Individual Assignment

Introduction

The goal of this assignment is to practice Apache Hive.

First you should ingest some tweets relevant to you (you are free to choose your topic) the same way we did during NiFi lab, and to perform some analytics on them.

Additionally you have available a dataset called **sentiment-dictionary** (in individual-assignment-resources.zip). This dataset is a **tab** delimited file containing english words (lower case) with their sentiment polarity. It has no header but this is schema schema (relevant columns in bold) :

type:string
length:integer
word:string
word_type:string
stemmed:string
polarity:string

This is a sample of the contents

weaksubj	1	abandoned	adj	n		negative
weaksubj	1	abandonment	noun	n		negative
weaksubj	1	abandon	verb	y	negative	
strongsubj	1	abase	verb	y	negative	
strongsubj	1	abasement	anypos	y		negative
strongsubj	1	abash	verb	y	negative	
weaksubj	1	abate	verb	y	negative	

The following is an example of tweet so that you can understand how data is structured:


```
{
  "contributors": null,
  "coordinates": null,
  "created_at": "Sun Nov 18 20:19:30 +0000 2018",
  "entities": {
    "hashtags": [{"indices": [109,117], "text": "Bitcoin"} ], "symbols": [],
    "urls": [{"display_url": "bit.ly/2ON1Mln", "expanded_url": "http://bit.ly/2ON1Mln", "indices": [85,108], "url": "https://t.co/oGkQDBQsH7"} ],
    "user_mentions": []
  },
  "favorite_count": 0,
  "favorited": false,
  "filter_level": "low",
  "geo": null,
  "id": 1064251774738067461,
  "id_str": "1064251774738067461",
  "in_reply_to_screen_name": null,
  "in_reply_to_status_id": null,
  "in_reply_to_status_id_str": null,
  "in_reply_to_user_id": null,
  "in_reply_to_user_id_str": null,
  "is_quote_status": false,
  "lang": "en",
  "place": null,
  "possibly_sensitive": false,
  "quote_count": 0,
  "reply_count": 0,
  "retweet_count": 0,
  "retweeted": false,
  "source": "<a href=\"https://ifttt.com\" rel=\"nofollow\">IFTTT</a>",
  "text": "In 2017 Bitcoin Went From $5.5k to $19k in 33 Days, Not Impossible in 2019 - newsBTC https://t.co/oGkQDBQsH7 #Bitcoin",
  "timestamp_ms": "1542572370596",
  "truncated": false,
  "user": {
    "contributors_enabled": false,
    "created_at": "Sat Dec 25 12:12:16 +0000 2010",
    "default_profile": false,
    "default_profile_image": false,
    "description": "Who is love Computer. #SKIDDOW #CyberSecurity #HackedBy #Anonymous News feed. \n#Bitcoin News",
    "favourites_count": 528,
    "follow_request_sent": null,
    "followers_count": 221,
    "following": null,
    "friends_count": 33,
    "geo_enabled": false,
    "id": 230423908,
    "id_str": "230423908",
    "is_translator": false,
    "lang": "en",
    "listed_count": 177,
    "location": null,
    "name": "SKIDDOW",
    "notifications": null,
    "profile_background_color": "000000",
    "profile_background_image_url": "http://abs.twimg.com/images/themes/theme5/bg.gif",
    "profile_background_image_url_https": "https://abs.twimg.com/images/themes/theme5/bg.gif",
    "profile_background_tile": false,
    "profile_banner_url": "https://pbs.twimg.com/profile_banners/230423908/1462000239",
    "profile_image_url": "http://pbs.twimg.com/profile_images/626267268448522240/H_9Mhamn_normal.png",
    "profile_image_url_https": "https://pbs.twimg.com/profile_images/626267268448522240/H_9Mhamn_normal.png",
    "profile_link_color": "9266CC",
    "profile_sidebar_border_color": "000000",
    "profile_sidebar_fill_color": "000000",
    "profile_text_color": "000000",
    "profile_use_background_image": false,
    "protected": false,
    "screen_name": "SKIDDOW_KIDDO",
    "statuses_count": 63795,
    "time_zone": null,
    "translator_type": "none",
    "url": "https://www.skiddow.net",
    "utc_offset": null,
    "verified": false
  }
}
```


You have additional information about tweet json structure in the following links:

[Tweet JSON Introduction]
[Tweet JSON Object Documentation]

Instructions

Each student should deliver a zip file name <your_name>.zip with the following contents:

- zeppelin notebook with all the required Hive statements.
- files with the tweets ingested to complete this assignment.

