WRANGLE REPORT: WeRateDogs

Udacity DAND: Wrangle and Analyze Data Project

INTRODUCTION:

This Wrangle and Analyze Data Project is part of Udacity's Data Analyst Nanodegree Term 2. The project involves wrangling of data from various sources associated with tweets from the Twitter user @dog_rates, also known as WeRateDogs.

This report briefly describes my wrangling efforts. The tasks of this project are as follows:

- Gathering data
- Assessing data
- Cleaning data

GATHERING:

The data for this project consist on three different dataset that were obtained as following:

- Twitter archive file: the twitter_archive_enhanced.csv was provided by Udacity and downloaded manually.
- The tweet image predictions, i.e., what breed of is present in each tweet according to a neural network. This file (image_predictions.tsv) is hosted on Udacity's servers and was downloaded programmatically using the Requests library and URL information
- Twitter API & JSON: by using the tweet IDs in the WeRateDogs Twitter archive, I queried the Twitter API for each tweet's JSON data using Python's Tweepy library and stored each tweet's entire set of JSON data in a file called tweet_json.txt file. I read this .txt file line by line into a pandas dataframe with tweet ID, favorite count, retweet count, followers count, friends count, source, retweeted status and url.

ASSESSING:

Assesing dataframe visually and programmatically and found lots of Quality and Tidiness issue but done minimum requirement for project 8 quality and 2 tidiness issue as below for cleaning.

Quality Issues:

It mainly include issues:

1. Completeness
2. Validity
3. Accuracy
4. Consistency.

Issues in dataset are as follows:

1. Removing Retweets from the dataset
2. Converting tweet_id and name into proper datatypes as string.
3. Removing columns that are not necessary for the process:'in_reply_to_status_id'and 'in_reply_to_user_id'
4. Timestamp should be datetime instead of object(string).
5. Converting tweet_id and name into proper datatypes as string.
6. Converting source and dog_stage as categorical data types for analysis.
7. Some ratings with decimals such as 13.5/10, 9.5/10 have been incorrectly exported as 5/10 (in addition to other numbers with decimals such as 11.26 and many are present as seen visually as well as programmaticaaly, Further added an additional column of final_rating.
8. Drop duplicate jpg_url present in the dataset
9. name has values that are the string "None" instead of NaN.Some names are inaccurate such as "a", "an", "the", "very", "by". Looking visually, I was able to find more names that are inaccurate including "actually", "quite", "unacceptable", "mad", "not" and "old.
10. Converting p1,p2,p3 into lowercase as we can see some unnecessary Captiliased forms of data in it.

Tidiness:

Tidy data is a standard way of mapping the meaning of a dataset to its structure. A dataset is messy or tidy depending on how rows, columns and tables are matched up with observations, variables and types. In tidy data:

1. Each variable forms a column.

2. Each observation forms a row.

3. Each type of observational unit forms a table.

1. Merge all dataframes: tweet_json_clean and image_clean data should be combined with the twitter_clean data since they are information about the same tweet.

2. Convert the dog stage or category into one column instead of the multiple columns.

CLEANING:

The issues found during the assessment process were cleaned and tested using the following methods and techniques. The (define, code, and test) steps were used in the cleaning process. First, copies of the DataFrames were created before cleaning. Then, the steps of cleaning

were applied iteratively on all issues. Further I have used merge(), info(), astype() and other methods to make my data clean and tidy.

CONCLUSION:

Rarely does all the data we want for a project come from one source and is already tidy. This project emphasized that one will need to using Python and its various libraries to scrape data from various sources in various formats, and clean various quality and tidiness issues, before any data analysis can be performed. Through Data Wrangling I got familiar to deal with the data easier for visualizations which is efficiently possible only through programmatic coding.

By: Amisha Rastogi