Data Wrangling Process

Data Gathering

Download data file that contain dog breed predictions based on tweet image, and save to file: image_predictions.tsv.

Use twitter_archive_enhanced.csv as the center piece of data, extract its tweet_id's. For each tweet_id, download its json data from Twitter website archive, save to tweet_json.txt.

Data Combining

Load twitter_archive_enhanced.csv as the main data frame. Merge with the dog breed predictions file by tweet_id.

Load tweet_json.txt to data frame, select only id, retweet_count, favorite_count, and merge the counts into the main data frame by (tweet) id. In tweet_json.txt data, id and id_str don't always match. But since id_str is not used in later analysis, the issue does not need to be addressed.

Save the combined data frame (main data frame) to an intermediate data file: combine_tweet_data.csv

Data Cleaning

- Retweets: we do not want to keep retweets, so I removed rows that has non-null retweeted_status_id, and then drop columns: retweeted_status_id, retweeted_status_user_id, retweeted_status_timestamp
- Rating_denominator: there are 18 entries with various values other than 10, cleaned these rows by manually reading the numbers from the tweet's text
- Rating_numerator: there are some entries with values > 20, cleaned these rows by manually reading the numbers from the tweet's text
- In_reply_to_user_id: this information is irrelevant, drop columns: in_reply_to_status_id, in_reply_to_user_id
- Combine dog type column doggo, puppo, pupper and floofer into a single column: type. In the consolidated column, 11 tweets have duplicated dog types: 'doggo' and one other type. Where there is more than 1 type for the tweet, put them into a list and enter under 'type'.
- Dog names: a sorted unique values list of dog names show that there are names that seem to be erroneous: 'a','actually', 'all', 'an', 'by', 'getting', 'his', 'incredibly', 'infuriating', 'just', 'light', 'my', 'not', 'officially', 'one', 'quite', 'space', 'such', 'the', 'unacceptable', 'very'. Replace these names with 'None'.
- P1_dog impact on p1: for image prediction of dog breed that is not a dog, set p1 to Nan ; remove p1_dog column as it is redundant now. Repeat for p2 and p3
- P1_conf impact on p1: very low confidence (<15%) in dog breed prediction is not reliable, set those predictions p1 to NaN. Repeat for p2 and p3
- Where p1 prediction fails to identify a dog breed, p2 or p3 may be successful. Add a new column p, which is assigned with value p1, if p1 is non-null. Otherwise, assign p with p2 or p3, if p2 or p3 is valid, in that order

Save the final cleaned data frame to tweeter_archive_master.csv.