

To begin my data analysis, I imported the necessary libraries and loaded each dataset into a Pandas DataFrame. The first dataset, `twitter_archive`, was already in CSV format, so I used a Pandas method to load it. The second dataset, the image prediction file, was retrieved by querying a link and using the `requests` module to send a GET request and write the bytes as a file named `image_predictions.tsv`. I loaded this file into a Pandas DataFrame named `predictions_df`. To load the last dataset, I utilized the Twitter API to create a file called `tweet_json.txt`, which was then used to create the `Required_Columns` DataFrame.

The initial cleaning process involved removing rows and dropping unnecessary columns from the `twitter_copy` DataFrame, which was a copy of the original `twitter_archive` dataset. Specifically, I dropped rows where the `retweeted_status_id` column was not null to maintain the original ratings and discarded additional columns that were irrelevant to my analysis.

Next, I focused on filtering out tweets without images by combining the `predictions` DataFrame and the `twitter` DataFrame to identify rows with null values in the `images` column. Additionally, I merged all three datasets on the `tweet_id` column and set the index accordingly.

To identify the most common dog types, I visualized the top ten using a bar chart and selected the top four for further analysis. I used the `.query` method to filter the merged dataset based on specific criteria, ensuring that the data I needed for subsequent analysis was readily available.

Throughout each step of the process, I ensured that I tested to make sure my changes were working as intended by visually and programmatically testing if the data was cleaned properly.

Overall, the data wrangling process involved importing libraries, loading datasets, cleaning and merging them, filtering out irrelevant data, and selecting the most relevant data for my analysis.