Proceedings of the 4th International Conference on Signal Processing and Machine Learning
DOI: 10.54254/2755-2721/51/20241188

Comparative analysis of strategies of knowledge distillation on BERT for text matching

Yifan Yang

School of Automation, Nanjing University of Information Science and Technology, NanJing, Jiangsu, 210044, China

202183250048@nuist.edu.cn

Abstract. Large language model is a highly effective and promising language model technology that can improve the performance and robustness of natural language processing tasks. As a representative work, Bidirectional Encoder Representations from Transformers (BERT) has excellent performance in various natural language processing tasks. This model is pre-trained on large-scale language dataset and has gained attention from all walks of life since its introduction. However, its huge number of participants and scale make its performance in mobile very limited. As an effective technique to compress neural network, knowledge distillation can obtain a lightweight model with smaller parameters without losing too much model performance. Therefore, distillation of BERT models has been started, aiming at obtaining lightweight BERT models. In this paper, we will introduce several common BERT distillation models and analyse their model architecture, distillation process, and finally the compression efficiency and model effectiveness. It is concluded that the process of increasing recompression efficiency is often accompanied by decreasing model effectiveness.

Keywords: Knowledge Distillation, BERT, Model Compression.

1. Introduction

With the proposal of the pre-training model Bidirectional Encoder Representations from Transformers (BERT), its effectiveness on NLP has been constantly refreshed [1]. However, the huge overhead of space and time associated with its large number of parameters limits its application to downstream tasks. Based on this, there is a desire to find a smaller BERT model that takes into account the capabilities of BERT and can have a smaller size. Combining the previously proposed idea of distillation, knowledge distillation and BERT are combined to obtain a desired model.

Knowledge distillation as a model compression technique consists of two models. The former is named teacher model. It has larger volume and is more capable. The latter one is called student model, which has small number of parameters. During training, the student model is allowed to learn to mimic the teacher's behaviour to learn the pre-trained teacher model's capability. The loss in the distillation process consists of the cross-entropy of the predicted probability of the two models and the cross-entropy of the student model and the true value [2,3]. BERT distillation that is, the parameters and volume of the BERT shrunk by the distillation method under the condition of ensuring that the model ability does not lose too much.

DOI: 10.54254/2755-2721/51/20241188

2. Preliminary Knowledge

2.1. Knowledge Distillation

The model output is the category probabilities processed by the softmax layer. In this approach, all negative labels are treated uniformly, and similarity information between labels is mostly ignored, which leads to a reduction in the amount of knowledge output from the teacher model [4]. Figure 1 demonstrates the entire training process. To take advantage of this similarity knowledge, T, a temperature variable, is designed to soften the output classification information of the traditional softmax layer:

$$q_i = \frac{\exp\left(\frac{z_i}{T}\right)}{\sum_j \exp\left(\frac{z_i}{T}\right)} \tag{1}$$

The higher the temperature, the smoother the obtained labels are and the more information they carry. Optimising the loss function in this way allows the teacher model to pass more knowledge to the student model.

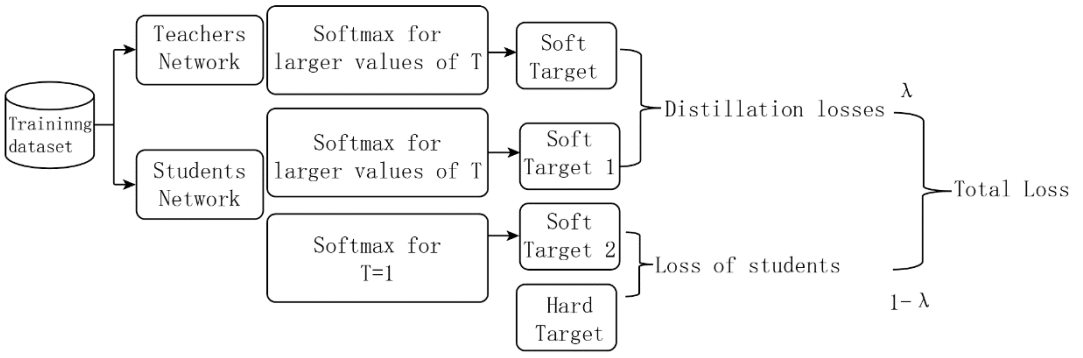

Figure 1. Training framework of knowledge distillation (Figure Credits: Original).

