

# 自然语言处理

实验指导

# 目 录

- 安装Anaconda
- 环境搭建1：conda或pip换源
- 环境搭建2：安装Pytorch
- 环境搭建3：安装其余依赖
- 华为云环境配置
- 模型训练
- 模型推断
- 配置IDE

# 1.1 安装Anaconda（可选，推荐）

## ■ 为什么需要虚拟环境

- 可以让每个项目配置一个自定义的Python解释器环境
- 尤其是AI领域中，不同参考实现的环境之间大多存在差异与冲突，更需要使用虚拟环境来管理

## ■ 为什么选用conda

- conda包含在Anaconda中，是虚拟环境和包管理的集成
- 相比于python自带的virtualenv虚拟环境：conda预安装了numpy等库，方便科学计算；conda集成了包管理器，能够方便地安装CUDA等GPU开发环境下需要的非Python工具

## ■ 如何安装Anaconda

- 参考<https://zhuanlan.zhihu.com/p/75717350> 推荐清华源下载

## 1.2 conda或pip换源

- 由于下载使用的默认服务器通常在国外，速度偏慢，一般直接采用国内镜像对下载服务器（源）进行替换。
- conda换源
  - 修改用户目录下的 `.condarc` 文件。Windows 用户无法直接创建名为 `.condarc` 的文件，可先执行 `conda config --set show_channel_urls yes` 生成该文件之后再修改。
  - 以换成清华源为例，修改内容参照：  
<https://mirrors.tuna.tsinghua.edu.cn/help/anaconda/>
- pip换源
  - 如果没有使用虚拟环境，则需要使用pip安装依赖，pip换源请参考：  
<https://mirrors.tuna.tsinghua.edu.cn/help/pypi/>

## 1.3 安装Pytorch

### ■ 使用Anacoda:

在Anaconda Prompt中进行如下操作:

# 创建虚拟环境并激活

```
conda create -n nlplab python=3.7 # 创建名为  
nlplab 的虚拟环境
```

```
conda activate nlplab # 激活虚拟环境nlplab, 成  
功执行后应看到命令行首部由 (base) 变为  
(nlplab)
```

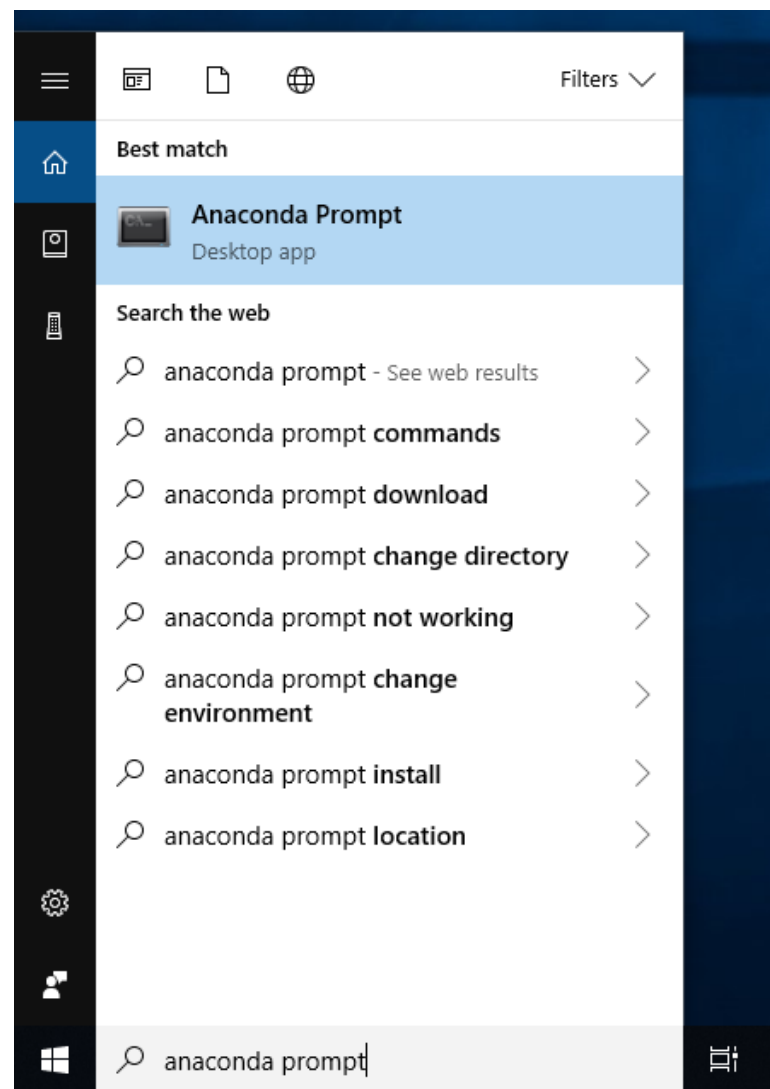
# 安装 Pytorch 1.7.1 CPU 版本

```
conda install pytorch=1.7.1
```

