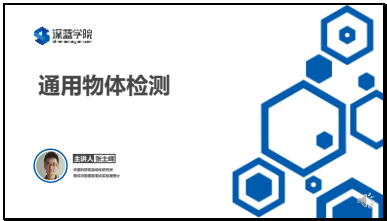


幻灯片 1



幻灯片 2



幻灯片 3



幻灯片 4



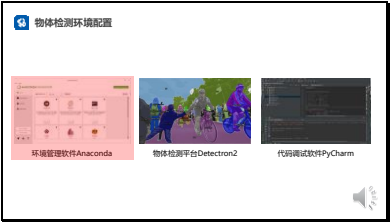
幻灯片 5



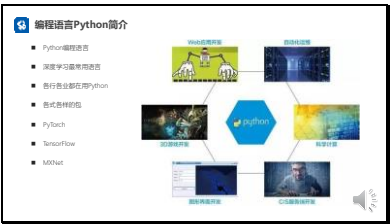
幻灯片 6



幻灯片 7



幻灯片 8



幻灯片 9





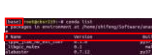
幻灯片 13

[illegible]

幻灯片 14

## Anaconda的安装

- ❖ 关闭网络，再打开终端启动Anaconda
- ❖ 终端左边显示提示，按回车安装成功
- ❖ base的Anaconda安装以环境
- ❖ 输入命令`conda info`，查看已安装的环境
- ❖ Anaconda的默认环境安装了九十多个包
- ❖ 名称、版本、build、channel
- ❖ conda 管理不同环境的环境
- ❖ python：启动python交互界面



