目 录

使用 FastDeploy 在英特尔 CPU 和独立显卡上端到端高效部署 AI 模型

作者：王一凡 英特尔物联网创新大使

1.1 产业实践中部署 AI 模型的痛点

1.1.1 部署 AI 模型的典型流程

对于来自于千行百业，打算将 AI 模型集成到自己的主线产品中，解决本行痛点的 AI 开发者来说，部署 AI 模型，或者说将 AI 模型集成到自己产品中去的典型步骤(以计算机视觉应用为例)有：

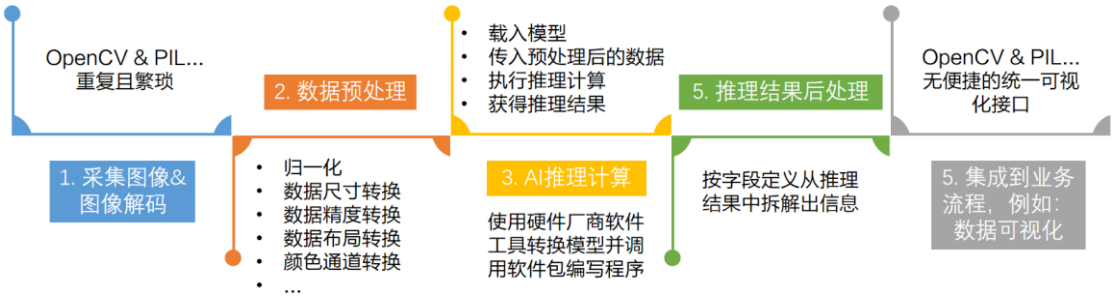
a. 采集图像&图像解码

b. 数据预处理

c. 执行 AI 推理计算

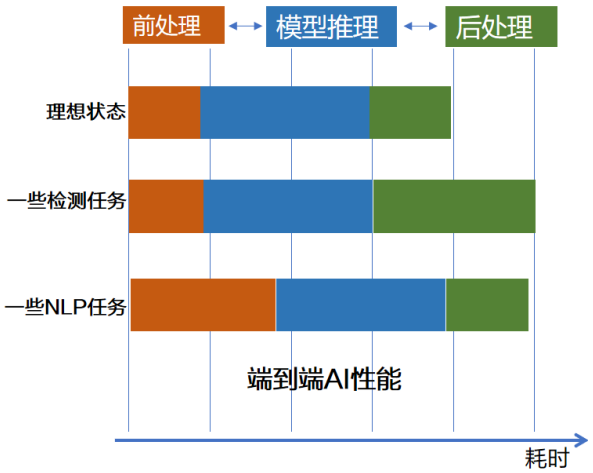
d. 推理结果后处理

e. 将后处理结果集成到业务流程



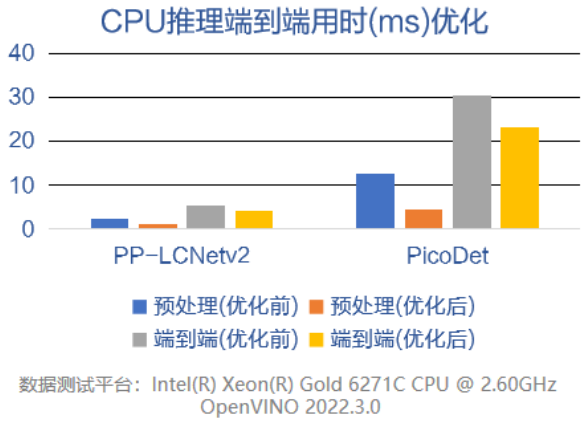
1.1.2 端到端的 AI 性能

当 AI 开发者将 AI 模型集成到业务流程后，不太关心 AI 模型在 AI 推理硬件上单纯的推理速度，而是关心包含图像解码、数据预处理和后处理的端到端的 AI 性能。



在产业实践中，我们发现不仅 AI 推理硬件和对应推理引擎(例如：OpenVINO Runtime)对于端到端的性能影响大，数据预处理和后处理代码是否高效对于端到端的性能影响也大。

