NLP Project Proposal

Pingwei Sun, Hanwen Sun, Han Liu

NEU-NLP Lab

Abstract

In this stage of project, our team will build up a sentiment classification system based on the pretrained BERT model. Introduction of BERT and its features is included in this proposal, along with the description of our methods and the timeline.

1 Introduction

In the task of natural language processing, the expression of token and the semantic relationship of context are always two key sessions.

In terms of word expression, the method has developed from dictionary to vector representation, and then to today's word embedding method. In context semantics fields, people began to manage to make the model learn features from the aspect of sentences and paragraphs through the structures designed for memory, from the early RNN, LSTM and other models. Since then, some bi-direction methods have also been developed.

However, with the launch of **Transformer** (Vaswani et al., 2017), the model's ability has risen to a new level in terms of learning word similarity and contextual semantic information. At the same time, with the improvement of computing power, more and more organizations are also trying to use larger models and huge data sets to obtain better results.

After completing the pretrained word vector expression (Mikolov et al., 2013), Google launched a more powerful pretraining model **BERT** (Devlin et al., 2018) in 2018, which has great ability to represent words and understand semantic in the context.

BERT is undoubtly a milestone in the field of natural language processing, however, there are too many parameters to train. We will try some strategies and tricks in training BERT during this period of experiment.

Initiate work of BERT:

- Using transformer as the framework of the model to better capture the bidirectional information.
- Multitask training strategies and inspiration on pretraining.
- Being competent for many downstream tasks after fine-tuning.

2 experiment environment

There are 7 classes in our dataset and we need to do seven classification.About 0.04 % sentence in the dataset are longer than bert 512, we just ignore these sentences.

There are three member in our group, We use pytorch as our framework and run model on RTX 6000, which has 24G video memory and is provided by northeastern university.

3 Approaches

To build up a sentiment classification system based on BERT, the following parts needs to be done.

3.1 An efficient classifier

We will firstly build up a classifier containing a dropout layer and a linear layer using cross entropy loss. The classifier will get features from different layers of bert, With the process of our experiment, its structure may be changed slightly to correspond with our strategy. However, complex classifier may have no help to our model, so we won't spend too many time on the classifier.

3.2 Data Augmentation

The aim of data augentation is to: 1) deal with data scarcity 2) improve the generalization ability of model 3) deal with sample imbalance 4) ensure model safety.

UDA unsupervised data augmentation for consistency training aims to decrease the demand for labeled data while utilizing the unlabeled data more. It uses RandAugment and Back-translation instead of simple noising operation, this method can generate various syntax while keeping semantics unchanged.

EDA Easy data augmentation for text classification tasks, including synonyms replace, randomly insert, randomly swap, randomly delete. It is useful especially on small data sets.

Sampling Sampling means we can sample new sample from origin data distribution, including Rules, Seq2Seq models, Language Models, Self-training ways.Sampling is more dependent on tasks, we need to ensure both the variety and reliability of data.

Feature Cut-off The dimension of Bert's vector are all 768, we can set the value of some of the dimesion to zero randomly.

dropout we temporary drop some of the node in the network during the training process.

3.3 Appropriate training strategies

First of all, we need to work out a basic setting that fits our model. Actually, the basic may not perform well. To improve its performance, we can refer to the various fine-tuning methods and experiments referred in this paper (Sun et al., 2019) and do experiments ourselves.

Time permitting, the contrastive learning (Gao et al., 2021) method will be introduced to our project in order to enhance our model. Although this method is used for learining sentence embedding, it may also work in sentiment classification.

Contrastive learning aims to learn effective representation by pulling semantically close neighbors together and pushing apart non-neighbors, we encode input sentences using bert and then fine-tune all the parameters using the contrastive learning objective. The unsupervised SimCSE simply put the same sentence to the pre-trained encoder twice and obtain two different embeddings as ”positive pairs”, Then we take other sentences in the same mini-batch as ”negatives”, the simple approach are likely to get good results.

3.4 More Experiments

Time permitting, we want to do some comparative experiment and ablation experiment to explore BERT further.

4 Timeline and Assignment

4.1 Work assignment

The entire project is divided into three main parts, coding, conducting experiment , results collation and visualization, and each of our teammates will take part in these three parts.

4.2 Timeline

11st, July Write proposal, understand the BERT base code by debugging.

12nd, July Build up a classifier and improve it.

13rd, July Discuss with seniors and fine-tune the model.

14th, July learn the paper written by fudan mentioned above and try some training strategy.

15th, July Data augmentation, improve training strategies and visualize the results.

16th, July Retirement, try some new strategies and test.

17th, July Collate the results, prepare for the final report.

18th, July Presentation of our final model.

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to fine-tune BERT for text classification? In *China National Conference on Chinese Computational Linguistics*, pages 194–206. Springer.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.