W205 – EXERCISE 2 – SRIRAM RAO - README

GitHub Repository

https://github.com/sriramrao1/exercise_2/

This is a public repository. So you should be able to make Pull requests.
I have provided 'DwMcclary' with access to contribute as well.

Data Source Parameters

Database:	tcount
Table:	tweetwordcount
DB Username used in the code:	w205
DB Password used in the code:	postgres

Repository File Structure:

File Name	Description
extweetwordcount	Name of the stream parse project. Folder contains all files to execute the twitter project
Screenshots	Folder has the screenshots of execution of the application

Finalresults.py	1. This script gets a word as an argument and returns the total number of word occurrences in the stream. For example:
	$ python finalresults.py and
	Count of records in tcount = 30
	word = and
	count = 30
	
	2. Running finalresults.py without an argument returns all the words in the stream and their total count of occurrences, sorted alphabetically in an ascending order, one word per line. For example:
	
	$ python finalresults.py
	Word = and
	Count = 30

	Word = to Count = 33 …
Histogram.py	This script gets two integers k1,k2 and returns all the words that their total number of occurrences in the stream is more or equal than k1 and less or equal than k2. For example: $ python histogram.py 3,8 <word2>: 8 <word3>: 6 <word1>: 3
Twittercredentials	Stores the twitter credentials to run the app

Please note that the name of the project is NOT ex2tweetwordcount. During execution, the ex2tweetwordcount name errored out with the message that name of the project cannot be alphanumeric. Hence the change in the name.

Instructions to run the application:

Pre-Requisites:

1. Clone the scripts from my github repository and store in a EC2 instance folder
2. Start the postgres database
3. Create a database 'tcount' with owner permissions for the user name and password defined in the 'Data Source Parameters' section
4. Create a Streamparse project called 'extweetwordcount'
5. Ensure that the code downloaded from GITHUB is copied to the respective folders of the 'extweetwordcount' project

Instructions:

1. Navigate to /home/w205/extweetwordcount folder
2. Execute 'Sparse run'
3. Run the twitter stream for about a minute. Then stop the stream with ctrl + c
4. Open the tcount database and check the record count in tweetwordcount database. If the record count is insufficient, then run the twitter stream longer
5. Once you have sufficient records in the database, execute finalresults.py:

a. METHOD 1: Execute finalresults.py with a word argument
   $ python finalresults.py and
   Count of records in tcount = 30
   word = and
   count = 30
b. METHOD 2: Execute finalresults.py without a word argument

   $ python finalresults.py
   word = and
   count = 30

   word = to
   count = 33

   …

6. Once you have sufficient records in the database, execute histogram.py:
   $ python histogram.py 3,8
   <word1>: 8
   <word2>: 6
   <word3>: 3

Screenshots

Project code layout

w205@ip-172-31-53-4:~

```
-rw-r--r-- 1 w205 w205  456 Apr  5 02:50 fabfile.py
drwxrwxr-x 2 w205 w205 4096 Apr 12 03:17 logs
-rw-rw-r-- 1 w205 w205  531 Apr  5 02:50 project.clj
drwxrwxr-x 4 w205 w205 4096 Apr  5 02:50 src
-rw-r--r-- 1 w205 w205  456 Apr  5 02:50 tasks.py
drwxrwxr-x 2 w205 w205 4096 Apr  5 03:26 topologies
drwxrwxr-x 2 w205 w205 4096 Apr  5 03:26 virtualenvs
[w205@ip-172-31-53-4 extweetwordcount]$ cd ..
[w205@ip-172-31-53-4 ~]$ ls -l
total 274600
-rw-r--r--  1 root root       657 Mar 27 02:44 Twittercredentials.py
-rw-rw-r--  1 w205 w205       693 Sep 28  2015 derby.log
drwxrwxr-x  8 w205 w205      4096 Apr  5 03:00 extweetwordcount
-rw-r--r--  1 root root      1934 Apr 12 02:17 finalresults.py
-rw-r--r--  1 root root      1748 Mar 27 23:44 hello-stream-twitter.py
-rw-r--r--  1 root root      1492 Apr 12 03:13 histogram.py
drwxrwxr-x  5 w205 w205      4096 Sep 28  2015 metastore_db
-rw-r--r--  1 root root      1337 Mar 27 02:44 psycopg-sample.py
-rw-rw-r--  1 w205 w205      3666 Jan 25 21:44 setup_spark.sh
-rw-rw-r--  1 w205 w205 280869269 Sep  9  2015 spark-1.5.0-bin-hadoop2.6.tgz
drwxr-xr-x 11 w205 w205      4096 Aug 31  2015 spark15
[w205@ip-172-31-53-4 ~]$
```

