Samantha Brownstein
Exercise 2

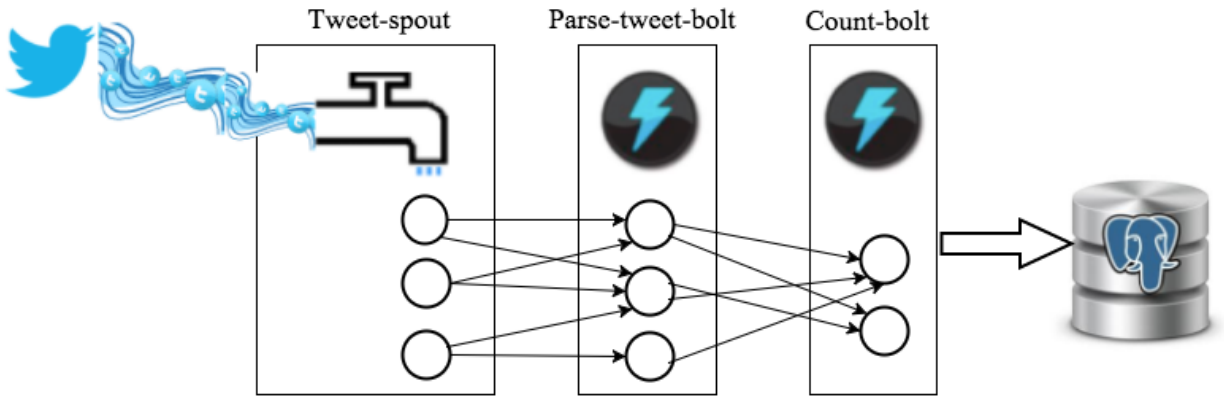
Tweetwordcount Application Architecture

Application Overview and Purpose

The Tweetwordcount application reads a stream of tweets from the Twitter streaming API, parses them, counts the number of each word in the stream, and writes the final results back to a Postgres database that can be queried for analysis.

The overall purpose of the application is to identify real-time trends in social media using live Twitter data by capturing frequently mentioned words and topics. This tool has widespread industry uses, for example in politics to see which candidate is being talked about most, or for businesses trying to understand the needs and interests of their target market.