The terminal screenshot shows the following commands and output:

```

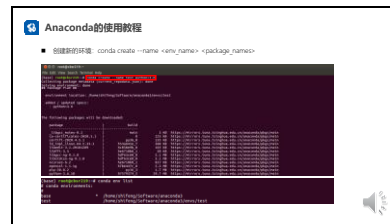
$ curl -O https://repo.continuum.io/archive/Anaconda3-4.2.0-Linux-x86_64.pkg.tar.xz
$ tar -xvf Anaconda3-4.2.0-Linux-x86_64.pkg.tar.xz
$ cd Anaconda3
$ ./install.sh
$ conda info

# Anaconda3 (base)
# Linux x86_64
# base environment
# installed 2016-09-26 16:04:00
# conda version 4.2.0
# conda build 3.0.26
# python 3.5.2
# setuptools 18.0
# pip 8.1.2
# openssl 1.0.2j
# sqlite 3.13.0
# xz 5.0.5
# zlib 1.2.8
# libyaml 0.2.2
# tk 8.5.13
# ncurses 6.0
# readline 6.2
# libc 2.23
# ld 2.23.2
# gcc 4.8.5
# glibc 2.23
# libffi 3.2.1
# libgcc 4.8.5
# libstdc++ 4.8.5
# libtinfo 6.0
# libz 1.2.8
# openssl 1.0.2j
# readline 6.2
# sqlite 3.13.0
# tk 8.5.13
# xz 5.0.5
# zlib 1.2.8
# libyaml 0.2.2
# tk 8.5.13
# ncurses 6.0
# readline 6.2
# libc 2.23
# ld 2.23.2
# gcc 4.8.5
# glibc 2.23
# libffi 3.2.1
# libgcc 4.8.5
# libstdc++ 4.8.5
# libtinfo 6.0
# libz 1.2.8
# openssl 1.0.2j
# readline 6.2
# sqlite 3.13.0
# tk 8.5.13
# xz 5.0.5
# zlib 1.2.8
# libyaml 0.2.2
# tk 8.5.13
# ncurses 6.0
# readline 6.2
# libc 2.23
# ld 2.23.2
# gcc 4.8.5
# glibc 2.23
# libffi 3.2.1
# libgcc 4.8.5
# libstdc++ 4.8.5
# libtinfo 6.0
# libz 1.2.8
# openssl 1.0.2j
# readline 6.2
# sqlite 3.13.0
# tk 8.5.13
# xz 5.0.5
# zlib 1.2.8
# libyaml 0.2.2
# tk 8.5.13
# ncurses 6.0
# readline 6.2
# libc 2.23
# ld 2.23.2
# gcc 4.8.5
# glibc 2.23
# libffi 3.2.1
# libgcc 4.8.5
# libstdc++ 4.8.5
# libtinfo 6.0
# libz 1.2.8
# openssl 1.0.2j
# readline 6.2
# sqlite 3.13.0
# tk 8.5.13
# xz 5.0.5
# zlib 1.2.8
# libyaml 0.2.2
# tk 8.5.13
# ncurses 6.0
# readline 6.2
# libc 2.23
# ld 2.23.2
# gcc 4.8.5
# glibc 2.23
# libffi 3.2.1
# libgcc 4.8.5
# libstdc++ 4.8.5
# libtinfo 6.0
# libz 1.2.8
# openssl 1.0.2j
# readline 6.2
# sqlite 3.13.0
# tk 8.5.13
# xz 5.0.5
# zlib 1.2.8
# libyaml 0.2.2
# tk 8.5.13
# ncurses 6.0
# readline 6.2
# libc 2.23
# ld 2.23.2
# gcc 4.8.5
# glibc 2.23
# libffi 3.2.1
# libgcc 4.8.5
# libstdc++ 4.8.5
# libtinfo 6.0
# libz 1.2.8
# openssl 1.0.2j
# readline 6.2
# sqlite 3.13.0
# tk 8.5.13
# xz 5.0.5
# zlib 1.2.8
# libyaml 0.2.2
# tk 8.5.13
# ncurses 6.0
# readline 6.2
# libc 2.23
# ld 2.23.2
# gcc 4.8.5
# glibc 2.23
# libffi 3.2.1
# libgcc 4.8.5
# libstdc++ 4.8.5
# libtinfo 6.0
# libz 1.2.8
# openssl 1.0.2j
# readline 6.2
# sqlite 3.13.0
# tk 8.5.13
# xz 5.0.5
# zlib 1.2.8
# libyaml 0.2.2
# tk 8.5.13
# ncurses 6.0
# readline 6.2
# libc 2.23
# ld 2.23.2
# gcc 4.8.5
# glibc 2.23
# libffi 3.2.1
# libgcc 4.8.5
# libstdc++ 4.8.5
# libtinfo 6.0
# libz 1.2.8
# openssl 1.0.2j
# readline 6.2
# sqlite 3.13.0
# tk 8.5.13
# xz 5.0.5
# zlib 1.2.8
# libyaml 0.2.2
# tk 8.5.13
# ncurses 6.0
# readline 6.2
# libc 2.23
# ld 2.23.2
# gcc 4.8.5
# glibc 2.23
# libffi 3.2.1
# libgcc 4.8.5
# libstdc++ 4.8.5
# libtinfo 6.0
# libz 1.2.8
# openssl 1.0.2j
# readline 6.2
# sqlite 3.13.0
# tk 8.5.13
# xz 5.0.5
# zlib 1.2.8
# libyaml 0.2.2
# tk 8.5.13
# ncurses 