WeRateDogs

Data Wrangling - Taryn Lyons

This project serves as the final project for the Data Wrangling module of the uDacity - Intro to Data Analytics nanodegree. This project involves gathering data from three different data sources, all utilizing different formats, and converting this information to multiple data frames. From here, the data is assessed for multiple quality/tidiness issues, cleaned and combined into one master data frame. After the master data frame is stored into a .csv, analysis (including visualization) is performed. The goal is to gain meaningful insight from our data set.

This project has proven challenging. I struggled with gathering the twitter API data immediately. After signing up for the twitter developer account, I realized that the method described in the project documentation was no longer current, or at least didn't apply to the developer account that I had obtained. I was unable to use tweepy.API to query, instead I had to use tweepy.Client. The main difference I discovered is that .Client does not pull the entire tweet data the way that .API does. I had to query the 'public_metrics' field in order to grab the information needed. Additionally, I was struggling with pulling the tweet_id from .Client in the appropriate JSON format. I worked around this by iterating through the tweet_id list and appending the current ID into the file. Obviously, there were tweets in the archive that did not have corresponding API tweet metrics. In order to only append the tweet_ids that corresponded to the data in the API, I nested the process inside my IF statement. I instructed the statement to pass over appending any ID's that did not have a corresponding API tweet metric. This is the issue that took the longest for me to work through.

The assessment phase went smoothly. I was able to immediately identify 8+ quality and 3+ tidiness issues. I had to pull myself away from the assessment phase, for the sake of time. There were a few issues that I identified but did not end up cleaning, and many more issues that I didn't end up identifying, I'm certain. I tried to think about the data that I was interested in analyzing while I was choosing which issues to clean.

Cleaning the data was challenging as well. There were several instances in the clean process that required me to use regex to extract parts of a string. I am not well versed in regex, so this took some time to get working. There were a couple instances where I manually updated values for some fields. I couldn't figure out a way to automate it with python (or any libraries). It seems like a common enough issue that there must be a way to do this automatically.

The last step in my wrangling process was to merge the 3 data frames into a master frame. I found this straightforward. The pandas Merge function operated enough like a SQL join that I did not have very much trouble. Then I saved this into a .csv for easier reference in my analytics step.