## Sentiment Analysis using Hadoop Eco-System

**Source of Data** – Social Network | Spool Directories

**Data Type**: - JSON – Semi Structured Data

**Framework** – Hadoop

**Tool** – PIG | Flume

**POC** – MAP-REDUCE

1. Language – Dataflow – Pig Latin

**Data Dictionary** – AFINN

Objective:-

In this project, we will showcase how to perform Sentiment Analysis on Twitter data using Pig. To begin with, we will be collecting real-time tweets from Twitter using Flume. With the help of AFINN dictionary we can find positive sentiment from the data which we have dumped into our HDFS/FLUME. We can prioritize our own customized words in AFINN dictionary as per the need of of Client.

As this project is most advance use case of hadoop in MNC & has wide impact in sentiment analysis. We need most advance concepts of PIG & MAP-REDUCE for getting into this project.

The data from Twitter is in ‘Json’ format, so a Pig JsonLoader is required to load the data into Pig. You need to download the required jars for the JsonLoader from the below link:-

<https://drive.google.com/drive/u/0/folders/0ByJLBTmJojjzUXdEOUpVQ1haNG8>

Name of Pig JSON LOADER are:-

1. elephant-bird-hadoop-compat-4.1.jar
2. elephant-bird-pig-4.1.jar
3. json-simple-1.1.1.jar

Execution Module:-

1. Invoke PIG grunt shell in map-reduce mode.
2. Register all the PIG JSON LOADER with PIG.
3. Load AFINN dictionary from HDFS to PIG bag.

**Step – 1**

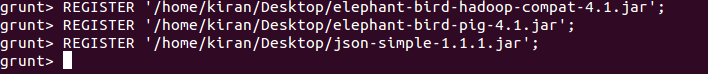
Register the downloaded jars in pig by using the below commands:

Grunt> REGISTER '/home/edureka/Desktop/elephant-bird-hadoop-compat-4.1.jar';

Grunt> REGISTER '/home/edureka/Desktop/elephant-bird-pig-4.1.jar';

Grunt> REGISTER '/home/edureka/Desktop/json-simple-1.1.1.jar';

You can refer to the below screen shot for the same.



**Note:**You need to provide the path of the jar file accordingly.

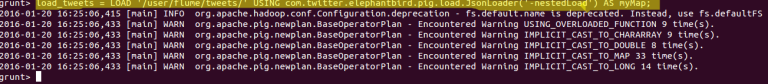
After registering the required jars, we can now write a Pig script to perform Sentiment Analysis.

**STEP-2**

The tweets are in nested Json format and consists of map data types. We need to load the tweets using JsonLoader which supports maps, so we are using **elephant bird JsonLoader**to load the tweets.

Below is the first Pig statement required to load the tweets into Pig:-

load\_tweets = LOAD '/user/flume/tweets/' USING com.twitter.elephantbird.pig.load.JsonLoader('-nestedLoad') AS myMap;



**Step-3**

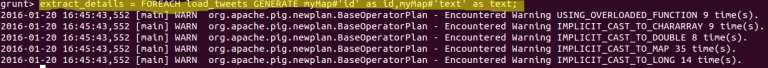
When we dump the above relation, we can see that all the tweets got loaded successfully.



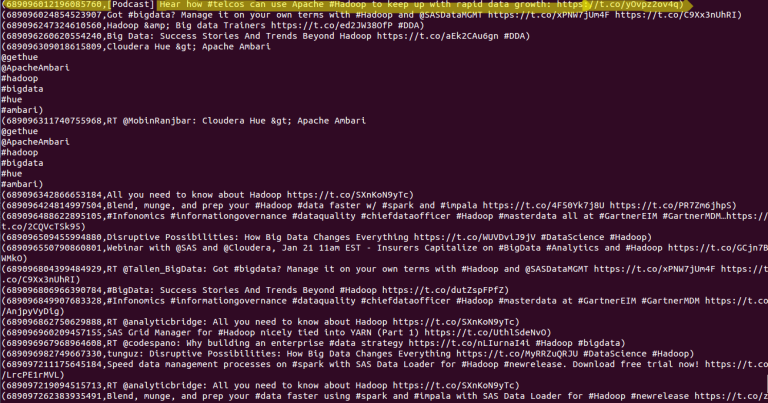
**Step -4**

Now, we shall extract the **id**and the **tweet text** from the above tweets. The Pig statement necessary to perform this is as shown below:

extract\_details = FOREACH load\_tweets GENERATE myMap#'id' as id,myMap#'text' as text;



We can see the extracted **id** and **tweet text**from the tweets in the below screen shot.

