

如何用iPad运行Python代码？



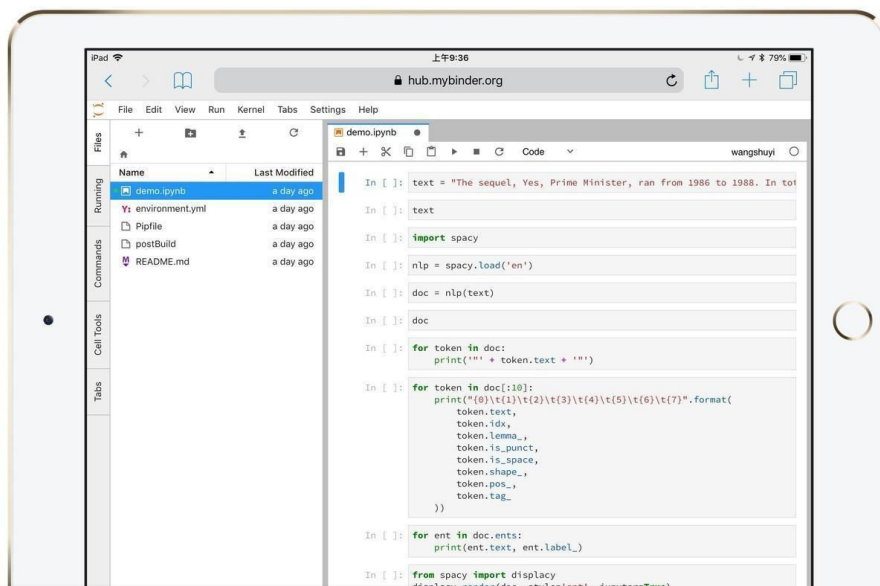
王树义

南开大学 情报学博士

关注他

编辑推荐

1,283 人赞同了该文章



赞同 1283

62 条评论

分享

喜欢

收藏

申请转载

知乎 @王树义



其实，不只是iPad，手机也可以。

痛点

我组织过几次线下编程工作坊，带着同学们用Python处理数据科学问题。

其中最让人头疼的，就是运行环境的安装。



知乎 @王树义

实事求是地讲，参加工作坊之前，我已经做了认真准备。

例如集成环境，选用了对用户很友好的Anaconda。

代码在我的Macbook电脑上跑，没有问题。还拿到学生的Windows 7上跑，也没有问题。这才上传到了Github。

在发布的[教程文章](#)里，我也已经把安装软件包的说明写得非常详细。



知乎 @王树义

赞同 1283

62 条评论

分享

喜欢

收藏

申请转载



还针对 Anaconda 这一 Python 运行环境的安装和运行，专门录制了视频。



但是，工作坊现场遇见的问题，依然五花八门。

有的是操作系统。例如你可能用Windows 10。实话实说，我确实没用过。拿着Surface端详，连安装后的Anaconda文件夹都找不到在哪儿。

有的是编码。不同操作系统，有的默认中文编码是UTF-8，有的是GBK。同样一段中文文本，我这里显示一切正常，你那里就是乱码。

有的是套件路径。来参加工作坊前，你可能看过我一些教程，并安装了 Python 2.7 版本 Anaconda。来到现场，一看需要 Python 3.6 版本，你就又安装了一份新的。结果执行起来，你根本分不清运行的 Python，pip 命令来自哪一个套件，更搞不清楚软件包究竟安装到哪里去了。再加上虚拟环境配置，你就要抓狂了。

还有的，甚至是网络拥塞问题。因为有时需要现场安装调用体积庞大的软件包，几十台电脑“预备——齐”一起争抢有限的Wifi带宽，后果可想而知。

痛定思痛，我决定改变一下现状。

目前的教程只提供基础源代码。对于许多新手同学来说，是不够的。

许多同学，就倒在了安装依赖软件包的路上，继而干脆放弃了。

变通的办法有许多。例如干脆录制代码执行视频给你看。

但是正如我在《MOOC教学，什么最重要？》一文中说过的，学习过程里，**反馈**最重要。

你需要能运行代码，并且第一时间获得结果反馈。

在此基础上，你还得能修改代码，对比前后执行结果的差别。

我得给你提供一个直接可以运行的环境。

零安装，自然也就没了上述烦恼。

这个事儿可能吗？

我研究了一下，没问题

只要你的设备上有个现代化浏览器（包括但不限于Google Chrome, Firefox, Safari和Microsoft Edge等）就行。



IE 8.0?

