Mixture of Experts (MoE)



Contents

1. 奠基工作: 90 年代初期

1991, Hitton, Adaptive Mixtures of Local Experts

2. 架构形成: RNN 时代

2017, Google, Outrageously Large Neural Networks

3. 提升效果: Transformer 时代

- 2020, Google, GShard
- 2022, Google, Switch Transformer

4. 智能涌现: GPT 时代

- 2021, Google, GLaM
- 。 2024, 幻方量化, DeepseekMoE/ Deepseek V2/ Deepseek V3





视频目录大纲

MoE 基本介绍 MoE 前世今生 奠基工作: Adaptive Mixtures of Local Experts LSTM MoE: Outrageously Large Neural Networks ... GShard: Scaling Giant Models with Conditional ... MoE 核心工作 Switch Transformers: Scaling to Trillion Parameter Models ... MoE 混合专家 GLaM: Efficient Scaling of Language Models with Mixture ... 好好想想// ST-MoE: Designing Stable and Transferable Sparse Ex MoE 架构原理 大模型遇上 MoE 手撕 MoE 代码 MoE 的未来



Switch Transformer

基本介绍

- 2022年4月,距离ChatGPT发布还有半年,Google发布了《Switch Transformers: Scaling to Trillion Parameter
 Models with Simple and Efficient Sparsity》(实际上2021年Google就提出Switch Transformer了)。
- Switch Transformer和GShard一样,是encoder-decoder结构,基于T5开发的,具有1.6T的参数,2048个expert。
- 和前面的很多工作一样, Switch Transformer有一个出发点,那就是参数量越大,模型效果越好,并且可以 通过稀疏激活来减少总计算量。



基本介绍

• 但是相比其他工作, Switch Transformer给出了一个更为具体的描述, 那就是模型参数量可以是一个独立于总计算量的, 单独的缩放轴。也就是说, 在改变参数量的同时, (几乎) 不改变训练和推理的计算量, 就可以带来效果的提升。因此Switch Transformer关注在"同样的FLOPS/token的计算量"下, 如何扩大模型, 提升效果。Switch Transformer所做的工作还是比较多的,包括:

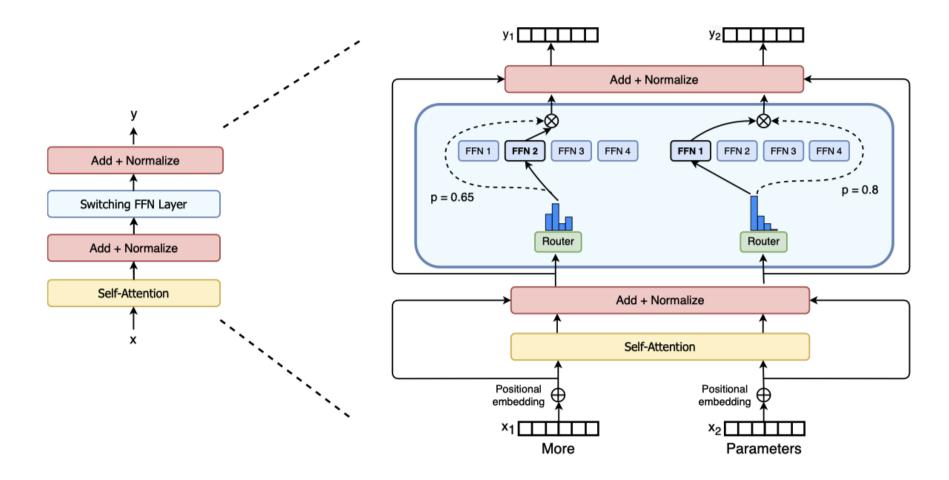
- 1. 模型结构简化:简化了Transformer上的MoE架构,提出Switch Transformer架构。
- 2. MoE to dense:把训出来的效果较好的MoE模型蒸馏到dense模型,在压缩MoE模型99%的参数的情况下, 效果还是比直接训练dense模型好。

基本介绍

- 训练和微调技术:首次使用bf16成功训练MoE模型;更适合MoE结构的模型初始化;增加的专家正则化, 改善了稀疏模型的微调和多任务训练。
- 2. 训练框架:结合数据、模型和专家并行性,训练了超过1T参数的MoE模型。
- 3. 多语言: 在多语言数据集上训练, 发现101种语言效果普遍有提升。
- 4. 训练效率:在同样的FLOPS/token的计算量下,Switch Transformer模型收敛速度有数倍的提升。

模型设计

• Switch Transformer的模型结构如下图,类似GShard,把transformer每层的FFN替换成MoE层





模型设计

- Switch Transformer一个重要的改进点就是简化了gating function的做法(Switch Transformer论文里叫routing)。
- 之前的工作大多探索了选择k个expert的做法,而Switch Transformer则直接把gating简化为只选择1个expert,即k=1。这样的MoE层叫做Switch layer。
- 这样简化之后, routing的实现更简单, router的计算量小了, 也减少了通讯量。



负载均衡

- 同GShard一样,Switch Transformer规定了一个专家容量expert capacity,来限制每个expert在一个batch里能 处理的最大token数。
- 如果一个token被分配到了一个已经满载的expert,就会出现overflow,那这个token在本层就不会被处理,而是直接通过残差链接,透传给下一层。这点也同GShard一样。
- 在Switch Transformer,专家容量通过容量系数capacity factor来控制。一个大的capacity factor意味着每个expert能够处理更多的token,从而减少overflow情况的发生,但是计算量和通讯量的压力也会增大,所以这是一个需要权衡的参数。

负载均衡

- 经验上,低的token丢弃率对模型的scaling很重要,想要训练超大规模的模型,就要解决这个问题。而通过负载均衡损失就可以确保良好的平衡,使得在使用较小容量系数的情况下,overflow尽量少,从而兼顾效果和计算速度。
- 关键问题来到负载均衡损失怎么设计。





把AI系统带入每个开发者、每个家庭、 每个组织,构建万物互联的智能世界

Bring Al System to every person, home and organization for a fully connected, intelligent world.

Copyright © 2024 XXX Technologies Co., Ltd. All Rights Reserved.

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. XXX may change the information at any time without notice.



 $Git Hub\ https://github.com/chenzomi I\ 2/AlInfra$

引用与参考

- https://mp.weixin.qq.com/s/6kzCMsJuavkZPG0YCKgeig
- https://www.zhihu.com/tardis/zm/art/677638939?source id=1003
- https://huggingface.co/blog/zh/moe
- https://mp.weixin.qq.com/s/mOrAYo3qEACjSwcRPG7fWw
- https://mp.weixin.qq.com/s/x39hqf8xn1cUlnxEIM0 ww
- https://mp.weixin.qq.com/s/ZXjwnO103e-wXJGmmKi-Pw
- https://mp.weixin.qq.com/s/8Y281VYaLu5jHoAvQVvVJg
- https://blog.csdn.net/weixin 43013480/article/details/139301000
- https://developer.nvidia.com/zh-cn/blog/applying-mixture-of-experts-in-llm-architectures/
- https://www.zair.top/post/mixture-of-experts/
- https://my.oschina.net/IDP/blog/16513157
- PPT 开源: https://github.com/chenzomi12/AlInfra
- 夸克链接: https://pan.quark.cn/s/74fb24be8eff

