SparseMAE: Sparse Training Meets Masked Autoencoders

Aojun Zhou1* Yang Li2 Zipeng Qin1 Jianbo Liu1 Junting Pan1 Renrui Zhang13 Rui Zhao2 Peng Gao3 Hongsheng Li1
1The Chinese University of Hong Kong 2SenseTime Research 3Shanghai AI Lab

Abstract

Masked Autoencoders (MAE) and its variants have proven to be effective for pretraining large-scale Vision Transformers (ViTs). However, small-scale models do not benefit from the pretraining mechanisms due to limited capacity. Sparse training is a method of transferring representations from large models to small ones by pruning unimportant parameters. However, naively combining MAE fine-tuning with sparse training make the network task-specific, resulting in the loss of task-agnostic knowledge, which is crucial for model generalization. In this paper, we aim to reduce model complexity from large vision transformers pretrained by MAE with assistant of sparse training. We summarize various sparse training methods to prune large vision transformers during MAE pretraining and finetuning stages, and discuss their shortcomings. To improve learning both task-agnostic and task-specific knowledge, we propose SparseMAE, a novel two-stage sparse training method that includes sparse pretraining and sparse finetuning. In sparse pretraining, we dynamically prune a small-scale sub-network from a ViT-Base. During finetuning, the sparse sub-network adaptively changes its topology connections under the task-agnostic knowledge of the full model. Extensive experimental results demonstrate the effectiveness of our method and its superiority on small-scale vision transformers. Code will be available at https://github.com/aojunzz/SparseMAE.

1. Introduction

Recently, several pretrained Masked Image Modeling (MIM) methods, such as MAE [15] and data2vec [1], have achieved great success in various computer vision tasks. They usually involve a two-stage training scheme where the model learns task-agnostic knowledge through a pretraining pretext task, and is subsequently finetuned on downstream tasks to acquire task-specific knowledge. Among these models, Masked Autoencoders (MAE) demonstrate superior learning capability on large-scale vision transformers (Fig. 1 (a)), thanks to large models’ strong capacity to learn powerful general representations in the pretraining phase and versatile transferability to specific vision tasks. Yet, the large deep models are burdensome and difficult to be deployed to computationally restricted real-world scenarios. On the other hand, smaller and more efficient models, such as ViT-Tiny and ViT-Small, fail to perform well after pretrained by MAE or data2vec frameworks. For example, ViT-Tiny’s finetuning results are even inferior to fully-supervised training (see Fig. 1 (b)). These empirical findings suggest that, under MIM frameworks, when the model is scaled down, it becomes more difficult to learn well via masked pretraining due to its limited capacity during pretraining and consequently hinders its performance when transferred to downstream tasks. In the realm of MIM pretraining, model capacity becomes a key ingredient to learning task-agnostic knowledge via the pretext tasks.

In our studies, we try to tackle the issues of MIM pretraining for small-scale vision transformers. Particularly, we apply pruning techniques [28, 20, 4, 42, 23] in the MIM frameworks in order to obtain performant small-scale models from larger ones. As a first step in our exploration, we use existing sparse training methods [42, 23] for MAE pretraining and finetuning on unstructured and hardware-friendly N:M sparsity to obtain a model with a similar scale to ViT-Tiny, which is pruned from ViT-Base (see Tab. 5 row 3). It achieves notably improvements compared to directly training ViT-Tiny using MAE on downstream ImageNet finetuning accuracy (71.6% vs. 71.6%) and ADE20K [43] semantic segmentation (39.8 mIoU vs. 37.6 mIoU). The empirical results indicate that on small-scale ViTs sparse training via pruning has promising applications under the MIM pretraining paradigm. However, there is still a large performance gap between the sparse model and the vanilla ViT-Base trained densely under MAE in their ImageNet finetuning accuracies (77.6% vs. 83.2%). The sparsity constraints adversely limit the model capacity and prevent it from learning good task-agnostic knowledge, similar to the predicament faced.
by ViT-Tiny when it is directly pretrained under MAE.

To obtain more powerful small-scale models, we propose SparseMAE, a simple and unified sparse training method for Masked Autoencoders. Specifically, SparseMAE trains an adaptive sub-network in conjunction with its corresponding large dense model during pretraining. Both the large model and the sparse sub-network predict the same reconstruction objective and are optimized end-to-end. The large full model provides a strong capacity to learn task-agnostic knowledge, while the sparse sub-network can better preserve the knowledge by rewiring the connections in the full model. To better learn task-specific knowledge in downstream tasks, the sparse sub-network inherits the task-agnostic knowledge from the pretraining stage and also dynamically adjusts its connections during the finetuning phase.

