# Otravo assignment

## Stack

Python3, Kafka, Elastic stack

## Used Python third party module

Kafka, json, elasticsearch, tweepy, configparser, textblob, pycountry

## Workflow logic

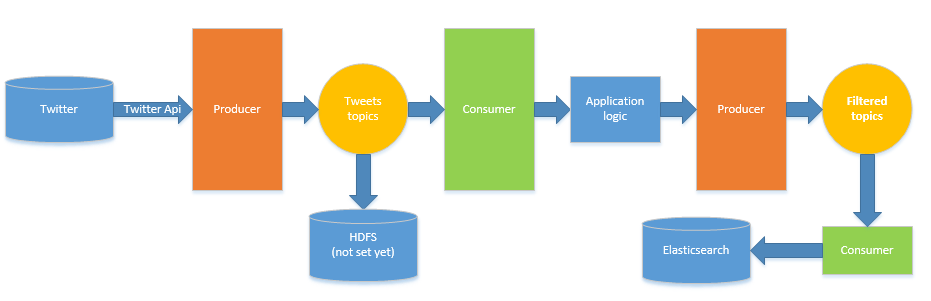


Figure 1 Data pipe workflow logic

## Code startup

1. Modify ***config.cfg*** to set up the all the configurations (You can use my settings).
2. Kafka configuration can be found in Kafka\_config.rar. This is the configuration I used for Kafka.
3. Start Zookeeper: zookeeper-server-start.bat config/zookeeper.properties
4. Start Kafka: kafka-server-start.bat config/server.properties
5. Create Kafka topics:

kafka-topics --create --zookeeper localhost:2181 --replication-factor 1 --partitions 3 --topic otravo\_twitter\_stream

kafka-topics --create --zookeeper localhost:2181 --replication-factor 1 --partitions 3 --topic otravo\_twitter\_output

1. Run **consumer\_final.py**
2. Run **my\_producer.py**
3. Run  **Kafka\_to\_es\_bulk\_final.py**

## Results

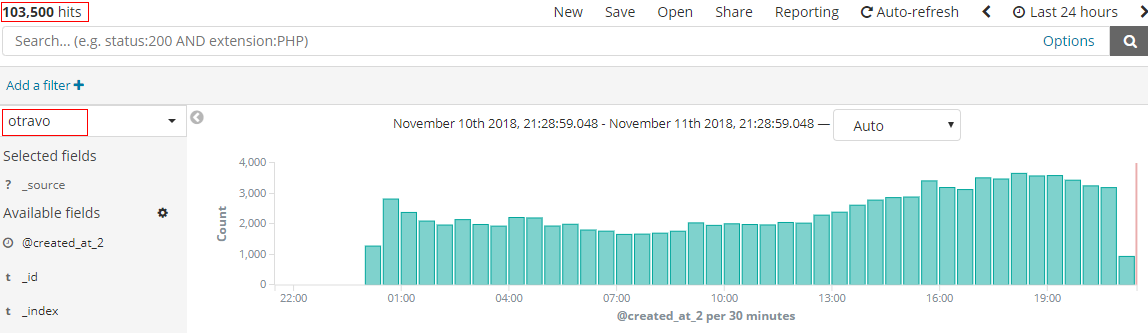


Figure 2 Tweets from 2018-11-10 21:30 to 2018-11-11 21:30

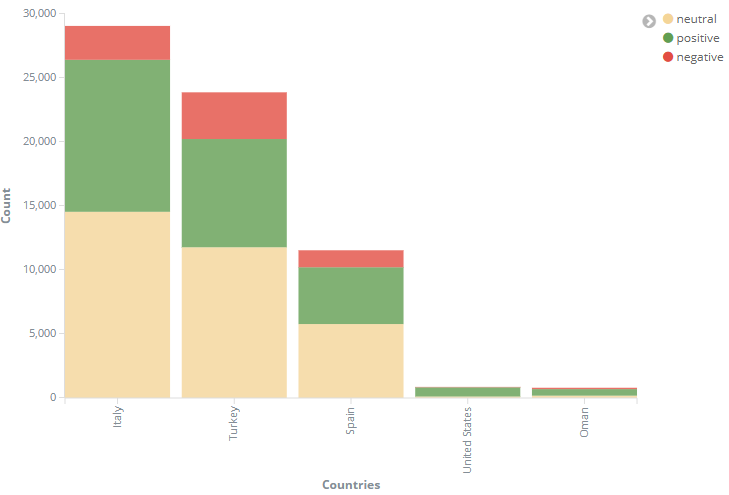
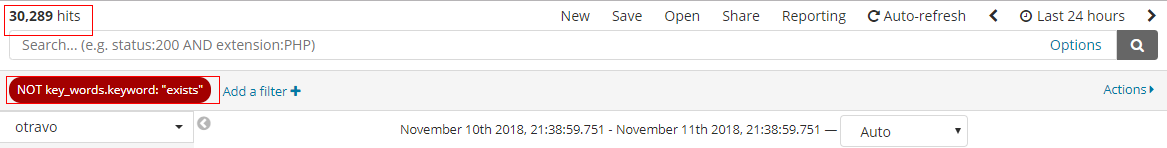


Figure 3 twitter sentiments for different countries

## Comments

* The data extraction from topics to HDFS is not set in my workflow
* There is a bug(TO\_DO) in the consumer\_final.py code:

It can only extract countries from the twitter txt, but not cities. Therefore for many records, the column ‘key\_words’ is empty.



For example, if the twitter text is ‘I love Milan’. The column ‘key\_words’ will be empty. But it’s supposed to be ‘Italy’.

This bug can be fixed by:

* 1. Using multiple producers and partitions. Assign only one country name to each producer and set partition name. Therefore each partition will only contain records for one country.
  2. Improve the logic in TO\_DO part. Let it map between cities and countries.