"We rate dogs" - Data wrangling project

Wrangle Report

This report briefly describes my efforts to gather, assess and clean the data needed for analysis of the "We rate dogs" Twitter data.

Gathering Data

Three sources have been used for the data analysis.

1. Tweet archive in csv format
2. additional tweet data collected via the Twitter API
3. Image prediction file loaded from a cloud server

The additional tweet data was downloaded via twitter API and stored as json strings in a local text file. The image prediction file was downloaded with help of Python's requests library. The file was stored locally. I then read the local file data into 3 raw data frames.

Assessing/Cleaning Data

I started assessing the data by opening the archive file in a spreadsheet software, where I noted several issues in terms of data quality and tidiness. There were implausible values in the two rating related columns and regular words instead of names in the name column. Probably both issues resulting from rather sloppy parsing of the tweets' texts. Even though it is quite obvious that the users consider the account's rating system (13/10, etc.) rather humorous, I decided in a quite rigorous manner to drop all rows with ratings with denominators unequal to 10 and numerators greater than 20. Also, I replaced regular words in the name column with NaN. In several columns the string "None" was used instead of panda's NaN values, so I replaced them throughout the data frame. Several columns not need for further analysis have been dropped. The project should exclude retweets and replies, so I dropped these rows depending on the entries in the corresponding columns. I also stripped the information in the source column of the unnecessary html code. Furthermore, I converted the timestamp data to datetime format. Also, I combined 4 columns for dog stages to one single column.

The additional tweet data and the data from the prediction file showed issues in term of completeness (not all rows from the archive had corresponding entries in the prediction file or the additional json file), but since there wasn't an option to complete missing data I dropped unmatched entries from the archive.

In the end I joined all 3 data frames together to one single data frame and stored the result into a local file which I imported for the analysis part.