We conduct extensive experiments on a variety of tasks, including image classification, object detection, segmentation, and robustness evaluation, to thoroughly evaluate the effectiveness of our proposed approach. Notably, our method achieves a top-1 accuracy of 80.5% with only 6M parameters on ImageNet-1K [8] classification and obtains a 45.2 mIoU on ADE20K segmentation with UperNet [37], significantly outperforming all previous works that have developed small-scale vision transformers using MAE (e.g., MAE-lite, TinyMIM and G2SD distillation).

In all, our contributions can be summarized in three folds:

- We thoroughly study the integration of sparse training as a pruning technique into the MAE framework.
- Based on the findings from above, we propose a novel sparse training framework, SparseMAE for Masked Autoencoders to improve the acquisition of both the task-agnostic and task-specific knowledge during the pretraining and finetuning stages.
- We present extensive experiments to validate the effectiveness of our proposed method and achieve state-of-the-art performance with small-scale vision transformers on ImageNet classification and various downstream tasks.

2. Related works

2.1. Masked Image Modeling

Unsupervised pretraining on large-scale images with Masked Image Modeling (MIM) has shown superior performance on various computer vision tasks. BEiT [2] explores Masked Image Modeling on vision transformers by reconstructing the vision dictionary [30]. MAE [15] further proposes an asymmetric encoder and decoder for scaling up MIM to huge models. Besides, it demonstrates a simple pixel reconstruction loss can learn good visual representations. Due to the simplicity and computational efficiency, MAE is raising to a popular generative pretraining paradigm [40, 39]. As MAE reconstructs low-level signals with an isotropic vision transformer architecture, researchers improve MAE by exploring high-level signals architectures [1, 35, 12], which are more effective than reconstructing low-level signals.

However, those methods demonstrate the large capacity model is essential for good representation learning through the MIM pretext tasks. MAE-lite [34] explores Masked Autoencoders to improve the performance of tiny-scale vision transformer models, and introduces knowledge distillation into the pretraining phase. Different from MAE-lite, TinyMIM [31] proposes sequential relation knowledge distillation to improve the performance of small-scale vision transformer models, which utilize the full set of patch tokens and distill the knowledge from a pretrained ViT-Large model. TinyMIM [31] needs the large pretrained vision transformer and the use of sequential models is time-consuming for practical applications. These methods attempt to utilize the task-agnostic knowledge from large models through knowledge distillation. Different from their approaches, we explore sparse training as a pruning technique in order to scale down ViTs from large-scale to small-scale under the Masked Autoencoders framework.
2.2. Pruning and Sparse Training

Pruning and sparse training are critical methods for reducing the inference costs of deep neural networks [18]. Post-pruning methods [13, 26] typically involve training a dense full network first, followed by one or several rounds of train-prune-retrain rounds. Post-pruning methods are time-consuming due to the iterative manner [11]. Different from post-training methods, sparse training methods [29, 36] aim to adaptively obtain a sparse sub-network while optimizing both model weights and sparse connections from random initialization.

Sparse training is a class of methods stemming from the Sparse Evolutionary Training (SET) algorithm [29]. It starts from a randomly sparsified network and subsequently prunes and grows connections during training. These methods can be applied end-to-end within the network training stage and have achieved promising results on various vision benchmarks with more efficient models by adaptively changing the model topology connections during training.

Sparse training methods have shown promising results in various computer vision benchmarks, achieving state-of-the-art performance on different types of sparsity granularities such as unstructured sparsity [4, 23, 27], N:M fine-grained structured sparsity [42, 32, 10, 41], and block sparsity [7]. However, to our knowledge, the studies of pruning techniques for Vision Transformers [4, 20] only focus on the fully-supervised settings. There still lacks a unified method to prune large-scale Vision Transformers under the unsupervised Masked Autoencoders framework.

3. Methods

In this section, we first revisit the sparse training technique and Masked Autoencoders (MAE) in Sec. 3.1. Then we emphasize the challenges of integrating existing sparse training methods into MAE in Sec. 3.2. Lastly, we introduce our proposed method to effectively obtain a powerful small-scale sub-network from a large-scale model using sparse training in Sec. 3.3.

