Originator: Michael Wei
Rev: B
Date: 3/31/2019

Introduction

This report describes the data wrangling efforts we employed to analyze the WeRateDogs twitter account. Data wrangling includes Gathering, Assessing, and Cleaning.

Gathering

First, data were gathered programmatically under the following formats:

- csv
- Twitter Application Programming Interface (API)/json
- tsv

The csv format was handled using simple Pandas methods for data import. The data were provided in a csv file generated from Udacity instructors.

Twitter API/json was handled using a Twitter Developer account to gather relevant data (retweet_count and favorite_count) associated with tweet_ids in the Twitter archive and dump into a json file. Because the Twitter API enforces a time and data gathering restriction, we set several arguments to True (wait_on_rate_limit and wait_on_rate_limit_notify) in order to allow the code to run in full and unsupervised.

Lastly, image prediction data were programmatically gathered from an internet source using standard Python os module methods.

Assessing

Second, we assessed the data visually and programmatically to detect issues based on Udacity classroom qualifications for clean data and the Hadley Wickham definition for Tidiness.

Programmatically, we could assess issues such as suboptimal datatypes (tweet_id and timestamp), inaccurate dog ratings (Both numerator and denominator), duplicated data (retweeted_status_id), and inaccurate dog types (image predictions). Part of our assessment of inaccurate ratings was based on the philosophy of “They’re good dogs, Brent.”

Visually, we could detect multiple URLs present, swear words (Because part of this project is a cleaning exercise, we opted to clean the dirty language), and multiple representations of “Dogtionary” names. Overall, most of the issues came from the twitter-archive-master.csv file.

Cleaning

Third, the data were cleaned based on the initial list of Assessment Items (Quality & Tidiness). We detected several additional issues in this process, most notably:

Originator: Michael Wei
Rev: B
Date: 3/31/2019

1) The issue of inaccurate ratings actually had multiple sub-issues: decimal ratings, negative ratings, excessively high ratings, and significantly more dog ratings under 10 (Violating the "They're good dogs, Brent" philosophy). We selected a numerator_rating range between 3 and 20 for analysis (Several good dogs showed up with a tweet rating of 3). Brief sampling of tweets with a numerator below 3 showed that they were more likely associated with non-dog tweets. While we also found many good dogs, most of them were below a rating of 20. Tweets with excessively high numerator ratings were associated with celebratory occasions (e.g. 1776 for a patriotic dog, or 420 for a dog associated with marijuana).
2) expanded_urls column had multiple representations of URLs. We opted to keep URLs with a photo or video format. We kept URLs with videos in case the image predictor included tweets associated with videos.
3) Potentially missed "Dogtionary" terms, such as "corgo," "fluffer," "woofer," and "boofer". We added these terms to our RegEx pattern to detect any missed terms in the text. If multiple "Dogtionary" terms were detected, we converted them to "multiple".
4) Potential retweets found based on presence of values in in_reply_to_status_id and reply_status columns. We removed tweets with a value in this column, because they are duplicated.

One issue that we could not clean is the obviously inaccurate dog types detected by the image predictor. Although we could have selected the correct dog type based on either the highest image prediction confidence value detected, a True value in one of the result columns (p1_dog, p2_dog, or p3_dog), or a combination, we could not entirely filter out all of the non-dog nouns.

Lastly, we combined the dataframes to create a master twitter dataframe for basic data analysis in the quest to find out who is the "goodest dog, Brent."