Data Wrangling Report

By Khalid AlRebdi

Introduction

The main purpose of this project is to wrangle the data of the twitter account “WeRateDogs” using python and its libraries such as pandas, matplotlib, Numpy etc.

The main purpose of the document is illustrating how the data have been wrangled in details.

Step 1: Gathering the Data

The data in this project is distributed across three different formats:

1- Twitter Archive File: this file was acquired by Udacity and provided as csv file.

This data is easily gathered on the code by using pandas function read_csv.

2- Image prediction file: This file hosted on Udacity servers and it can be downloaded using the request library.
3- Twitter API and Tweet Json file: By accessing twitter API and using python library tweetpy I was able to get tweets and store them on tweet_json file with txt format.
Note: you can get access to twitter API by requesting from the URL Apps.twitter.com.

Step 2: Assessing the Data

After the data gathering phase, we move to the data assessment phase witch is separated in two ways:

Visual assessment: I used Sample() or head() function for visually assessing the data I also used excel.

Programmatic Assessment: I used different pandas functions such as Isnull(), duplicated().

Step 3: Cleaning the Data

The cleaning part is divided into two section:

First, Cleaning Tidiness Issues

Two tidiness issues illustrated on the code which are:

1- The dog stage is dstributed across 4 different attributes: (doggo, floofer, pupper, puppo). Here the data is should be gathered on columns better than making 4 columns, this step make the analysis easier and the data cleaner.

2- We need to join the three Dataframes together (tw - images prediction - Json tweets).

Here the data should be gathered on one place. This method makes it easier to manipulate the data.

This was done using the merge function but first you need to make sure that the Id data type is the same type across all three tables.

Second, Cleaning Quality Issues.

There eight quality issues in our analysis which are:

1- timestamp is not a datetime format.

This easily done by using the to_datetime() function.

2- Some of the tweet_ids have the same jpg_url (dupplication).

This one dealtwith using drop_duplicates function.

3- tweet_id should be object instead of int because there is no need for calculation.

Done by using the function .astype('str').

4- The first name of the dog should be capital letter.

Dealt with by using the function .str.capitalize().

5- dropping unnecessary columns to our work.

By using drop() I removed unwanted columns in my analysis.

6-Removing Retweets are duplicates from actual tweets.

Done by using .str.startswith('RT') this code will look for every text that start with rt in order to make false.

7- Dropping unncessary columns in the tweet_json data frame.

By using drop() function I deleted unwanted columns on the tweet_json data frame.

8- Renaming columns of Images Prediction to clear and understandable names.

By using .rename(columns={ ‘ ’ }, inplace = True) I became able to rename column names in order to make it more meaningful.

Step 4: Storing the Data

After the data cleaning part, I needed to merge all three tables into one table called twitter.

After merging, storing the data is easily done using the function to_csv().

Step 5: Visualizing the Data

Matplotlib and Seaborn libraries offer various chart which were included in the analysis such as bar chart and scatterplot.