那个不行，赶紧升级吧！

读到这里，你应该想明白了。因为只挑浏览器，不挑操作系统，所以别说你用Windows 10，你就是用iPad，都能运行代码。

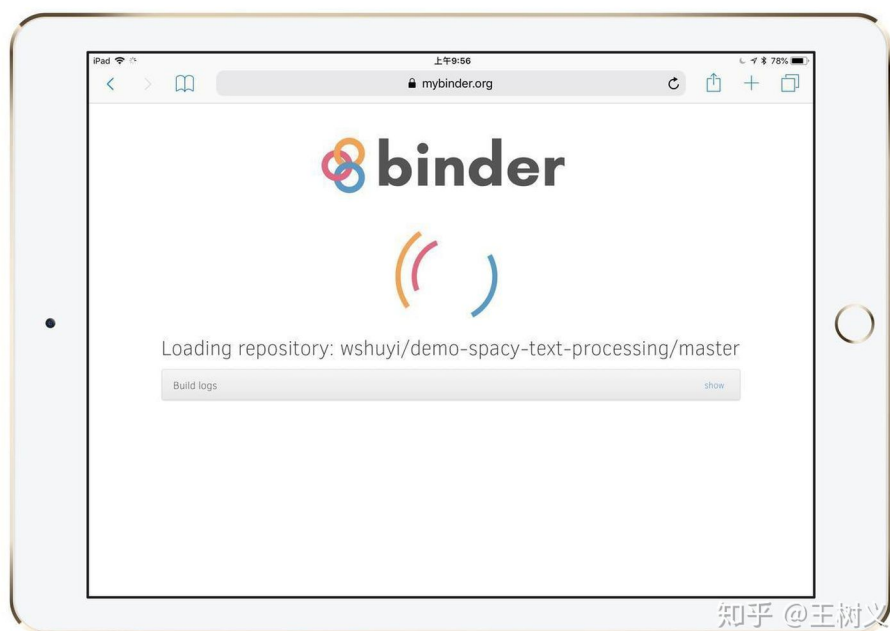
尝试

请你打开浏览器，输入[这个链接](https://t.cn/R35fElv)（t.cn/R35fElv）。

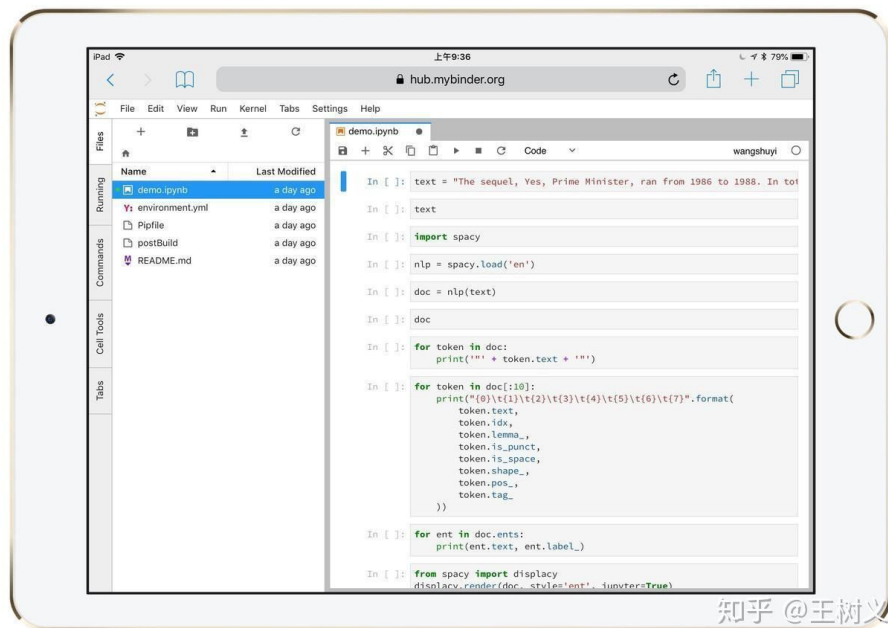
看看会发生什么？

我这里用iPad给你演示。

一开始会有个启动界面出来。请你稍等10几秒钟。



然后，你就能看到熟悉的Python代码运行界面了。



这个界面来自 Jupyter Lab。

你可以将它理解为 Jupyter Notebook 的**增强版**，它具备以下特征：

- 代码单元直接鼠标拖动；
- 一个浏览器标签，可打开多个Notebook，而且分别使用不同的Kernel；
- 提供实时渲染的Markdown编辑器；
- 完整的文件浏览器；
- CSV数据文件快速浏览
-

