1. Quality Issues

1.1. CSV Dataset

1.1.1. Remove tweets beyond 1 Aug 2017

Tweets beyond 1 Aug 2017 are dropped from the dataset since dog breed prediction could not be performed for these tweets.

1.1.2. Remove reply and retweets

Since we aim to create a unique entry for each dog, rewteets and reply tweets are dropped from csv dataset.

1.1.3. Unknown dog names

After detailed search in internet, I found that NLTK library that is developed by Stanford University can be used to detect names in the tweet texts. The package is downloaded, installed. The function, *retrieve_name* in the Jupyter notebook is downloaded from stackoverflow resources and modified for purpose.

282 more dog names previously unknown are captured by the NLTK library used. However, 402 names are still unknown. I decided to exclude the dog name in insight and analysis chapter.

The same algorithm is run on the tweet texts which are collected by API dataset one more time.

1.1.4. Rating denominator

There are 18 different denominator values found in the dataset, starting from 0 to 170:

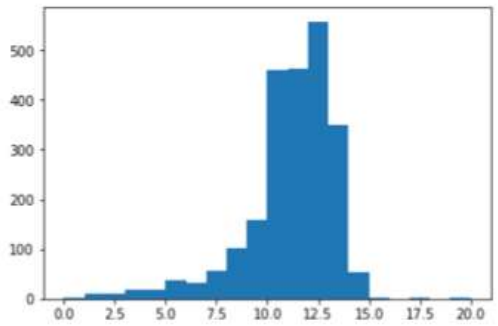
count	2356.000000
mean	10.455433
std	6.745237
min	0.000000
25%	10.000000
50%	10.000000
75%	10.000000
max	170.000000

All of the denominator values are set to 10 as the rating is done by this rule.

1.1.5. Rating numerator

Although rating is to be done between 0 and 10, the numerators are between 0 and 1776 with below statistics:

count	2356.000000
mean	13.126486
std	45.876648
min	0.000000
25%	10.000000
50%	11.000000
75%	12.000000
max	1776.000000



All numerator values are set between 0 and 20. The values beyond 20 are clipped to 20.

1.2. API Dataset

1.2.1. Remove rows with Favorite and Retweet Count are NaN

Since I plan to use these 2 columns for insights and analysis section, I need to ensure data quality. 2 rows with empty favorite and data count are dropped from the dataset.

1.2.2. Retweet and Reply tweets to be removed

The reply tweets and retweets are dropped automatically by left join merge operation with the cleaned CSV dataset. No further work is needed.

1.3. TSV Dataset

1.3.1. False predicted dog breeds

In some tweets, people sent faulty images or the algorithm predicted false results. Consequently, in some entries p1 could not be used directly and p2 and p3 are needed although their confidence level is far below. Although I did not check for confidence level threshold, the confidence level is copied to final dataframe since it may be referred in case of need.

2. Tidiness Issues

2.1. CSV Dataset

2.1.1. Normalised Rating column instead of numerator and denominator columns

The dataset has 2 columns for numerator and denominator of rating. It would be difficult to use in analysis. A new column is created for rating and the rating value here is normalized between 0 and 1 so that a universal rating comparison is possible for all dogs.

2.1.2. Normalised Rating column instead of numerator and denominator columns

There are 4 different columns carrying the stage data. The new created “breed” column gathers the data distributed in these columns and also I searched the tweet texts to extract missing data for some dogs.

2.2. TSV Dataset

2.2.1. New Breed Column for gathering dog breed information

The dog breed predictions are found in 3 separate columns with confidence levels. A new column is created to carry the breed data. Hierarchially, I tried to used p1 if the prediction is a dog, if not I checked p2 and then p3.