Sentimental Analysis for Yelp review

Group 5 - Y. Li, W. Song, W. Tam, N. Yeung

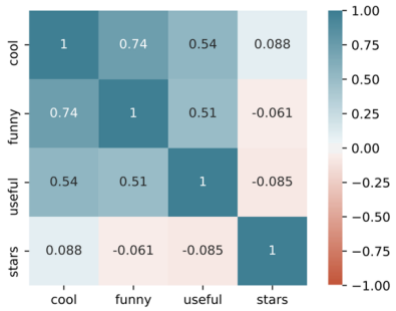
1. Objective

The objective of the project is to design a natural language processing model to predict the comment scores given by Yelp reviewers based on the features of the comment.

2. Data Exploratory Analysis

2.1 Numerical variables

Cool, *funny* and *useful* are numerical variables representing the reactions given to the review by other users. To see their correlation with *stars*, we plot the correlation indexes.



3. Data Pipeline Design

3.1 Model Design

Since the *Cool*, *funny* and *useful* attributes have little influence on the review stars according to the correlation matrix shown above, we decided to focus on the text information. We adopted the BERT-class models, both BERT-base-uncased and RoBERTa, and fine-tuned it on our training set.

BERT-class Models

BERT is a state-of-the-art bi-directional transformer which has high performance in language representation. Unlike some conventional models such as LSTM which can only learn from the previous word or the next word at one time, It leverages the attention mechanism and transformers to learn from words in all positions of the sentence simultaneously and accurately. Based on the BERT, RoBERTa is developed with improved training methodology, a larger dataset and compute power.

The three input embeddings, token embedding, segment embedding and position embeddings can not only tell the model which words are used, which sentence do they belong to, but also their positions in the sentence so that the model can be aware of the different meanings of the word regard to its position.

In the final classifier, we adopted RoBERTa for its improved accuracy over BERT-base-uncased.

Data Preprocessing

We customized a tokenize function to remove the special symbols such as '#', '\n', \r' from the original text and to turn all the letters to lowercase.

To prepare the input data for the RoBERTa model, we also used the pre-trained RoBERTa tokenizer to encode the text and converted the string to a sequence of ids. We fixed the length of each sentence by truncating the longer sentences and padding the short ones. We also created attention masks to tell the model if the id corresponds to a padding word or a real word.

All the preprocessed text and their corresponding labels were stored into PyTorch DataLoader before the training session.

Model architecture

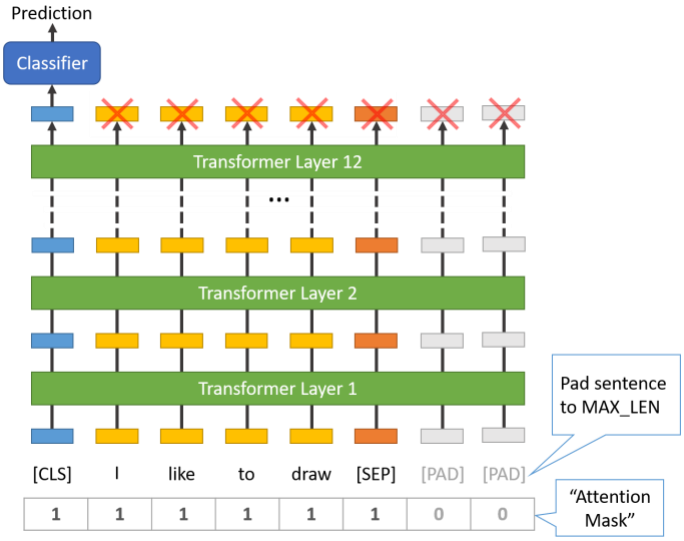


Figure 1: Model Architecture [1]

3.2 Model Training/Fine-tuning

To have the model better fit into our dataset, we took the pre-trained Roberta model and added a dense layer with softmax activation at the end for our classification problem. Adam optimizer and a linear scheduler is used to optimize the model and the learning rate is gradually reduced during training. After several experiments, we found the below settings to produce the highest validation accuracy.

- **Hyper-parameter tuning**
  1. Learning rate = 1.5e-5
  2. Number of epoch = 3
  3. Batch size = 32
  4. Max length of sentences = 256
- **Result**

  The training took around 20 minutes in total with GPU on Google Colab. Training loss (categorical cross entropy) decreased from 1.64 to 0.73 while training accuracy increased from 0.25 to 0.739. Finally our model achieved accuracy of 68.7% on the validation set after 3 epochs of fine-tune training.

4. Discussion

From trials with 'BERT-base-uncased' (see Appendix), we have noticed the validation performance does not necessarily increase with pre-processing mechanics. This highlights the contextual learning nature in the representation learnt by the BERT-class models. As stemming and stopword-removal removes some contextual information from the original sentences, it may not give a better contextual representation, thus may not imply better performance.

The optimal max length of sentences with RoBERTa model empirically observed to be 256. When considering the 90-percentile of the sentence length parameters with only lowercase and remove special symbols pre-processing is 260 (see Appendix), the optimal model is trained with the full review text of almost 90% of the training samples, which allows the models to effectively generalise contextual representations.

5. Conclusion

Our final RoBERTa model obtained an accuracy of 68.7% over the validation set.

6. Reference:

1. Singh, A. (2020, January 15). Building State-of-the-Art Language Models with BERT. Retrieved from https://medium.com/saarthi-ai/bert-how-to-build-state-of-the-art-language-models-59dddfa9ac5d