Wrangle Report

By Mahmoud Adel Taya Mohamed

Introduction:-

WeRateDogs is a Twitter account that rates people's dogs with a humorous comment about the dog. The account was started in 2015 by college student Matt Nelson, and has received international media attention both for its popularity and for the attention drawn to social media copyright law

WeRateDogs asks people to send photos of their dogs, then tweets selected photos rating and a humorous comment. Dogs are rated on a scale of one to ten, but are invariably given ratings in excess of the maximum, such as "13/10". Popular posts are re-posted on Instagram and Facebook.[2] In 2017, Nelson started a spin-off Twitter account, Thoughts of Dog.

In this project we will try to make data wrangler for most favorite tweeter group to understand many of information about WeRateDogs

Gathering Data for this Project

This project involved gathering of data from three different sources as listed below. For each of the data source a different method of data gathering:-

1-Importing data via csv

2-Using requests to download data off internet

3-Scrape data from an API

This was challenging and fun at the same time.

Three data sources

Enhanced Twitter Archive

The WeRateDogs Twitter archive provided by Udacity. This contains basic tweet data for all 5000+ of their tweets, but not everything.I manually downloaded this file manually by clicking the following link: twitter_archive_enhanced.csv

Image Predictions File

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: image_predictions.tsv

Data via the Twitter API

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.

Assessing data

The data we got this seldom in standard formats or to the way we want it. After gathering the required data, I came up with the following issues with it:-

I used her two type of assessment for data

1-visual assess

2-Programmatic assessment(using python and many libraries like Pandas ,NumPy and other)

For this assessment got some issues ,

Tidiness Issues

In twitter-archive-enhanced file

1-Create dog stage to put all type of dogs classification

2-'timestamp' change from string to date format day, month , year columns and delete timestamp

3-Combine three different data frames into one master data set

Quality Issues (Completeness, validity, accuracy, consistency (content issues)

In twitter-archive-enhanced file

1-'rating_denominator' columns standard 10

2-rating_numerator columns mistyping

3- Change 'None' to empty cell in doggo ,floofer ,pupper ,puppo to add in dogs stage

After that delete doggo,floofer,pupper,puppo columns

4-Change name for text column

5-Delete unusual columns we will not use in analysis

6-Delete timestamp

7- Replace 'None' with NaN to indicate the missing values

8- keep original tweets (no retweets) that have images

9- The dog names format should be consistent. Make the first letter capital for all the names.

10-name column have error value 'a' should be correct or delete this value NaN

In Image Predictions

11- Drop duplicates values from jpg_url

12- The column names such as p1,p2 are not descriptive.

13- The prediction dog breeds involve both uppercase and lowercase for the first letter.

In Tweet JSON

14- Delete columns that won't be used for analysis

Cleaning Data

I used my knowledge of python and searching over the internet i.e. google, stackoverflow, stackabuse etc for references and possible guideance to resolve the above mentioned issues to the best of my knowledge. There was lot trial and error for difficult cases where regular expressions had to be used but at the same time some things for instance dropping the not so useful columns was pretty straight forward.

We start with cleaning by python and python librarses to get clean data as:

I start cleaning by clean tidiness issues to prepare the structure data frame starting by twitter-archive-enhanced file, first Create dog stage to pull all dogs classification in one columns, at 'timestamp' column change from string to date format day, month , year columns and delete timestamp column, after clean tidiness issues, I jumped to solve quality issues starting **In twitter-archive-enhanced file** 'rating denominator' column has a standard rate 10 but it have different rate start to set all to 10 rate, same mistyping happen in rating numerator columns, some columns have None value which mean nothing but not NaN Changed empty cell in doggo ,floofer ,pupper ,puppo to add in dogs stage during merge in one columns to be eassy to replace empty cell to ’NaN, then no need for doggo,floofer,pupper,puppo columns so deleted.

In same file Change name for text column to be tweets to indicate as tweet to any text, some column insertded but not useful deleted that because I will not use in analysis same for timestamp

keep original tweets (no retweets) that have images, change in name for dogs the first letter capital for all the names.

By check name column have error value 'a' should be correct or delete this value NaN

In Image Predictions file , Drop duplicates values from jpg_url, The column names such as p1,p2 are not descriptive and others , The prediction dog breeds involve both uppercase and lowercase for the first letter for that I set all uppercase.

In Tweet JSON, delete columns that won't be used for analysis,

After clean most of tidiness and quality issues Combine three different data frames into one master data set.