# Real-Time Tweet count using Apache Storm

# Application:

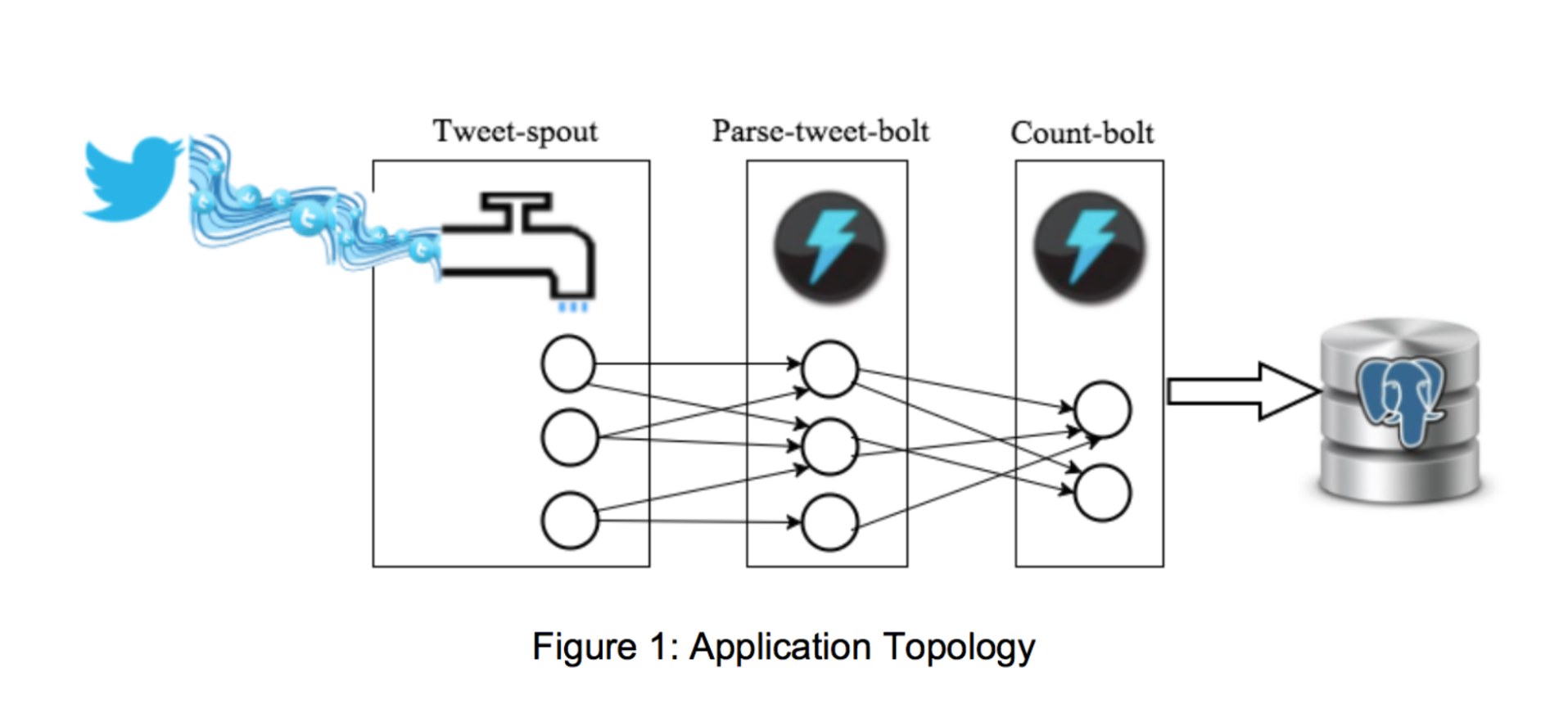
The idea of the application is to gain insights into tweets in real time. For that we analyze the tweets in real time. To attain real time analysis of tweets while having fault tolerance, we use Apache storm and Postgres SQL DB for storing aggregate data.

# Architecture:

Figure 1 ([Reference document](https://github.com/UC-Berkeley-I-School/w205-spring-17-labs-exercises/blob/master/exercise_2/Exercise-2.pdf) )shows the high-level architecture of the application. It basically comprises of 3 components

1. Spout: The spouts read the tweets in real time from twitters’ Streaming API
2. Parse Bolt: The parse bolt processes the tweets read by the spout and splits individual tweet into words.
3. Count Bolt: The count bolt tokenizes the tweets from the parse bolt and updates the counters in the DB.
4. SQL DB: The db stores the counts for individual words in Tweetwordcount table.

The spouts and bolts are implemented in Python and [StreamParse](https://github.com/Parsely/streamparse) is used to run them via Apache Storm. Tweepy library is used to read tweets from twitters’ Streaming API and psycopg2 library is used to talk to PostgresSQL.



Dependencies

The following tools and libraries are needed to run the application

**Tools**:

1. Apache Storm
2. PostgresSQL
3. StreamParse
4. Python 2.7

**Libraries**:

1. tweepy
2. psycopg2

File Structure:

|  |  |  |
| --- | --- | --- |
| **File Name** | **Location** | **Description** |
| tweetwordcount.clj | \extweetwordcount\topologies\tweetwordcount.clj | Contains the Storm topology |
| tweets.py | \extweetwordcount\src\spouts\ | Reads the tweets from twitters’ streaming API |
| parse.py | \extweetwordcount\src\bolts\ | Tokenizes individual tweets into words |
| wordcount.py | \extweetwordcount\src\bolts\ | Counts the words and updates the DB |
| finalresults.py | \extweetwordcount\ | Reads the word from command line argument and queries the DB to get the counts |
| histogram.py | \extweetwordcount\ | Prints words with counts within a range |
| runapp.sh | \extweetwordcount\ | Runs the application   1. Setup DB 2. Run application |
| setupdb.sql | \extweetwordcount\ | SQL file for creating table and stored procedure |

# Steps to run application:

1. Clone the repository “EXTweetwordcount”

***git clone https://github.com/stp8954/EXTweetwordcount.git***

1. Check whether PostgreSQL is up and running:

ps auxw | grep postgres

1. If not, change your current path to /data directory

cd /data

1. And start Postgres

start\_postgres.sh

1. CD to EXTweetwordcount
2. Make runapp.sh executable

Chmod 755 ./runapp.sh

1. Run script “runapp.sh”.This runs the storm topology for 120 seconds.

**If you see this in the cmd window**

WARNING: You're currently running as root; probably by accident.

Press control-C to abort or Enter to continue as root.

Set LEIN\_ROOT to disable this warning.

**Press ENTER**

1. To view list of all words in the DB, “finalresults.py” script can be run “python finalresults.py”
2. To view list of a specific word in the DB, “finalresults.py” script can be run “python finalresults.py wordname”
3. To create histogram of words with counts in a given range histogram.py script can be used. “python histogram.py 5,15”