# wrangle_act

December 26, 2018

# 1 Project: Data Wrangling and Analysis of We rate Dogs!

## 1.1 Table of Contents

## Introduction

> Real-world data rarely comes clean. Using Python and its libraries, we will gather data from a variety of sources and in a variety of formats, assess its quality and tidiness, then clean it. This is called data wrangling. We will document your wrangling efforts in a Jupyter Notebook, plus showcase them through analyses and visualizations using Python (and its libraries) and/or SQL.
>
> The dataset that we will be wrangling (and analyzing and visualizing) is the tweet archive of Twitter user @dog_rates, also known as WeRateDogs. WeRateDogs is a Twitter account that rates people's dogs with a humorous comment about the dog. These ratings almost always have a denominator of 10. The numerators, though? Almost always greater than 10. 11/10, 12/10, 13/10, etc. Why? Because "they're good dogs Brent." WeRateDogs has over 4 million followers and has received international media coverage.
>
> This archive contains basic tweet data (tweet ID, timestamp, text, etc.) for all 5000+ of their tweets as they stood on August 1, 2017. More on this soon.

```
In [1]: # Importing important Packages for the Data Wrangling and analysis

        import pandas as pd
        import numpy as np
        import seaborn as sbn
        import matplotlib.pyplot as pylt
        % matplotlib inline
        import tweepy
        import requests
        import json
import os
import re
from collections import Counter
```

## Data Wrangling

In this section of the report, I have loaded the data and checked for cleanliness, and then trimed and cleaned my dataset for analysis.

### 1.1.1 I) Gather

*Gathering the data from 3 sources:*

1. The WeRateDogs Twitter archive. I am giving this file to you, so imagine it as a file on hand. Download this file manually by clicking the following link: twitter_archive_enhanced.csv

2. The tweet image predictions, i.e., what breed of dog (or other object, animal, etc.) is present in each tweet according to a neural network. This file (image_predictions.tsv) is hosted on Udacity's servers and should be downloaded programmatically using the Requests library and the following URL: https://d17h27t6h515a5.cloudfront.net/topher/2017/August/599fd2ad_image-predictions/image-predictions.tsv

3. Each tweet's retweet count and favorite ("like") count at minimum, and any additional data you find interesting. Using the tweet IDs in the WeRateDogs Twitter archive, query the Twitter API for each tweet's JSON data using Python's Tweepy library and store each tweet's entire set of JSON data in a file called tweet_json.txt file. Each tweet's JSON data should be written to its own line. Then read this .txt file line by line into a pandas DataFrame with (at minimum) tweet ID, retweet count, and favorite count. Note: do not include your Twitter API keys, secrets, and tokens in your project submission.

```
In [2]: #Source 1: Import we rate Dogs twitter archived
        tae = pd.read_csv("twitter-archive-enhanced.csv")
        tae.head()
```

```
Out[2]:              tweet_id  in_reply_to_status_id  in_reply_to_user_id  \
        0  892420643555336193                    NaN                  NaN
        1  892177421306343426                    NaN                  NaN
        2  891815181378084864                    NaN                  NaN
        3  891689557279858688                    NaN                  NaN
        4  891327558926688256                    NaN                  NaN

                          timestamp  \
        0  2017-08-01 16:23:56 +0000
        1  2017-08-01 00:17:27 +0000
        2  2017-07-31 00:18:03 +0000
        3  2017-07-30 15:58:51 +0000
```

```
4  2017-07-29 16:00:24 +0000

                                              source  \
0  <a href="http://twitter.com/download/iphone" r...
1  <a href="http://twitter.com/download/iphone" r...
2  <a href="http://twitter.com/download/iphone" r...
3  <a href="http://twitter.com/download/iphone" r...
4  <a href="http://twitter.com/download/iphone" r...

                                                text  retweeted_status_id  \
0  This is Phineas. He's a mystical boy. Only eve...                  NaN
1  This is Tilly. She's just checking pup on you...                  NaN
2  This is Archie. He is a rare Norwegian Pouncin...                  NaN
3  This is Darla. She commenced a snooze mid meal...                  NaN
4  This is Franklin. He would like you to stop ca...                  NaN

   retweeted_status_user_id retweeted_status_timestamp  \
0                        NaN                        NaN
1                        NaN                        NaN
2                        NaN                        NaN
3                        NaN                        NaN
4                        NaN                        NaN

                                       expanded_urls  rating_numerator  \
0  https://twitter.com/dog_rates/status/892420643...                13
1  https://twitter.com/dog_rates/status/892177421...                13
2  https://twitter.com/dog_rates/status/891815181...                12
3  https://twitter.com/dog_rates/status/891689557...                13
4  https://twitter.com/dog_rates/status/891327558...                12

   rating_denominator      name doggo floofer pupper puppo
0                  10   Phineas  None    None   None  None
1                  10     Tilly  None    None   None  None
2                  10    Archie  None    None   None  None
3                  10     Darla  None    None   None  None
4                  10  Franklin  None    None   None  None
```

In [3]:

```
#Source 2: import the tweet image predictions using requests library
ip_url = 'https://d17h27t6h515a5.cloudfront.net/topher/2017/August/599fd2ad_image-predic
response = requests.get(ip_url)

with open('image-predictions.tsv', mode = 'wb') as file:
    file.write(response.content)

#read the image predictions file
ip = pd.read_csv('image-predictions.tsv', sep = '\t')
ip.head()
```

Out[3]:

```
             tweet_id                                   jpg_url  \
0  666020888022790149  https://pbs.twimg.com/media/CT4udn0WwAA0aMy.jpg
1  666029285002620928  https://pbs.twimg.com/media/CT42GRgUYAA5iDo.jpg
2  666033412701032449  https://pbs.twimg.com/media/CT4521TWwAEvMyu.jpg
3  666044226329800704  https://pbs.twimg.com/media/CT5Dr8HUEAA-lEu.jpg
4  666049248165822465  https://pbs.twimg.com/media/CT5IQmsXIAAKY4A.jpg

   img_num                      p1   p1_conf  p1_dog                  p2  \
0        1  Welsh_springer_spaniel  0.465074    True              collie
1        1                 redbone  0.506826    True  miniature_pinscher
2        1         German_shepherd  0.596461    True            malinois
3        1     Rhodesian_ridgeback  0.408143    True             redbone
4        1      miniature_pinscher  0.560311    True          Rottweiler

    p2_conf  p2_dog                   p3   p3_conf  p3_dog
0  0.156665    True    Shetland_sheepdog  0.061428    True
1  0.074192    True  Rhodesian_ridgeback  0.072010    True
2  0.138584    True           bloodhound  0.116197    True
3  0.360687    True   miniature_pinscher  0.222752    True
4  0.243682    True             Doberman  0.154629    True
```

```
In [4]: # Source 3: Access data using Twitter API
        import tweepy

        #Removing my twitter account details due maintain privacy. If you want to test my code,
        consumer_key = #'Confidential'#
        consumer_secret = #'Confidential'#
        access_token = #'Confidential'#
        access_secret = #'Confidential'#

        auth = tweepy.OAuthHandler(consumer_key, consumer_secret)
        auth.set_access_token(access_token, access_secret)

        api = tweepy.API(auth)

In [5]: #add tweets to tweet_json.txt
        with open('tweet_json.txt', 'w', encoding='utf8') as f:
            for tweet_id in tae['tweet_id']:
                try:
                    tweet = api.get_status(tweet_id, tweet_mode='extended')
                    json.dump(tweet._json, f)
                    f.write('\n')
                except:
                    continue

In [6]: #append the tweets to a list
        tweets_data = []

        tweet_file = open('tweet_json.txt', "r")
        for line in tweet_file:
            try:
                tweet = json.loads(line)
                tweets_data.append(tweet)
            except:
                continue

        tweet_file.close()
```

```
In [7]: #create the df_tweets data frame
        df_tweets = pd.DataFrame()
```

```
In [8]: #add the necessary columns to the data frame
        df_tweets['id'] = list(map(lambda tweet: tweet['id'], tweets_data))
        df_tweets['retweet_count'] = list(map(lambda tweet: tweet['retweet_count'], tweets_data)
        df_tweets['favorite_count'] = list(map(lambda tweet: tweet['favorite_count'], tweets_dat
```

```
In [9]: df_tweets.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 609 entries, 0 to 608
Data columns (total 3 columns):
id                  609 non-null int64
retweet_count       609 non-null int64
favorite_count      609 non-null int64
dtypes: int64(3)
memory usage: 14.4 KB
```

### 1.1.2 II) Assess

After gathering the data from multiple sources, here I have assessed the data in visually and programatically.

