TEE

Simple AI (SAI)

用户手册

2018.12.10

V1.0

本文档内容来自于北京梯易易科技有限公司（TEE），基于本文档内容，可用于评估本公司产品的性能，本文档包括环境创建，基于 Pytorch 工具的量化模型训练，模型转换，以及如何快速部署在 Windows, Linux 等平台。

1 SAI 说明

SAI 是基于 PyTorch 的卷积神经网络模型训练，转换，部署工具——可用于将 float 型卷积神经网络模型转换为定点量化模型（1bit 或者 3bit），并可通过 TEE 公司的算力棒来运行，或者从头开始训练量化模型。部署的时候可使用主机的 CPU 和 TEE 公司的算力棒通过联合通信进行推断，支持 Windows，Linux 等主流平台。

基于 SAI 和本公司出品的算力棒，可以非常方便的训练一个精度损失较低的量化模型，并转换成可以在算力棒上运行的模型，基于转换好的模型，最后 SAI 还为开发者提供了快速的部署到 Windows, Linux 等平台的一键部署工具。

2 硬件与系统要求

SAI 运行环境对主机配置的相关要求如下：

- CPU >= Intel i5 (推荐 i7)
- 内存 >=8 GB

当前支持在如下系统上运行

- Windows 10
- Ubuntu LTS 16.04

3 软件环境依赖

启动 SAI 工具，需要先安装以下软件：

Python:

推荐直接安装 anaconda 集成 python 环境，python2.7 或者 python3.7 均可，可在 https://www.anaconda.com/download/ 上根据自己的系统选择下载 Windows 或者 Linux 的安装包进行安装。

安装完成后可以在控制台（Windows 下打开 Windows Command Prompt， Linux 下打开 Terminal）输入以下命令来确认 python 环境是否安装成功:

```
python
```

如果安装成功，则会显示以下信息：

```
Python 3.7.0 (default, Jun 28 2018, 08:04:48) [MSC v.1912 64 bit (AMD64)] ::
Anaconda, Inc. on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

Pytorch:

Pytorch 是 Facebook 开源的一款神经网络框架，可在官网：https://pytorch.org 自行选择和你的环境相符合的下载命令进行安装，安装完成后，可在控制台（Windows 下打开 Windows Command Prompt， Linux 下打开 Terminal）输入以下命令来确认 pytorch 环境是否安装成功:

```
import torch
import torchvision
```

CUDA(可选):

推荐使用 GPU 来跑训练，CPU 跑训练实在是太慢，你会无法忍受的，^o^.

若要使用 GPU 来训练，就需要安装 CUDA.一般来说，需要通过以下几步安装 CUDA：1. 安装 NVIDIA 显卡驱动，去官网(http://www.nvidia.cn/Download/index.aspx?lang=cn)查找适配自己电脑 GPU 的驱动；2. 安装 CUDA9.0, 去官网(https://developer.nvidia.com/cuda-90-download-archive)下载;3.安装 cuDNN，去官网(https://developer.nvidia.com/cudnn)下载

可通过以下命令来确认 GPU 是否成功安装:

```
import torch
torch.cuda.is_available()
```

4 安装算力棒驱动

将 SAI_v1.0.zip 解压，Windows 10 系统会自动安装驱动，Linux 系统需要一些额外的步骤，请参考以下命令为算力棒安装驱动：

```
sudo cp lib /libftd3xx.so.0.5.21 /usr/lib/
sudo cp lib /*.rules /etc/udev/rules.d/
```

5 训练数据准备

将你的训练数据分为 train 和 val 两个目录，基于标签数目 N，创建 0 -- N-1 个子目录，每个子目录中放入对应标签的图像数据。然后将 train 和 val 两个目录放置于 SAI_ROOT/data 目录下。

6 模型训练与转换

因为 TEE 算力棒仅支持 VGG 类型的卷积结构，所以 SAI 的模型训练工具也只提供了基于 VGG 类型的网络模型训练。当前版本支持三种类型的 VGG 网络：

- teeNet1:标准的 VGG16 网络，包括 13 个卷积结构和 3 个全连接层
- teeNet2:简化后的 VGG 网络，包括 18 个卷积结构，1 个 GAP 层，1 个全连接层
- teeNet3:去掉全连接层后的 VGG 网络，包括 16 个卷积结构

SAI 通过加载 training.json 文件来进行模型训练与转换，training.json 文件放置在 SAI_ROOT 目录下，可通过文本编辑器对其进行编辑修改。training.json 文件里的每个关键词描述如下：

- num_classes – 类别数目
- max_epoch – 最大迭代次数
- learning_rate – 学习率

- train_batch_size – 一次加载的训练数据数目
- test_batch_size – 一次加载的测试数据数目
- mask_bits – 每个主层的量化 bit 数
- act_bits – 每个主层的激活量化 bit 数
- resume – 接着之前中断的训练继续开始训练
- finetune – 加载一个预训练模型来微调
- full – 训练一个全精度的模型

在前面的工作都准备好后，你可以在命令行窗口输入以下命令来启动模型的训练与转换工作：

