W205
Exercise 2
Jen Jen Chen

Directory/File structure

Main directory: exercise_2

Contents:

- Moved files (Twittercredentials.py, hello-stream-twitter.py, psycopg-sample.py
- Architecture.pdf
- Readme.txt
- Plot.png

Subdirectory: extweetwordcount

Contents:

- topologies (contains tweetwordcount.clj)
- src (contains sub subdirectory bolts and spout)
    - bolts contain parse.py and wordcount.py
    - spouts contain tweets.py
- histogram.py
- finalresults.py

Subdirectory: screenshots

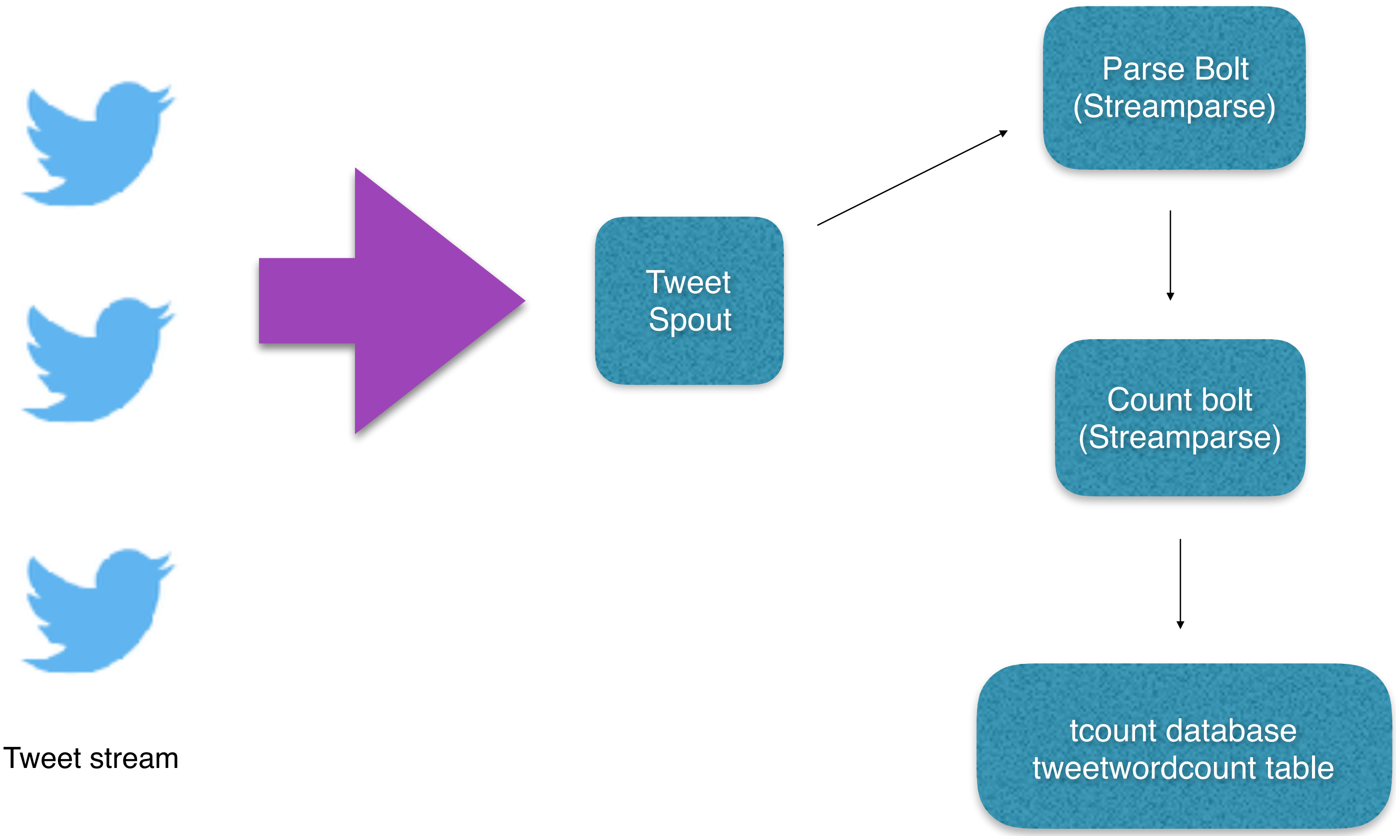
Contents:

- screenshot-stormComponents.png
- screenshot-twitterStream.png
- screenshot-extractResults.png

Application Idea

This application collects tweets in real-time using the Tweeps Library. This stream is then fed into the Storm topology, starting with the tweet-spout which reads the tweets, then is shuffled into the parse-tweet bolt, which uses the Streamparse package to parse out unique words within the store and tallies up the counts of each word, and finally updating the count database in Postgres. The python files (histogram.py, finalresults.py) are designed to, depending on user input, give some form of words and their counts from the total stream.

Architecture Description



File Dependencies

word count.py (in bolts subdirectory) dependencies:

- collections import Counter
- streamparse.bolt import Bolt
- psycopg2
- psycopg2.extensions import ISOLATION_LEVEL_AUTCOMMIT

parse.py (in bolts subdirectory) dependencies:

- re
- streamparse.bolt import Bolt

tweets.py (in spouts subdirectory) dependencies:

- itertools, time
- tweepy, copy
- Queue, threading
- streamparse.spout import Soup

finalresults.py (in extweetwordcount) dependencies:
- pscycopg2
- psycopg2.extensions import ISOLATION_LEVEL_AUTOCOMMIT
- sys

histogram.py (in extweetwordcount) dependencies:
- psycopg2
- psycopg2.extensions import ISOLATION_LEVEL_AUTOCOMMIT
- sys

Running the Application

Please refer to Readme.txt file in exercise_2 directory for step-by-step instructions