Centaurus Mid-Point:Twitter Sentiment Visualization

Team Members

Rohan Gokhale, Travis Robinson, Chase Hu

Submission Contents

- CentaurusMidPointReport.pdf
- ZIP file for repository code and instructions
- Link to website demo: http://web.engr.oregonstate.edu/~robitrav/capstone_test_location/mainpage.php

Project Status

The Centaurus team is about halfway done with the Twitter Sentiment Visualization project. We have been able to produce several visualizations that show scores of different TV shows for every state. We have successfully started a cluster using Google Cloud Platform, and created a program to continuously gather tweets that pertained to the content we want to analyze. We stopped collecting when we had about 98K tweets. After gathering the raw tweets, we processed them to filter out any that did not contain the exact name of the content we want to analyze, did not contain information about what state the tweet originated from, or were not in english. We also created a program to conduct a sentiment analysis on those tweets to produce a number between 0 and 1. That data has been loaded into a HDFS, been made into an HQL data and table, and analyzed. We then used Tableau to create visualizations to compare how people across the country felt about each title we collected data for, and which titles were the most well-liked in each state. In the remaining weeks of the project, we intend to refine the web page that the visualizations will be hosted, upload our visualizations to it, (if possible) refine on and improve our sentiment scorer, and ensure via continued testing that all aspects of the project continue to work.

User Instructions

(Git link at: https://github.com/travisrobinson2006/CS467)

To get access to the master node:

1) Open the email to accept the invite to our project (should be from Google Cloud Platform).

2) Create a Google Cloud Platform account, if you do not already have one.

3) Log into Google Cloud Platform. Next to the text "Google Cloud Platform," click and select our project (the name is "hdp-dataproc")

4) Click the "Products and Services" button (looks like three horizontal lines at the top left of the window). Then scroll down and select "Compute Engine."

5) You should now be brought to a page where you can see the VM instances that comprise our cluster (should be at https://console.cloud.google.com/compute/instances?project=hdp-dataproc&authuser=1). The first one will be name 'tweet-cluster-m' (m for master). To the right of the name, under the "Connect" column, click the "SSH" button. This will open a new window and transfer all keys automatically to SSH into the instance.

6) Once the console is brought up, enter "cd .." followed by "ls" to see the directories for each user.

7) Enter "cd robitrav" followed by "cd CS467". This will bring you to a directory that contains our project files (identical to the ones on our GitHub page). (Alternatively, due to permissions set-up, such as when running the below hive script, it may be best to clone the repository into your own user directory, via git clone https://github.com/travisrobinson2006/CS467.git).

To collect tweets:

1) Navigate into get_tweets directory

2) Run Python script called run_me_to_get_tweets

This will be done from command line via 'python run_me_to_get_tweets.py' (without the quotes)*

This script will call on the tweet_scraper script contained int he acquire_tweets directory, which uses the Twitter streaming API to collect some portion of the tweets as they are being sent out. The script will also run the tweet_cleaner script, which is also contained in the acquire_tweets directory. The cleaning script will check that the scraped tweets have appropriate/usable locations, and cleans up the text a little bit (discarding hashtags, etc). It also converts the tweets to a tab delimited text file, which is generally preferred by HDFS.

The run_me_to_get_tweets script will run an initial cleaning to take care of anything that may have been left behind from the last time that the script was run, and then runs the cleaning script once an hour. The tweet_scraper runs the dumpjson script once an hour

as well. This is to prevent a large build-up of data that could lead to large processing times if data was only dumped/cleaned once.

The dumpjson script will place timestamped json files in the directory unclean_tweets, located in the acquire_tweets directory. The tweet_cleaner script will append all cleaned tweets to the file called tweets_ready_for_use, located inside the clean_tweetsdirectory, which in turn is located in the get_tweets directory.

To get a sentiment score for each tweet

1) In the sentiment_analyzer folder (located in the CS467 directory), open the file named "textblobAnalyzer.py". On lines 21 and 22, you will see a place to specify the names of the input file (the tweets to be scores) and the output file (the file that will contain the name of the show a tweet was about, the state it originated from, and its sentiment score). The input file should be the one you produced in the previous set of steps (tweets_ready_for_use, which will need to be copied into the sentiment_analyzer directory (via cp -i /home/robitrav/CS467/get_tweets/clean_tweets/tweet s_ready_for_use /home/robitrav/CS467/sentiment_analyzer/). You may choose whatever output file name you like.

