兰州大学 ASC-2023 预选招募题

基础条件

具有基本的Linux操作运维能力，有C语言编程或深度学习基础（若对超算知识有一定了解优先考虑），有较好的自我学习能力、英语阅读能力和团队合作能力，对计算机体系结构、高性能计算有浓厚兴趣的同学均可以报名。

赛题选择

本次预选招募会择优选择人工智能方向和科学计算方向的同学，若选择人工智能方向只需做人工智能相关部分的题目，科学计算方向同理。

1. 人工智能部分纳新（CV或NLP部分选择一个做即可）

1.1 CV

基础赛题（30）：单图超分辨率挑战

问题简述

单图超分辨率(SR)是近二十年来一个非常有吸引力的研究课题，从低分辨率(LR)图像中提取高分辨率(HR)图像是一项极具挑战性的任务。从卫星和航空成像到医学图像处理、人脸图像分析、文本图像分析、符号和数字车牌读取、生物特征识别，它在许多现实世界的问题中都有实际应用。

目前，随着深度学习的复兴，利用深度卷积神经网络(CNN)来执行超分辨率的研究前景十分广阔。超分辨率的最终目标之一是产生高视觉质量的输出。然而，目前针对超分辨率重建方法一部分集中在最小化均方误差和峰值信噪比(PSNR)高,但他们往往缺乏高频细节和感知不满意,他们无法匹配富达预期更高的分辨率。如何才能克服这一缺点，得到更优的方法解决这一问题呢？

注：在SR中引入了生成对抗性网络(GAN)，以鼓励网络偏爱看起来更像自然图像的解决方案。

评价指标

SR常用的评价指标有两种，一种是**PSNR（峰值信噪比）**，另一种是**SSIM（结构相似性评价）**，这两种评价指标是SR中最基础的测量被压缩的重构图像质量的指标。

(1) PSNR是信号的最大功率和信号噪声功率之比，来测量已经被压缩的重构图像的质量，通常以分贝(dB)来表示。PSNR越高说明图像质量越好。

$$PSNR = 10 * log_{10}(\frac{MAX_I^2}{MSE}) = 20 * log_{10}(\frac{MAX_I}{\sqrt{MSE}}) \tag{1}$$

其中，MAX_I 表示图像中像素值中的最大值，MSE 表示的两个图像之间对应像素之间差值的平方的均值，单通道图像的 MSE 可以表示为：

$$MSE = \frac{1}{M * N}\sum_{i=1}^{N}\sum_{j=1}^{M}(f_{ij} - f_{ij}^{'})^2 \tag{2}$$

多通道图像的 MSE 可以表示为：

$$MSE = \frac{1}{C * M * N}\sum_{k=1}^{C}\sum_{i=1}^{N}\sum_{j=1}^{M}(f_{ijk} - f_{ijk}^{'})^2 \tag{3}$$

（2） SSIM 是衡量两幅图像相似度的指标，其取值范围为[0,1]，**SSIM 的值越大，表示图像失真程度越小，说明图像质量越好。**

$$SSIM(X,Y) = L(X,Y) * C(X,Y) * S(X,Y)$$
$$L(X,Y) = \frac{2\mu_X\mu_Y + C_1}{\mu_X^2 + \mu_Y^2 + C_1}$$
$$C(X,Y) = \frac{2\sigma_X\sigma_Y + C_2}{\sigma_X^2 + \sigma_Y^2 + C_2}$$

$$S(X,Y)=\frac{\sigma_{XY}+C_3}{\sigma_X\sigma_Y+C_3} \tag{4}$$

其中， μ_X 和 μ_Y 为图像 X，图像 Y 的像素均值， σ_X 和 σ_Y 为图像 X，图像 Y 的像素的标准值， σ_{XY} 表示图像 X 和 Y 的协方差。此外，C_1，C_2，C_3是常数，$C_1=(K_1*L)^2$，$C_2=(K_2*L)^2$，$C_3=\frac{1}{2}C_3$，一般的，$K_1=0.01$，$K_2=0.03$，$L=255$。

基准模型

本文提供一基准模型：SRGAN（不限于此模型）。该方法将 GAN 应用到了 SR 问题领域，SRGAN 基于 ResNet 和 GAN， 是首个可以推演出原图像四倍分辨率而充分还原自然细节的框架。

参考模型代码和数据集：

```
# 模型
SRGAN.tar
# 数据集
Urban100.tar
```

注：可以参考2020cvpr有关超分辨率的文章。https://blog.csdn.net/moxibingdao/article/details/106726667（CSDN博客）

