Wrangle and Analyze Data: WeRateDogs

Project Outline:

Introduction

I. Data wrangling:
Step 1. Gathering Data:
Step 2. Assessing Data:
Step 3. Cleaning Data:

II. Analysis and Visualization:

References:

Introduction:

In this project, we are going to gather data from several different sources and in a variety of formats, assess their quality and tidiness, then clean them and combine them all to create a master dataset which we will then use to answer interesting questions through analysis and produce stunning visualization and/or modeling using Python and its libraries.

The dataset that we will be wrangling, analyzing, and visualizing is the tweet archive of Twitter user @dog_rates, also known as **WeRateDogs** which contains basic tweet data for all 5000+ of their tweets as they stood on August 1, 2017.

I. Data wrangling:

Step 1. Gathering Data:

The first step in our data wrangling process is gathering data from several different sources and file formats such as:

- The **WeRateDogs Twitter Archive**, which is a cvs file that contains about 2356 tweets for each dog's name, its breed and rating as well as its development stage.
- The **Tweet Image Prediction** which is extracted programmatically using web scraping, refers to what breed of dog (or other object, animal, etc.) is present in each tweet according to a neural network.
- The **Twitter's API and JSON data**, which will be used to gather retweet count and favorite count (two missing columns in the Twitter Archive), and then **query the Twitter API** for each **tweet's JSON data** using **Python's Tweepy library** and finally **store each tweet's entire set of JSON data** in a file called **tweet_json.txt file**.

Step 2. Assessing Data:

After gathering data, the 3 tables which are **twitter_archive**, **image_prediction** and **tweet_json** were saved and assessed visually and programmatically to identify data quality issues as well as tidiness issues.

1. Quality Issues:

Twitter Archive Table:

- Q 1: There are 181 retweets as indicated by retweeted status id
- Q 2: Missing values in columns: retweeted_status_id, retweeted_status_user_id, retweeted_status_timestamp, in_reply_to_status_id, in_reply_to_user_id, in_reply_to_user_id.
- Q 3: tweet_id should be object instead of dtype int64
- Q 4: timestamp should be a datetime64 dtype type
- Q 5: Invalide dog names: "None", "a", "an", etc
- Q 6: Innacurate ratings: The number of rating numerator less than 10 are 440; 23 rating denominators are not equal to 10; and the tweet_id 835246439529840640 has a rating denominator with zero value. In addition, some ratings with decimals have been incorrectly exported.
- Q 7: There are two variables in one column which violates the "each variable forms a column" requirement. There is "href" and also "rel" which should mention only the main source of both of them.

Image Prediction Table:

- Q 3: tweet_id should be object instead of dtype int64
- Q 8: Dog breed name values in the p1, p2, and p3 had some uppercase and lowercase letters.
- Q 9: Multiple columns containing the same type of data in the predictions table.
- Q 10:There are 66 jpg_url duplicates Image Predictions Table

Tweet JASON Table:

- Q 3: tweet_id should be object instead of dtype int64
- Q 11: There are duplicates in tweet_id in tweet_data_clean Table.
- Q 12: Some additional work on our 3 DataFrames should be done for better readability and before merging them.

2. Tidiness issues:

- T 1: Dog stage data is separated into 4 columns (doggo, floofer, pupper, and puppo) which should be merged in one column.
- T 2: All 3 tables are related by a common key "tweet_id" and shouldn't be divided into separate DataFrame

Step 3. Cleaning Data:

Cleaning data is the third step in data wrangling process. We are going to fix the quality and tidiness issues that we have identified in the assess step.

It is worth noting that programmatic data cleaning as its own separate process within data wrangling has 3 steps: **defining**, **coding** and **testing**. First of all, we will define cleaning data plan in writing, then we'll translate it into code, and finally, we'll test the dataset often using code to make sure the cleaning code worked.

1. Cleaning for Quality:

	Quality Issues:	**Definition:**
Q1	There are 181 retweets as indicated by retweeted status id	Remove all retweets from table and keep only original tweets
Q2	Missing values in columns: retweeted_status_id, retweeted_status_user_id, retweeted_status_timestamp, in_reply_to_status_id, in_reply_to_user_id, in_reply_to_user_id, expanded_urls.	Remove columns that won't be used for analysis using drop() method
Q3	tweet_id should be object instead of dtype int64	Convert tweet_id from an integer to object using the astype() function for all DataFrame.
Q5	timestamp should be a datetime64 dtype type	Convert Timestamps from object to datetime format using pandas to_datetime() function.
Q6	Invalide dog names: "None", "a", "an", etc	Replace Invalide dog names with NaNs.
Q7	Innacurate numerators and denominators ratings: Some ratings with decimals have been incorrectly exported	Find all ratings that contained decimals and replace the numerators with the correct values.
Q8	There are two variables in one column which violates the "each variable forms a column" requirement. There is "href" and also "rel" which should be split into 2 separate columns.	Optimize the source content by "Twitter for iphone", "Vine - Make a Scene", "Twitter Web Client", and "TweetDeck".
Q9	Dog breed name values in the p1, p2, and p3 had some uppercase and lowercase letters.	Capitalize the first letter of dog breed (p1, p2, and p3) for consistence.
Q10	There are 66 jpg_url duplicates in Image Predictions Table.	Drop 66 jpg_url duplicated.
Q11	There are duplicates in tweet_id in tweet_data_clean Table.	Drop the duplicates in tweet_id from tweet_data_clean.
Q12	Some additional work on our DataFrames should be done for better redability	We have to set column width to infinite; then change the column names; and replace the underscores with spaces; and finally reorder the column placement .

2. Cleaning for Tidiness:

After addressing missing data and quality issues first, cleaning for tidiness is the next step.

	Tidiness Issues:	**Definition:**
T1	Dog stage data is separated into 4 columns which should be merged in one column	Extract dog stage from text column into the new dog_stage using pandas.Series.str.cat to concatenate strings in the Series and merge the 4 columns into one column: **dog_stage** and then drop the unrequired columns
T2	All 3 tables are related by a commun key "tweet_id" and shouldn't be divided into separate DataFrame	Merge the twitter archive into the image predictions using Join Method and merge the resulting DataFrame with Twitter JSON.

This concludes the process of data wrangling: gathering data, assessing data and cleaning data. The next step is to analyze and visualize our data and communicate our findings. Before starting Analysis and Visualization of the dataset, we have to store our cleaned DataFrame into a new cvs file.

II. Analysis and Visualization:

The next step is to start analysing and visualizing our dataset, and then drawing any valuable conclusions.

- Most popular dog breed
- Most frequent Dog stages
- Relationship between the number of retweet and favorite count:
- Retweets and favorites over time

References:

WeRateDogs Twitter:
https://twitter.com/dog_rates?ref_src=twsrc%5Egoogle%7Ctwcamp%5Eserp%7Ctwgr%5Eauthor
Pandas Documentation:
https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.Series.str.cat.html
Twitter API:
https://stackoverflow.com/questions/47612822/how-to-create-pandas-dataframe-from-twitter-search-api
30 Days of Python - Day 21 - Twitter API with Tweepy - Python TUTORIAL:
https://www.youtube.com/watch?v=dvAurfBB6Jk

Done by: Zakariya Boutayeb
Udacity