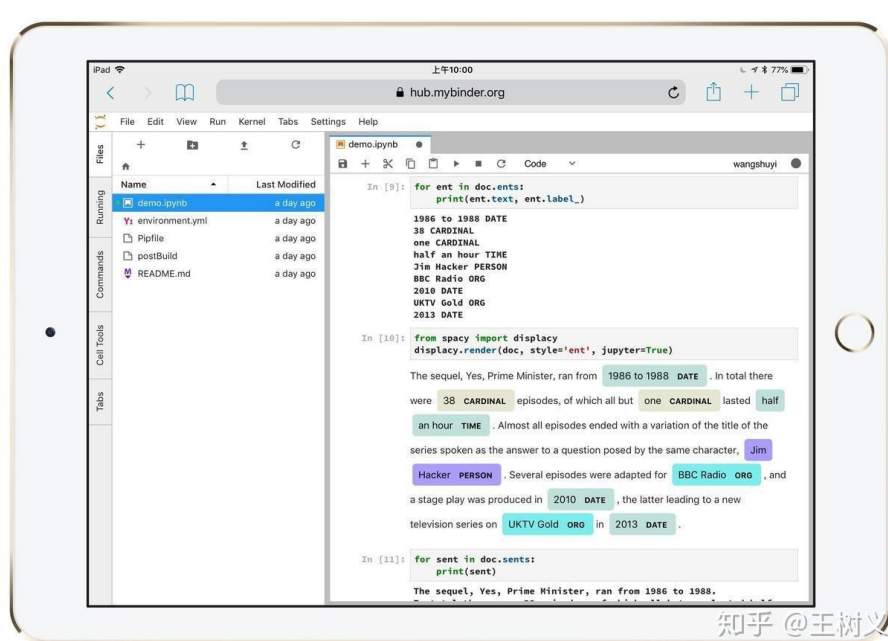
图中左侧分栏，是工作目录下的全部文件。

右侧打开的，是咱们要使用的ipynb文件。

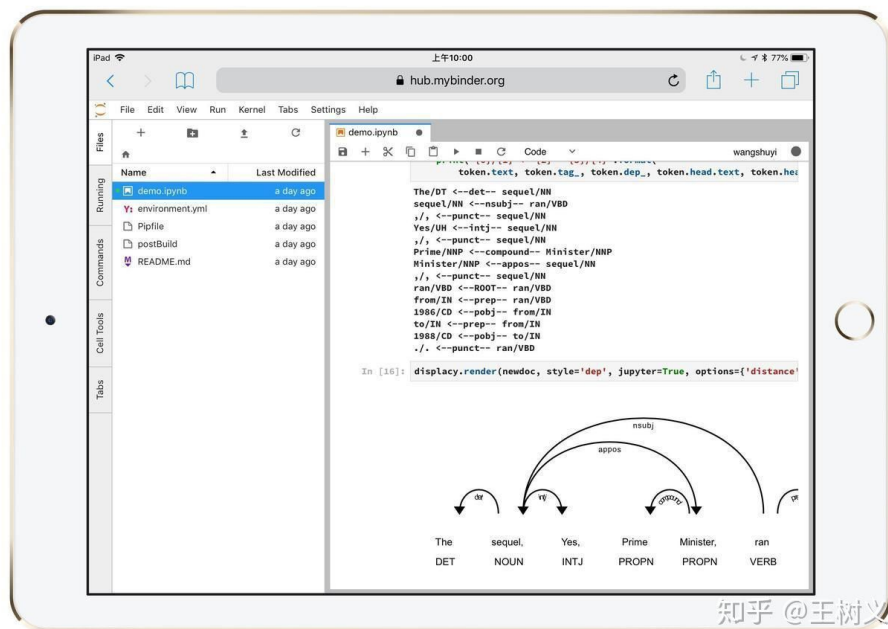
为了证明这不是逗你玩儿，请你点击右侧代码上方工具栏的运行按钮。

点击一下，就会运行出当前所在代码单元的结果。

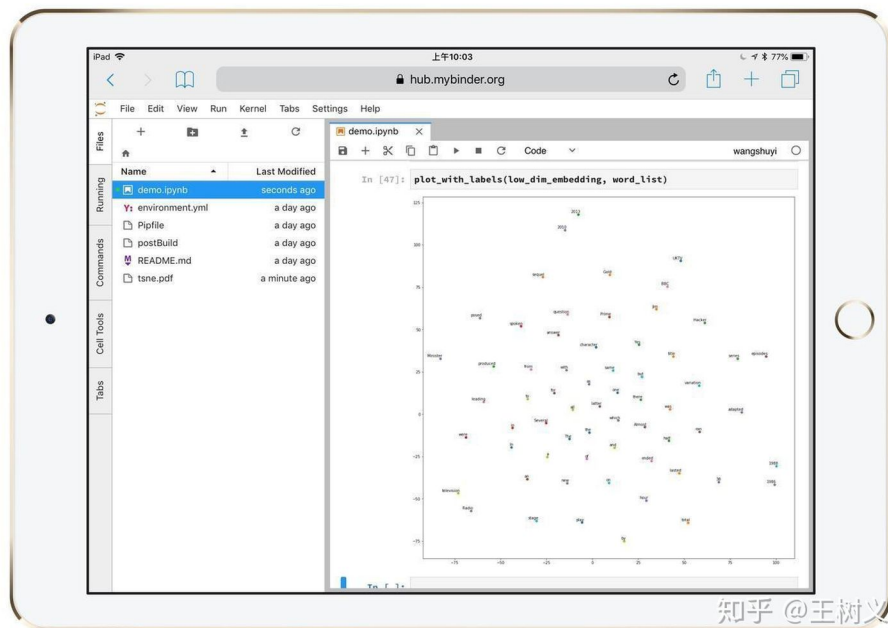
不断点击下来，你可以看见，结果都被正常渲染。



连图像也能正常显示。



甚至连下面这种需要一定运算量的可视化结果，都没问题。



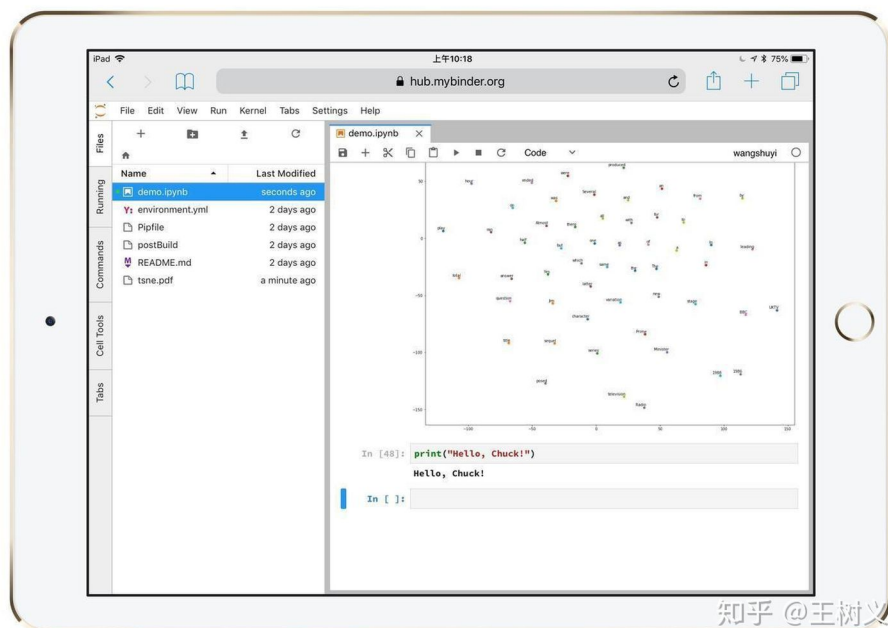
为了证明这不是变魔术，你可以在新的单元格，写一行输出语句。

就让Python输出你的名字吧。

假如你叫 Chuck，就这样写：

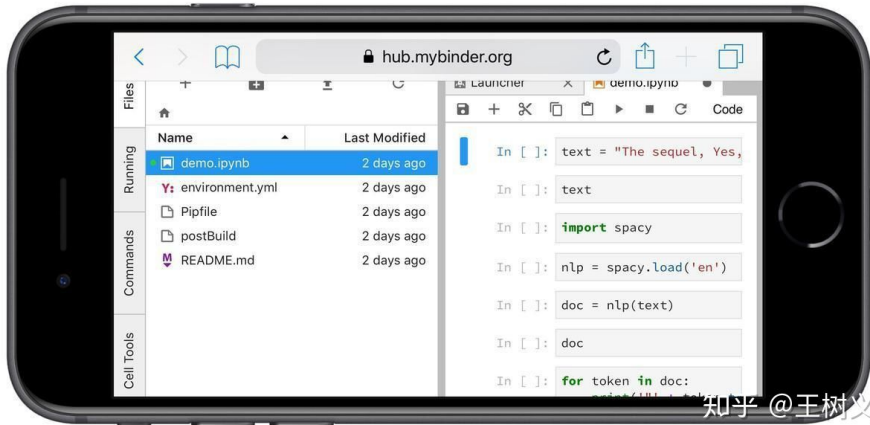
```
print("Hello, Chuck!")
```

把它替换成你自己的姓名，看看输出结果是否正确？



其实，又何止是iPad而已？

就像这样。



流程

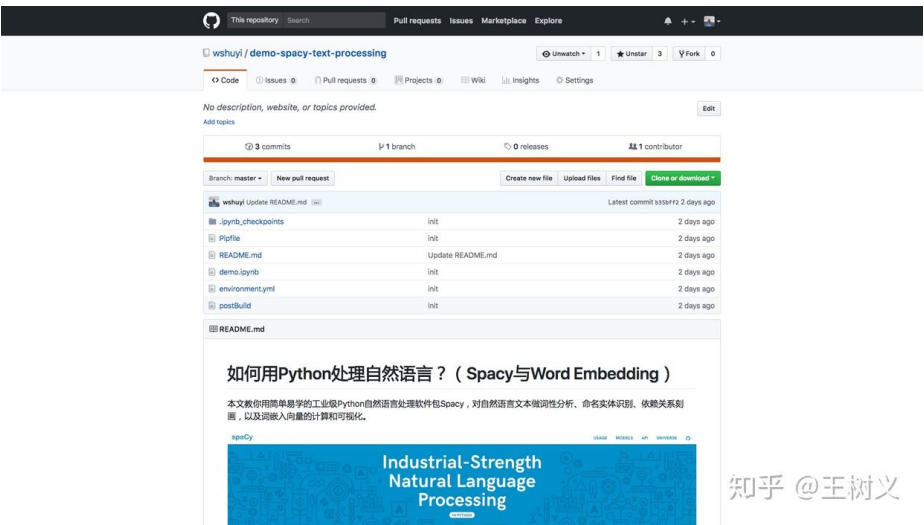
下面我给你讲讲，这种效果是怎么做出来的。

我们需要用到一款工具，叫做 mybinder 。它可以帮助我们，把 github 上的某个代码仓库（repo），快速转换成为一个可运行的环境。

注意 mybinder 为我们提供了云设施，也就是计算资源和存储资源。因此即便许许多多的用户同时在线使用同一份代码转换出来的环境，也不会互相冲突。

我们先来看看，怎么准备一个可供 mybinder 顺利转换的代码仓库。

我为你提供的样例在[这里](https://t.cn/R35MEqk)（t.cn/R35MEqk）：



顺便说一句，这个样例来自于我的数据科学系列教程之《[如何用Python处理自然语言？（Spacy与Word Embedding）](#)》。感兴趣的同学可以点击链接，查看原文。

在该 GitHub 页面展示的文件列表中，你需要注意以下3个文件：

