Wrangling Report

Gathering

First step is data gathering:

- 'twitter-archive-enhanced.csv' was downloaded manually and stored to a DataFrame called 'archive'.
- 'image-predictions.tsv' was downloaded programmatically using the requests library from the provided link and was stored to a DataFrame 'image'.

- Additional data was gathered by querying Twitter API using Tweepy based on the tweet IDs found in the twitter-archive-enhanced.csv. The info was written line by line using 'json.dumps' into tweet_json.txt. Then stored the file content in 'data_json' DataFrame.

Assessing

Then I moved to Assessing steps, where I assessed visually and programmatically for quality and tidiness issues.

Below some of the quality issues found:

Tweets without images, and tweets that are actually retweets. Also I found replies to original Tweet for the same user ID @weratedogs.

Rating numerator and Rating denominator are not always accurate as some tweets included more than one rating, and other tweets had decimal numerators but they were wrongly identified. Also the rating was not unified numerator should always be 10, while in some cases where we had more than one dog in the image the denominator was a multiple of 10. Tweet that don't have any rating.

Some duplicate tweet IDs found in image DataFrame due to retweets.

We had also erroneous names including values like 'a, an, the, his, incredibly…' that should be fixed. Also some names were missed from tweet texts.

We had 'None' Values that should be replaced with 'NaN', e.g in Name column.

Some columns types should be fixed.

Below some of Tidiness issues found:

Some columns can be removed like rating numerator and denominator and be replaced by one column.

doggo,floofer,pupper and puppo columns can be replaced by one column

Breed’s prediction can be added into one column.

One final Table is needed.

I preferred in some cases to Assess and clean (Define, Code and test) directly case by case. Before cleaning I took copies for the three DataFrame.

Cleaning

First I decided to create 2 new columns ‘breed’ and ‘prediction_confidence’ from the columns generated by the neural network Alogrithm in the image_clean table.

Then I merged `archive_clean` with the `image_clean` with right_join to keep only tweet IDs found in `image_clean` get rid of the tweets without images.

Removed retweets from archive_clean by removing rows with retweeted_status_id not Null.

Dropped retweeted_status_id, retweeted_status_user_id and retweeted_status_timestamp columns from image_clean

Then merged 'retweet_count',
'favorite_count'from data_json_clean to archive_clean

Kept only columns where in_reply_to_status_id and in_reply_to_user_id null and dropped columns in_reply_to_status_id,in_reply_to_user_id in archive_clean.

I created a dog_stage column out of doggo, pupper, floofer and puppo columns and in archive_clean_t replaced 'None' dog_stage column values with NaN and dropped the below columns:

- doggo
- floofer
- pupper
- puppo

And used regular expression that matches decimal numerators to get the float points and used modulo to check denominator if it is not a multiple of 10 so in order not to use it. And created a rating column that is unified numerator over 10.

And then used a set of Regular expressions to extract dogs' names from tweets' text column.

Also used replace function to replace 'None' values in *name* column with NaN.

I dropped the following columns, from archive_clean:

- jpg_url
- img_num

Finally

- Converted 'timestamp' column from string to datetime using pandas pd_datetime functions
- Converted 'breed' column from string to category
- Converted 'dog_stage' column from string to category