Wrangle and analyze WeRateDogs Twitter data

Gathering data:

The first dataset the file for WeRateDogs Twitter archive (twitter_archive_enhanced.csv) was given by Udacity to download.

The second dataset the tweet image predictions (image_predictions.tsv) is hosted on Udacity's servers and I downloaded programmatically using the Requests library using requests.get().

The third dataset was gathered using Twitter API using Python Tweepy library in a file called (tweet_json.txt) file. Pulling tweet ID, retweet count, and favorite count from the json file.

Assessing data:

After gathering each of the above pieces of data, assess them visually and programmatically for quality and tidiness issues. Using Pandas functions .info() .duplicated() .describe() .count() etc.

Quality issues (content issues)

twitter_archive table

- Delete retweet.
- Clean the name column from the letters & unreal names, some names in the column are wrong such as (the, an, a,).
- Timestamp have (+0000) need to be removed because we only need (hour, minute, second).
- Drop unused columns for the analysis.
- Converting timestamp to datetime format which expose convenient attributes.
- Converting tweet_id to object in order to merge tables together.

image_predictions table

- Some Columns names should be more descriptive without shortcuts.
- Inconsistency in uppercase and lowercase names.
- Drop duplicates in jpg_url.
- Converting tweet_id to object in order to merge tables together.

Tidiness issues (structural issues)

- Doggo, floofer, Pupper, Pupp put those headers in one column.
- Merge the tables together as one dafaframe.

Cleaning Data:

Before cleaning I created copy of each dataset with .copy() function, in order to compare both datasets later on after cleaning.

```
twitter_archive_clean = twitter_archive.copy()

image_predictions_clean = image_predictions.copy()

tweet_json_clean = tweet_json.copy()
```

All of the issues mentioned above was fixed programmatically, following this order for each issue (Define issue, Code, Test) to make each issue to be identified easier. I used Pandas functions to the cleaning process such as .drop() .merge() .rename() etc.

Store the DataFrame in a CSV file:

The last step was storing the cleaned dataframe in a CSV file using .to_csv().

Conclusion:

Taking Udacity data analyst nanodegree to start new career path with very limited knowledge about coding, coming to this stage I can say udacity did expand my knowledge in many areas such as coding, analyzing, research and more. Handling this project was challenging but I enjoyed it since it was real world data that need to be cleaned.