Data Wrangling Report

1. Gathering Data

Dataset(s)

The dataset I'll be wrangling is the tweet archive of Twitter user @dog_rates (https://twitter.com/dog_rates), also known as WeRateDogs. This archive/dataset consists of 2356 basic tweet data from November, 2015 to August, 2017. WeRateDogs is a Twitter account that rates people's dogs with a humorous comment about the dog.

Based on the images in the above dataset (i.e. WeRateDogs Twitter archive), another dataset is created which consists of image predictions (the top three only) alongside each tweet ID, image URL, and the image number that corresponded to the most confident prediction (numbered 1 to 4 since tweets can have up to four images). Though no wrangling will be done directly on this image predictions dataset, it will definitely provide some additional data for our main tweet archive dataset

.Gather Twitter archive CSV file

Using the link provided by Udacity, I downloaded the WeRateDogs Twitter archive manually as twitter_archive_enhanced.csv (https://d17h27t6h515a5.cloudfront.net/topher/2017/August/59a4e958_twitter-archiveenhanced/twitter-archiveenhanced.csv) file and imported this file into a dataframe (arc_df).

Gather tweet image predictions

I downloaded the tweet image predictions file hosted on Udacity's servers programmatically using Python's Requests library and saved it locally to image_predictions.tsv file. Then, I imported this file into a Python Pandas dataframe (img_df).

Gather data from Twitter API

Using the tweet IDs in the Twitter archive, I accessed the entire data for every tweet from Twitter API and stored every tweet's entire set of JSON data in a file called tweet_json.txt file. Created a dataframe status_df from this JSON including only tweet_id, retweet_count, favorite_count and display_text_range data

2. Assessing Data

I opened the twitter_archive_enhanced.csv and image_predictions.tsv in Excel and scrolled through them, looking for quality and tidiness issues.

issues:

1-Tidiness: doggo, floofer, pupper and puppo columns in arc_df table should be merged into one column named "stage" 2-Quality:

unnecessary html tags in source column of twitter archive in place of utility name e.g. Twitter for iPhone

Programmatic Assessment

Quality

1-Incorrect dog names such as "None", "a", "the", "an"

2- rename id to tweet_id to match

3-missing name should be NaN instead of string 'None'

4-timestamp should have datetime values, not strings

5-Separate timestamp into day - month - year (3 columns)

6-Source includes complete url, only need source name

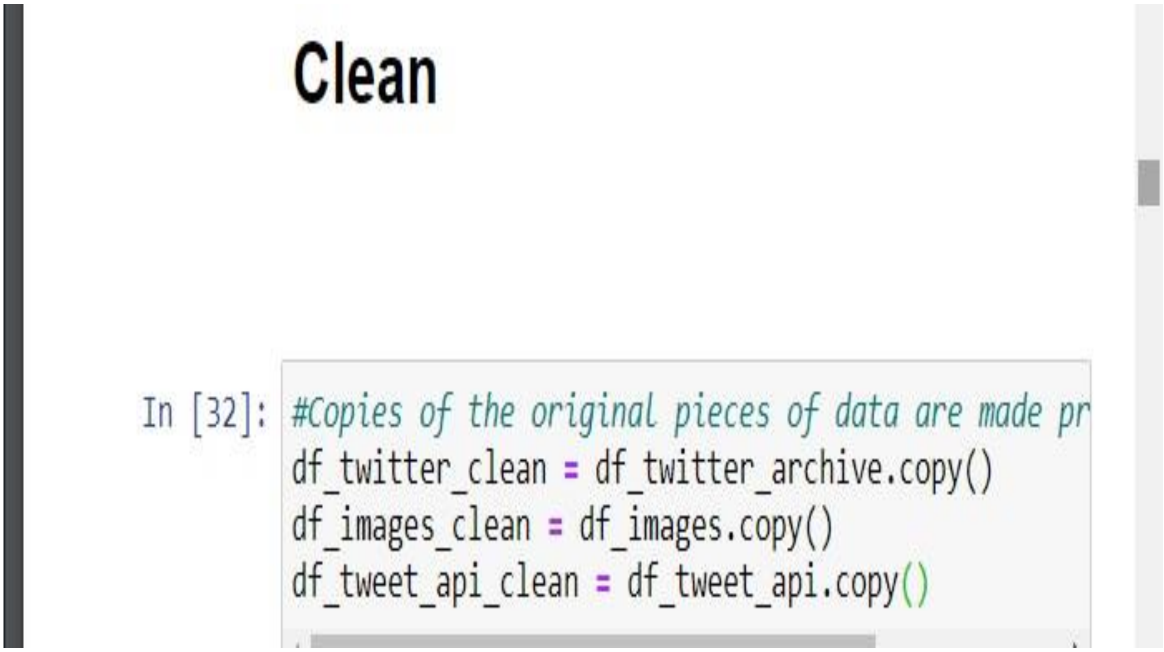
7-Remove retweets (only want the original ratings) based on retweeted_status_id

8-Delete columns that won't be used for analysis

Tidiness

1- Stage name should be one column because each variable forms a column('doggo', 'floofer', 'pupper', 'puppo')
2-Merge all dataframes into one using tweet_id

3. Cleaning Data



Clean

```
In [32]: #Copies of the original pieces of data are made pr
         df_twitter_clean = df_twitter_archive.copy()
         df_images_clean = df_images.copy()
         df_tweet_api_clean = df_tweet_api.copy()
```

As all the quality and tidiness issues were related to (df_twitter_archiive, df_images and df_tweet_api)tables, I created a copy of only this tables and named it (df_twitter_archiive_clean, df_images_clean and df_tweet_api_clean) . For each quality/tidiness issue, I performed the programmatic data cleaning process in 3 stages - Define, Code & Test. During the cleaning process.

4-Storing Data

After the completion of the cleaning process, I stored the archive_clean DataFrame in twitter_archive_master.csv file.