6.0
# readline 6.2
# libc 2.23
# ld 2.23.2
# gcc 4.8.5
# glibc 2.23
# libffi 3.2.1
# libgcc 4.8.5
# libstdc++ 4.8.5
# libtinfo 6.0
# libz 1.2.8
# openssl 1.0.2j
# readline 6.2
# sqlite 3.13.0
# tk 8.5.13
# xz 5.0.5
# zlib 1.2.8
# libyaml 0.2.2
# tk 8.5.13
# ncurses 6.0
# readline 6.2
# libc 2.23
# ld 2.23.2
# gcc 4.8.5
# glibc 2.23
# libffi 3.2.1
# libgcc 4.8.5
# libstdc++ 4.8.5
# libtinfo 6.0
# libz 1.2.8
# openssl 1.0.2j
# readline 6.2
# sqlite 3.13.0
# tk 8.5.13
# xz 5.0.5
# zlib 1.2.8
# libyaml 0.2.2
# tk 8.5.13
# ncurses 6.0
# readline 6.2
# libc 2.23
# ld 2.23.2
# gcc 4.8.5
# glibc 2.23
# libffi 3.2.1
# libgcc 4.8.5
# libstdc++ 4.8.5
# libtinfo 6.0
# libz 1.2.8
# openssl 1.0.2j
# readline 6.2
# sqlite 3.13.0
# tk 8.5.13
# xz 5.0.5
# zlib 1.2.8
# libyaml 0.2.2
# tk 8.5.13
# ncurses 6.0
# readline 6.2
# libc 2.23
# ld 2.23.2
# gcc 4.8.5
# glibc 2.23
# libffi 3.2.1
# libgcc 4.8.5
# libstdc++ 4.8.5
# libtinfo 6.0
# libz 1.2.8
# openssl 1.0.2j
# readline 6.2
# sqlite 3.13.0
# tk 8.5.13
# xz 5.0.5
# zlib 1.2.8
# libyaml 0.2.2
# tk 8.5.13
# ncurses 6.0
# readline 6.2
# libc 2.23
# ld 2.23.2
# gcc 4.8.5
# glibc 2.23
# libffi 3.2.1
# libgcc 4.8.5
# libstdc++ 4.8.5
# libtinfo 6.0
# libz 1.2.8
# openssl 1.0.2j
# readline 6.2
# sqlite 3.13.0
# tk 8.5.13
# xz 5.0.5
# zlib 1.2.8
# libyaml 0.2.2
# tk 8.5.13
# ncurses 6.0
# readline 6.2
# libc 2.23
# ld 2.23.2
# gcc 4.8.5
# glibc 2.23
# libffi 3.2.1
# libgcc 4.8.5
# libstdc++ 4.8.5
# libtinfo 6.0
# libz 1.2.8
# openssl 1.0.2j
# readline 6.2
# sqlite 3.13.0
# tk 8.5.13
# xz 5.0.5
# zlib 1.2.8
# libyaml 0.2.2
# tk 8.5.13
# ncurses 6.0
# readline 6.2
# libc 2.23
# ld 2.23.2
# gcc 4.8.5
# glibc 2.23
# libffi 3.2.1
# libgcc 4.8.5
# libstdc++ 4.8.5
# libtinfo 6.0
# libz 1.2.8
# openssl 1.0.2j
# readline 6.2
# sqlite 3.13.0
# tk 8.5.13
# xz 5.0.5
# zlib 1.2.8
# libyaml 0.2.2
# tk 8.5.13
# ncurses 6.0
# readline 6.2
# libc 2.23
# ld 2.23.2
# gcc 4.8.5
# glibc 2.23
# libffi 3.2.1
# libgcc 4.8.5
# libstdc++ 4.8.5
# libtinfo 6.0
# libz 1.2.8
# openssl 1.0.2j
# readline 6.2
# sqlite 3.13.0
# tk 8.5.13
# xz 5.0.5
# zlib 1.2.8
# libyaml 0.2.2
# tk 8.5.13
# ncurses 6.0
# readline 6.2
# libc 2.23
# ld 2.23.2
# gcc 4.8.5
# glibc 2.23
# libffi 3.2.1
# libgcc 4.8.5
# libstdc++ 4.8.5
# libtinfo 6.0
# libz 1.2.8
# openssl 1.0.2j
# readline 6.2
# sqlite 3.13.0
# tk 8.5.13
# xz 5.0.5
# zlib 1.2.8
# libyaml 0.2.2
# tk 8.5.13
# ncurses 6.0
# readline 6.2
# libc 2.23
# ld 2.23.2
# gcc 4.8.5
# glibc 2.23
# libffi 3.2.1
# libgcc 4.8.5
# libstdc++ 4.8.5
# libtinfo 6.0
# libz 1.2.8
# openssl 1.0.2j
# readline 6.2
# sqlite 3.13.0
# tk 8.5.13
# xz 5.0.5
# zlib 1.2.8
# libyaml 0.2.2
# tk 8.5.13
# ncurses 6.0
# readline 6.2
# libc 2.23
# ld 2.23.2
# gcc 4.8.5
# glibc 2.23
# libffi 3.2.1
# libgcc 4.8.5
# libstdc++ 4.8.5
# libtinfo 6.0
# libz 1.2.8
# openssl 1.0.2j
# readline 6.2
# sqlite 3.13.0
# tk 8.5.13
# xz 5.0.5
# zlib 1.2.8
# libyaml 0.2.2
# tk 8.5.13
# ncurses 6.0
# readline 6.2
#
```