#### Assessment of tae:

```
In [10]: # To show tae data for manual assessment
         tae
```

```
Out[10]:                 tweet_id  in_reply_to_status_id  in_reply_to_user_id  \
         0     892420643555336193                    NaN                  NaN
         1     892177421306343426                    NaN                  NaN
         2     891815181378084864                    NaN                  NaN
         3     891689557279858688                    NaN                  NaN
         4     891327558926688256                    NaN                  NaN
         5     891087950875897856                    NaN                  NaN
         6     890971913173991426                    NaN                  NaN
         7     890729181411237888                    NaN                  NaN
```

```
8     890609185150312448                    NaN                  NaN
9     890240255349198849                    NaN                  NaN
10    890006608113172480                    NaN                  NaN
11    889880896479866881                    NaN                  NaN
12    889665388333682689                    NaN                  NaN
13    889638837579907072                    NaN                  NaN
14    889531135344209921                    NaN                  NaN
15    889278841981685760                    NaN                  NaN
16    888917238123831296                    NaN                  NaN
17    888804989199671297                    NaN                  NaN
18    888554962724278272                    NaN                  NaN
19    888202515573088257                    NaN                  NaN
20    888078434458587136                    NaN                  NaN
21    887705289381826560                    NaN                  NaN
22    887517139158093824                    NaN                  NaN
23    887473957103951883                    NaN                  NaN
24    887343217045368832                    NaN                  NaN
25    887101392804085760                    NaN                  NaN
26    886983233522544640                    NaN                  NaN
27    886736880519319552                    NaN                  NaN
28    886680336477933568                    NaN                  NaN
29    886366144734445568                    NaN                  NaN
...                  ...                    ...                  ...
2326  666411507551481857                    NaN                  NaN
2327  666407126856765440                    NaN                  NaN
2328  666396247373291520                    NaN                  NaN
2329  666373753744588802                    NaN                  NaN
2330  666362758909284353                    NaN                  NaN
2331  666353288456101888                    NaN                  NaN
2332  666345417576210432                    NaN                  NaN
2333  666337882303524864                    NaN                  NaN
2334  666293911632134144                    NaN                  NaN
2335  666287406224695296                    NaN                  NaN
2336  666273097616637952                    NaN                  NaN
2337  666268910803644416                    NaN                  NaN
2338  666104133288665088                    NaN                  NaN
2339  666102155909144576                    NaN                  NaN
2340  666099513787052032                    NaN                  NaN
2341  666094000022159362                    NaN                  NaN
2342  666082916733198337                    NaN                  NaN
2343  666073100786774016                    NaN                  NaN
2344  666071193221509120                    NaN                  NaN
2345  666063827256086533                    NaN                  NaN
2346  666058600524156928                    NaN                  NaN
2347  666057090499244032                    NaN                  NaN
2348  666055525042405380                    NaN                  NaN
2349  666051853826850816                    NaN                  NaN
2350  666050758794694657                    NaN                  NaN
```

```
2351  666049248165822465                    NaN                  NaN
2352  666044226329800704                    NaN                  NaN
2353  666033412701032449                    NaN                  NaN
2354  666029285002620928                    NaN                  NaN
2355  666020888022790149                    NaN                  NaN

                      timestamp  \
0     2017-08-01 16:23:56 +0000
1     2017-08-01 00:17:27 +0000
2     2017-07-31 00:18:03 +0000
3     2017-07-30 15:58:51 +0000
4     2017-07-29 16:00:24 +0000
5     2017-07-29 00:08:17 +0000
6     2017-07-28 16:27:12 +0000
7     2017-07-28 00:22:40 +0000
8     2017-07-27 16:25:51 +0000
9     2017-07-26 15:59:51 +0000
10    2017-07-26 00:31:25 +0000
11    2017-07-25 16:11:53 +0000
12    2017-07-25 01:55:32 +0000
13    2017-07-25 00:10:02 +0000
14    2017-07-24 17:02:04 +0000
15    2017-07-24 00:19:32 +0000
16    2017-07-23 00:22:39 +0000
17    2017-07-22 16:56:37 +0000
18    2017-07-22 00:23:06 +0000
19    2017-07-21 01:02:36 +0000
20    2017-07-20 16:49:33 +0000
21    2017-07-19 16:06:48 +0000
22    2017-07-19 03:39:09 +0000
23    2017-07-19 00:47:34 +0000
24    2017-07-18 16:08:03 +0000
25    2017-07-18 00:07:08 +0000
26    2017-07-17 16:17:36 +0000
27    2017-07-16 23:58:41 +0000
28    2017-07-16 20:14:00 +0000
29    2017-07-15 23:25:31 +0000
...                         ...
2326  2015-11-17 00:24:19 +0000
2327  2015-11-17 00:06:54 +0000
2328  2015-11-16 23:23:41 +0000
2329  2015-11-16 21:54:18 +0000
2330  2015-11-16 21:10:36 +0000
2331  2015-11-16 20:32:58 +0000
2332  2015-11-16 20:01:42 +0000
2333  2015-11-16 19:31:45 +0000
2334  2015-11-16 16:37:02 +0000
2335  2015-11-16 16:11:11 +0000
```

```
2336  2015-11-16 15:14:19 +0000
2337  2015-11-16 14:57:41 +0000
2338  2015-11-16 04:02:55 +0000
2339  2015-11-16 03:55:04 +0000
2340  2015-11-16 03:44:34 +0000
2341  2015-11-16 03:22:39 +0000
2342  2015-11-16 02:38:37 +0000
2343  2015-11-16 01:59:36 +0000
2344  2015-11-16 01:52:02 +0000
2345  2015-11-16 01:22:45 +0000
2346  2015-11-16 01:01:59 +0000
2347  2015-11-16 00:55:59 +0000
2348  2015-11-16 00:49:46 +0000
2349  2015-11-16 00:35:11 +0000
2350  2015-11-16 00:30:50 +0000
2351  2015-11-16 00:24:50 +0000
2352  2015-11-16 00:04:52 +0000
2353  2015-11-15 23:21:54 +0000
2354  2015-11-15 23:05:30 +0000
2355  2015-11-15 22:32:08 +0000

                                                 source  \
0     <a href="http://twitter.com/download/iphone" r...
1     <a href="http://twitter.com/download/iphone" r...
2     <a href="http://twitter.com/download/iphone" r...
3     <a href="http://twitter.com/download/iphone" r...
4     <a href="http://twitter.com/download/iphone" r...
5     <a href="http://twitter.com/download/iphone" r...
6     <a href="http://twitter.com/download/iphone" r...
7     <a href="http://twitter.com/download/iphone" r...
8     <a href="http://twitter.com/download/iphone" r...
9     <a href="http://twitter.com/download/iphone" r...
10    <a href="http://twitter.com/download/iphone" r...
11    <a href="http://twitter.com/download/iphone" r...
12    <a href="http://twitter.com/download/iphone" r...
13    <a href="http://twitter.com/download/iphone" r...
14    <a href="http://twitter.com/download/iphone" r...
15    <a href="http://twitter.com/download/iphone" r...
16    <a href="http://twitter.com/download/iphone" r...
17    <a href="http://twitter.com/download/iphone" r...
18    <a href="http://twitter.com/download/iphone" r...
19    <a href="http://twitter.com/download/iphone" r...
20    <a href="http://twitter.com/download/iphone" r...
21    <a href="http://twitter.com/download/iphone" r...
22    <a href="http://twitter.com/download/iphone" r...
23    <a href="http://twitter.com/download/iphone" r...
24    <a href="http://twitter.com/download/iphone" r...
25    <a href="http://twitter.com/download/iphone" r...
```

```
26    <a href="http://twitter.com/download/iphone" r...
27    <a href="http://twitter.com/download/iphone" r...
28    <a href="http://twitter.com/download/iphone" r...
29    <a href="http://twitter.com/download/iphone" r...
...                                                ...
2326  <a href="http://twitter.com/download/iphone" r...
2327  <a href="http://twitter.com/download/iphone" r...
2328  <a href="http://twitter.com/download/iphone" r...
2329  <a href="http://twitter.com/download/iphone" r...
2330  <a href="http://twitter.com/download/iphone" r...
2331  <a href="http://twitter.com/download/iphone" r...
2332  <a href="http://twitter.com/download/iphone" r...
2333  <a href="http://twitter.com/download/iphone" r...
2334  <a href="http://twitter.com/download/iphone" r...
2335  <a href="http://twitter.com/download/iphone" r...
2336  <a href="http://twitter.com/download/iphone" r...
2337  <a href="http://twitter.com/download/iphone" r...
2338  <a href="http://twitter.com/download/iphone" r...
2339  <a href="http://twitter.com/download/iphone" r...
2340  <a href="http://twitter.com/download/iphone" r...
2341  <a href="http://twitter.com/download/iphone" r...
2342  <a href="http://twitter.com/download/iphone" r...
2343  <a href="http://twitter.com/download/iphone" r...
2344  <a href="http://twitter.com/download/iphone" r...
2345  <a href="http://twitter.com/download/iphone" r...
2346  <a href="http://twitter.com/download/iphone" r...
2347  <a href="http://twitter.com/download/iphone" r...
2348  <a href="http://twitter.com/download/iphone" r...
2349  <a href="http://twitter.com/download/iphone" r...
2350  <a href="http://twitter.com/download/iphone" r...
2351  <a href="http://twitter.com/download/iphone" r...
2352  <a href="http://twitter.com/download/iphone" r...
2353  <a href="http://twitter.com/download/iphone" r...
2354  <a href="http://twitter.com/download/iphone" r...
2355  <a href="http://twitter.com/download/iphone" r...

                                                   text  retweeted_status_id  \
0     This is Phineas. He's a mystical boy. Only eve...                  NaN
1     This is Tilly. She's just checking pup on you...                  NaN
2     This is Archie. He is a rare Norwegian Pouncin...                  NaN
3     This is Darla. She commenced a snooze mid meal...                  NaN
4     This is Franklin. He would like you to stop ca...                  NaN
5     Here we have a majestic great white breaching ...                  NaN
6     Meet Jax. He enjoys ice cream so much he gets ...                  NaN
7     When you watch your owner call another dog a g...                  NaN
8     This is Zoey. She doesn't want to be one of th...                  NaN
9     This is Cassie. She is a college pup. Studying...                  NaN
10    This is Koda. He is a South Australian decksha...                  NaN
```

```
11    This is Bruno. He is a service shark. Only get...                  NaN
12    Here's a puppo that seems to be on the fence a...                  NaN
13    This is Ted. He does his best. Sometimes that'...                  NaN
14    This is Stuart. He's sporting his favorite fan...                  NaN
15    This is Oliver. You're witnessing one of his m...                  NaN
16    This is Jim. He found a fren. Taught him how t...                  NaN
17    This is Zeke. He has a new stick. Very proud o...                  NaN
18    This is Ralphus. He's powering up. Attempting ...                  NaN
19    RT @dog_rates: This is Canela. She attempted s...         8.874740e+17
20    This is Gerald. He was just told he didn't get...                  NaN
21    This is Jeffrey. He has a monopoly on the pool...                  NaN
22    I've yet to rate a Venezuelan Hover Wiener. Th...                  NaN
23    This is Canela. She attempted some fancy porch...                  NaN
24    You may not have known you needed to see this ...                  NaN
25    This... is a Jubilant Antarctic House Bear. We...                  NaN
26    This is Maya. She's very shy. Rarely leaves he...                  NaN
27    This is Mingus. He's a wonderful father to his...                  NaN
28    This is Derek. He's late for a dog meeting. 13...                  NaN
29    This is Roscoe. Another pupper fallen victim t...                  NaN
...                                                 ...                  ...
2326  This is quite the dog. Gets really excited whe...                  NaN
2327  This is a southern Vesuvius bumblegruff. Can d...                  NaN
2328  Oh goodness. A super rare northeast Qdoba kang...                  NaN
2329  Those are sunglasses and a jean jacket. 11/10 ...                  NaN
2330  Unique dog here. Very small. Lives in containe...                  NaN
2331  Here we have a mixed Asiago from the Galápagos...                  NaN
2332  Look at this jokester thinking seat belt laws ...                  NaN
2333  This is an extremely rare horned Parthenon. No...                  NaN
2334  This is a funny dog. Weird toes. Won't come do...                  NaN
2335  This is an Albanian 3 1/2 legged  Episcopalian...                  NaN
2336     Can take selfies 11/10 https://t.co/ws2AMaNwPW                  NaN
2337  Very concerned about fellow dog trapped in com...                  NaN
2338  Not familiar with this breed. No tail (weird)...                  NaN
2339  Oh my. Here you are seeing an Adobe Setter giv...                  NaN
2340  Can stand on stump for what seems like a while...                  NaN
2341  This appears to be a Mongolian Presbyterian mi...                  NaN
2342  Here we have a well-established sunblockerspan...                  NaN
2343  Let's hope this flight isn't Malaysian (lol). ...                  NaN
2344  Here we have a northern speckled Rhododendron...                  NaN
2345  This is the happiest dog you will ever see. Ve...                  NaN
2346  Here is the Rand Paul of retrievers folks! He'...                  NaN
2347  My oh my. This is a rare blond Canadian terrie...                  NaN
2348  Here is a Siberian heavily armored polar bear ...                  NaN
2349  This is an odd dog. Hard on the outside but lo...                  NaN
2350  This is a truly beautiful English Wilson Staff...                  NaN
2351  Here we have a 1949 1st generation vulpix. Enj...                  NaN
2352  This is a purebred Piers Morgan. Loves to Netf...                  NaN
2353  Here is a very happy pup. Big fan of well-main...                  NaN
```

```
2354  This is a western brown Mitsubishi terrier. Up...                 NaN
2355  Here we have a Japanese Irish Setter. Lost eye...                 NaN

      retweeted_status_user_id retweeted_status_timestamp  \
0                          NaN                        NaN
1                          NaN                        NaN
2                          NaN                        NaN
3                          NaN                        NaN
4                          NaN                        NaN
5                          NaN                        NaN
6                          NaN                        NaN
7                          NaN                        NaN
8                          NaN                        NaN
9                          NaN                        NaN
10                         NaN                        NaN
11                         NaN                        NaN
12                         NaN                        NaN
13                         NaN                        NaN
14                         NaN                        NaN
15                         NaN                        NaN
16                         NaN                        NaN
17                         NaN                        NaN
18                         NaN                        NaN
19                4.196984e+09  2017-07-19 00:47:34 +0000
20                         NaN                        NaN
21                         NaN                        NaN
22                         NaN                        NaN
23                         NaN                        NaN
24                         NaN                        NaN
25                         NaN                        NaN
26                         NaN                        NaN
27                         NaN                        NaN
28                         NaN                        NaN
29                         NaN                        NaN
...                        ...                        ...
2326                       NaN                        NaN
2327                       NaN                        NaN
2328                       NaN                        NaN
2329                       NaN                        NaN
2330                       NaN                        NaN
2331                       NaN                        NaN
2332                       NaN                        NaN
2333                       NaN                        NaN
2334                       NaN                        NaN
2335                       NaN                        NaN
2336                       NaN                        NaN
2337                       NaN                        NaN
2338                       NaN                        NaN
```

```
2339                       NaN                       NaN
2340                       NaN                       NaN
2341                       NaN                       NaN
2342                       NaN                       NaN
2343                       NaN                       NaN
2344                       NaN                       NaN
2345                       NaN                       NaN
2346                       NaN                       NaN
2347                       NaN                       NaN
2348                       NaN                       NaN
2349                       NaN                       NaN
2350                       NaN                       NaN
2351                       NaN                       NaN
2352                       NaN                       NaN
2353                       NaN                       NaN
2354                       NaN                       NaN
2355                       NaN                       NaN

                                          expanded_urls  rating_numerator  \
0     https://twitter.com/dog_rates/status/892420643...                13
1     https://twitter.com/dog_rates/status/892177421...                13
2     https://twitter.com/dog_rates/status/891815181...                12
3     https://twitter.com/dog_rates/status/891689557...                13
4     https://twitter.com/dog_rates/status/891327558...                12
5     https://twitter.com/dog_rates/status/891087950...                13
6     https://gofundme.com/ydvmve-surgery-for-jax,ht...                13
7     https://twitter.com/dog_rates/status/890729181...                13
8     https://twitter.com/dog_rates/status/890609185...                13
9     https://twitter.com/dog_rates/status/890240255...                14
10    https://twitter.com/dog_rates/status/890006608...                13
11    https://twitter.com/dog_rates/status/889880896...                13
12    https://twitter.com/dog_rates/status/889665388...                13
13    https://twitter.com/dog_rates/status/889638837...                12
14    https://twitter.com/dog_rates/status/889531135...                13
15    https://twitter.com/dog_rates/status/889278841...                13
16    https://twitter.com/dog_rates/status/888917238...                12
17    https://twitter.com/dog_rates/status/888804989...                13
18    https://twitter.com/dog_rates/status/888554962...                13
19    https://twitter.com/dog_rates/status/887473957...                13
20    https://twitter.com/dog_rates/status/888078434...                12
21    https://twitter.com/dog_rates/status/887705289...                13
22    https://twitter.com/dog_rates/status/887517139...                14
23    https://twitter.com/dog_rates/status/887473957...                13
24    https://twitter.com/dog_rates/status/887343217...                13
25    https://twitter.com/dog_rates/status/887101392...                12
26    https://twitter.com/dog_rates/status/886983233...                13
27    https://www.gofundme.com/mingusneedsus,https:/...                13
28    https://twitter.com/dog_rates/status/886680336...                13
```

```
29    https://twitter.com/dog_rates/status/886366144...                12
...                                                 ...               ...
2326  https://twitter.com/dog_rates/status/666411507...                 2
2327  https://twitter.com/dog_rates/status/666407126...                 7
2328  https://twitter.com/dog_rates/status/666396247...                 9
2329  https://twitter.com/dog_rates/status/666373753...                11
2330  https://twitter.com/dog_rates/status/666362758...                 6
2331  https://twitter.com/dog_rates/status/666353288...                 8
2332  https://twitter.com/dog_rates/status/666345417...                10
2333  https://twitter.com/dog_rates/status/666337882...                 9
2334  https://twitter.com/dog_rates/status/666293911...                 3
2335  https://twitter.com/dog_rates/status/666287406...                 1
2336  https://twitter.com/dog_rates/status/666273097...                11
2337  https://twitter.com/dog_rates/status/666268910...                10
2338  https://twitter.com/dog_rates/status/666104133...                 1
2339  https://twitter.com/dog_rates/status/666102155...                11
2340  https://twitter.com/dog_rates/status/666099513...                 8
2341  https://twitter.com/dog_rates/status/666094000...                 9
2342  https://twitter.com/dog_rates/status/666082916...                 6
2343  https://twitter.com/dog_rates/status/666073100...                10
2344  https://twitter.com/dog_rates/status/666071193...                 9
2345  https://twitter.com/dog_rates/status/666063827...                10
2346  https://twitter.com/dog_rates/status/666058600...                 8
2347  https://twitter.com/dog_rates/status/666057090...                 9
2348  https://twitter.com/dog_rates/status/666055525...                10
2349  https://twitter.com/dog_rates/status/666051853...                 2
2350  https://twitter.com/dog_rates/status/666050758...                10
2351  https://twitter.com/dog_rates/status/666049248...                 5
2352  https://twitter.com/dog_rates/status/666044226...                 6
2353  https://twitter.com/dog_rates/status/666033412...                 9
2354  https://twitter.com/dog_rates/status/666029285...                 7
2355  https://twitter.com/dog_rates/status/666020888...                 8

      rating_denominator      name  doggo floofer  pupper  puppo
0                     10   Phineas   None    None    None   None
1                     10     Tilly   None    None    None   None
2                     10    Archie   None    None    None   None
3                     10     Darla   None    None    None   None
4                     10  Franklin   None    None    None   None
5                     10      None   None    None    None   None
6                     10       Jax   None    None    None   None
7                     10      None   None    None    None   None
8                     10      Zoey   None    None    None   None
9                     10    Cassie  doggo    None    None   None
10                    10      Koda   None    None    None   None
11                    10     Bruno   None    None    None   None
12                    10      None   None    None    None  puppo
13                    10       Ted   None    None    None   None
```

```
14                    10    Stuart   None    None    None  puppo
15                    10    Oliver   None    None    None   None
16                    10       Jim   None    None    None   None
17                    10      Zeke   None    None    None   None
18                    10   Ralphus   None    None    None   None
19                    10    Canela   None    None    None   None
20                    10    Gerald   None    None    None   None
21                    10   Jeffrey   None    None    None   None
22                    10      such   None    None    None   None
23                    10    Canela   None    None    None   None
24                    10      None   None    None    None   None
25                    10      None   None    None    None   None
26                    10      Maya   None    None    None   None
27                    10    Mingus   None    None    None   None
28                    10     Derek   None    None    None   None
29                    10    Roscoe   None    None  pupper   None
...                  ...       ...    ...     ...     ...    ...
2326                  10     quite   None    None    None   None
2327                  10         a   None    None    None   None
2328                  10      None   None    None    None   None
2329                  10      None   None    None    None   None
2330                  10      None   None    None    None   None
2331                  10      None   None    None    None   None
2332                  10      None   None    None    None   None
2333                  10        an   None    None    None   None
2334                  10         a   None    None    None   None
2335                   2        an   None    None    None   None
2336                  10      None   None    None    None   None
2337                  10      None   None    None    None   None
2338                  10      None   None    None    None   None
2339                  10      None   None    None    None   None
2340                  10      None   None    None    None   None
2341                  10      None   None    None    None   None
2342                  10      None   None    None    None   None
2343                  10      None   None    None    None   None
2344                  10      None   None    None    None   None
2345                  10       the   None    None    None   None
2346                  10       the   None    None    None   None
2347                  10         a   None    None    None   None
2348                  10         a   None    None    None   None
2349                  10        an   None    None    None   None
2350                  10         a   None    None    None   None
2351                  10      None   None    None    None   None
2352                  10         a   None    None    None   None
2353                  10         a   None    None    None   None
2354                  10         a   None    None    None   None
2355                  10      None   None    None    None   None
```

```
         [2356 rows x 17 columns]
```

```
In [11]: tae.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2356 entries, 0 to 2355
Data columns (total 17 columns):
tweet_id                      2356 non-null int64
in_reply_to_status_id         78 non-null float64
in_reply_to_user_id           78 non-null float64
timestamp                     2356 non-null object
source                        2356 non-null object
text                          2356 non-null object
retweeted_status_id           181 non-null float64
retweeted_status_user_id      181 non-null float64
retweeted_status_timestamp    181 non-null object
expanded_urls                 2297 non-null object
rating_numerator              2356 non-null int64
rating_denominator            2356 non-null int64
name                          2356 non-null object
doggo                         2356 non-null object
floofer                       2356 non-null object
pupper                        2356 non-null object
puppo                         2356 non-null object
dtypes: float64(4), int64(3), object(10)
memory usage: 313.0+ KB
```

```
In [12]: tae.describe()
```

```
Out[12]:            tweet_id  in_reply_to_status_id  in_reply_to_user_id  \
         count  2.356000e+03           7.800000e+01         7.800000e+01
         mean   7.427716e+17           7.455079e+17         2.014171e+16
         std    6.856705e+16           7.582492e+16         1.252797e+17
         min    6.660209e+17           6.658147e+17         1.185634e+07
         25%    6.783989e+17           6.757419e+17         3.086374e+08
         50%    7.196279e+17           7.038708e+17         4.196984e+09
         75%    7.993373e+17           8.257804e+17         4.196984e+09
         max    8.924206e+17           8.862664e+17         8.405479e+17

                retweeted_status_id  retweeted_status_user_id  rating_numerator  \
         count         1.810000e+02              1.810000e+02       2356.000000
         mean          7.720400e+17              1.241698e+16         13.126486
         std           6.236928e+16              9.599254e+16         45.876648
         min           6.661041e+17              7.832140e+05          0.000000
         25%           7.186315e+17              4.196984e+09         10.000000
         50%           7.804657e+17              4.196984e+09         11.000000
         75%           8.203146e+17              4.196984e+09         12.000000
         max           8.874740e+17              7.874618e+17       1776.000000
```

