**Wrangling Report**

**Author: Phuong Tran**

**Date: 06 - 14 - 2020**

**Note:** This is the report of my wrangling effort. In this project, I was doing data wrangling with dogs’ data set. Data wrangling consists of gathering, assessing, and cleaning data for the purpose of completing the fifth project at Data Analysis Udacity Nanodegree.

* ***Gather:*** There are three files that I need to gather so as to complete this project
  + Twitter-Archive:I received the twitter\_archive\_enhanced.csv through Udacity project detail and it can be accessed/interpreted using panda framework.
  + Image-Predictions: This file is hosted on Udacity's server and I used Python's requests to get accessed. Unlike csv, this file is a tsv file with tab ("\t") as delimiters.
  + Json-tweets.csv: In order to do this, I first had to get a twitter developer account and its API keys to query data. Then, using a self- made 'seperate\_tweet' function to filter out only tweets with a tweet ID. Next, I converted json object to a dictionary and saved it to 'json\_tweets.txt' file. Having that file, I then reopen it, get tweet\_id, favorite and retweet count and put it in a data frame using Panda.
* **Assess:**
  + The first thing I always do when it comes to assessing a data frame is to overview it using df.head() and df.info() to check columns' heading as well as data type for each column.
  + Next, I drop columns and rows that contain a lot of Nan, which is known as null value, or columns that I will not use or not related to the data set.
  + I will make the data set cleaner and more accurate with df.dropna(inplace = True) to get rid of all rows with Nans value.
  + I also checked for messy data such as duplicated by using df.duplicated().sum() and using drop\_duplicate() if needed.
  + I then check for each column data type and cast it to the right one if necessary.
  + The most important thing, make note of what needs to be changed.
* **Cleaning:**
  + Before I started cleaning, I made a copy of the original one to keep it clear of what I did and to avoid re-running the whole thing if mistakes are made. I will only make changes to the copy versions.
  + I used a markdown cell to describe what feature I am making change to, followed by a code cell to make the change happen. I then double check it with another code cell to test my change.
  + I follow through with my "what needs to be changed" note and that speeds up the process and leaves nothing undone behind.
  + I then merge my data sets since it share the same ‘tweet-id’
  + After finish cleaning up, I will re-inspect my final data set to make sure everything is good, the data is clean and tidy
* **Storing:**
  + I then convert it to a csv and store it in the same directory with the other files.