Wrangle_Report

December 24, 2018

1 Introduction

WeRateDogs is a Twitter account that rates people's dogs with a humorous comment about the dog. In this project, the wrangling efforts are conducted on the tweet archive of WeRateDogs account in a Jupyter notebook using Python and its libraries.

2 Data Gathering

Data is gathered from the following 3 sources:

1. The WeRateDogs twitter archive which is given to be downloaded manually and uploaded to the jupyter notebook.
2. The tweet image predictions that contains the breed of dogs identified from dog images bu running every image through a neural network. This data was downloaded programmatically using the *requests* library.
3. 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. The outputs of the twitter API are in json and these outputs were stored in a text file called *tweet_json.txt* in separate lines. Deleted tweets were also stored in a dictionary. After storing the data, the text file was opened and read to extract tweet_id, retweet_counts and favorite_counts from each json content.

3 Data Assessing

In this section gathered data are assessed for their quality and tidiness both visually and programmatically. The aim is to identify at least 8 quality issues and 2 tidiness issues. Here, we walk through each file separately to understand issues associated with it.

3.1 WeRateDogs Twitter Archive

This file has some important information about evey single tweet including tweet_id, tweet_text, name, timestamp, source, dog ratings and the stage of dogs. A quick glimpse through the data revealed the following quality and tidiness issues:

3.1.1 Quality issues:

- dog names column has missing values (None instead of NaN) and incorrect names ('a', 'the', etc.).
- *in-reply_to_status_id* and *in_reply_to_user_id* and *source* columns are unnecessary for our analysis.
- There is infromation about retweets in *retweeted_status_id*, *retweeted_status_user_id* and *retweeted_status_timestamp* columns.
- Missing values in dog stages columns are filled with None instead of NaN.
- Dog stages columns have object datatype.
- Timestamp column has object datatype.
- The text column contains the url. Expanded_url also has that information.

3.1.2 Tidiness issues:

- Dog stages are in 4 different columns.

3.2 Image Prediction

This file consists of columns like tweet_id, jpg_url, and predictions of dog_breed from every image. A quick glimpse through the data revealed the following quality and tidiness issues:

3.2.1 Quality issues:

- A number of breed names start with small letters.
- There are rows in p1-dog, p2-dog, p3-dog columns with False values meaning that the prediction was incorrect.

3.2.2 Tidiness issues:

- The table has 9 columns for dog's breed, confidence and dog prediction.

3.3 Twitter API

This data has 2342 rows and 3 columns; tweet_id, retweet_count and favorite_count. Datatypes are all integer and there is no missing value.

4 Data Cleaning

A copy of each dataframe was taken to perform the cleaning process. The following steps were taken to clean the data and save as a .csv file.

- "None" values and incorrect names in dog names were replaced with NaN .
- Unnecessary columns from twitter archive were dropped.
- Rows relating to retweets were removed from twitter archive and then the retweeted columns were dropped.
- "None" values in dog stages were replaced with NaN.
- One column was created representing the dog stage.

- The datatype of dog_stage and timestamp columns were changed to 'category' and 'datetime' respectively.
- The url from the text column was removed as already provided in expanded_url.
- Breed of dog was identified and a single column was created for that. "NaN" was placed for False values.
- First letter of the dog_breed names was uppercased.
- A master tables containing all the values was created.