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

```
In [13]: #To check for duplicates
         sum(tae.tweet_id.duplicated())
```

```
Out[13]: 0
```

```
In [14]: #To check the source details
         tae.source.value_counts()
```

```
Out[14]: <a href="http://twitter.com/download/iphone" rel="nofollow">Twitter for iPhone</a>
         <a href="http://vine.co" rel="nofollow">Vine - Make a Scene</a>
         <a href="http://twitter.com" rel="nofollow">Twitter Web Client</a>
         <a href="https://about.twitter.com/products/tweetdeck" rel="nofollow">TweetDeck</a>
         Name: source, dtype: int64
```

```
In [15]: tae.groupby(["doggo","floofer","pupper","puppo"]).count()['tweet_id']
```

```
Out[15]: doggo  floofer  pupper  puppo
         None   None     None    None     1976
                                 puppo      29
                         pupper  None      245
                floofer  None    None        9
         doggo  None     None    None       83
                                 puppo       1
                         pupper  None       12
                floofer  None    None        1
         Name: tweet_id, dtype: int64
```

```
In [16]: tae.groupby(['doggo']).count()['tweet_id']
```

```
Out[16]: doggo
         None     2259
         doggo      97
         Name: tweet_id, dtype: int64
```

```
In [17]: tae.name.value_counts()
```

```
Out[17]: None           745
         a               55
```

```
Charlie        12
Cooper         11
Oliver         11
Lucy           11
Tucker         10
Penny          10
Lola           10
Bo              9
Winston         9
Sadie           8
the             8
Bailey          7
Buddy           7
Daisy           7
an              7
Toby            7
Jack            6
Rusty           6
Jax             6
Leo             6
Koda            6
Dave            6
Scout           6
Milo            6
Bella           6
Oscar           6
Stanley         6
Louis           5
             ...
Bowie           1
Ronnie          1
Noosh           1
Augie           1
Bloo            1
Brandonald      1
Nimbus          1
Chadrick        1
Genevieve       1
Strudel         1
Kobe            1
Gabby           1
old             1
Fabio           1
Aja             1
Lipton          1
Marvin          1
Beemo           1
Jennifur        1
```

```
Mookie          1
Moofasa         1
Kaiya           1
Caryl           1
Andru           1
Pupcasso        1
Cora            1
Mauve           1
Brudge          1
Glacier         1
Damon           1
Name: name, Length: 957, dtype: int64
```

```
In [18]: #Finding what is in lower case of name
         lower = []

         for word in tae['name']:
             if word.islower():
                 lower.append(word)

         # To check the counts of eacy name in lower case
         Counter(lower)
```

```
Out[18]: Counter({'such': 1,
                  'a': 55,
                  'quite': 4,
                  'not': 2,
                  'one': 4,
                  'incredibly': 1,
                  'mad': 2,
                  'an': 7,
                  'very': 5,
                  'just': 4,
                  'my': 1,
                  'his': 1,
                  'actually': 2,
                  'getting': 2,
                  'this': 1,
                  'unacceptable': 1,
                  'all': 1,
                  'old': 1,
                  'infuriating': 1,
                  'the': 8,
                  'by': 1,
                  'officially': 1,
                  'life': 1,
                  'light': 1,
                  'space': 1})
```

### 1.1.3 Quality Assessment of "tae":

1. Timestamp is in Object format
2. unnecessary information along with the hastags are present in the text column
3. Rating Denominator is greater than 10
4. Rating Numerator is greater than rating denominator
5. names are in lower case are not seems like a name

### 1.1.4 Tideness Assessment of "tae":

1. Multiple columns ("doggo","floofer","pupper","puppo") of Dog types
2. unnecessary columns for our analysis such as "retweeted_status_id", "retweeted_status_user_id", "retweeted_status_timestamp", "in_reply_to_status_id", "in_reply_to_user_id" and "expanded_urls"

