

《网络测量与分析技术》报告

软硕231 王泽文

软硕232 丁浩宸

选取论文

Jie You, Jae-Won Chung, & Mosharaf Chowdhury. (2022). Zeus: Understanding and Optimizing GPU Energy Consumption of DNN Training.

背景

随着神经网络在不同领域的大量使用，以及神经网络规模的不断增大，人们对gpu计算的需求也不断增大，进而导致了能量消耗的不断增大，比如训练GPT-3模型大概会消耗1287MWh，相当于120年的美国住宅平均用电量。然而此前人们还没有研究过gpu以及训练过程中的一些参数对能量消耗可能带来的影响，只是一味使用最大的power limit来追求最短的时间。

作者针对这一现象，提出了一个名为Zeus的框架，可以通过额外的训练来学习到gpu最佳的power limit以及训练过程中的batch size来优先降低能量消耗，同时也尽可能对时间进行优化。在论文中，作者声称Zeus系统能够对各种场景下的DNN网络训练达到15.3% - 75.8%的GPU功耗优化效果，同时能降低60.1%的训练时间。

随后我们在较低级的gpu上进行了复现实验，并模拟实验室环境的要求进行了额外实验，并没有取得如此显著的优化效果，最后总结了这种差异出现的原因，并分析了这一系统在实际使用中的特点以及一些缺陷。

方法

Zeus的工作流程如下：1.用户上传训练任务以及可以实现的batch size以及power limit；2.Zeus通过过去的执行历史来预测优化方案；3.将这个方案应用在训练任务上；4.收集训练过程的数据，包括准确率以及能量消耗等，将这些数据重新提交给Zeus，让它从这些数据中持续学习并且改变内部状态。

具体来说，Zeus会首先列出所有可能的batch size，从中挑选一个开始训练。然后从power limit列表中大到小遍历，当该次power limit没有历史数据时，就会在一小段时间内使用该power limit进行训练，记录该段时间的能量消耗，并根据cost公式算出cost记录下来。cost公式如下：

$$\begin{aligned} & \text{EpochCost}(b_t; \eta) \\ &= \min_{p_t \in \mathcal{P}} \frac{\eta \cdot \text{AvgPower}(b_t, p_t) + (1 - \eta) \cdot \text{MAXPOWER}}{\text{Throughput}(b_t, p_t)}. \end{aligned} \quad (7)$$

在遍历了所有power limit后，Zeus会选择一个最优的power limit并进入稳定训练状态。在达到训练目标停止后，Zeus会根据本次训练内的表现来修改该batch size对应的分布均值以及方差。在对所有batch size也进行遍历之后，每次训练前Zeus都会进行采样来确定下一次训练的batch size，并且根据训练cost继续修改该batch size的分布。具体公式如下：

Input: Batch size b and observed cost C
Previous cost observations C_b for b
Belief prior parameters $\hat{\mu}_0$ and $\hat{\sigma}_0^2$
Output: Belief posterior parameters $\hat{\mu}_b$ and $\hat{\sigma}_b^2$

