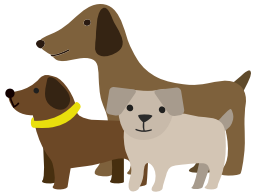
Data Wrangling Report



Noha Fayed

October 2020

Data Wrangling Report

As an assignment for the Udacity Data Analyst Nanodegree; this report illustrates the main steps involved in the data-wrangling of Twitter account "WeRateDogs".

→ Data Gathering

In this step, collecting data takes place. For this project, there were three main sources for the data to deal with:

1- **twitter_archive_enhanced.csv:**

This file was uploaded manually to our working directory and then imported into our working environment using Pandas function "pd.read_csv".

2- **Additional Data via the Twitter API**

It gathered from twitter REST API via the tweepy library by querying the API to obtain extra information pertinent to the tweets' ids in the first file, e.g. retweets count and favorite count aspects.

3- **Image_Prediction.tsv:**

This file contains image predictions (the top three only) alongside each tweet ID, image URL, and the image number that corresponded to the most confident prediction. It has been hosted on a webpage and downloaded from its relevant URL, using the request library get function and pd.read_csv pandas' function.

→ Data Assessment

After gathering each of the above pieces of data, we assess them visually and programmatically for quality and tidiness issues.

1- The visual assessment done on spreadsheet application like excel and then the programmatic assessment is conducted in Jupiter notebook.
2- Completeness issues were addressed first then tidiness issues were addressed to facilitate addressing the rest of quality issues, e.g. Validity, Accuracy, and consistency issues.
3- Some of the data cleaning efforts were guided by the scope of the project that mandated the exclusion of retweets, replies, and tweets featuring no images.

→ Cleaning Data

Each of the issues we documented while assessing was cleaned, as illustrated in the following tables:

`Quality issues: twitter_archive`

Issue Type	Issue	Solution
Completeness issues	Some tweets are replies.	-Replies records were removed using in_reply_to_status_id column, then in_reply_to_status_id and in_reply_to_user_id columns were droped
	Some tweets are retweets.	- Retweets records were removed using retweeted_status_id column, then retweeted_status_id, retweeted_status_user_id, and retweeted_status_timestamp columns were droped
	Nulls in expanded_urls column represent tweets without images.	The records with null in expanded_urls column were removed
	Some tweets with no images.	- The records from twitter_archive_clean with tweet_id not in Image_predictions_cleantweet_id column were removed. - The records from api_clean with tweet_id not in twitter_archive_cleantweet_id column were removed. - The records from Image_predictions_clean with tweet_id not in twitter_archive_cleantweet_id column were removed.
Validity issues	tweet_id is float not string	tweet_id datatype was changed to string.
	timestamp is string not time	timestamp datatype was changed to time.
Accuracy issues	Erroneous data in name column such as a, an	- 'a' and 'an' were replace with the proper name.
consistency issues	Null values for columns name, doggo, floofer, pupper,	-'None' in name column was replace with np.nan.

	puppo represented with 'None' as a string.	

`Tidiness aspects:`

DataFrame	Issue	Solution
api_df	Tweets observational unit is stored in multiple DataFrames (twitter_archive_df, api_df)	Merge the favorite_count and retweet_count columns to the twitter_archive_clean DF, joining on tweet_id.
twitter_archive	doggo, floofer, pupper, or, puppo are column values not columns names	-replace None with '' in the four columns. -merge the four columns using the operator '+' to form column 'dog_stage' -use '-' to make: 'doggopupper', 'doggopuppo', and 'doggofloofer' in the 'dog_stage' column more readable. -replace '' with np.NaN in 'dog_stage' column -finally drop the four columns.
Image_predictions	p1, p1_conf, p1_dog, p2, p2_conf, p2_dog, p3, p3_conf, p3_dog are column values not columns names.	renaming columns to be more meaningful then use pd.wide_to_long method for the reshape process.

→ Cleaning Data

→ The result is two high quality and tidy master DataFrames twitter_archive_clean and Image_predictions_clean.