- demo.ipynb
- environment.yml
- postBuild



其中 demo.ipynb 就是你在上一节看到的包含源代码的Jupyter Notebook文件。你需要首先在本地安装相关软件包，并且运行测试通过。

如果你本地运行都有错误，放到云上去，想必也难以正常运行。

environment.yml 文件非常重要，它来告诉 mybinder，需要如何为你的代码运行准备环境。

我们打开看看该文件的内容：

```
dependencies:
- python=3
- pip:
  - spacy
  - ipykernel
  - scipy
  - numpy
  - scikit-learn
  - matplotlib
  - pandas
  - thinc
```

这个文件首先告诉 mybinder，你的 Python 版本。我们采用的是 3.6 版。所以只需要指定 python=3 即可。mybinder 会自动为你下载安装最新的。

然后这个文件说明需要使用 pip 工具安装哪些软件包。我们需要把所有依赖的安装包都罗列出来。

这就是之前，我总在教程里给你说明的那些准备步骤。

但是这还没有完，因为 mybinder 只是为你安装好了一些软件依赖。

这里还有两个步骤需要处理：

- 为了分析语义，我们需要调用预训练的Word2vec模型，这需要 mybinder 为我们提前下载好。
- Jupyter Notebook 打开后，应当使用的 kernel 名称为 wangshuyi，这个 kernel 目前还没有在 Jupyter 里面注册。我们需要 mybinder 代劳。

为了完成上述两个步骤，你就需要准备最后一个 postBuild 文件。



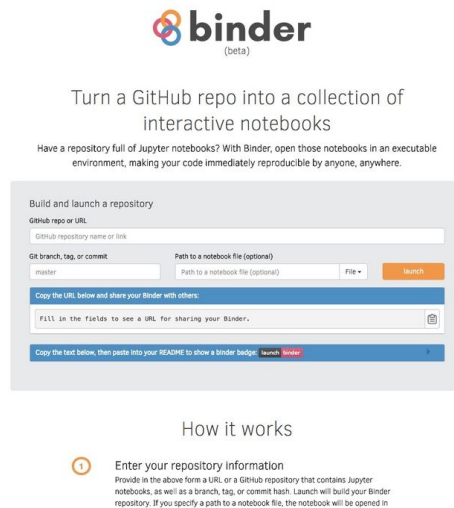
```
python -m spacy download en
python -m spacy download en_core_web_lg
python -m ipykernel install --user --name=wangshuyi
```

