JUNE 2020

Project: Wrangling and Analyzing Twitter Data

Author: Andrea Claudia Villanca Rosales

WeRateDogs™ @dog_rates · Apr 16
Say hello to Alice. I'm told she enjoys car rides and smells good. 12/10 would give her everything she could ever want

WeRateDogs™ @dog_rates · Apr 16
A photographer took pictures before and after he told his bunny he's a good boy. Here are the results. 13/10

Boston Magazine

Introduction

Real-world data rarely comes clean... that's why data wrangling becomes an important part of the data analysis process.

The dataset that we wrangled, analyzed and visualized in this project 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.

The goal of the project was to wrangle WeRateDogs Twitter data to create interesting and trustworthy analyses and visualizations. Using Python and its libraries, we gathered data from a variety of sources and in a variety of formats, assessed its quality and tidiness, then cleaned it before proceeding with the analyses. In this document we will briefly explain the process performed to obtain valuable insights.

Relevant Questions

As a first step in our Data Analysis process, we posed the important questions we wanted to answer to satisfy the Project Motivation.

Wrangling Process

Gathering

We gathered three pieces of data, as described below, in a Jupyter Notebook titled wrangle_act.ipynb:

- We downloaded twitter_archive_enhanced.csv manually by clicking the link provided by Udacity.
- The image_predictions.tsv, hosted on Udacity's servers, was downloaded programmatically using the Requests library and the following URL: https://d17h27t6h515a5.cloudfront.net/topher/2017/August/599fd2ad_image-predictions/image-predictions.tsv
- Using the tweet IDs in the WeRateDogs Twitter archive, we queried the Twitter API for each tweet's JSON data using Python's Tweepy library and stored each tweet's entire set of JSON data in a file called tweet_json.txt file. Each tweet's JSON data was written to its own line. Then we read this .txt file line by line into a pandas DataFrame.

As a result, we ended up having the following three dataframes:

- twitter_archive
- tweet_predictions
- tweets_info

Assessing

We then proceeded to visually and programmatically assess the data, to identify quality and tidiness issues in the three dataframes.

