W205 Exercise 2 Architecture Definition

Sombiri Enwemeka

INTRODUCTION

The application created for exercise 2 captures live Twitter data, stores the data, and performs analysis on it. To complete this task, the application uses Apache Storm with Streamparse or data collection, a Postgres data base for storage, and various python scripts for analysis and data representation. This document describes the application's architecture, directory and structure, dependencies, and usage instructions.

APPLICATION ARCHITECTURE

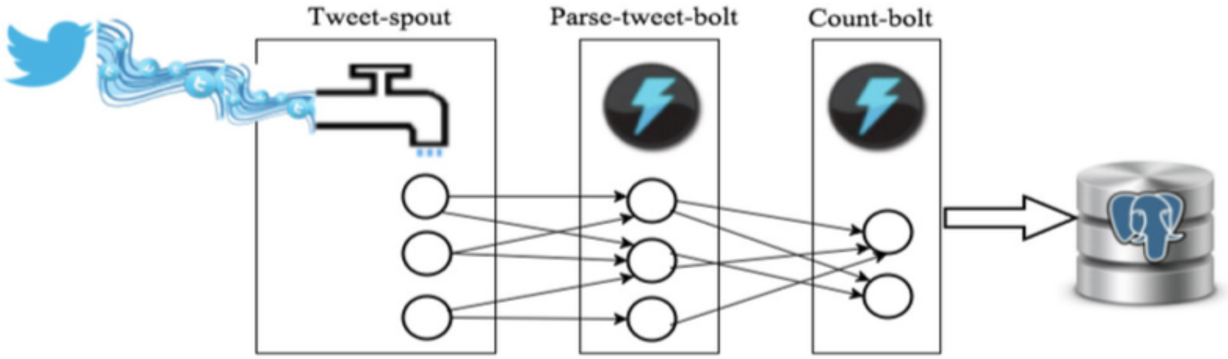


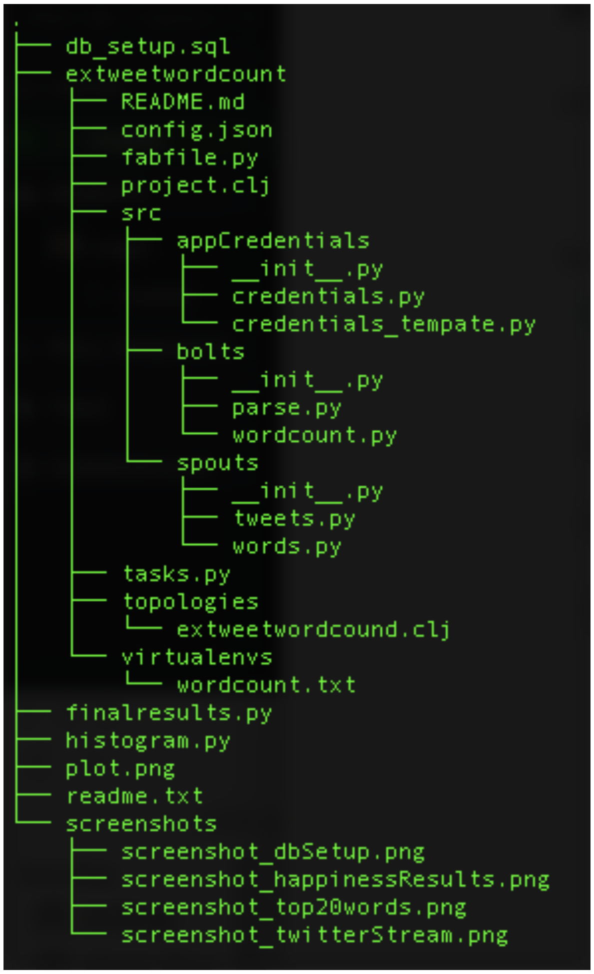
Figure 1 Application Topology

Figure 1 Shows at a high-level how data flows through the application. The tweet-spout captures live twitter data using the Tweepy to access the Twitter API. This data is fed to the Parse-tweet-bolt which identifies words in the data stream and attempts to remove special characters or other gibberish text before feeding the data into the Count-bolt which aggregates the number of each word encountered within the data stream. This list of words and associated counts are then stored in the **tcount** database table **tweetwordcount.** Because there are multiple Count-bolt threads it is possible for data concurrency issues to occur when both threads attempt to access the database at the same time. Postgres handles concurrency issues internally, processes that modify data acquire locks as necessary to prevent data loss and maintain consistency.

APPLICATION COMPONENTS

- **Amazon EC2** – A scalable computing platform in the Amazon Web Services Cloud. The application is deployed here but could also run from any machine which has the correct dependencies.
- **Apache Storm** – A distributed real-time computation system. It is an open source software that make it easy to reliably process unbounded streams of data in real-time processing.
- **PostgreSQL** – Open source relational database used to store words and counts.
- **Psycopg** – Python module used to allow python scripts to access Postgres database
- **Python** – High-level interpreted programming language used to process received data.
- **Streamparse** – Streamparse integrates Python with Apache Strom.
- **Tweepy** – Python module used to access and manage the Twitter API

DIRECTORY STRUCTURE



```
.
├── db_setup.sql
├── extweetwordcount
│   ├── README.md
│   ├── config.json
│   ├── fabfile.py
│   ├── project.clj
│   ├── src
│   │   ├── appCredentials
│   │   │   ├── __init__.py
│   │   │   ├── credentials.py
│   │   │   └── credentials_tempate.py
│   │   ├── bolts
│   │   │   ├── __init__.py
│   │   │   ├── parse.py
│   │   │   └── wordcount.py
│   │   └── spouts
│   │       ├── __init__.py
│   │       ├── tweets.py
│   │       └── words.py
│   ├── tasks.py
│   ├── topologies
│   │   └── extweetwordcound.clj
│   └── virtualenvs
│       └── wordcount.txt
├── finalresults.py
├── histogram.py
├── plot.png
├── readme.txt
└── screenshots
    ├── screenshot_dbSetup.png
    ├── screenshot_happinessResults.png
    ├── screenshot_top20words.png
    └── screenshot_twitterStream.png
```

Figure 2 Exercise 2 file structure

W205 Exercise 2 Architecture Definition

Sombiri Enwemeka

DEPENDENCIES

- Apache Storm
- PostgreSQL database installation
- Psycopg
- Python 2.7
- Streamparse
- Tweepy
- Twitter Credential

ADDITIONAL INFORMATION

See readme.txt