

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:	当前版本:	V1.4.0
<input type="checkbox"/> 正在修改	作者:	HPC
<input checked="" type="checkbox"/> 正式发布	完成日期:	2022-8-20
	审核:	熊伟
	完成日期:	2022-8-20

瑞芯微电子股份有限公司

Rockchip Electronics Co., Ltd

(版本所有,翻版必究)

## 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
V1.0.0	HPC	2021-4-21	版本更新	熊伟
V1.1.0	HPC	2021-6-30	版本更新	熊伟
V1.2.0	HPC	2022-1-10	版本更新	熊伟
V1.2.5	HPC	2022-4-1	版本更新	熊伟
V1.3.0	HPC	2022-4-22	版本更新	熊伟
V1.4.0	HPC	2022-8-20	版本更新	熊伟

---

# 目 录

1. 主要功能说明.....	4
2. 系统依赖说明.....	5
3. 基于 Ubuntu 平台快速上手.....	6
3.1. 环境准备.....	6
3.2. 安装 RKNN-Toolkit2 (以 Python3.6 为例) .....	6
3.3. 运行安装包中附带的示例.....	7
3.3.1. 在 PC 仿真上运行示例.....	7
3.3.2. 在 RK3568 上运行示例.....	9
4. 附录.....	10
4.1. 参考文档.....	10
4.2. 问题反馈渠道.....	10

---

## 1. 主要功能说明

RKNN-Toolkit2 是为用户提供在 PC 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 等模型转换为 RKNN 模型，并支持 RKNN 模型导入导出，RKNN 模型能够在 Rockchip NPU 平台上加载使用。
- 2) 量化功能：支持将浮点模型量化为定点模型，目前支持的量化方法为非对称量化（asymmetric\_quantized-8），并支持混合量化功能。
- 3) 模型推理：能够在 PC（Linux x86 平台）上模拟 Rockchip NPU 运行 RKNN 模型并获取推理结果；或将 RKNN 模型分发到指定的 NPU 设备上运行推理并获取推理结果。
- 4) 性能和内存评估：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和内存占用情况。
- 5) 量化精度分析：该功能将给出模型量化前后每一层推理结果与浮点模型推理结果的余弦距离，以便于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。
- 6) 模型加密功能：使用指定的加密等级将 RKNN 模型整体加密。因为 RKNN 模型的加密是在 NPU 驱动中完成的，使用加密模型时，与普通 RKNN 模型一样加载即可，NPU 驱动会自动对其进行解密。

---

## 2. 系统依赖说明

本开发套件支持运行于 Ubuntu（目前版本 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 / 3.8
Python 库依赖	详见 doc/requirements*.txt

---

## 3. 基于 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-Toolkit2（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 libgl2.0-0 libsm6 \
libgl1-mesa-glx libprotobuf-dev gcc
```

3. 获取 RKNN-Toolkit2 安装包，然后执行以下步骤：

- a) 安装 Python 依赖：

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

- b) 进入 package 目录:

```
cd package/
```

- c) 安装 RKNN-Toolkit2

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

- d) 检查 RKNN-Toolkit2 是否安装成功

```
rk@rk:~/rknn-toolkit2/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 仿真上运行示例

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

2. 执行 test.py 脚本

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

3. 脚本执行完后得到如下类似结果：

```
--> Config model
done
--> Loading model
```





---

### 3.3.2. 在 RK3568 上运行示例

这里以 mobilenet\_v1 为例。工具包中带的 mobilenet\_v1 示例是在 PC 模拟器上运行的，如果在 RK3568 EVB 板上运行这个示例，可以参考以下步骤：

1. 参考 [https://github.com/rockchip-linux/rknpu2/blob/master/rknn\\_server\\_proxy.md](https://github.com/rockchip-linux/rknpu2/blob/master/rknn_server_proxy.md) 说明更新 EVB 板的 runtime 库和 rknn\_server 库，并确保 rknn\_server 服务已经启动（大部分平台需要手动通过串口启动）
2. 进入 examples/tflite/mobilenet\_v1 目录

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

3. 修改 test.py 脚本里的初始化环境变量时带的参数

```
rk@rk:~/rknn-toolkit2/examples/tflite/mobilenet_v1$ vim test.py
# 找到脚本里调用模型配置接口 config 的地方，如下
rknn.config(mean_values=[128, 128, 128], std_values=[128, 128, 128])
# 指定 RKNN 模型的目标使用平台，默认是 RK3566，这里指定为 RK3568
rknn.config(mean_values=[128, 128, 128], std_values=[128, 128, 128],
            target_platform='rk3568')
# 找到脚本里初始化环境变量的接口 init_runtime，如下
ret = rknn.init_runtime()
# 修改该接口的参数
ret = rknn.init_runtime(target='rk3568')
# 保存修改并退出
```

4. 执行 test.py 脚本，输出分类结果 TOP5:

```
mobilenet_v1
-----TOP 5-----
[156]: 0.93505859375
[155]: 0.057037353515625
[205]: 0.0038814544677734375
[284]: 0.0031185150146484375
[285]: 0.00017189979553222656
```

---

## 4. 附录

### 4.1. 参考文档

RKNN Toolkit2 使用指南：《Rockchip\_User\_Guide\_RKNN\_Toolkit2\_CN.pdf》

OP 支持列表：《RKNNToolKit2\_OP\_Support.md》

与 Toolkit1 的 API 差异：《RKNNToolKit2\_API\_Difference\_With\_Toolkit1.md》

以上文档均存放在发布包的 doc 目录中，也可以访问以下链接查阅：

<https://github.com/rockchip-linux/rknn-toolkit2/tree/master/doc>

### 4.2. 问题反馈渠道

请通过 RKNN QQ 交流群，瑞芯微 redmine 将问题反馈给 Rockchip NPU 团队。

RKNN QQ 交流群：1025468710

Rockchip Redmine: <https://redmine.rock-chips.com/>

注：Redmine 账号需要通过销售或业务人员开通。如果是第三方开发板，请先找第三方厂方反馈问题。