2) Run the program by entering "python textblobAnalyzer.py" in the command line. Depending on the size of the input file, this may take up to 2 minutes.

3) In the same folder, you will now find the output file that you named in step 1.

To enter scores file into the Hadoop Distributed File System:

1) To put the output file you just created into the HDFS (distributed across the three worker nodes), type "hdfs dfs -put *outputfilename* /databases"

2) Check that your file was transferred successfully by entering "hadoop fs -ls /databases". The output file name should appear.

To use hive:

1) From any location within the cluster instance, enter at command line: 'hive;' or 'hive' (different terminals need the semi-colon)**. This will launch the hive shell, where we will be able to use extract the tweet sentiment scores for shows and states from our HQL database.

2) From the hive command line (denoted by the line starting with 'hive>') enter 'use testdb;' This tells Hive which database we want to use. testdb is the database that is currently storing the sentiment data.

3) Enter at the hive command line (or copy and paste):

INSERT OVERWRITE LOCAL DIRECTORY '*directory****'

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ','

SELECT content_name, user_location, avg(score) FROM tweets_2_15 GROUP BY content_name, user_location;

This will create a csv file called 000000_0 (kept in the /home/robitrav/temp directory). This will be the file**** that is used by Tableau to generate our maps and state graphs.

Outputting to a CSV file:

INSERT OVERWRITE LOCAL DIRECTORY '/home/robitrav/temp' ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' SELECT content_name, user_location, avg(score) FROM tweets_2_15 GROUP BY content_name, user_location;

*To allow yourself the ability to run other programs, navigate directories, etc it's recommended to run the run_me_to_get_tweets script in the background, via the command line command python 'run_me_to_get_tweets.py &'

**It will most likely be easier to navigate to the main (the CS467) directory and enter at the command line 'hive -hiveconf dir='*your_choice_of_dir*' -f hive_script.hql', where *your_choice_of_dir* is the directory name you'd like to use. This will save the csv file in the directory name you specify.

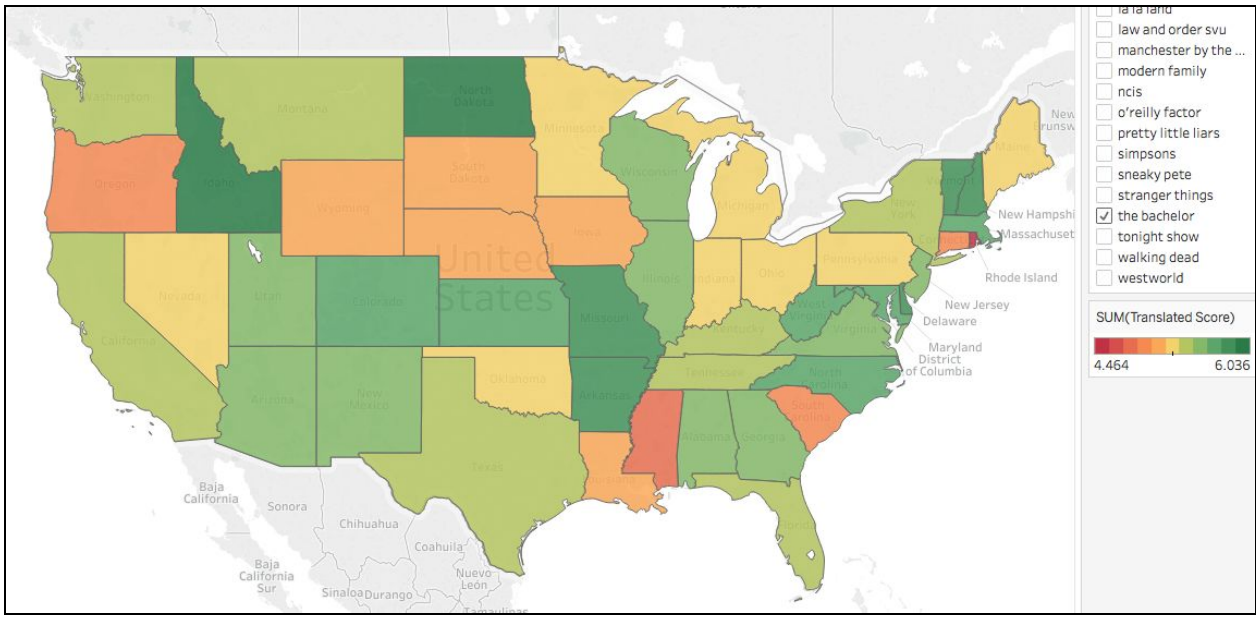
***Your directory of choice, for testing purposes the directory 'home/robitrav/temp' was used, though due to permissions you may or may not be able to create a file or directory in the robitrav user directory