# 其他虚拟环境相关命令

```
conda deactivate # 退出当前虚拟环境
```

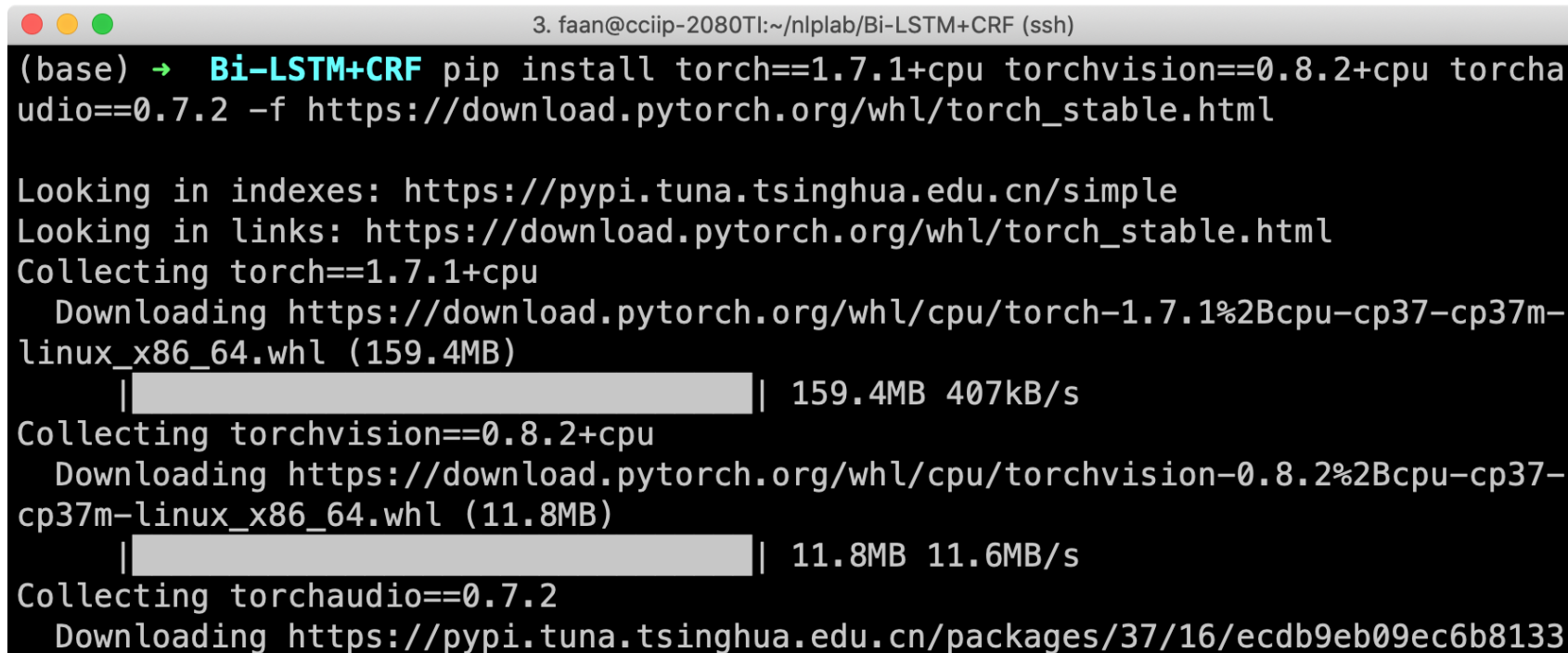
```
conda info -e # 查看所有虚拟环境, *指示当前所  
处环境
```



