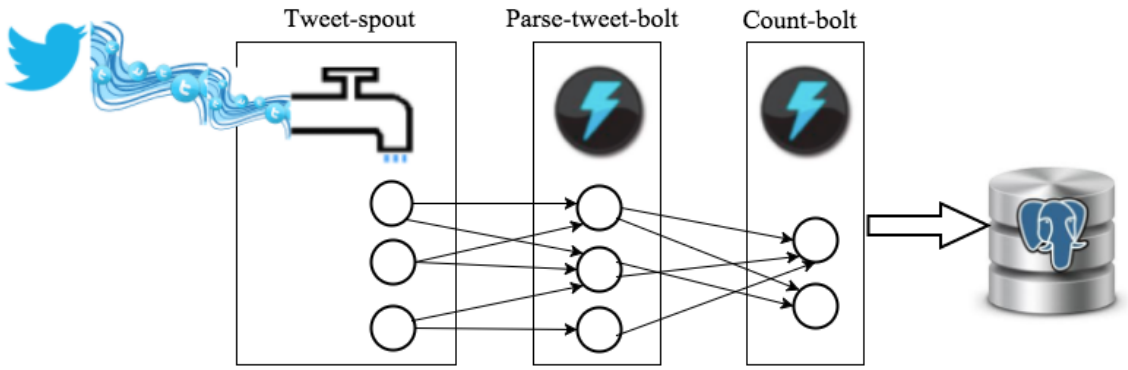
Architecture Overview

>> Application Topology



>> Directory/File Structure

File	Location	Description
tweets.py	EXTweetwordcount/ src/spouts	Access twitter feed through API, pass on the feeds to parse bolt
parse.py	EXTweetwordcount/ src/bolts	Parse through the tweets and separate them into words and eliminate special characters
wordcount.py	EXTweetwordcount/ src/bolts	Take result from parse and count word frequencies
twitterwordcount.clj	EXTweetwordcount/t opologies	Set up the topology structure and determine the flow of information between spouts and bolts
finalresults.py	(root folder)	Gets a word as an argument and returns the total number of word occurrences in the stream; return all word counts if no argument is given
histogram.py	(root folder)	Takes two integers, k1 and k2, and returns all words that their total number of occurrences in the stream is more than or equal to k1 and less than or equal to k2

>> Application Idea

Get twitter feeds from Twitter, parse through the tweets and find out word count in the feed, then store the word count into a table in the database. It can be potentially used for further text analysis. For example, determining the up-trending topic using the frequency of mentioned words.

>> Description of the Architecture

Use spouts to access tweet feeds; use parse bolts to parse through the tweets and obtain words; use count bolt to count the word frequencies, and finally pass word frequency information to a Postgres SQL database to store.

>> File Dependencies

tweets.py requires proper access key to twitter API
parse.py requires connecting to tweets.py
wordcount.py requires information from parse.py and also requires connection to tables in Postgres SQL
finalresults.py will return information after the application has been run and some data has been stored in the Postgres SQL database
histogram.py also need the information from Postgres SQL database to return words along with their frequencies

>> Necessary Information for Running the Application (repeat with readme.txt)

Prerequisites:

1) PostgreSQL is properly installed
2) Python Version is 2.7
3) Storm is properly setup
4) Streamparse is setup (Virtualenv, Lein are installed)

To-do List:
1) Install Tweepy
    pip install tweedy
2) Install psycopg2
    pip install psycopg2
3) Create database on PostgreSQL
    - check PostgreSQL is up and running:
        ps auxwww | grep postgres
    - log into postgres as the postgres user:
        psql -U postgres
    - create database
        create database tcount;
    - change password for postgres user
        ALTER USER postgres WITH PASSWORD ‘password’;
4) Create table on PostgreSQL
    - run createtweetwordcounttable.py
        python createtweetwordcounttable.py
5) Run Tweetwordcount Project

- cd /…/EXTweetwordcount (navigate to the directory)
- sparse run

6) Show Final Result (under the folder where the .py file is stored)
- python finalresults.py
- python finalresults.py you

7) Run Histogram (under the folder where the .py file is stored)
- python histogram.py 3 8 (note: no comma between two numbers)