Wrangling reprot

Introduction

In this project is a practice of our skills in data wrangling, which is an import skill for any Data Analyst to can perform any analysis and get the results wished. The data wrangled is from tweet database of Twitter account **@dog_rates**, also known as **WeRateDogs**. This dataset is about the humorous comment and rating of dogs in the **@dog_rates** account.

In this report I will describe my wrangling efforts in this project.

Gathering data

The data of this project are gathered from three sources as following:

- **Twitter archive enhanced:** it’s a given data set by the Udacity Data Analyst Nanodegree instructors, this data set is the backbone of this project;
- **Image predictions:** I must gather data from a TSV file hosted in web server, I got the data by using the request python package to a TSV dataset to can read and analysis the data;
- **Twitter API & JSON:** the expected is gathering data from Twitter database by using the Tweepy library but I had a problems to get Twitter API access from the company, for that I used the given json file to gathering additional data for my project by using the Json library and create a new data set.

Assessing data

After creating three data frame from the gathering data (***twitter-archive-enhanced data frame, retweet_count data frame, predict data frame***), I explore the data frames to found the issues in each data frame. I perform a visual assessing and programmatically assessing to looking missing issues, quality issues and tidiness issues. I found the following issues:

Missing issues¶

twitter-archive-enhanced table

- Unretrieved rating data from some tweets texts.

Quality issues

twitter-archive-enhanced table

- Abnormal values for rating_numerator, rating_denominator columns;
- The timestamp columns must be date time not string;
- Keep only date for timestamp columns;
- Rename timestamp column;
- Rename timestamp, year, month and day columns;
- The tweet_id columns must be string not integer;
- Drop no dog names;
- Drop useless columns.

predict table

- Capital first letters for breeds dogs;
- Underscore for many breed dogs names;
- Rounding float numbers;
- Convert the numbers to percentage format;
- The right predictions;
- The high right prediction percentage;
- No need for source of JPG url;
- Drop useless columns.

retweet table

- Tweet_id column is a string.

Tidiness issues

- Merging doggo, floofer, pupper, puppo columns;
- Assembling many parts of 3 tables in same table.

All this issues are not an exhaustive list of all data problems in this tables, is just all what I can found in my assessing process performance.

Cleaning data

In this step, I tried to solve all issues found in the assessing step. The cleaning of each issue is performing on three steps: define, code and test.

Before starting the cleaning process, I make a copy of each table to avoid losing or changing the gathered data by mistake.

For the missed values issues detected after the visualization assessing, I tried to extract more values for numerator and denominator ratings.

Then, the more important quality problems that needed more work are cleaning the abnormal values of numerator rating and get the year, month also day in **Twitter-archive-enhanced table**. On the other hand, in the **Predict table** the most complicate quality issues are get just the right predict with high percentage prediction, also extract the dog images names. Beside this complicated issues, I performed some formatting tasks like capital letters, type of columns, deleting useless columns…

Thereafter, they are two tidiness issues in this project; I merged all three tables in one table after melting the dog stages columns on one column. This two cleaning steps are for facilitation the analysis process after.

Conclusion

I debriefed 20 issues in this project but did not mean the final table is free of issues; the wrangling data is an iterative process in future needing data analysis process.

The final **twitter_archive_master** table contain 2223 observations with 13 columns, point out same columns had not the same number of observations but not mean are a NaN observations.