Wrangling Report for The WeRateDogs Project

This document highlights the summary of how I carried out a data-wrangling project.

Project Aim

The aim of this project is to demonstrate my proficiency in collecting data using different methods, accessing data, and cleaning data for use. The datasets used in this project contains information about tweets from the WeRateDogs Twitter account.

Project Objectives

1. Collect data about the WeRateDogs Twitter account using three methods.
   - Manual download.
   - Programatic download.
   - Web scraping using Tweepy.
2. Access the collected data to discover data cleanliness issues.
3. Clean identified issues using the 'Define, Code, and Test' model.

Note

The objectives; in the order listed above represent the different phases of the project. The orderliness in the objectives depicts a clean process. However, it should be noted that the different phases overlapped and were iterative (especially the last 3 phases).

Data Collection

Three different datasets were collected using different methods. The table below shows the different datasets, how they were collected and the variable name used to store them.

Title	Collection method	Variable name
twitter-archive-enhanced.csv	Manual download	tweets_df
image-predictions.tsv	Programatic download	images_df
Web scraping using Twitter API	Web Scraping using Tweepy	tweets_data

- **Manual download**

I downloaded the '*twitter-archive-enhanced.csv*' file using a web-browser. The file was read into a pandas DataFrame object using the *pandas.read_csv( )* method.

- **Programatic download**

I downloaded the '*image-predictions.tsv*' file using a programatic download method. To achieve this, I made use of the *requests* and *os* built-in Python libraries. The requests.get(url) method returned a '200' response which signifies a successful request.

- ***Web Scraping Using Tweepy***

In order to pull data from Twitter's API using the *Tweepy* library, I signed up for a Twitter Developer's account. This gave me access to the API through the use of a *consumer_key*, *consumer_secret*, *access_token*, and *access_secret***.** Pulling data about tweets from Twitter's API requires you to pass the tweet_id as an argument to an API object's get_status( ) method. The 'twitter-archive enhanced.csv' dataset has a column titled tweet_id. I used a for loop to iterate over every ID in the tweet_id column. I used try and except blocks to collect the retweet_count and favorite_count for every tweet_id.

Issues Discovered After Data Accessing

The data accessing and wrangling was done in Python.

Data Quality Issues

Title	Remarks	Remedy
Missing tweets	Some tweet_id in the tweets_df datasets could not be found because the tweets with those IDs have been deleted.	Drop records containing deleted tweets.
Invalid Timestamp values	The timestamp values have a trailing ' +0000'.	Trim values to remove the trailing ' +0000'.
Wrong data-types	Some features have the wrong data-types and will affect the wrangling process.	Change data-types to the right ones.
HTML Tags	The source column containing urls has html tags	Remove html tags
Missing Images	Some records do not have images	Drop records without images.
Invalid Dog Names	Some records have invalid dog names e.g a, this, etc.	Replace invalid dog names with np.nan

Title	Remarks	Remedy
Incorrect rating numerator	Ratings with decimals were incorrectly extracted	Use re library to extract correct ratings
Wrong datatypes for replies and retweets ID	Replies and retweets ID columns have the wrong data type (float)	Change the datatype to object (str).

Data Tidiness Issues

Title	Remark	Remedy
Trailing white spaces	The tweets_data column have trailing white space.	Rename the columns and remove white spaces
Duplicate features	Dog classes were stored in 4 different columns.	Convert the dog class columns into one column.
Remove retweets and replies	tweets_df contains tweets, retweets and replies	Filter dataset to contain only records of tweets and store in a new dataframe.
Irrelevant features	Some columns contain data about replies and retweets.	Drop irrelevant columns