Udacity: WeRateDogs Wrangle Analysis

WeRateDogs is known as a twitter handle that rates dog pictures that have been posted on twitter.

In accordance with the fulfilment of this project I will be detailing and describing the steps I took to gather and clean the data I used for the project.

Gathering of Data

A csv file was provided by Udacity which was basically a twitter archive csv file which contained reviews made by WeRateDogs twitter handle all arranged using tweet_ids

Another of the files which contained predicted images of tweets using neural networks. This tsv file was hosted on the Udacity Server and was gathered using requests.

To fetch the retweet and favorite count, we were asked to use Tweepy API. I was not given access to Twitter API but it was to return a JSON object which was given to us by the instructor in one of the Sunday evening sessions.

All these data was read into three different dataframes for assessment and cleaning

Assessing and Cleaning

I did both the visual and programmatic assessment as detailed in my ipynb file. Both the Quality and Tidiness assessments were first recorded before being cleaned.

For sake of writing this in a rush due to work, I have decided to list and write how I cleaned together.

Quality Issues:

- Tweet_id columns in all the dataframes are in integer and should be in string instead as it is not involved in calculations. I used the .astype(str) to set them all to string object datatype
- Timestamp column of the twitter archive provided by Udacity was in string of which I used .apply(pd.to_datetime) to convert it to datetime
- Redundant Data in some data columns this was filtered and removed so as to maintain uniformity.
- In the dataframe containing data from Twitter API the displayed_text_range contains two variables which was separated using lambda and finally dropping the display_text_range column
- Also noticed in the visual assessment was an issue that the source column was in HTML which was cleaned using regex to extract just the source.
- Predictions in the predicted images data gotten from Udacity servers were not all uniform as some started in capital letters while others were in lower case letters I used the .str.lower() function to convert all to lower case
- Also during the visual assessment it was noticed that instead of whitespace there was a use of underscore character. This was replaced using .replace() function
- The Predictions on the Predicted Images dataframe was changed to Capital letters in the beginning

Tidiness Issues:

- Dog types were not classified together under a single column but instead had their own different columns which would not enable proper analysis of the data. I used the melt function to melt them together while removing the ensuing duplicates to ensure data reliability.
- All tables were merged together as all were related to each other using tweet_id
- Merging of the Predictions columns into Breed column with highest confidence level of all predictions

Further Cleaning

- Tweets that did not have an ensuing image were removed for better analysis
- Wrong names such as ‘a’, ‘just’ and the likes were all changed to none so as to enable uniformity. This was done as it was noticed through the unique() function that all or most of the wrong names were in lower case.

After all these, lone standing data frame was stored in a master dataset and exported as a csv file