Wrangle Report

Wrangle and Analyze Data: Disha Mukherjee

This Wrangle and Analyze Data Project is a component of Udacity's Data Analyst Nanodegree. The project involves haggle of information from varied sources related to tweets from the Twitter user @dog_rates, conjointly called WeRateDogs. WeRateDogs rate's photos of people's dogs during a comic manner, most frequently giving ratings beyond 10/10. With of scraping data, quality and tidiness problems were assessed then cleansed. Finally, 3 visualizations were created, and insights may be found within the act_report.pdf document.

GATHERING DATA

Data was gathered from three totally different sources:

- The enhanced_twitter_archive file was provided and downloaded manually. This file includes varied variables for every tweet as well as tweet id, timestamp, text, rating, name, etc.
  Additional data that were favorite count and retweet count gathered using the Twitter API.
- The image_predictions.tsv and tweet_json.txt file was downloaded programmatically using the Requests library from Udacity's servers. Through machine learning techniques, the breed of dog was foreseen supported the image. Following instruction from Udacity project detail page initial which I have downloaded from Twitter archive file named 'twitter_archive_enhanced.csv' that was provided for transfer and second file 'image_predictions.tsv' transfer programmatically from Udacity server using request library and third file tweet_json.txt file using Twitter API for @WeRateDogs as I requested for permission from Twitter for pretty much a week ago so that I can scan this file line by line to form Dataframe for a minimum of 3 columns id(tweet) , favorite and retweet counts.

ASSESSING DATA

After the data was gathered, assessment was performed through the subsequent methods:

- .head()
- sample()
- .info()
- .value_counts()

Tidiness issues that were cleaned:

twitter1 dataframe

- twitter1 variable (dog stage) in 4 different columns (doggo, floofer, pupper, and puppo)

twitter2 dataframe

- twitter2 data should be combined with the twitter1 data since they are information about the same tweet

images dataframe

- images data could be combined with the twitter1 data as it has all information about same tweet

Quality issues that were cleaned:

- Keep original ratings (no retweets) that have images
- Delete columns that won't be used for analysis
- Erroneous datatypes (doggo, floofer, pupper & puppo columns)
- Separate timestamp into day/month/year (3 columns)
- Correct numerators with decimals
- Correct denominators other than 10:
- a. Manually
- b. Programatically
- Data contains retweets (ie. rows where retweeted_status_id and retweeted_status_user_id have a number instead of NaN)
- 'tweet_id' is an integer
- 'timestamp' and 'retweeted_status_timestamp' are currently of type 'object'
- 'source' is in HTML format with a and \a tags surrounding the text
- 'name' has values that are the string "None" instead of NaN
- Looking programmatically, some names are inaccurate such as "a", "an", "the", "very", "by", etc. Looking visually in Excel, I was able to find more names that are inaccurate including "actually", "quite", "unacceptable", "mad", "not" and "old. It seems like the method used to extract the names was using the word the followed "This is..." and "Here is..." which leads to some inaccuracies.
- An instance where name being "O" instead of "O'Malley"
- doggo, floofer, pupper, and puppo have values that are the string "None" instead of NaN

- Upon visual inspection in Excel, there are ratings that are incorrect. It could be confusing to interpret unstandardized ratings.
- There are many columns in this dataframe making it hard to read, and some will not be needed for analysis

CLEANING DATA

The issues found throughout the assessment method were cleansed and tested through the subsequent ways and techniques:

- merge()
- reduce()
- .extract()
- .drop()
- .isnan
- .astype()
- to_datetime()
- islower()
- .replace()
- .rename()
- set_option()
- .loc[]
- .value_counts()
- .info()
- .head()

CONCLUSION

As Matt aforesaid throughout the course lectures that rarely it happens that all the data, we want for a project come from 1 source and is already tidy. This project emphasised that you simply would wish to go through the Python and its varied libraries to scrape data from varied sources in varied formats, and clean varied quality and tidiness problems, before any data analysis may be performed. I feel solely once learning completely from Udacity platform I even have begun to grasp these tools and techniques.