3.1. Preliminaries

Revisiting Sparse Training. The pioneering work of Lottery Ticket Hypothesis (LTH) [11] demonstrates the dense
networks contain sparse matching subnetworks capable of training in isolation to full accuracy. Its findings inspire searching for strong sub-networks within a larger full network. Sparse training starts from randomly initializing the network weights $W$ and associating them with a sparse mask $M$, which is a binary tensor with 0 indexing the sparsified entries and 1 indicating the retained ones. $M$ is optimized end-to-end and is usually obtained by magnitude pruning [14]. We denote the sparse sub-network as $W = W \odot M$. Sparse-to-sparse training regime optimizes the sparse weights $W$ and the corresponding binary mask $M$, which can be updated at each iteration or periodically. Variants of sparse training, such as unstructured sparsity and fine-grained structured sparsity [23, 25, 43, 9] achieve state-of-the-art performance on various convolutional networks with improved efficiency.

**Masked Autoencoders.** Masked Autoencoders are a family of self-supervised methods for pretraining Vision Transformers (ViT). They involve reconstructing masked RGB patches from visible tokens $x_{vis}$ with an encoder $f$ and a decoder $g$. Mathematically, we have

$$L_{MAE}(W) = \mathbb{E}(\|g_\theta(f(x_{vis}, W))) - x_{mask}\|^2). \quad (1)$$

Here, an encoder $f$ maps inputs $x_{vis}$ to latent features $z_{vis} = f(x_{vis})$, and a decoder $g_\theta$ parameterized by $\theta$ predicts the masked token $x_{mask}$ in the RGB pixel space from the latent feature $z_{vis}$. Encoder weights $W$ and decoder weights $\theta$ are jointly trained. After pretraining, the encoder is discarded and the encoder is finetuned on task-specific downstream datasets. During finetuning, the model acquires downstream specific knowledge $D_s$, which is built on general task-agnostic knowledge $G$ learnt in pretraining, as shown in Fig. 2 (a).

### 3.2. Challenges of Sparse Training for MAE

Despite the effectiveness of MAE for large-scale vision transformers, it encounters great difficulties when applied to smaller models (see Fig. 1). The inadequate model capacity prevents small models from learning meaningful task-agnostic knowledge and consequently transfers to inferior performance in downstream tasks. To enable the potential of large models’ capability for smaller ones to learn powerful representations under MAE framework, we can modify Eq. 1 to combine it with sparse training as follows:

$$L_{Sparse}(W, M) = \mathbb{E}(\|g_\theta(f(x_{vis}, W \odot M))) - x_{mask}\|^2), \quad (2)$$

where $f$ represents the large full encoder with parameters $W$. The encoder binary mask $M$ produces a sparse sub-network by magnitude pruning during pretraining, while the decoder is kept dense as it is not used for downstream finetuning. Following the notation in Sec. 3.1, we use $G^s$ to represent the general knowledge of the sparse sub-network learnt in pretraining, and we denote $D^s$ the specific knowledge of the finetuned sparse sub-network for downstream tasks. Considering both the pretraining and finetuning phases, we outline three intuitive strategies to integrate sparse training into the MAE framework:

- **1 Dense pretraining then sparse finetuning as shown in Fig. 2 (b).** Although this strategy allows the sparse finetuning phase to optimize from the large full model’s task-agnostic knowledge $G$, the sub-network does not benefit from the pretraining task and lacks its knowledge $G^s$. This may hinder the model to acquire better downstream knowledge $D^s$.

- **2&3 Sparse pretraining using Eq. (2) then sparse finetuning (see Fig. 2 (c)).** During finetuning, the sparsity mask can be kept as fixed or dynamic (dynamic mask is the third strategy). This strategy relies on the architecture and task-agnostic knowledge $G^s$ obtained by the sub-network during the pretraining phase, while it ignores the potential benefits brought by incorporating the knowledge $G$ of the large full model.

In summary, it is challenging to tailor a sparse training mechanism for MAE framework to sufficiently learn task-agnostic and task-specific knowledge for a small model. In the following section, we introduce our approach to solve this dilemma by fully utilizing the task-agnostic knowledge $G$ of the large full model and adaptively optimizing sub-network’s topology connections for various downstream tasks for better task-specific knowledge.

