Data Wrangling - Enhanced Twitter Archive

Background

We have been provided with a csv archive of twitter archives for the account we rate dogs that we need to enrich with data from two other sources to remain with a master clean data set that can be analyzed.

Gathering

1. We have been provided with an enhanced-twitter archive csv that we will just import.
2. We have a link to the image predictions that we need to download using the requests library.
3. We also need to download likes and retweet counts from twitter using the tweepy api wrapper library.

Steps done below.

```
#import required packages
import pandas as pd
import numpy as np
import requests
import tweepy
import json
#Read the manually downloaded twitter enhanced archive
twitter_enhanced = pd.read_csv('twitter-archive-enhanced.csv')
```

```
download the image predictions and read them to a dataframe.
url =
'https://d17h27t6h515a5.cloudfront.net/topher/2017/August/599fd2ad_image-predictions/image-predictions.tsv'
response = requests.get(url)
with open('image_predictions.tsv', mode='wb') as file:
    file.write(response.content)

image_predictions = pd.read_csv('image_predictions.tsv', sep='\t')

image_predictions.head()
from distutils.log import error
```

```
#create an instance of the tweepy API client
auth = tweepy.OAuth2BearerHandler(') #Authentication required to run this!!!
api = tweepy.API(auth)

#Determine how many sets containing ~100 ids you can get from the tweet_id column of enhanced archive
pieces = np.round(len(twitter_enhanced.tweet_id)/100)
print(pieces) # to check number of sets

#split the tweet_ids into the above chunks to use with the lookup_statuses api.
id_chunks = np.array_split(twitter_enhanced.tweet_id, pieces)
chunked_list = [list(array) for array in id_chunks]

#extract tweets for each of the chunks and write them line by line to the tweets_json.txt
for chunk in chunked_list:
    try:
       tweets = api.lookup_statuses(chunk,trim_user = True)
       tweets_data = [json.dumps(tweet._json) for tweet in tweets]
    except:
        print(error)
    with open('tweet_json.txt', 'a', encoding='utf-8') as f:
        for tweet in tweets_data:
            f.write(tweet)
            f.write('\n')
```

Assessing

Using both visual and programmatic assessments like below:

```
#Asses the image predictions dataframe
print(image_predictions.info())#structure of data and missing values
print(image_predictions.describe())#summary stats and value distributions
print(image_predictions.p1_dog.value_counts())# look for false values are per the #1 predictor
print('duplicates', image_predictions.duplicated().sum())#duplicates
image_predictions.head()#Visual assessment of first five rows

#Assess the tweets_df dataframe
```

```
print(tweets_df.describe())
print(tweets_df.info())
print('duplicates', tweets_df.duplicated().sum())
tweets_df.head()
```

We found the following issues with the data, mostly from the enhanced twitter archive.

Type	Issue
Quality	The `enhanced_twitter_archive` contains some tweets that are not dog ratings but rather replies or **quoted tweets**.
Quality	There are some **retweets** in the `enhanced_twitter_archive` while we only require original ratings.
Quality	There are some missing values for the **dog stages** in the `enhanced_twitter_archive` data.
Quality	The `enhanced_twitter_archive` is missing the *favorite_count* and *retweet_count* columns.
Quality	Some of the *numerator* scores seem either too high or too low in the `enhanced_twitter_archive`.
Quality	Some of the *denominator* scores seem either too high or too low in the `enhanced_twitter_archive`.
Quality	Based on its most confident prediction, some of the images in the `image_predictions` data are not for dogs.
Quality	`Enhanced_twitter_archive` has no image data.
Tidiness	**Dog stage** is a variable but is spread over four columns not one in the `enhanced_twitter_archive`.
Tidiness	`Image_predictions` prediction values are spread over many columns and need to be reshaped for easier analysis.

[1]Quality Issue

[1] I had overlooked the timestamp of the enhanced twitter archive which I cleaned later.

Cleaning

I created copies of the original data frames and cleaned them through various ways, like melting some columns in the enhanced twitter archive and image predictions, joined the data frames and cleaned up any residuals including tweets with missing images, to remain with the data frame below.[2]

```
#One more thing I picked up before saving the data: timestamp is text, lets convert it to datetime
enhanced_twitter_archive['timestamp'] = enhanced_twitter_archive['timestamp'].astype('M')

enhanced_twitter_archive.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1658 entries, 0 to 1657
Data columns (total 15 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   tweet_id            1658 non-null   int64
 1   timestamp           1658 non-null   datetime64[ns]
 2   source              1658 non-null   object
 3   text                1658 non-null   object
 4   expanded_urls       1658 non-null   object
 5   rating_numerator    1658 non-null   int64
 6   rating_denominator  1658 non-null   int64
 7   name                1658 non-null   object
 8   dog_stage           1658 non-null   object
 9   favorite_count      1658 non-null   float64
 10  retweet_count       1658 non-null   float64
 11  jpg_url             1658 non-null   object
 12  img_num             1658 non-null   int64
 13  p                   1658 non-null   object
 14  breed               1658 non-null   object
dtypes: datetime64[ns](1), float64(2), int64(4), object(8)
memory usage: 207.2+ KB
```

Which I then save to the master archive for further analysis later.

```
enhanced_twitter_archive.to_csv('twitter_archive_master.csv')
```

[2] Further details of the cleaning exercise can be found in the wrangle_act.ipynb notebook shared together with this file.