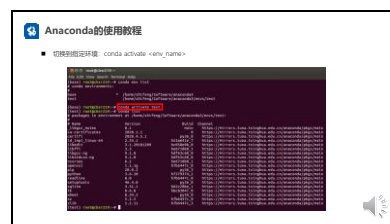
幻灯片 15

[illegible]

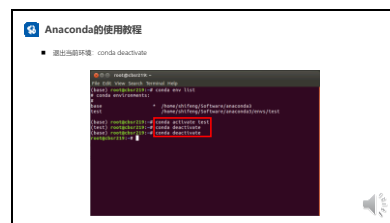
幻灯片 16



幻灯片 17



幻灯片 18



幻灯片 19

[illegible]

幻灯片 20

[illegible]

幻灯片 21

## 1. Anaconda 的使用教程

- 创建 detectron2 环境

**Requirements**

- Linux or macOS with Python > 3.5
- PyTorch > 1.4
- An NVIDIA card installed for PyTorch installation. You can install them together at [pytorch.org](https://pytorch.org) to make sure of this.
- OpenCL optional, needed for some visualization
- Some extras (e.g. Jupyter, Colab, etc.) listed in [github.com/facebookresearch/detectron2/blob/master/INSTALL.md](https://github.com/facebookresearch/detectron2/blob/master/INSTALL.md)

<https://github.com/facebookresearch/detectron2/blob/master/INSTALL.md>

幻灯片 22



- conda create --name detectron2 python=3.6
- conda activate detectron2

幻灯片 23



幻灯片 24



- conda install pytorch torchvision cudatoolkit=10.1 -c pytorch



幻灯片 25



幻灯片 26



- conda install --channel <https://conda.anaconda.org/menpo> opencv3

幻灯片 27



- pip install cython; pip install -U 'git+https://github.com/cocodataset/cocoapi.git#subdirectory=PythonAPI'

幻灯片 28

14

Anaconda的使用教程

■ 创建detection2环境

✓ catboost

✓ cython

✓ opencl

✓ pycocotools

✓ pytorch

✓ torchvision



幻灯片 29

14

物体检测环境配置

  
环境管理软件Anaconda

  
物体检测平台Detectron2

  
代码调试软件PyCharm

幻灯片 30

14

物体检测平台对比

■ 现有的物体检测平台汇总

平台名称	Detection	mmdetection	maskrcnn-benchmark	simpledet	detection2
发布时间	2018年12月2日	2018年6月22日	2018年10月25日	2018年12月9日	2019年10月11日
维护团队	FAIR	清华大学MLab	FAIR	图森	FAIR
深度学习框架	Caffe2	PyTorch	PyTorch	MXNet	PyTorch
1s+训练	23.3s	9.6s	7.6s	2.6s	10.4s
训练状态	已停止维护	维护中	已停止维护	维护中	维护中
其他点	第一个, Caffe2	基于PyTorch	官方, PyTorch	MXNet	官方, PyTorch

幻灯片 31

物体检测平台对比

■ 支持最新过图物体检测平台 (绿色) 有经过, 红色 无经过, 黄色 有没测试过

平台名称	Detection	mmdetection	mmdetection-benchmark	simpledet	detectron2
开源时间	2018年10月24日	2018年9月24日	2018年10月29日	2018年1月29日	2019年10月11日
维护团队	FAIR	清华大学MMLab	FAIR	匿名	FAIR
深度学习框架	Caffe2	PyTorch	PyTorch	MXNet	PyTorch
GPU性能	23.3s	9.6s	7.6s	2.6s	10.4s
当前状态	已停止维护	维护中	已停止维护	维护中	维护中
来源	第一个, Caffe2	基于PyTorch	官方, PyTorch	MXNet	官方, PyTorch

幻灯片 32

物体检测平台对比

■ mmdetection与detectron2比较

平台名称	mmdetection	detectron2
开源时间	2018年8月22日	2019年10月11日
维护团队	清华大学MMLab	FAIR
深度学习框架	PyTorch	PyTorch
GPU性能	9.6s	10.4s
当前状态	维护中	维护中
来源	基于PyTorch	官方, PyTorch

■ 性能数据: mmdetection维护安全, detectron2目前支持  
最主要的检测算法

■ 维护团队: mmdetection团队多, 模块化程度更高,  
detectron2团队少, 模块化程度一般;  
detectron2团队少, 模块化程度一般

