DATA WRANGLING STEPS FOR WERATEDOGS DATASET

Data Wrangling is usually the first step in data analysis and very critical in providing error-free meaningful insights. It consists of three main steps:

Gathering Data

Assessing Data

Cleaning Data

This project addresses a subset of data wrangling issues for the WeRateDogs dataset which is explained in detail below:

Gathering Data:

Three separate datasets are required for this analysis.

- The first dataset is a .csv file which is directly imported through Pandas library (tweet_archive).
- The second dataset is a tsv file which is downloaded from the internet using requests library through a url provided. It is then opened in 'wb' mode which indicates that the file is opened for writing in binary mode. When writing in binary mode, Python makes no changes to the data as it is written to the file. Following this step, the file is then loaded for analysis using the Pandas library (image_prediction).
- The third dataset for analysis needs to be gathered by querying the Twitter API using the Tweepy library. Since twitter data is in JSON format by default, the data is parsed using JSON parser. For each tweet_id in the dataset 'tweet_archive', a set of attributes {favorites, retweets, followers, user_favorites, date_time} are extracted and appended into a new empty list called tweet_list. This list is converted to a data frame, written into a .txt file and then loaded using Pandas for analysis.

Assessing Data:

This step basically involves getting an idea of the existing quality and tidiness of the datasets. For all datasets random samples are analyzed to understand the layout and structure. The .info() function gives more information on missing values, number of rows and all the different columns. The .describe() function gives an idea of max and min values in numerical columns. Number of original tweets, retweets to be discarded and duplicate rows are also identified for all datasets.

For 'tweet_archive' dataset, using value_counts on the rating columns tells us the number of each numerator and denominator ratings present and gives us an idea of what can be

eliminated. Using the same function on source column tells us that data in this column is hard to read. Rows with incorrect numerator ratings are also identified.

Cleaning Data:

- *Quality*:

The common steps to cleaning data are to replicate the dataset to be cleaned and testing each step after cleaning to re-confirm if it has taken place the right way.

Tweet_archive:

- All entries with a Null retweeted_status_id are retained, eliminating all retweets.
- Several columns which are not required for analysis are discarded using the drop function.
- All null values indicated as 'None' is converted to NaN using replace function.
- The 'Source' column is cleaned to be made more legible using re library.
- Tweet_id, being the common column among all datasets, we ensure that its datatype is consistent across all datasets as Int datatype. If not, it is converted to Int.
- Denominator is made consistent across the dataset by eliminating rows with denominators other than '10', as this is a small number of rows and is negligible.
- Numerator and Denominator rating columns are converted to float datatypes using astype(float).
- Numerator ratings with wrong values are replaced and corrected with the right decimal value ratings provided in the 'Text' column.

Image_prediction:

- Duplicate rows are deleted.
- A single column is created for the dog_type prediction and for the confidence level of prediction using a function that parses through all the Boolean value columns to identify the True prediction and captures the associated confidence level. This information is then added as new columns to the dataset. Rows with no true prediction is discarded.
- Several unnecessary columns are eliminated using .drop().

- *Tidiness:*
    - A dog_stage column is created by combining all 4 columns (doggo, floofer, puppo, pupper) using string concatenation following by replacement of irrelevant characters.
    - All three datasets are merged using left join into a single dataset resulting in a clean dataset, rows with missing values are eliminated. This is then stored as

twitter_archive_master.csv file which is then loaded using Pandas and used for further analysis.