Function Observe ($b, C, C_b, \hat{\mu}_0, \hat{\sigma}_0^2$):

```

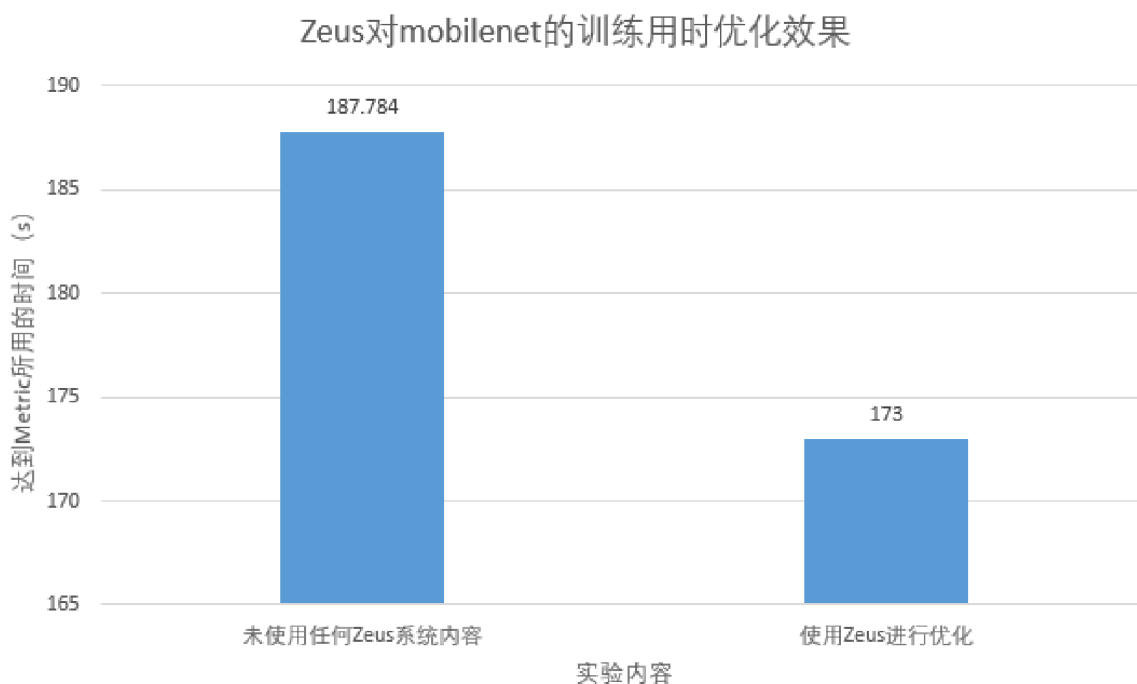
1  /* Add the most recent cost observation to history */
    $C_b \leftarrow C_b \cup \{C\}$ 
   /* Compute the variance of the cost */
2   $\tilde{\sigma}^2 \leftarrow \text{Var}(C_b)$ 
   /* Compute the belief distribution's posterior variance */
3   $\hat{\sigma}_b^2 \leftarrow \left( \frac{1}{\hat{\sigma}_0^2} + \frac{|C_b|}{\tilde{\sigma}^2} \right)^{-1}$ 
   /* Compute the belief distribution's posterior mean */
4   $\hat{\mu}_b \leftarrow \hat{\sigma}_b^2 \left( \frac{\hat{\mu}_0}{\hat{\sigma}_0^2} + \frac{\text{Sum}(C_b)}{\tilde{\sigma}^2} \right)$ 

