Data Wrangling Report

Michael Stephenson

Udacity Data Analyst

Introduction:

The Data Wrangling for this project is built by Data Gathering, Data Assessing, then Data Cleaning to form the data into useful information to make observations about. Once complete the data will be easy to access and analyze for project purposes. The data will be perpetually stored on csv files as the cleanliness and tidiness of the data is suitable.

Gathering Data

A. The Twitter Account Archive, “We Rate Dogs” stored with in “twitter-archive-enhanced.csv”
B. URL retrieval of the “image-predictions.tsv” file
C. “tweet-json.txt” supplied by the Udacity site.

Project Library included:

```
import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import tweepy

import os

import json

import time

import seaborn as sns

import datetime

import re

import warnings

from IPython.display import Image

from functools import reduce

import requests

from wordcloud import WordCloud, STOPWORDS, ImageColorGenerator

%matplotlib inline
```

Steps Gathering

Import the CSV File of Twitter Archve:

df_archive is created to store the data set to Data Frame from twitter-archive-enhanced-2.csv by using pd.read_csv() from the Pandas Library.

Download Image predictions file from Udacity.

- url="https://d17h27t6h515a5.cloudfront.net/topher/2017/August/599fd2ad_image-predictions/image-predictions.tsv"

  response = requests.get(url)

- The data link is read from the server and stored in the Data Frame **image_prediction**

Import JSON file from Twitter API

- **'tweet-json.txt'** is written with pd.read_json into the Data Frame **tweet_list_json** via the Twitter API.

Assessing Data

Quality Issues Identified

Quality

- tweet_id's are sometimes integers or floats (numeric)

- "in_reply_to_staus_id, in_reply_to_user_id" and "retweeted_status_id, retweeted_status_ user_id "are numeric

- retweets are present in the data

- "timestamp" and "retweeted_status_timestamp" are not a datetime variable

- "source" values are formatted as <a> href=url <a/>

- rating_numerators and rating_denominator are integers instead of float

- the dog names not standardized

- timestamp column +0000 not necessary

- dog names contain: "'such', 'a', 'quite', 'not', 'one', 'incredibly', 'mad', 'an',

  'very', 'just', 'my', 'his', 'actually', 'getting', 'this',

  'unacceptable', 'all', 'old', 'infuriating', 'the', 'by',

  'officially', 'life', 'light', 'space'"

- there are 4 columns for doggo, floffer, pupper, and puppo but should be one column

-"tweet_id" and "tweet_id" are numeric and not categorical (string)

- 2075 tweet ids present. archive dataset has 2356 ids, 2075 - 2356 = 281 IDs are missing)

- p1, p2, and p3 contain underscores instead of spaces in the labels

Tidiness Issues

Structure (tidyness)

- more than one stage is filled for a particular dog

- "source" and "expanded_urls" have several informations inside them

- columns "doggo", "floofer", "pupper" and "puppo" refer to the same measurement unit , i.e, dog stage

- there are 3 dataframes, only 1 dataframe should be needed

Data Cleaning attempts to resolve issues identified within the Assessing section.

Functions used within this step include:

.unique()

.islower()

.replace()

.extract()

.head()

.append()

.join()

.astype()

.info()

.capitalize()

.replace()

.describe()

.merge()

.drop()

.to_csv()

Data Storage

Data is stored within the scope of the project as it is cleaned and merged into the .csv format for persistent use, using .to_csv().

Files created are:

df_archive_clean.csv

df_image_clean.csv

json_clean.csv

twitter_master.csv