Do Multiple Listeners to the Public Twitter Sample Stream Receive the Same Tweets?

Jiaul H. Paik[1] and Jimmy Lin[2,1]

[1] Institute for Advanced Computer Studies (UMIACS)
[2] The iSchool — College of Information Studies
University of Maryland, College Park

{jiaul,jimmylin}@umd.edu

ABSTRACT

Do multiple listeners to the public Twitter sample stream receive the same tweets? Due to limitations on redistribution of Twitter data, the answer to this question is important for the replicability and reproducibility of research findings. A negative answer creates barriers for different research groups to evaluate algorithms and systems on the same collection of tweets. We describe a pilot experiment in preparation for the TREC 2015 Microblog track that answers this question in the affirmative, which means that an evaluation methodology built on geographically dispersed research groups independently crawling the Twitter streaming API is feasible.

1. INTRODUCTION

Twitter provides a streaming API through which clients can obtain a sample of public tweets—this is useful for researchers who are developing real-time information systems that analyze social media streams, or, alternatively, tweets can be saved to persistent storage for offline analysis. This level of access is available to anyone who signs up for an account. In this paper, we attempt to answer a straightforward question: do multiple listeners to the public Twitter sample stream receive the same tweets? Twitter's API documentation[1] would seem to suggest so, but we wish to answer this question empirically.

The outcome is important for the replicability and reproducibility of research findings. Although anyone can tap into the streaming API to gather tweets, Twitter's terms of service forbid redistribution of the tweets themselves. Thus, if research group A publishes a result on a particular collection of tweets gathered during time period T, they are prohibited from directly distributing the tweets on which those findings are based. Researchers, however, *are* permitted to share the ids of the tweets, from which others can "reconstruct" the dataset by crawling the individual tweets themselves—this was the data distribution mechanism devised for the TREC 2011 Microblog track [4]. While tenable for sharing small collections of tweets, this approach has scalability limitations [2]. However, since anyone can access the streaming API, another research group B might happen to have also gathered tweets during the same time period. In this case, would B be able to replicate the results of A on the same collection of tweets? This is not an unrealistic scenario, as many research groups around the world are continuously gathering tweets for research.

This paper describes a small pilot experiment in preparation for the TREC 2015 Microblog track that answers this question. The TREC task this year revolves around monitoring a stream of social media posts (i.e., Twitter) with respect to a user's interest profile—the goal is for a system to push (i.e., recommend or suggest) interesting content to a user. The evaluation methodology involves participants independently listening to Twitter's streaming API to complete the task. Tweets that are returned by systems will be assessed using a standard pooling methodology. The critical question, thus, is whether such an evaluation design is tenable—will geographically dispersed teams listening to the Twitter stream encounter the same tweets? If no, then there is no way to structure an evaluation in this manner. Fortunately, our pilot experiment suggests that the answer is *yes*, which gives us some confidence that the evaluation methodology for the TREC 2015 Microblog track is feasible.

2. EXPERIMENT DESIGN

As part of the community discussion in formulating the task in the TREC 2015 Microblog track, we asked volunteers to independently gather tweets from the Twitter streaming API during a defined period, from March 11, 2015 00:00:00 UTC to March 13, 2015 23:59:59 UTC. This time period was communicated on the track mailing list and volunteers were solicited to participate. Following the conclusion of the crawl period, we asked the volunteers to extract the ids of the tweets that were gathered and to send us the data for analysis (along with the hardware configuration of the system that performed the crawl). The participants were provided code in Java[2] built on the popular twitter4j library to crawl the streaming API as well as code for extracting and packaging the tweet ids.

In total, four teams participated in this experiment, which we denote T2, T3, T4, and T5. Details about these crawls are shown in Table 1. In addition, we have two crawls (T1

[1] https://dev.twitter.com/streaming/reference/get/statuses/sample

[2] http://twittertools.cc/

TAIA'15, August 13, 2015, Santiago, Chile.

Team	Location	CPU	RAM	OS	Connection
T2	Europe	1 core	1.75 GB	Ubuntu 14.04	0.5 Gbps, wired
T3	USA	4 core	16 GB	Red Hat 4.4.7-3	1.0 Gbps, wired
T4	Asia	4 core	8 GB	Windows 7	1.0 Gbps, wired
T5	USA	2 core	62 GB	Debian Linux 6.0.10	1.0 Gbps, wired
T1	USA	Amazon EC2, US East, instance type t2.small			
T6	USA	Amazon EC2, US West, instance type m1.small			

Table 1: Specifications of the crawls that participated in the experiment.

Crawl	T1	T2	T3	T4	T5	T6
T1	-	.9994	.9995	.9966	.9997	.9994
T2	-	-	.9997	.9967	.9996	.9994
T3	-	-	-	.9968	.9998	.9996
T4	-	-	-	-	.9968	.9966
T5	-	-	-	-	-	.9996
T6	-	-	-	-	-	-

Table 2: Pairwise Jaccard overlap between the different crawls.