## 1.3 安装Pytorch

### ■ 如果使用pip，直接运行：

- `pip install torch==1.7.1+cpu torchvision==0.8.2+cpu torchaudio==0.7.2 -f https://download.pytorch.org/whl/torch_stable.html`



```

3. faan@cciiip-2080TI:~/nlp/BI-LSTM+CRF (ssh)
(base) → Bi-LSTM+CRF pip install torch==1.7.1+cpu torchvision==0.8.2+cpu torchaudio==0.7.2 -f https://download.pytorch.org/whl/torch_stable.html

Looking in indexes: https://pypi.tuna.tsinghua.edu.cn/simple
Looking in links: https://download.pytorch.org/whl/torch_stable.html
Collecting torch==1.7.1+cpu
  Downloading https://download.pytorch.org/whl/cpu/torch-1.7.1%2Bcpu-cp37-cp37m-linux_x86_64.whl (159.4MB)
    |████████████████████████████████████████| 159.4MB 407kB/s
Collecting torchvision==0.8.2+cpu
  Downloading https://download.pytorch.org/whl/cpu/torchvision-0.8.2%2Bcpu-cp37-cp37m-linux_x86_64.whl (11.8MB)
    |████████████████████████████████████████| 11.8MB 11.6MB/s
Collecting torchaudio==0.7.2
  Downloading https://pypi.tuna.tsinghua.edu.cn/packages/37/16/ecdb9eb09ec6b8133
  
```

## 1.4 安装其余依赖

- 由于本项目所使用的部分依赖项并未被conda收录，只能使用pip安装所有依赖：
  - `pip install -r requirements.txt`
- 如果安装速度过慢请检查是否成功配置了1.2中的pip换源。

## 1.5 华为云

### ■ 创建OBS桶，上传实验数据：

- ❑ 登录OBS平台并创建桶
- ❑ 新建“NLP”文件夹
- ❑ 本地解压实验数据，上传至云端“NLP”文件夹

### ■ 创建notebook开发环境

- ❑ 登录华为云modelarts控制台
- ❑ 点击开发环境->notebook->创建，并输入notebook描述
- ❑ 选择Ascend 910环境
- ❑ notebook存储位置选择之前创建的“NLP”文件夹

MindSpore中Bi-LSTM+CRF链接：[https://www.mindspore.cn/tutorials/application/zh-CN/r1.7/nlp/sequence\\_labeling.html](https://www.mindspore.cn/tutorials/application/zh-CN/r1.7/nlp/sequence_labeling.html)



# 1.6 模型训练

## ■ 依次运行数据预处理、模型训练

### □ 数据预处理: `cd data && python data_u.py`

```

2. faan@ccitp-2080TI:~/nlp/Bi-LSTM+CRF/data (ssh)
(nlplab) → Bi-LSTM+CRF cd data && python data_u.py
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 7, 22, 23, 24, 25, 18,
10, 11, 12, 13, 26, 27, 28, 7, 29, 30, 31, 28, 18, 32, 33, 18, 34, 35, 36, 37]
['"', '人', '们', '常', '说', '生', '活', '是', '一', '部', '教', '科', '书', ' ', ' ', '而', '血', '与',
'火', '的', '战', '争', '更', '是', '不', '可', '多', '得', '的', '教', '科', '书', ' ', ' ', '她', '确',
'实', '是', '名', '副', '其', '实', '的', ' ', '我', '的', '大', '学', ' ', ' ', '。']
[3, 0, 2, 3, 3, 0, 2, 3, 3, 3, 0, 1, 2, 3, 3, 3, 3, 3, 0, 2, 3, 3, 0, 1, 1, 2, 3, 0, 1, 2, 3, 3, 0
, 2, 3, 0, 1, 1, 2, 3, 3, 3, 3, 0, 2, 3, 3]
['S', 'B', 'E', 'S', 'S', 'B', 'E', 'S', 'S', 'S', 'B', 'M', 'E', 'S', 'S', 'S', 'S', 'S', 'S', 'S', 'B',
'E', 'S', 'B', 'M', 'M', 'E', 'S', 'S', 'B', 'M', 'E', 'S', 'S', 'B', 'E', 'S', 'B', 'M', 'M', 'E',
S', 'S', 'S', 'S', 'B', 'E', 'S', 'S']
(nlplab) → data

