Wrangling Report

Wrangling is an iterative process of gathering, assessing and cleaning data.

I used Jupyter notebook in carrying this wrangling process together with Microsoft excel for the assessment phase.

I started out by importing the required libraries for my wrangling.

The First step in this wrangling is the **Gathering phase**;

Here, I gathered data from three different sources.

-The we rate dogs twitter archive.

- Image prediction.

- The twitter API dataset.

The we rate dog twitter archive was a ‘’csv’’ file so I uploaded and imported it into a DataFrame using pandas in python.

The image prediction dataset was downloaded with the url link that was provided Using Request and os library in python to create a folder. It was then opened with a context manager and was saved as a ”tsv” file. I then uploaded it and imported it into a DataFrame using ‘’read_csv’’ and a separator in pandas.

The API dataset was a json file. I used ‘’read_json’’ to upload this dataset into a dataframe in pandas.

The second step was the **Assessing phase**;

In this phase I Visually assessed my data by storing and uploading it in my Microsoft excel software, where I noted some issues like

- Some names in this dataset are specifically 'a' and 'None'

-The doggo, floofer, pupper and puppo are in different column instead of one.

I then programmatically assessed them for further quality and tidiness issues and documented them in my jupyter notebook.

The Third step in my wrangling phase was the **Cleaning Phase;**

I first started by creating copies of my Three gathered datasets.

Then I started cleaning each quality and tidy data issue that I documented in the assess phase of my data wrangling process.

- In the archive dataframe

- - 1. Some names in this dataset are specifically 'a' and 'None' and further assessment showed there were other names in lowercase that are not actual names of dogs. -**I cleaned this by dropping the lowercase names.**
- - 2. There are 745 entries in the name column that is 'None' – **I ignored the None as the entries were a lot**.
- - 3. The source column can be cleaned properly to display the different types of values that are just before the final tag'</a>' **- I used regex to extract the text just after the last tag.**
- - 4. The 'in_reply_to_status_id', 'in_reply_to_user_id' have only 78 non null values
- - 5. The 'retweeted_status_id','retweeted_status_user_id and retweeted_status_timestamp columns have only 181 non null values
- **I handle 4 and 5 by dropping rows containing the retweet and replies to get only original tweets.**
- - 6. The timestamp column is in object type – **I changed the data type from object to a datetime using ''to_datetime'' in pandas.**

- - 7. After excluding tweets with retweets and replies.There are 17 tweets with denominator not equal to 10. – **I dropped the rows with denominator not equal to 20.**

  df_prediction

- in the df_archive dataset. Missing 281 entries.(This issue will not be cleaned)

  df_api

- - 9. I only need the 'id','retweet_count'and 'favourite_count' columns – **I extracted only these columns and reassigned to the original dataset.**

Tidiness Issues

df_archive

- - 1. The doggo,floofer,pupper and puppo are in different column instead of one **– I melted this into one column using pandas melt function.**

  df_api

- - 3. This table should be combined with the archive **– I used merged function on this dataset.**

df_prediction

- - 4. The P1,P2 and P3 columns all contain the same type of data and should be in a single column.(This issue will not be cleaned.)
- - 8. There are 2075 entries in this dataset as compared to the 2356 entries. This was later rectified during the cleaning.

Further cleaning was on reorganizing columns and renaming the columns.

Stored stage- I concluded by storing the two datasets.