■ PyTorch生态: detectron2维护团队, 官方, PyTorch也  
是FAIR开发的

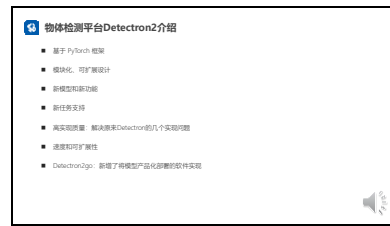
幻灯片 33

物体检测平台Detectron2介绍

- Detection -> mmdetection -> Detectron2
- Detectron2支持物体检测, 实例分割, 姿态估计, 语义分割, 全景分割等



幻灯片 34



幻灯片 35



<https://github.com/facebookresearch/detectron2>

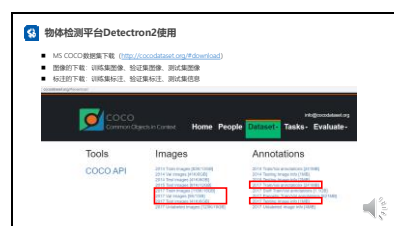
幻灯片 36



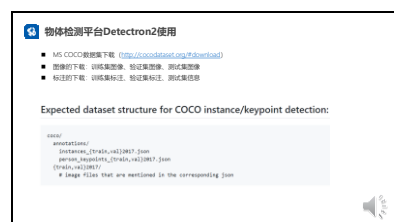
## 幻灯片 37



## 幻灯片 38



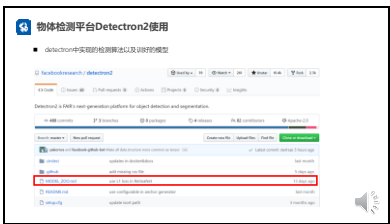
## 幻灯片 39



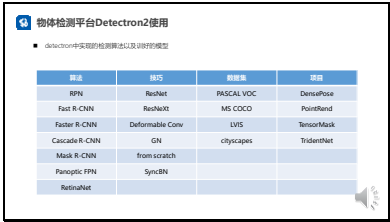
幻灯片 40



幻灯片 41



幻灯片 42



幻灯片 43

## 物体检测平台Detron2使用

■ MS COCO物体检测: Faster R-CNN

Name	arch	lr	train time (s/iter)	inference time (s/iter)	train mem (GB)	box AP	model id	download
R101-C1	1x	0.01	0.102	4.8	30.7	10.070680	model   metrics	
R101-C1S	1x	0.001	0.088	1.8	17.3	10.784529	model   metrics	
R101-C1S	1x	0.001	0.086	1.8	17.9	10.921784	model   metrics	
R101-C1	3x	0.001	0.104	4.8	30.6	10.784529	model   metrics	
R101-C1S	3x	0.01	0.070	1.8	38.0	10.784529	model   metrics	
R101-C1S	3x	0.01	0.088	1.8	40.2	10.784529	model   metrics	
R101-C1S	3x	0.01	0.100	1.8	40.7	10.921784	model   metrics	
R101-C1S	3x	0.02	0.086	4.7	40.8	10.921784	model   metrics	
R101-C1S	3x	0.200	0.091	4.7	40.8	10.921784	model   metrics	
R101-C1S	3x	0.001	0.086	4.7	40.8	10.921784	model   metrics	



幻灯片 44

## 物体检测平台Detron2使用

■ MS COCO实例分割: Mask R-CNN

Name	arch	lr	train time (s/iter)	inference time (s/iter)	train mem (GB)	box AP	mask AP	model id	download
R101-C1	1x	0.001	0.100	1.2	30.0	10.070680	10.070680	model   metrics	
R101-C1S	1x	0.01	0.076	6.5	38.0	10.070680	10.070680	model   metrics	
R101-C1S	1x	0.001	0.080	1.8	38.0	10.070680	10.070680	model   metrics	
R101-C1	3x	0.01	0.111	1.2	38.0	10.070680	10.070680	model   metrics	
R101-C1S	3x	0.01	0.076	6.5	40.0	10.070680	10.070680	model   metrics	
R101-C1S	3x	0.001	0.080	1.8	40.0	10.070680	10.070680	model   metrics	
R101-C1S	3x	0.001	0.080	6.5	40.0	10.070680	10.070680	model   metrics	
R101-C1S	3x	0.001	0.080	7.0	40.0	10.070680	10.070680	model   metrics	
R101-C1S	3x	0.001	0.080	6.5	40.0	10.070680	10.070680	model   metrics	
R101-C1S	3x	0.001	0.080	7.2	40.0	10.070680	10.070680	model   metrics	

