Inference Acceleration

Diffusion：

Post-training Quantization on Diffusion Models

- Quantize diffusion 中降噪用的神经网络
- Post-training
- 8-bit 时能接近甚至超过原模型性能
- 针对 diffusion 有 time step 的特性改进了传统的 Post-training Quantization 方法
- GPU 使用量：未给出，但 post-training 应该比较可控
- 代码完备度：4/10

Q-Diffusion: Quantizing Diffusion Models

- 贡献和上一篇类似，性能更好，能做到 4bit 时性能损失较小
- GPU 使用量：未给出
- 代码完备度：8/10

Efficient Spatially Sparse Inference for Conditional GANs and Diffusion Models

- 对于编辑图片的 conditional GAN 和 diffusion，缓存每一小步编辑时的 feature map
- GPU 使用量：很小
- 代码完备度：9/10，甚至支持苹果 M 系列芯片

Accelerating Diffusion Sampling with Classifier-based Feature Distillation

- Distill diffusion，在传统的 teacher-student 以外用一个独立的 classifier 提取出重要特征，重点训练
- GPU 使用量：提供 checkpoint，应该不大
- 代码完备度：5/10

Transformer：

GPTQ

- After-training quantization

    - Traditional compression methods need finetuning/retraining, which is costly for large models -> **post-training one-shot methods**
    - **extremely accurate language models with hundreds of billions of parameters can be quantized to 3-4 bits/componen**
    - While all existing works—ZeroQuant (Yao et al., 2022), LLM. Int8 () (Dettmers et al., 2022), and nuQmm (Park et al., 2022)— carefully select quantization granularity, e.g., vector-wise, they ultimately just round weights to the nearest (RTN) quantization level, in order to maintain acceptable runtimes for very large models.
    - OBQ quantizes weights in greedy order, i.e. it always picks the weight which currently incurs the least additional quantization error
- GPU 使用量：大约 A100 4 小时
- 代码完备度：7/10

SparseGPT: Massive Language Models Can Be Accurately Pruned in One-Shot

- 对于千亿级大模型，能够 one-shot 剪枝达到约 60%的 sparsity，无需重新训练
- GPU 使用量：大约 4 小时
- 代码完备度：6/10

AWQ: Activation-aware Weight Quantization for LLM Compression and Acceleration

- 对对模型结果影响最小的权重进行剪枝
- 保留不同领域的 generality
- 3x 加速
- GPU 使用量：未给出
- 代码完备度：8/10

The Lazy Neuron Phenomenon: On Emergence of Activation Sparsity in Transformers

- 实验得出对于几乎所有模型和数据集，都只有 5%左右的神经元数值非零
- 只保留 top-k 的 neuron 有较好的结果
- 无代码

Knowledge Distillation of Large Language Models

- 提出一个更适合 LLM 的衡量 student 与 teacher 之间差异的 metric，与更适应这个 metric 的优化方法
- 可以 scale 到百亿级
- GPU 使用量：未给出
- 代码完备度：7/10

Distilling Step-by-Step! Outperforming Larger Language Models with Less Training Data and Smaller Model Sizes

- 结构化 query teacher model，监督 student
- 需要大约 70%的训练数据，能够匹敌大 3 个数量级的模型
- 代码完备度：6/10

Framework:

Pytorch: Accelerating Generative AI with PyTorch II: GPT, Fast

- `torch.compile` + quantization + speculative decoding