

知

首发于
`from __future__ import *`

写文章

登录

scikit-learn 机器学习模型跨平台部署的思路



jj1118 · 5 个月前

(欢迎分享源网页，禁止未经本人书面许可转载至他处，或进行其他不当使用行为)

你是一名数据科学家/算法工程师，在 Python/R/Spark/MATLAB（业界有几个用？）等环境下用最爱的机器学习框架训练好了模型，准确度不错。想就这样交差？公司当初招聘你难道就为了看一个 98% 的数字，或者期待你建立新算法好发表论文？当然是要促进业务。于是问题来了，你并不能保证业务平台上能原封不动部署原本运行在个人电脑上的代码，简单的例子：

- 非 x86 平台对 Python 的 NumPy 库支持较为有限——移动平台最好不要指望直接运行或调用 Python。
- iOS 应用商店禁止模拟器类 app，包括 Python 解释器。
- 嵌入式市场有 MicroPython，但比起原生 C 资源消耗还是太大，且也有兼容性问题。

另一方面，我们一般不想让之前训练的模型直接报废，而在新平台上分别用其原生工具重新训练（况且还不是所有平台都能这样做——要是一块 Cortex-M 微处理器呢？）。所以，一次训练、跨平台部署是不少算法工程师必须要跨过的坎。

本文将试图简介在移动平台等处部署机器学习模型的方法，并阐述个人体验后所觉之利弊。目前只涉及线下学习线上预测的简单情况，暂不考虑线上学习。虽然讨论 scikit-learn 所生成模型的部署问题，但希望对其他库和其他环境也能有举一反三之效。

如果是 PC 或服务器等以 x86 架构为主的非移动平台，考虑到其本身对 NumPy 支持较好，可以采取直接调用 Python 代码的思路，典型的工具包括 Jython（jython.org/）和 Jep（github.com/mrj0/jep）等。

对于移动平台，由前所述直接调用 Python 代码不可行，只能考虑转制，有两种思路可供选择：

1. 转换成跨平台通用格式，典型如 PMML（dmg.org/pmml/v4-3/Gener...）。例如在 Java 平台上，Java PMML API（github.com/jpmml）提供了各类接口；在 Android 上由于平台对 XML 解析

的限制有人提供了 iomml-android（[aithub.com/iomml/iomml-...](https://github.com/iomml/iomml-...)）示例项目，其中包含一个

Maven 插件，将 PMML 文件序列化后编译为一个静态库。简单来说，将任何 scikit-learn 模型用 sklearn2pmml ([github.com/jpmml/sklearn...](https://github.com/jpmml/sklearn2pmml)) 导出为 PMML 文件后，放进 jpmml-android 项目的对应位置，编译后的 .pmml.ser 文件即可在任意 Android 项目中调用，方法可参照模板代码。至于 iOS 则未受支持。目前以上所提及的 PMML 相关项目均已支持该语言最新标准（4.3），如条件限制只能使用旧标准，则需要到 GitHub 相应仓库下载支持旧标准的最后版本。

2. 转换成原生代码，可以通过 sklearn-porter ([github.com/nok/sklearn-...](https://github.com/nok/sklearn-porter)) 等实现，尽管支持模型种类有限。将模型用 joblib dump 成 pkl 后（假设其名为 model.pkl，注意模型若为 Pipeline，则所有步骤均需得到 sklearn-porter 支持），命令行中运行

```
python -m sklearn_porter -i model.pkl -l <目标语言>
```

取决于模型大小，等候一段时间后即可见到原生语言的模型实现，文件名可能为 brain.c 或与之类似。自动生成的代码可能需要手动修改后才能编译运行。

以上两种思路各有利弊。通用格式只需处理一次模型（如包括 Android 则为两次），之后在各平台上分别编写代码以调用，支持算法较多，但并非所有平台都受到支持，且解析 PMML 格式效率不算高。原生代码则需要每种语言编译一次，自动化框架所支持算法较少（剩下的则需要设法导出参数后手动编写），但在各自业务平台上基本不存在兼容性问题，效率也较高。为数不多的例外，一是自动生成的代码对嵌入式系统仍不够最优，可能需要较多人工编辑或直接手动编写，当然这种情况下应该在建模时即优先考虑复杂度；二是 Java 单个方法字节码不能超过 64 KB，但机器学习模型稍大就会达到此限制，故 Java 上仍以 jpmml 为佳。

希望能对读者有所帮助。欢迎评论切磋。

「真诚赞赏，手留余香」

赞赏

还没有人赞赏，快来当第一个赞赏的人吧！

sklearn

跨平台

机器学习



还没有评论

写下你的评论...

文章被以下专栏收录



from __future__ import *

编程 · 科技 · 生活

[进入专栏](#)

推荐阅读



机器学习各语言领域工具库汇总-中文版

主要资源来自TensorFlow中文社区，翻译借助谷歌翻译，仅用于资源分享。相关链接：[机器学习分... 查看全文](#) >

十四君 · 6 个月前 · 编辑精选 · 发表于 UrIteam

机器学习算法线上部署方法

我们经常会碰到一个问题：用了复杂的GBDT或者xgboost大大提升了模型效果，可是在上线的时候又犯难了，工程师说这个模型太复杂了，我没法上线，满足不了工程的要求，你帮我转换成LR吧，直接... [查看全文](#) >

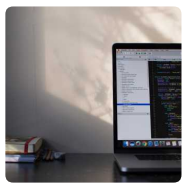
携程技术中心 · 10 个月前 · 编辑精选 · 发表于 携程技术中心



搭建python机器学习环境以及一个机器学习例子

这篇文章介绍了Python机器学习环境的搭建，我用的机器学习开源工具是scikit-learn。下面具体... [查看全文](#) >

[已重置] · 9 个月前 · 编辑精选 · 发表于 Python开发者社区



TensorFlow从单点到分布式编程实践

概述TensorFlow是Google开源的深度学习框架，起源于Google之前的深度学习框架 DistBelief，... [查看全文](#) >

青烟雨后 · 6 天前 · 编辑精选