Crawl	Distinct Tweets	Missing Tweets
T1	12,781,339	3,990
T2	12,781,706	3,623
T3	12,782,842	2,487
T4	12,746,068	39,261
T5	12,783,638	1,691
T6	12,780,614	4,715
Union	12,785,329	-

Table 3: Total and missing tweets by crawl.

and T6), which are crawls by us that have been running continuously for some time. From Table 1, we see a good variety in terms of hardware configuration, operating system, and geographic location. T1 and T6 were crawled from Amazon's EC2 service, which provides a comparison between crawling from virtual vs. physical machines. For all crawls, T1–T6, we analyzed the gathered tweets from March 11 to March 13 (the crawl period discussed above). For T1 and T6, we additionally analyzed the crawled tweets during all of March 2015.

In addition to providing access to a (supposedly unbiased) sample of the tweet stream, Twitter's streaming API also allows users to specify a set of keywords or a geographic bounding box to restrict the retrieved tweets. Previous work has shown that tweets acquired in this manner may exhibited biases and that they may not be representative of overall Twitter activity (as reflected in the complete "Firehose") [3]. A follow-up study shows that multiple instances tracking the same keywords receive essentially the same tweets [1]. Our study, however, is different in two substantive ways. First, we do not utilize filtering based on keywords or geographic bounding boxes. Second, we are primarily concerned with whether independent "unfiltered" crawls receive the *same* tweets. Our focus is on reproducibility of results, whereas Morstatter et al. [3] are more concerned with external validity of conclusions drawn from observing the sample stream.

3. ANALYSIS

In our first analysis, we computed the pairwise similarity of all participating crawls in term of their Jaccard overlap:

$$\text{Jaccard}(A, B) = \frac{|A \cap B|}{|A \cup B|}. \quad (1)$$

These results are presented in Table 2, which shows that during the March 11–13 evaluation period, teams independently listening to the Twitter stream received almost exactly the same tweets. The lowest Jaccard observed was 0.9966, while the highest was 0.9998.

There were 12,785,329 distinct tweets crawled by the six teams combined, i.e., the cardinality of the union. Table 3 shows the size of each crawl and the number of missing tweets with respect to this union. We see that crawl T5 was able to gather the most tweets among the teams, while T4 crawled the fewest tweets (about an order of magnitude more missing tweets compared to the other crawls). Nevertheless, the fraction of tweets missing from the T4 crawl is negligible in absolute terms (0.31%).

Cross-referencing the above two tables with the description of the crawls in Table 1, we make a few observations: First, it appears that the bandwidth requirements of gathering tweet samples from the Twitter streaming API are relatively modest. Crawl T2 used a connection that has lower bandwidth than T4 and yet T2 managed to collect more tweets than T4 did. Second, the CPU and memory requirements of performing the crawl are similarly modest as well. This bodes well for evaluations on the sample Twitter stream, as it suggests that a single machine is sufficient for processing and participants likely do not need to deal with distributing computations across a cluster (more on this later). Third, it is unclear if there are any geographic effects on the crawled tweets: the crawl from Asia had many more missing tweets, but as previously mentioned, the fraction of missing tweets is negligible overall (0.31%). Finally, there does not seem to be noticeable differences between crawling from a virtualized environment (T1 and T6, on Amazon's EC2 service) and from physical machines (the other crawls). One might have expected the managed environment of a cloud service to provide an advantage, but this does not appear to be the case, as three of the four on-site crawls had fewer missing tweets.

We wanted to gain a better understanding of the missing tweets from the crawls: in particular, what is their distribution? Is it a "slow trickle", i.e., more or less uniform distribution of missing tweets, or "peaky", which might correspond to transient issues? This analysis is shown in Figure 1, which plots the distribution of missing tweets for all six crawls broken down by hour (total of 72 hours for three days). Note that the figure plots the *distribution* (i.e., fraction) of missing tweets; we had to normalize for presentation purposes since the absolute number of missing tweets varied

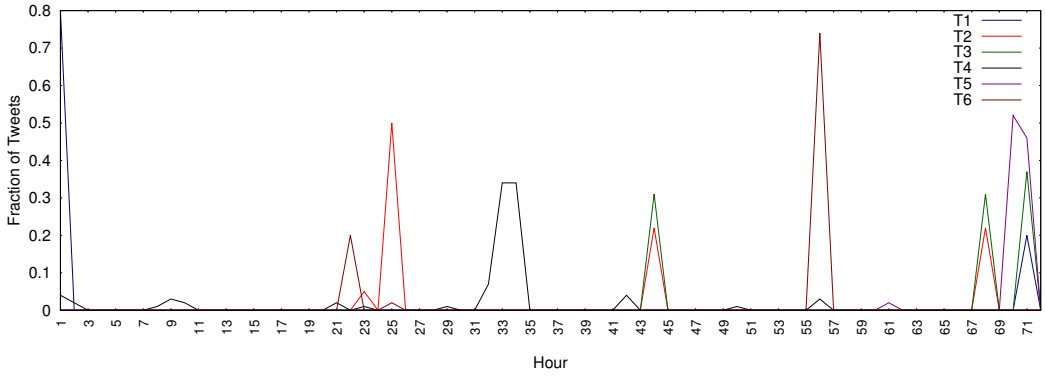

Figure 1: Distribution of missing tweets by hour across all crawls.