完成后也可尝试下列加速方法（加分项）：

1. DataParallel
2. DistributedDataParallel
3. Apex混合精度
4. 分布式训练工具horovod

文档要求

latex编写：

1. 模型简介
2. 源码及介绍
3. 训练结果

高阶赛题（100）：阿里巴巴优酷视频增强和超分辨率挑战赛

简介

视频增强和超分是计算机视觉领域的核心算法之一，目的是恢复降质视频的内容，提高视频的清晰度。该技术在工业界有着重要的实用意义，例如早期胶片视频的质量和清晰度的提升。优酷将为学术界推出了业界最大、最具广泛性的数据集。该数据集的生成模型完全是模拟实际业务中的噪声模式，研究人员可以真正的在实际场景中打磨算法，推动视频增强和超分算法在实际问题中的应用，促进工业界和学术界的深度合作。

数据集描述

本赛题选区的数据集包括视频数据、评测程序和数据说明等三部分组成。高分辨率视频来自优酷高清媒资库，其数据集包括10,000+样本，包括不同内容品类，不同业务场景下的噪声模型、不同难度等等。

数据每个样本由低分辨率视频和高分辨率视频组成的视频对构成。低分辨率视频为算法的输入，高分辨率视频为增强和超分后的真值。每个视频的时间长度为5秒左右。绝大部分高清数据的分辨率是 1080P ， 大概300M。由于是4倍超分，低质视频分辨率为 270P，大概19M。 少量高清数据的分辨率是2048×1152，低质视频分辨率为512×288。视频数据为无压缩的y4m格式。

视频命名规则：`Youku_视频序列号（%05d)_h/l_Sub抽帧频率(%2d)_GT/Res.y4m`

- 视频序列号（%05d):为5位数字的视频序列号
- h：表示高清视频
- l: 表示低分辨率视频
- GT:表示ground-truth视频
- Res:表示计算结果视频

例如，（`Youku_ 00101_l.y4m`，`Youku_ 00101_h_GT.y4m`）表示是第101个视频对，前者为低分辨路视频，后者为高清真值视频。算法恢复结果中，完整视频需要命名为`Youku_ 00100_h_Res.y4m`，抽帧视频为`Youku_00100_h_Sub25_Res.y4m`。`Sub25`表示时间上每25帧抽取结果。

本数据集推荐采用PSNR和VMAF两种评价指标，评测程序代码示例也将包含在数据集中。评估程序将计算PSNR和VMAF两种指标，均采用逐帧计算。评估程序最终VMAF结果为完整视频所有帧VMAF结果的平均值；最终PSNR的结果为完整视频和抽帧视频中所有帧的平均值。PSNR和VMAF得分进行加权得到得分。

score = PSNR指标得分×80% + VMAF指标得分×20%

- PSNR（Peak Signal To Noise Ratio）https://en.wikipedia.org/wiki/Peak_signal-to-noise_ratio
- VMAF（Video Multi-Method Assessment Fusion）https://github.com/Netflix/vmaf

基本要求

由于考虑到纳新成员资源配置问题，本次纳新将以youku_00000_00049_h_GT.zip,youku_00000_00049_l.zip二者作为数据集。

请自行设计模型，并编写文档。文档主要内容包括模型简介，训练结果，并在附录中包含主要代码。

模型训练速度也是本赛题着重考虑的方向之一，如果尝试使用分布式训练方法，例如数据并行，模型并行，或使用Apex混合精度抑或是或采用分布式训练工具等，即使最终模型训练得分不佳，我们也非常欣赏您的积极探索的精神，可破例获得加分。

1.2 NLP

基础题目（50）：我的中文语言“大”模型

简介

GPT-3 是一种用于自然语言处理(NLP)的具有 1750 亿个参数的大型语言模型(LM)，它引发了 AI 的全新趋势。GPT-3 可广泛用于多种 NLP 应用，例如阅读理解、问答、文本演绎等。此后，发布了大量大型LM 。

训练具有数十亿或数万亿参数的大型语言模型非常困难，因为不仅需要大量的计算资源，而且还需要复杂的训练方法来处理如此庞大的模型参数，这可能超出现代处理器的内存限制。所以应该使用一些特殊的训练方法，比如模型并行和管道并行。