跟它的名字一样。它是在 mybinder 依据 environment.yml 安装了依赖组建后，依次执行的命令。如果你的代码需要其他的命令提供环境支持，也可以放在这里。

至此，你的准备工作就算结束了。

魔法表演正式开始。

请打开 mybinder 的[网址](https://mybinder.org/) (mybinder.org/) 。



知乎 @王树义

在 “GitHub repo or URL” 一栏，填写我们的 github 代码仓库链接，即：

```
https://github.com/wshuyi/demo-spacy-text-processing
```

我们希望一进入界面，就自动打开 demo.ipynb ，因此需要在 “Path to a notebook file (optional)” 一栏填写 demo.ipynb 。

这时，你会发现 “Copy the URL below and share your Binder with others:” 一栏中，出现了你的代码运行环境网址。

```
https://mybinder.org/v2/gh/wshuyi/demo-spacy-text-processing/master?filepath=demo.ipynb
```

点击右侧的 “复制” 按钮保存到你的记事本里面。将来找到你转换好的运行环境，就全靠它了。

妥善保存地址后，点击 “Launch” 按钮。



Turn a GitHub repo into a collection of interactive notebooks

Have a repository full of Jupyter notebooks? With Binder, open those notebooks in an executable environment, making your code immediately reproducible by anyone, anywhere.

Build and launch a repository

GitHub repo or URL

Git branch, tag, or commit

Path to a notebook file (optional)

Copy the URL below and share your Binder with others:
<https://mybinder.org/v2/gh/wangchuj/demo-spacy-text-processing/master?filepath=demo.ipynb>

Copy the text below, then paste into your README to show a binder badge:

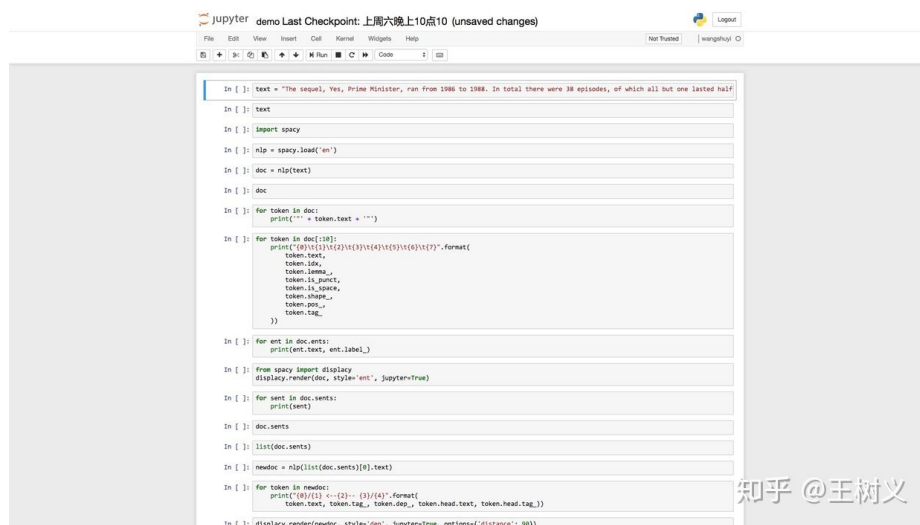
How it works

知乎 @王树义

根据你的依赖安装包数量等因素，你需要等待的时间长短不一。但是只有**第一次构建**的时候，需要花一些时间。

以后每一次调用执行，就都会非常快了。

构建完毕后，mybinder 会自动为我们开启对应的运行环境。



知乎 @王树义

很有成就感吧！

测试一下，能够正常运行代码，就证明我们成功了。

但是你会发现，不对啊！

老师你刚才用 iPad 展示的，不是高级版的 Jupyter Lab 吗？怎么又变成了 Jupyter Notebook 了？

我也想要高级版！

别着急。

看看你目前的链接地址：

▲ 赞同 1283 ▼

● 62 条评论

➦ 分享

❤ 喜欢

★ 收藏

📄 申请转载

...

ster?filepath=demo.ipynb



你只需要做个小小的调整，将其中的：

```
?filepath=
```

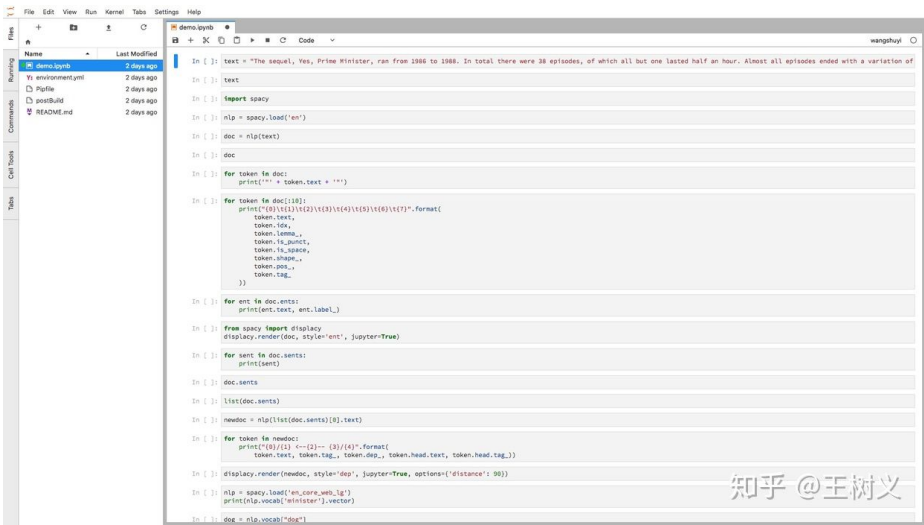
替换为：

```
?urlpath=lab/tree/
```

替换后的链接为：

```
https://mybinder.org/v2/gh/wshuyi/demo-spacy-text-processing/master?urlpath=lab/tree/d
```

