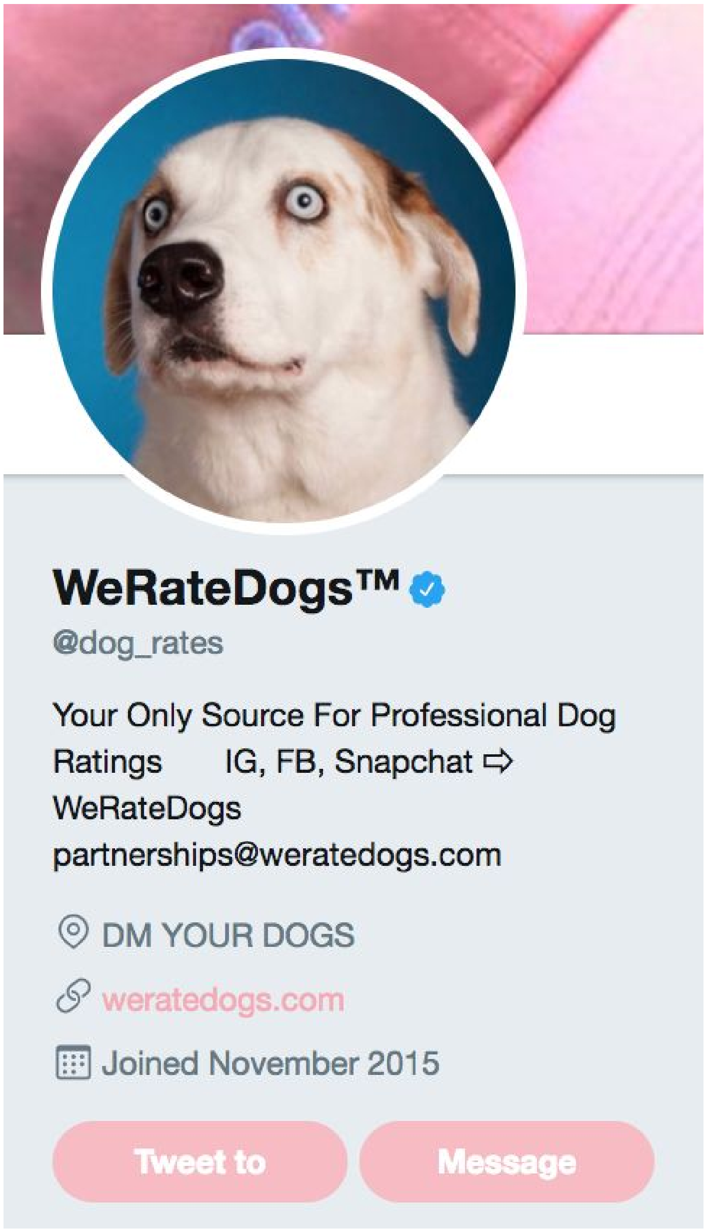
Data Wrangling: **@WeRateDogs**

By Bayan AlArifi



WeRateDogs™

@dog_rates

Your Only Source For Professional Dog Ratings IG, FB, Snapchat ⇨ WeRateDogs partnerships@weratedogs.com

DM YOUR DOGS

weratedogs.com

Joined November 2015

Introduction

This reports on data wrangling steps: gather, assess, and clean of the @WeRateDogs tweets. This Twitter account rates dogs with humorous commentary. The rating denominator is usually 10. However, the numerators are usually greater than 10. This aspect was not cleaned as it is part of the humor and popularity of WeRateDogs.

Gathering Data

The data was gathered from multiple sources, as listed below. The twitter archive is great, but it only contains very basic tweet information. Additional gathering, then assessing and cleaning was required for "Wow!"-worthy analyses and visualizations.

- **Load @WeRateDogs twitter archive file**

  Load the **@WeRateDogs** tweets archive (2356 tweets). This archive contains basic tweet data (tweet ID, timestamp, text, etc.) for all 5000+ of their tweets as they stood on August 1, 2017.

- **Extracting a file from a server**

  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 is downloaded programmatically using the Requests library.

- **Extract data from twitter API**

  Retweet count and favorite count (likes) are two of the notable column omissions from the Twitter archive. Fortunately, this additional data can be gathered, along with any additional interesting data by using tweepy library. The tweet IDs in the @WeRateDogs Twitter archive were used to query the Twitter API for each tweet's JSON data, and store each tweet's entire set of JSON data in a file called tweet_json.txt.

Assessing Data

In this step data was assessed visually and programmatically using pandas library. Furthermore, dirty data (content or “quality” issues) and messy data (structural or “tidiness” issues) were distinguished identified. A summary of the findings is as follows;

Quality Issues

1. Some columns have inaccurate data types. For example, the timestamp and retweeted_status_timestamp columns are objects, although they should be datetime objects. Some other columns should be integers/strings instead of floats.
2. Remove retweets, i.e. tweets that are not original or are responses.
3. Remove tweets that are not ratings, i.e. tweets without images.
4. The name column has many entries which do not look like names. The most frequent entry in name column is "a", which is not a name.
5. There are unnecessary columns in the dataframes like source, img_num, expanded_urls, source, retweeted_status_id, retweeted_status_user_id, retweeted_status_timestamp, in_reply_to_status_id, and in_reply_to_user_id. The cleaned dataframe can be condensed.
6. There are 2075 rows in the image predictions dataframe and 2356 rows in the archive dataframe.
7. Remove tweets in predictions where all probabilities of p_dog are False, i.e. where the algorithm is certain the image is not a dog.
8. The columns p1, p2, p3 in predictions are inconsistent in their capitalization.

Tidiness Issues

1. The columns doggo, floofer, pupper and puppo represent the dog stages, and should be collapsed into one column named dog_stage.
2. All three dataframes should be merged into one clean dataframe since they all hold information about the same entity; tweet.
3. Sort the timestamp column.

Cleaning Data

In this step all the data was cleaned to fix the quality and tidiness issues identified in the Assessing Data step. Each step of the data cleaning process (defining, coding, and testing) was identified. The cleaned code was then tested visually and programmatically. A summary of the cleaning done is as follows;

1. Change some of the columns data types in the cleaned_archive dataframe.
2. Remove retweets, i.e. tweets that are not original. or responses and delete their respective columns.
3. Remove tweets that are not ratings, i.e. tweets without images.
4. The name column has many entries which do not look like names.
5. Remove the unused/unneeded columns; source, retweeted_status_id, retweeted_status_user_id, retweeted_status_timestamp, in_reply_to_status_id, in_reply_to_user_id.
6. There are missing data in the image predictions dataframe. It only has 2075 rows instead of 2356. Hence, only keep the rows in the archive that have image predictions.
7. Remove tweets in predictions where all probabilities of p_dog are False, i.e. where the algorithm is certain the image is not a dog.
8. Make the values of columns p1, p2, p3 in predictions all lowercase.
9. Melt the 'doggo', 'floofer', 'pupper' and 'puppo' columns into one column 'dog_stage'.
10. Merge the dataframes into one cleaned dataframe.
11. Sort the timestamp column in the cleaned dataset.
12. Store the final cleaned dataframe cleaned_archive into a CSV file.