### 3.3. Sparse Masked Autoencoders

To tackle the limitations of the sparse training strategies introduced above, we propose a novel method for pruning ViT models under the MAE framework (see Fig. 2 (d)).

Firstly, to ensure a good general task-agnostic knowledge $G$ to be learnt via pretraining, we utilize the dense full model to reconstruct the RGB pixels following Eq. 1, which learns more discriminative representations thanks to its larger capacity. Simultaneously, a sparse sub-network within the full dense network is trained using the same visible tokens as the inputs and regresses the same targets as the large full model. The sparseMAE loss function is therefore extended from Eq. 1 and Eq. 2 as follows:

$$L = L_{Sparse}(\tilde{W}) + \alpha L_{MAE}(W), \quad (3)$$

where $\tilde{W}$ represents the dense parameters of the large full model, $\tilde{W} = W \odot M$ is the masked sparse parameters of the sub-network, and $\alpha$ is the scaling hyperparameter balancing the reconstruction losses for the dense full model and the
sparse sub-network. In our experiments, we set $\alpha = 1.0$ by default.

As shown in Eq. 3, the gradients for the sparse weights $\tilde{W}$ shall be updated following both the reconstruction losses of the sub-network and the full network, which may lead to gradient imbalance between the masked and unmasked weights [38]. Therefore, in each training iteration we split the weights $W$ of the full model into sparse sub-network weights $\tilde{W}$ and unmasked weights $\bar{W} = W \odot (1 - M)$ and optimize them separately:

$$
\tilde{W}_{t+1} = \tilde{W}_t - \eta (g(W_t \odot M_t) \odot M_t),
$$

$$
\bar{W}_{t+1} = \bar{W}_t - \eta (g(W_t) \odot (1 - M_t)),
$$

where $g(W_t \odot M_t) = \frac{\partial \mathcal{L}_{\text{Sparse}}(W_t \odot M_t)}{\partial W_t}$ are gradients of the sparse sub-network, $g(W_t) = \frac{\partial \mathcal{L}_{MAE}(W_t)}{\partial W_t}$ are the gradients of the full models, $\eta$ is the learning rate and $t$ indexes the training iteration. The binary mask $M_t$ is obtained by on-the-fly magnitude pruning. Here, we only show gradient descent for simplicity and omit other parts (e.g., weight decay). In this way, the large full model can utilize all the model parameters in the forward pass to produce discriminative outputs, while in the backward pass, the sparse sub-network stays unaffected by the gradients from the large full model. During the pretraining phase, the task-agnostic knowledge $G^*$ of the sparse sub-network can dynamically select the important sparse connections in the context of the task-agnostic knowledge $G$ of the full dense model, which helps to identify the important sparse topology and is retained for downstream finetuning.

After pretraining, the full dense model with a sparse sub-network is transferred and finetuned on different downstream tasks (e.g., classification, detection, segmentation, etc.). For the finetuning stage, the deep representations from the sparse sub-network are used for the downstream tasks, while its topology connections are optimized end-to-end given the specific downstream objectives. We refrain from fine-tuning dense full models, as the fine-tuning process for both dense and sparse models is costly. Hence, the learning of the task-specific knowledge $D^*$ for downstream tasks can benefit from inheriting a pretrained sparse sub-network with knowledge $G^*$, and adaptively selecting the more suitable sparse architecture built on the general knowledge $G$ of the full model.

4. Experiments

In this section, we validate the effectiveness and robustness of our SparseMAE. First, we introduce the implementation details of our SparseMAE in Sec. 4.1. Then, we present
our main experimental results in Sec. 4.2, which include classification results on ImageNet-1K [8] and semantic segmentation results on ADE20K [43]. To demonstrate the superiority of our proposed method, we compare SparseMAE with other state-of-the-art ViT pruning methods in Sec. 4.4. In Sec. 4.3, we evaluate the transferring ability of the models trained with our SparseMAE to object detection, instance segmentation tasks on MS COCO [24] and robustness evaluation. Finally, we perform ablative experiments to evaluate the performance of SparseMAE under different settings in Sec. 4.5.

