

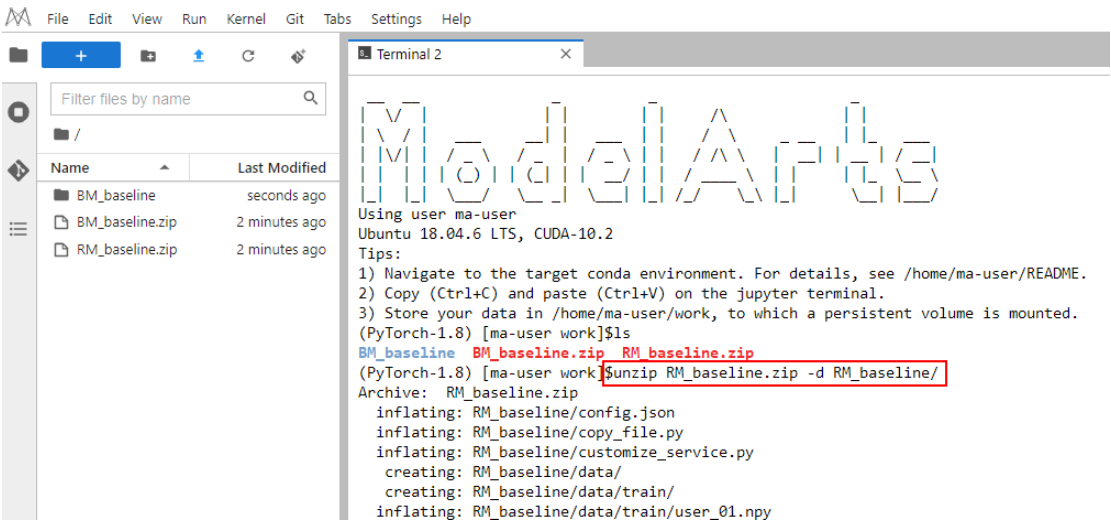
本文档旨在帮助选手快速上手比赛，指引大家使用华为云平台进行模型的训练，生成可提交到比赛的模型包。

1 获取竞赛数据集和示例代码

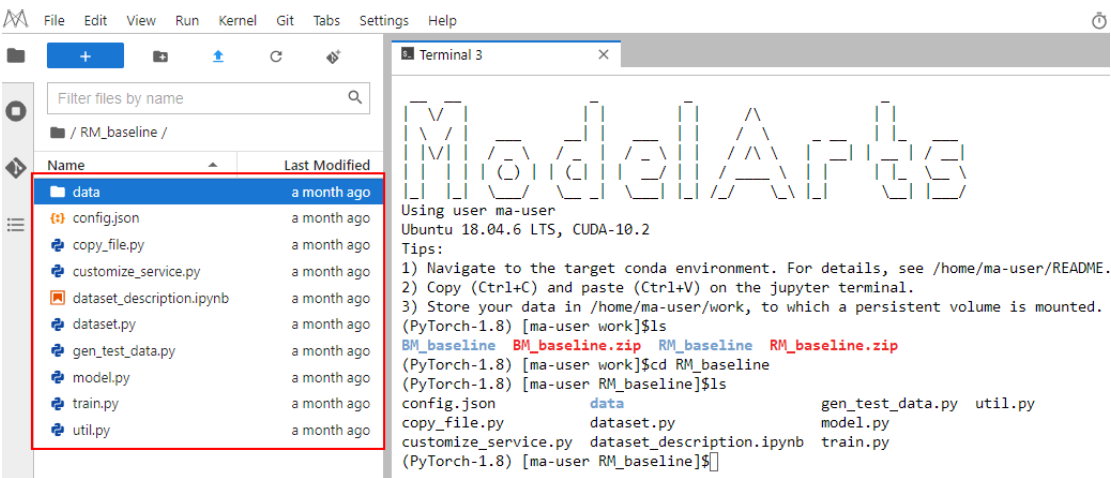
下载本赛题的训练数据集和示例代码（RM_baseline.zip压缩文件）到本地。阅读《华为云大赛平台使用说明文档》第4章，按照文档说明将压缩包RM_baseline.zip上传到ModelArts中新建的Notebook。

