Deep-to-bottom Weights Decay: A Systemic Knowledge Review Learning Technique for Transformer Layers in Knowledge Distillation

Anonymous ACL submission

Abstract

There are millions of parameters and huge computational power consumption behind the outstanding performance of pre-trained language models in natural language processing tasks. Knowledge distillation is considered as a compression strategy to address this problem. However, previous works (i) distill partial transformer layers of the teacher model, which ignore the importance of bottom base information, or (ii) neglect the difficulty differences of knowledge from deep to shallow, which corresponds to different level information of teacher model. We introduce a deep-to-bottom weights decay review mechanism to knowledge distillation, which fuses teacher-side information taking each layer's difficulty level into consideration. To validate our claims, we distill a 12-layer BERT into a 6-layer model and evaluate it on the GLUE dataset. Experimental results show that our review approach is able to outperform other existing techniques.

1 Introduction

In recent years, pre-trained language models, such as BERT (Devlin et al., 2019), GPT-3 (Brown et al., 2020), Switch Transformer (Fedus et al., 2021) have achieved great success in many NLP tasks. However, pre-trained language models suffer from expensive overhead on computation and memory for inference due to the large number of parameters. This makes it impractical to deploy such models on resource-constrained devices. Therefore, it is important to obtain a lightweight pre-trained language model using the compression method while maintaining performance. A number of approaches have been proposed to solve this problem, including parameter sharing (Lan et al., 2020), pruning (Michel et al., 2019), quantization (Zafrir et al., 2019; Shen et al., 2020), dynamic early exit (Xin et al., 2020; Liu et al., 2020) and knowledge distillation (Sanh et al., 2019; Jiao et al., 2020). However, parameter sharing reduces storage overhead but does not reduce computation, unstructured pruning and quantization can only work together with specific hardware devices or libraries, structured pruning may completely prune the key components of the model, resulting in a large decline in accuracy, dynamic early exit reduces the amount of computation but cannot reduce the storage overhead.

Knowledge distillation transfers knowledge from "large" teacher model to "small" student model, which can reduce the computation and storage overhead of the model at the same time without special devices (Hinton et al., 2015). Therefore, Knowledge distillation is considered as a practical way for compression. So far, several studies have used knowledge distillation to compress the pre-trained language model, such as DistillBERT (Sanh et al., 2019), BERT-PKD (Sun et al., 2019), TinyBERT (Jiao et al., 2020), BERT-EMD (Li et al., 2020), ALP-KD (Passban et al., 2021). However, (Jawahar et al., 2019) observes an interesting fact that the knowledge learned by each transformer layer of BERT from bottom layer to high layer shows a state from shallow to deep, from easy to difficult, from phrase level information, syntactic level information, to semantic level information. Thus, the above works have the following problems from this perspective: (i). just select a subset of the intermediate layers, some important base information such as phrase level information, syntactic level information is omitted in the remaining parts. (ii). neglect the difficulty of knowledge in different intermediate layers. The student model does not carry out step-by-step learning, it may directly learn the difficult knowledge in the teacher model such as BERT-EMD and ALP-KD. This is not in line with the law of learning.

In this paper, we propose DWD-KD (**D**eep-to-bottom **W**eights **D**ecay: A Systemic Knowledge Review Learning Technique for Transformer Layers in **K**nowledge **D**istillation) to solve these prob-

lems from a new perspective, which takes each layer's difficulty level into consideration. The inspiration comes from a well-known phenomenon that human is often taught to review the old knowledge to understand the new knowledge better. (Chen et al., 2021) has applied knowledge review in computer vision tasks for CNN networks and achieved competitive results. Motivated by these works, DWD-KD utilizes multi-task learning to learn new knowledge and review old knowledge at the same time. Moreover, it allows the model systematically understand knowledge through decay weights for deep-to-bottom transformer layers. Our experiments on BERT with GLUE (Wang et al., 2019) show that DWD-KD outperforms other existing methods, and validate that deep-to-bottom weights decay is an effective review learning technique as compared to others.

2 Related Work

We mainly review the related works that use knowledge distillation to compress the BERT model with intermediate layers. BERT-PKD (Sun et al., 2019) selects a portion of the middle layer of the teacher model for distillation. BERT-EMD (Li et al., 2020) borrows the Earth Mover's Distance algorithm of Operations Research for many-to-many layer alignment while distilling the BERT model. ALP-KD (Passban et al., 2021) gives different weights to the intermediate layer of the teacher model based on the layer-to-layer similarity between the teacher model and student model.

The main difference between our work and previous methods is that they do not discuss the possibility to review knowledge in BERT compression, which is found in our work.

3 Methodology

Suppose that teacher and student models respectively have M and N transformer layers, M is larger than N. We overview DWD-KD method as shown in Fig. 1. It consists of four components: word embedding distillation (3.1), transformer layer distillation with review mechanism (3.2), soft label loss and hard label loss (3.3).