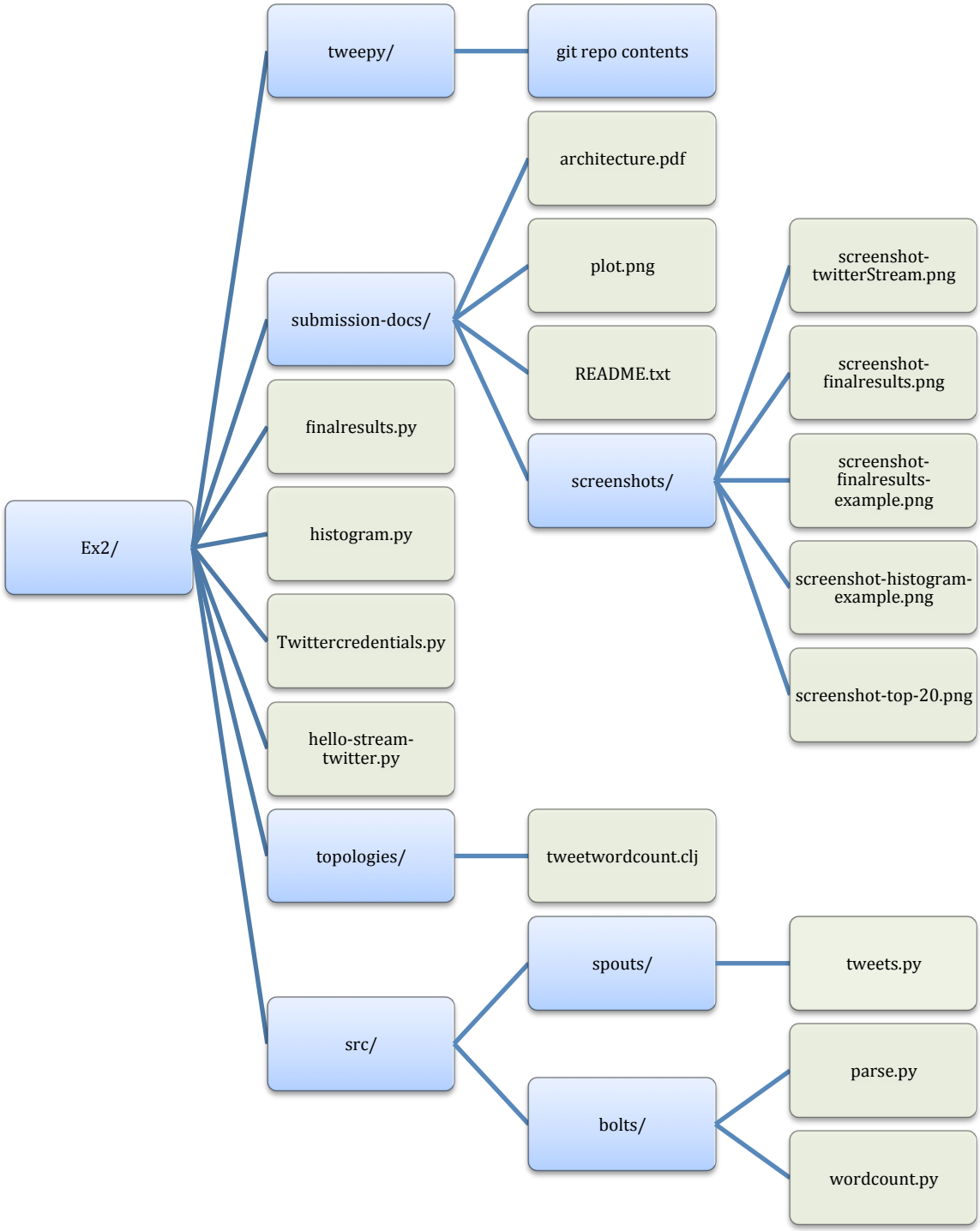
Application Architecture



source: Exercise-2-Subject-205-Real Time Data Processing Using Apache Storm.pdf

The 3:3:2 architecture above shows the Tweetwordcount topology, which creates one spout and two bolts. First, the Tweet-spout component pulls tweets from the Twitter streaming API and emits them to the Parse-tweet-bolt. The Parse-tweet-bolt then parses the tweets and extracts and emits the words to the Count-bolt. The Count-bolt counts the number of occurrences of a given word emitted by the Parse-tweet-bolt and updates the total count for that word in the Tweetwordcount table, which exists in a Postgres database called Tcount.

Directory and File Structure



EC2 Instance Information

Instance ID	i-04ccc45559741c7da	Public DNS	ec2-52-91-170-140.compute-1.amazonaws.com
Instance state	running	Public IP	52.91.170.140
Instance type	m3.medium	Elastic IPs	
Private DNS	ip-172-31-10-213.ec2.internal	Availability zone	us-east-1a
Private IPs	172.31.10.213	Security groups	launch-wizard-8. view rules
Secondary private IPs		Scheduled events	No scheduled events
VPC ID	vpc-0df0316a	AMI ID	UCB W205 Spring 2016 (ami-be0d5fd4)

Instructions for Application Use

1. Clear existing table contents
    a. As user w205, log into postgres. **psql --host=localhost --username=w205 --password --dbname=tcount**
    b. Delete existing tweetwordcount table contents. **TRUNCATE TABLE tweetwordcount;**
    c. Disconnect from postgres. **\q**
2. Run the EX2Tweetwordcount project
    a. **cd Ex2/**
    b. **sparse run**
    c. Cancel application after allowing it to run for desired amount of time. **CTRL+C**
3. Run queries
    a. To find the number of occurrences of a specific word in the stream, run the finalresults script. **python finalresults.py <word>**
    b. For a list of all words in the stream and their total count of occurrences, run the finalresults script with no argument. **python finalresults.py**
    c. For a list of all words with counts between two integers k1 and k2, run the histogram.py script. **python histogram.py k1 k2**