幻灯片 45

物体检测平台Detron2使用

■ MS COCO物体检测: RetinaNet

Name	lr	train time (s/iter)	inference time (s/iter)	train mem (GB)	box AP	model id	download
R101	1x	0.205	0.056	4.1	37.4	190397773	model   metrics
R101	3x	0.205	0.056	4.1	38.7	190397829	model   metrics
R101	3x	0.291	0.060	5.2	40.4	190397697	model   metrics

幻灯片 46



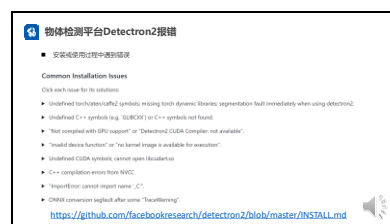
- `conda activate detectron2`
- `cd ./detectron2/demo`
- `python demo.py --config-file ../configs/COCO-Detection/retinanet_R_50_FPN_1x.yaml --input 000000000785.jpg [--other-options] --opts MODEL.WEIGHTS detectron2://COCO-Detection/retinanet_R_50_FPN_1x/190397773/model_final_bfca0b.pkl`

幻灯片 47



- `conda activate detectron2`
- `cd ./detectron2`
- `python ./tools/train_net.py --config-file ../configs/COCO-Detection/retinanet_R_50_FPN_1x.yaml --eval-only MODEL.WEIGHTS detectron2://COCO-Detection/retinanet_R_50_FPN_1x/190397773/model_final_bfca0b.pkl`