3.1 Word Embedding Distillation

A good word embedding vector will help to extract text features better (Jiao et al., 2020). Thus during the knowledge distillation process, we minimize

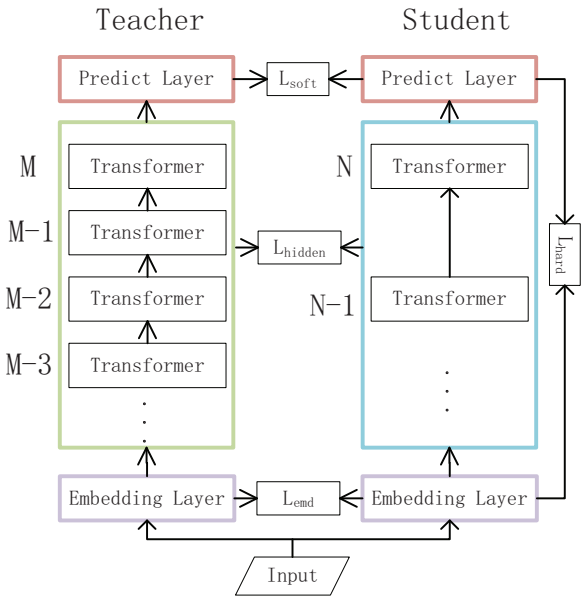

Figure 1: An overview of the DWD-KD method. Embedding layer and prediction layer of teacher and student model are one-to-one alignment, while the intermediate transformer layers are many-to-one alignment by review learning from deep to bottom as Eq. 2 describes.

the mean square error (MSE) between the embedding vector of the teacher model and student model as Eq. 1:

$$L_{emd} = MSE(E^T, E^S) \tag{1}$$

where the matrices E^S and E^T respectively represent the embeddings of student model and teacher model, which have the same shape.

3.2 Transformer layer Distillation with Review Mechanism

From the bottom layer to the high layer, the knowledge goes from easy to difficult (Jawahar et al., 2019), DWD-KD gives corresponding weight goes from small to large. The layer alignment method of DWD-KD is described in Eq. 2, which means n_{th} transform layer of the student model corresponds to first $\lfloor \frac{n*M}{N} \rfloor$ transform layers of the teacher model. Then we combine multiple layer 's hidden states as Eq. 3 describes. We utilize Eq. 4 to normalize the layer number as weights.

$$A(n) = \{1, \cdots, \lfloor \frac{n*M}{N} \rfloor\} \tag{2}$$

$\forall n \in \{1, \cdots, N\}$, however, we discover that there is no obvious disparity among weights calculated by Eq. 4, which means there is no emphasis between the old knowledge and the new knowledge.

The new knowledge represents deep knowledge, which should be much larger than the weight of old knowledge.

$$F_{\lfloor \frac{n*M}{N} \rfloor} = \sum_{m=1}^{\lfloor \frac{n*M}{N} \rfloor} w_m \cdot h_m^t \tag{3}$$

$$w_m = m / \sum_{i \in A(n)} i \tag{4}$$

$$w_m = exp(m) / \sum_{i \in A(n)} exp(i) \tag{5}$$

So we also use the softmax function to calculate the weight of each layer as Eq. 5. Then, we take mean square error to calculate loss between $F_{\lfloor \frac{n*M}{N} \rfloor}$ and h_n^t as Eq. 6 and Eq. 7 describe.

$$L_{hidden}^n = MSE(F_{\lfloor \frac{n*M}{N} \rfloor}, h_n^s), \tag{6}$$

$$L_{hidden} = \sum_{n=1}^{N} L_{hidden}^n \tag{7}$$

m, n denotes layer num, h_m^t stands for m_{th} hidden states of teacher model, h_n^s means n_{th} hidden states of student model. w_m represents m_{th} transformer layer 's weight of teacher model. $F_{\lfloor \frac{n*M}{N} \rfloor}$ denotes integrated result of first $\lfloor \frac{n*M}{N} \rfloor$ transformer layer 's hidden states.

3.3 Prediction Distillation

Prediction distillation include soft label loss and hard label loss. Soft label is the probability logits of teacher model. Hard label is one-hot label vector of the sample. Like (Hinton et al., 2015), we utilize KL divergence to calculate soft label loss, and use the Cross Entropy to calculate hard label loss:

$$L_{soft} = KL(Z^T/t, Z^S/t) \tag{8}$$

where Z^T and Z^S respectively represent the probability logits predicted by the teacher and student, t denotes temperature value. V is a one-hot label vector of the sample.

$$L_{hard} = CE(Z^S, V) \tag{9}$$

3.4 Total Loss

Finally, we combine word embedding distillation loss, transformer layer distillation loss, soft label loss, and the hard label loss to form the total loss function as Eq. 10:

$$\begin{aligned} L = &\alpha \cdot L_{emd} + \beta \cdot L_{soft} \\ &+ (1-\beta) \cdot L_{hard} + \gamma \cdot L_{hidden} \end{aligned} \tag{10}$$

α, β and γ are hyper-parameters.

