SCIENCE CHINA
Information Sciences

• **LETTER** •

A Unified Pruning Framework for Vision Transformers

Hao YU[1] & Jianxin WU[1*]

[1] *State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing* 210023*, China*

Citation

Dear editor,

The transformer architecture [5] has been widely used for natural language processing (NLP) tasks. Under the inspiration of its excellent performance in NLP, transformer-based models [2, 4] have established many new records in various computer vision tasks. However, most vision transformers (ViTs) suffer from large model sizes, large run-time memory consumption, and high computational costs. Therefore, impending needs exist to develop and deploy lightweight and efficient vision transformers.

Network pruning is a useful technique for striking a balance between model accuracy, inference speed, and memory usage. The most time-consuming module in a transformer is the feed-forward network (FFN), but efforts in pruning FFN remain scarce. Recent ViT pruning methods [6, 7] mainly involve recursively sampling informative tokens (or equivalently, their corresponding image patches) to increase the inference speed in image classification, which achieves similar accuracy as that of using all the tokens with less computation. Unfortunately, these token sampling methods impede the vision transformers' generalization ability on downstream tasks. In addition, this type of sampling is also difficult to apply to NLP tasks, which limits the application domain of these methods.

Hence, we believe that successfully accelerating and slimming vision transformers requires a *unified* approach that simultaneously prunes *all* components in a transformer, does *not* alter the transformer structure, generalizes well to downstream tasks with *high* accuracies, applies to not only ViTs but also its many *variants*, and can be easily *extended* to the NLP tasks.

To fulfill these goals, we propose UP-ViTs, a unified pruning framework for vision transformers, which prunes the channels in ViTs in a unified manner, including those inside and outside the residual connections in all the blocks, multihead self-attentions (MHSAs), FFNs, normalization layers, *and* convolution layers in ViT variants. We first devise an efficient evaluation module to estimate the importance score of each filter in a pre-trained ViT model. Then, on the basis of the compression goals, all redundant channels are simultaneously removed, leading to a thinner structure. In particular, when compressing the attention layers, we investigate the influence of MHSA and propose a novel method for discarding channels. We also design an effective progressive block pruning method that removes the least important block and proposes new *hybrid* blocks in ViTs. Experiments on ImageNet show that UP-ViTs considerably outperform previous ViTs with the same or even higher throughput. Our contributions are as follows:

- We propose a novel framework for structured compression of ViTs and their variants. The resulting compressed models achieve higher accuracy than previous state-of-the-art ViTs and existing pruning algorithms.
- Our method maintains the consistency of the token representation. Therefore, we can generalize the compressed model to various downstream tasks.
- Our model can be applied not only for ViTs but also for Transformers in NLP tasks. We demonstrate that our approach improves the state of the art on language modeling benchmarks and results in lower perplexity.

Methods. In this section, we demonstrate how to prune the standard ViT. More details about compressing its variants can be found in the appendix.

First, we calculate the importance scores of all channels. We aim to minimize the information loss of the last layer after pruning channels. In particular, a ViT's block contains two components, an attention layer and an FFN. Each component contains one LN layer. We refer to the FC layer in the attention as FC_q, FC_k, FC_v, and FC_{proj}, plus FC_1 and FC_2 in the FFN. Generally, we divide ViTs into several uncorrelated components and evaluate the performance change after removing each channel in every component.

Let us take one ViT block as an example. We divide the block into three irrelevant structural components as follows:

- Component 1: the shortcut connections that chain representations across *all* blocks, i.e., the input channels of FC_q, FC_k, FC_v, and FC_1, the output channels of FC_{proj} and FC_2, and the two LN layers.
- Component 2: the attention embedding filters inside the attention layer in *every* block, i.e., the input channels of FC_{proj} and the output channels of FC_q, FC_k, and FC_v.
- Component 3: the FFN inter-layer filters in *every* block, i.e., the input channels of FC_2 and the output channels of FC_1.

[*] Corresponding author (email: wujx2001@nju.edu.cn)

To measure the channel importance, we randomly select 2000 images from the training dataset to establish a proxy dataset $\mathcal{D}$. We then extract the output logits on $\mathcal{D}$ and evaluate the performance change before/after removing a specific channel. Inspired by CURL [3], we calculate the score from the KL-divergence between two models with and without this particular channel, i.e.:

$$s = \sum_{i \in \mathcal{D}} D_{\mathrm{KL}}(q_i || p_i), \tag{1}$$

where i enumerates samples from $\mathcal{D}$, p_i is the output of the model without this particular channel, and q_i is the output of the original model. The larger the score s is, the more important this channel is. Note that because MHSA contains a reshaping operation when calculating the importance scores of component 2, we mask the target channel as 0 instead of removing it.