**Assessment of ip:**

```
In [19]: # To assess the ip data manually
         ip
```

```
Out[19]:              tweet_id                                          jpg_url  \
         0     666020888022790149  https://pbs.twimg.com/media/CT4udn0WwAA0aMy.jpg
         1     666029285002620928  https://pbs.twimg.com/media/CT42GRgUYAA5iDo.jpg
         2     666033412701032449  https://pbs.twimg.com/media/CT4521TWwAEvMyu.jpg
         3     666044226329800704  https://pbs.twimg.com/media/CT5Dr8HUEAA-lEu.jpg
         4     666049248165822465  https://pbs.twimg.com/media/CT5IQmsXIAAKY4A.jpg
         5     666050758794694657  https://pbs.twimg.com/media/CT5Jof1WUAEuVxN.jpg
         6     666051853826850816  https://pbs.twimg.com/media/CT5KoJ1WoAAJash.jpg
         7     666055525042405380  https://pbs.twimg.com/media/CT5N9tpXIAAifs1.jpg
         8     666057090499244032  https://pbs.twimg.com/media/CT5PY90WoAAQGLo.jpg
         9     666058600524156928  https://pbs.twimg.com/media/CT5Qw94XAAA_2dP.jpg
         10    666063827256086533  https://pbs.twimg.com/media/CT5Vg_wXIAAXfnj.jpg
         11    666071193221509120  https://pbs.twimg.com/media/CT5cN_3WEAAlOoZ.jpg
         12    666073100786774016  https://pbs.twimg.com/media/CT5d9DZXAAALcwe.jpg
         13    666082916733198337  https://pbs.twimg.com/media/CT5m4VGWEAAtKc8.jpg
         14    666094000022159362  https://pbs.twimg.com/media/CT5w9gUW4AAsBNN.jpg
         15    666099513787052032  https://pbs.twimg.com/media/CT51-JJUEAA6hV8.jpg
         16    666102155909144576  https://pbs.twimg.com/media/CT54YGiWUAEZnoK.jpg
         17    666104133288665088  https://pbs.twimg.com/media/CT56LSZWoAAlJj2.jpg
         18    666268910803644416  https://pbs.twimg.com/media/CT8QCd1WEAADXws.jpg
         19    666273097616637952  https://pbs.twimg.com/media/CT8T1mtUwAA3aqm.jpg
         20    666287406224695296  https://pbs.twimg.com/media/CT8g3BpUEAAuFjg.jpg
         21    666293911632134144  https://pbs.twimg.com/media/CT8mx7KW4AEQu8N.jpg
         22    666337882303524864  https://pbs.twimg.com/media/CT9OwFIWEAMuRje.jpg
         23    666345417576210432  https://pbs.twimg.com/media/CT9Vn7PWoAA_ZCM.jpg
         24    666353288456101888  https://pbs.twimg.com/media/CT9cx0tUEAAhNN_.jpg
         25    666362758909284353  https://pbs.twimg.com/media/CT9lXGsUcAAyUFt.jpg
         26    666373753744588802  https://pbs.twimg.com/media/CT9vZEYWUAAlZ05.jpg
         27    666396247373291520  https://pbs.twimg.com/media/CT-D2ZHWIAA3gK1.jpg
```

```
28    666407126856765440    https://pbs.twimg.com/media/CT-NvwmW4AAugGZ.jpg
29    666411507551481857    https://pbs.twimg.com/media/CT-RugiWIAELEaq.jpg
...                  ...                                                ...
2045  886366144734445568    https://pbs.twimg.com/media/DE0BTnQUwAApKEH.jpg
2046  886680336477933568    https://pbs.twimg.com/media/DE4fEDzWAAAyHMM.jpg
2047  886736880519319552    https://pbs.twimg.com/media/DE5Se8FXcAAJFx4.jpg
2048  886983233522544640    https://pbs.twimg.com/media/DE8yicJW0AAAvBJ.jpg
2049  887101392804085760    https://pbs.twimg.com/media/DE-eAq6UwAA-jaE.jpg
2050  887343217045368832  https://pbs.twimg.com/ext_tw_video_thumb/88734...
2051  887473957103951883    https://pbs.twimg.com/media/DFDw2tyUQAAAFke.jpg
2052  887517139158093824  https://pbs.twimg.com/ext_tw_video_thumb/88751...
2053  887705289381826560    https://pbs.twimg.com/media/DFHDQBbXgAEqY7t.jpg
2054  888078434458587136    https://pbs.twimg.com/media/DFMWn56WsAAkA7B.jpg
2055  888202515573088257    https://pbs.twimg.com/media/DFDw2tyUQAAAFke.jpg
2056  888554962724278272    https://pbs.twimg.com/media/DFTH_O-UQAACu20.jpg
2057  888804989199671297    https://pbs.twimg.com/media/DFWra-3VYAA2piG.jpg
2058  888917238123831296    https://pbs.twimg.com/media/DFYRgsOUQAARGhO.jpg
2059  889278841981685760  https://pbs.twimg.com/ext_tw_video_thumb/88927...
2060  889531135344209921    https://pbs.twimg.com/media/DFg_2PVW0AEHN3p.jpg
2061  889638837579907072    https://pbs.twimg.com/media/DFihzFfXsAYGDPR.jpg
2062  889665388333682689    https://pbs.twimg.com/media/DFi579UWsAAatzw.jpg
2063  889880896479866881    https://pbs.twimg.com/media/DFl99B1WsAITKsg.jpg
2064  890006608113172480    https://pbs.twimg.com/media/DFnwSY4WAAAMliS.jpg
2065  890240255349198849    https://pbs.twimg.com/media/DFrEyVuW0AAO3t9.jpg
2066  890609185150312448    https://pbs.twimg.com/media/DFwUU__XcAEpyXI.jpg
2067  890729181411237888    https://pbs.twimg.com/media/DFyBahAVwAAhUTd.jpg
2068  890971913173991426    https://pbs.twimg.com/media/DF1eOmZXUAALUcq.jpg
2069  891087950875897856    https://pbs.twimg.com/media/DF3HwyEWsAABqE6.jpg
2070  891327558926688256    https://pbs.twimg.com/media/DF6hr6BUMAAzZgT.jpg
2071  891689557279858688    https://pbs.twimg.com/media/DF_q7IAWsAEuuN8.jpg
2072  891815181378084864    https://pbs.twimg.com/media/DGBdLU1WsAANxJ9.jpg
2073  892177421306343426    https://pbs.twimg.com/media/DGGmoV4XsAAUL6n.jpg
2074  892420643555336193    https://pbs.twimg.com/media/DGKD1-bXoAAIAUK.jpg

      img_num                           p1   p1_conf  p1_dog  \
0           1       Welsh_springer_spaniel  0.465074    True
1           1                      redbone  0.506826    True
2           1              German_shepherd  0.596461    True
3           1          Rhodesian_ridgeback  0.408143    True
4           1           miniature_pinscher  0.560311    True
5           1         Bernese_mountain_dog  0.651137    True
6           1                   box_turtle  0.933012   False
7           1                         chow  0.692517    True
8           1                shopping_cart  0.962465   False
9           1             miniature_poodle  0.201493    True
10          1             golden_retriever  0.775930    True
11          1                Gordon_setter  0.503672    True
12          1                 Walker_hound  0.260857    True
```

```
13          1                          pug  0.489814   True
14          1                   bloodhound  0.195217   True
15          1                        Lhasa  0.582330   True
16          1               English_setter  0.298617   True
17          1                          hen  0.965932  False
18          1             desktop_computer  0.086502  False
19          1             Italian_greyhound  0.176053   True
20          1                  Maltese_dog  0.857531   True
21          1             three-toed_sloth  0.914671  False
22          1                           ox  0.416669  False
23          1             golden_retriever  0.858744   True
24          1                     malamute  0.336874   True
25          1                   guinea_pig  0.996496  False
26          1  soft-coated_wheaten_terrier  0.326467   True
27          1                    Chihuahua  0.978108   True
28          1      black-and-tan_coonhound  0.529139   True
29          1                         coho  0.404640  False
...       ...                          ...       ...    ...
2045        1               French_bulldog  0.999201   True
2046        1                  convertible  0.738995  False
2047        1                       kuvasz  0.309706   True
2048        2                    Chihuahua  0.793469   True
2049        1                      Samoyed  0.733942   True
2050        1             Mexican_hairless  0.330741   True
2051        2                     Pembroke  0.809197   True
2052        1                    limousine  0.130432  False
2053        1                       basset  0.821664   True
2054        1               French_bulldog  0.995026   True
2055        2                     Pembroke  0.809197   True
2056        3               Siberian_husky  0.700377   True
2057        1             golden_retriever  0.469760   True
2058        1             golden_retriever  0.714719   True
2059        1                      whippet  0.626152   True
2060        1             golden_retriever  0.953442   True
2061        1               French_bulldog  0.991650   True
2062        1                     Pembroke  0.966327   True
2063        1               French_bulldog  0.377417   True
2064        1                      Samoyed  0.957979   True
2065        1                     Pembroke  0.511319   True
2066        1                Irish_terrier  0.487574   True
2067        2                   Pomeranian  0.566142   True
2068        1                  Appenzeller  0.341703   True
2069        1     Chesapeake_Bay_retriever  0.425595   True
2070        2                       basset  0.555712   True
2071        1                  paper_towel  0.170278  False
2072        1                    Chihuahua  0.716012   True
2073        1                    Chihuahua  0.323581   True
2074        1                       orange  0.097049  False
```

```
                            p2   p2_conf  p2_dog                           p3  \
0                       collie  0.156665    True            Shetland_sheepdog
1           miniature_pinscher  0.074192    True          Rhodesian_ridgeback
2                     malinois  0.138584    True                   bloodhound
3                      redbone  0.360687    True           miniature_pinscher
4                   Rottweiler  0.243682    True                     Doberman
5             English_springer  0.263788    True   Greater_Swiss_Mountain_dog
6                   mud_turtle  0.045885   False                     terrapin
7              Tibetan_mastiff  0.058279    True                     fur_coat
8              shopping_basket  0.014594   False             golden_retriever
9                     komondor  0.192305    True  soft-coated_wheaten_terrier
10             Tibetan_mastiff  0.093718    True           Labrador_retriever
11           Yorkshire_terrier  0.174201    True                     Pekinese
12            English_foxhound  0.175382    True                 Ibizan_hound
13                bull_mastiff  0.404722    True               French_bulldog
14             German_shepherd  0.078260    True                     malinois
15                    Shih-Tzu  0.166192    True               Dandie_Dinmont
16                Newfoundland  0.149842    True                       borzoi
17                        cock  0.033919   False                    partridge
18                        desk  0.085547   False                     bookcase
19                 toy_terrier  0.111884    True                      basenji
20                  toy_poodle  0.063064    True             miniature_poodle
21                       otter  0.015250   False               great_grey_owl
22                Newfoundland  0.278407    True                  groenendael
23    Chesapeake_Bay_retriever  0.054787    True           Labrador_retriever
24              Siberian_husky  0.147655    True                   Eskimo_dog
25                       skunk  0.002402   False                      hamster
26                Afghan_hound  0.259551    True                       briard
27                 toy_terrier  0.009397    True                     papillon
28                  bloodhound  0.244220    True        flat-coated_retriever
29                  barracouta  0.271485   False                          gar
...                        ...       ...     ...                          ...
2045                 Chihuahua  0.000361    True                  Boston_bull
2046                sports_car  0.139952   False                    car_wheel
2047            Great_Pyrenees  0.186136    True               Dandie_Dinmont
2048               toy_terrier  0.143528    True                   can_opener
2049                Eskimo_dog  0.035029    True    Staffordshire_bullterrier
2050                  sea_lion  0.275645   False                   Weimaraner
2051       Rhodesian_ridgeback  0.054950    True                       beagle
2052                 tow_truck  0.029175   False                shopping_cart
2053                   redbone  0.087582    True                   Weimaraner
2054                       pug  0.000932    True                 bull_mastiff
2055       Rhodesian_ridgeback  0.054950    True                       beagle
2056                Eskimo_dog  0.166511    True                     malamute
2057        Labrador_retriever  0.184172    True               English_setter
2058           Tibetan_mastiff  0.120184    True           Labrador_retriever
2059                    borzoi  0.194742    True                       Saluki
```

```
2060        Labrador_retriever  0.013834    True                      redbone
2061                     boxer  0.002129    True    Staffordshire_bullterrier
2062                  Cardigan  0.027356    True                      basenji
2063        Labrador_retriever  0.151317    True                       muzzle
2064                Pomeranian  0.013884    True                         chow
2065                  Cardigan  0.451038    True                    Chihuahua
2066              Irish_setter  0.193054    True     Chesapeake_Bay_retriever
2067                Eskimo_dog  0.178406    True                     Pembroke
2068             Border_collie  0.199287    True                    ice_lolly
2069             Irish_terrier  0.116317    True              Indian_elephant
2070          English_springer  0.225770    True  German_short-haired_pointer
2071        Labrador_retriever  0.168086    True                      spatula
2072                  malamute  0.078253    True                       kelpie
2073                  Pekinese  0.090647    True                     papillon
2074                     bagel  0.085851   False                       banana

       p3_conf  p3_dog
0     0.061428    True
1     0.072010    True
2     0.116197    True
3     0.222752    True
4     0.154629    True
5     0.016199    True
6     0.017885   False
7     0.054449   False
8     0.007959    True
9     0.082086    True
10    0.072427    True
11    0.109454    True
12    0.097471    True
13    0.048960    True
14    0.075628    True
15    0.089688    True
16    0.133649    True
17    0.000052   False
18    0.079480   False
19    0.111152    True
20    0.025581    True
21    0.013207   False
22    0.102643    True
23    0.014241    True
24    0.093412    True
25    0.000461   False
26    0.206803    True
27    0.004577    True
28    0.173810    True
29    0.189945   False
...        ...     ...
```

```
         2045  0.000076   True
         2046  0.044173  False
         2047  0.086346   True
         2048  0.032253  False
         2049  0.029705   True
         2050  0.134203   True
         2051  0.038915   True
         2052  0.026321  False
         2053  0.026236   True
         2054  0.000903   True
         2055  0.038915   True
         2056  0.111411   True
         2057  0.073482   True
         2058  0.105506   True
         2059  0.027351   True
         2060  0.007958   True
         2061  0.001498   True
         2062  0.004633   True
         2063  0.082981  False
         2064  0.008167   True
         2065  0.029248   True
         2066  0.118184   True
         2067  0.076507   True
         2068  0.193548  False
         2069  0.076902  False
         2070  0.175219   True
         2071  0.040836  False
         2072  0.031379   True
         2073  0.068957   True
         2074  0.076110  False

         [2075 rows x 12 columns]
```

```
In [20]: ip.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2075 entries, 0 to 2074
Data columns (total 12 columns):
tweet_id    2075 non-null int64
jpg_url     2075 non-null object
img_num     2075 non-null int64
p1          2075 non-null object
p1_conf     2075 non-null float64
p1_dog      2075 non-null bool
p2          2075 non-null object
p2_conf     2075 non-null float64
p2_dog      2075 non-null bool
p3          2075 non-null object
```

```
p3_conf     2075 non-null float64
p3_dog      2075 non-null bool
dtypes: bool(3), float64(3), int64(2), object(4)
memory usage: 152.1+ KB
```

```
In [21]: ip.describe()
```

```
Out[21]:            tweet_id      img_num      p1_conf       p2_conf       p3_conf
         count  2.075000e+03  2075.000000  2075.000000  2.075000e+03  2.075000e+03
         mean   7.384514e+17     1.203855     0.594548  1.345886e-01  6.032417e-02
         std    6.785203e+16     0.561875     0.271174  1.006657e-01  5.090593e-02
         min    6.660209e+17     1.000000     0.044333  1.011300e-08  1.740170e-10
         25%    6.764835e+17     1.000000     0.364412  5.388625e-02  1.622240e-02
         50%    7.119988e+17     1.000000     0.588230  1.181810e-01  4.944380e-02
         75%    7.932034e+17     1.000000     0.843855  1.955655e-01  9.180755e-02
         max    8.924206e+17     4.000000     1.000000  4.880140e-01  2.734190e-01
```

```
In [22]: sum(ip.tweet_id.duplicated())
```

```
Out[22]: 0
```

