Parameter and Computation Efficient Transfer Learning for Vision-Language Pre-trained Models

Submission 1550

A The detailed skipping results

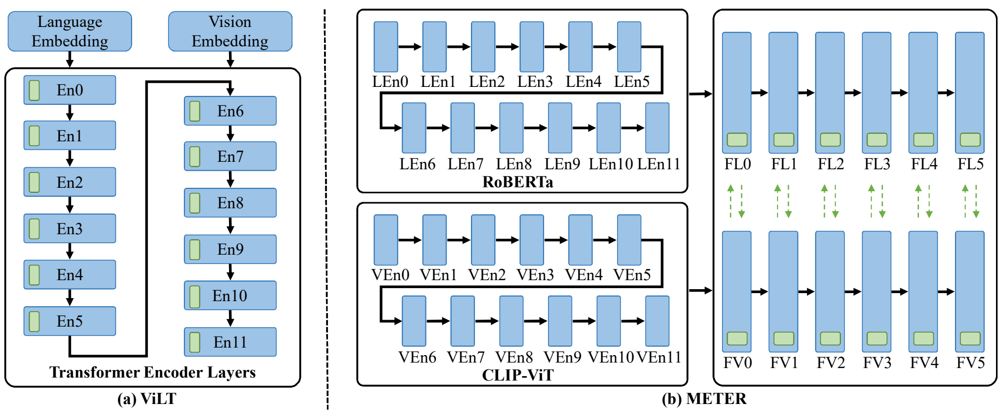

Figure 1: Architectures of the baseline models (a) ViLT and (b) METER. The blue modules are the default Transformer layers that are frozen during the adaptation, while the green ones are the trainable adapters. “En” denotes the encoding layers. “LEn” and “FEn” represent the encoding layers of METER for texts and images, and “FL” and “FV” are the fusion layers for language and vision, respectively.

The architectures of two based models are given in Fig. 1. We also report their detailed skipping results by DAS in Tab. A. Here, “LEn” represents Language Encoder, and “VEn” represents Vision Encoder. We can first see that ViLT is a relatively compact model to METER, which only has 12 Transformer layers without any modality-specific encoder. In this case, it can only be skipped one or two layers without obviously degrading the performance. In stark contrast, METER is a deep and huge VLP model, of which redundancy is much more obvious. By skipping up to 8 layers, its performance drops are still marginal on all tasks. Meanwhile, we also observe that discarding its visual encoder layers will greatly disturb its training and performance during experiments, thus these layers are not considered as the skipping candidates. From Tab. A, we also have some interesting observations. For instance, the language encoding layers are less important to VQA. This may suggest that most questions in VQA2.0 are shorter and less complex, and the model needs to focus more on the visual understanding and cross-modal interactions. This case is less significant on NLVR^2, which requires a detailed comparison between images and texts. Overall, these results confirm that the large VLP models exhibit obvious redundancy to downstream VL tasks. More importantly, the importance of their modules is different to different tasks, requiring proper estimations.

B The results of random sampling

Tab. B gives the detailed results of random sampling mentioned in Fig.3 of the main paper. We can see that random sampling is not only consistently worse than our DAS, but also varies greatly in

Submitted to 37th Conference on Neural Information Processing Systems (NeurIPS 2023). Do not distribute.

Table A: The kipped layers and performance for different base models and tasks. For VQA, we report the test-Dev as the performance. For NLVR^2, we report the test-P as the performance. For Flickr30k, we report IR/TR R@1 as the performance. "Fusion" refers to only skipping the layers in the multimodal fusion modules of METER, while "Global" denotes the skipping scope of the fusion modules and the language encoder.