```

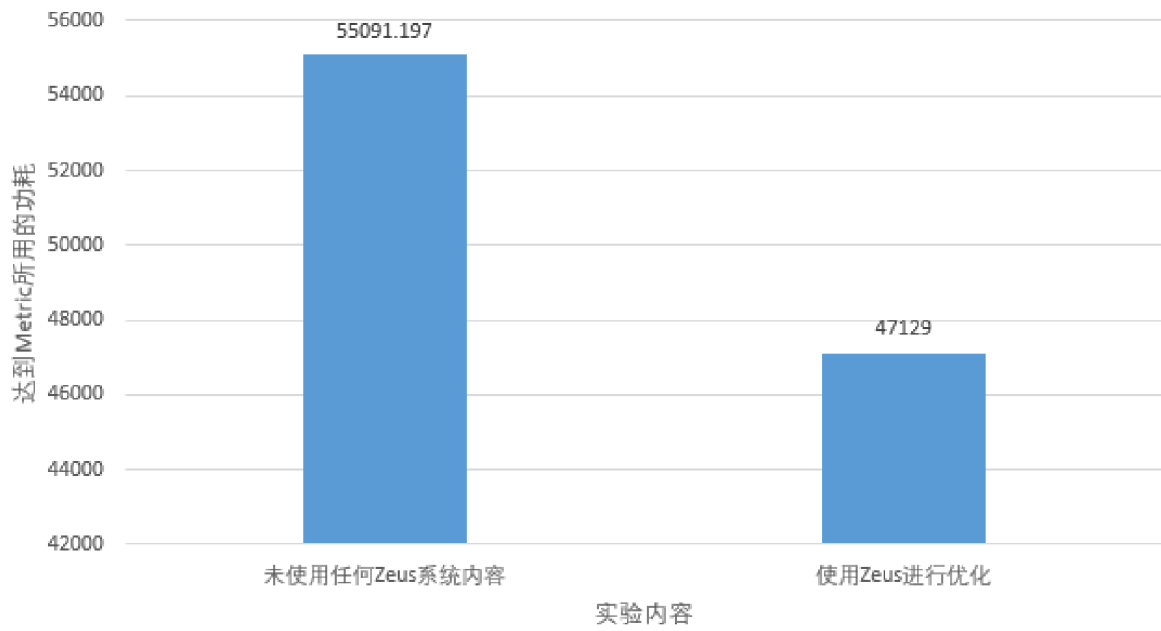
此外，还需要对一部分不能达到训练目标的batch size进行剪枝。

实验结果

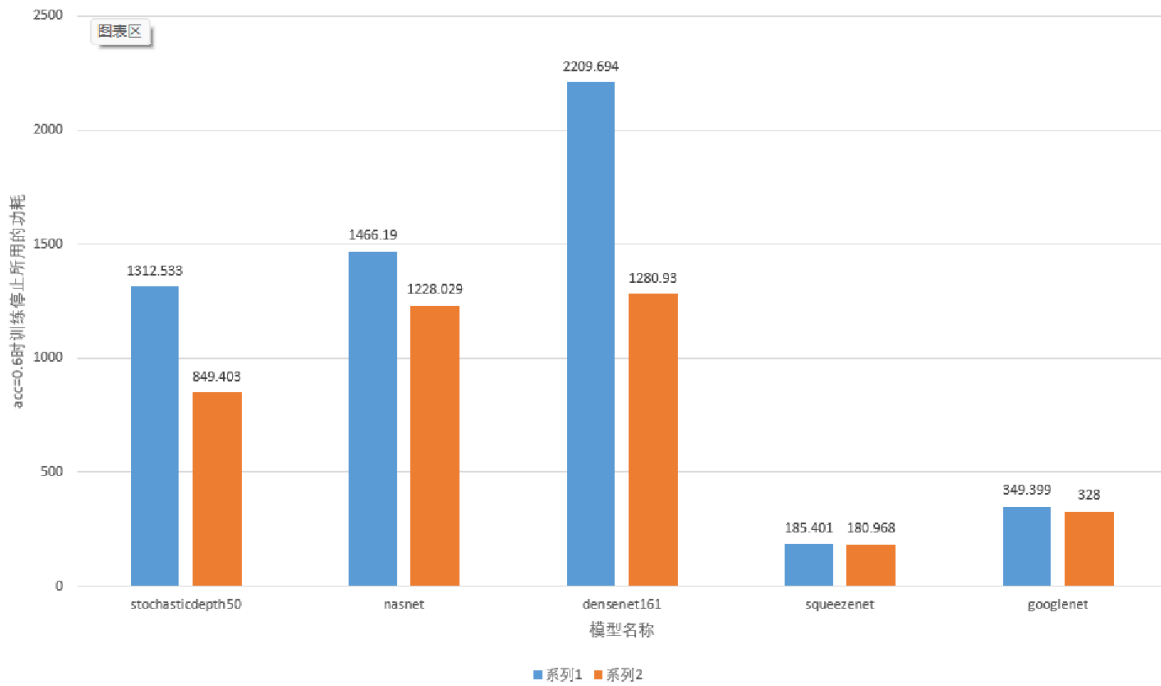
我们对在mobile net以及其他网络中运行了Zeus完整优化流程的实验，选取Cifar100数据集，每组实验允许跑50次，目标准确率设定为0.5，取Zeus在稳定运行阶段的数据进行平均，结果如下（蓝色为优化前，橙色为优化后）：