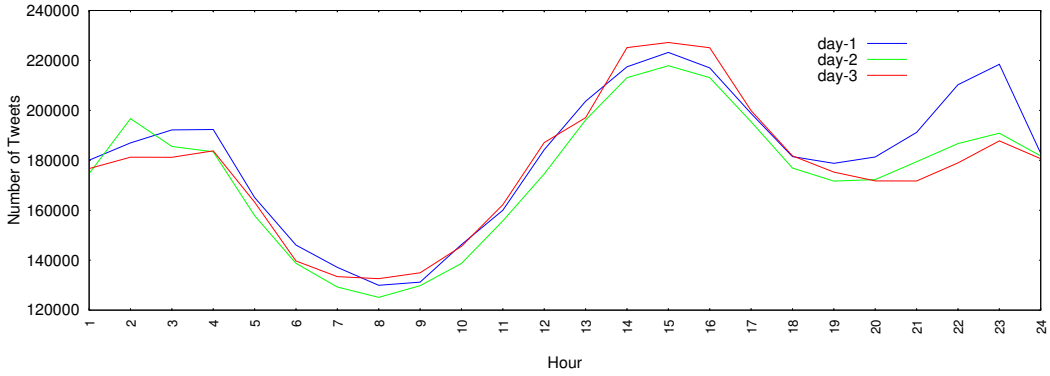

Figure 2: Counts of observed tweets per hour (crawl T5).

by crawl. It is clear that the distributions are non-uniform, without any apparent patterns. For example, most of the tweets missing from T1 are near the beginning of the crawl, whereas other crawls have missing tweets near the end of the observation period. The missing tweets from crawl T4, which had the most missing tweets, were concentrated in a period between hour 31 and hour 35. These results seem to suggest that missing tweets arise from transient issues, as opposed to any persistent systemic effects. These transient issues might stem from the network, system load, or a variety of other issues, but there is nothing we can definitely conclude beyond this.

In our next analysis, we focused on the hourly volume of tweets across the evaluation period. For this, we used crawl T5, which had the fewest missing tweets overall. Figure 2 shows the number of observed tweets in each hour across the three days (starting from midnight UTC). We see variations in tweet volumes that correspond to diurnal cycles of Twitter users (as expected). Note that these volumes represent the composite activities of Twitter users worldwide, not a particular time zone. Each day follows the same general pattern, although there are also deviations from the "typical" daily cycle—for example, near the end of day one.

In Figure 3, we plot the maximum observed tweets per second across all crawls over the experiment period. The overall maximum across these three days is 168 and the minimum is 14. These values are interesting because they quantify the typical and burst volumes that must be handled for systems wishing to operating on the tweet stream in real time—as is the case in the setup of the TREC Microblog track in 2015. These results show that, at least for the sample stream, only modest processing resources are required (with modern hardware). That is, participants likely do not need a distributed online processing framework such as Spark Streaming [6].

The analyses above compare six different crawls from the evaluation period of March 11 to March 13. However, for two of the crawls (T1 and T6), we have tweets gathered from a much longer time frame: our final analysis compared these two crawls across all of March (from the 1st to the 31th, inclusive). We find that T1 crawled 140,993,129 tweets and T6 crawled 140,980,902 tweets, with 140,994,338 tweets in the union. The observed Jaccard is 0.9998. These results suggest that nearly perfect overlap is observed across longer periods of time.

Finally, Figure 4 plots the number of tweets gathered per day throughout all of March. We see that the lines for both T1 and T6 are nearly identical. Another interesting observation is that we do not see any clear weekly patterns, unlike in previous work [5]. That work, however, was over *all* tweets and restricted to certain geographic areas, so perhaps some differences are to be expected.