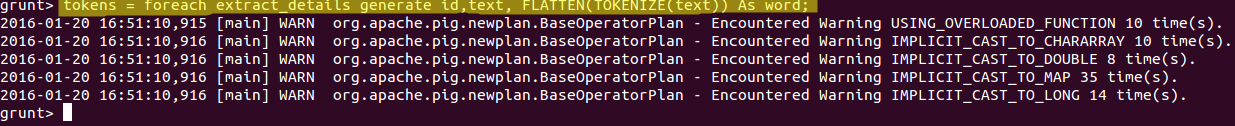


**Step -5**

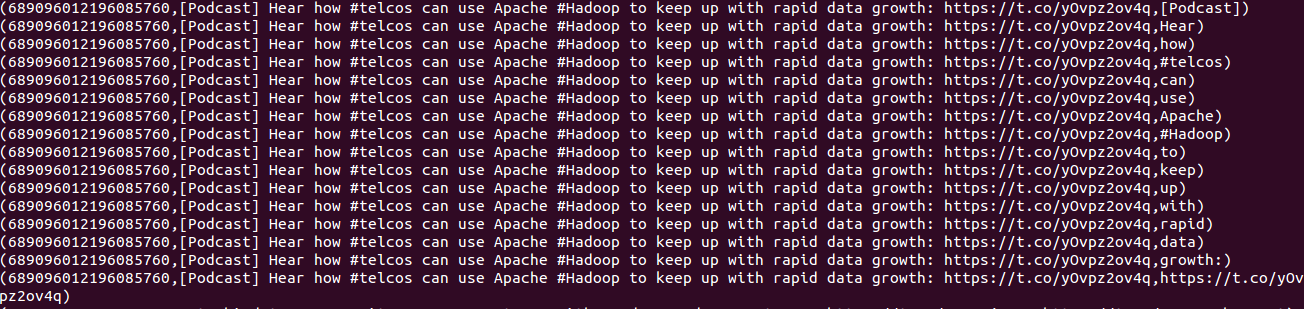
We have the tweet id and the tweet text in the relation named as **extract\_details**.

Now, we shall extract the words from the text using the TOKENIZE key word in Pig.

tokens = foreach extract\_details generate id,text, FLATTEN(TOKENIZE(text)) As word



From the below screen shot, we can see that the text got divided into words.

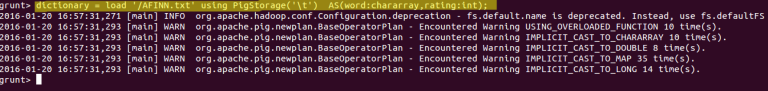


**STEP -6**

Now, we have to analyse the Sentiment for the tweet by using the words in the text. We will rate the word as per its meaning from +5 to -5 using the dictionary AFINN. The AFINN is a dictionary which consists of 2500 words which are rated from +5 to -5 depending on their meaning. You can download the dictionary from the above given link.

We will load the dictionary into pig by using the below statement:

dictionary = load '/AFINN.txt' using PigStorage('\t') AS(word:chararray,rating:int);



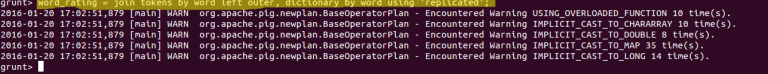
The output of this will be look like following image :-



**Step -7**

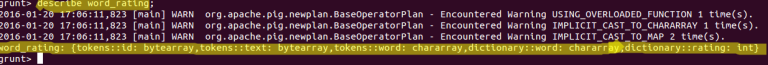
Now, let’s perform a map side join by joining the **tokens**statement and the dictionary contents using this command:

word\_rating = join tokens by word left outer, dictionary by word using 'replicated';



We can see the schema of the statement after performing join operation by using the below command:

describe word\_rating;

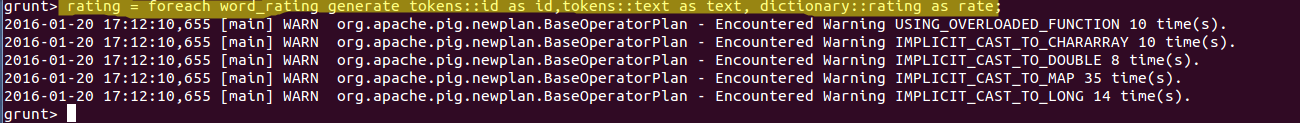


In the above screenshot, we can see that the word\_rating has joined the **tokens(**consists of id, tweet text, word**)**statement and the **dictionary**(consists of word, rating).