2 数据集说明

使用unzip命令解压缩RM_baseline.zip，如下图所示：

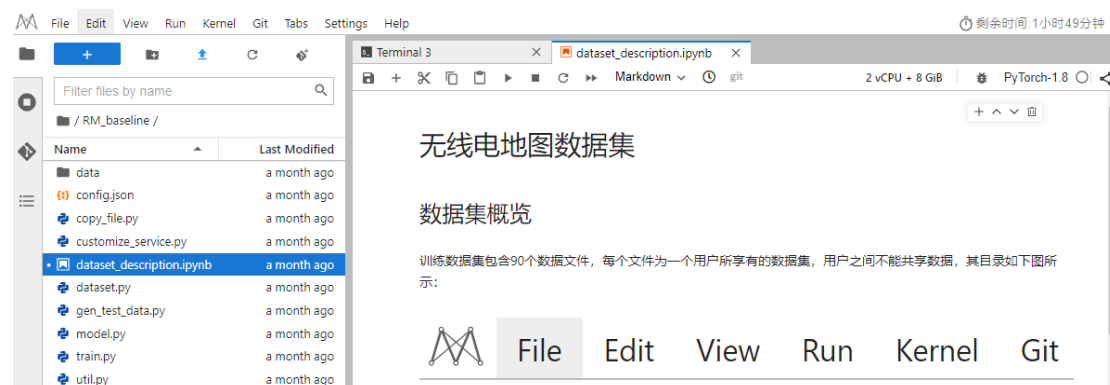


接着cd RM_baseline进入该文件夹，可以得到如下文件：



其中，data文件夹即为我们的训练数据集。对于该数据集的详细介绍，可以参考

dataset_description.ipynb。单击该文件，即可查看训练数据集的介绍以及如何训练数据集的读取，如下图所示：

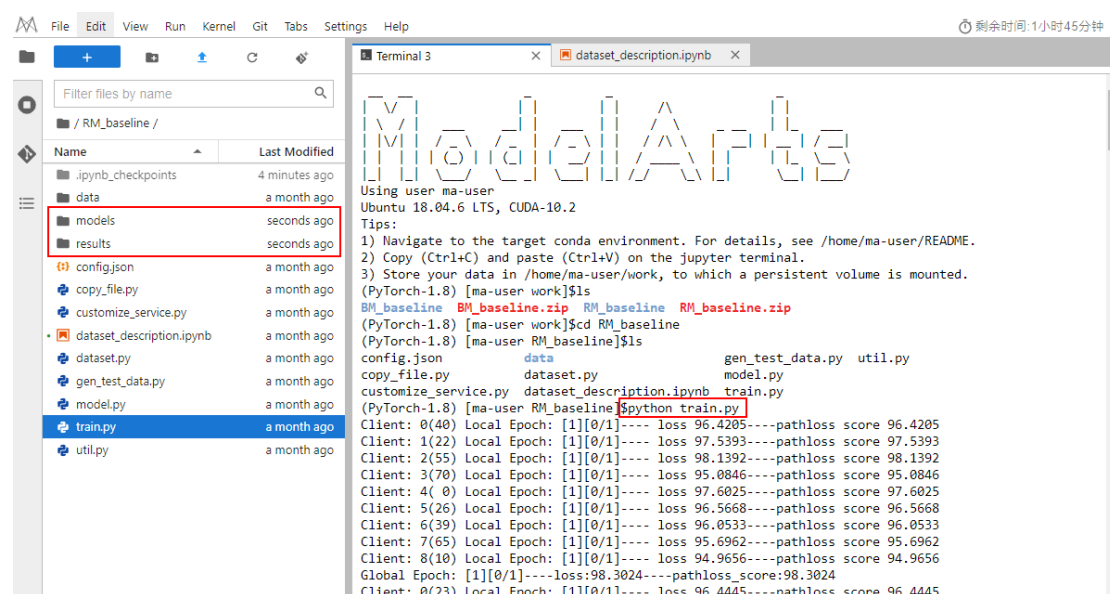


3 样例代码训练说明

在上述训练数据集的基础上，我们可以进行模型的训练，训练模型相关的文件包含：

- **dataset.py**: 读取数据集，选手可以在这对原始数据进行特征工程
- **model.py**: 模型的定义，选手可以在这设计更加复杂的模型
- **train.py**: 模型的训练，选手可以在这设置各种超参数以及损失函数设计
- **util.py**: 一些定义好的常用函数集合，选手可以在这定义一些常用的函数