4.1. Implementation Details

Sparse Training Methods. We conduct experiments with two different granularities for sparse training, consisting of unstructured sparsity [14, 21] and hardware-friendly N:M sparsity [28, 42]. As we focus on studying how to learn and transfer task-agnostic and task-specific knowledge under the Masked Autoencoders paradigm with sparse training methods, we select two simple and effective sparse training methods, namely DFP [23] for unstructured sparsity and SR-STE [42] for N:M sparsity. We prune all the linear layers within the vision transformers and set the relevant hyperparameters for sparse training the same as [23, 42] by default.

Models. For all of our main experiments, unless otherwise stated, we select ViT-Base as our large full model and prune it for smaller ones. For hardware-friendly N:M sparsity, we use 2:32 and 4:32 sparse ratios to obtain models with similar scales to ViT-Tiny (5.8M) and ViT-Small (11.3M), the main experiments use N:M sparsity for default, which is applied to all linear layers, including self.qkv, self.proj in the Attention block, and self.fc1 and self.fc2 in the MLP block. For unstructured sparsity, we extend the above N:M ratios and prune ViT-Base with 93.5% and 87.0% sparsity. We name our ViT-Base with 2:32/93.5% sparsity as SparseMAE-Small (SparseMAE-S).

Pretraining. Following MAE [15] pretraining scheme, models start from random initialization, we use ImageNet-1K (IN-1K) [8] for all pretraining experiments. We pretrain SparseMAE for 400 epochs using 224 × 224 images as inputs. The other settings also follow those used in [15]. For example, We use random resized cropping and horizontal flipping for data augmentation, a 75% mask ratio during pretraining, and a decoder with a single transformer layer and 512 hidden dimensions. Unlike other methods [34, 19] for small and tiny-scale vision transformers pretraining, we do not grid-search the decoder design, possible putting our method at a disadvantage.


For image classification, we take a ViT-Base model pretrained using MAE for 400 epochs as the reference model, which is implemented using the officially released codebase [15] and achieves 83.2% top-1 accuracy. We finetune SparseMAE-Tiny/Small for 200 epochs using the same receipt described in [15] for a fair comparison with other works [31, 19] and set the batch size as 1024 and the base learning rate as $10^{-3}$.

For semantic segmentation, we follow BEiT [2] to use UperNet[37] as our segmentation framework. We train the model with an input resolution of $512 \times 512$ for 160k iterations. A batch size of 16 is adopted. For object detection and instance segmentation tasks, we follow ViTDet [22] framework. We train our SparseMAE models with a batch size of 64 for 100 epochs, and the input image resolution is set to $1024 \times 1024$. Further details on the hyper-parameters can be found in Appendix.

4.2. Main Results

Classification and Semantic Segmentation. Tab. 1 shows the evaluation of our SparseMAE model against self-supervised methods for ViT-Tiny/Small on ImageNet-1K image classification and ADE20K semantic segmentation. For ImageNet-1K classification, SparseMAE achieves 80.5% top-1 accuracy. To our knowledge, this is currently the best result achieved by tiny-scale transformer models. Furthermore, SparseMAE outperforms TinyMIM [31]/G2SD [19]/MAE-lite [34] with +4.7/4.2/4.0 accuracy gains on tiny-scale transformers. For semantic segmentation, SparseMAE achieves an mIoU of 45.2, outperforming G2SD/MAE-lite/TinyMIM by +3.8/1.3/1.2 mIoUs. Similar improvements can be observed for the small-scale transformers. SparseMAE achieves a 83.2% ImageNet-1K classification accuracy and a 48.4% mIoU on ADE20K semantic segmentation, surpassing TinyMIM [31]/G2SD [19] by +0.2/1.2 and +0.0/2.2 on the two tasks respectively. The significant improvements over previous results show the superiority of our method in obtaining performant small models under MAE framework.

Speedups. Unstructured sparsity and N:M structured sparsity is efficient for commercial CPUs and FPGAs respectively. For our SparseMAE with unstructured sparsity, we tested the throughput (img/s) on an Intel RXeon(R) 6238R CPU using the deepsparse engine with a batch size of 64. For our SparseMAE with N:M sparsity, we tested the throughput on Xilinx XCVU13P with a batch size of 16. Our SparseMAE-S with N:M and unstructured sparsity achieved throughput gains of +71% and +9% over ViT-small with similar amount of parameters on FPGAs and CPUs. Therefore, our approach demonstrate high feasibility in actual scenarios.
Table 1: Finetuning results on ImageNet-1K and ADE20K. All our models are pre-trained only on ImageNet-1K. Extra pretrained teacher means the compared method cannot trained from scratch. ‡ the model adopts the MAE pretrained model before perform the sparse pretraining.

