Project 7: Data Wrangling

The data wrangling for this project was challenging and satisfying.

While the tweet archive and dog predictions were given files, most effort initially was spent in getting the data from the twitter api.

Gather:

The json methods were useful. However, a lot of time was spent on getting the json file to be read initially into a dataframe. After multiple efforts and research, addition of a single newline in the code helped to read the data.

A lot of time was spent in exploring the twitter api data and how that can be further wrangled. Some time was also spent in trying to pre-process the data in the tweet_text as it contained hashtags, common words, rating info as well as name of the dog in most instances. A set of words to be excluded was built and this can be valuable in wrangling data where natural language processing may be required. Additionally there are libraries that can be leveraged.

The preliminary result from the twitter api feed contained lists within individual cells and needed to be further split into multiple columns – with the result that there were 89 columns at one point in the processed api data, some of which were redundant

Once, the 3 data sets were created, the checking for quality issues and tidiness issues were very iterative. While a set of broad issues were identified, as these issues were getting fixed, you could identify potentially additional cleansing rules.

Assess:

Quality Issues¶
1 14 tweets in the tweet archive were not matched from the Twitter API download¶
2 78 tweets in the tweet archive seem to be replies to other tweets¶
3 181 tweets in the tweet archive seem to be retweets¶
4 59 tweets dont have an image¶
5 745 tweets have 'None' as the name of the dog;¶
6 55 have the name of the dog as a¶
7 Some dogs have been called both a doggo and pupper and so on¶
8 Some of the numerators for the rating dont make sense¶
9 Some of the denominators for the rating dont make sense¶
10 The source column shows where the tweets were sent from iPhone, Android, Tweetdeck, web client etc. but have long descriptions¶
Tidyness Issues¶
1 Dog_stages - doggo, puppo, pupper and floofer are on separate columns. Ideally, they should be on a single column defining the stage of the dog¶
2 The predictions file includes 3 different predictions with different confidence levels. Ideally, there should be a single column with the prediction with the highest confidence level

Pandas and the stackoverflow website was leveraged extensively for pandas syntax issues as well as other common issues.

When joining, the field type of the join field was critical and needed to be manipulated

Column names needed to be renamed at various points in time

The columns were getting unwieldy and needed to be dropped at some points

Cleaning:

Once broad issues such as replies, retweets, no_images were addressed, one needed to go into individual tweet texts to address certain issues, where the rating_numerator and rating_denominator were picking up incorrect values

Individual ratings also needed to be adjusted in a handful of cases by reviewing the tweet text based on weird values in the numerator and denominator

The fact that there could be multiple dog_stages for certain tweets was interesting and needed to be addressed

The 3 predictions P1, P2, P3 needed to be reduced to a single prediction with the highest confidence levels