ETL Project Report

By: Siddhant Desai and Alexandre Geraldo

For our ETL project, we wanted to extract data that would help us answer one question: Does the volume of tweets from Trump correlate to a higher volume of articles from news websites. Below we will describe the three steps of data collection that we performed.

**Extract:** The first data source we used was a CSV file of trump’s tweets from 07/16/2015 to 11/11/2016. This dataset was found on the website Kaggle.com.

We then looked at the date range within the data to find the X days on which trump was the most and least active, where X is a user defined variable for the number of days we want to extract. If necessary, our code can also extract all the articles in the date range.

We pulled the articles from two news sources, New York Times and The Guardian. In both cases, we wrote functions that queried the API using the date as a parameter. Both APIs had rate limits of 5000 queries a day and 10 articles per query. Therefore, to gather all articles in a date range, we had to paginate through the article results using a while loop. We used a while loop to allow for the code to break when there were no more results. We decided on a while loop to allow for variability in result size since a for loop has a defined number of runs.

**Transform:** The main part of our transformation was about getting the data in a JSON format so that we could load it into Elasticsearch/MongoDB. For the csv, we looped through each row and transformed it into a dictionary with each column being a variable.

The APIs, luckily, output in a json format already, which is the format needed to load into the database. Therefore, for The Guardian, we simply took the json output of the queries.

For NYT, we found that the json included a large amount of data for all the multimedia (images, videos) from the article. Since this was not pertinent to our needs, we removed this data. To do this, we took each json object and moved all the info except the multimedia into a new dictionary. Some results lacked certain fields that other results had, so we had to set up a series of if statements for each field that allowed for the field to be absent.

**Load:** We loaded our data into two databases. To begin with, we loaded it into mongoDB. After this, for an additional challenge, we also wrote programs that would follow the above steps, but load the data into ElasticSearch instead of mongoDB.

We chose noSQL databases because our data was primarily in a JSON format, which lended itself easily to a noSQL database. The noSQL databases would also allow us quicker access to the data due to its non-relational structure. ElasticSearch was chosen due to its search capabilities, which are more robust than mongoDB.

Our database name was TheDonald. The collections were named tweets, NYT and theguardian. This was the extent of the database. On our github, you will find the code to ETL into mongoDB and separate code for loading into ElasticSearch. In the resource folder, there is sample data collected for one day using our code.