wrangle report

abdo hashem

August 2020

Abstract

this document is brief about main points in wrangling the data required for dogs rating project, information doesn't include code but has an overview about the problems and possible solution for and related functions for detailed code refer to notebook for code implementation

1 gathering the data

during data gathering there was three sources of data the goal here is to convert those data into a pandas data_frame and save them as CSV file that can be loaded easily

1.1 the data sources

local file : here i used the pandas pandas.read_csv function

web files this required the 'requests' library which has functionality to download the file then used the pandas read_csv to get the data frame taking into account that we have a tsv with sepration 'sep='$\widehat{\ }$'

twiter api here i used the 'tweepy' library the loading steps are the following

1. use key from tweeter api to get authonication
2. iterate over the tweet_id
3. use tweet_id to get the json and save it in one line

finishing the previous steps we now have tweet.txt we use this to constract a dataframe by making dictionary of

- tweet_id
- number retweets
- number of likes

2 assessing/cleangin the twiter archive

in assessing the data i tried to check from most to least important problems

2.1 the rating

2.1.1 problems

the dogs rating hade sevral issues caused by poor extraction and the nature of the tweet text those problem can be summarized as following

float ratings some rates has decimal point for example 13.5/10 which will be reported as 5/10

rating like text here some text has same form as rating for example 24/7 or date like 4/20

two ratings this is most confusing as tweet photo has two dogs and the tweet give a rating for every one of them

joking another tricky rating happens when giving strangely very high or low rating then giving the actual rate examples at tweet_id 835246439529840640

> ok jomny I know you're excited but 960/00 isn't a valid rating, 13/10 is tho

2.1.2 solutions

solving those problems drives some wild choices for example the two rating case which happened many times and required looking at the photos but others where much easier and had lucky general way so the cleaning went as following

float ratings use regex that takes into account the decimal point (\d+\.?\d* / \d+\.?\d*)

rating like text this can be handled by removing those spotted rating like text. but luckily in all cases the real rating was written at the end so i had only to take the last rate at the tweet

two ratings here i only took the first rating hoping it represents the recognized dog **??**

joking had the same luck as *rating like text* and taking the last rate was enough

2.2 dog stages

2.2.1 problem

- dog stages had critical tidiness issue as the column name were used to indicate the stage by giving a dog the stage name or the None if it isn't in this stage
- another issue appears after tidying the data that some dogs had two stages or a photo has two dogs

2.2.2 solution

tidiness problem using pandas melt function we can get those stages into one column we can then remove those with None value

duplication problem following same approach as in [**?**] we can take only the first stage mentioned as it is most likely to represent the first rated dog

2.3 extra data

those data extracted using Twitter API can be merged to the archive as it represents related information to the tweet

2.4 retweets and comments

some tuples doesn't represent original tweets such as comments and retweets

2.5 extra cleaning issues

complex values in source column

2.6 data types

wrong data types in timestamp which is date time and dog stages as categorical

3 image predictions

3.1 dog breed

the main point of the image prediction data frame is to get the dog breed which was recorded as multilevel prediction about the content of image with confidence to each prediction and whither it is breed or not this represent two problems

- a tidiness issue as dog breed had to be a one column with value representing the breed itself
- dirty tweets that had no dogs at all in the three predictions

3.1.1 solution

tidiness problem we can get the dog breed easily by getting the first prediction representing a breed as we have only three prediction

no breed images those can simply be dropped