<table>
<thead>
<tr>
<th>Method</th>
<th>Params (M)</th>
<th>Pretraining epochs</th>
<th>Encoder ratio</th>
<th>Extra Pretrained/Teacher</th>
<th>Classification Top-1 Acc (%)</th>
<th>Segmentation mIoU (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DeiT [33]</td>
<td>5.8</td>
<td>300</td>
<td>-</td>
<td>Label</td>
<td>72.2</td>
<td>38.0</td>
</tr>
<tr>
<td>Tiny-scale</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAE [15]</td>
<td>5.8</td>
<td>1600</td>
<td>25%</td>
<td>×</td>
<td>71.6</td>
<td>37.6</td>
</tr>
<tr>
<td>MoCo [6]</td>
<td>5.8</td>
<td>300</td>
<td>100%</td>
<td>EMA</td>
<td>73.3</td>
<td>39.3</td>
</tr>
<tr>
<td>TinyMIM [31]</td>
<td>5.8</td>
<td>300</td>
<td>100%</td>
<td>TinyMIM-S</td>
<td>75.8</td>
<td>44.0</td>
</tr>
<tr>
<td>G2SD [19]</td>
<td>5.8</td>
<td>300</td>
<td>100%</td>
<td>ViT-B</td>
<td>76.3</td>
<td>41.4</td>
</tr>
<tr>
<td>MAE-lite [34]</td>
<td>5.8</td>
<td>400</td>
<td>25%</td>
<td>ViT-B</td>
<td>76.5</td>
<td>43.9</td>
</tr>
<tr>
<td>SparseMAE (ours)</td>
<td>5.8</td>
<td>400</td>
<td>25%</td>
<td>×</td>
<td>80.5</td>
<td>45.2</td>
</tr>
<tr>
<td>DeiT [33] (ours)</td>
<td>22</td>
<td>300</td>
<td>-</td>
<td>Label</td>
<td>79.9</td>
<td></td>
</tr>
<tr>
<td>Small-scale</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAE [15]</td>
<td>22</td>
<td>1600</td>
<td>25%</td>
<td>×</td>
<td>80.6</td>
<td>42.8</td>
</tr>
<tr>
<td>MoCo [6]</td>
<td>22</td>
<td>300</td>
<td>100%</td>
<td>EMA</td>
<td>81.4</td>
<td>43.9</td>
</tr>
<tr>
<td>DINO [3]</td>
<td>22</td>
<td>300</td>
<td>100%</td>
<td>EMA</td>
<td>81.5</td>
<td>45.3</td>
</tr>
<tr>
<td>CAE [5]</td>
<td>22</td>
<td>300</td>
<td>100%</td>
<td>DALL-E</td>
<td>82.0</td>
<td>-</td>
</tr>
<tr>
<td>TinyMIM [31]</td>
<td>22</td>
<td>300</td>
<td>100%</td>
<td>TinyMIM-ViT-B</td>
<td>83.0</td>
<td>48.4</td>
</tr>
<tr>
<td>G2SD [19]</td>
<td>22</td>
<td>300</td>
<td>25%</td>
<td>ViT-B</td>
<td>82.0</td>
<td>46.2</td>
</tr>
<tr>
<td>SparseMAE (ours)</td>
<td>11.3</td>
<td>300</td>
<td>25%</td>
<td>ViT-B‡</td>
<td>83.2</td>
<td>48.4</td>
</tr>
<tr>
<td>SparseMAE (ours)</td>
<td>11.3</td>
<td>400</td>
<td>25%</td>
<td>×</td>
<td>82.1</td>
<td>46.7</td>
</tr>
</tbody>
</table>

4.3. Evaluation on Other Downstream Tasks

Object Detection and Instance Segmentation. We finetune the SparseMAE pretrained models on MS COCO with ViT-Det [22]. As shown in Tab. 2, we report AP\textsubscript{bbox} for object detection and AP\textsubscript{mask} for instance segmentation. We compare SparseMAE-Tiny with MAE-Tiny and its improved variants using distillation methods, such as MAE-lite [34] and G2SD [19]. The results in Tab. 2 show that our SparseMAE obtains more than 3.1 AP\textsubscript{bbox} and 2.4 AP\textsubscript{mask} gains compared to MAE-lite and G2SD, validating the transferability of the learnt representations in our models.

