Data Wrangling Project - WeRateDogs Dataset Wrangling

Udacity - Data Analyst Nanodegree

Prepared by
Haifaa Mohamad Alzahrani

IHAIFAA@GMAIL.COM

Data Wrangling

Overview

This report covers the 4th project of Data Analyst Nanodgeree, the wrangling process of WeRateDogs Twitter account, @dog_rates. This account posts dogs' images and asks its followers to rate them for fun. It has about 9.2M followers.

There are five steps in this project:

1. Data Wrangling:
    a. Data Gathering
    b. Data Assessing
    c. Data Cleaning
2. Store the cleaned dataset
3. Basic insights and analysis
4. Documenting the wrangling process

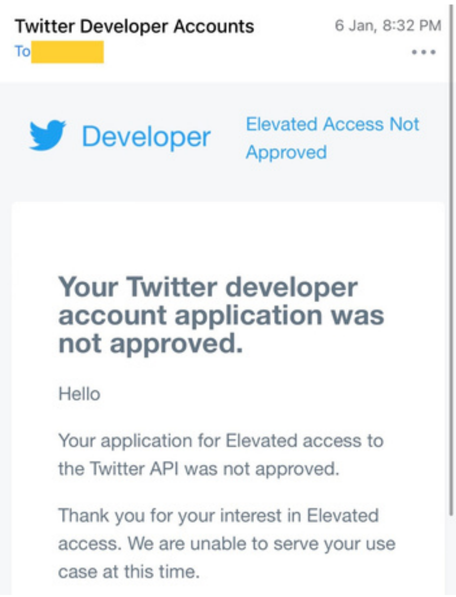
In the next section, I will discuss the first part in detail.

Data Wrangling Process

1. Data Gathering

The required data for this project comes from three different sources as follows:

1. @dog_rates archive, was provided by Udacity and it can be directly downloaded. It has 2356 tweets including tweet_id, rating_numerator, rating_denominator etc.
2. The tweet image predictions which is hosted on Udacity, and I obtained it by consuming the API via Request library. It includes predictions based on CNN to classify the dogs breed.
3. More details about the tweets such as retweet count and favorite count. I attempted to get these data via Twitter API or Tweepy, but my request was rejected as shown below. Therefore, I used the tweet json file provided by Udacity.



Twitter Developer Accounts

6 Jan, 8:32 PM

To [redacted]

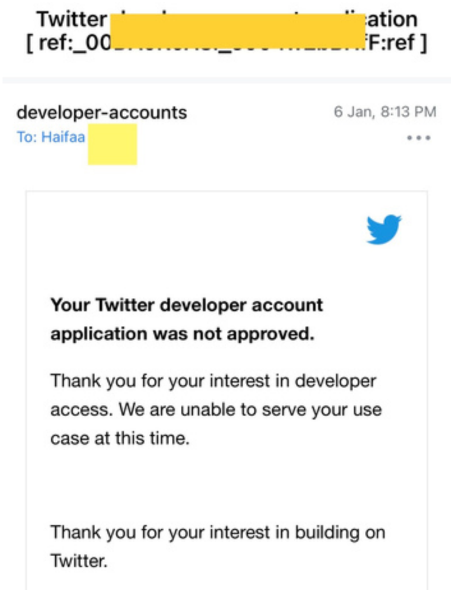
Elevated Access Not Approved

Your Twitter developer account application was not approved.

Hello

Your application for Elevated access to the Twitter API was not approved.

Thank you for your interest in Elevated access. We are unable to serve your use case at this time.



Twitter [redacted]ation [ref:_00[redacted]F:ref]

developer-accounts

6 Jan, 8:13 PM

To: Haifaa [redacted]

Your Twitter developer account application was not approved.

Thank you for your interest in developer access. We are unable to serve your use case at this time.

Thank you for your interest in building on Twitter.

Data Wrangling Process

2. Assessing Data

After gathering all the required data, I started the assessing step both visually and programmatically. Here, I defined 11 quality issues and 3 tidiness issues.

Quality issues

In twitter_archive data:

1. No need for all observations, we can exclude retweet data and in-reply data.
2. No need for in_reply_to_status_id, in_reply_to_user_id, retweeted_status_id, retweeted_status_user_id, retweeted_status_timestamp.
3. The data type of timestamp must be datetime.
4. The rating_denominator must be 10, no need for different values.
5. The source of images should be a direct link or label such as "Twitter for iPhone", not an HTML element for more readability.

In image_prediction data:

1. No need for all attributes, we only need the highest prediction.
2. 66 images are duplicated.
3. The predictions are not always for dogs, e.g. orange & paper_towel.

In tweet_json data:

1. The number of observations is different than what is included in twitter_archive.

Generally

1. The dataframes have inconsistent headers names, so I'll rename them after creating the final twitter_archive_master.

Data Wrangling Process

2. Assessing Data

Tidiness issues
In twitter_archive data:

1. doggo, floofer, pupper, puppo are all stages of dog, should be in one column.

Generally

1. There are 3 tables, with different attributes but they refer to the same observational unit or tweets. So, we need to join them in order to make one tidy table. This will be done by filtering data based on tweet_id to keep only the tweets that match with twitter_archive. Moreover, the attributes must be filtered to use only the relevant data. Thus, each observation will form a row, and each type of observational unit forms a table (this is a result of solving this issue and the previous one)

Data Wrangling Process

3. Cleaning Data

In order to clean data and resolve the previous issues, I went through: defining, coding, and testing. Here is a summary of this process:

1. I excluded the retweet and replied tweets
2. Then removed related columns from twitter_archive: in_reply_to_status_id', 'in_reply_to_user_id', 'retweeted_status_id', 'retweeted_status_user_id', and 'retweeted_status_timestamp'.
3. Convert timestamp type into datetime and remove unwanted parts (+0000)
4. Drop all data with rating_denominator differs from 10, they might have misleading ratings.
5. Make source of images either Twitter for iPhone, Twitter Web Client, TweetDeck, or Vine.
6. Keep only image_prediction with the highest probability and drop the others.
7. Remove 66 duplicated images.
8. Keep only the predictions that refer to dogs and remove the rest.
9. Merge tweet_json with twitter_archive based on tweet_id.
10. Create one column to hold dog_stage and solve tidiness issue.
11. Merge all 3 tables into one table.
12. Make consistent header names for the resulting table.