```

### □ 模型训练: 项目根目录下运行 `python run.py` [--cuda 使用此参数 需要系统拥有Nvidia独显, 且Pytorch安装gpu版本]

```

2. python run.py (ssh)
(nlplab) → Bi-LSTM+CRF python run.py
2021-05-31 12:31:08,956 DEBUG word_embeddings.weight: torch.Size([5168, 100]), require_grad=True
2021-05-31 12:31:08,956 DEBUG lstm.weight_ih_l0: torch.Size([400, 100]), require_grad=True
2021-05-31 12:31:08,956 DEBUG lstm.weight_hh_l0: torch.Size([400, 100]), require_grad=True
2021-05-31 12:31:08,957 DEBUG lstm.bias_ih_l0: torch.Size([400]), require_grad=True
2021-05-31 12:31:08,957 DEBUG lstm.bias_hh_l0: torch.Size([400]), require_grad=True
2021-05-31 12:31:08,957 DEBUG lstm.weight_ih_l0_reverse: torch.Size([400, 100]), require_grad=True
2021-05-31 12:31:08,957 DEBUG lstm.weight_hh_l0_reverse: torch.Size([400, 100]), require_grad=True
2021-05-31 12:31:08,957 DEBUG lstm.bias_ih_l0_reverse: torch.Size([400]), require_grad=True
2021-05-31 12:31:08,957 DEBUG lstm.bias_hh_l0_reverse: torch.Size([400]), require_grad=True
2021-05-31 12:31:08,957 DEBUG hidden2tag.weight: torch.Size([4, 200]), require_grad=True
2021-05-31 12:31:08,957 DEBUG hidden2tag.bias: torch.Size([4]), require_grad=True
2021-05-31 12:31:08,957 DEBUG crf.start_transitions: torch.Size([4]), require_grad=True
2021-05-31 12:31:08,957 DEBUG crf.end_transitions: torch.Size([4]), require_grad=True
2021-05-31 12:31:08,957 DEBUG crf.transitions: torch.Size([4, 4]), require_grad=True
2021-05-31 12:35:10,478 DEBUG epoch 0-step 100 loss: 25.685805
2021-05-31 12:38:41,506 DEBUG epoch 0-step 200 loss: 12.174740

```

## 1.7 模型预测

### ■ 运行infer.py: `python infer.py`

- ❑ 该脚本在第五行指定了用于预测的模型，需要将其替换为你的保存结果；
- ❑ 脚本默认使用save目录中的初始模型进行预测，结果保存为文件。

```
2. faan@cciiip-2080Tl:~/nlplab/Bi-LSTM+CRF (ssh)
(nlplab) → Bi-LSTM+CRF python infer.py
(nlplab) → Bi-LSTM+CRF head cws_result.txt
扬帆 远 东 做 与 中 国 合 作 的 先 行
希 腊 的 经 济 结 构 较 特 殊 。
海 运 业 雄 踞 全 球 之 首 ， 按 吨 位 计 占 世 界 总 数 的 1 7 % 。
另 外 旅 游 、 侨 汇 也 是 经 济 收 入 的 重 要 组 成 部 分 ， 制 造 业 规 模 相 对 较 小 。
多 年 来 ， 中 希 贸 易 始 终 处 于 较 低 的 水 平 ， 希 腊 几 乎 没 有 在 中 国 投 资 。
十 几 年 来 ， 改 革 开 放 的 中 国 经 济 高 速 发 展 ， 远 东 在 崛 起 。
瓦 西 里 斯 的 船 只 中 有 4 0 % 驶 向 远 东 ， 每 个 月 几 乎 都 有 两 三 条 船 停 靠
中 国 港 口 。
他 感 受 到 了 中 国 经 济 发 展 的 大 潮 。
他 要 与 中 国 人 合 作 。
他 来 到 中 国 ， 成 为 第 一 个 访 华 的 大 船 主 。
```

