

Data Wrangling Report

1. Gathering Data

About the Dataset(s)

The dataset to be wrangling is the tweet archive of Twitter user @dog_rates (https://twitter.com/dog_rates), also known as WeRateDogs. This archive dataset consists of 2356 basic tweet data from November 2015 to August, 2017. WeRateDogs is a Twitter account that rates people's dogs with a comment about the dog.

Based on the images in the above dataset, another dataset is created which consists of image predictions with each tweet ID, image URL, and the image number corresponding to the most confident prediction (numbered 1 to 4 since tweets can have up to four images).

Gather Twitter archive CSV file

Using the link provided by Udacity, I downloaded the WeRateDogs Twitter archive manually as twitter_archive_enhanced.csv (https://d17h27t6h515a5.cloudfront.net/topher/2017/August/59a4e958_twitter-archiveenhanced/twitter-archive-enhanced.csv) file and imported this file into a Dataframe.

Gather tweet image predictions

I downloaded the tweet image predictions file hosted on Udacity's servers programmatically using Python's requests library and saved it locally to `image_predictions.tsv` file. Then, I imported this file into a Python Pandas dataframe.

Gather data from Twitter API

Using the tweet IDs in the Twitter archive, I accessed all data for every tweet from Twitter API and stored every tweet's entire set of JSON data in a file called `tweet_json.txt` file. Created a dataframe `status_df` from this JSON including only `tweet_id`, `retweet_count`, `favorite_count` and `display_text_range` data.

2. Assessing Data

Visual Assessment

I opened the `twitter_archive_enhanced.csv` and `image_predictions.tsv` in Excel and scrolled through them, looking for quality and tidiness issues. I was able to spot the following 2 quality and 2 tidiness issues:

- Quality: unnecessary html tags in source column of twitter archive in place of utility name:
`Twitter for iPhone`
- Quality: text column of twitter archive contains untruncated text instead of displayable text
- Tidiness: doggo, floofer, pupper and puppo columns in `arc_df` table should be merged into one column named "stage"
- Tidiness: Twitter archive data without any duplicates (i.e. retweets) will have empty `retweeted_status_id`, `retweeted_status_user_id` and `retweeted_status_timestamp` columns, which can be dropped

Programmatic Assessment

I used pandas `info` method on `arc_df` to spot erroneous datatypes and other quality issues. Then I used `value_counts` method on `rating_numerator`, `rating_denominator` and `name` columns to look up the range of their values and its distribution. Also, to verify 1 tidiness issue that found during the visual assessment, I queried the archive Dataframe to see if any of its tweets has more than one dog-stage mentioned. This entire activity helped me to identify the following 7 quality issues.

- contains retweets and therefore, duplicates
- many `tweet_id(s)` of `arc_df` table are missing in `img_df` (image predictions) table
- erroneous datatypes (`in_reply_to_status_id`, `in_reply_to_user_id` and `timestamp` columns)
- `rating_numerator` column has values less than 10 as well as some very large numbers (e.g. 1176)
- `rating_denominator` column has values other than 10
- erroneous dog names starting with lowercase characters (e.g. a, an, actually, by)
- some records have more than one dog stage

The `info` method on the other 2 Dataframes (`img_df` and `status_df`) didn't reveal any quality issues. However, after looking at the sample of each of these Dataframes, I was able to identify the following 2 tidiness issues:

- "breed" column should be added in `arc_df` table; its values based on `p1_conf` and `p1_dog` columns of `img_df` (image predictions) table
- `retweet_count` and `favorite_count` columns from `status_df` (tweet status) table should be joined with `arc_df` table

3. Cleaning Data

As all of the quality and tidiness issues were related to `arc_df` table, I created a copy of only this table and named it `archive_clean`. For each quality/tidiness issue, I performed the programmatic data cleaning process in 3 stages - Define, Code & Test. During the cleaning process, I converted the datatypes of source and newly created stage columns of `archive_clean` to category datatype.

Storing Data

After the completion of the cleaning process, I stored the `archive_clean` DataFrame in `twitter_archive_master.csv` file.