Data Wrangling Report

Project Details

Your tasks in this project are as follows:

- Data wrangling, which consists of:
  1-Gathering data Downloading files manually , programmatically and obtaining data from twitter api
  2-Assessing data
  3-Cleaning data
- Storing, analyzing, and visualizing the wrangled data
- Reporting on
  1- The Data wrangling Process
  2- The Data analyses and visualizations

Data Wrangling:

1)Gathering Data:

We needed three files for this project :

- The WeRateDogs Twitter archive this file was manually downloaded through the following link: twitter_archive_enhanced.csv (https://d17h27t6h515a5.cloudfront.net/topher/2017/August/59a4e958_twitter-archive-enhanced/twitter-archive-enhanced.csv)

- The tweet image predictions, i.e., what breed of dog (or other object, animal, etc.) is present in each tweet according to a neural network. This file (image_predictions.tsv) is hosted on Udacity's servers and should be downloaded programmatically using the Requests library and the following URL: https://d17h27t6h515a5.cloudfront.net/topher/2017/August/599fd2ad_image-predictions/image-predictions.tsv

- Each tweet's retweet count and favorite ("like") count at minimum, and any additional data you find interesting. Using the tweet IDs in the WeRateDogs Twitter archive we searched the Twitter API for each tweet's JSON data using Python's Tweepy library and stored each tweet's entire set of JSON data in a file called tweet_json.txt file. This file shall be recalled as a .txt file line by line into a pandas DataFrame with (at minimum) tweet ID, retweet count, and favorite count.

2) Assessing Data:

After the three pieces of data have been obtained it's time for applying visual and programmatic assessment in order to detect any issues regarding the data quality to get the data ready for the cleaning process.

The Functions that have been used for assessment are as follows:

- The .head() function on each of the data frames in order to view the column names and detect any visual issues in the data
- The .info() function in order to demonstrate the type of each column of data and detect any problems in each columns data type , moreover this function is used for counting the number of null values in each column .
- The .describe() function to calculate the maximum and minimum values of the numeric data in order to make sure they are within certain limits.
- The .loc() function have been used to search the data for abnormal values of data.
- The .list() function was used to list the name of the columns of each data set.
- The .value_counts() function has been used to count the number of occurrences of each value in a certain column to detect errors in the data like : NaN values were Written as None .
- The .duplicated() function has been used to detect any duplicate rows.

3)Cleaning Data :

I started by making a list of the quality and tidiness issues to have a clear image of the steps that are going to be taken in order to present the data in a clean and professional manner.

Observations

Quality Issues:

- rating_denominator should be equal to 10 .
- rating_numerator should be more than 10 (the rating can't be less than 1) and less than 15 (15 is the highest rating ever given to a dog on weratedogs).
- rating_numerator,rating_denominator columns should be of a float type rather than int.
- missing values in the doggo, floofer, pupper, puppo,name columns should be written as NaN not none .
- timestamp is given an object (String) type istead of datetime.
- Many columns are nearly all null values so they should be removed.
- should rename id column in df_tweet_json to match the column names in the other 2 datasets.
- Remove img_num as it is not useful.
- we are only intrested in the original tweets so retweets are not important.
- we are only intrested in the original tweets so replies are not important.
- The extended_entities , expanded_urls have null values.
- Some columns are not useful or contain mostly NaN values.

Tidiness Issues:

- doggo,floofer, pupper, puppo can be in one column.
- We can combine all of the data into one set.
- Replace rating_denominator and rating_numerator columns with a single rating column

Now, we can start cleaning.

The methods used for cleaning are :

- .copy() function to create copies of the data we are about to clean.
- Setting the values of the nominator and denominator to their appropriate values and eliminating irregularities .
- .astype() function to Change the data type of the rating_ nominator and rating_ denominator columns to float.
- Replacing The rating_ nominator and rating_ denominator columns with a single rating column by dividing their values.
- .to_datetime() to change the time format to datetime.
- .drop() function to drop columns with majority missing values and unnecessary columns .
- .Rename() function to rename the id column in the last dataset to tweet_id.
- Merging the doggo ,floofer , pupper,puppo columns in one column dog_persona.
- .merge() Merging the Datasets into one clean dataset to be used for visualizations.