Data Wrangling Process

by Arturo Parrales Salinas

August 26, 2018

This project consisted in performing all the data wrangling pipeline for WeRateDogs tweets from late 2015 to late 2017. The data wrangling process involved three main steps:

I. Gather data from multiple sources
II. Asses the collected data
III. Clean the data based on the assessment done

There is an additional step in case the cleaned data will be used in the future. This additional step is:

IV. Store cleaned data (optional)

I. Gather Data

We started by collecting data:

1. Download archived data from Udacity website. The file was saved as twitter-archived-enhanced.csv and contained tweets from the WeRateDogs twitter account and some additional columns that have attempted to extract data from the tweets.
2. Use a GET HTTP request to a server to ask for a file named image-predictions.tsv file. This file contained predictions on whether the pictures in WeRateDogs were dogs and what breed they were.
3. Collect more information of the tweets in twitter-archived-enhanced.csv file from the TwitterAPI using tweepy and python. We store the JSON responses in a twitter_json.csv file and parse it later on to create the twitter_json_refined.csv file.

II. Assessment

Once the data was collected, it was crucial to understand the issues in every dataset. The challenge here was to think how to clean the data to accomplish insights. This required:

1. Visual assessment. Useful to get familiar with the data and to find tidiness (or messy data) and typos problems (quality issues)
2. Programatic assessment. Helpful to find further quality issues such as wrong data types, missing data, and typos

Once an issue was found, it was listed under its corresponding dataset, but sometimes it needed to be listed under more than one dataset.

III. Cleaning

After the assessment, it was crucial to start cleaning data using three steps per issue:

1. Define. Describe the issues and how one plans to solve it.
2. Code. The actual programatic implementation to clean the issue.
3. Test. A form to ensure the fix was done properly and that it yield the expected result.

However, I first started cleaning data and soon realized I was not getting anywhere. Thus, it was crucial to create a plan for cleaning all the issues. It was also extremely important to realize that it is better to start with tidiness issues since it can help simplify the quality cleaning process. Based on the assessment, the plan to clean WeRateDogs data was:

1. Create a single _stage_ column
2. Create columns for _prediction_, _probability_ and _IsDog_
3. [Optional] Split the text column into web address and text

Once the stage and predictions are in a tidy format, there will be quality issues to clean before merging

4. Choose the rows of predictions with highest probability (eliminates duplicated tweet_id data)
5. Eliminate the non-dogs records
6. Convert the tweet_id from int64 to str
7. Merge all tables together and drop the necessary columns from each* _(merge solves tidiness issues)_
8. Drop retweets
9. Drop columns with null values
10. Clean source
11. Fix identified names (if they were not dropped)
12. Make the invalid names (a,an, such, ..) a single tag Nameless
13. Make breed all lowercase to have same style
14. Fix the rating numerator and denominator when denominator is different than 10

With the plan above in place, it all got easier to follow and I did find a path to keep moving forward without getting lost in the cleaning process.

IV. Storing Data

Finally, I stored the clean dataset with the name twitter_archive_master.csv to be able to use it in the future without having to go through all the wrangling process.

From the clean data, I started my analysis to get interesting insights. This part helped me understand the importance of cleaning and how easy it was to work with clean data to make outstanding insights and visualizations.