以 CPU 上预处理操作融合优化为例，经过优化后的前处理代码，可以使得 AI 端到端性能得到较大提升。



数据来源：感谢 FastDeploy 团队完成测试并提供数据

结论：优秀且高效的前后处理代码，可以明显提高端到端的 AI 性能！

1.1.3 部署 AI 模型的难点和痛点

在产业实践中，在某个任务上当前最优的 SOTA 模型的很有可能与部署相关的文档和范例代码不完整，AI 开发者需要通过阅读 SOTA 模型源代码来手动编写模型的前后处理代码，这导致：

- 耗时耗力：阅读 SOTA 模型源代码来理解模型的前后处理，提高了部署模型的技术门槛。另外，手动编写前后处理代码，也需要更多的测试工作来消除 bug。
- 精度隐患：手动或借助网上开源但未经过实践验证过的前后处理代码，会有精度隐患，即当前对于某些图片精度很好，但对于另外的图片精度就下降。笔者就遇到过类似问题，原因在于调用了一个 GitHub 上下载的 NMS()函数，这个函数对代码仓提供的范例模型有效，但对于笔者使用的模型恰恰就出现丢失检测对象的问题。
- 优化困难：解决了精度问题后，下一步就是通过多线程、模型压缩、Batch 优化等软件技术进一步提升端到端的 AI 性能，节约硬件采购成本。这些软件技术对于计算机专业的工程师不算挑战，但对于千行百业中非计算机专业的工程师，却无形中建立起了一道极高的门槛。

为了赋能千行百业的工程师，高效便捷的将 AI 模型集成到自己的产品中去，急需一个专门面向 AI 模型部署的软件工具。

1.2 FastDeploy 简介

FastDeploy 是一款**全场景、易用灵活、极致高效**的 AI 推理部署工具。提供**开箱即用**的**云边端**部署体验，支持超过 150+ Text, Vision, Speech 和跨模态模型，并实现**端到端**的推理性能优化。包括图像分类、物体检测、图像分割、人脸检测、人脸识别、关键点检测、抠图、OCR、NLP、TTS 等任务，满足开发者**多场景、多硬件、多平台**的产业部署需求。

FastDeploy 项目链接: https://github.com/PaddlePaddle/FastDeploy

1.3 英特尔独立显卡简介

英特尔在 2021 年的构架日上发布了独立显卡产品路线图，OpenVINO 从 2022.2 版本开始支持 AI 模型在英特尔独立显卡上做 AI 推理计算。

当前已经可以购买的消费类独立显卡是英特尔锐炫 TM 独立显卡 A7 系列，并已发布在独立显卡上做 AI 推理计算的范例程序。

1.4 使用 FastDeploy 在英特尔 CPU 和独立显卡上部署模型的步骤

1.4.1 搭建 FastDeploy 开发环境

当前 FastDeploy 最新的 Release 版本是 1.0.1，一行命令即可完成 FastDeploy 的安装：

```
pip install fastdeploy-python -f https://www.paddlepaddle.org.cn/whl/fastdeploy.html
```

1.4.2 下载模型和测试图片

FastDeploy 支持的 PaddleSeg 预训练模型下载地址：https://github.com/PaddlePaddle/FastDeploy/tree/develop/examples/vision/segmentation/paddleseg

测试图片下载地址：https://paddleseg.bj.bcebos.com/dygraph/demo/cityscapes_demo.png

使用命令，下载模型和测试图片

```
图片：

wget https://paddleseg.bj.bcebos.com/dygraph/demo/cityscapes_demo.png

模型：

https://github.com/PaddlePaddle/FastDeploy/tree/develop/examples/vision/segmentation/paddleseg
```

1.4.3 三行代码完成在英特尔 CPU 上的模型部署

基于 FastDeploy，只需三行代码即可完成在英特尔 CPU 上的模型部署，并获得经过后处理的推理结果。

```
import fastdeploy as fd

import cv2

# 读取图片

im = cv2.imread("cityscapes_demo.png")

# 加载飞桨 PaddleSeg 模型

model = fd.vision.segmentation.PaddleSegModel(“model.pdmodel”,

“model.pdiparams”,“deploy.yaml”)

# 预测结果

result = model.predict(im)

print(result)
```

