密级状态：绝密(　　)　　秘密(　　)　　内部(　　)　　公开(√)

RKNN Toolkit2 快速上手指南

（技术部，图形计算平台中心）

文件状态： [　] 正在修改 [√] 正式发布	当前版本：	V0.7.0
	作　　者：	HPC
	完成日期：	2021-3-30
	审　　核：	熊伟
	完成日期：	2021-3-30

瑞芯微电子股份有限公司

Rockchip Electronics Co., Ltd

更新记录

版本	修改人	修改日期	修改说明	核定人
V0.5.0	HPC	2020-12-18	初始版本	熊伟
V0.6.0	HPC	2021-2-24	版本更新	熊伟
V0.7.0	HPC	2021-3-30	版本更新	熊伟

目 录

1 主要功能说明

RKNN-Toolkit2 是为用户提供在 PC、 Rockchip NPU 平台上进行模型转换、推理和性能评估的开发套件，用户通过该工具提供的 Python 接口可以便捷地完成以下功能：

1) 模型转换：支持 Caffe、TensorFlow、TensorFlow Lite、ONNX、Darknet、Pytorch 模型转成 RKNN 模型，支持 RKNN 模型导入导出，后续能够在 Rockchip NPU 平台上加载使用。
2) 量化功能：支持将浮点模型转成量化模型，目前支持的量化方法有非对称量化（asymmetric_quantized-8、asymmetric_quantized-16），并支持混合量化功能。

   asymmetric_quantized-16 和混合量化暂不支持。
3) 模型推理：能够在 PC 上模拟 Rockchip NPU 运行 RKNN 模型并获取推理结果；也可以将 RKNN 模型分发到指定的 NPU 设备上进行推理。
4) 性能评估：可以将 RKNN 模型分发到指定 NPU 设备上运行，以评估模型在实际设备上运行时的性能。目前暂不支持。
5) 内存评估：评估模型运行时对系统和 NPU 内存的消耗情况。使用该功能时，必须将 RKNN 模型分发到 NPU 设备中运行，并调用相关接口获取内存使用信息。目前暂不支持。
6) 量化精度分析功能：该功能将给出模型量化前后每一层推理结果的余弦距离，以分析量化误差是如何出现的，为提高量化模型的精度提供思路。

2 系统依赖说明

本开发套件支持运行于 Ubuntu （Windows、MacOS、Debian 暂不支持）等操作系统。需要满足以下运行环境要求：

表 1 运行环境

操作系统版本	Ubuntu18.04（x64）及以上
Python 版本	3.6
Python 库依赖	numpy==1.16.6 onnx==1.7.0 onnxoptimizer==0.1.0

	onnxruntime==1.7.0 tensorflow==1.14.0 tensorboard==1.14.0 protobuf==3.12.0 torch==1.6.0 torchvision==0.7.0 mxnet==1.7.0 psutil==5.6.2 ruamel.yaml==0.15.81 scipy==1.2.1 tqdm==4.27.0 requests==2.21.0 tflite==2.3.0 opencv-python==4.4.0.46 PuLP==2.4

3 Ubuntu 平台快速上手

本章节以 Ubuntu 18.04、Python3.6 为例说明如何快速上手使用 RKNN-Toolkit2。

3.1 环境准备

- 一台安装有 ubuntu18.04 操作系统的 x86_64 位计算机。
- RK356x EVB 板。
- 将 EVB 板通过 USB 连接到 PC 上，使用 adb devices 命令查看，结果如下：

```
rk@rk:~$ adb devices
List of devices attached
515e9b401c060c0b        device
c3d9b8674f4b94f6        device
```

其中标红的为设备 ID。

3.2 安装 RKNN-Toolkit（以 Python3.6 为例）

1. 安装 Python3.6 和 pip3

```
sudo apt-get install python3 python3-dev python3-pip
```

2. 安装相关依赖

```
sudo apt-get install libxslt1-dev zlib1g zlib1g-dev libglib2.0-0 libsm6 \
libgl1-mesa-glx libprotobuf-dev gcc
```

3. 获取 RKNN-Toolkit2 安装包，然后执行以下步骤：

   a) 进入 package 目录：

```
cd package/
```

   b) 安装 Python 依赖

```
pip3 install -r doc/requirements.txt
```

c) 安装 RKNN-Toolkit2

```
sudo pip3 install rknn_toolkit2*.whl
```

d) 检查 RKNN-Toolkit 是否安装成功

```
rk@rk:~/rknn-toolkit2-v0.7.0/package$ python3
>>> from rknn.api import RKNN
>>>
```

如果导入 RKNN 模块没有失败，说明安装成功。

3.3 运行安装包中附带的示例

3.3.1 在 PC 上仿真运行示例

RKNN-Toolkit2 自带了一个模拟器，可以用来仿真模型在 npu 上运行时的行为。

这里以 mobilenet_v1 为例。示例中的 mobilenet_v1 是一个 Tensorflow Lite 模型，用于图片分类，它是在模拟器上运行的。

运行该示例的步骤如下：

1. 进入 examples/tflite/mobilenet_v1 目录

```
rk@rk:~/rknn-toolkit2-v0.7.0/package$ cd ../examples/tflite/mobilenet_v1
rk@rk:~/rknn-toolkit2-v0.7.0/examples/tflite/mobilenet_v1$
```

2. 执行 test.py 脚本

```
rk@rk:~/rknn-toolkit2-v0.7.0/examples/tflite/mobilenet_v1$ python3 test.py
```

3. 脚本执行完后得到如下结果：

```
--> config model
done
--> Loading model
INFO: Initialized TensorFlow Lite runtime.
done
--> Building model
```

```
    Analysing : 100%|████████████████████████████████████████████████
████████████████████████████████| 58/58 [00:00<00:00, 292.13it/s]
    Quantizating : 100%|████████████████████████████████████████████
████████████████████████████████| 58/58 [00:00<00:00, 1020.78it/s]
    I RKNN: set log level to 0
    done
    --> Export RKNN model
    done
    --> Init runtime environment
    done
    --> Running model
    W init_runtime: target is None, use simulator!
    mobilenet_v1
    -----TOP 5-----
    [156]: 0.8544921875
    [155]: 0.080322265625
    [205]: 0.0129241943359375
    [284]: 0.0084075927734375
    [194]: 0.0025787353515625

    done
```

这个例子涉及到的主要操作有：创建 RKNN 对象；模型配置；加载 TensorFlow Lite 模型；构建 RKNN 模型；导出 RKNN 模型；加载图片并推理，得到 TOP5 结果；释放 RKNN 对象。

examples 目录中的其他示例的执行方式与 mobilenet_v1 相同，这些模型主要用于分类、目标检测。

3.3.2 在 RK356x 上运行示例

该功能暂不支持。

4 参考文档

有关 RKNN-Toolkit 更详细的用法和接口说明，请参考《Rockchip_User_Guide_RKNN_Toolkit2_CN.pdf》手册。