### 1.1.5 Quality Assessment of "ip":

1. It has mix of both proper case and lower case names in p1, p2, and p3
2. I can see outliers in img_num, p2_conf and p3_conf columns
3. Number of rows (#2075) of "ip" dataset are lesser than than "tae" dataset (#2356)

### 1.1.6 Tideness Assessment of "ip":

- img_num is not a useful column in our analysis

#### Assessment of df_tweets:

```
In [23]: #To assess the df_tweets data manually
         df_tweets
```

```
Out[23]:                      id  retweet_count  favorite_count
         0    892420643555336193           8332           38117
         1    892177421306343426           6158           32704
         2    891815181378084864           4075           24614
         3    891689557279858688           8477           41470
         4    891327558926688256           9169           39638
         5    891087950875897856           3053           19906
         6    890971913173991426           2025           11632
         7    890729181411237888          18498           64302
         8    890609185150312448           4193           27349
         9    890240255349198849           7234           31374
         10   890006608113172480           7190           30160
```

| | | | |
|---|---|---|---|
| 11 | 889880896479866881 | 4879 | 27327 |
| 12 | 889665388333682689 | 9863 | 47307 |
| 13 | 889638837579907072 | 4451 | 26705 |
| 14 | 889531135344209921 | 2202 | 14860 |
| 15 | 889278841981685760 | 5267 | 24820 |
| 16 | 888917238123831296 | 4408 | 28621 |
| 17 | 888804989199671297 | 4205 | 25127 |
| 18 | 888554962724278272 | 3480 | 19498 |
| 19 | 888078434458587136 | 3423 | 21398 |
| 20 | 887705289381826560 | 5281 | 29671 |
| 21 | 887517139158093824 | 11453 | 45509 |
| 22 | 887473957103951883 | 17824 | 67917 |
| 23 | 887343217045368832 | 10201 | 33121 |
| 24 | 887101392804085760 | 5850 | 30075 |
| 25 | 886983233522544640 | 7625 | 34588 |
| 26 | 886736880519319552 | 3220 | 11856 |
| 27 | 886680336477933568 | 4376 | 22059 |
| 28 | 886366144734445568 | 3139 | 20836 |
| 29 | 886267009285017600 | 4 | 115 |
| .. | ... | ... | ... |
| 579 | 799063482566066176 | 2711 | 8696 |
| 580 | 798933969379225600 | 4931 | 14123 |
| 581 | 798925684722855936 | 1581 | 7989 |
| 582 | 798705661114773508 | 7270 | 0 |
| 583 | 798701998996647937 | 8591 | 0 |
| 584 | 798697898615730177 | 7168 | 0 |
| 585 | 798694562394996736 | 5460 | 0 |
| 586 | 798686750113755136 | 2560 | 0 |
| 587 | 798682547630837760 | 5196 | 0 |
| 588 | 798673117451325440 | 6104 | 0 |
| 589 | 798665375516884993 | 4281 | 0 |
| 590 | 798644042770751489 | 2036 | 0 |
| 591 | 798628517273620480 | 2165 | 0 |
| 592 | 798585098161549313 | 6276 | 0 |
| 593 | 798576900688019456 | 6455 | 0 |
| 594 | 798340744599797760 | 3700 | 0 |
| 595 | 798209839306514432 | 2817 | 11141 |
| 596 | 797971864723324932 | 3457 | 12508 |
| 597 | 797545162159308800 | 5351 | 15569 |
| 598 | 797236660651966464 | 7300 | 21452 |
| 599 | 797165961484890113 | 29 | 248 |
| 600 | 796904159865868288 | 9776 | 0 |
| 601 | 796865951799083009 | 2096 | 8246 |
| 602 | 796759840936919040 | 3364 | 12751 |
| 603 | 796563435802726400 | 8036 | 0 |
| 604 | 796484825502875648 | 1942 | 8140 |
| 605 | 796387464403357696 | 4588 | 11815 |
| 606 | 796177847564038144 | 15776 | 0 |

```
         607  796149749086875649          15776           34757
         608  796125600683540480           1966            5299

         [609 rows x 3 columns]
```

```
In [24]: df_tweets.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 609 entries, 0 to 608
Data columns (total 3 columns):
id                  609 non-null int64
retweet_count       609 non-null int64
favorite_count      609 non-null int64
dtypes: int64(3)
memory usage: 14.4 KB
```

```
In [25]: df_tweets.describe()
```

```
Out[25]:                   id  retweet_count  favorite_count
         count  6.090000e+02     609.000000      609.000000
         mean   8.393464e+17    5522.880131    16873.231527
         std    2.733673e+16    6256.559767    17103.907267
         min    7.961256e+17       0.000000        0.000000
         25%    8.171713e+17    2261.000000     6498.000000
         50%    8.352464e+17    3752.000000    13763.000000
         75%    8.610051e+17    6311.000000    23054.000000
         max    8.924206e+17   61082.000000   140666.000000
```

```
In [26]: sum(df_tweets.duplicated())
```

```
Out[26]: 0
```

### 1.1.7 Quality Assessment of "df_tweets":

1. There are less records in the "df_tweets" dataset than "tae" datasets(#2356)
2. "id" column name is not consistent with other two sources

### 1.1.8 Tidiness Assessment of "df_tweets":

- There are no tidiness issues in this dataset.

```
In [27]: df_tweets.rename(columns={'id':'tweet_id'}, inplace=True)
```

### 1.1.9 II) Clean

Cleaning the data based on the assessment

```
In [28]: # Creating a copy for cleaning the data
         tae_copy=tae.copy()
         ip_copy=ip.copy()
         df_tweets_copy=df_tweets.copy()
```

### 1.1.10 Quality: tae_copy

**1. Define** *Since the Timestamp column is in object format, changing it to datetime format.*

**1. Code**

```
In [29]: tae_copy['timestamp'] = pd.to_datetime(tae_copy['timestamp'])
```

**1. Test**

```
In [30]: tae_copy.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2356 entries, 0 to 2355
Data columns (total 17 columns):
tweet_id                      2356 non-null int64
in_reply_to_status_id         78 non-null float64
in_reply_to_user_id           78 non-null float64
timestamp                     2356 non-null datetime64[ns]
source                        2356 non-null object
text                          2356 non-null object
retweeted_status_id           181 non-null float64
retweeted_status_user_id      181 non-null float64
retweeted_status_timestamp    181 non-null object
expanded_urls                 2297 non-null object
rating_numerator              2356 non-null int64
rating_denominator            2356 non-null int64
name                          2356 non-null object
doggo                         2356 non-null object
floofer                       2356 non-null object
pupper                        2356 non-null object
puppo                         2356 non-null object
dtypes: datetime64[ns](1), float64(4), int64(3), object(9)
memory usage: 313.0+ KB
```

**2. Define** *Unnecessary information is present in the text column along with the hastag. Therefore, creating Hashtag column from the text column.*

**2. Code**

```
In [31]: tae_copy['hashtag'] = tae_copy['text'].str.extract(r"#(\w+)", expand=True)
```

**2. Test**

```
In [32]: tae_copy['hashtag'].value_counts()
```

```
Out[32]: BarkWeek                 9
         PrideMonth               3
         notallpuppers            1
         dogsatpollingstations    1
         FinalFur                 1
         K9VeteransDay            1
         WKCDogShow               1
         ScienceMarch             1
         Canada150                1
         ImWithThor               1
         WomensMarch              1
         GoodDogs                 1
         NoDaysOff                1
         PrideMonthPuppo          1
         BATP                     1
         LoveTwitter              1
         BellLetsTalk             1
         Name: hashtag, dtype: int64
```

**3. Define** *Since rating numerator is greater than rating denominator, I am changing rating numerator with rating denominator value. This is because if numerator is greater then it could be a full rating.*

**3. Code**

```
In [33]: #replacing the value greater 10 as 10
         def get_name(tae_copy):
             if tae_copy['rating_denominator'] > 10:
                 return tae_copy['rating_denominator']==10
             else:
                 return tae_copy['rating_denominator']
         tae_copy['rating_denominator'] = tae_copy.apply(get_name, axis = 1)
```

**3. Test**

```
In [34]: #To check if there is a True
         (tae_copy['rating_denominator']>10).value_counts()

Out[34]: False    2356
         Name: rating_denominator, dtype: int64
```

**4. Define** *If Rating Denominator is greater than 10 then create value as 10.*

**4. Code**

```
In [35]: #replacing the value if rating_numeratior is greater than rating_denominator as rating_
         def get_name(tae_copy):
             if tae_copy['rating_numerator'] > tae_copy['rating_denominator']:
                 return tae_copy['rating_denominator']
```

```
            else:
                return tae_copy['rating_numerator']
        tae_copy['rating_numerator'] = tae_copy.apply(get_name, axis = 1)

        # convert it to int
        tae_copy['rating_numerator']=pd.to_numeric(tae_copy['rating_numerator'])
        tae_copy['rating_denominator']=pd.to_numeric(tae_copy['rating_denominator'])
```

**4. Test**

```
In [36]: #To check if there is a True
         (tae_copy['rating_numerator'] > tae_copy['rating_denominator']).value_counts()
```

```
Out[36]: False    2356
         dtype: int64
```

```
In [37]: tae_copy.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2356 entries, 0 to 2355
Data columns (total 18 columns):
tweet_id                      2356 non-null int64
in_reply_to_status_id         78 non-null float64
in_reply_to_user_id           78 non-null float64
timestamp                     2356 non-null datetime64[ns]
source                        2356 non-null object
text                          2356 non-null object
retweeted_status_id           181 non-null float64
retweeted_status_user_id      181 non-null float64
retweeted_status_timestamp    181 non-null object
expanded_urls                 2297 non-null object
rating_numerator              2356 non-null int64
rating_denominator            2356 non-null int64
name                          2356 non-null object
doggo                         2356 non-null object
floofer                       2356 non-null object
pupper                        2356 non-null object
puppo                         2356 non-null object
hashtag                       27 non-null object
dtypes: datetime64[ns](1), float64(4), int64(3), object(10)
memory usage: 331.4+ KB
```

**5. Define** *Names are in lower case are not seems like a name. Hence, replcing all lowercase values with "None"*

**5. Code**

```
In [38]: def get_name(tae_copy):
             if tae_copy['name'] in lower:
                 return
             else:
                 return tae_copy['name']

         tae_copy['name'] = tae_copy.apply(get_name, axis = 1)
         tae_copy.replace({'name':{0: "None"}},inplace=True)
```

### 5. Test

```
In [39]: tae_copy.name.value_counts()
```

```
Out[39]: None         745
         Charlie       12
         Cooper        11
         Oliver        11
         Lucy          11
         Penny         10
         Tucker        10
         Lola          10
         Winston        9
         Bo             9
         Sadie          8
         Bailey         7
         Toby           7
         Buddy          7
         Daisy          7
         Bella          6
         Jack           6
         Milo           6
         Oscar          6
         Dave           6
         Jax            6
         Scout          6
         Stanley        6
         Rusty          6
         Koda           6
         Leo            6
         Larry          5
         Finn           5
         Phil           5
         Gus            5
                     ...
         Maude          1
         Luther         1
         Stormy         1
         Berb           1
```

```
Dobby          1
Callie         1
Jazz           1
Mookie         1
Remy           1
Shiloh         1
Joshwa         1
Tonks          1
Skye           1
Mairi          1
Binky          1
Jo             1
Billy          1
Bayley         1
Blakely        1
Kevon          1
Shnuggles      1
Bertson        1
Jazzy          1
Skittle        1
Diogi          1
Gordon         1
Al             1
Snicku         1
Ralf           1
Fabio          1
Name: name, Length: 932, dtype: int64
```

### 1.1.11 Tidiness: tae_copy

**1. Define** *Since there are multiple columns (”doggo”,”floofer”,”pupper”,”puppo”) of Dog types, creating a single column as dog_type.*

