Wrangle Report

Edim Ernest

27th June, 2022

Introduction

Data wrangling is the process of taking raw data and making it ready for analysis.

It involves a process of applying data science methods to organize and clean raw data, and getting it ready for analysis and gaining insights about data. Data is everywhere, but unstructured data makes analysis very difficult. The end goal of data wrangling is to extract the needed information necessary for making informed business decisions and having better understanding about a company or an entity.

Data for wrangling could be obtained from existing database or a publicly available webpage or dataset. This is done so that useful information can be gained in order to solve a problem or answer an important question. Over time, data wrangling has evolved into a highly sophisticated art form.

Gathering

For this project, Udacity made available data from the @dog_rates twitter handle and we are required to wrangle the given data, query some more via twitter API.
In my case, I downloaded the twitter_archive_enhanced data from Udacity as well as the image_prediction data programmatically.
I applied for twitter elevated access which was approved but querying with the provided API for more data such as retweet_count and favorite_count was a bit of a challenge. It was based on this that I had to make use of the available .json_txt file provided by Udacity to complete my wrangling process.

Assessment Phase

After the gathering phase of the wrangling process, I assessed the data for quality and tidiness issues. I tackled these issues mostly programmatically and with visual assessment support. Some of the issues are:

- Wrong data type for tweet id.
- The use of underscores in the p1, p2, p3 prediction columns.
- Converted p1_conf, p2_conf and p3_conf values to percentages
- Columns like in_reply_to_status_id, in_reply_to_user_id, retweeted_status_id, retweeted_status_user_id in the dataframe have NAN values
- There were many retweets in most rows
- Inaccurate data types for tweet_id, timestamp and rating_numerator
- Some dogs are represented by inaccurate names like 'None' in the name column
- Inaccurate values in the rating_numerator and rating_denominator columns.
- The source column is a bit untidy because of the links which made it difficult to read.

I fixed the quality issues for the three dataframes of data obtained first before moving ahead to tidiness issues. Focusing on one data at a time while assessing them helped me gain more details to be clean so as to obtain usable data for analysis.

Cleaning Phase

The cleaning phase of this exercise is the most interesting. After the assessment phase, I made copies of each of the three dataframes so as not tamper with the original datasets. This will ensure I can I always go back to the original data should I have any need to do so.

I handled the datatype of each variable and converted them to their appropriate datatypes, like converting timestamp datatype from object to datetime which will ease the calculations for this variable. Other cleaned issues are:

- I cleaned issues such as extracting dog stages from different columns to a single corresponding column
- I converted tweet_id columns data type to string
- I removed unnecessary columns and rows like **retweeted_status_id** and their likes
- I replaced underscores in the p1, p2, p3 prediction columns.
- p1_conf, p2_conf and p3_conf values should be percentages and not proportions which I did
- I also converted timestamp column to datetime
- There are two values in the timestamp column: date and time and I split them accordingly

tweet_df id column was renamed to **tweet_id** to match the **tweet_archive_enhanced** dataframe before I could merge them to create the expected master dataset.

Finally, this was a huge opportunity for me to learn and carry out a project that has clearly widen my understanding about wrangling processes.