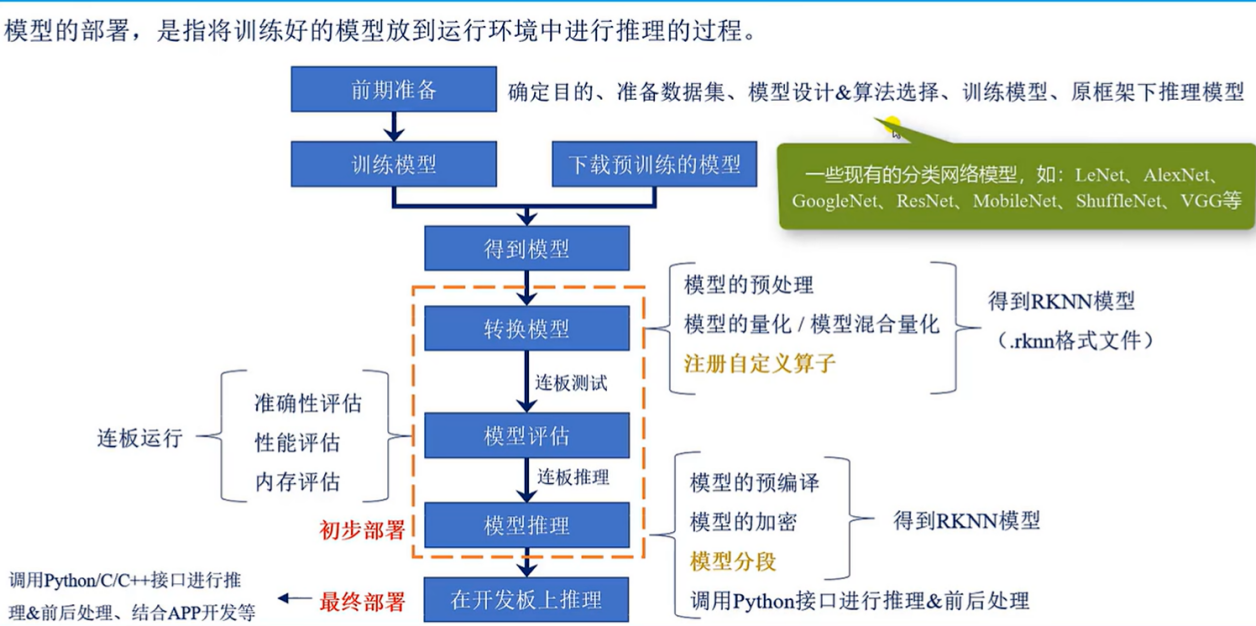
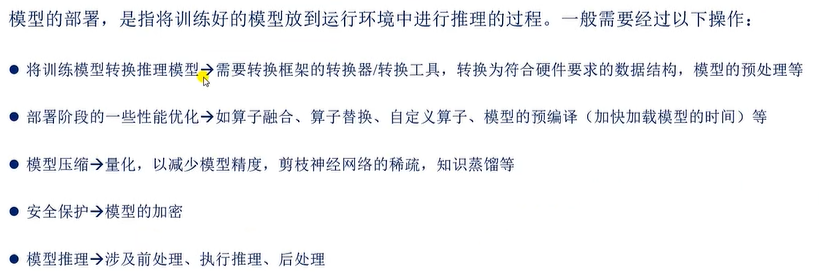
本节主要介绍在RKNPU上部署模型需要什么工具，以及大概流程是什么样的（也就是在NPU上完成模型部署需要的准备工具）：

（第一期有简单介绍过）





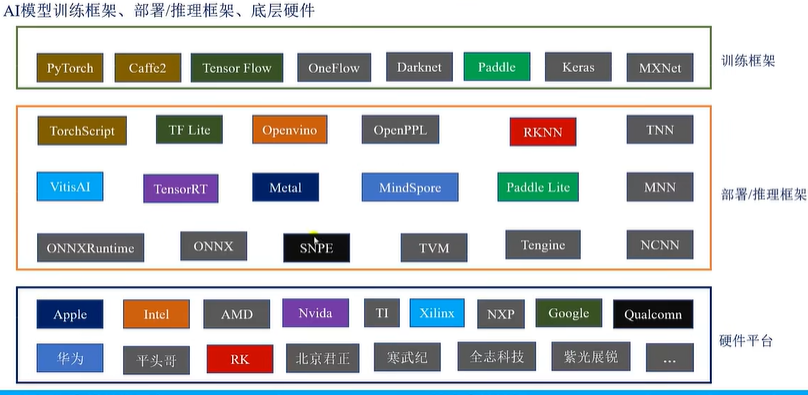
这就是AI模型在RK系列的芯片上部署的流程，

1. 首先要得到一个模型：一种是自己训练，另一种就是网上下载别人已经训练好的模型。
   1. 要训练一个模型，就要确定训练的目的，准备数据集、网络模型设计&算法选择、训练模型、原框架下推理模型。
      1. 如果是分类的网络模型，可以选择：一些现有的网络模型，如：LeNet、AlexNet、GoogleNet等
      2. 如果是目标检测的网络模型，可以选择：