4 Experiments

4.1 Experimental Setup

We evaluate our DWD-KD on the General Language Understanding Evaluation (GLUE) (Wang et al., 2019) benchmark. The teacher model is a 12-layer BERT model (BERT-base-uncased), which is fine-tuned for each task to perform knowledge distillation. We initialize the student model with the parameters of the first six layers of the teacher model. We implement DWD-KD using the TextBrewer (Yang et al., 2020), which is an open-source knowledge distillation library.

4.2 Baseline Methods

We not only compare our DWD-KD with fine-tuned 6-layer BERT models (BERT-FT) but also with several BERT compression approaches, including BERT-PKD (Sun et al., 2019), BERT-EMD,(Li et al., 2020), ALP-KD (Passban et al., 2021), all of them focus on how to distill knowledge from the intermediate layer of the teacher model, We also reproduce the experimental results of the above model through the TextBrewer (Yang et al., 2020) library. DWD-KD1 and DWD-KD2 respectively mean utilizing the Eq. 4 and Eq. 5 to calculate weights.

4.3 Main Results

We summarize the experimental results on the GLUE val sets in Tab. 1. Following previous works (Li et al., 2020; Passban et al., 2021), we also report the average scores. (1). DWD-KD achieves the best average results among all the 6-layer models. DWD-KD achieves a better result than BERT-FT on all the datasets with an absolute improvement of 3% on average score. (2). DWD-KD performs better than teacher on 5 out of 8 tasks. For example, DWD-KD achieves a noticeable improvement of 3.61% accuracy on RTE and 0.94% on MRPC. (3). We observe that all models do not perform well on the CoLA dataset. The CoLA dataset is a corpus of language acceptability, which means the model needs to judge whether a sentence is grammatically correct. The difficulty of grammatical

Model	MNLI(m/mm)	QQP	QNLI	SST-2	CoLA	STS-B	MRPC	RTE	Average
Teacher	83.82/84.35	90.69	91.32	92.32	59.07	88.13	89.35	66.43	82.83
BERT-FT	80.37/80.64	89.66	86.87	89.68	40.39	87.78	87.62	63.90	78.55
BERT-PKD	83.13/83.03	90.80	89.46	90.83	38.56	87.90	88.89	67.51	80.01
ALP-KD	83.68/83.51	**91.39**	89.36	90.94	46.10	88.82	89.78	66.43	81.11
BERT-EMD	83.44/83.36	91.31	89.09	90.71	44.61	88.82	88.89	66.79	80.78
DWD-KD1	83.53/83.96	**91.39**	90.12	91.28	44.12	88.58	89.78	**70.04**	81.42
DWD-KD2	**84.44/84.39**	91.30	**90.76**	**91.40**	43.27	88.46	**90.29**	69.68	**81.55**
equal weights	82.74/83.02	91.30	88.49	90.71	44.77	**88.77**	88.74	67.14	80.63
growth weights	80.24/79.57	90.66	85.92	88.76	27.19	86.35	85.05	60.65	76.04
random weights	83.08/83.27	91.23	89.40	**91.40**	**46.19**	88.42	87.66	67.87	80.95

Table 1: We evaluate the model on GLUE val sets. the Teacher is a 12-layer BERT model, all other models are 6-layer models and have the same architecture as the teacher. BERT-FT stands for fine-tuning the first 6 layers of the teacher. The data of the last three lines are the results of the strategy comparison experiment. CoLA scores are Matthews Correlation Coefficient. SST-B scores are average value of Pearson correlation coefficient and Spearman correlation coefficient. MPRC scores are F1-Score and the rest are accuracy scores.

errors between negative samples in CoLA dataset is quite different. The grammatical errors in some negative examples are missing a word or having an extra word in a sentence. Such errors are relatively simple, but other grammatical errors can only be correctly identified with a deeper understanding of linguistic knowledge such as voice and tense. (Jiao et al., 2020) shows pre-trained language models can obtain more linguistic knowledge from more corpus for better results on the CoLA dataset.

4.4 Strategy Comparison

To prove the effectiveness of the weights decay review mechanism, we also use three other strategies for distillation experiments. The last three lines in Tab. 1 shows the experiment results. "equal weights" denotes the weight of each layer is the same. The weight value is equal to $\lfloor \frac{N}{n*M} \rfloor$. "growth weights" denotes we reverse the original weights. "random weights" denotes we randomly shuffle the original weights. Original weights are the same as DWD-KD2. As we can observe from Tab. 1, student model performs poorly in almost all data sets when we reverse the original weights. The average score is even 2.51% lower than BERT-FT. This is because the old knowledge of the teacher model contains relatively elementary linguistic knowledge, it is insufficient. The average score of random weights is slightly higher than that of equal weights. But both are smaller than that of DWD-KD with decreasing weight.

