WeRateDogs Twitter Archive – Wrangle Report

In this report, I document the wrangling process to gather and cleaning data which's used for analyze WeRateDogs Twitter Archive.

Data Gathering

The data used in this analysis is collected in three methods:

1. Directly download the WeRateDogs Twitter archive data (twitter_archive_enhanced.csv)
2. Use the Requests library to download the tweet image prediction (image_predictions.tsv)from Udacity. This data contains the prediction of dog breed by neutral network.
3. Use the Tweepy library to query additional data via the Twitter API (tweet_json.txt): Since the tweeter just update their developer account, my free account can't download the data. So I have to use the prepared data. This data contains the favorite count and retweet count.

Assess & cleaning

Though access these three data, I make clear of the consist of each table, each column presenting meaning and missing value, duplicated row in each table. Thus I summarize the quality and tidiness problems in the three table.

The *archive_df* , tweets, predictions tables have other 8 quality and 3 tidiness issues.

Since the *tweets* table only provide the favorite count and retweet count, the first step is to merge it with the the *archive_df* table and only keep the row where **tweet_id column** match both two joint table. The new combined table is named as *df_clean* table.

Due to the unnecessarily of retweet and in reply information, dropping the column **retweeted_status_id, retweeted_user_id, retweeted_status_timestamp** and **in_reply_to_status_id, in_reply_to_user_id** in *df_clean* table.

In the dog "name" column, there are some invalid values like "a", "an", "the", "None", etc. I try to extract the dog from the tweeter text. By analysis the name appearance in text, The word "This is" and the uppercase in first letter in the extracted name become the indicators to extract the dog name. At the end, dropping the row with Nan value in **name** column is also a necessary step.

After the investigate of rating score columns in *df_clean* table, including denominator and numerator column, there exits such issue, such as the some values in denominator is not equal to 10. Since the text also contain the rating score, I extracted the numerator and denominator from the **text** column and change the string to float data type. Comparing values of the denominator and numerator column and the corresponding extracted denominator, I dropp the different rows. The last step is to drop the rows which the denominators are not equal to 10.

For the timestamp column in *df_clean* table, I remove "+0000" and change it into datetime.

In the *df_clean* table, I drop the unnecessary **expaned_url** column.

To merge the prediction table, it contain some row with **tweet_id** in the prediction table, which does not match with *df_clean* table. So I drop these row in prediction table and then merge it with *df_clean* table. The new data is named as *archive_df_master* table.

Considering the tidiness issue, I convert the four columns doggo, floofer, pupper, puppo, presenting the stage of the dog to one categorical column named dog_stage. The converted dog stage column posses some unknow dog stage and then replace them “None”. From the column value count, I find some row with multiple dog stages and try to extract the information from the text column. Due to no clue to extract the dog stage in text, I delete these rows with mixed dog stage.

Under the consideration of connivence of rating score, I convert the numerator and denominator into a factional form as a separate column named as rating score.

For the messy information in the **source** column, I extract the text with ‘<’ and ‘>’ as indicator.