DATA WRANGLING REPORT

Project Goal:

The goal of this project is to effectively wrangle data related related to dog ratings. The data is sourced from the twitter user @WeRateDogs. Once we have effectively gathered, assessed, and cleaned our data in this project, it can be used for our analysis.

This report briefly escribes my wrangling effort.

Project Details:

The tasks of this project are as follows:

Gathering data Assessing Data Cleanig Data

Gathering Data

The data used for this project consisted of three different datasets that were obtained as following:

Twitter archive file: This data was provided in the project guideline. I downloaded it to my workspace by clicking on the jupyer icon then upload. I imported the python pandas library as pd and used the pandas read_csv() function to read the file into a dataframe named twitter_archive.

Tweet image prediction file: I imported the Python requests and os libraries. With the get() function of the requests library, I got the data through its url and saved it in a response variable.

Using the Python with open function, I wrote the response's content to a tsv file in the same working directory. I then read the downloaded tsv file into a dataframe named image_prediction.

Tweet_Json text: I created a twitter developer account and created an application for the project. I used the app credentials (consumer_key, consumer_secret, access_toke, and access_secret) for the twitter API authentication. I imported tweepy and json, authenticated tweepy.OAuthHandler and set wait_on_limit to True in the API parameter in order to wait after tweet limit but no luck so I just downloaded tweet json from the site

With the Python with open function, I created the tweet_json.txt and wrote the output to it, I appended failed ones to the empty dictionary created above. I printed the time taken and the failed dictionary.

With the Python with open function again and a for loop, I read the tweet_json.txt line by line and loaded each line as json file. I saved each tweet_id, retweet_count, favorite_count, followers_count and friends_count which I later converted to a dataframe named tweet_json.

Assessing Data

Once the three tables were obtained, I assessed the data as following:

Visually: I printed the three different dataframes individually in a jupiter notebook and scrolled through left and righ, up and down. Secondly, I visually assessed the csv files in Excel spreadsheet.

Programmatically: I did various programmatic assessment with various python and pandas methods and functions such as .info(), .describe(), .isnull(), .head(). sample(), .duplicated(), .value_counts() and shape.

Cleaning Data

This part of the data wrangling process was divided into three parts: Define, Code and Test.

These three steps were each on the issues stated in the assess section.

First, I made a copy of the original three datasets.

Twitter_archive = df1_clean Image_predictions = df2_clean Tweet_json = df3_clean

Then, I followed the Define, Code amd Test process and made the following cleaning efforts:

- I dropped retweeted_status_id, retweeted_status_user_id, retweeted_status_timestamp, in_reply_to_status_id and in_reply_to_user_id columns because they have over 90% of missing values each.
- And also I used the code is null to make sure all RT are deleted
- I combined the four dog stages spread across four columns into one single column.
- I dropped followers_count and friends_count columns as they don't contain necessary values that would be relevant to the analysis.
- I converted the timestamp column from an int to datetime.
- I converted the tweet_id column from integer to string.
- the types of dogs in columns p1, p2, and p3 had some uppercase and lowercase letters I changed them all to lowercase.

- I dropped all values in the name column that started with small letters because it was confirmed that those names weren't dog names.
- I converted the tweet_id column in image prediction table to a string.
- I converted tweet_id coulmn in the tweet_json dataframe from integer to string.
- I changed the column label from 'id' to 'tweet_id' in tweet_json(df3) dataset.
- I merged the three dataframes to become one dataframe and merge them on tweet_id column.

Storing the Data

After gathering, assessing and cleaning the data, I saved the merged data in a csv file named twitter_archive_master.csv.

Conclusion

I did a lot of mistakes in this project because I was frustrated and I really regret it and I was doing this now for almost 5 hours and it was fun and I hope I learn a lot from this mistakes.