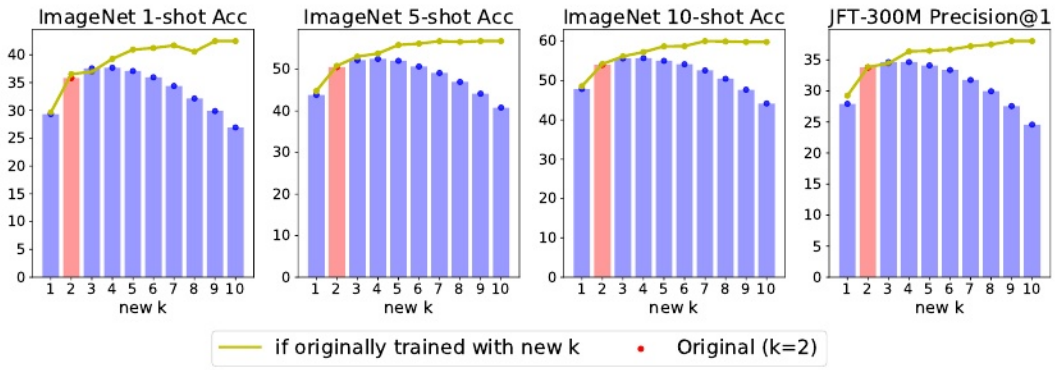


图34:k = 2训练的原始V-MoE-S/32 every-2模型。

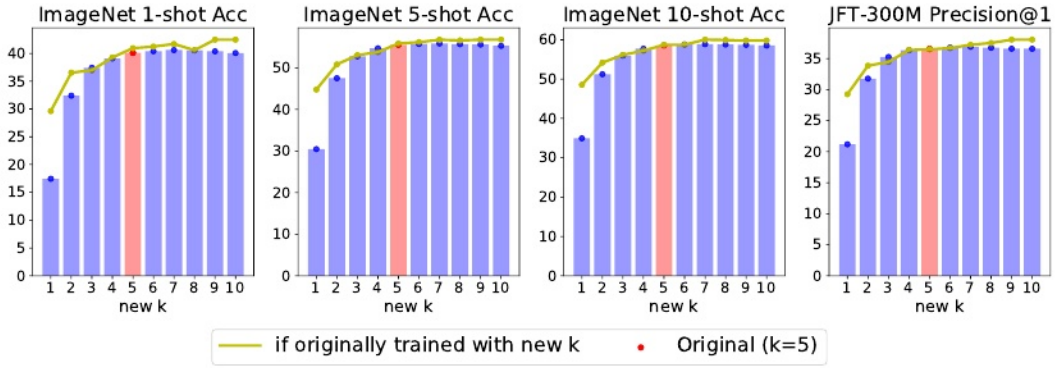


图35:k = 5训练的原始V-MoE-S/32 every-2模型。

K ={1，…，9}专家。然后用不同的k对这些模型进行微调，我们在图36中展示了这个结果。正如我们之前的结果所期望的那样，通常增加k会提高性能。无论上游k如何，通常在微调期间通过增加k来提高精度。同样，在预训练期间增加k也会提高下游的性能。

相反，当k = 1下游时，所有模型都无法从上游k更高的预训练中提高。用k > 1预训练的模型似乎学会了组合专家输出，因为它们不能很好地泛化到选择单个下游专家，并且失去了k更大的预训练的好处。

E.6 用较少的数据进行预训练

我们已经证明，使用大数据集进行预训练的标准配方允许在可用数据较少的下游视觉任务上使用强大的稀疏模型。问题自然出现了:这些模型需要大量的上游数据吗?我们在这里就这个方向提出一些初步的探索。

在数据较少的JFT300M上进行训练。我们首先在JFT300M的子集上训练V-MoE-L/32。[20]中的密集模型也是这样做的，在图37中，我们直接与他们的结果进行了比较。V-MoE最初似乎对减少的数据相当稳健，但在减少到9M个预训练样本(数据集的3%)之后，它变得稍微更适合训练密集模型。

在ImageNet21k上的训练。ImageNet21k[16]是一个大型公共数据集，大约有14M张图像和21k个类。之前的作品[20,36]已经在其上成功地进行了预训练，在下游任务中取得了较强的效果。特别是，在ImageNet21k上训练的密集ViT模型表现得相当好。除了ViT-S, V-MoE的性能立即优于密集的对应对象，应用稀疏缩放通常会损害性能。我们观察到过拟合，两者

