Data Wrangling Report

In this project we have three Dataframes coming from three different sources, `df_archive` which is the main dataframe and the most messy and was available just by manual download, `df_images` which have a prediction of dog breed for the images in the tweets, this one existed online and had to be downloaded using `requests` library, and last one `df_stats` which have the full details of tweets and collected through tweeter API (However, I used the provided data as my developer account not yet approved), it has all data but actually we were interested in just two `retweet_count` and `favorite_count`.

Challenges and Solutions

Dataframe `df_archive` was in a very bad shape, many quality and tidiness issues can be found from the first glance. First of all, it had records that we were not interested in, records that's of retweets or replies, those were found with the help of `in_reply_to_status_id` and `retweeted_status_id`, after removing those records, many columns appeared to by with no use as they were empty, hence, we deleted `in_reply_to_status_id`, `retweeted_status_user_id`, `retweeted_status_timestamp`, `retweeted_status_id` and `in_reply_to_user_id` columns. With more investigation, found that there are values in `name` column, that's miss extracted, values like 'a', 'an', 'the', those had to be deleted a long with 'None' which was considered to represent a missing name. Dog rating found in two columns, for better analysis and visualization those had to be combined in one column, hence we divided `rating_numerator` by `rating_denominator` to generate new column `rating` and then the former two were deleted. Actually that helped in solving a problem of many dogs rated in one tweet with one number. Another thing, those rating columns had some outliers we did remove it before all that calculations. One more important issue was; that the dataframe has four columns for dog stage, that we converted to one and dealt with the miss extracted stages and the photos that had more than one dog in different stages. More minor issues were there and can be found in the code file `wrangle_act.ipynb`.

The two other dataframes `df_images` and `df_stats` had much less issues. First, there were three predictions for each tweet photo, were one was enough, hence, we took only the prediction with the highest confidence as long as it was a dog breed, and then we get rid of the other columns and combine the data with the `df_archive` using `tweet_id` column. In `df_stats` there were a lot of quality and

tidiness issues, however we were interested in just two columns `retweet_count` and `favorite_count`, thus, we took those and merge them with `df_archive`, again with help of tweet id.

Conclusion

However, data was messy and required a lot of work, we reached to a pretty good version of it comparing to the begining version, but there are always room for improvements.