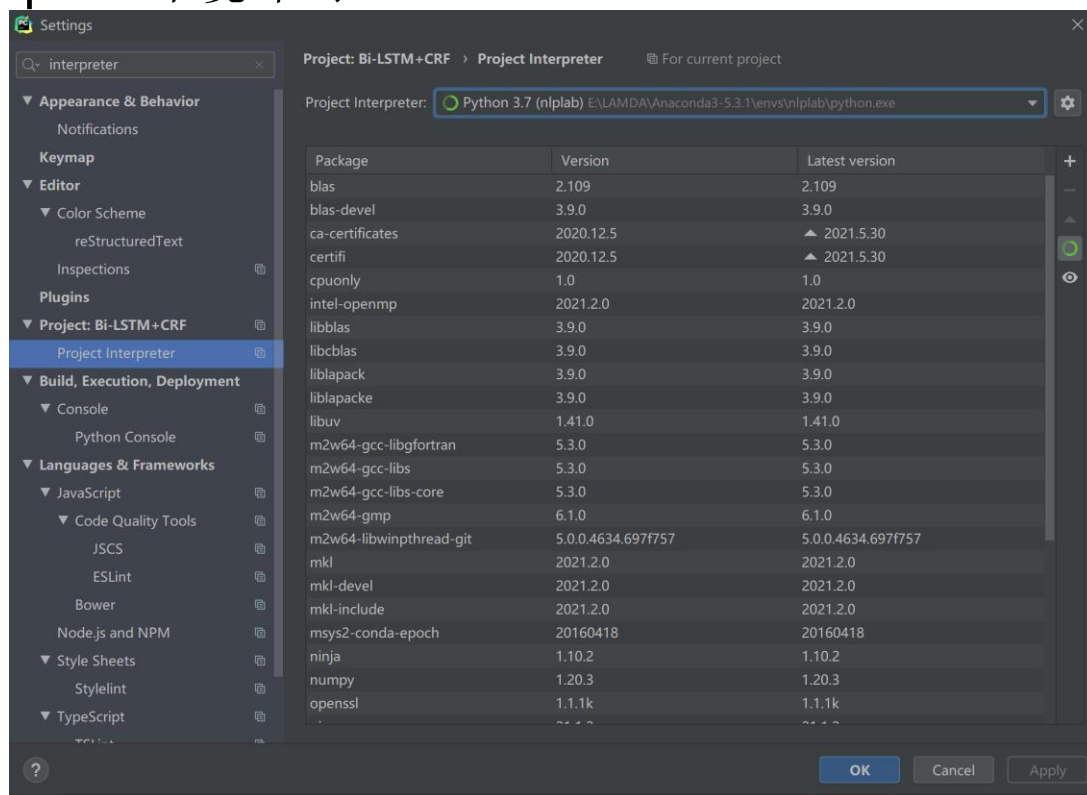
## 1.8 配置IDE

- 在需要对于已有代码进行修改时，IDE能够方便你进行调试等操作。推荐使用的IDE有：功能最为完备的PyCharm、或者更轻量的VsCode。
  - **PyCharm**：功能更加完备，如图形化管理环境依赖、远程项目修改时可增量修改。
  - **VsCode**：轻量，拓展性强，通过插件支持各类语言。因为需要安装插件，所以不是开箱即用的。

## 1.8 配置IDE

### ■ PyCharm开发环境配置

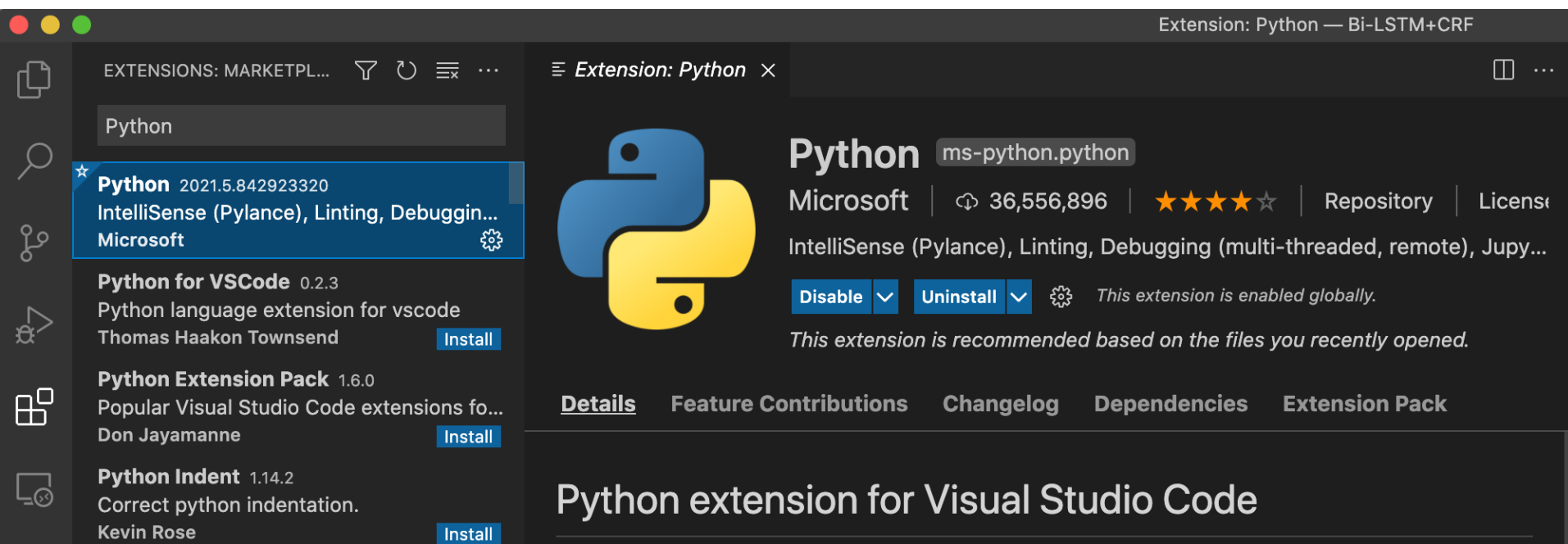
- 点击 File/Settings 打开设置页，搜索 Project Interpreter 选择配置好的 nlplab 环境即可