4. CONCLUSION

Twitter presents unique challenges from the perspective of evaluation for both technical and non-technical reasons, which has necessitated the development of novel evaluation

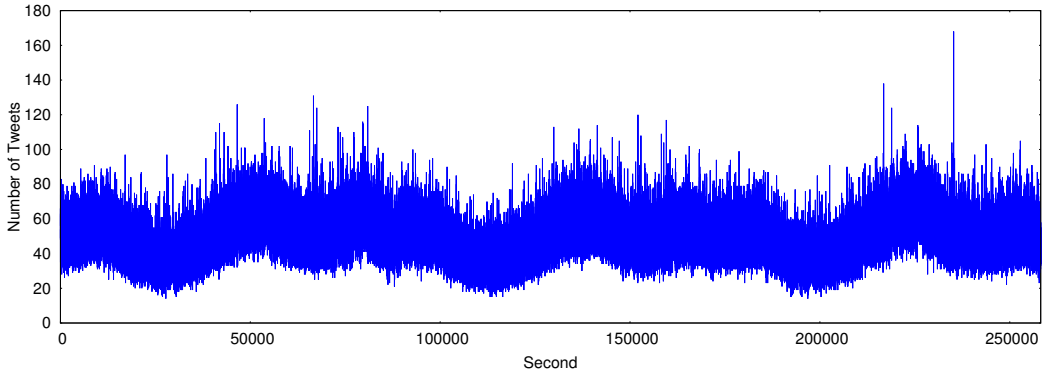

Figure 3: Maximum of tweets observed per second (across all crawls).

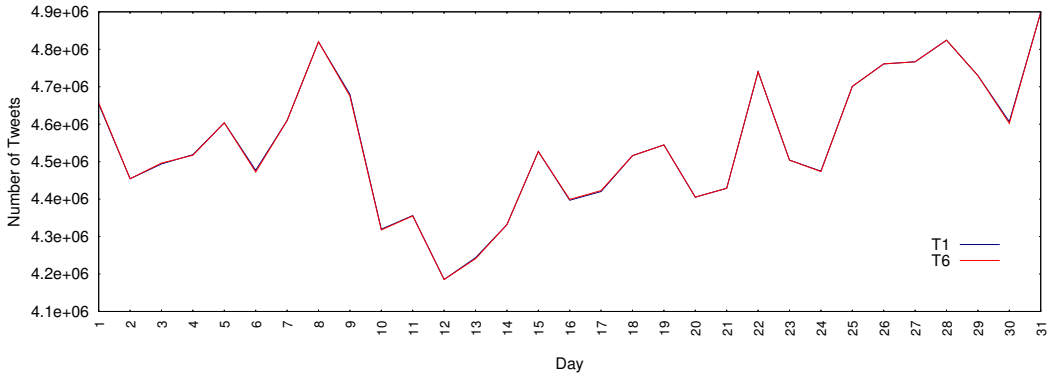

Figure 4: Tweets crawled per day by T1 and T6 from March 1st to March 31st, 2015 (inclusive).

methodologies for community evaluations such as TREC. These new methodologies require validation before their results can be considered trustworthy. In this paper, we explored an evaluation approach where multiple geographically dispersed teams independently crawl the public Twitter sample stream. Analyses show that teams receive nearly the same set of tweets, which means that it is possible to run a "true" real-time evaluation using Twitter data using this approach.

5. ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation under IIS-1218043, CNS-1405688, and the Laboratory for Telecommunication Sciences. Any opinions, findings, conclusions, or recommendations expressed are those of the authors and do not necessarily reflect the views of the sponsors. We are grateful to the participants of the crawling experiment described in this paper, without whom these analyses would not have been possible.

6. REFERENCES

[1] K. Joseph, P. M. Landwehr, and K. M. Carley. Two 1%s don't make a whole: Comparing simultaneous samples from Twitter's streaming API. *Proceedings of the 7th International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction (SBP 2014)*, pp. 75–83, 2014.

[2] J. Lin and M. Efron. Overview of the TREC-2013 Microblog Track. *Proceedings of the Twenty-Second Text REtrieval Conference (TREC 2013)*, 2013.

[3] F. Morstatter, J. Pfeffer, H. Liu, and K. M. Carley. Is the sample good enough? comparing data from Twitter's streaming API with Twitter's firehose. *Proceedings of the 7th International AAAI Conference on Weblogs and Social Media (ICWSM 2013)*, pp. 400–408, 2013.

[4] I. Ounis, C. Macdonald, J. Lin, and I. Soboroff. Overview of the TREC-2011 Microblog Track. *Proceedings of the Twentieth Text REtrieval Conference (TREC 2011)*, 2011.

[5] M. Rios and J. Lin. Visualizing the "pulse" of world cities on Twitter. *Proceedings of the 7th International AAAI Conference on Weblogs and Social Media (ICWSM 2012)*, pp. 717–720, 2013.

[6] M. Zaharia, T. Das, H. Li, T. Hunter, S. Shenker, and I. Stoica. Discretized streams: Fault-tolerant streaming computation at scale. *Proceedings of the 24th ACM Symposium on Operating Systems Principles (SOSP 2013)*, pp. 423–438, 2013.