**1. Code**

```
In [40]: def get_name(tae_copy):
             if tae_copy['doggo'] == "doggo" and tae_copy['floofer'] == "floofer":
                 return "doggo & floofer"
             elif tae_copy['doggo'] == "doggo" and tae_copy['pupper'] == "pupper":
                 return "doggo & pupper"
             elif tae_copy['doggo'] == "doggo" and tae_copy['puppo'] == "puppo":
                 return "doggo & puppo"
             elif tae_copy['doggo'] != "None":
                 return tae_copy['doggo']
             elif tae_copy['floofer'] != "None":
                 return tae_copy['floofer']
             elif tae_copy['pupper'] != "None":
                 return tae_copy['pupper']
```

```
    elif tae_copy['puppo'] != "None":
        return tae_copy['puppo']
tae_copy['dog_type'] = tae_copy.apply(get_name, axis = 1)
```

**1. Test**

```
In [41]: tae_copy.dog_type.value_counts()
```

```
Out[41]: pupper             245
         doggo               83
         puppo               29
         doggo & pupper      12
         floofer              9
         doggo & floofer      1
         doggo & puppo        1
         Name: dog_type, dtype: int64
```

**2. Define** *Dropping unnecessary columns such as "retweeted_status_id", "retweeted_status_user_id", "retweeted_status_timestamp", "in_reply_to_status_id", "in_reply_to_user_id" and "expanded_urls.*

**2. Code**

```
In [42]: tae_copy=tae_copy.drop(["retweeted_status_id", "retweeted_status_user_id", "retweeted_s
                                "in_reply_to_status_id", "in_reply_to_user_id", "expanded_urls"]
```

**2. Test**

```
In [43]: tae_copy.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2356 entries, 0 to 2355
Data columns (total 13 columns):
tweet_id              2356 non-null int64
timestamp             2356 non-null datetime64[ns]
source                2356 non-null object
text                  2356 non-null object
rating_numerator      2356 non-null int64
rating_denominator    2356 non-null int64
name                  2247 non-null object
doggo                 2356 non-null object
floofer               2356 non-null object
pupper                2356 non-null object
puppo                 2356 non-null object
hashtag               27 non-null object
dog_type              380 non-null object
dtypes: datetime64[ns](1), int64(3), object(9)
memory usage: 239.4+ KB
```

### 1.1.12 Quality: ip_copy

**1. Define** *Since it has mix of both proper case and lower case names in p1, p2, and p3, making all the starting letter to propcase.*

**1. Code**

```
In [44]: ip_copy['p1'] = ip_copy.p1.str.title()
         ip_copy['p2'] = ip_copy.p2.str.title()
         ip_copy['p3'] = ip_copy.p3.str.title()
```

**1. Test**

```
In [45]: ip_copy
```

```
Out[45]:                 tweet_id                                          jpg_url  \
         0     666020888022790149  https://pbs.twimg.com/media/CT4udn0WwAA0aMy.jpg
         1     666029285002620928  https://pbs.twimg.com/media/CT42GRgUYAA5iDo.jpg
         2     666033412701032449  https://pbs.twimg.com/media/CT4521TWwAEvMyu.jpg
         3     666044226329800704  https://pbs.twimg.com/media/CT5Dr8HUEAA-lEu.jpg
         4     666049248165822465  https://pbs.twimg.com/media/CT5IQmsXIAAKY4A.jpg
         5     666050758794694657  https://pbs.twimg.com/media/CT5Jof1WUAEuVxN.jpg
         6     666051853826850816  https://pbs.twimg.com/media/CT5KoJ1WoAAJash.jpg
         7     666055525042405380  https://pbs.twimg.com/media/CT5N9tpXIAAifs1.jpg
         8     666057090499244032  https://pbs.twimg.com/media/CT5PY90WoAAQGLo.jpg
         9     666058600524156928  https://pbs.twimg.com/media/CT5Qw94XAAA_2dP.jpg
         10    666063827256086533  https://pbs.twimg.com/media/CT5Vg_wXIAAXfnj.jpg
         11    666071193221509120  https://pbs.twimg.com/media/CT5cN_3WEAAlOoZ.jpg
         12    666073100786774016  https://pbs.twimg.com/media/CT5d9DZXAAALcwe.jpg
         13    666082916733198337  https://pbs.twimg.com/media/CT5m4VGWEAAtKc8.jpg
         14    666094000022159362  https://pbs.twimg.com/media/CT5w9gUW4AAsBNN.jpg
         15    666099513787052032  https://pbs.twimg.com/media/CT51-JJUEAA6hV8.jpg
         16    666102155909144576  https://pbs.twimg.com/media/CT54YGiWUAEZnoK.jpg
         17    666104133288665088  https://pbs.twimg.com/media/CT56LSZWoAAlJj2.jpg
         18    666268910803644416  https://pbs.twimg.com/media/CT8QCd1WEAADXws.jpg
         19    666273097616637952  https://pbs.twimg.com/media/CT8T1mtUwAA3aqm.jpg
         20    666287406224695296  https://pbs.twimg.com/media/CT8g3BpUEAAuFjg.jpg
         21    666293911632134144  https://pbs.twimg.com/media/CT8mx7KW4AEQu8N.jpg
         22    666337882303524864  https://pbs.twimg.com/media/CT9OwFIWEAMuRje.jpg
         23    666345417576210432  https://pbs.twimg.com/media/CT9Vn7PWoAA_ZCM.jpg
         24    666353288456101888  https://pbs.twimg.com/media/CT9cx0tUEAAhNN_.jpg
         25    666362758909284353  https://pbs.twimg.com/media/CT9lXGsUcAAyUFt.jpg
         26    666373753744588802  https://pbs.twimg.com/media/CT9vZEYWUAAlZ05.jpg
         27    666396247373291520  https://pbs.twimg.com/media/CT-D2ZHWIAA3gK1.jpg
         28    666407126856765440  https://pbs.twimg.com/media/CT-NvwmW4AAugGZ.jpg
         29    666411507551481857  https://pbs.twimg.com/media/CT-RugiWIAELEaq.jpg
         ...                  ...                                              ...
         2045  886366144734445568  https://pbs.twimg.com/media/DE0BTnQUwAApKEH.jpg
         2046  886680336477933568  https://pbs.twimg.com/media/DE4fEDzWAAAyHMM.jpg
```

```
2047  886736880519319552    https://pbs.twimg.com/media/DE5Se8FXcAAJFx4.jpg
2048  886983233522544640    https://pbs.twimg.com/media/DE8yicJW0AAAvBJ.jpg
2049  887101392804085760    https://pbs.twimg.com/media/DE-eAq6UwAA-jaE.jpg
2050  887343217045368832  https://pbs.twimg.com/ext_tw_video_thumb/88734...
2051  887473957103951883    https://pbs.twimg.com/media/DFDw2tyUQAAAFke.jpg
2052  887517139158093824  https://pbs.twimg.com/ext_tw_video_thumb/88751...
2053  887705289381826560    https://pbs.twimg.com/media/DFHDQBbXgAEqY7t.jpg
2054  888078434458587136    https://pbs.twimg.com/media/DFMWn56WsAAkA7B.jpg
2055  888202515573088257    https://pbs.twimg.com/media/DFDw2tyUQAAAFke.jpg
2056  888554962724278272    https://pbs.twimg.com/media/DFTH_O-UQAACu20.jpg
2057  888804989199671297    https://pbs.twimg.com/media/DFWra-3VYAA2piG.jpg
2058  888917238123831296    https://pbs.twimg.com/media/DFYRgsOUQAARGhO.jpg
2059  889278841981685760  https://pbs.twimg.com/ext_tw_video_thumb/88927...
2060  889531135344209921    https://pbs.twimg.com/media/DFg_2PVW0AEHN3p.jpg
2061  889638837579907072    https://pbs.twimg.com/media/DFihzFfXsAYGDPR.jpg
2062  889665388333682689    https://pbs.twimg.com/media/DFi579UWsAAatzw.jpg
2063  889880896479866881    https://pbs.twimg.com/media/DFl99B1WsAITKsg.jpg
2064  890006608113172480    https://pbs.twimg.com/media/DFnwSY4WAAAMliS.jpg
2065  890240255349198849    https://pbs.twimg.com/media/DFrEyVuW0AAO3t9.jpg
2066  890609185150312448    https://pbs.twimg.com/media/DFwUU__XcAEpyXI.jpg
2067  890729181411237888    https://pbs.twimg.com/media/DFyBahAVwAAhUTd.jpg
2068  890971913173991426    https://pbs.twimg.com/media/DF1eOmZXUAALUcq.jpg
2069  891087950875897856    https://pbs.twimg.com/media/DF3HwyEWsAABqE6.jpg
2070  891327558926688256    https://pbs.twimg.com/media/DF6hr6BUMAAzZgT.jpg
2071  891689557279858688    https://pbs.twimg.com/media/DF_q7IAWsAEuuN8.jpg
2072  891815181378084864    https://pbs.twimg.com/media/DGBdLU1WsAANxJ9.jpg
2073  892177421306343426    https://pbs.twimg.com/media/DGGmoV4XsAAUL6n.jpg
2074  892420643555336193    https://pbs.twimg.com/media/DGKD1-bXoAAIAUK.jpg

      img_num                           p1   p1_conf   p1_dog  \
0           1       Welsh_Springer_Spaniel  0.465074     True
1           1                      Redbone  0.506826     True
2           1              German_Shepherd  0.596461     True
3           1          Rhodesian_Ridgeback  0.408143     True
4           1           Miniature_Pinscher  0.560311     True
5           1         Bernese_Mountain_Dog  0.651137     True
6           1                   Box_Turtle  0.933012    False
7           1                         Chow  0.692517     True
8           1                Shopping_Cart  0.962465    False
9           1             Miniature_Poodle  0.201493     True
10          1             Golden_Retriever  0.775930     True
11          1                Gordon_Setter  0.503672     True
12          1                 Walker_Hound  0.260857     True
13          1                          Pug  0.489814     True
14          1                   Bloodhound  0.195217     True
15          1                        Lhasa  0.582330     True
16          1               English_Setter  0.298617     True
17          1                          Hen  0.965932    False
```

```
18          1             Desktop_Computer  0.086502   False
19          1            Italian_Greyhound  0.176053    True
20          1                  Maltese_Dog  0.857531    True
21          1             Three-Toed_Sloth  0.914671   False
22          1                           Ox  0.416669   False
23          1             Golden_Retriever  0.858744    True
24          1                     Malamute  0.336874    True
25          1                   Guinea_Pig  0.996496   False
26          1  Soft-Coated_Wheaten_Terrier  0.326467    True
27          1                    Chihuahua  0.978108    True
28          1      Black-And-Tan_Coonhound  0.529139    True
29          1                         Coho  0.404640   False
...       ...                          ...       ...     ...
2045        1               French_Bulldog  0.999201    True
2046        1                  Convertible  0.738995   False
2047        1                       Kuvasz  0.309706    True
2048        2                    Chihuahua  0.793469    True
2049        1                      Samoyed  0.733942    True
2050        1             Mexican_Hairless  0.330741    True
2051        2                     Pembroke  0.809197    True
2052        1                    Limousine  0.130432   False
2053        1                       Basset  0.821664    True
2054        1               French_Bulldog  0.995026    True
2055        2                     Pembroke  0.809197    True
2056        3               Siberian_Husky  0.700377    True
2057        1             Golden_Retriever  0.469760    True
2058        1             Golden_Retriever  0.714719    True
2059        1                      Whippet  0.626152    True
2060        1             Golden_Retriever  0.953442    True
2061        1               French_Bulldog  0.991650    True
2062        1                     Pembroke  0.966327    True
2063        1               French_Bulldog  0.377417    True
2064        1                      Samoyed  0.957979    True
2065        1                     Pembroke  0.511319    True
2066        1                Irish_Terrier  0.487574    True
2067        2                   Pomeranian  0.566142    True
2068        1                  Appenzeller  0.341703    True
2069        1     Chesapeake_Bay_Retriever  0.425595    True
2070        2                       Basset  0.555712    True
2071        1                  Paper_Towel  0.170278   False
2072        1                    Chihuahua  0.716012    True
2073        1                    Chihuahua  0.323581    True
2074        1                       Orange  0.097049   False

                            p2   p2_conf  p2_dog                           p3  \
0                       Collie  0.156665    True            Shetland_Sheepdog
1           Miniature_Pinscher  0.074192    True          Rhodesian_Ridgeback
2                     Malinois  0.138584    True                   Bloodhound
```

```
3                      Redbone  0.360687   True           Miniature_Pinscher
4                   Rottweiler  0.243682   True                     Doberman
5             English_Springer  0.263788   True   Greater_Swiss_Mountain_Dog
6                   Mud_Turtle  0.045885  False                     Terrapin
7              Tibetan_Mastiff  0.058279   True                     Fur_Coat
8              Shopping_Basket  0.014594  False             Golden_Retriever
9                     Komondor  0.192305   True  Soft-Coated_Wheaten_Terrier
10             Tibetan_Mastiff  0.093718   True           Labrador_Retriever
11           Yorkshire_Terrier  0.174201   True                     Pekinese
12            English_Foxhound  0.175382   True                 Ibizan_Hound
13                Bull_Mastiff  0.404722   True               French_Bulldog
14             German_Shepherd  0.078260   True                     Malinois
15                    Shih-Tzu  0.166192   True               Dandie_Dinmont
16                Newfoundland  0.149842   True                       Borzoi
17                        Cock  0.033919  False                    Partridge
18                        Desk  0.085547  False                     Bookcase
19                 Toy_Terrier  0.111884   True                      Basenji
20                  Toy_Poodle  0.063064   True             Miniature_Poodle
21                       Otter  0.015250  False               Great_Grey_Owl
22                Newfoundland  0.278407   True                  Groenendael
23    Chesapeake_Bay_Retriever  0.054787   True           Labrador_Retriever
24              Siberian_Husky  0.147655   True                   Eskimo_Dog
25                       Skunk  0.002402  False                      Hamster
26                Afghan_Hound  0.259551   True                       Briard
27                 Toy_Terrier  0.009397   True                     Papillon
28                  Bloodhound  0.244220   True        Flat-Coated_Retriever
29                  Barracouta  0.271485  False                          Gar
...                        ...       ...    ...                          ...
2045                 Chihuahua  0.000361   True                  Boston_Bull
2046                Sports_Car  0.139952  False                    Car_Wheel
2047            Great_Pyrenees  0.186136   True               Dandie_Dinmont
2048               Toy_Terrier  0.143528   True                   Can_Opener
2049                Eskimo_Dog  0.035029   True    Staffordshire_Bullterrier
2050                  Sea_Lion  0.275645  False                   Weimaraner
2051       Rhodesian_Ridgeback  0.054950   True                       Beagle
2052                 Tow_Truck  0.029175  False                Shopping_Cart
2053                   Redbone  0.087582   True                   Weimaraner
2054                       Pug  0.000932   True                 Bull_Mastiff
2055       Rhodesian_Ridgeback  0.054950   True                       Beagle
2056                Eskimo_Dog  0.166511   True                     Malamute
2057        Labrador_Retriever  0.184172   True               English_Setter
2058           Tibetan_Mastiff  0.120184   True           Labrador_Retriever
2059                    Borzoi  0.194742   True                       Saluki
2060        Labrador_Retriever  0.013834   True                      Redbone
2061                     Boxer  0.002129   True    Staffordshire_Bullterrier
2062                  Cardigan  0.027356   True                      Basenji
2063        Labrador_Retriever  0.151317   True                       Muzzle
2064                Pomeranian  0.013884   True                         Chow
```

```
2065                Cardigan  0.451038     True                   Chihuahua
2066            Irish_Setter  0.193054     True    Chesapeake_Bay_Retriever
2067              Eskimo_Dog  0.178406     True                    Pembroke
2068           Border_Collie  0.199287     True                   Ice_Lolly
2069           Irish_Terrier  0.116317     True             Indian_Elephant
2070        English_Springer  0.225770     True  German_Short-Haired_Pointer
2071      Labrador_Retriever  0.168086     True                     Spatula
2072                Malamute  0.078253     True                      Kelpie
2073                Pekinese  0.090647     True                    Papillon
2074                   Bagel  0.085851    False                      Banana

       p3_conf  p3_dog
0     0.061428    True
1     0.072010    True
2     0.116197    True
3     0.222752    True
4     0.154629    True
5     0.016199    True
6     0.017885   False
7     0.054449   False
8     0.007959    True
9     0.082086    True
10    0.072427    True
11    0.109454    True
12    0.097471    True
13    0.048960    True
14    0.075628    True
15    0.089688    True
16    0.133649    True
17    0.000052   False
18    0.079480   False
19    0.111152    True
20    0.025581    True
21    0.013207   False
22    0.102643    True
23    0.014241    True
24    0.093412    True
25    0.000461   False
26    0.206803    True
27    0.004577    True
28    0.173810    True
29    0.189945   False
...        ...     ...
2045  0.000076    True
2046  0.044173   False
2047  0.086346    True
2048  0.032253   False
2049  0.029705    True
```

```
2050  0.134203   True
2051  0.038915   True
2052  0.026321  False
2053  0.026236   True
2054  0.000903   True
2055  0.038915   True
2056  0.111411   True
2057  0.073482   True
2058  0.105506   True
2059  0.027351   True
2060  0.007958   True
2061  0.001498   True
2062  0.004633   True
2063  0.082981  False
2064  0.008167   True
2065  0.029248   True
2066  0.118184   True
2067  0.076507   True
2068  0.193548  False
2069  0.076902  False
2070  0.175219   True
2071  0.040836  False
2072  0.031379   True
2073  0.068957   True
2074  0.076110  False

[2075 rows x 12 columns]
```

