WeRateDogs Twitter Data Wrangling Report

Three different sets of data all gotten from weratedogs account were used for this analysis. The first was the enhanced tweet archive which contained 2356 samples of data. The second is image prediction dataset that had 2075 samples. In this report, I will refer to them as archive dataset, tweet dataset and prediction dataset.

After careful assessment using both visual and programmatic approach, data quality and tidiness issues are discovered within the datasets. Under quality issues, we have:

Quality issues

- Missing records (2327 instead of 2356) in tweet dataset
- Non original tweets and Unwanted columns
- Wrong column data type
- 9/11 event mistaken as rating (archive dataset)
- Non 10 rating denominator value (archive)
- Incorrect value for dog stages (archive)
- Underscore used instead of space to seperate words (p1, p2 and p3 columns) (prediction)
- No uniform character case in p1, p2 and p3 values (prediction)
- Erroneous datatypes (p1_dog, p2_dog, p3_dog) (prediction)

Tidines issues

- Same column bearing different title in another table (timestamp/created_at, id/tweet_id, text/full_text)
- Tweet text duplicated in tweet and archive table
- Dog stages in four columns instead of one column in archive table

I started off by trying to resolve all the missing data issues. Out of 2356 tweet ids residing in archive dataset, I was able to fetch 2327 tweets using tweepy. Next was the tidiness issue. Using df.rename pandas method, I changed the column names timestamp to created, tweet_id to id, text to full_text all in archive dataset. I also changed column name tweet_id to id in prediction dataset. Next, I dropped the text (now full_text) in the tweet dataset. This is to avoid repetition of columns in multiple datasets (tweet and archive).
For the dog stages issue, I used pandas melt function to merge the four columns (floofer, pupper, doggo and puppo into two columns titled stage and staged_value. Dropped stage and renamed stage_value to dog_stage which contained string values of stages.
One of the rules of data tidiness is to have one variable in one column. Since entities and extended entities column in tweet dataset has multiples and werent needed in my future analysis, I dropped them.

Resolving Quality issues

Tweet dataset: The first issue I tackled was converting created_at column dtype to datetime using pandas to_datetime() method. Then i used df.drop function to drop columns that had entire value set to null (geo, coordinates, contributors). I dropped id_str column since it was

the same as id only of type object. Then, I removed all retweets, quotes and reply type samples by dropping samples with non null in_reply_to_status_id and quoted_status_id. Lastly I selected these columns; created_at, id, retweet_count, favorite_count and full_text and dropped the rest.

Archive dataset: Dropped retweets, quotes and replies by deleting data samples that had non null in_reply_to_status_id and retweeted_status_user_id column values. in_reply_to_status_id, in_reply_to_user_id, retweeted_status_id, retweeted_status_user_id and retweeted_status_timestamp columns was also dropped since it wont be contributing to analysis. Next created_at column (formerly timestamp) was converted to dtype datetime.
Using regex, I extracted rating numerator and denominator from full_text thereby resolving the 9/11 issue. I then dropped all non 10 rating denominator since they were basically outliers. Lastly, I applied regex on the full_text column and extracted any occurrence of either of the four known dog stages and filled in the dog_stage column where found or the string *unknown* otherwise.

Prediction dataset: I used df.column_name.str.replace method to replace all underscores in p1, p2 and p3 columns with space and changed the character case to lower. lastly , I converted the p1_dog, p2_dog and p3_dog column to dtype categorical.