RMBI4310/COMP4332 Project 1 Report

Group 35: Jamie, Lo Sze Yuen (20619883), NANDA, Nikhil (20491384), LAI, Wai Chun (20510702), LIN Chi Wing

Abstract

This project aims to solve the multiclass sentiment classification problem on a subset of the Yelp dataset. The data consists of reviews as well as attributes such as 'funny', 'cool' and useful and the stars for each review that range from 1-5 which serve as the labels. Two different approaches for models are compared - 1) Heavyweight feature engineering-based ensemble model and 2) Contextualized word representation (BERT) based model.

1) Ensemble Model: Multi-Features + TFIDF +Logistic Regression + [Glove + CNN + RNN]

Feature Engineering / Data Preprocessing

All words from the training corpus after filtering out punctuations. The 3 attributes namely, 'cool', 'funny' and 'useful' are used as there might exist some kind of relationship between these attributes and the label-stars. In addition, we use Glove to make the embedding layer as it allows us to pre-load large word data and reuse the word similarities to calculate the weights from the original text corpus. Apart from these similarities, we also add subjectivity and polarity into the embedding index. The model also uses TfidfVectorizer to process text data into term frequency information. The TFIDF weights act as another input layer in the ensemble model [1]. It has been included as we found that TFIDF along with logistic regression is better than other classifiers that we tried, which include Random Forest, KNN, Decision Tree, AdaBoost and XGB Boost.

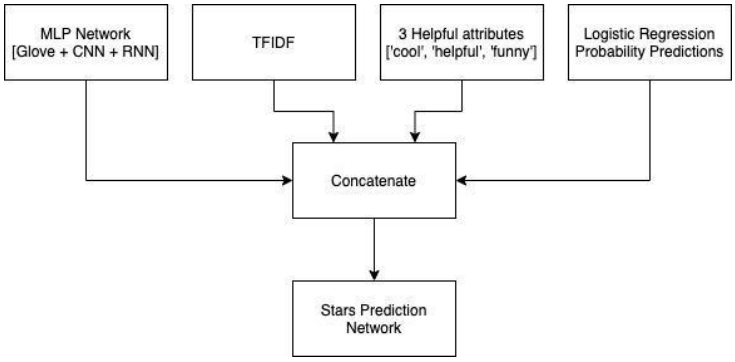

Figure 1: Ensemble Model

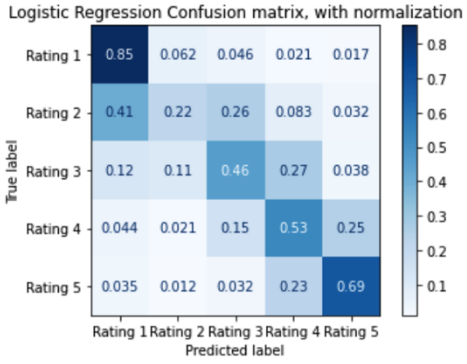

Figure 2: Logistic Regression Probability Predictions

Implementation

From Figure 1, it can be seen that the model combines multiple classifiers to make a prediction. The model contains 4 input layers as data input, which are a MLP network, a logistic regression model, helpful features ('cool', 'funny', 'useful') and TFIDF weights. Firstly, the model will calculate the probability for 1-5 stars based on the text by Logistic Regression (as shown in Figure 2). Next, it feeds the text into the embedding layer to calculate the weights provided by Glove. Then, the model will passes the embeddings to a Bi-Directional GRU network layer and then pass to 2 consecutive CNN network layers along with a maxpolling1D layer. After that, the inputs are passed to a Bidirectional GRU network layer. This concludes the MLP network architecture. Finally, the others 3 input layers are combined to output the prediction result.

2) Bert Model

Feature Engineering / Data Preprocessing

An interesting observation from the glove embeddings used in the above model was that lot of words like "unhygenic" and "wellmaintained" have no corresponding embedding and are considered out of vocabulary words. Thus, in an attempt to represent these words, the pretrained tokenizer BertTokenizerFast from huggingface has been leveraged that is able to capture sub-word embeddings [2].

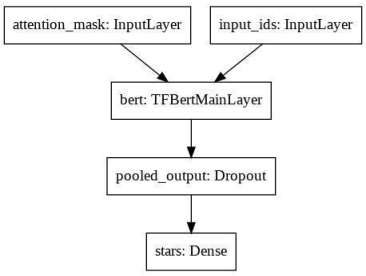

Figure 3: BERT model

Implementation

The length of a sentence is capped at size of 100 based on dataset analysis. The 2 inputs (word token ids and attention masks) output from the pretrained BERT tokenizer are passed to the BERT layer and then Dropout is used to prevent overfitting. Finally, predictions are made through a Dense layer.

Evaluation

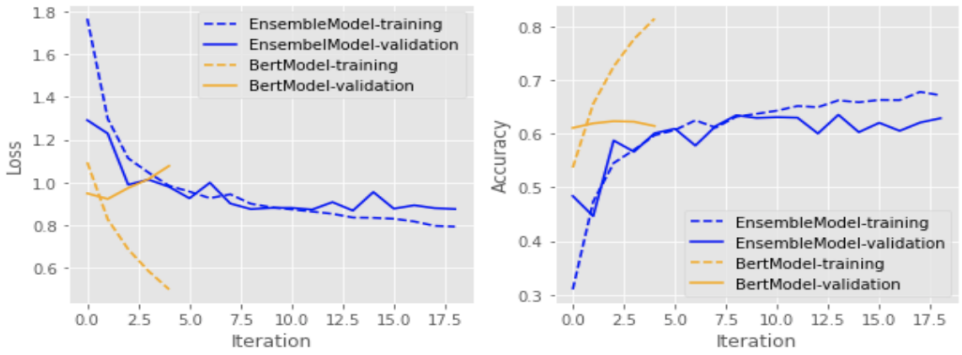

Figure 4: Model Comparison using loss and accuracy curves

It can be seen that the BERT model overfits very easily and thus is trained for very few iterations. However, the ensemble model takes many more iterations to achieve a comparable performance to the BERT model. We also noticed that both the models are able to judge whether the review is positive / negative and majority of the wrong predictions are failures in determining the more fine grained stars labels. For example, the model may predict rank 1 for a review whose true label is rank 2 and vice versa. The best Ensemble model got an accuracy of 63.55% and the best BERT model got an accuracy of 62.65% on the validation set. Since we feel that the BERT sub word embeddings might better generalize to out of vocabulary words that might be present in the test set, our final predictions use the BERT model.

References

[1] V. Ajith, “Word_Embeddings with TFIDF ensemble”, 30 March 2020. [Online]. Available: https://www.kaggle.com/ajithvajrala/word-embeddings-with-tfidf-ensemble

[2] E. L. Jensen, "Towards Data Science," 25 August 2020. [Online]. Available: https://towardsdatascience.com/multi-label-multi-class-text-classification-with-bert-transformer-and-keras-c6355eccb63a.