However, since the teacher model also has a certain error rate, the loss function also needs to incorporate the loss between the predicted and true values of the student model to reduce the likelihood of errors being passed on to the students. Therefore, the loss function for the whole of knowledge distillation is weighted by these two components. That is

$$\mathrm{L} = \alpha \mathrm{L}_{\mathrm{soft}} + (1-\alpha)\mathrm{L}_{\mathrm{hard}}, \quad \alpha \in (0,1) \tag{2}$$

2.2. BERT Model

BERT is a pre-trained language model. It is widely leveraged in several natural language processing tasks. It was first proposed by Google in 2018, and was proposed by Jacob Devlin et al. The emergence of BERT breaks records in several tasks, such as question and answer, machine reading comprehension, and natural language reasoning [5,6]. Traditional natural language processing models (e.g., recurrent neural networks and convolutional neural networks) can usually only be processed in a one-way order, resulting in limited performance when dealing with some complex semantic tasks. BERT, on the other hand, employs the Transformer model, which allows the model to process in both directions at the same time, thus better capturing the contextual and semantic relationships in a sentence.

The design of BERT is based on two key ideas: pre-training and fine-tuning. First, BERT is pretrained on large-scale unlabelled text corpus. This pre-training phase is called "Masked Language Model" (MLM) and "Next Sentence Prediction" (NSP) tasks. In the MLM task, BERT is designed to

DOI: 10.54254/2755-2721/51/20241188

randomly mask a number of words in the input text and then predict these words according to the context; in the NSP task, BERT predicts whether two sentences are consecutive. Through pre-training, BERT learns a language model for huge texts, and with this generalised language understanding, it can be applied to a variety of natural language processing tasks. Then, fine-tuning on specific tasks. The BERT model is connected before a new task-specific neural network architecture and trained end-to-end with labelled data. Through fine-tuning, BERT can be adapted to the specific requirements of the task to achieve better performance.

3. BERT Improved by Knowledge Distillation

3.1. DualTrain+SharedProj

Pre-trained models achieve top-notch performance on NLP tasks, but are very constrained for many application scenarios and on mobile devices due to their large number of covariates and body size, which requires a large amount of memory space. Whereas knowledge distillation has achieved success in large neural network model compression, they cannot efficiently generate student models with a vocabulary different from that of the teacher's original model. Therefore, a novel distillation approach is proposed to distil the capabilities of the teacher model into a student model with a much smaller vocabulary.

Distinguish from other distillation methods, DualTrain+SharedProj has two special features [7]. One is Dual Training, the other is Shared Projection. Its architecture is demonstrated in Figure 2. Dual Training is mainly to solve the problem that the two models do not share the same word list. In the distillation process, the word list of the teacher model or the student model will be randomly selected for word segmentation. It can be understood that the word lists of the two models are mixed, and in this way, two word lists of different sizes can be aligned. For example, in the left part of the figure, "I" and "machine" use the participle result of the teacher model while the rest of the tokens use the participle result of the student model.