把它输入到浏览器，看看出来的结果：



这下没问题了吧？

原理

你是不是觉得，mybinder 很黑科技？

其实，也不算。

它只是把已有的几项技术，链接了起来。

这大概也算是“积木式创新”的一个实例吧。

我们看看 mybinder 的说明：

How it works



1

Enter your repository information

Provide in the above form a URL or a GitHub repository that contains Jupyter notebooks, as well as a branch, tag, or commit hash. Launch will build your Binder repository. If you specify a path to a notebook file, the notebook will be opened in your browser after building.

2

We build a Docker image of your repository

Binder will search for a dependency file, such as requirements.txt or environment.yml, in the repository's root directory ([more details on more complex dependencies in documentation](#)). The dependency files will be used to build a Docker image. If an image has already been built for the given repository, it will not be rebuilt. If a new commit has been made, the image will automatically be rebuilt.

3

Interact with your notebooks in a live environment!

A JupyterHub server will host your repository's contents. We offer you a reusable link and badge to your live repository that you can easily share with others. 知乎 @王树义

可以看到，其中最为关键的技术，是用了 docker。

Docker 是个什么东西呢？

简单来说，Docker 就是为了不同平台上，都能够顺利执行同一份代码的保障工具。

你有些犹疑，这说的不是 Java 吗？

没错，Java 的宣传口号，就是一次编码，各处运行。

它利用虚拟机，来保障这种能力。

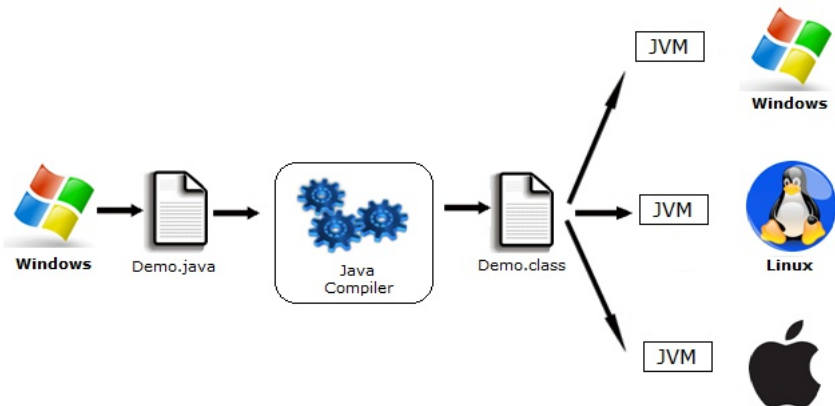
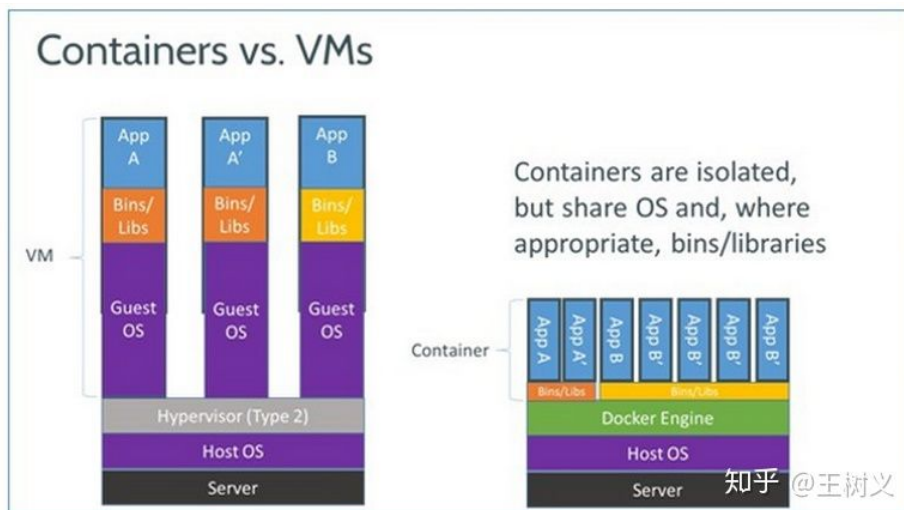


