Wrangle report

Project Motivation

Context

My goal: wrangle WeRateDogs Twitter data to create interesting and trustworthy analyses and visualizations. The Twitter archive is great, but it only contains very basic tweet information. Additional gathering, then assessing and cleaning is required for "*Wow!*"-worthy analyses and visualizations.

The data

a) Twitter enhanced: This data set contains the tweet IDs, texts and URLs
b) df_Json: this data frame contains tweet id ,retweet counts and likes count, this will help us to analyze the popularity of each tweet.
c) Image predictions dataset: Deep learning algorithms are used to predict the dog breed by analyzing the images associated with each tweet.

Project Details

Your tasks in this project are as follows:

Data wrangling, which consists of:

a) Gathering data (downloadable sources and Twitter API service)
b) Assessing data
c) Cleaning data

a) Gathering data

- Twitter_archive_enhanced was downloaded directly from Udacity and then saved directly into a data frame.
- Image_predictions.tsv was downloaded programmatically from the internet and saved as s csv file .
- Tweet_Json : this file was downloaded directly from Udacity resources.

b) Assessing data

After assessing data, the quality issues are:

1)the timestamp shall be of date type , we shall correct that for e.g by using parsetimes
2)some dog names are : a
3)some dog names hold Nan Values while the text included the dog names
4) outliers in dog rating numerators like 1776 , O ..etc.
5) there are retweets . we shall drop them .

6) 324 false images that don't contain dog names
7) Downsize P1,P2,P3 predictions
8) drop unneeded columns

The tidiness issues are :

1) merge the three dataframes into one dataframe
2) each variable shall be in one column , for example : there shall be one column for dog stages.

C) Clean section :

First of all I made a copy of each data frame ,then I started cleaning the twitter enhanced file by merging dog stages in one column , and after that dropping the unwanted columns like floofer, doggo , pupper and poppo .

Dog names are sometimes : a , it seems that the problem rises because the algorithm checks for what is after "this is " ,however in some tweets the dog name doesn't show immediately after "this is " .

Also some tweets are retweets and we need to drop the retweets .

Also we will make one column that holds that ratings instead of having two columns named rating numerator and rating denominator.

I used the image predictions file to filter the images that don’t contain dog images and dropped them .

Lastly I merged the three data frames into one data frame and dropped any un-needed columns

Conclusion

1) Some tweets arenot for real dogs , however some of them got dozens of likes . this is due to the sense of humor of the tweet owner .
2) High confidence level doesn't mean that it is the correct one . In one tweet it detected the shopping cart as 0.9 Confidence level however there was a Gold retriever dog there . So it is better to use an algorithm that is more tailored to detect dog breeds.
3) There were high outliers in ratings. I removed those outliers from the analysis because the outliers effect the analysis process badly