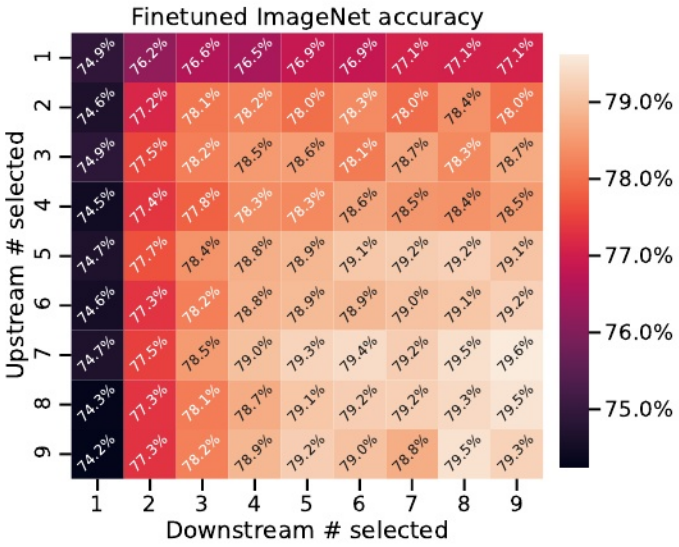


图36:使用不同k值预训练的V-MoE-S/32模型在微调/推理时间下k(所选专家数量)的变化

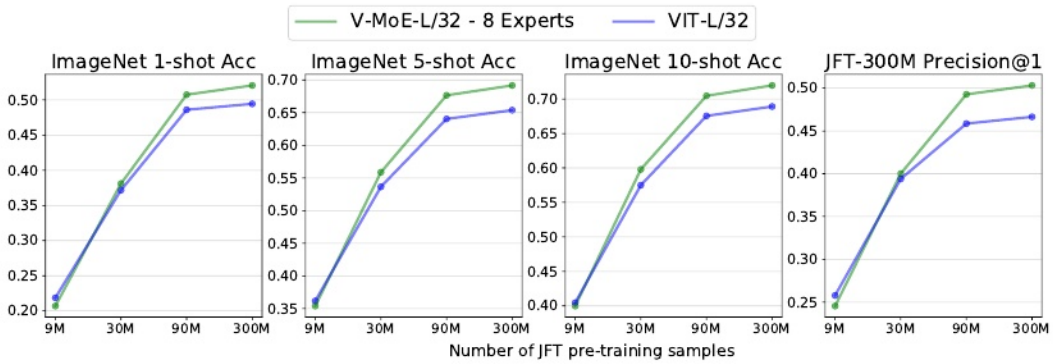
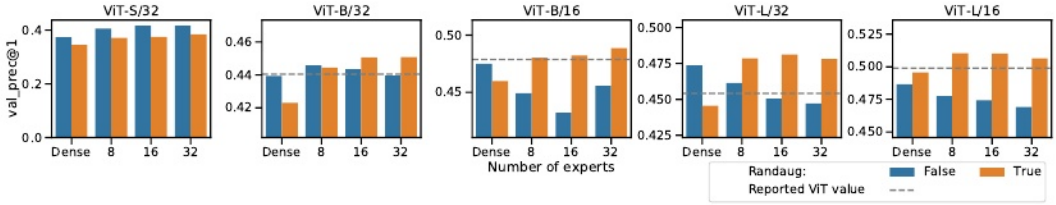


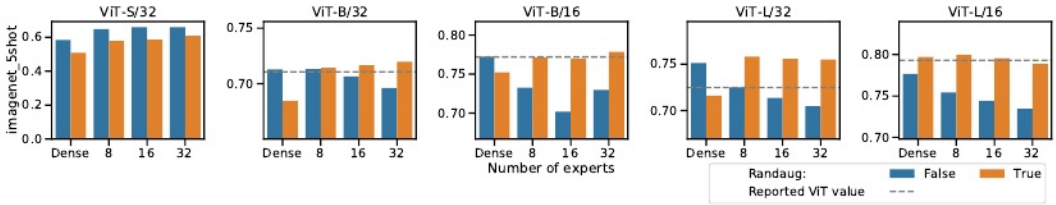
图37:改变预训练数据量的效果。我们比较了V-MoE-L/32和viti - l /32在增加数据大小方面的性能。特别是,我们取JFT-300M的子集，分别有9M、30M、90M和300M数据点——注意，完整的数据集包含大约305M数据点。考虑到我们使用较小的数据集进行训练，我们决定使用8位专家而不是32位(每隔2次)。在最低的数据量下(9M，约为原始数据的3%)，MoE模型无法利用其额外的容量。对于剩下的数据，从30M开始(大约是原始数据集的10%)，就可以了。

在降低预训练数据集上的验证精度的意义上，随着训练的继续，也会降低迁移性能。作为解决这个问题的初步尝试，我们使用RandAugment[14]进行N = 2个量级M = 10的变换。如图38所示。有趣的是，RandAug通常会在伤害密集模型的同时帮助专家模型。有了这个应用，对于每个架构，都有一个优于密集基线的专家模型。

这还远远不是一个完整的探索;它表明，这些模型可以在较小的数据源上工作，其有效性的关键可能在于更仔细地考虑数据增强和正则化。我们希望最近在密集变压器[32,60]中探索这一点的工作在这里是有用的，并且在数据高效视觉变压器[59,65]中也可以进一步释放V-MoE的潜力，减少预训练数据。



(a) Precision@1 on the ImageNet-21k validation set



(b) 5-shot linear ImageNet performance

图38:ImageNet-21k预训练模型的性能。