**Final Project**

1. Introduction

In recent years, the deep learning approach has achieved great results in all sorts of NLP tasks. In this final project, I have tried to use three different methods to do sentiment analysis. Two of them are Naïve Bayes and logistic regression which are considered as traditional approaches. And the neural net approach called BERT. The sentiment dataset is call SST-2. The full name is Stanford Sentiment Treebank and it has 65k entries. The dataset has two labels, 1 for positive sentiment and 0 for negative. Throughout the training and evaluation, I did find the neural approach has a better performance than the traditional methods.

1. Related work

The Naive Bayes and logistic regression are the old, classical method to do a classification task. The BERT is one of the most recent and best performing neural network architectures. The BERT is introduced by Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova from Google. BERT has advanced the state of the art for eleven NLP tasks. The good performance is because it has found a way to utilize bidirectional representations.

1. Experimental setup

The SST2 dataset has around 65k labeled items. The data is spitted into 80% and 20% part for training and testing. The vocabulary size is 16185.

In the Naïve Bayes part, I have used the bag-of-word representations and normalize to 1 for each sentence and then feed the feature vector into the Naive Bayes algorithms.

In the logistic regression part, I have almost done the same thing as the Naïve Bayes part. The feature extraction part is the same as the last part. The model is changed to logistic regression model. The sklearn package has implemented the L2 regularization as default, so I simply utilize it.

In the BERT part, I have use the PyTorch framework. The pytorch\_pretrained\_bert package has implemented all the tools you need to train a BERT model. To use these interfaces, you have to make the dataset format into BERT compatible formats. Since we only did the sentiment anlysis, we can simply make a dataset with four columns which are index number, label, padding column and the sentence respectively. BertTokenizer will help to tokenize each sentence into separated words. The BinaryClassificationProcessor class will process all the data into memory represented as class. Then, to make a data format for the convert\_examoke \_to\_features function to use, we have to make each input sentence class into a tuple containing sentence, label, max sentence length, atokenizer and ouput mode. The we can call the convert\_examoke \_to\_features to make the sentences ready for training.

1. Results and analysis

In this project, I have used MCC metrics. This evaluation score measures the performance of the model in range of -1 to 1. -1 means the model make total opposite prediction, 0 means the model make random prediction and 1 means the model make perfect prediction. The second metrics I have used is F1 score. This metric considers both the recall and precision and give you a single measure score. Also, I have done all the traditional evaluation scores like recall, precision, accuracy etc.

In the Naïve Bayes part, the model gets 57.6% MCC score, 76.4% F1 score and 77.2% accuracy. We can observe that this model is better than the random guess which should be 50% accuracy.

In the logistic regression part, the model gets 78.9% MCC, 90.7% F1 score and 89.6% accuracy. We observe that the model gets 12%-21% absolute improvement. We can see the machine learning approach is better than the statistical probability approach.

In BERT part, the model gets 90.1% MCC, 95.6% F1 score and 95.1% accuracy. We observe that the BERT get 5%-12% improvement over the logistic regression approach. We can see the deep neural network approach get the best performance.

1. Why BERT get such good performance?

1). Better word representations. The first two approaches use the bag-of-word representations and the BERT uses word embeddings. Bag-of-word representation will lose a lot of context information which is importance in sentiment analysis.

2). Bert utilizes the bidirectional attention mechanism. Every token can attend to every work in the sentence which make the model fully utilizes the contextual information.

3). Higher level feature representations, deeper understanding. The BERT is a neural net approach which mean it has multiple layer architectures. The multiple layers can help the model gain deeper understanding of the data which make it surpass the logistic regression in a big margin.

4). Pretraining and fine tuning. The BERT do the pretraining on lots of unlabeled data and then use these weights as the basis of the various down streaming tasks.

6. Conclusion

Trough this project, I get the directly perception of the different performance of the traditional ways to do sentiment analysis and the deep neural net ways. The BERT indeed get the best results over the Naïve Bayes and logistic regression.