Robustness Evaluation. In Tab. 3, we evaluate the robustness of our models on different variants of ImageNet validation sets. We use the same models finetuned on original ImageNet (Tab. 1) and only run inference on the different validation sets, such as Imagenet-R [17], ImageNet-A [17] and ImageNet-C [16]), without any specialized fine-tuning. Tab. 3 shows that our method has better generalization capability than distillation-based methods, such as G2SD [19] and TinyMIM [31]).

4.4. Comparison with State-of-the-art ViT Pruning

To the best of our knowledge, our proposed SparseMAE is the first to study network pruning technique under the Masked Autoencoders paradigm. To validate the effectiveness of SparseMAE, we compare it against recent state-of-the-art fully-supervised pruning methods for ViTs. We choose SViT [4] and oViT [20] as our compared models, which exploit end-to-end sparse training for ViTs and achieve state-of-the-art results in the fully-supervised setting. SViT [4] adopts a dynamic sparse training method to prune the dense model using random initialization. However, this approach requires training the sparse model for
Table 3: Robustness evaluation on ImageNet variants (top-1 accuracy, except for ImageNet-C which evaluates mean corruption error). We test the small-scale and tiny-scale models from Tab. 1 on different ImageNet validation sets, without any fine-tuning.

<table>
<thead>
<tr>
<th>Method</th>
<th>Model Size</th>
<th>IN-A↑</th>
<th>IN-R↑</th>
<th>IN-C↓</th>
</tr>
</thead>
<tbody>
<tr>
<td>DeiT [33]</td>
<td>Tiny-scale</td>
<td>11.0</td>
<td>39.8</td>
<td>50.1</td>
</tr>
<tr>
<td>MAE [15]</td>
<td></td>
<td>12.9</td>
<td>39.0</td>
<td>-</td>
</tr>
<tr>
<td>SparseMAE (ours)</td>
<td></td>
<td>18.2</td>
<td>45.9</td>
<td>47.2</td>
</tr>
<tr>
<td>DeiT [33]</td>
<td>Small-scale</td>
<td>18.3</td>
<td>42.3</td>
<td>41.4</td>
</tr>
<tr>
<td>MAE [15]</td>
<td></td>
<td>20.1</td>
<td>45.6</td>
<td>40.6</td>
</tr>
<tr>
<td>SparseMAE-T</td>
<td></td>
<td>27.5</td>
<td>48.8</td>
<td>35.8</td>
</tr>
<tr>
<td>SparseMAE-S</td>
<td></td>
<td>29.4</td>
<td>46.8</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 4: Comparison with state-of-the-art supervised Vision Transformer pruning methods on ImageNet-1K.

<table>
<thead>
<tr>
<th>Model</th>
<th>Method</th>
<th>Params (M)</th>
<th>Top-1 Acc (%)</th>
<th>Seg mIoU (%)</th>
<th>Corr. Strtg.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ViT-Base</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SparseMAE-T</td>
<td>D</td>
<td>34.6</td>
<td>81.3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SparseMAE-T</td>
<td>S</td>
<td>34.6</td>
<td>81.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SparseMAE-T</td>
<td>S+D</td>
<td>21.6</td>
<td>81.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SparseMAE-T</td>
<td>D+S</td>
<td>17.3</td>
<td>80.8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SparseMAE-S</td>
<td></td>
<td>8.7</td>
<td>79.7</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 5: Different sparse pretraining or finetuning strategies with tiny-scale models on ImageNet-1K and ADE20K. “S” means sparse pretraining, “D” means dense pretraining, and “D+S” means sparse pretraining together with a dense full model finetuning. ‡: the model is finetuned with the fixed sparse masks from pretraining. Corr. Strategy: corresponding sparse training strategies explained in Sec. 4.5.1.