4.5 Visualization of Review mechanism

To better show the effectiveness of the review mechanism, we select 100 samples from RTE dataset and visualize the Manhattan Distance between the third transformer layer 's output of ALP-KD, DWD-KD and sixth transformer layer's output of teacher model in Fig. 2. The original output is a 768-dimensional vector, we use PCA to select the first two principal components of it. As we can observe from Fig. 2, the distance between the output of DWD-KD and that of the teacher model is closer, which proves that our method can get a better student model.

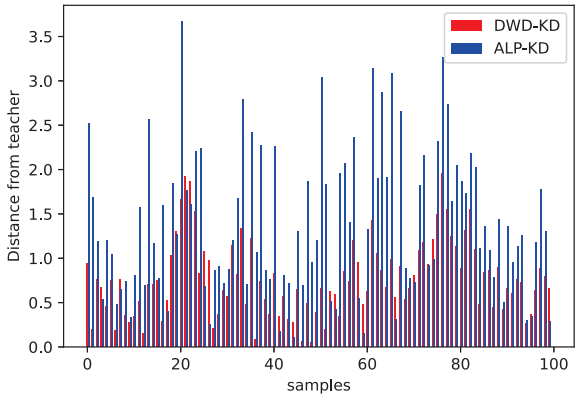

Figure 2

5 Conclusions

In this paper, we have studied knowledge distillation from a new perspective and accordingly proposed the deep-to-bottom weights decay review mechanism applying in BERT compression, which enables the student model systematically learn the basic knowledge during distillation process. Our method achieves competitive results on 5 out of 8 tasks as compared to the original model.

References

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Pengguang Chen, Shu Liu, Hengshuang Zhao, and Jiaya Jia. 2021. Distilling knowledge via knowledge review. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 5008–5017. Computer Vision Foundation / IEEE.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

William Fedus, Barret Zoph, and Noam Shazeer. 2021. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. abs/2101.03961.

Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the knowledge in a neural network.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 3651–3657. Association for Computational Linguistics.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2020. Tinybert: Distilling BERT for natural language understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 4163–4174. Association for Computational Linguistics.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. ALBERT: A lite BERT for self-supervised learning of language representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*.

Jianquan Li, Xiaokang Liu, Honghong Zhao, Ruifeng Xu, Min Yang, and Yaohong Jin. 2020. BERT-EMD: many-to-many layer mapping for BERT compression with earth mover's distance. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 3009–3018. Association for Computational Linguistics.

Weijie Liu, Peng Zhou, Zhiruo Wang, Zhe Zhao, Haotang Deng, and Qi Ju. 2020. Fastbert: a self-distilling BERT with adaptive inference time. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 6035–6044. Association for Computational Linguistics.

Paul Michel, Omer Levy, and Graham Neubig. 2019. Are sixteen heads really better than one? In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 14014–14024.

Peyman Passban, Yimeng Wu, Mehdi Rezagholizadeh, and Qun Liu. 2021. ALP-KD: attention-based layer projection for knowledge distillation. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 13657–13665. AAAI Press.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. abs/1910.01108.

Sheng Shen, Zhen Dong, Jiayu Ye, Linjian Ma, Zhewei Yao, Amir Gholami, Michael W. Mahoney, and Kurt Keutzer. 2020. Q-BERT: hessian based ultra low precision quantization of BERT. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8815–8821.

Siqi Sun, Yu Cheng, Zhe Gan, and Jingjing Liu. 2019. Patient knowledge distillation for BERT model compression. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 4322–4331. Association for Computational Linguistics.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019.

GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*.

Ji Xin, Raphael Tang, Jaejun Lee, Yaoliang Yu, and Jimmy Lin. 2020. Deebert: Dynamic early exiting for accelerating BERT inference. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 2246–2251. Association for Computational Linguistics.

Ziqing Yang, Yiming Cui, Zhipeng Chen, Wanxiang Che, Ting Liu, Shijin Wang, and Guoping Hu. 2020. Textbrewer: An open-source knowledge distillation toolkit for natural language processing. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, ACL 2020,Online, July 5-10,2020*, pages 9–16. Association for Computational Linguistics.

Ofir Zafrir, Guy Boudoukh, Peter Izsak, and Moshe Wasserblat. 2019. Q8BERT: quantized 8bit BERT. In *Fifth Workshop on Energy Efficient Machine Learning and Cognitive Computing - NeurIPS Edition, EMC2@NeurIPS 2019, Vancouver, Canada, December 13, 2019*, pages 36–39. IEEE.