# 1.8 配置IDE

## ■ VsCode开发环境配置

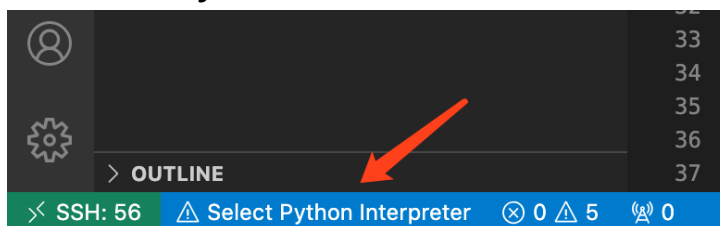
- 在插件页安装Python解释器插件：



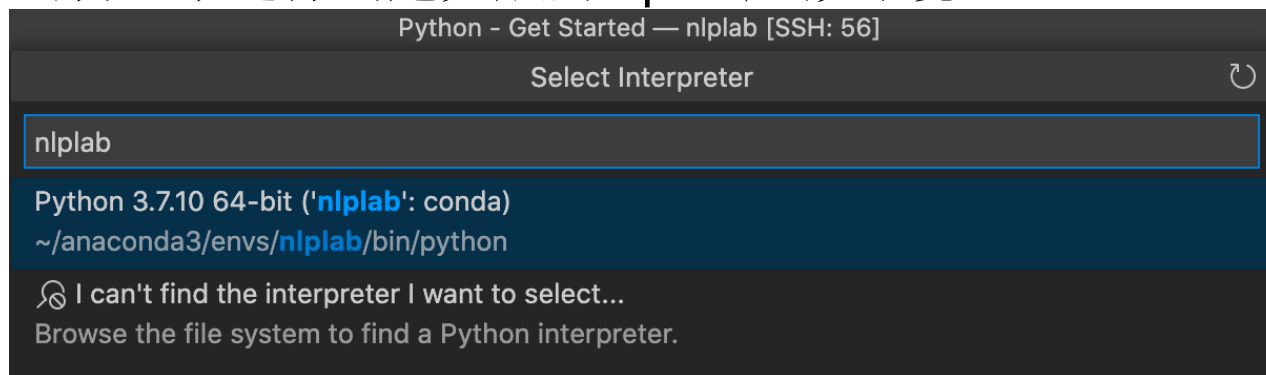
## 1.8 配置IDE

### ■ VsCode开发环境配置

- 在Vscode中打开任意Python文件，点击左下角选择环境：



- 在弹出窗口中选择创建完成的nlplab虚拟环境：



- 单击键盘F5按键，选择调试方式即可。

## 1.9 课外拓展

- 了解信息抽取前沿论文，在 ModelArts 平台上尝试复现

参考文献

- Boundary Smoothing for Named Entity Recognition
- Label Semantics for Few Shot Named Entity Recognition
- A Simple yet Effective Relation Information Guided Approach for Few-Shot Relation Extraction
- Event-Event Relation Extraction using Probabilistic Box Embedding

- 华为云CSIG2022:中英文购物小票信息理解大赛

华为云大赛 > 大赛列表 > 大赛详情

### CSIG2022:中英文购物小票信息理解大赛 火热进行中

该任务旨在探索算法对复杂视觉信息理解的能力。参赛队伍需要设计算法，结合人工标注的OCR文字框，以及文字，对图像中的关键信息进行理解，重点考察参赛选手对关键信息的抽取，筛选，整理和汇聚的能力。

举办方：中国图象图形学学会、华为竞赛部、华为质量与流程IT、华为云大赛平台、华为云ModelArts

**奖金：¥100,000**

**67**  
团队数

**326**  
报名人数

初赛截止时间：2022/07/18