WRANGLE AND ANALYZE DATA PROJECT 3

Wrangle Report

Introduction

We are going to walk through **the data wrangling process** (Gathering - Assesment - Cleaning) on *WeRateDogs twitter account*, and then trying to figure out some insights and create some visualizations..

The first step of the **wrangling process** is:

Gathering

I have obtained the data of the project through three different resources:

1. The WeRateDogs twitter archive: I downloaded this data through a file that called “twitter-archive-enhanced.csv”, from the resources section to our environment using pandas library. And then initialize it to data frame named “twitter_archive”.
2. The tweet image prediction file: It’s a table of information that include a data about the images of the tweets. Also running these images of the dogs in an image prediction model (neural network) to predict the breed of the dog. I downloaded this data from a web link using request library. And then I wrote the content of this file in file called “image_predictions.tsv”. Then upload it in a data frame of the same name.
3. Twitter API: Using the tweet ids in the WeRateDogs Twitter archive, We queried the twitter API for each tweet's json data using python's tweepy library. And we take retweet count and the favorite count from the tweet’s data. Then we store each tweet's json data in a file called “tweet_json.txt” file. After that, we extracted the json data from the text file and upload it in a data frame called “tweet_data”.

The second step of the **wrangling process** is:

Assessing

I assessed the data through two ways:

1. Visually: We have assessed our data by only going through the data visually and scrolling through it. Trying to explore some issues.
2. Programmatically: We tried to explore our data by using some code to view specific portions and get some summaries of the data.

And we figured out some issues we need to solve and clean:

Quality Issues:

"twitter_archive" table:

1. Wrong data type of "timestamp" date.
2. Unusual dog names like 'a' or 'O' (less than 3 characters).
3. None values in the dog stages and their "name" instead of NaN.
4. "source" column has html tag.
5. Dog stage and "source" column should be categorical data type not string.
6. We only want original ratings (no retweets) that have images.
7. Unuseful columns "in_reply_to_status_id" & "in_reply_to_user_id" & "retweeted_status_id" & "retweeted_status_user_id" & "retweeted_status_timestamp" & "timestamp".
8. Duplicate "expanded_urls".
9. Missing values in "expanded_urls".
10. Invalid ratings in column "rating_denominator".
11. Invalid ratings in column "rating_numerator".

"***Image_prediction***" table:

12. Duplicate "jpg_url".

Tidiness Issues:

1. “retweet_count” and “favorite_count” should be in “***twitter_archive***” table.
2. Better to be “***Image_prediction***” table with “***twitter_archive***” table.
3. “doggo” &”floofer” & “pupper” & “puppo” should be in one column (dog stage) in “***twitter_archive***” table.
4. “timestamp” column should two separate column “date” and “time” in “***twitter_archive***” table.

The third and final step of the **wrangling process** is:

Cleaning

I started the cleaning step by merging all the table (Tidiness issues 1 & 2), since this will help so much and will make the cleaning easier.

And then I started solving the other tidiness issues one by one, because it will make the cleaning smoother and much easier...

I perform the melt function to solve the third issue of the tidiness and make them all in one column.

After that, I split the content of the timestamp which is date and time and put them in new columns. And in the meantime, I convert the data type of “date” column to date data type.

Then I tried solving the quality issues and started with fixing the Duplicate “jpg_url” by dropping them. And I solve the unusual dog names like 'a' or 'O' (less than 3 characters) by the same way.

Moreover, I solved None values in the dog stages and their “name” by indexing the None value row, and trying to use loc indexing to assign those values NaN.

Also, I use the re (regex) python library and perform the findall function to solve html tag issue in “source” column.

And I changed the data type of the dog stage and “source” column to categorical data type by implementing astype method.
And then We only take original ratings (no retweets) tweets that have images.
After that, we dropped some Unuseful columns that we don’t need it (Issue #7).
And fortunately, we implicitly solved the duplicate and missing values of “expanded_urls” when we merged the tables together.
Furthermore, we solved the Invalid ratings in column “rating_denominator” by assign all the values of the column to 10. Since it’s the standard and the most common rating and also to remove the outliers.
Eventually, we finish our cleaning by changing the numerator of the rating to let it have values between 4 and 25. Because we found that this range is the best range of the rating of numerator and to remove the outlier values.

At the end, we store our clean data in a csv file “twitter_archive_master.csv”.