METER					
Datasets	**Candidates**	**Number of Skipped**	**Per.**	**Additional FLOPs**	**Skipped Layers**
VQA	-	0	75.28	1.68G	-
	Fusion	2	74.92	-9.06G	FV0, FV4
		4	74.80	-11.16G	FL2, FL3, FV0, FV5
		6	74.67	-17.58G	FL1, FL2, FL3, FV0, FV4, FV5
		8	73.70	-24.00G	FL1, FL2, FL3, FL4, FV0, FV1, FV4, FV5
	Global	2	75.24	-3.96G	FV0, LEn6
		4	75.13	-4.51G	FV0, LEn10, LEn11
		6	75.02	-5.06G	FV0, LEn4, LEn6, LEn8, LEn10, LEn11
		8	74.05	-5.61G	FV4, LEn4, LEn5, LEn6, LEn8, LEn9, LEn10, LEn11
NLVR^2	-	0	81.28	0.99G	-
	Fusion	2	80.07	-2.66G	FL4, FV5
		4	80.11	-4.14G	FL2, FL3, FL5, FV5
		6	78.16	-9.97G	FL3, FL4, FL5, FV1, FV3, FV4
		8	79.30	-11.45G	FL1, FL2, FL3, FL4, FL5, FV0, FV3, FV4
	Global	2	81.37	-2.19G	FV5, LEn1
		4	81.34	-3.67G	FL2, FL3, FV5, LEn4
		6	80.04	-4.22G	FL2, FL5, FL6, LEn5, LEn6, LEn11
		8	79.61	-8.34G	FL2, FL3, FL4, FL5, FV1, FV5, LEn4, LEn11
Flickr30k	-	0	81.20/92.40	1.68G	-
	Fusion	4	80.12/91.80	-11.16G	FL4, FL5, FV0, FV3
	Global	4	80.42/91.40	-6.06G	FL2, FL5, FV0, LEn8
ViLT					
Datasets	**Candidates**	**Number of Skipped**	**Per.**	**Additional FLOPs**	**Skipped Layers**
VQA	-	0	70.13	0.73G	-
	Global	1	69.28	-1.03G	En3
		2	67.64	-2.79G	En1, En3
NLVR^2	-	0	76.26	0.73G	-
	Global	1	74.89	-1.03G	En5
		2	73.00	-2.79G	En5, En11
Flickr30k	-	0	62.44/82.10	0.73G	-
	Global	1	60.66/80.80	-1.03G	En7

Table B: The detailed experiment results of random sampled subnetworks for Fig.3 in the main paper.

METER					
Datasets	**Candidates**	**Number of Skipped**	**VQA test-Dev**	**Additional FLOPs**	**Skipped Layers**
VQA	Fusion	4	74.24	-11.16G	FL2, FL4, FV2, FV3
			74.67	-11.16G	FL1, FL5, FV0, FV3
			74.08	-11.16G	FL1, FL4, FV1, FV4
		6	74.05	-17.58G	FL1, FL2, FL3, FV2, FV3, FV5
			73.26	-17.58G	FL0, FL1, FL4, FV0, FV2, FV5
			73.03	-17.58G	FL2, FL4, FL5, FV1, FV2, FV5
		8	71.81	-24.00G	FL0, FL1, FL3, FL5, FV2, FV3, FV4, FV5
			68.56	-24.00G	FL1, FL3, FL4, FL5, FV1, FV2, FV3, FV4
			69.88	-24.00G	FL0, FL2, FL5, FV1, FV2, FV3, FV4, FV5

terms of skipped layers and performance, especially when the number of skipped layers is large. On the contrary, these results just confirm the effectiveness of the proposed DAS.

C Generalization on Pre-trained Language Model

To validate the generalization ability of DAS, we also apply it to a pre-trained language model called RoBERTa [5], as shown in Tab. C. Due to the time limit, we do not conduct careful tunings for RoBERTa. The settings of DAS follow the main paper, while the rest are the same with MAM [1]. From this table, we can first see that DAS is also applicable to pre-trained language models. It can also achieve the target of PCETL in terms of computation and update parameter scales, while obtaining limited performance drops. However, we can also see that the competitiveness of DAS to

Table C: Comparison between DAS and PETL methods for RoBERTa on MNLI and SST2. "En" denotes the encoding layers. "Acc." denotes the accuracy.

Methods	**Updated Parameter**	**Additional FLOPs**	**MNLI**		**SST2**	
			Acc.	**Skipped Layers**	**Acc.**	**Skipped Layers**
Full Tuning	124.65M	0.0	87.6	-	94.6	
Bit-Fit [6]	0.10M	0.0	84.7	-	93.7	
Pre-fix [4]	0.14M	1.20G	86.3	-	94.0	
LoRA [3]	0.59M	0.0	87.2	-	94.2	
Adapter [2]	0.63M	0.33G	87.2	-	94.2	
MAM [1]	0.61M	0.79G	87.4	-	94.2	
DAS_1	0.63M	-3.71G	86.8	En10	94.1	En10
DAS_2	0.63M	-7.74G	86.7	En10, En11	93.9	En8, En10
DAS_3	0.63M	-11.77G	86.2	En9, En10, En11	93.8	En7, En8, En10

other PETL methods is slightly worse on MNLI, of which objective is close to the pre-training ones. We think that the task gap may be a potential factor affecting PCETL. Overall, these results well validate the generalization ability of DAS on LLMs towards PCETL.

References

[1] Junxian He, Chunting Zhou, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. Towards a unified view of parameter-efficient transfer learning. In *International Conference on Learning Representations (ICLR)*, 2022.

[2] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In *International Conference on Machine Learning (ICML)*, pages 2790–2799, 2019.

[3] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.

[4] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 4582–4597, 2021.

[5] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized BERT pretraining approach. *Computing Research Repository (CoRR)*, 2019.

[6] Elad Ben Zaken, Shauli Ravfogel, and Yoav Goldberg. Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. *arXiv preprint arXiv:2106.10199*, 2021.