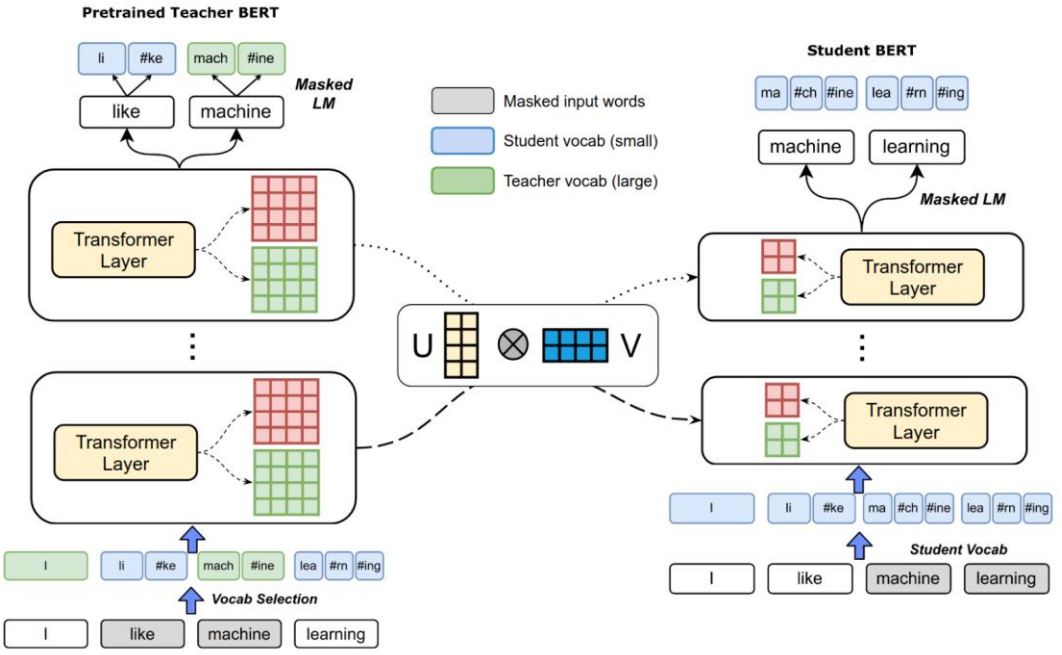

Figure 2. Architecture of DualTrain+SharedProj [7].

DOI: 10.54254/2755-2721/51/20241188

The second part is the Shared Projection. This part is very easy to understand, because the embedded layer latitude of the student model is reduced, resulting in the latitude of each transformer layer is reduced. However, it is hoped that the trainable parameters of the corresponding transformer layers are close enough to each other, so here it requires a trainable matrix to scale the parameters of the two different dimensions of the transformer layer to the same dimension in order to make a comparison. If the scaling is done on the parameters of the teacher model, it is called down projection, and if the scaling is done on the parameters of the student model, it is called up projection.Meanwhile, the parameters of the 12 layers of transformer share the same scaling matrix, so it is called shared projection.

The aim of distillation is to make the student model to maintain similar performance with the teacher model, but with less computational resources and number of parameters. DualTrain+SharedProj achieves this goal through two training phases.

As for the distillation process, the first training phase is DualTraining, in which the teacher model is leveraged to generate pseudo-labels to augment the training data. Specifically, unlabelled data is used for prediction, and then the teacher model's output is used as pseudo-labels, and these predicted samples are used along with the original labelled samples to train the student model. This not only increases the training data, but also improves the generalisation of the student model.

The second training phase is Shared Projection, in which the hidden representation of the student model shares the same projection space as the hidden representation of the teacher model. By minimising the distance between the two models in the Shared Projection space, the performance of the student model can be further improved and its output can be ensured to be consistent with that of the teacher model.

By combining the two training phases, DualTrain+SharedProj is able to efficiently distil the BERT model and transfer its knowledge into a simplified student model. This approach is able to maintain high performance while reducing model complexity and computational resource requirements, and is applicable to a variety of natural language understanding tasks.

3.2. DistillBert

DistilBERT is a lightweight BERT model developed by Hugging Face. It aims to reduce the model size and computational burden, meanwhile maintaining high performance [8].

DistilBERT has significantly reduced the number of parameters compared to BERT. Compared with the original design with 110 million parameters the DistilBERT model has only 66 million parameters. distilBERT follows the structure of Bert, but the number of transfromer layers is only 6 (compared to 12 in Bert Base), and the embedding layer token-type embedding and the last pooling layer are deleted. In order to make DistilBERT have a more reasonable initialisation, the transformer parameters of DistilBERT are derived from Bert Base, and the parameters of one of the two layers are taken from every two layers of the transformer to be used as the initialisation parameters of DistilBERT.

During training, DistilBERT uses two different objective functions. Firstly, it trains the teacher model by means of a common language model training objective, i.e. modelling by predicting the next word. Then, the student model takes the teacher model's output as a secondary objective to minimise the differences between the two models. To achieve this goal, DistilBERT uses a mean square error (MSE) loss function as the objective function for student model training. The student model minimises the loss between the predictions with the teacher model to gradually approach optimal performance. In the distillation process, a self-supervised training loss (the loss of the MLM task) was added in addition to the regular distillation part of the loss. In addition, the experiments in Hugging face also found that adding a word embedding loss is beneficial to align the hidden layer representations of the two models.

