wrangle_report

June 10, 2020

1 Wrangle Report

1.1 Objective of the Project

- Section **??**
- Section **??**
- Section **??**
- Section **??**

Data Gathering

Data always comes from different sources. In this project, Data is collected from 3 different sources. The twitter archive is a good source of data but it does not contain complete information such as tweets count , or retweets count. We'll use tweepy to to get data from Twitter API.

First load the archive data using panda's `read_csv` method. Our data contains alot of rows to visualy assess few (3) rows, take **transpose** using `.head(3).T`

We have other data set in the form of tab seperated values of image prediction of each dogs for each `tweet_id`. We used request library to download the tab seprated values of image prediction. As our downloaded or requested file was .tsv, We used same pandas `.read_csv` method but this time used `sep='\t'` in the arguments to tell the pandas that it is tab seprated file

To gather data from Twitter API, first we have to configure and set the environment for requesting the data from Twitter API We 'll use pandas `.read_json` method using `lines=True` to read each line as row

Data Assessing #### Visual Assessment To Assess our data visualy we'll set the pandas option for displaying every row and column as by defaut pandas truncate few column to fit the data in frame

```
pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)
pd.options.display.max_colwidth = 10000
```

Programmatic Assessment After assessing the data frames visually its good practice to assess them programmaticly to check the data types and number of non null values panda's `.info()` method used and statistical description or summary of numeric data is generated using `.describe()` method. There were alot of missing values and wrong data types as well as extra or unecessary columns and dog style was in 4 different columns. While checking the `values_count()` of names of the dog it was assessed that there were alot of wrong name such as `the, a, an, all, e.t.c`. For duplicated rows `.duplicated` method was used. Following are the Quality and Tidiness Issue that was addressed in this project

Quality Issue

1. Some tweets do not contain an image
2. Timestamp is string and should be datetime
3. Tweet id should be string not integer
4. Dog stage and img_num are string they must be categorical
5. p1, p2, and p3 are inconsistent
6. Name column contains articles instead of names
7. Remove retweets
8. Some columns are extraneous and must be deleted

Tidiness Issue

1. Dog breed could be in one column instead of 4
2. All the dataframes could be fit into 1 master data

Data Cleaning Before starting the cleaning proccess create the copies of each dataframes using `.copy()` method

Tidiness Issue 1 In twitter_enhanced dog stage are in 4 different columns the must be in one column. used list expression to generate a list using four different column. and then add the new column and drop the prevous four columns of `'doggo', 'floofer', 'pupper', 'puppo`

Tidiness Issue 2 also deals **Quality issue 1**
In order to satisfy the project motivation which was prediction of dog breed from image. Inner Merge the dataset on tweet_id was required used pandas `.merge` method and perform `inner join` on `tweet_id` of all 3 dataframes. This Tidiness Issue solution also solved our 1st quality issue now our all tweets have image as well

Quality Issue 2,3,4 Different 3 qulaity issue was related to the data types of few columns. Change the data types of column, timestamp should be in DateTime and used panda's `.to_datetime` method to convert into datetime type. Although tweet id are numbers but they are not only numeric values they are specific for each id thus they should be in string object not integer and img_num should be categorical as some have 1, 2,3, or only 4 number in that column

Quality Issue 5 There was inconsistency and everything should be in lower cases in `'p1', 'p2' and 'p3'` so we replaced all *'' and '-' with spaces and we used str.lower().str.replace('', ' ')* for this purpose

Quality Issue 6 As some of dog names were wrong. from visual assessment it was noted that wrong name was in lower case. first we detect the names that start with lower case then we removed the wrong names from the name column

Quality Issue 7 few tweets were not real ratings about the dogs they were just retweets. Remove reweets from the data set using `isnull` method of pandas

Quality Issue 8 In order to make our master dataframe contains only valid or non null rows and columns removed extra columns `'in_reply_to_user_id','retweeted_status_id','retweeted_status_user_id', 'retweeted_status_timestamp','in_reply_to_status_id'` from the master dataframe using panda's `.drop` method. Following is the `info` of the final master dataframe

```
In [47]: clean_master_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1964 entries, 0 to 2058
Data columns (total 22 columns):
tweet_id              1964 non-null object
timestamp             1964 non-null datetime64[ns]
source                1964 non-null object
text                  1964 non-null object
expanded_urls         1964 non-null object
rating_numerator      1964 non-null int64
rating_denominator    1964 non-null int64
name                  1866 non-null object
stage                 363 non-null category
jpg_url               1964 non-null object
img_num               1964 non-null category
p1                    1964 non-null object
p1_conf               1964 non-null float64
p1_dog                1964 non-null bool
p2                    1964 non-null object
p2_conf               1964 non-null float64
p2_dog                1964 non-null bool
p3                    1964 non-null object
p3_conf               1964 non-null float64
p3_dog                1964 non-null bool
retweet_count         1964 non-null int64
favorite_count        1964 non-null int64
dtypes: bool(3), category(2), datetime64[ns](1), float64(3), int64(4), object(9)
memory usage: 384.0+ KB
```

Storing and Acting on Wrangled Data Now our final, clean and tidy, dataset was ready. its time to store the dataframe in csv. first we reset the indexes using `.reset_index` method and then stored the `twitter_master.csv` using pandas `.to_csv` method. Now our data is ready for analysis