Then, given the original model and every channel's importance score, we first generate the sub-model candidate. In components 1 and 3, on the basis of the preset compression ratio, we *independently* rank the importance scores and delete the less important ones. For simplicity, we use the same compression ratio in all blocks.

In particular, the attention layer in ViT benefits from the multi-head attention mechanism, which captures richer information by using multiple different heads. However, it also brings difficulties to compressing component 2. Therefore, we specifically design a simple but effective method for pruning the multi-heads. In detail, given a D_b-dimensional attention layer with h_b heads, we need to prune it into D_t dimensions with h_t heads. Note that every head contains the same number of dimensions across all attention layers, and h_b is divisible by h_t. Both settings are common in transformer-based models. When pruning multi-head attention, we first merge h_b/h_t heads into one head. Hence, the attention layers will each have h_t heads with D_b dimensions. Then, we remove $\frac{D_b - D_t}{h_t}$ dimensions from every head, and the remaining module is the desired attention layer.

Lastly, we fine-tune the sub-model with all training samples and use the original model as the teacher to distill the sub-model. In contrast to the training strategy of DeiT, we use the classic soft distillation to measure the training loss:

$$\mathcal{L} = \mathcal{L}_{\mathrm{CE}}(y, p) + \alpha \mathcal{L}_{\mathrm{KL}}(q, p)\,, \tag{2}$$

where p is the output probability (after softmax) of the sub-model, q is the output probability of the teacher (i.e., the original model), and y is the true label. $\mathcal{L}_{\mathrm{CE}}$ and $\mathcal{L}_{\mathrm{KL}}$ denote the cross-entropy loss and the KL-divergence.

Experiments. We prune DeiT-B into UP-DeiT-S and test the effectiveness on ImageNet-1k [1]. More experiments are provided in the appendix.

Figure 1 shows the pruning results of DeiT-B. The x-axis represents the throughput, and the y-axis represents the accuracy of the ImageNet-1k validation dataset. Performance comparison between DeiT-S and UP-DeiT-S proves the effectiveness of our framework. Additionally, UP-DeiT-S also *consistently* outperforms previous state-of-the-art ViT variants. Note that we only lost 0.28% accuracy when pruning DeiT-B into UP-DeiT-S (which is 3.92-fold smaller in size and 3.03-fold faster than DeiT-B). This result is better than the previous state-of-the-art pruning ViT methods, such as Evo-ViT [6], which achieves 81.11% accuracy and is 1.53-fold faster.

Conclusion. In this letter, we proposed a novel method called UP-ViTs to prune ViTs in a unified manner. Our framework can prune all components in a ViT and its variants, maintain the models' structure, and generalize well into downstream tasks. UP-ViTs achieve state-of-the-art results when pruning various ViT backbones. Moreover, we studied the transferring ability of the compressed model and found that our UP-ViTs also outperform original ViTs. We also extended our method into NLP tasks and obtained more efficient Transformer models. Please refer to the appendix for more details.

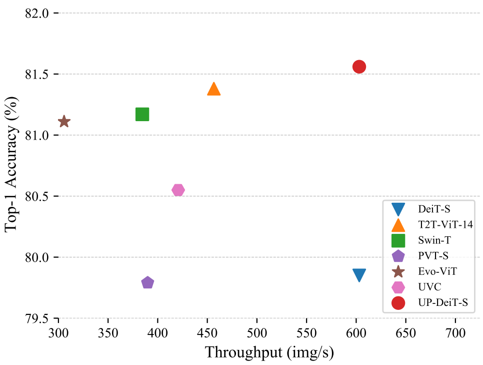

Figure 1 Results of pruning DeiT-B on ImageNet-1k.

Acknowledgements This paper was supported by the National Natural Science Foundation of China under Grant 62276123 and Grant 61921006.

Supporting information Appendix A-E. The supporting information is available online at info.scichina.com and link.springer.com. The supporting materials are published as submitted, without typesetting or editing. The responsibility for scientific accuracy and content remains entirely with the authors.

References

1 Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A large-scale hierarchical image database. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 248–255 (2009)

2 Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (ICLR) (2021)

3 Luo, J.H., Wu, J.: Neural network pruning with residual-connections and limited-data. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1458–1467 (2020)

4 Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jegou, H.: Training data-efficient image transformers & distillation through attention. In: International Conference on Machine Learning. pp. 10347–10357 (2021)

5 Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems. vol. 30, pp. 5998–6008 (2017)

6 Xu, Y., Zhang, Z., Zhang, M., Sheng, K., Li, K., Dong, W., Zhang, L., Xu, C., Sun, X.: Evo-ViT: Slow-Fast token evolution for dynamic vision transformer. arXiv preprint arXiv:2108.01390 (2021)

7 Yu, S., Chen, T., Shen, J., Yuan, H., Tan, J., Yang, S., Liu, J., Wang, Z.: Unified visual transformer compression. In: International Conference on Learning Representations (ICLR) (2021)