Wrangle Report

10th August 2019
Kyle McMillan

Introduction

This project for Udacity's Data Analyst Nanodegree involves taking data from many different sources, wrangling the data, and producing an output, in the way of visualisations, from the data.
The main aspect of this project is about wrangling the data to produce clean data. Data for this project was done using the twitter data from the WeRateDogs (twitter: @dog_rates) account. There was also additional data supplied by Udacity that using a machine learning algorithm to predict the dog breed from the pictures for each tweet by WeRateDogs.

The wrangling aspects were broken down into 3 main parts in order:

1. Gathering the data
2. Assessing the data
3. Cleaning the data

1 – Gathering the data

This project involved gathering the data from 3 different sources that were 3 different types.

The first set of data was supplied by Udacity that had programmatically taken WeRateDogs's tweets and made a file for the various ratings and other aspects of each of the tweets. This file was in CSV format and downloaded from Udacity's website.

The second set of data was also supplied by Udacity and this data was the output of each of the machine learning breed prediction program for images supplied in WeRateDogs's tweets. This file was in TSV format and also downloaded from Udacity's website.

The third dataset was taken from twitters API for WeRateDogs. The API supplies a JSON string of each tweet's data in full. Unfortunately for this project, I was unable to use twitters API, and I used a Udacity supplied TXT file that contained the full JSON output as if it had been downloaded via twitter's API.

2 – Assessing the data

As both of the downloaded datasets had around 2000 rows of data each, I decided to programmatically asses the data. The main reason for this is because they were too large to open and scroll through using a program such as Excel.

Using a Jupyter Notebook all of the data was stored into 3 dataframes using python. I carried out a detailed investigation using the various tools that python and pandas had to offer; visually insepcting the rows using the df.head() option and looking at the types of data in the dataframes using df.info().
Also using options such as df.unique() to see if erroneous items were in various columns.

To asses the data, the data was judged on 2 criteria: quality Issues and tidiness issues.

Quality issues, sometimes referred to as dirty data, are when data is not complete; there are validity concerns; accuracy is low; or the the consistency of the data is not good. An example could be that the /10 rating denominator is not /10 and is listed as some other number: this is a consistency issue.

In this project the quality issues that I came across were:

- Retweets are included and need to be removed.
- Some columns are not needed is not needed.
- Tweet_stats_df 'id' column needs to be 'tweet_id'. Timestamp is not in timestamp form.
- Rating denominator not always /10 some need to be nominalised.
- Rating numerator not always the correct value and need to be fixed.
- Some items in predicted dataframe are not dogs.
- Only the p1 prediction will be necessary for the project as this has the highest prediction rate and other predictions can leave the data cluttered - p2, p3 to drop.
- Some tweets are a reply and have no rating.

Tidiness issues, sometimes refereed to as messy data, are in regards to the structure of the data. There are 3 main aspects that are kept in mind was investigating tidiness issues: each variable forms a column, each observation forms a row, and each type of observational unit forms a table.

In this project the tidiness issues that I came across were:

- Doggo, floofer, etc, only need to be one column.
- Only need to be one table.

2 – Assessing the data

Cleaning the data is the final step in wrangling data. This step is where the issue that were mentioned above are corrected to leave a nice, clean dataset that is ready to be used.

The data was cleaned programmatically using various modules of python. Some cases were done using programming techniques such as iterating over columns/rows to combine the dog nicknames into one column. Other cases, each item had to be individually changed, such as the times were the rating denominators were not listed out of /10. The text needed to be read for that tweet and manually changed in the dataframe.

The detailed steps that I took in order to clean the data are listed in the wrange_act.ipynb Jupyter Notebook.

Conclusion

The main reason for needing clean data is that highly accurate visualisations and conclusions can be drawn from the data. It also makes data easier to work with allowing for other people to recreate your findings from the data.