Twitter Sentiment Analyzer using tweepy (Twitter API), Elasticsearch and Kibana. This Sentiment Analyzer is developed using Python language.

Tools required:

1. Python 2.7 and above
2. Tweepy API Python
3. Elasticsearch and Kibana installed
4. Pandas
5. Python libraries(pandas, pyelasticsearch/Elasticsearch API )

Important files/folders:

1. twitter\_streaming\_api.py

This python file is used to stream live tweets from twitter. It uses tweepy API, which brings tweets for a given topic. The topic can be entered as a command line argument.

Run: python twitter\_streaming\_api.py <Topic>

Example: python twitter\_streaming\_api.py NetNeutrality

The above command will collect live tweets for a given topic in a file ‘tweet\_collection.txt’. The destination file can be modified. We can collect tweets for a given period of time (say 1-2 days) and then we will have enough tweets for testing our sentiment analyzer.

1. es\_indexer.py

This file is used to create an index named ‘sentiment\_analysis’ in the Elasticsearch with the given fields. In this case, I have used two fields: 1) text 2) sentiment

In order to index in Elasticsearch, its instance need to be started. Following step must be carried out:

First Set the JRE path: set JAVA\_HOME=C:\Program Files\Java\jre7

From elasticsearch-1.4.4\elasticsearch-1.4.4\bin (may vary) path <run Elasticsearch.bat > . This will start the ES instance.

Checking ES instance:

<http://localhost:9200/>

checking Marvel and Sense:

<http://localhost:9200/_plugin/marvel/sense/>

Initializing Kibana:

From the path where kibana is installed , run kibana.bat

<http://localhost:5601>

1. preprocess\_tweet.py

This file is used for the twitter preprocessing. Once we have a large dataset of tweets (for training purpose), we need to preprocess it so that the tweets is simplified and can be used as a training set for a ML classifier.

1. Sentiment\_analyzer\_draft.py

In this file we give a full training set of about 21k tweets to the Naïve Bayes classifier to train the model. We also input feature list. I ran this program once to train the model and then stored the trained NB as pickle objects. I then commented those lines which were used to train the NB and just kept pickle file to train NB classifier. We can give a test tweet to evaluate the classifier. Though I do not use this Python file any more while using SA analyzer but it can be used to train the ML classifier.

1. Sentiment\_analyzer\_elasticsearch.py

This is the main python file that I am using for classifying the tweets using NB classifier and then storing the result in the Elasticsearch. I store two fields in the Elasticsearch namely the tweet and the associated sentiment. Then using Kibana as the graphical tool to plot the graphs showing different sentiments about a topic from the collected tweets. We read the collected tweet file and then using Pandas, I pick up the text field (the tweet) as they are Json format data, so I use Pandas for picking up the text field. We pass it on to Naïve bayes and then once we have the sentiment, we index it to Elasticsearch.

After the tweets and associated sentiments are indexed in ES, we can use Kibana to plot the graph and visualize the overall sentiments about a given topic.