Stream Parse – Project run

w205@ip-172-31-53-4:~/extweetwordcount

```
 main
    args.func(args)
  File "/usr/lib/python2.7/site-packages/streamparse/cli/run.py", line 81, in ma
in
    debug=args.debug)
  File "/usr/lib/python2.7/site-packages/streamparse/cli/run.py", line 52, in ru
n_local_topology
    run(full_cmd)
  File "/usr/lib/python2.7/site-packages/invoke/__init__.py", line 27, in run
    return Context().run(command, **kwargs)
  File "/usr/lib/python2.7/site-packages/invoke/context.py", line 53, in run
    return runner_class(context=self).run(command, **kwargs)
  File "/usr/lib/python2.7/site-packages/invoke/runners.py", line 276, in run
    raise exception
KeyboardInterrupt
[w205@ip-172-31-53-4 extweetwordcount]$ sparse run
Running extweetwordcount topology...
Routing Python logging to /home/w205/extweetwordcount/logs.
Running lein command to run local cluster:
lein run -m streamparse.commands.run/-main topologies/extweetwordcount.clj -t 0
--option 'topology.workers=2' --option 'topology.acker.executors=2' --option 'st
reamparse.log.path="/home/w205/extweetwordcount/logs"' --option 'streamparse.log
.level="debug"'
```

Twitter Stream

```
w205@ip-172-31-53-4:~/extweetwordcount
count-bolt s*: 1
23071 [Thread-31] INFO  backtype.storm.task.ShellBolt - ShellLog pid:9621, name:
count-bolt House: 1
23239 [Thread-31] INFO  backtype.storm.task.ShellBolt - ShellLog pid:9621, name:
count-bolt Version: 1
23280 [Thread-35] INFO  backtype.storm.task.ShellBolt - ShellLog pid:9623, name:
count-bolt About: 1
23350 [Thread-35] INFO  backtype.storm.task.ShellBolt - ShellLog pid:9623, name:
count-bolt to: 2
23406 [Thread-31] INFO  backtype.storm.task.ShellBolt - ShellLog pid:9621, name:
count-bolt have: 1
23423 [Thread-35] INFO  backtype.storm.task.ShellBolt - ShellLog pid:9623, name:
count-bolt To: 1
23525 [Thread-31] INFO  backtype.storm.task.ShellBolt - ShellLog pid:9621, name:
count-bolt Wet: 1
23530 [Thread-35] INFO  backtype.storm.task.ShellBolt - ShellLog pid:9623, name:
count-bolt The: 1
23621 [Thread-35] INFO  backtype.storm.task.ShellBolt - ShellLog pid:9623, name:
count-bolt Swisha: 1
23706 [Thread-35] INFO  backtype.storm.task.ShellBolt - ShellLog pid:9623, name:
count-bolt Remix: 1
23722 [Thread-31] INFO  backtype.storm.task.ShellBolt - ShellLog pid:9621, name:
count-bolt economy: 1
```

Postgres Table word count

```
w205@ip-172-31-53-4:~
 count
-------
   996
(1 row)

tcount=# SELECT count(*) from tweetwordcount;
 count
-------
  1000
(1 row)

tcount=# SELECT count(*) from tweetwordcount;
 count
-------
  1005
(1 row)

tcount=# SELECT count(*) from tweetwordcount;
 count
-------
  1031
(1 row)

tcount=#
```

Execute FinalResults.py – with an argument (word exists)

```
w205@ip-172-31-53-4:~
[w205@ip-172-31-53-4 ~]$ python finalresults.py and
Count of records in tcount = 1
word = and
count = 30

[w205@ip-172-31-53-4 ~]$
```

Execute FinalResults.py – with an argument (word does not exist)

```
w205@ip-172-31-53-4:~

[w205@ip-172-31-53-4 ~]$ python finalresults.py hello
Count of records in tcount = 0
[w205@ip-172-31-53-4 ~]$
```

