prefix Sampling.” *Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference - IMC '11*, 2011, doi:10.1145/2068816.2068851.