将推理结果 print 出来，如下图所示，经过 FastDeploy 完成的 AI 推理计算，拿到的是经过后处理的结果，可以直接将该结果传给业务处理流程。

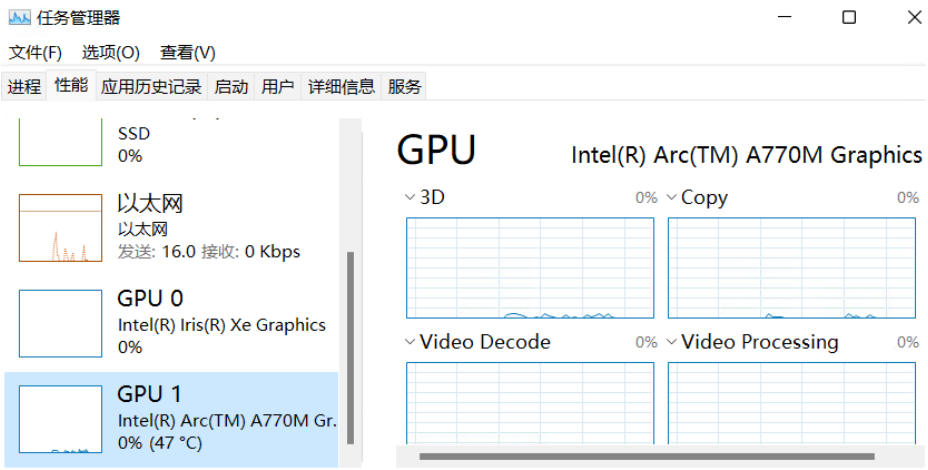
```
SegmentationResult Image masks 10 rows x 10 cols:
[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, .....]
[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, .....]
[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, .....]
[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, .....]
[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, .....]
[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, .....]
[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, .....]
[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, .....]
[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, .....]
[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, .....]
...........
result shape is: [1024 2048]
```

1.4.4 使用 RuntimeOption 将 AI 推理硬件切换英特尔独立显卡

在上述三行代码的基础上，只需要使用 RuntimeOption 将 AI 推理硬件切换为英特尔独立显卡，完成代码如下所示：

```
import fastdeploy as fd

import cv2

# 读取图片

im = cv2.imread("cityscapes_demo.png")

h, w, c = im.shape

# 通过 RuntimeOption 配置后端

option = fd.RuntimeOption()

option.use_openvino_backend()

option.set_openvino_device("GPU.1")

# 固定模型的输入形状

option.set_openvino_shape_info({"x": [1,c,h,w]})

# 加载飞桨 PaddleSeg 模型

model = fd.vision.segmentation.PaddleSegModel(“model.pdmodel”,
“model.pdiparams”,“deploy.yaml”,
                                              runtime_option=option)


# 预测结果

result = model.predict(im)
```

set_openvino_device()中字符串填写“GPU.1”是根据英特尔独立显卡在操作系统的中设备名称，如下图所示：



当前，在英特尔独立显卡上做 AI 推理，需要注意的问题有：

- 需要固定模型输入节点的形状(Shape)
- 英特尔 GPU 上支持的算子数量与 CPU 并不一致，在部署 PPYOLE 时，如若全采用 GPU 执行，会出现如下提示

```
RuntimeError: Operation: multiclass_nms3_0.tmp_1 of type MulticlassNms(op::v0) is not supported
```

这是需要将推理硬件设置为异构方式

```
option.set_openvino_device("HETERO:GPU.1,CPU")
```

到此，使用 FastDeploy 在英特尔 CPU 和独立显卡上部署 AI 模型的工作全部完成。

1.5 总结

面对千行百业中部署 AI 模型的挑战，FastDeploy 工具很好的保证了部署 AI 模型的精度，以及端到端 AI 性能问题，也提高了部署端工作的效率。通过 RuntimeOption，将 FastDeploy 的推理后端设置为 OpenVINO，可以非常便捷将 AI 模型部署在英特尔 CPU、集成显卡和独立显卡上。