Cleaning

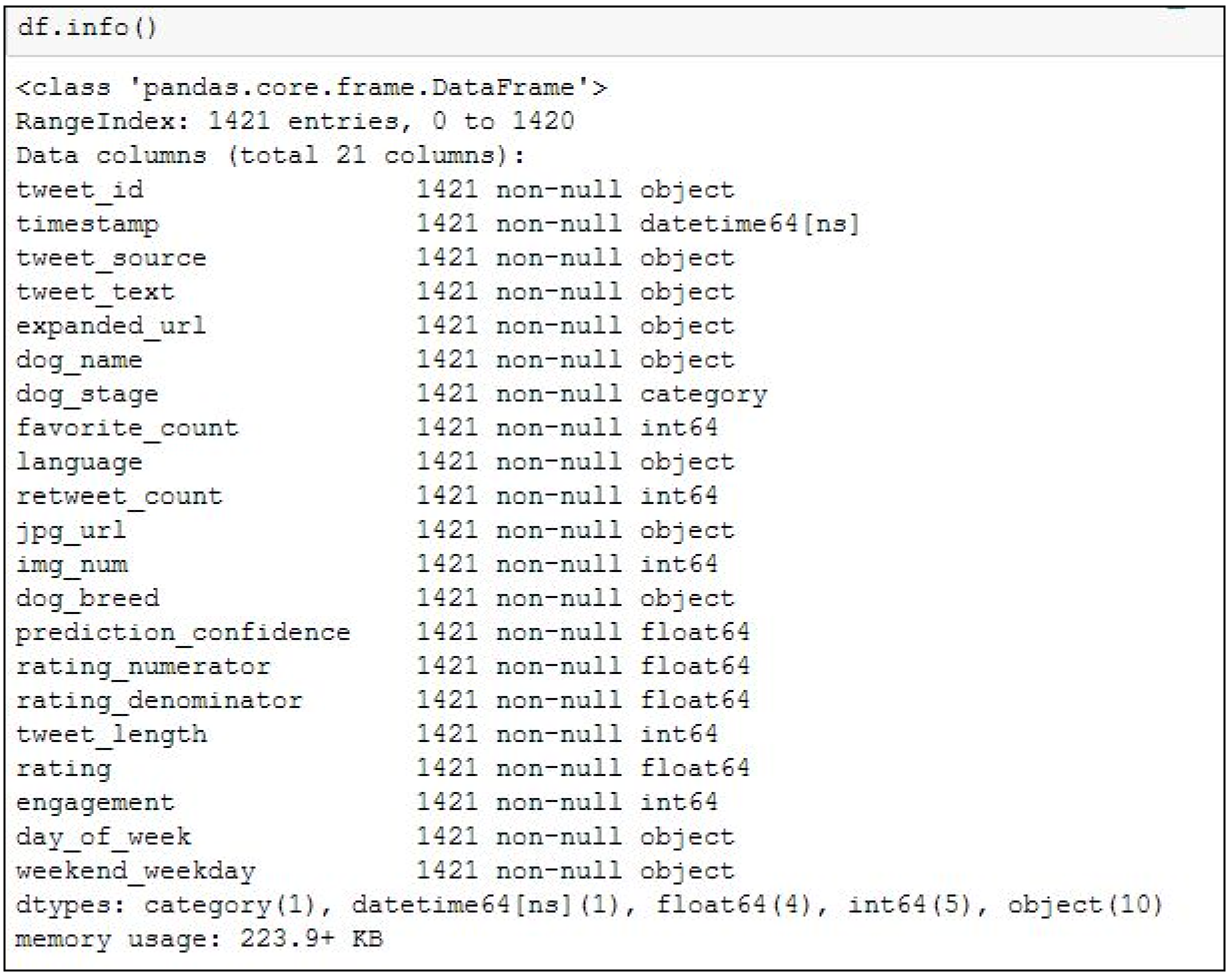
Next, in the same Jupyter Notebook wrangle_act.ipynb, we proceeded to clean each of the issues we previously assessed using the define, code, and test steps.

Feature Engineering

To answer some of our questions and to get a deeper understanding of our data, there were variables we needed to create from other existing columns. So we also created those new variables:

- Tweet Length (without url)
- Rating
- Engagement (retweet count + favorite count)
- Day of the week
- Weekend/Weekday

The resulting DataFrame had the following summary information:



```
df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1421 entries, 0 to 1420
Data columns (total 21 columns):
tweet_id                 1421 non-null object
timestamp                1421 non-null datetime64[ns]
tweet_source             1421 non-null object
tweet_text               1421 non-null object
expanded_url             1421 non-null object
dog_name                 1421 non-null object
dog_stage                1421 non-null category
favorite_count           1421 non-null int64
language                 1421 non-null object
retweet_count            1421 non-null int64
jpg_url                  1421 non-null object
img_num                  1421 non-null int64
dog_breed                1421 non-null object
prediction_confidence    1421 non-null float64
rating_numerator         1421 non-null float64
rating_denominator       1421 non-null float64
tweet_length             1421 non-null int64
rating                   1421 non-null float64
engagement               1421 non-null int64
day_of_week              1421 non-null object
weekend_weekday          1421 non-null object
dtypes: category(1), datetime64[ns](1), float64(4), int64(5), object(10)
memory usage: 223.9+ KB
```

Storing, Analyzing, and Visualizing Data

We stored the gathered, assessed, and cleaned DataFrame in a CSV file named twitter_archive_master.csv.

Then, we analyzed and visualized our wrangled data in the wrangle_act.ipynb Jupyter Notebook.

Conclusions

The insights found in our Data Analysis process are described in act_report.pdf.