Data Wrangling and Analysis

The data wrangling and analysis was performed on a twitter account called WeRateDogs. This account rates people's dogs with a humorous comment. Data was gathered, assessed, stored and visualized.

Gathering

In total three datasets were used in this project. The first dataset was the twitter archive which was retrieved by downloading it from a link provided by udacity, this file was saved to my local machine and then called into the jupyter noteboook and called twitter_archive.

The second dataset was an image pridictions file which is hosted on Udacity's servers and was downloaded programmatically using the Requests library and a URL. This was called img when loaded into jupyter.

The third dataset was retrieved via the Twitter API by using the tweet id's in the twitter archive. Each tweets JSON data was pulled using Python's Tweepy library and stored in a text file. This text file was read into a pandas dataframe and the final variables saved were tweet ids, favorite count, retweets and time the tweet was created. This dataset was called tweet_info.

Assessing

Data was assessed visually and programmatically. It was found that the the datatypes were incorrect for tweet_id in the twitter archiv edataset as well as the image predictions dataset, which was recorded as an interger and timestamp which was recorded as a string.

There were many names that were recorded incorrectly as non-names such as 'a', 'the', 'an' ect. Some of these rows did have the actual dogs' names in the text column. All of these cases were recorded as lowercase words. There were 109

rows with names in name column that are lowercase(not dog names), 22 of these rows contain the dogs' name in the text column and there were 745 rows that contain 'None' in the name column, 8 of these rows contain the dogs' name in the text column

Some tweets recorded were not tweets actually relating to dogs and dog ratings and some tweets posted were not pertaining to actual dogs at all, but to other animals or even people(snoop dog).

There were cases were some ratings were unusually high (in the 60s, 70s, 80s, some even over 100), this was the case with the rating denominaters too. It was found that in some of these instances the rating were actually just recorded incorrectly and that the real rating was in the text column.

There were unusual characters after '&' in the text column: 'amp:' and there was a large number of the tweets that were actually retweets.

Dog Stages have values as columns, instead of one column called 'stage' that identifies dogs in the catagories puppo, pupper, doggo or floofer.

The twitter archive dataset had 2356 rows and 17, the image predictions dataset had 2075 rows and 12 columns and the twitter_info dataset had 2648 rows and 4 columns.

There were columns that was not needed for our analysis of teh data.

Cleaning

Firstly, each dataset had to be copied so that these copies could be worked with.

Unnecessary columns were dropped in the datasets using .drop() adn indexing these given columns.

The column widths were expanded in order ro be able to properly read the text in the text column.

In order to deal with the rows that did not have the correct dog names in the names column or that had 'None' but had the names in the text column, a for loop was

used to itterate through rows where the name is lowercase and 'named', 'name is' appears in text column and then also where the name was recorded as "None' and 'named' or 'name is' appeared in the text and then replace the inconsistencies with the name that is mentioned in the text. After this was done there were still 22 rows that had lowercase names and these rows were also delt with with a for loop and replaced the lowercase names with 'None'.

Another visual assessment was done on the names that were recorded as non but this time we looked at the rows that included 'None' in the name column and the words 'this is' in the text column. The names that were found in the text column were then corrected manually in the name column.

192 rows with retweets were removed using .drop()

The inconcistencies with the numerators and denominators were delt with by visually and programmatically locating the ratings thst were recorded incorrectly and then recording them properly in the rating_numerator and rating_denominator columns.

Next, tweets with decimals in text column were located and viewed using regex and then the correct numerator was recorded for these tweets.

The incorrectly recorded datatypes were changes to the appropriate datatypes by using .astype()

The unuasul characters in the text column were located and then replaced with the appropriate symbol by using str.replace('&', '&').

The dog stage column was delt with by using .str.extract to extract the variables that were in columns and put them in a new column called dog_stage that then matched the relevant tweets. The columns doggo, puppo, pupper and floofer were dropped.

In the image dataset the columns p1, p2, and p3 were not all neede, A fuction was created to choose the best predictions and incorrect prediction were replaced with 'NaN' and these rows were dropped. The columns that were not needed were dropped.

After the cleanon gof the datasets we were left with 2161 rows and 9 columns for

twitter_archive, 1751 rows and 5 columns for the image dataset and 2648 rows and 4 columns for tweet_info.

The three datasets were then merged. However, after the merge we were left with duplicates and the duplicates had to be dropped. After th emerge we were left with a dataset with 1671 rows 12 columns for analysis.

I have some questions for the reviewer.

I had initially converted tweet_id from int to str before i continued with the step to find and replace names that were in the text column. However these changes were not executed when run. The changes were only applied if i did not change the datatypes before this step. Why is this? I was stuck on this particular section for a very long time before i realized what was causing the code not to apply the changes i wanted.

Why were there duplicates after i merged the datasets?

How can i go about dropping multiple ids at once instead of having to do them one by one?

Why is it that i could only drop rows by referencing the text instead of the ids after the merge?

Thanks in advance.