****The file will actually need to be converted to an Excel file, which can be done by most spreadsheet programs.

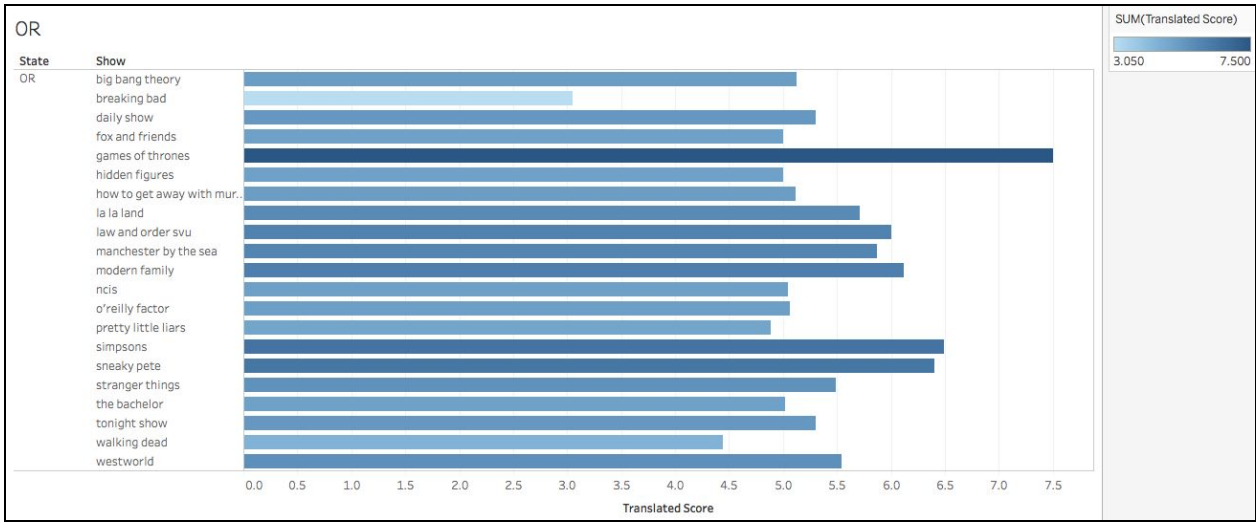
Visualizations

After we connected Tableau to the CSV file generated using Hive commands, we used the data to generate different views. Here are several screenshots of the visualizations we developed from the Twitter sentiment scores based on location:

National Sentiment for the TV Show “The Bachelor”



State Sentiment for All TV Shows



TV Show Sentiment by State of “The Bachelor”

References

http://xpo6.com/list-of-english-stop-words/

http://pbpython.com/pandas-list-dict.html

https://www.analyticsvidhya.com/learning-paths-data-science-business-analytics-business-intelligence-big-data/tableau-learning-path/

http://mpqa.cs.pitt.edu/#subj_lexicon

https://www.youtube.com/watch?v=ziqx2hJY8Hg
https://www.mapr.com/services/mapr-academy/big-data-hadoop-online-training
https://blog.insightdatascience.com/spinning-up-a-free-hadoop-cluster-step-by-step-c406d56bae42#.cqg85eh3o
https://www.youtube.com/watch?v=y3nFfsTnY3M

http://spark.apache.org/docs/latest/api/python/pyspark.sql.html
http://blog.cloudera.com/blog/2012/09/analyzing-twitter-data-with-hadoop/
https://www.tableau.com/academic/students
https://dev.twitter.com/streaming/overview/processing
https://dev.twitter.com/streaming/overview
https://dev.twitter.com/streaming/public
https://dev.twitter.com/streaming/overview/connecting
http://hc.apache.org/httpcomponents-client-ga/
https://dev.twitter.com/oauth/overview