**2. Define** *I can see outliers in img_num, p2_conf and p3_conf columns. However, we are not cleaning it because we are not using this columns for our analysis.*

**3. Define** *Number of rows (#2075) of "ip" dataset are lesser than than "tae" dataset (#2356). However, we cannot do anything on it.*

### 1.1.13 Tidiness: ip_copy

**1. Define** *Since "img_num" column is not a useful in our analysis, dropping this column.*

**1. Code**

```
In [46]: ip_copy=ip_copy.drop(["img_num"],axis=1)
```

**1. Test**

```
In [47]: ip_copy.head(2)
```

```
Out[47]:              tweet_id                                          jpg_url  \
         0  666020888022790149  https://pbs.twimg.com/media/CT4udn0WwAA0aMy.jpg
```

```
1  666029285002620928  https://pbs.twimg.com/media/CT42GRgUYAA5iDo.jpg

                        p1   p1_conf  p1_dog                  p2   p2_conf  \
0  Welsh_Springer_Spaniel  0.465074    True              Collie  0.156665
1                 Redbone  0.506826    True  Miniature_Pinscher  0.074192

   p2_dog                   p3   p3_conf  p3_dog
0    True    Shetland_Sheepdog  0.061428    True
1    True  Rhodesian_Ridgeback  0.072010    True
```

### 1.1.14 Quality: df_tweets_copy

**1. Define** *There are less records in the "df_tweets" dataset than "tae" datasets(#2356). However, we cannot do anything on it.*

**2. Define** *"id" column name is not consistent with other two sources. Hence, renaming "id" to "tweet_id".*

**2. Code**

```
In [48]: df_tweets_copy.rename(columns={'id':'tweet_id'}, inplace=True)
```

**2. Test**

```
In [49]: df_tweets_copy.head(2)

Out[49]:              tweet_id  retweet_count  favorite_count
         0  892420643555336193           8332           38117
         1  892177421306343426           6158           32704
```

### 1.1.15 Tidiness: df_tweets_copy

*There are no tidiness issues in this dataset.*

### 1.1.16 Master File:

***Data Wranglis is completed at this step and before moving to the Data Analysis and Visualization, we need to join all the 3 sources' files into one file.***

```
In [50]: # Checking the joining ID before joining these 3 files
         print(tae.info(),ip.info(),df_tweets.info())

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2356 entries, 0 to 2355
Data columns (total 17 columns):
tweet_id                      2356 non-null int64
in_reply_to_status_id         78 non-null float64
in_reply_to_user_id           78 non-null float64
timestamp                     2356 non-null object
```

```
source                        2356 non-null object
text                          2356 non-null object
retweeted_status_id           181 non-null float64
retweeted_status_user_id      181 non-null float64
retweeted_status_timestamp    181 non-null object
expanded_urls                 2297 non-null object
rating_numerator              2356 non-null int64
rating_denominator            2356 non-null int64
name                          2356 non-null object
doggo                         2356 non-null object
floofer                       2356 non-null object
pupper                        2356 non-null object
puppo                         2356 non-null object
dtypes: float64(4), int64(3), object(10)
memory usage: 313.0+ KB
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2075 entries, 0 to 2074
Data columns (total 12 columns):
tweet_id    2075 non-null int64
jpg_url     2075 non-null object
img_num     2075 non-null int64
p1          2075 non-null object
p1_conf     2075 non-null float64
p1_dog      2075 non-null bool
p2          2075 non-null object
p2_conf     2075 non-null float64
p2_dog      2075 non-null bool
p3          2075 non-null object
p3_conf     2075 non-null float64
p3_dog      2075 non-null bool
dtypes: bool(3), float64(3), int64(2), object(4)
memory usage: 152.1+ KB
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 609 entries, 0 to 608
Data columns (total 3 columns):
tweet_id          609 non-null int64
retweet_count     609 non-null int64
favorite_count    609 non-null int64
dtypes: int64(3)
memory usage: 14.4 KB
None None None
```

```
In [51]: #Joining all three files of different sources into a single file
         #Joining condition is tweet-id
         df = pd.merge(pd.merge(tae_copy,ip_copy,on='tweet_id',how='left'),df_tweets_copy,on='tw
         df.to_csv('twitter_archive_master.csv', encoding = 'utf-8')
```

```
In [52]: #Test
```

```
         df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 2356 entries, 0 to 2355
Data columns (total 25 columns):
tweet_id              2356 non-null int64
timestamp             2356 non-null datetime64[ns]
source                2356 non-null object
text                  2356 non-null object
rating_numerator      2356 non-null int64
rating_denominator    2356 non-null int64
name                  2247 non-null object
doggo                 2356 non-null object
floofer               2356 non-null object
pupper                2356 non-null object
puppo                 2356 non-null object
hashtag               27 non-null object
dog_type              380 non-null object
jpg_url               2075 non-null object
p1                    2075 non-null object
p1_conf               2075 non-null float64
p1_dog                2075 non-null object
p2                    2075 non-null object
p2_conf               2075 non-null float64
p2_dog                2075 non-null object
p3                    2075 non-null object
p3_conf               2075 non-null float64
p3_dog                2075 non-null object
retweet_count         609 non-null float64
favorite_count        609 non-null float64
dtypes: datetime64[ns](1), float64(5), int64(3), object(16)
memory usage: 478.6+ KB
```

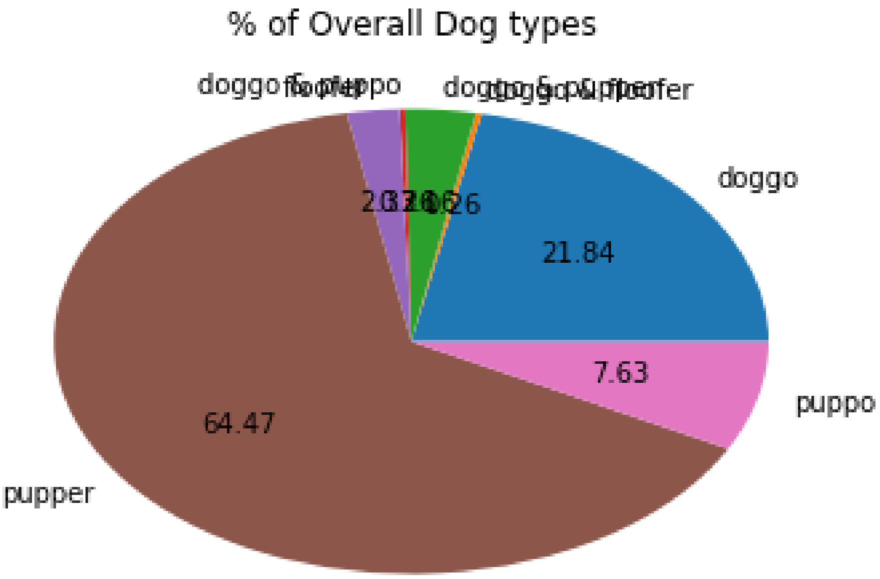
## Exploratory Data Analysis:

***Question 1: Which dog has more counts?***

```
In [53]: df1=df.groupby(['dog_type']).count()['tweet_id']
```

```
In [54]: print(df1)
         # variety 1: Pie Chart
         df1.plot(kind='pie', title='% of Overall Dog types', autopct='%.2f');
         pylt.ylabel('', fontsize=1);
```

```
dog_type
doggo                83
doggo & floofer       1
doggo & pupper       12
doggo & puppo        1
floofer              9
pupper             245
puppo               29
Name: tweet_id, dtype: int64
```