Yuan1.0是最大的单例汉语模型之一。它在一个新的 5TB 高质量文本中文数据集上进行训练，该数据集是从互联网的 850TB 原始数据中提取的。元 1.0 的架构是通过将其内在模型结构与严重影响大规模分布式训练性能的关键因素相结合而设计的。Yuan1.0在一个有 2128 个 GPU 的集群上训练了大约 16 天。训练期间实际稳定表现达到了理论峰值的45 % 。Yuan1.0的源代码可以从https://github.com/Shawn-Inspur/Yuan-1.0。

模型要求

我们希望您可以自行配置好环境后，使用源代码中的`/Yuan-1.0/src/pretrain_yuan_13B.sh`脚本经过部分参数修改后成功训练出一个属于你的“AI大模型”。

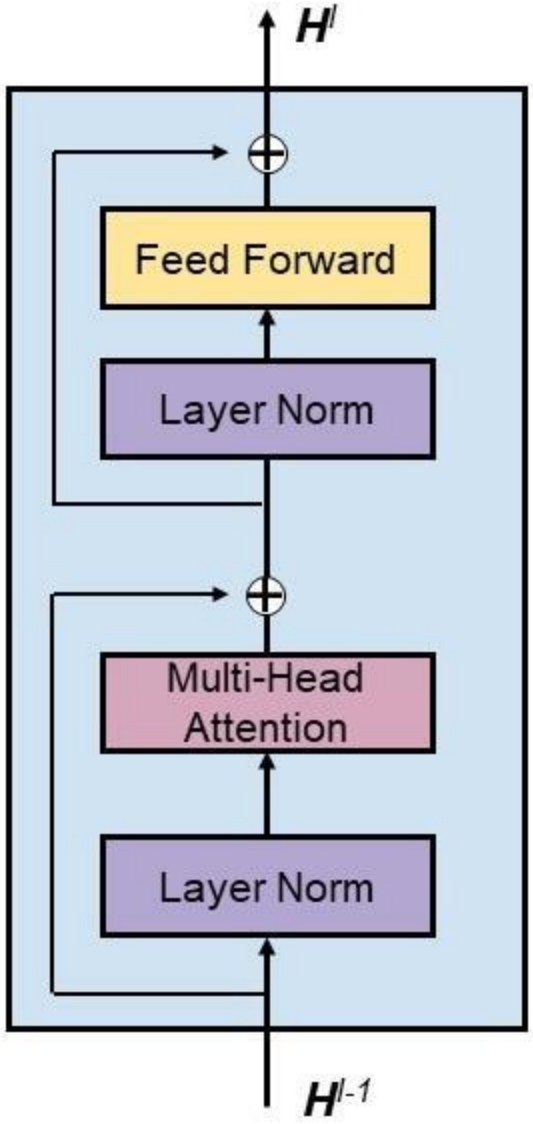
模型参数要求如下：

Model	hidden size(d)	attention-heads	Layers(L)	Parameters(Million)	Sequence length(n)	Train tokens (Million)
27M	384	20	4	27M	60	20

模型的loss计算公式如下所示，Loss便是loss计算函数, n是训练的sequence length, E 是word嵌入矩阵，以及$\boldsymbol{P}$是位置嵌入矩阵, L 是transformer blocks的数目，该值越低则表示模型精度越高。

$$
\begin{aligned}
\text{Loss } &= \sum_{k=1}^{n} -\log P\left(x_k \mid x_1, x_2, \ldots, x_{k-1}\right) \\
\boldsymbol{H}^0 &= \boldsymbol{E} + \boldsymbol{P} \\
\boldsymbol{H}^l &= \text{Transformer block}\left(\boldsymbol{H}^{l-1}\right), 1 \ll l \ll L \\
P\left(x_k \mid x_1, x_2, \ldots, x_{k-1}\right) &= \text{softmax}\left(W_v \boldsymbol{H}_k^L\right)\Big|_{x_k}
\end{aligned}
\tag{5}
$$

Transformer_block的架构如下图所示。



FFN 块中有两层，权重 $W_1 \in R^{d\times 4d}$ 和 $W_2 \in R^{4d\times d}$

$$\mathrm{FFN}(x) = \max\left(0, xW_1 + b_1\right) W_2 + b_2 \tag{6}$$

数据集

数据集如要使用下载链接如下

百度云盘: https://pan.baidu.com/s/1zQgKjf1tafzw__COaufyPA (extract code: 1jjz)

