Autoregressive Model Beats Diffusion: Llama for Scalable Image Generation (LlamaGen)

1. 论文工作内容

总体来看，文章的要点总结：

- 一种更优的 **Image tokenizer**，（当其下采样比例为16时，**codebook的利用率达到了97%**），具体的设计学习了 VQ-GAN中的encoder-quantizer-decoder 模式，具体设计的时候：codebook的向量维度很小，codebook size 设置的很大；
- **Scalable image generation model**，具体就是**Llama架构**，证实了**不添加视觉信息归纳偏置的原生自回归模型**也可在图像生成中work；
- ***High-quality training data***，具体：先在LAION-COCO上50M的子数据集上训练一个775M的text-conditional图像生成模型，然后在***10M私有的高质量图片数据上进行微调***；（可能在后续的对比中，这也是227M的 VQGAN 与111M的 LlamaGen 相差太多的原因？不确定是否只是在benchmark上对比但训练方式不一样。）
- **使用vLLM框架加速推理速度。**

具体看 image tokenizer的设计：

论文中具体阐释了 LlamaGen 的tokenizer设计，其实整个设计的流程已经损失函数的约束都跟VQ-GAN一致，明显差异处在：LlamaGen 的**codebook的维度很小**（8），**尺寸很大**，某种程度上**解决了 VQVAE中编码表坍塌问题**（编码表增大时，编码表并没有被充分利用起来，反而由于恶性竞争导致编码表聚集到一块。）

Autoregressive Models 实现Image Generation的思路：

- ***Class-conditional image generation***

  维护可学习的 embeddings，从中索引出 ***具体class embedding 作为 prefilling token embedding*** (Autoregressive 的第 -1 个Token)，接下来就是next-token prediction 模式来做生成；

- ***Text-conditional image generation***

  不同于如 Stable Diffusion 等模型加入 Text-condition 的模式（通过cross attention来融合不同domain 的信息），**Text condition 先通过 text encoder (CLIP/)获得 text feature**, text feature 通过**额外的MLP映射为**Autoregressive models的 **prefilling Token**。

- ***Classifier-free guidance***

  训练过程中，**条件信息**被**随机 drop** 并被**替换为 unconditional的embedding**；**推理**时，**每一个token的logits值**由条件约束下的logits(c)和无条件约束下的logits(u)以scale参数线性组合。

2. 思考整理

首先最 interesting 的点在于 其 image tokenizer 的设计

a. 证实了**离散表征**的方式是**可以 scale up**的（其实也就是codebook增大后不会出先灾难性的codebook坍塌）
b. 看具体的设计 感觉跟VQGAN等的差异性主要体现在 本作的**codebook中token的维度小（8，对比VQGAN为256）**，尺寸数目大（但有效利用率为97%）。这里在思考是否原本高维空间下出现的恶行聚集情况，其所聚集的区域其实就是当下学习到的这个空间维度只有8的空间呢？那是否跟数据类型质量等挂钩，以及这个空间的维度也会随着不同数据类型而重新去探索最佳的dim?
c. 具体的一些设计可能需要**结合代码**去看，暂时还未去看，添加到to-do-list中。（TODO）

关于自回归模型去进行图像生成的方式

a. 本作中使用自回归模型去做图形生成，具体的范式都是将**condition信息作为prefilling token** 去引导接下来的生成，主观上我不确定这种方式**是否可以进行充分的 condition信息利用**，以及**信息的利用是否可以解耦可控**，是否有其他方法？
b. 关于自回归式的生成模型，之前又看到 T2M-GPT 也是使用这种自回归方式去做Motion生成，但本质上使用的还是原生的 VQ-VAE，不知道这篇文章***是否可以进行一个研究的启发：寻找一个更好的 Motion tokenizer***？