fig : Portability in Java

知乎 @王树义

但是，如果你经常使用 Java 开发出来的工具，就应该了解痛点有哪些了。

至少，你应该对 Java 程序的运行速度，有一些体会。



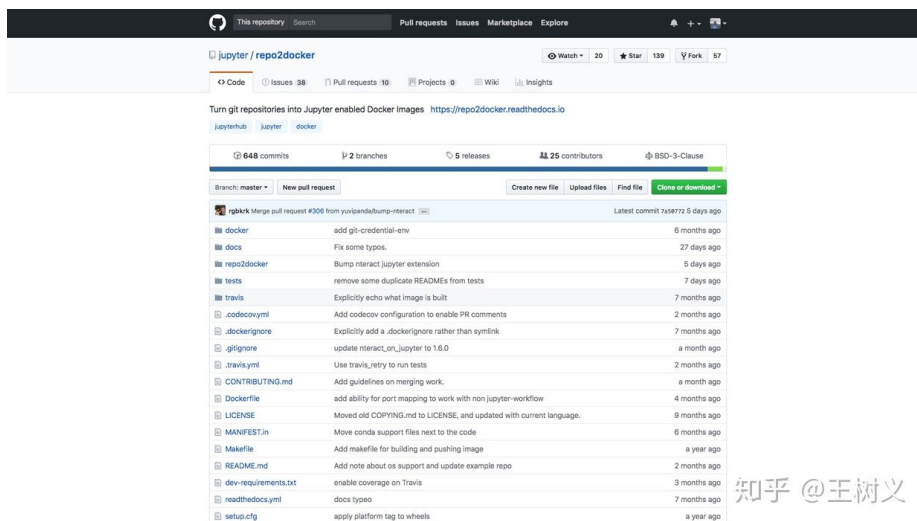
上图中，左侧是虚拟机，右侧是Docker。

Docker 不但效率上要强过 Java 虚拟机，而且它支持的编程语言也不仅仅是一种。

至于其他好处，咱们就不展开了。否则听起来像广告。

其实，把 github 代码仓库转换为 docker 镜像（image）的工作，也不是 mybinder 自己来做的。

它调用的，是另外的一个工具，叫做 [repo2docker](https://github.com/jupyter/repo2docker) (github.com/jupyter/repo2docker) 。



而你的浏览器能够执行 Python 代码，是因为 Jupyter Notebook（或者Lab）本来就是建立在“浏览器/服务器”（Browser / Server, B/S）结构上。

如果你已经在本地计算机安装过 Anaconda，那不妨看看本地执行这个语句：

```
jupyter lab
```

会出现什么？

```
jupyter lab -- python -- python /anaconda2/bin/jupyter-lab -- 80x24
[I 09:47:41.335 LabApp] Saving file at /demo.ipynb
[I 09:49:41.467 LabApp] Saving file at /demo.ipynb
[I 09:50:01.771 LabApp] Saving file at /demo.ipynb
[I 09:50:57.345 LabApp] Saving file at /demo.ipynb
[I 10:07:20.780 LabApp] Starting buffering for fdb593f6-fa99-4e14-a041-b6acfd2e971a:a65fa686f0b17f4a8acae86f43f67dda
[I 10:07:23.997 LabApp] Kernel restarted: fdb593f6-fa99-4e14-a041-b6acfd2e971a
[I 10:07:24.081 LabApp] Saving file at /demo.ipynb
[I 10:07:24.913 LabApp] Adapting to protocol v5.1 for kernel fdb593f6-fa99-4e14-a041-b6acfd2e971a
[I 10:07:24.914 LabApp] Restoring connection for fdb593f6-fa99-4e14-a041-b6acfd2e971a:a65fa686f0b17f4a8acae86f43f67dda
[I 10:07:24.914 LabApp] Replying 3 buffered messages
[W 10:08:37.209 LabApp] delete /tsne.pdf
[W 20:39:29.617 LabApp] WebSocket ping timeout after 959955 ms.
[I 10:29:35.805 LabApp] Starting buffering for fdb593f6-fa99-4e14-a041-b6acfd2e971a:a65fa686f0b17f4a8acae86f43f67dda
[I 10:29:51.568 LabApp] Adapting to protocol v5.1 for kernel fdb593f6-fa99-4e14-a041-b6acfd2e971a
[I 10:29:51.570 LabApp] Restoring connection for fdb593f6-fa99-4e14-a041-b6acfd2e971a:a65fa686f0b17f4a8acae86f43f67dda
[I 10:31:21.309 LabApp] Starting buffering for fdb593f6-fa99-4e14-a041-b6acfd2e971a:a65fa686f0b17f4a8acae86f43f67dda
```

知乎 @王树义

对，它开启了一个服务器，然后打开你的浏览器，跟这个服务器通讯。

Jupyter 的这种设计，本身就让它扩展极为方便。

无论 Jupyter 服务器是运行在你的本地笔记本上，还是摆在另一个大洲的机房，对你执行 Python 代码来说，都是没有本质区别的。

另外，如果你以为 mybinder 只能让你在浏览器上跑 Python 代码，那就太小瞧它了。

学过 R 的同学，请点击[这个链接\(t.cn/R3JLY2S\)](https://t.cn/R3JLY2S)，看看有什么惊喜。

小结

总结一下，本文为你讲述了以下内容：

- 如何利用 mybinder，把一个 github repo 一键转换成 Jupyter Lab 运行环境；
- 如何在各种不同操作系统的浏览器上，运行该环境，编写、执行与修改代码；
- mybinder 转换 github repo 的幕后英雄 docker 简介。

我希望能想到的，不仅仅是这点儿简单的用途。

提几个问题给你，作为思考题：

- 如果代码执行都在云端完成，教学实验室机房还有没有必要预装一大堆软件，且不定期更新维护



- 学校的编程练习、作业和考试有没有可能通过这种方式，直接远程进行，并且自动化评分？
- 既然应用的技术都是开源的，你有没有可能利用这些开源工具搞个创业项目。例如提供深度学习环境，租赁给科研机构与创业公司？

期待你举一反三，做出有趣又有意义的创新来。

讨论

在 iPad 上运行 Python 代码的感觉怎么样？你用过类似的产品吗？你觉得有了这种技术，在日常工作和学习中，还可以有哪些有趣的应用场景？欢迎留言，把你的经验和思考分享给大家，我们一起交流讨论。

如果你对我的文章感兴趣，欢迎点赞，并且微信关注和置顶我的公众号“[玉树芝兰](#)”(nkwangshuyi)。

如果本文可能对你身边的亲友有帮助，也欢迎你把本文通过微博或朋友圈分享给大家。让他们一起参与到我们的讨论中来。

延伸阅读

[如何高效入门数据科学？](#)

发布于 2018-05-14

「原创不易，感谢打赏！」

赞赏

还没有人赞赏，快来当第一个赞赏的人吧！

[Python](#) [编程](#) [科技](#)

文章被以下专栏收录



玉树芝兰

推荐阅读

▲ 赞同 1283 ▼ ● 62 条评论 ↗ 分享 ♥ 喜欢 ★ 收藏 📄 申请转载 ...

