Sentimental Analysis of Twitter using Map Reduce

Use Case

edureka!

Version 2.0

Sentimental Analysis of Twitter using Map Reduce

Table of Contents

Sentimental Analysis of Twitter using Map Reduce

Problem Statement:

Here, we have shown the sentimental analysis of twitter data which is done with the apache flume.

Important Links:

Fetching Data from Twitter using Apache Flume:
https://www.youtube.com/watch?v=723krKpwe_k

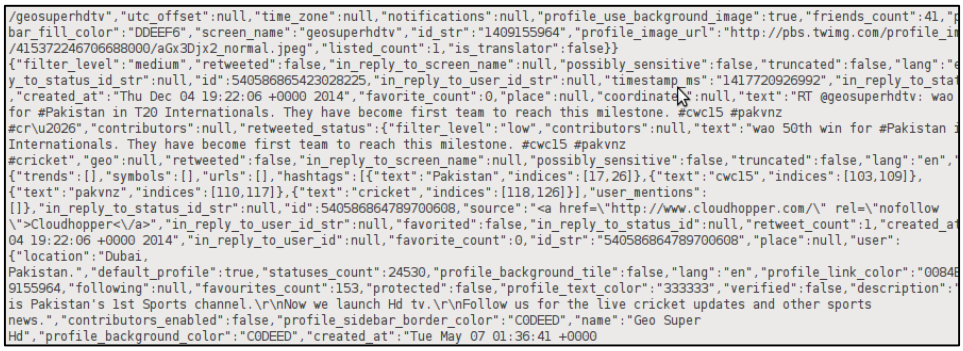
Edureka VM Installation:
Please refer to Installation guide section present in the LMS for accessing the Edureka VM Installation Guide.

Codes along with the Dataset:
https://edureka.wistia.com/medias/d7pjesxzy5/download?media_file_id=57962991

Snippet of JSON Data:
https://edureka.wistia.com/medias/w1j2cqohlq/download?media_file_id=61275117

Dataset:

Let us consider a sample dataset as in the below screenshot.



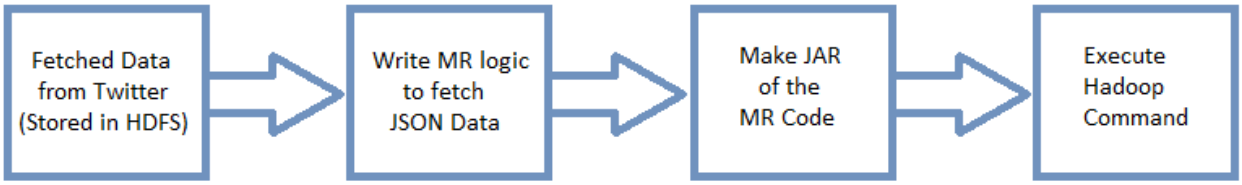
Dataset Description:

The above data represents the JSON format which is fetched from twitter. For better understanding, you can refer the snippet of JSON data which is given in previous section.

Tools and Technologies used:

- Apache Flume
- Eclipse

Dataflow Diagram:



Implementation:

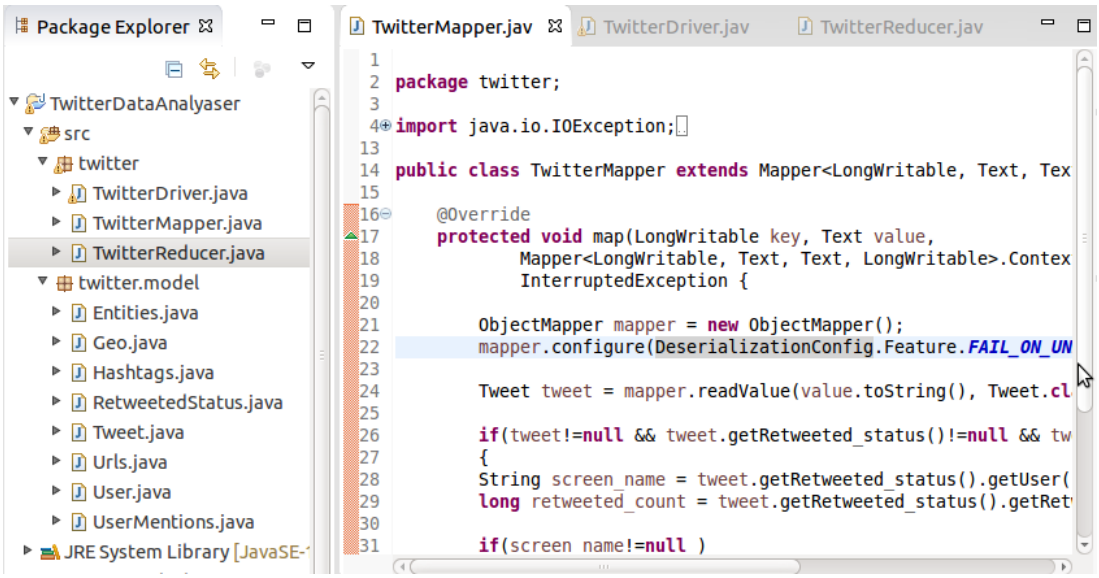
To fetch the data from twitter, first by using apache flume we store the data in HDFS which is in JSON format by default. Further to read the all the fields, we used map-reduce, JSON SerDe (Serializer/Deserializer) to read JSON data.

In this, we are storing data in hdfs from twitter by using apache flume, reading the data and after map-reduce, storing the result again in hdfs.

Let us see how to do that:

First we have fetched the data from twitter using flume and data is stored in HDFS.

Wrote a map-reduce logic to fetch JSON data.



In Map-Reduce, firstly we read the complete data line by line. After that, we have created the objects according to our requirements. In this, we have created the objects of all fields which are mentioned in JSON file.

The format of first line of flume data (JSON) file is on Page 2.

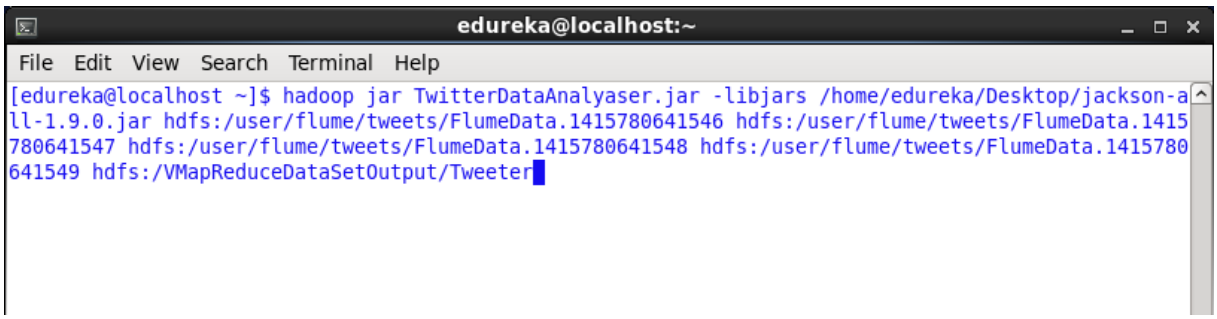
In this, we have focused in only two sections:

1. *Retweet_Count*
2. *Screen_Name, which is sub-section of User*

So, we have taken Screen_Name as Key and Retweet_Count as Value, both of these parameters is passed by mapper in the format of Key_Value pair.

Reducer receives both the parameters and put a counter for each Screen_Name and count all the Retweet_Count that is reducer output which is stored in HDFS.

Command: hadoop jar TwitterDataAnalyaser.jar -libjars /home/cloudera/Desktop/jackson-all-1.9.0.jar hdfs:/user/flume/tweets/FlumeData.1415780641546 hdfs:/user/flume/tweets/FlumeData.1415780641547 hdfs:/user/flume/tweets/FlumeData.1415780641548 hdfs:/user/flume/tweets/FlumeData.1415780641549 hdfs:/VMapReduceDataSetOutput/Tweeter;



You can check the output in the HDFS, if you are using all the data files which is given with the codes. Then the output we will be as:

```
ImFaizhr        6
SRTrendulkar    1
bhaleraosarang  1
dna     3
rahulnanda86    15
tangerine_army  1
```

We have successfully fetched the data using MR code.