幻灯片 48





幻灯片 49



幻灯片 50



幻灯片 51



幻灯片 52



幻灯片 53



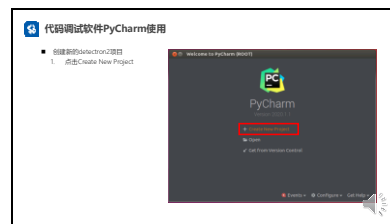
幻灯片 54



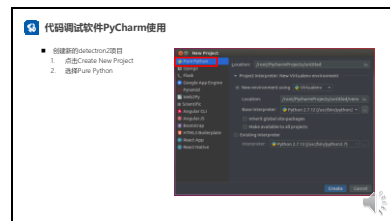
幻灯片 55



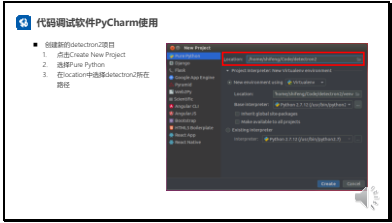
幻灯片 56



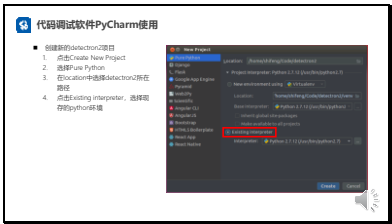
幻灯片 57



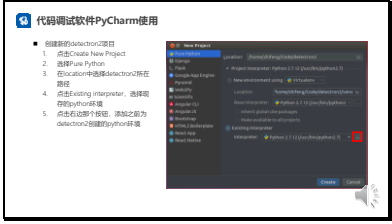
幻灯片 58



幻灯片 59



幻灯片 60



幻灯片 61

代码调试软件PyCharm使用

创建新的datac2项目

1. 点击Create New Project

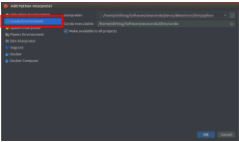
2. 选择Pure Python

3. 在Location中选择datac2所在路径

4. 点击Existing Interpreter，选择数据科学Python环境

5. 点击右侧那个按钮，添加之

6. 选择Conda Environment



幻灯片 62

代码调试软件PyCharm使用

创建新的datac2项目

1. 点击Create New Project

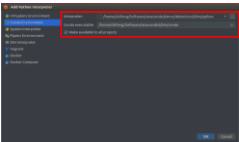
2. 选择Pure Python

3. 在Location中选择datac2所在路径

4. 点击Existing Interpreter，选择数据科学Python环境

5. 点击右侧那个按钮，添加之

6. 选择Conda Environment



幻灯片 63

代码调试软件PyCharm使用

创建新的datac2项目

1. 点击Create New Project

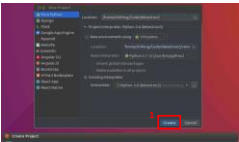
2. 选择Pure Python

3. 在Location中选择datac2所在路径

4. 点击Existing Interpreter，选择数据科学Python环境

5. 点击右侧那个按钮，添加之前为datac2创建的Python环境

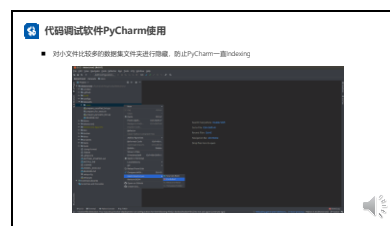
6. 选择Conda Environment



## 幻灯片 64



## 幻灯片 65



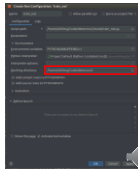
## 幻灯片 66



## 幻灯片 67

### 代码调试软件PyCharm使用

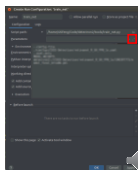
- 调试测试代码
- 1. 右键click 下面的train\_net.py文件，选择 create train\_net
- 2. 把working directory设置到detectron2项目 按下



## 幻灯片 68

### 代码调试软件PyCharm使用

- 调试测试代码
- 1. 右键click 下面的train\_net.py文件，选择 create train\_net
- 2. 把working directory设置到detectron2项目 按下
- 3. 点击右侧栏中的按钮，添加运行所需需要的参数

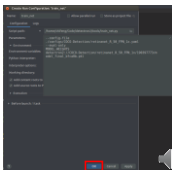


```
--config-file ./configs/COCO-  
Detection/retinanet_R_50_FPN_  
1x.yaml --eval-only  
MODEL.WEIGHTS  
detectron2://COCO-  
Detection/retinanet_R_50_FPN_  
1x/190397773/model_final_bfca  
0b.pkl
```

## 幻灯片 69

### 代码调试软件PyCharm使用

- 调试测试代码
- 1. 右键click 下面的train\_net.py文件，选择 create train\_net
- 2. 把working directory设置到detectron2项目 按下
- 3. 点击右侧栏中的按钮，添加运行所需需要的参数
- 4. 点击+号添加快速启动的界面



## 幻灯片 70

4 代码调试软件PyCharm使用

■ 调试测试代码

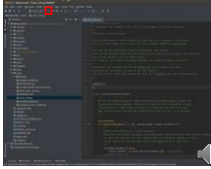
1. 创建如下代码train\_net.py文件，选择create train\_net

2. 把working directory设置到testchoc2项目下

3. 点击右侧窗口中的按钮，添加运行所需需要的参数

4. 点击运行按钮运行测试代码

5. 点击三角按钮开始调试测试代码



## 幻灯片 71

4 代码调试软件PyCharm使用

■ 调试测试代码

1. 创建如下代码train\_net.py文件，选择create train\_net

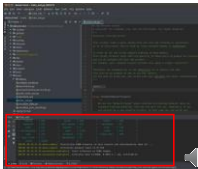
2. 把working directory设置到testchoc2项目下

3. 点击右侧窗口中的按钮，添加运行所需需要的参数

4. 点击运行按钮运行测试代码

5. 点击三角按钮开始调试测试代码

6. 测试代码运行成功，显示结果



## 幻灯片 72

4 代码调试软件PyCharm使用

■ 调试测试代码

1. 创建如下代码train\_net.py文件，选择create train\_net

2. 把working directory设置到testchoc2项目下

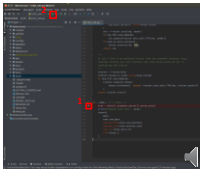
3. 点击右侧窗口中的按钮，添加运行所需需要的参数

4. 点击运行按钮运行测试代码

5. 点击三角按钮开始调试测试代码

6. 测试代码运行成功，显示结果

7. 在train\_net.py文件中设置断点，再点击三角按钮开始调试测试代码





代码调试软件PyCharm使用

■ 调试测试代码

1. 新建with 下面的train\_net.py文件，选择 create train\_net

2. 把working\_directory设置到testchoco2项目 目录下

3. 点击红色框中的按钮，添加运行所需需要的参数

4. 点击OK返回快捷的界面

5. 点击运行按钮运行测试代码

6. 测试代码开始运行，显示结果

7. 在train\_net.py文件中可以生成红色的断点，断点处会立即开始调试测试代码

8. 进入调试状态，可以多调试测试代码