<table>
<thead>
<tr>
<th>Model</th>
<th>Pretrain</th>
<th>Top-1 Acc (%)</th>
<th>Seg mIoU (%)</th>
<th>Corr. Strtg.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ViT-Base</td>
<td>D</td>
<td>80.4</td>
<td>45.4</td>
<td></td>
</tr>
<tr>
<td>SparseMAE-T</td>
<td>D</td>
<td>78.4</td>
<td>40.7</td>
<td></td>
</tr>
<tr>
<td>SparseMAE-T</td>
<td>S</td>
<td>76.3</td>
<td>37.2</td>
<td></td>
</tr>
<tr>
<td>SparseMAE-T</td>
<td>S+D</td>
<td>78.7</td>
<td>41.3</td>
<td></td>
</tr>
<tr>
<td>SparseMAE-T</td>
<td>D+S</td>
<td>80.5</td>
<td>45.2</td>
<td></td>
</tr>
</tbody>
</table>

4.5. Ablation Studies and Discussion

4.5.1 Comparison of Sparse Training Strategies.

We compare three sparse training strategies for MAE framework introduced in Sec. 3.2 under two sparse granularities. Namely, we (1) pretrain a large full model using MAE and finetune it with sparse training, (2) combine sparse training with MAE pretraining and subsequently finetune the model on downstream tasks with the fixed sparse mask from pretraining, or (3) modify (2) to use dynamic sparse mask for finetuning. The results are shown in Tab. 5. Under unstructured sparsity, comparing (1) and (2), we see that the classification performance drops from 77.8% to 76.7%, showing that the knowledge $G_s$ learnt by the sparse sub-network during pretraining cannot generalize well to the downstream tasks compared to the task-agnostic knowledge $G$ learnt by the large full model. (3) further allows the adaptation of sparse sub-network architecture during finetuning and partially alleviates the problem by better acquiring downstream specific knowledge $D_s$. However, its result still lags significantly behind our proposed method (77.6% vs. 80.4%), where the difference is only the absence of a concurrently pretrained large full model. On the other hand, when comparing (1) with our method, the addition of a pretrained sparse sub-network can boost the performance by 2.6%. Similar observations can also be made under N:M sparsity settings and semantic segmentation results. These empirical results suggest that both the task-agnostic knowledge $G$ of the large full model and $G_s$ of the sparse sub-network are necessary for small and tiny-scale vision transformers to effectively benefit from MAE pretraining.
4.5.2 Ablation of Full Model Capacity

We conduct an ablation study to investigate the performance with different capacities of the full model. We use our SparseMAE and set ViT-Small and ViT-Base as the full models, where ViT-Small only has 25% parameters of ViT-Base. As shown in Tab. 6, the results show that our SparseMAE in ViT-Base achieves 3.1% top-1 accuracy advantage over ViT-Small, indicating that larger capacity of the full model with SparseMAE can significantly improve its performance by incorporating better pretraining task-agnostic knowledge.

4.5.3 Impact of Different Pruning Sparsity Ratios

The main experiments of sparse ratio setting is compared to the ViT-tiny and ViT-small. We provide additional sparse ratios to verify the performance of SparseMAE on ImageNet-1K as in Tab. 7.

Table 7: Comparison between different sparse ratios for SparseMAE on ImageNet-1K.

<table>
<thead>
<tr>
<th>Sparse Pattern</th>
<th>1:32</th>
<th>2:32</th>
<th>4:32</th>
<th>8:32</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top-1 Acc</td>
<td>76.8</td>
<td>80.5</td>
<td>83.2</td>
<td>83.7</td>
</tr>
</tbody>
</table>

5. Conclusion

In this paper, we introduce SparseMAE, an innovative method for incorporating sparse training into the Masked Autoencoders (MAE) framework to create efficient and powerful small and tiny-scale transformers. As the first study in this area, we first point out the limitations of sparse training under MAE and design a unified framework to transfer the task-agnostic and task-specific knowledge of large models to lightweight sparse sub-networks. SparseMAE trains the sparse sub-networks by persevering the task-agnostic knowledge of large full models and then adaptively finds the optimal specific sub-networks for downstream tasks. The proposed method outperforms the state-of-the-art small-scale vision transformers in both unsupervised pretraining and fully-supervised settings. We hope this work could shed light on sparse training with the Masked Autoencoders framework in the vision community.

Acknowledgement

This project is funded in part by National Key R&D Program of China Project 2022ZD0161100, by the Centre for Perceptual and Interactive Intelligence (CPII) Ltd under the Innovation and Technology Commission (ITC)’s InnoHK, by General Research Fund of Hong Kong RGC Project 14204021. Hongsheng Li is a PI of CPII under the InnoHK.

References