Hugging face's experimental results show that DistilBERT has a smaller model size compared to the original BERT model. It achieves model streamlining by reducing model size and the dimensionality of the hidden layers. This makes DistilBERT easier to use and deploy in resource-constrained environments. DistilBERT has faster inference speed compared to the original BERT model. Because of the smaller model size, DistilBERT can predict and reason faster on the same hardware. This is useful for application scenarios that require real-time or large-scale inference. Since DistilBERT has fewer

DOI: 10.54254/2755-2721/51/20241188

parameters and faster inference, it can lower the cost of training and inference. This means that users can train and use DistilBERT models more efficiently with less computational resources.

3.3. BERT with LSTM

Distillation learning does not require that the teacher and the student model shares the same architecture. Therefore, some people have the idea of using BiLSTM as the student model to carry the huge capacity of Bert Base [9].

The teacher model remains Bert Base here, and the student model is split into three parts. The first is a word embedding layer. The second part is bi-directional LSTM+pooling, here the hidden layer state obtained from BiLSTM will be used to generate the representation of the sentence through max pooling. The third part is the fully connected layer, which outputs categorical probabilities. The distillation process consists of two stages. First, large-scale pre-training is performed on the BERT model to obtain its strong linguistic representation. Then, distillation training is performed using unlabelled transfer data by minimising the difference between the simplified model and the BERT output. Distillation loss and consists of three components: hard target loss, soft target loss, and unlabelled transfer data loss. The first loss measures the difference between the student model and the real labels, and the soft target loss measures the difference between the student model and the teacher model. The third part is the KL distance between generated representations of the two models, which is the distance between BiLSTM+pooling and the last layer of state output of Bert base. But since these two may not have the same dimensions, a fully connected layer needs to be introduced here for scaling as well.

The training has two main phases: the pre-training and the distillation phase.

In the pre-training phase, a massive corpus of unlabelled text is used to pre-train BERT. The purpose of this phase is to enable BERT to learn common language representations and models. The pre-training tasks include MLM and NSP. In the first one, certain words in the input text are randomly masked and the correct labels of these masked words are predicted from the context. In this way, BERT learns contextually relevant word representations. In the NSP task, BERT receives two consecutive sentences as input and predicts whether these two sentences are truly consecutive or are randomly sampled in the original text. This task helps BERT to learn the correlation information between sentences.

After completing the pre-training, a distillation phase is carried out in which the knowledge is transferred to TinyBERT. This process involves the use of unlabelled transfer data for distillation. Firstly, a small number of layers (n Transformer layers) from the BERT model are selected as the teacher model. Then, this teacher model is employed to generate pseudo labels for the transferred data. The transfer data is a set of unlabelled data similar to the target task corpus. Inference is performed on this transfer data using the teacher model to obtain the corresponding output probabilities and attention distributions. These output probabilities and attention distributions become the targets of TinyBERT. Next, TinyBERT is trained using pseudo-labels with transfer data.During training, TinyBERT is gradually approximated to the teacher model by minimising the difference between the output probabilities and attention distributions of TinyBERT and the teacher model's output. By distilling the unlabelled transfer data with the pseudo-labels of the teacher model, TinyBERT is able to capture similar language representation capabilities as BERT at a smaller model size.

This distillation process can be carried out through repeated iterations to further improve the performance of TinyBERT. A validation set can be used to monitor the TinyBERT's performance on the target task and select the best model for inference and evaluation. The BiLSTM obtained by distillation is significantly better than that of direct finetune, and the effectiveness of distillation learning is demonstrated here.

3.4. Patient Knowledge Distillation (PKD)

PKD is based on the idea of knowledge distillation, which is improved by introducing a "patient" concept and iterative training [10].

PKD wants to compress the number of layers of Bert Base's transformer through distillation learning. But conventional approaches only learn the results from the last layer of the teacher model. While being

DOI: 10.54254/2755-2721/51/20241188

able to achieve results comparable to the teacher model during training, its testing performance quickly converges. This phenomenon looks like overfitting on the training set, thus affecting the generalisation ability of the student model. Afterwards, PKD adds new constraint terms to the original to drive the student model to learn to mimic the intermediate process of the teacher model. There are two possible ways to do this specifically. The first is to learn the student model from the results of the every few layers in the teacher model. The second is to have the student model learn the results of the last few layers of the teacher model transformer.

In PKD, the concept of patient denotes the gradual reduction of the gap between the two models. This process is achieved by gradually decreasing the temperature parameter, which controls the degree of smoothing of the teacher model's output. Initially, a higher temperature makes it easier for the target model to distil the knowledge of the teacher model. And the temperature is gradually lowered as training progresses so that the target model better captures detail and complexity.