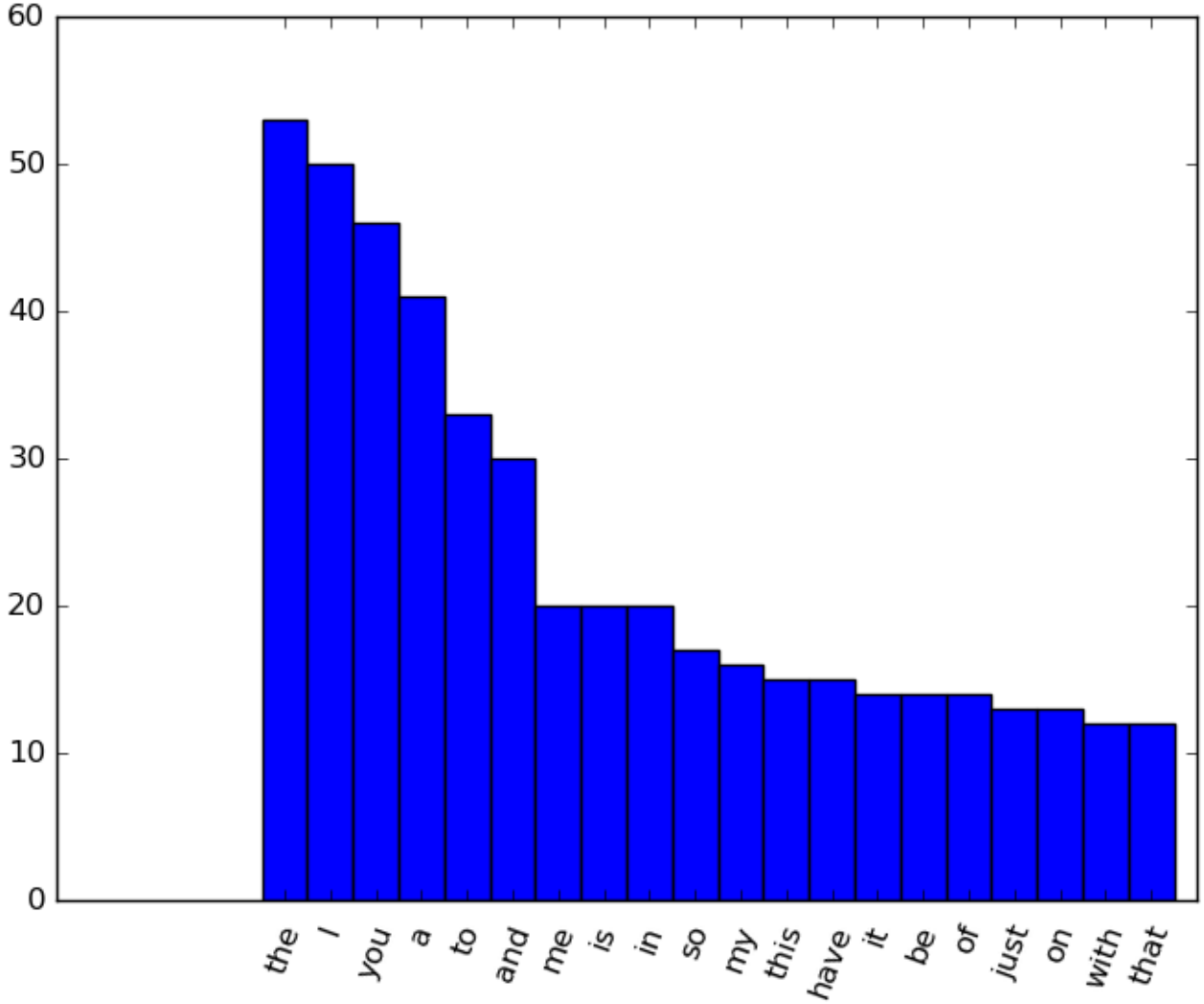
Execute Histogram.py – with correct arguments

```
w205@ip-172-31-53-4:~

[w205@ip-172-31-53-4 ~]$ python histogram.py 4,10
Count of records = 69
best: 5
want: 4
still: 4
know: 9
at: 5
how: 5
do: 7
When: 6
get: 10
your: 6
man: 4
Facebook: 4
said: 4
This: 4
right: 4
when: 7
time: 5
we: 6
The: 9
video: 5
```

Execute Histogram.py – with incorrect arguments

```
w205@ip-172-31-53-4:~

[w205@ip-172-31-53-4 ~]$ python histogram.py 4, 10
Incorrect number of arguments
[w205@ip-172-31-53-4 ~]$
```

Plot:


60
50
40
30
20
10
0
the
I
you
a
to
and
me
is
in
so
my
this
have
it
be
of
just
on
with
that