```
python TEE_SAI.py
```

运行结束后可在根目录下得到两个文件：conv.dat 和 fc.dat（如果是 teeNet3 网络，则只会得到 conv.dat 文件，因为该网络结构没有全连接层）。其中 conv.dat 是算力棒上加载运行的模型。

Tips: 关于模型训练，我们建议先使用 full 模式训练一个全精度的模型 F，再通过加载这个全精度模型 F 来 finetune 训练量化模型，得到最终的可部署模型。

7 推断部署

通过前面的模型训练与转换步骤，得到了可以在算力棒上运行部署的模型，接下来我们可以通过 SAI 的 infer 工具，结合分类任务，将该模型快速的部署到终端设备上。TEE_SAI SDK 目前支持 windows/linux/arm-linux 三个平台的推断部署，后续会增加 android/ios 等平台支持。

当前版本仅提供了针对 teeNet1 网络结构的分类任务推断部署。

下面我们详细介绍 3 种平台的推断的编译和部署。首先进入 SAI_ROOT/infer/目录，可根据实际需要部署的平台选择 Windows，Linux 或者 Arm-Linux 文件夹下的部署工具。

平台	依赖	描述
Windows	Opencv/openblas/ffmpeg	TEE 发布包中已经包含，无需编译

Linux	Opencv/ffmpeg	TEE 发布包中已经包含，需要时编译
Arm-linux	Opencv/ffmpeg/QT	TEE 发布包中已经包含，需要时编译

此处 ffmpeg 和 QT 依赖只是用于显示 demo 和界面。实际部署时可以根据使用场景选择是否去掉。

Tips：以 Windows 平台为例，请将前面转换好的 conv.dat 和 fc.dat 文件拷贝到 SAI_ROOT/infer/windows/bin/model 目录下，运行 run.bat 即可看到演示界面。如果需要修改类别数或者输出显示方式或者其他后处理，可以打开 SAI_ROOT/infer/windows/目录下的 TEE_SAI.sln 工程自行修改定制。

7.1 Windows 平台

文件	功能	描述
windows/TEE_SAI.sh	Windows 平台推断工程	需要 visual studio 2015 版本以上
windows/Lib	Windows 平台编译和运行需要的静态和动态库	
windows/bin	Windows 平台编译输出和运行目录	将 model 文件夹拷贝到此目录下，直接双击运行 run.bat

7.2 Linux 平台

文件	功能	描述
linux/CMakeLists.txt	Linux 平台编译文件	
linux/lib	Linux 平台编译和运行需要的静态和动态库	
linux/build	Linux 平台独立编译目录	1. cd build 2. cmake ..

		3. make
linux/bin	Linux 平台运行目录	将编译生成的可执行文件 TEEClassifierDemo 和 model 文件夹拷贝到此目录，运行 run.sh

7.3 arm linux 平台

文件	功能	描述
Arm64/CMakeLists.txt	Arm64 linux 平台编译文件	
Arm64/lib	Arm64 Linux 平台编译和运行需要的静态和动态库	
Arm64/build	Arm64 Linux 平台独立编译目录	1. cd build 2. cmake-gui .. 3. make
Arm64/bin	Arm64 Linux 平台运行目录	将编译生成的可执行文件 TEEClassifierDemo 和 model 文件夹拷贝到此目录，运行 run.sh

编译 aarch64 linux 时需要使用 linaro 的交叉编译工具，SAI_ROOT/arm64/toolchains/gcc-linaro-6.3.1-2017.05-x86_64_aarch64-linux-gnu.tar.xz 为交叉编译工具。