Report on Wrangling

The data wrangling was performed on 2 files:

1. twitter-archive-enhanced.csv: downloaded from the resource directory of the project
2. image_predictions.tsv: file read using requests library

–> twitter-archive-enhanced.csv

df name: twitter_data

Details of analysis of each column is as follows:

The following columns 1. in_reply_to_status_id 2. in_reply_to_user_id 3. retweeted_status_id 4. retweeted_status_user_id

We dont need the these columns as we don't want tweets which are retweets or replies to another tweet. Just delete the rows which have values in these columns and then finally drop these columns.

timestamp

- The column should have data in datetime datatype instead of string. (I have noticed that although we convert it to timestamp, it still get converted to string while saving to CSV file. We have to manually indicate that this column contains datetime while reading the csv)
- Even the +0000 is not necessary as its same in all the rows. Hence its better to remove it.

source

- The data is source is wrapped in anchor tags which is not necessary. We can keep only the text which is inside the anchor tags.
- After checking for unique values I decided that its better to convert it into category as it contains only 4 options.

text

- The data had image urls too which was not necessary as its same as in the expanded_urls column. I used regex to identity the urls in the column and used the str.replace() method to remove it.
- While performing visual analysis I also saw that there were many \n at the end of the text. I decided to remove it was it is obvious that the end of the string will be considered as new line. I used string slicing to remove it.
- Another problem which I noticed during visual analysis was that the text column had interpreted & as &. So I again used str.replace() method to replace &amp with **and**. Its better to use **and** instead of **&**.

expanded_urls

- Few expanded_urls had NaN values. I fetched data from the tweets_info and saw that even they didn't have the expanded_urls in them. So I visited few tweets and noticed that they didn't any images. Hence, I decided to remove rows with NaN values in expanded_urls column as our primary focus is to analyse the tweets with dogs images in them.
- While performing visual analysis I saw that some of them were link of www.vine.co which is actually a link to short videos. As we need images only, I decided to delete tweets which contained only the vines link and no image links.
- The urls also were repeated so I decided to delete duplicate urls. I first splitted the data using comma and used the set method to keep unique values only and then converted it to string again by joining it using comma.

rating_numerator and rating_denominator

- Tweet with rating_numerator as 1 and rating_denominator 2 was actually miscalculated and fixed with the original data of 9 and 10 respectively.
- Tweets with rating of 1 were not for dogs so they were deleted.
- Some rating were in float so I decided to shift the denominator. Its better than converting the entire column to float. Ex: 9.5/10 was converted to 95/100
- **rating** column was creating by dividing rating_numerator by rating_denominator to create a generalized rating.

name

- The following names were replaced with None as they didn't have any name for dog: "by", "infuriating", "getting", "unacceptable", "this", "all", "just", "an".
- Names such as 'None' and 'a' were found using value_counts() and other quality issues were found using visual analysis.
- Few names with None and a had dog names after the keyword 'named', 'name is' and 'this is'. So they were fixed.
- If still no name was found then it was replaced with None.

dog_stage and 4 columns of dog stage(doggo, floofer, puppo, pupper)

- Few rows had data in 2 out of 4 colums of dog stage. For example in doggo and floofer. Such data's text column was viewed and its original dog stage was fixed.
- A single column named **dog_stage** was created and it was converted into category.
- Few values as none actually had the dog stage mentioned in the text column so it was extracted using regex.

–> image_predictions.tsv

df name: image_pred

- Tweets with video thumbnails were deleted. A regex pattern was used to identify them.
- Columns from p2 didn't have valid recognition of dogs. Only p1, p1_conf and p1_dog had correctly labelled dogs and identified them so columns from p2 to p3_dog were deleted.