PKD employs an iterative training strategy to gradually improve the performance of the target network through multiple iterations. In each iteration, the predictions of the teacher network are first obtained by forward propagating the training data using the teacher network. These predictions are then used to guide the training of the target model to approximate the teacher network's output. In each iteration, different temperature parameters and learning rate strategies are used to improve the robustness and efficiency of training.

According to Microsoft Dynamics 365 AI Research, PKD-skip scored 92 on the SST-2 (Stanford Sentiment Treebank) dataset, and reached 80.6 and 88.9 on the MPRC and QQP datasets, respectively.For Natural Language Processing, it reached 81 and 89 on MNL and QNLI, respectively. Compared to the other models, PKD-skip outperforms the baseline method on almost all datasets except Microsoft Research Paraphrase Corpus (MRPC).

The PKD approach has the advantage of being able to maintain high performance while the model size is reduced. By slowly decreasing the temperature and iterative training, PKD can efficiently convey complex knowledge in BERT models and make full use of the rich information in large teacher models.

4. Result

In order to visually compare the compression efficiency and modelling effectiveness of the distillation methods mentioned above. In this paper, specific information on several models and their performance on the MRPC dataset is summarized in Table 1. MRPC (Microsoft Research Paraphrase Corpus) is a dataset commonly used for text similarity and text matching tasks. The dataset contains a series of sentence pairs, each of which is labelled as "similar" or "dissimilar". On the MRPC dataset, the above models are used for text similarity measurement and text matching, and compared with the BERT BASE model.

Table 1. Performance of various models.

Model	Hidden Dim	Compress factor	MRPC
BERT BASE		1	88.9
DualTrain+SharedProjUp	192	5.74	84.9
	96	19.41	84.9
	48	61.94	79.3
DistillBERT		1.67	87.5
PKD	6	1.64	85.0
	3	2.40	80.7

It has been found that higher compression efficiency is often accompanied by a steady decline in model effectiveness. the upper limit of the student model is the teacher model. For the same student model, a large teacher model may not help. This is because a larger teacher model means greater compression efficiency, which also means more severe performance degradation. We also found that phased distillation is effective that is, learning the generic teacher model first, and then learning the

DOI: 10.54254/2755-2721/51/20241188

finetune teacher model for a specific task. Distillation across model structures is also effective, with the ability to learn Bert Base with BiLSTM outperforming direct finetune BiLSTM.

5. Conclusion

This paper mainly introduces the BERT model and knowledge distillation. This paper selects several common distillation models, analyses their model architectures as well as distillation effects, and finally summarizes the comparison of the effects of different models. Upon comparison, it is found that the higher compression efficiency is often accompanied by a decrease in model effect. Distillation across model structures is effective, with the ability to learn Bert Base with BiLSTM outperforming direct finetune BiLSTM. More applications of BERT distillation still need deeper research in the future.

References

[1] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

[2] Gou, J., Yu, B., Maybank, S. J., & Tao, D. (2021). Knowledge distillation: A survey. International Journal of Computer Vision, 129, 1789-1819.

[3] Cho, J. H., & Hariharan, B. (2019). On the efficacy of knowledge distillation. In Proceedings of the IEEE/CVF international conference on computer vision, 4794-4802.

[4] Hinton, G., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531.

[5] Acheampong, F. A., Nunoo-Mensah, H., & Chen, W. (2021). Transformer models for text-based emotion detection: a review of BERT-based approaches. Artificial Intelligence Review, 1-41.

[6] Huan, L., Zhixiong, Z., & Yufei, W. (2021). A review on main optimization methods of BERT. Data Analysis and Knowledge Discovery, 5(1), 3-15.

[7] Zhao, S., Gupta, R., Song, Y., & Zhou, D. (2020). Extreme language model compression with optimal subwords and shared projections ICLR, 1-11.

[8] Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. arXiv preprint arXiv:1910.01108.

[9] Mukherjee, S., & Awadallah, A. H. (2019). Distilling bert into simple neural networks with unlabeled transfer data. arXiv preprint arXiv:1910.01769.

[10] Sun, S., Cheng, Y., Gan, Z., & Liu, J. (2019). Patient knowledge distillation for bert model compression. arXiv preprint arXiv:1908.09355.