W205 – Jay Cordes

Exercise 2 Documentation - Spring 2017

Program Goal

This application uses the python Tweepy library to collect live streaming Tweets from the Twitter streaming API and then uses Streamparse to split up the Tweets into words and count up the number of occurrences of each. It then stores those words and counts in a Postgres database table called Tweetwordcount in a database called Tcount. A couple other small python files are used to query the database and output the data to the screen, namely:

(1) Finalresults.py provides a list of all words and counts if no command-line argument is passed or returns the count for a particular word if it is passed to the script.
(2) Histogram.py produces a list of words with counts between two passed-in numbers and also outputs the top 20 words and counts, to help with the histogram requirement.

Possible uses for this application include tracking the most popular words at any given time.

Program Architecture

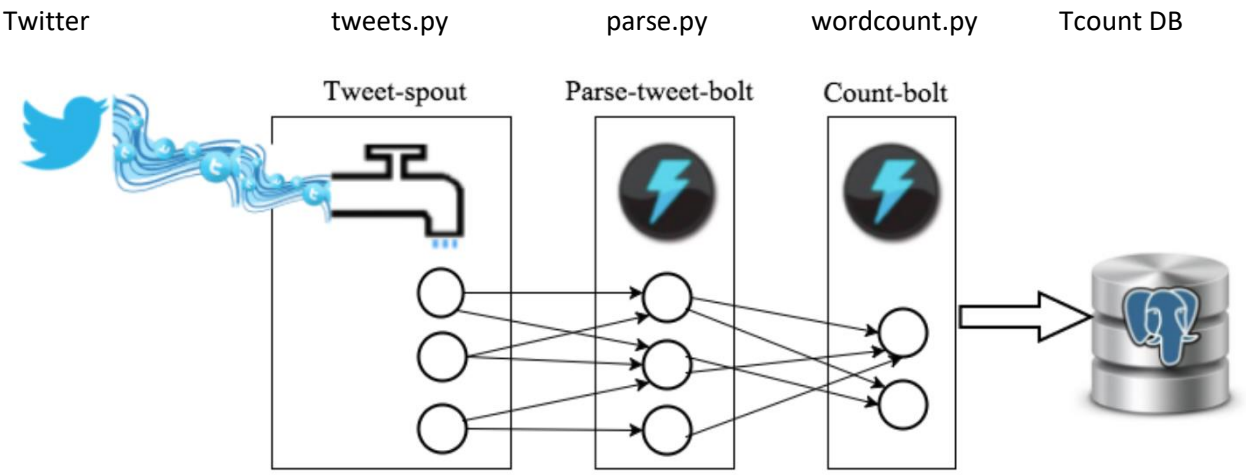

Figure 1: Application Topology

Tweets.py → Collects tweets from Twitter feed and emits them to the parse-tweet-bolt.

Parse.py → Takes emitted Tweets and separates them into words while removing special characters.

Wordcount.py → Takes emitted words, counts frequencies, and updates/inserts into Tweetwordcount table in a Postgres database called Tcount.

File Dependencies

This application has the following dependencies:

(1) Amazon EC2 Instance
(2) Apache Storm
(3) Nedis
(4) Postgres
(5) PsycoPG
(6) Python
(7) Streamparse
(8) Tweepy
(9) Twitter API
(10) VirtualEnv

Directory Structure / Location of key files

\Ex2Tweetwordcount

Readme.txt

Finalresults.py

Histogram.py

Architecture.pdf (this file)

Plot.png (histogram of top 20 words)

\Screenshots (three required screenshots)

*[**StreamParse Files**]*

\src

\spouts

Tweets.py

\bolts

Parse.py

Wordcount.py

\topologies

Tweetwordcount.clj (defines spouts and bolts)

Data Structure

Database: Tcount

Table: tweetwordcount

Word	Text	[PK]	(Stores the words streaming from Twitter)
Count	Int		(Maintains the total count of word occurrences during execution)