Jacob Haywood

801182240

Assignment 6

Github: https://github.com/jhaywoo6/Deep_Learning/tree/main/Assignment6

This assignment aims to find out if Vision and Swin Transformers are applicable to use on small scale systems and image datasets. The first problem will focus on training four randomly configured Vision Transformers on the CIFAR100 dataset with varying patch sizes, embedding dimensions, transformer layers, and attention heads to determine their effects on total parameters, FLOPS per forward pass, training time, and accuracy. These four will be compared to a ResNet-18 model with a similar setup. Both models will use an effective batch size of 64 and an Adam optimizer with a learning rate of 0.001. The epochs for the vision transformers have been randomly selected between 20-50, while the ResNet-18 model will run over 10 epochs. Gradient Accumulation is used to reduce VRAM usage while matching the requested batch size. 16-bit floats are also used to further reduce VRAM usage. The second problem will focus on Swin transformers, a variant of the traditional Vision Transformer. Two pretrained models will be compared with a variant from scratch training on the CIFAR100 dataset. Unlike problem one, the parameters will be consistent across all tested models apart from learning rate, which will be increased from 2e-5 to 0.001 for the scratch model. The models will be trained over 5 epochs with a batch size of 32. Effects on total parameters, FLOPS per forward pass, training time, and accuracy will be measured. Gradient Accumulation and 16-bit floats were not found to be necessary for training these models. The models will be trained on an Nvidia RTX 4070.

Problem 1:

Model	Average Time (s)	Num Parameters	FLOPs	Accuracy (%)
ViT-A	232.9332767	17374820	1.07E+11	0.01
ViT-B	943.3577144	18505316	4.24E+11	0.01
ViT-C	66.19944401	3435364	5.41E+10	0.01
ViT-D	416.6080148	7159652	4.24E+11	0.01
ResNet-18	47.26588571	11227812	5.84E+10	0.5479

Patch_Size	Embed_dim	Heads	Layers	Epochs
8	512	2	8	29
4	512	4	8	29
8	256	2	4	27
4	256	2	8	27
				10

Across all ViT's, the accuracy results are not good. It appears altering these parameters does not affect the accuracy to result in a meaningful difference. What they do affect is the training time, parameter count, and FLOPS per forward pass. A smaller patch size, higher embedded dimension, heads, and layers are all associated with more parameters, higher FLOPS, and a longer training time. It was found more VRAM was required for these circumstances as well. Patch Size had the greatest impact on these results. ResNet-18 preformed much better than all the ViT's with a lower training time and decent accuracy. The FLOPs are on the low end compared to Vit's, while the parameter count is between that of the smallest and largest ViT models. It was thought that in assignment 5 the transformer model may have been set up incorrectly leading to poor accuracy, but with two assignments providing poor accuracy compared to non-transformer models, it can be concluded that these small datasets simply do not work well with fresh transformers. But what if we use a model that was pretrained on a significantly larger dataset?

Problem 2:

Model	Average Time (s)	Num Parameters	FLOPs	Accuracy (%)
Tiny	110.23	27596254	1.40E+11	0.5511
Small	161.82	48914158	2.73E+11	0.5949
Scratch	261.24	18918736	1.40E+11	0.0491

The Tiny and Small Swin Models are pretrained models from Hugging face transformers library. These models have been trained on a very large dataset, and this test used the CIFAR100 dataset to refine the search results. The results are significantly better than the scratch model. These models have significantly more parameters, although the FLOPs aren't significantly more than those of the scratch model. These models likely outperformed the scratch model due to seeing patterns in larger datasets than that of the CIFAR100 dataset and having more parameters which leads to better image recognition. The accuracy even exceeds that of the ResNet-18 model with fewer epochs. While the Small model is much larger than the Tiny dataset, the accuracy gain is relatively small. The element of being pretrained may be more important than the size of the model itself. Both were able to pick up similar patterns, the larger model was just able to see a few more connections. Further improvements could be achieved with a significantly larger model,

more training time, or by training a scratch model with a very large data set that has similarities to the CIFAR100 dataset on the hardware and timeframe these pretrained models were trained on.