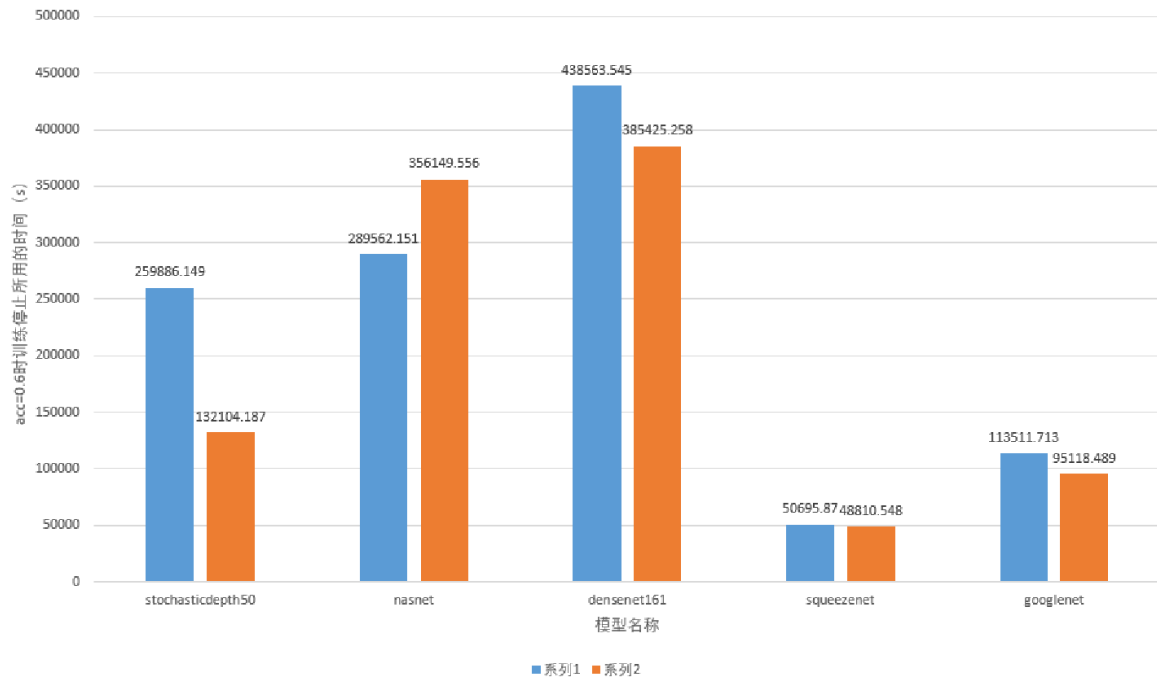
Zeus对mobilenet的功耗优化效果



Zeus对其他模型的训练功耗优化效果

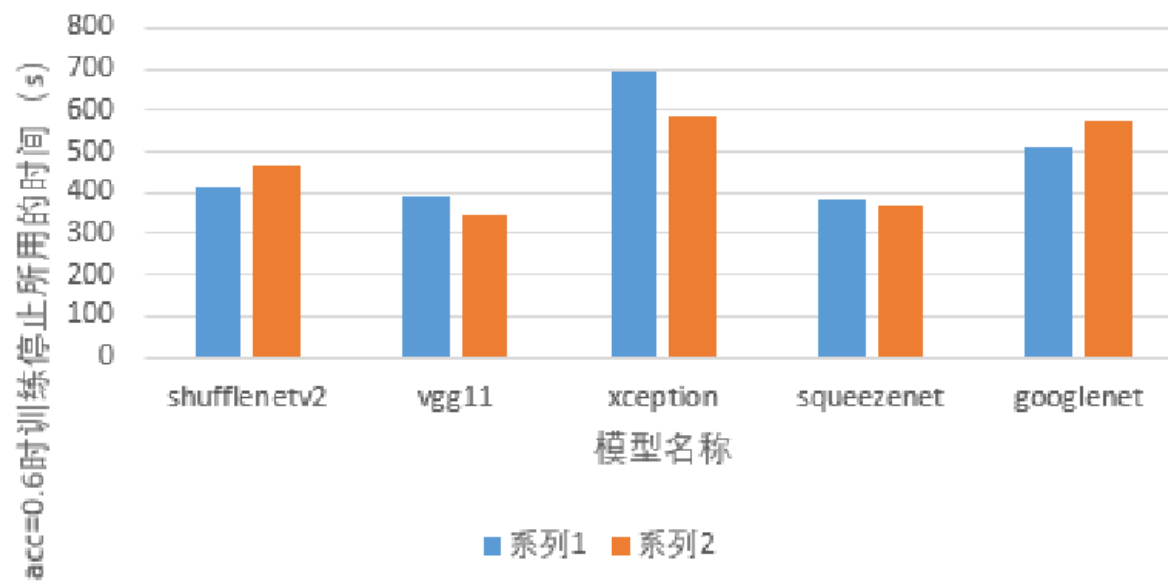


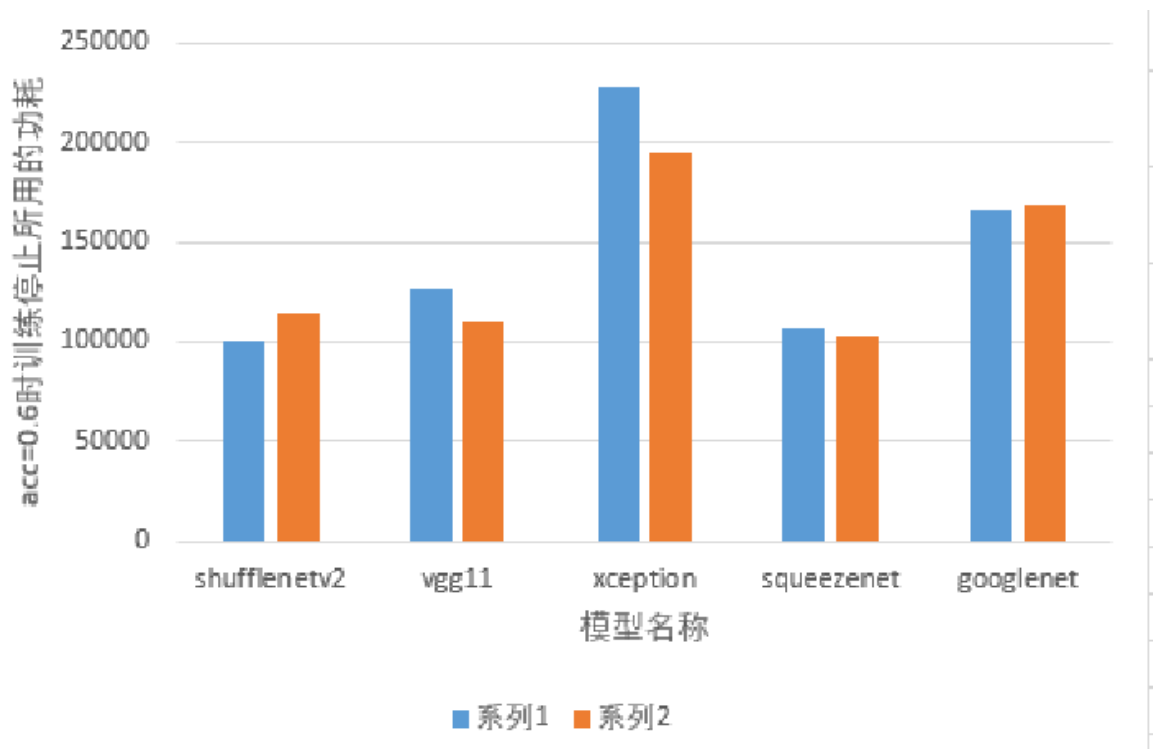
Zeus对其他模型的训练时间优化效果



可以看到，复现实验的结果并没有取得论文中那样好的结果，对于实验中的所有模型，Zeus均可以在能耗上取得不同程度的优化，在5%到40%左右，但时间上大部分模型优化程度就更低，甚至出现了负优化。

随后我们还模拟实验室环境，只对power limit进行单轮优化，得到的结果如下（蓝色为优化前，橙色为优化后）：





可以看到，如果不使用Zeus的全部优化策略的话，会出现更多的负优化情况，总体来说和不使用Zeus没有太大差别。

总结

我们对Zeus这一旨在降低gpu训练过程中能量开销的系统进行了复现实验以及模拟实验，虽然没有得到论文中那样出色的结果，但经过分析后不难发现，Zeus需要对同一训练任务重复多次才能确定好优化方案，这对于实验室环境下的训练来说是很奢侈的，因为这种环境下人们往往追求的是更低的训练时间，每次训练规模不会太大，并且会有多种多样的训练需求。这些特点都与Zeus的设计相悖，节省的能量与时间相比重复大量实验的过程中消耗的能量与时间来说微乎其微，之后换训练任务后又需要重复这个过程，浪费实在太太。实际上Zeus的设计更多应该是考虑了工业界的需求，工业界的训练任务往往比较单一，并且一次训练规模足够大，寻找最优方案时消耗的能量相比长时间大规模的训练过程中节省的能量来说反而微乎其微，在论文中作者也提到Zeus阿里巴巴的生产环境中部署后降低能耗的效果非常显著。

此外，Zeus还有很多在实验室环境下使用的缺陷，比如修改gpu参数需要root权限，然而市面上出租的GPU服务器一般不提供Root权限，并且即使使用了sudo权限，也可能面临sudo对应python环境不正确的问题等等，我们在复现过程中遇到了很多诸如此类的环境问题。

总的来说，虽然我们可能没法在写论文跑实验的时候很好的利用Zeus来降低能耗或者时间，不过也不能因此就去质疑Zeus的能力，毕竟它并不是一个为实验室环境准备的解决方案。