Microsoft OneDrive: https://1drv.ms/f/s!Ar_0HIDyftZTsFkbM8eQFtquk4ZH

数据集因为过于庞大，因此只需下载`001.txt`完成本次训练任务即可。

基本要求

结果精度要求：

由于各位可用的计算资源和时间有限，我们只要求参与者在最短的时间内完成 5百万个train-token的训练，并达到小于 7.0的loss值。如果最终的loss值大于7.0，说明模型训练过程不能收敛，提交的结果无效。

文档编写要求：

请使用`latex`编写整个文档。简单介绍您是如何进行环境配置的，过程中遇到的一些麻烦以及是如何解决的，尽可能详细记录整个过程。

其他要求：

除了需要提交文档外，我们还希望您可以将如下文件提交

文件夹名称	内容
MYLM	根目录
MYLM/Log	预训练日志文件
MYLM/tensorboard	tensorboard日志文件
MYLM/model	语言模型预训练脚本源代码

进阶题目（100）：训练中文语言大模型——“Yuan”

要求使用各种优化方式，使得模型在达到以上基本要求的条件上，需要更快的完成预训练任务。

可以参考的优化方式如下所示：

Deepspeed：

`Zero`:https://www.deepspeed.ai/tutorials/zero/

`spare attention`:https://www.deepspeed.ai/tutorials/sparse-attention/

除此外模型参数量也发生一些变化，具体变动如下所示：

Model	hidden size(d)	attention-heads	Layers(L)	Parameters(Billion)	Sequence length(n)	Train tokens (Billion)
4.7B	3072	20	40	4.7	2048	1

2. 科学计算题目：OpenLB

试题简述

1. 选定题目使用的编译运行环境，推荐使用实体机+Linux环境。
2. 对编译运行环境进行一次硬件+系统性能的简单评测，例如内存带宽测试Stream、OSU_Benchmark测试MPI通信效率、文件系统性能评测IOZone等。
3. 自行根据OpenLB官网资料，编译安装OpenLB软件，在版本olb-1.4r0或olb-1.5r0选择一个进行后续实验即可。
4. 选择测试基准（如以未开任何优化进行编译的OpenLB的要求测试程序的运行时间等为基准），对优化后的OpenLB软件的测试程序的性能进行评测。

编译安装要求

选择合理的策略优化OpenLB的编译安装，如采取多进程+多线程混合模式安装（MPI + OpenMP）、换用效率更高的编译器（Intel C/C++ Compiler，Clang/LLVM，PGI等），使用合理的手段链接上加速库（如Intel mkl数学库套件，FFTW，OpenBlas等）

若有相关的GPU计算资源，可以尝试olb-1.5r0版本开启GPU加速进行分析。

无论是否产生了优化，请对你的编译加速手段进行解释。

基准测试

以`examples`目录下的三个测试程序为基准进行分析比较：

- **multiComponent/contactAngle2d/contactAngle2d**
- **porousMedia/porousPoiseuille3d/porousPoiseuille3d**
- **laminar/bstep3d/bstep3d**

附加测试（加分项）

在完成基本的编译安装测试后，可尝试完成下列操作，无论是否成功，在文档中体现你的尝试与失败分析：

1. 使用 Intel Vtune 或 scalasca 或 perf 等性能分析工具进行分析，尝试寻找瓶颈。
2. 尝试使用编译器的编译优化参数进行加速优化（如函数内联、指令重排，Intel编译器的IPO优化，PGO优化等）
3. 若尝试了GPU加速的olb-1.5r0版本，使用 NVIDIA Nsight Systems 对加速后的 OpenLB 测试程序进行性能分析。
4. 可尝试使用加速库（如Intel数学库、OpenBlas矩阵运算库）的函数或其他手段进行源代码上的优化，无论是否产生优化，请写出你源代码优化的理由。
5. 对OpenLB软件整个的编译安装过程进行解释，例如整个框架如何编译出来，又如何作为库文件链接至各个测试文件并被使用。

产出要求

文档

文档使用LaTeX编写（建议直接搜索使用现成的模板，最好能是贴近论文的格式），关于LaTeX的一个比较好的简单教程

文档中要有：

1. 环境配置
2. 硬件+系统的简单评测
3. 基准测试+性能优化分析过程
4. 附加测试过程
5. 测试程序运行成功截图及结果、分析

安装脚本（可选）

在提交文档的时候可以另带 OpenLB 自动化编译安装的脚本文件（尽量满足开箱即用）。

截至时间12月4日，发送题解至邮箱supercomputing@lzu.edu.cn

如果完成人工智能或科学计算中的进阶题目，我们诚挚的邀请您参加我们之后的例会讨论，我们将准备精美礼品等您参加。