Please upload the zip file to campus/blackboard before the due date

due date: Friday 2020/12/18 23:59:59

Plagiarism will not be tolerated and is considered a sackable offence.
You can help your classmates but please do not share your work directly with them.

If you need help please post your questions and doubts on the discussion board.

Useful links

Hive DDL
Hive Functions

Statements

1. Create a database named **2020s1_<your_name>**.

2. Select the database you just created so that all the tables you are going to create belong to that database.

3. Create an external table named **sentiment_dictionary** with the csv file provided.

   You first have to upload this file to HDFS.

   Notice that this csv has no header and is delimited with the character tab:
   with serdeproperties ("separatorChar" = "\t")

4. Create an external table named **tweets_json** over the HDFS directory where your tweets are located.

   You don't need to reference all the fields in a tweet, just the ones to solve your assignment.
   You can reuse the create table statement we use during Hive lab as a template (you may have to add or remove some fields).

5. Write a query that returns the total number of tweets in table **tweets_json**.

   Annotate both the number of records and the amount of seconds that it took.

6. Create a managed table **tweets_parquet** with the same schema as tweets_json but stored in parquet format.

   create table tweets_parquet stored as parquet as select * from tweets_json;

7. Write a query that returns the total number of tweets in table **tweets_parquet**.

   Annotate both the number of records and the amount of seconds that it took.

8. Verify that both tables (tweets_json and tweets_parquet) contain the same number of tweets.
   Which of the queries was faster?

9. Write a query that returns the total number of users with geolocation enabled from table **tweets_parquet**.

10. Write a query that returns the total number of tweets per language from table **tweets_parquet**.

11. Write a query that returns the top 10 users with more tweets published from table **tweets_parquet**.

12. Write a query that returns the top 10 users with more followers from table **tweets_parquet**.

13. Write a query that returns the top 10 more popular hashtags from table **tweets_parquet**

14. Write a query that returns the average number of words in a tweet from table **tweets_parquet**

15. Write a query that returns the max and average number of hashtags present in the tweets from table **tweets_parquet**

16. Write a query that returns the top 10 users with more mentions from table **tweets_parquet**

17. Create a managed table called **tweet_words** stored in orc format exploding the words in the tweets and normalize the words to lowercase.
    Hint: You have to use a lateral view

    Example

id	text
12345	“This a test”

id	word
12345	this
12345	is
12345	test

18. Create a managed table called **tweet_words_sentiment** stored in parquet format that returns the polarity of each word by left joining **tweet_words** with **sentiment_dictionary.** The polarity for non joining words should be neutral. You can codify the polarity as an integer in the following way:

```
case when polarity = 'positive' then 1
      when polarity = 'negative' then -1
      else 0
 end as polarity
```

Example

id	word
12345	bad
12345	wewew

id	word	polarity
12345	bad	-1
12345	wewew	0

19. Create a managed table called **tweets_sentiment** stored in parquet format as the result of a query that sums the polarity of every tweet so that

sum(polarity) > 0 -> ‘positive’
sum(polarity) < 0 -> ‘negative’
sum(polarity) = 0 -> ‘neutral’

Example

id	word	polarity
12345	bad	-1
12345	wewew	0

id	polarity
12345	‘negative’

20. Write a query that returns the hourly evolution of sentiment of tweets with a hashtag of your choice (a hashtag that it present in your tweets)

    Example

hour	positive	negative
2019062522	1233	235
2019062523	2355	124

    You can use the following macro to extract the hour**:**

```
create temporary MACRO tweet_hour(created_at string)
from_unixtime(unix_timestamp(created_at, 'EEE MMM dd HH:mm:ss ZZZZZ yyyy'),'yyyyMMddHH');
```