Twitter API assignment

Readings to prepare for this assignment:

- Field guide to Tweets:
- https://dev.twitter.com/docs/platform-objects/tweets
- Twitter API v1.1:
- https://dev.twitter.com/docs/api/1.1
- Kostas' slides and lecture
- http://blogs.ischool.berkeley.edu/i290-abdt-s12/files/2012/08/Kostas_Trends_Sept_13_2012.pdf
- http://blogs.ischool.berkeley.edu/i290-abdt-s12/2012/09/13/video-lecture-kostas-t-on-how-to-detect-twitter-trends/
- Rion's slides
- http://people.ischool.berkeley.edu/~hearst/twitter_lectures/snow_twitter_api_sept_11.pdf

Recommended Libraries

- Java: download latest stable twitter4j per instructions at http://twitter4j.org/en/index.html
- Python tweepy, see https://github.com/ucbtwitter/getting-started/wiki/Accessing-Twitter-API

Assignment Goals

The goals of this assignment are to get you familiar the Twitter API, to get you started downloading some of your own tweets, and to have you learn a bit about recognizing trends.

Note: this assignment works best if you start early, so you can collect data during two different times, at least a week apart!

Background

As we learned in class, there are two main modes for using the Twitter API: REST and Streaming. In class we went through some details about how use the API both ways and both in Java and Python. See Rion's Snow's lecture notes for more details.

The Streaming API gives you real time access to tweets are they are generated, but you only get access to about 1% of the tweets when you don't specify any details. If you request a specific subset of the tweets

though, such as all the tweets in one region, my understanding is that you get a larger proportion of them.

One way you can use the Streaming API to find out about breaking trends. The REST API gives you more flexibility in what data you access.

Question 1: Trend Detection
(15 points)

a) Connect to the statuses/sample Streaming API. Measure how many tweets per second you're receiving (total tweets and total amount of time). Be sure to state how many seconds you measured this over and when you took the measurement (time and day).

b) Count the rate of specific terms: "the", "of", "lol", "omg", "berkeley", "stanford", "twitter", "facebook", "happy", "sad", "obama", "romney". Again, state how many seconds or minutes you measured this over and when you took the measurement (time and day).

c) Measure the most-frequently mentioned terms over (at least) a 10-minute period. Give the top 10 terms and their raw counts (and estimate frequency per second). State how many seconds or minutes you measured over, and when you took the measurement (time and day).

d) Refer to Kostas' slides or lecture and then build a "background model" of terms over the course of (at least) a 10-minute period. (i) How many unique terms appear in the model? (ii) How many of the terms occur only once? (iii) Plot the distribution of terms. State the time period you measured this over and when you took the measurements.

e) Now measure word frequencies over a new time period, again of at least 10 minutes (it's better if you wait a week before doing this part). Use the background model built in step (d) to provide a set

of expected probabilities for the terms. Using the expected probabilities, find the most 'trending' or 'surprising' terms by calculating the 10 highest-ranked terms according to the chi-square score using the new measurements. Again, state the time periods measured.

f) Compare what you found in (e) to what you found in (c). Discuss.

Question 2: Learning about Twitter Users (10 points)

Now let's get some practice and have some fun with the REST API. Use the REST API for the following.

g) If you don't have a twitter account yet, please set one up. If you don't have any followers, please start following at least a few twitter accounts. Now write code to gather the most recent status update of all the people you follow and store the results.
h) Choose two popular twitter users, each of whom have at least 100K followers. Download their follower lists. Find the intersection of their followers (the followers they have in common). You might want to use Pig for this! Report the names of the users, the number of followers they have, and the number of users at the intersection. Report on a sample of at least 10 users from the intersection if there are that many.

Acknowledgements

Most of question 1 was written by @rion, and other parts written with help from @kostas and @gilad.