如何为Python程序制作Windows安装包？

我们使用Python可以编写很多提高工作效率和学习效率的小工具，在编写完之后，如果我们想将其分享给更多的人使用，那么最便捷的方法就是将其打包为可执行程序。在Windows环境下，我们使用...
州的先生 发表于Pytho...



办公自动化与python编程探索 (二)
冠冠公考

62 条评论

切换为时间排序

写下你的评论...



黑夜不再来

2018-05-14

Pythonista了解一下

👍 13



谢明家 回复 黑夜不再来

2019-04-25

这个安装库方便吗

👍 赞



贾晓明 回复 谢明家

2019-06-19

不能自己安装库，包括pandas，有些鸡肋

👍 5



jufei9494

2018-05-15

看了半天，原来是广告

👍 12



xxxxx

2018-05-14

使用ssh，远程登陆服务器加端口映射，本地就能打开远程的jupyter了

👍 8



Milo Wang 回复 xxxxx

2018-05-31

请问这个是不是得有个云服务器才能实现？

👍 赞



xxxxx 回复 Milo Wang

2018-05-31

云只是一个手段而已，如果是家庭网络，电脑上开着jupyter，用iPad登陆也是可以的

👍 1

展开其他 1 条回复



王伟强

2018-05-17

拿jvm和docker比，这两者根本就不是一个抽象级别的东西，jvm上的东西能跨硬件平台（x86 mips power arm）跨操作系统（linux mac unix win），docker只是跨宿主主机而已，根本不能跨硬件平台和操作系统（docker的win版本和mac版本都是跑了一个linux虚拟机）

👍 6



看得远

2018-05-20

Colab了解一下

👍 5

赞同 1283

62 条评论

分享

喜欢

收藏



申请转载

...

2018-05-22

死在环境搭建路上的就不要学了，基本的悟性和毅力都没有能学什么



👍 4

 吴咏伟  回复 刺猬爱学习 2019-02-14
对于初学者，装环境远比开发本身复杂好不好。你这么说话，应该也是个熟手了，但明显你低估了搭环境所需的各种琐碎知识了.....



👍 18

 空心邻域 回复 刺猬爱学习 06-03
你就这么怕自己被淘汰嘛😏
连会按照教程配环境都有优越感了呢

👍 1

 吴咏伟  2019-02-14
顶一下。真不知道为什么喷子这么多。mybinder只是个有用的工具，有用就用，拿它跟其他没可比性的东西比我也真服了。

👍 2

 居然熊猫  2018-05-22
稍加使用即可发现，jupyter web在ipad safari上连基本的打开文件都做不到（因为需要双击）。

👍 2

 metalluster 回复 居然熊猫  06-18
亲测双击可以打开

👍 赞

 居然熊猫  回复 metalluster 06-19
现在是2021年了 hhhhh


👍 赞

 小不点 2018-05-22
store有现成的IDE。。。好吧

👍 2

 「已注销」 回复 小不点 2020-09-19
有吗，马上入手自己第一个iPad

👍 赞

 吴星 2018-05-17
ssh 就能做到的事情，这也能推荐给我？

👍 2

 丁毅 2018-06-18
termux做服务器就好了，我的就是这样ipad连，用emacs

👍 1

 林哇哇 回复 丁毅 2019-12-02
请问下具体您ipad连termux是怎么操作

👍 赞

 爷爷泡的猪 2018-05-28
从入门到放弃

👍 1





其实就是远程服务器，做了一个UI包装，测试不如本地terminal，界面相应速度不如notebook（当然取决于网络环境，不过再好也没法和localhost比），那么局域网内搭远程桌面是个最高效的解决方案（要折腾），或者teamviewer。还有就是其他人说的对，Pythonista了解一下，不过涉及编辑，调试，GitHub部署，要跟iFiles，Markdown app，还有几个忘了叫啥，一起用，文件传来传去的特么蛋疼（或许可以用Workflow app解决？），明明PC上subl一个能顶全部。还有很多想说的，先mark一下

👍 1



stevapple

2018-05-19

Pythonista 了解一下 +1

👍 1



Yehowah

2018-05-14

QPython了解一下

👍 1



purewater1 回复 Yehowah

2018-05-29

qPython有些库不好安装

👍 赞



大白

09-19

? 所以说说了半天到底用什么软件运行

👍 赞



Hakureirm 回复 大白

09-20

这是个网站，所以浏览器就行啊

👍 赞



safiers

07-25

老师您好，想请教一个基础问题。

我在用python模块pandas导入ipad中的excel数据时，不知道怎么指向数据文件所在的路径。比如我的excel数据文件4.xlsx放在我的iPad当中，当我输入

```
import pandas as pd
```

```
pd.read_excel('~/.我的iPad/4.xlsx')
```

结果提示No such file or directory

请问要怎么解决，希望得到一点帮助，感谢。

👍 赞



小白菜菜子 📧

03-17

老师，我按照这种方法创建的，import时提示包不存在，是什么原因呢

👍 赞



梵乐

2020-12-24

cool

👍 赞



渔火

2020-12-11

我试了，保存后GitHub没有同步哦，大家是这样的吗

👍 赞

1 2 3 下一页