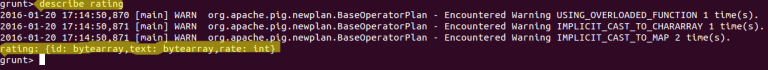
**STEP -8**

Now we will extract the **id,tweet text**and **word rating(**from the dictionary**)**by using the below relation:

rating = foreach word\_rating generate tokens::id as id,tokens::text as text, dictionary::rating as rate;



We can now see the schema of the relation **rating**by using the command describe rating.

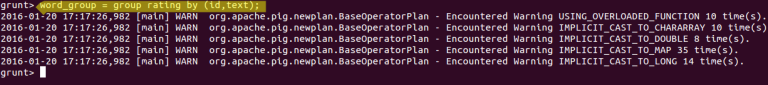


In the above screen shot we can see that our relation now consists of **id,tweet text** and **rate(**for each word**).**

**Step-9**

Now, we will group the **rating of all the words in a tweet**by using the below relation:

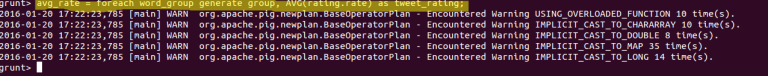
word\_group = group rating by (id,text);



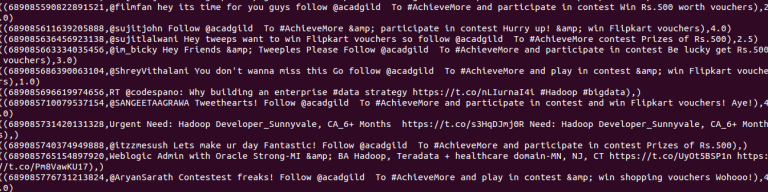
**STEP-10**

Now, let’s perform the **Average** operation on the **rating of the words per each tweet**.

avg\_rate = foreach word\_group generate group, AVG(rating.rate) as tweet\_rating;



**Now we have calculated the Average rating of the tweet using the rating of the each word.**You can refer to the below image for the same.



**STEP-11**

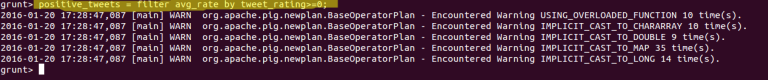
From the above relation, we will get all the tweets i.e., both positive and negative.

Here, we can classify the positive tweets by taking the rating of the tweet which can be from **0-5.**We can classify the negative tweets by taking the rating of the tweet from**-5 to -1.**

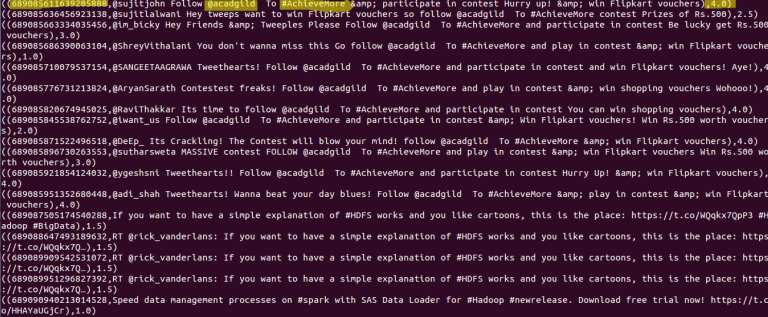
We have now successfully performed the Sentiment Analysis on Twitter data using Pig. We now have the tweets and its rating, so let’s perform an operation to filter out the positive tweets.

Now we will filter the positive tweets using the below statement:

positive\_tweets = filter avg\_rate by tweet\_rating>=0;



We can see the positive tweets and its rating in the below screen shot.



In the above screen shot we can see the tweet\_id,tweet\_text and its rating.