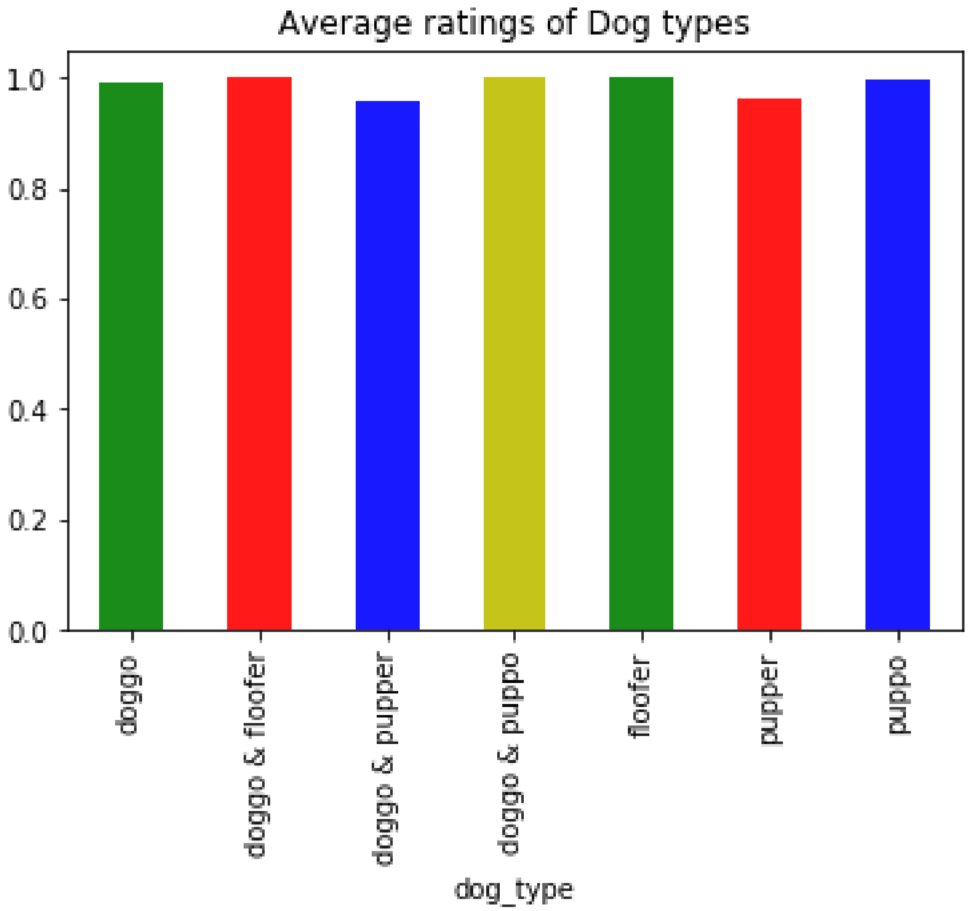


*Answer / Insight 1: Pupper, Doggo and Puppo are standing at 1st, 2nd and 3rd places in terms of tweets. However, Pupper has very high tweets than other dog types.*

***Question 2: Which dog has a better average ratings?***

```
In [55]: df1=(df.groupby(['dog_type']).sum()['rating_numerator'])/(df.groupby(['dog_type']).sum(
         print(df1)
         # variety 1: Pie Chart
         df1.plot(kind='bar', title='Average ratings of Dog types', color=['grby'], alpha=.9);
```

```
dog_type
doggo              0.990361
doggo & floofer    1.000000
doggo & pupper     0.958333
doggo & puppo      1.000000
floofer            1.000000
pupper             0.964490
puppo              0.996552
dtype: float64
```



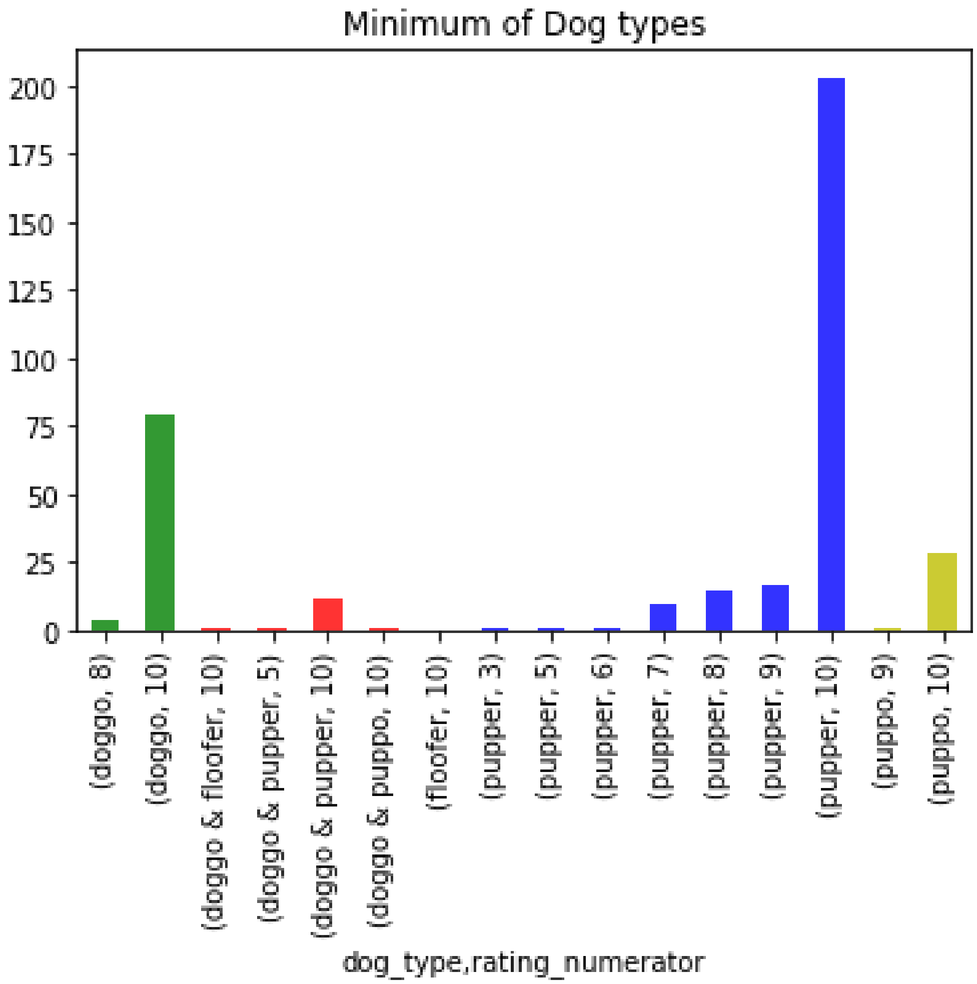
*Answer / Insight 2: Doggo, Puppo, Floofer and pipper have almost same ratings on an average. However, Pupper has more number of tweets.*

***Question 3: Which dog has more number of high ratings?***

```
In [56]: df1=(df.groupby(['dog_type','rating_numerator']).count()['tweet_id'])
         print(df1)
         # variety 1: Pie Chart
         df1.plot(kind='bar', title='Minimum of Dog types', color=['ggrrrrwbbbbbbbyy'], alpha=.8
```

```
dog_type          rating_numerator
doggo             8                     4
                  10                   79
doggo & floofer   10                    1
doggo & pupper    5                     1
                  10                   11
doggo & puppo     10                    1
floofer           10                    9
pupper            3                     1
```

```
                 5                     1
                 6                     1
                 7                     9
                 8                    14
                 9                    16
                 10                  203
puppo            9                     1
                 10                   28
Name: tweet_id, dtype: int64
```

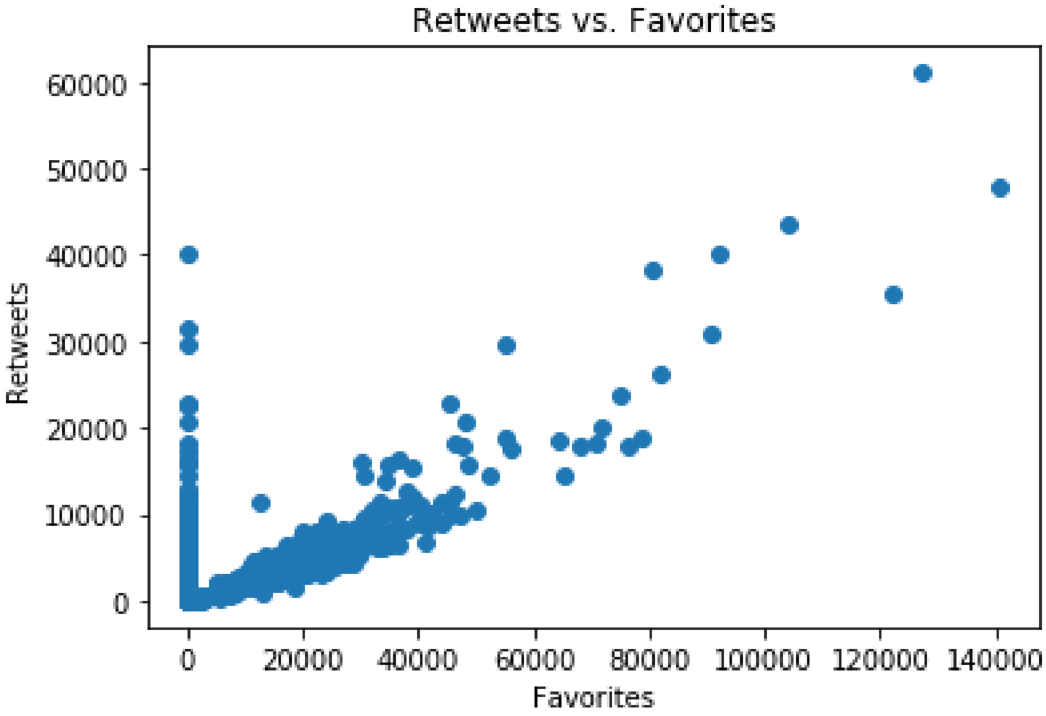


*Answer / Insight 3: Pupper has more better ratings than other Dogs.*

***Question 4: What is the relationship between Retweets and Favorites?***

```
In [57]: #creating scatter plot between retweets and favorites
         pylt.scatter(df['favorite_count'],df['retweet_count']);
         pylt.title('Retweets vs. Favorites')
```

```
        pylt.xlabel('Favorites')
        pylt.ylabel('Retweets');
```

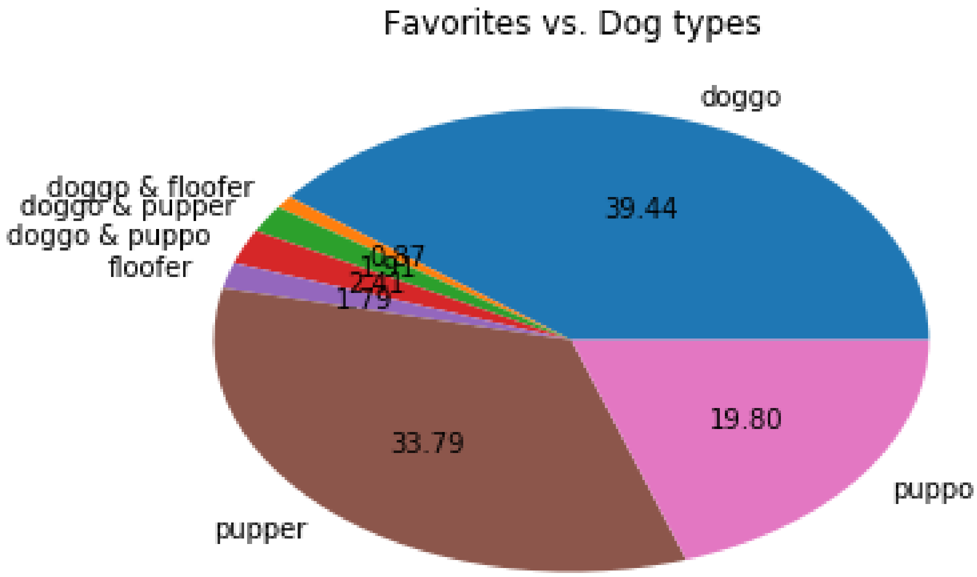


*Answer / Insight 4: There is a positive correlation between Favorites and Retweets on a dog type. It means, when there are more retweets it is likley to be favorite/like and vise versa.*

***Question 5: Which dog type was received more favorites?***

```
In [58]: df1=(df.groupby(['dog_type']).sum()['favorite_count'])
         print(df1)
         # variety 1: Pie Chart
         df1.plot(kind='pie', title='Favorites vs. Dog types', autopct='%.2f');
         pylt.ylabel('', fontsize=1);
```

```
dog_type
doggo                754957.0
doggo & floofer       16567.0
doggo & pupper        36514.0
doggo & puppo         46200.0
floofer               34271.0
pupper               646742.0
puppo                379011.0
Name: favorite_count, dtype: float64
```



*Answer / Insight 5: Doggo has more number of favorites or likes among all. However, pupper is standing at 2nd place interms of likes/favorites.*

## Conclusions:

1. Gathered data from three sources
2. Assessed the data of three files
3. I have assessed the data by visualizing it and by programming
4. Took a copy of three files and then cleaned the data of copied files
5. Created a master file "df" by joining all the three cleaned files for a quick and easy analysis
6. Based on my questions, I conclude that, although "Doggo" Dog type has more likes/favorites than other dog types. "Pupper" is also a good dog, if we consider number of top ratings
7. "Pupper" dog type is the second best among all based on favorites
8. There is a positive correlation between tweets and favorities. It means, when there are more retweets it is likley to be favorite/like and vise versa

## Limitations:

1. I felt gathering was the toughest part in the data wrangling due two multiple sources and especially pulling the data from other websites (here it is twitter)
2. Since there were 3 different sources, felt difficult while assessing and cleaning the files separately
3. Assessment varies from type of questions that you want to answer from the data
4. We have found 10 quality issues and 4 tidy issues in the data wrangling steps
5. P1, P2 and P3 columns were not used in my analysis due to lack of understanding on those columns

6. I feel urls were not useful in my analysis

```
In [60]: from subprocess import call
         call(['python', '-m', 'nbconvert', 'wrangle_act.ipynb'])

Out[60]: 0
```