Rating prediction on Yelp Dataset
Machine Learning for Natural Language Processing 2020

Author
Louis Pruvot-Caprioli
louis.pruvot-caprioli@ensae.fr

Abstract

The goal of this project is to predict number of stars from text reviews from Yelp Dataset available here: https://www.yelp.com/dataset/documentation/main. The original aim was to use BERT model for predictions but I did not achieve it and finally used textblob. Results of the predictions are assessed in quantitative way through confusion matrix and in qualitative way through analysis on Wordclouds. The Github link of the project is here: https://github.com/LouisChess/Yelp_sentiment_analysis/. The Colab Notebook link of the project is here: https://colab.research.google.com/drive/1xRhPmIAXNZCyPiwD81Kpi3Ue4va6xiZa.

1 Problem Framing

Online ratings have a huge impact on the consumer behavior. Before entering in a restaurant, a considerable proportion of the population is checking the Online reviews. My hypothesis is that the sentiments expressed in those reviews are likely to be predictive of the ratings. Therefore, this project aims to extract quantitative metrics from Online reviews and to compare it with the ratings. This is a classification problem with five classes, because number of stars go from 1 to 5.

2 Experiments Protocol

I chose to split the data into a train dataset and a test dataset and the train dataset were used to evaluate the proportion of elements of each class. These proportions are used to get a class number between 1 and 5 from a sentiment prediction score between -1 and 1. The textblob algorithm is providing a negative number if the text seems to include negativity. This is a pre-trained model For the wordclouds, the text preprocessing consisted of removing punctuation and english stopwords, tokenize text content and selecting only adjectives through the Spacy package.

3 Results

The accuracy was close to 50% in this 5-classes prediction problem. The confusion matrix is interesting because it shows the predictions are better in the extreme number of stars (1 and 5). I think this problem is intuitive, it is harder for the algorithm to predict classes between 2, 3 or 4 stars.
As specified in the notebook, the wordclouds are a good way to visualize why error are made on certain predictions. Wordclouds from five stars reviews linked to a good prediction are characterized by adjectives easy to receive as positive, while Wordclouds from five stars reviews linked to a bad prediction contains notably also negative adjectives. It shows the limits of word embeddingwith simple algorithms like textblob.

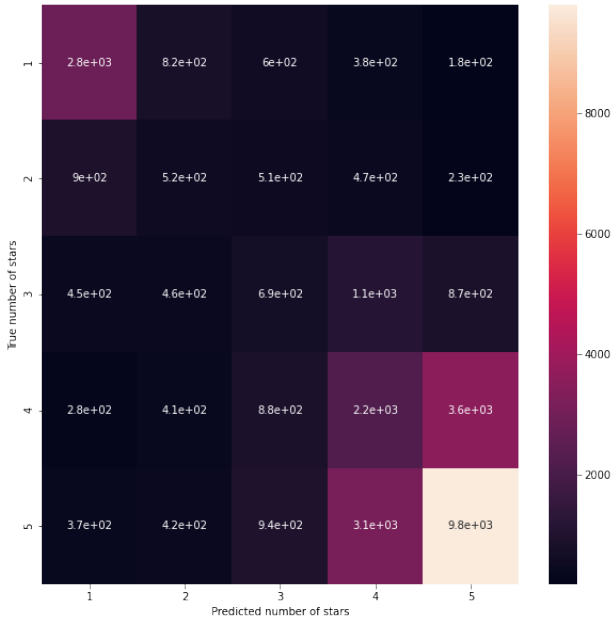

Figure 1: Confusion matrix of the predictions

Figure 2: Wordclouds on the well predicted 1 star ratings

4 Discussion/Conclusion

In the beginning I was quite disapointed to do not be able to implement a runnable BERT algorithm. But finnaly for an implementation as simple as in the Colab Notebook, the results of the predictions are encouraging. In a possible future work I would like to compare these results with a BERT pre-trained model, notably from this source https://huggingface.co/transformers/pretrained_models.html.

[1] Steven Loria, 2018. https://buildmedia.readthedocs.org/media/pdf/textblob/dev/textblob.pdf

[2] SpaCy documentation: https://spacy.io/api/doc