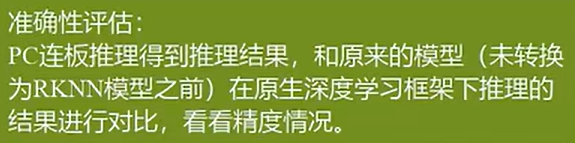
* 1. 训练模型就要进行多次调参和多次调整算法，其中可以根据一些参数指标来评估模型，如使用测试集进行验证，查看lose、准确率等

1. 得到模型后，就可以将这个模型，在原生的训练深度学习框架下进行模型的推理。也就是在原来的深度学习框架下跑模型。看看效果怎么样。如果效果可以，就可以把模型导出来使用。
2. 要AI模型部署落地，实际上就是进行黄框内的步骤：
   1. 进行模型的转换。将得到的模型（原生框架下导出的模型）进行转换为RKNN模型。之所以要转换，就是因为从这些深度学习模型框架下导出的模型不能在RKNPU上跑。为什么不能跑呢？
      1. 首先不同的处理器或硬件加速器，他们对输入模型的数据类型和数据的格式是有不同的要求的。所以需要对模型进行转换。把模型的参数和操作转换为目标硬件平台所支持的格式，这样才能被目标硬件平台所处理和理解。目标硬件平台才能对模型进行加速等运算。
      2. 不同的硬件设备的性能是不一样的，有的设备是资源有限的，如移动设备，他们是不能直接将深度学习框架下的模型放到这些设备内跑的，特别是大型模型，很难放到里面跑。所以需要对这些模型减值量化等操作。将这些模型压缩之后，再部署到这些资源受限的设备上。
      3. 随着深度学习应用技术的发展，模型的训练框架和模型的推理框架已经分化了，也就是说：有训练模型的深度学习框架，也有用于部署和推理的深度学习框架。训练的框架是有很多种的，当要将这些框架训练出来的模型部署到硬件设备上的时候，要部署的硬件设备也是各不相同的，如：硬件设备可能是GPU也可能是FPGA或NPU等，在这些设备里，各大厂商的硬件设备架构也是不一样的，如因特尔、华为、英伟达等。 也就是说没有哪一个框架是支持所有深度学习框架训练并且支持将这些模型部署到所有的硬件设备上。没有哪一个框架能够做到。所以就分为：训练框架和部署框架。

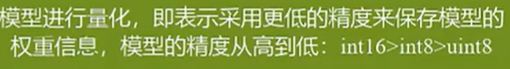


模型转换阶段主要就是进行：模型的预处理（在加载模型的时候，有些深度学习框架要明确模型的输入输出节点）、是否要对模型量化/混合量化等、自定义算子(可能模型里，也就是通过深度学习框架导出的模型 可能会包含RKNN所不支持的算子，就会导致模型转换失败，此时就需要自定义算子或对模型进行裁剪操作)，模型转化的目的就是得到RKNN模型。其它深度学习框架下的模型经过模型转换之后得到我们需要的RKNN模型。

* 1. 得到模型之后，就可以进行模型的评估：在PC端把RKNN模型发送到开发板端的RKNPU中去运行。然后看开发板端的运行效果。模型评估主要分为：
     1. 准确性评估：



将开发板运行的结果 与 原框架下运行的结果进行对比。如果精度不行，就看是否为预处理（例如输入的图片大小规格等不对）或量化（选择的量化的数据集不够有代表性，以及量化的方式导致量化的精度影响了结果，如：量化的时候，选择量化为int8或int16不同精度的类型的时候，模型推理的精度是有差异的）做得不好。



* + 1. 性能评估

看模型推理的时候，耗时。首先可以评估模型的整体耗时，若整体耗时长，可以评估模型每一层的耗时，看看是那一层的耗时比较严重，如：模型在初始化阶段耗时长，则可以考虑对导出的模型进行预编译。

* + 1. 内存评估：看RKNN运行时，内存的使用情况：内存分为：非NPU驱动分配的内存 和 NPU驱动在运行期间使用的内存。 二者相加就是总内存使用情况。
  1. 将模型初步部署到开发板，跑模型推理的效果。即：采用PC端连接开发板端来进行推理。PC端把RKNN模型发送到RKNPU端，让RKNPU去运行推理。

1. 如果RKNPU跑模型没有问题，就实现最终的部署，在开发板上推理模型。模型最终的推理效果，以在开发板上跑的为准。若开发板上跑的效果不理想，则可以反向看前面的步骤是否有问题。



本课程对于注册自定义算子和模型分段不支持。