训练模型时，只需在终端输入命令`python train.py`即可。注意，可能会出现缺少包的问题，我们可以通过`pip`进行安装，然后再次运行`train.py`文件，如下图所示：



等待程序运行完毕，会在目录中多出两个文件夹**models**和**results**（需要刷新后出现），分别

存储训练过程中的模型(.pth文件)以及训练过程中pathloss、uplink和downlink开销(pathloss.npy、ul_commCost.npy和dl_commCost.npy)。注意：上述示例代码仅作参考，选手可以选择按照自己的方式来训练模型，只需要最后提供模型文件model.pth以及模型定义文件model.py。

4 配置文件说明

最后选手需要提交到比赛的模型包，包含模型文件model.pth、模型定义文件model.py、配置文件config.json、customize_service.py文件。在示例代码中，我们也给出了这两个配置文件。根据选手提供的模型文件model.pth，可能需要进行如下修改：

4.1 config.json 的修改

当选手调用一些包时，可能需要在config.json文件中的dependencies一项中添加包，添加包的方式可以举一反三，如下图所示：



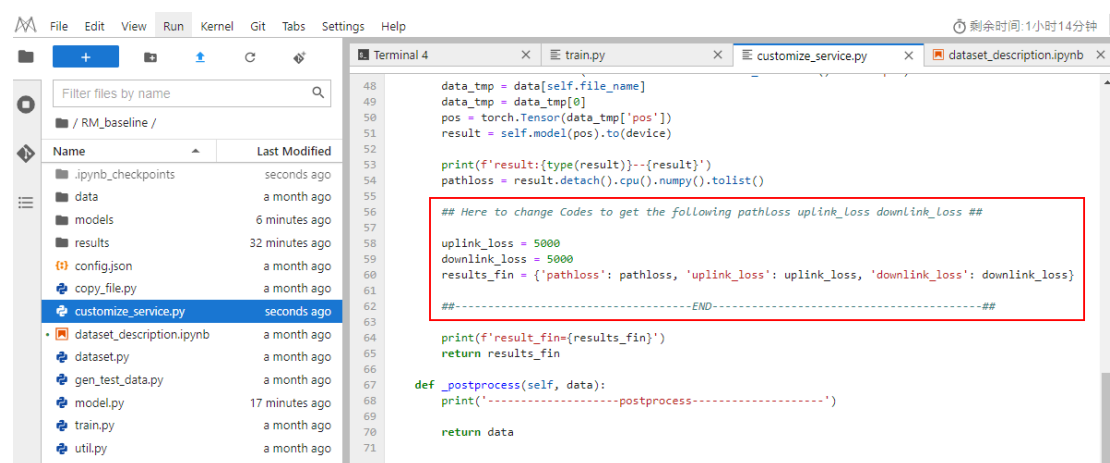
```
46     }
47   }],
48   "dependencies": [
49     {
50       "installer": "pip",
51       "packages": [
52         {
53           "restraint": "ATLEAST",
54           "package_version": "1.15.0",
55           "package_name": "numpy"
56         }, {
57           "package_name": "pillow"
58         }, {
59           "package_name": "tqdm"
60         }, {
61           "package_name": "matplotlib"
62         }, {
63           "package_name": "einops"
64         }
65       ]
66     }
67   ]
68 }
69 }
70 }
```

其中，package_version和restraint不是必选项，当对包版本有明确要求时，可以在

package_version中写出版本号。restraint为版本限制条件,可选EXACT/ATLEAST/ATMOST,分别表示安装给定版本、不小于给定版本以及不大于给定版本。配置文件的详细说明可以参考: <https://support.huaweicloud.com/inference-modelarts/inference-modelarts0056.html#inference-modelarts-0056>

4.2 customize_service.py 修改

在赛题二中,为统一起见,我们规定模型最终的结果为信号强度、上行通信开销和下行通信开销,分别用pathloss、uplink_loss和downlink_loss表示。当选手自定义的模型输出不是这种格式时,需要进行修改:

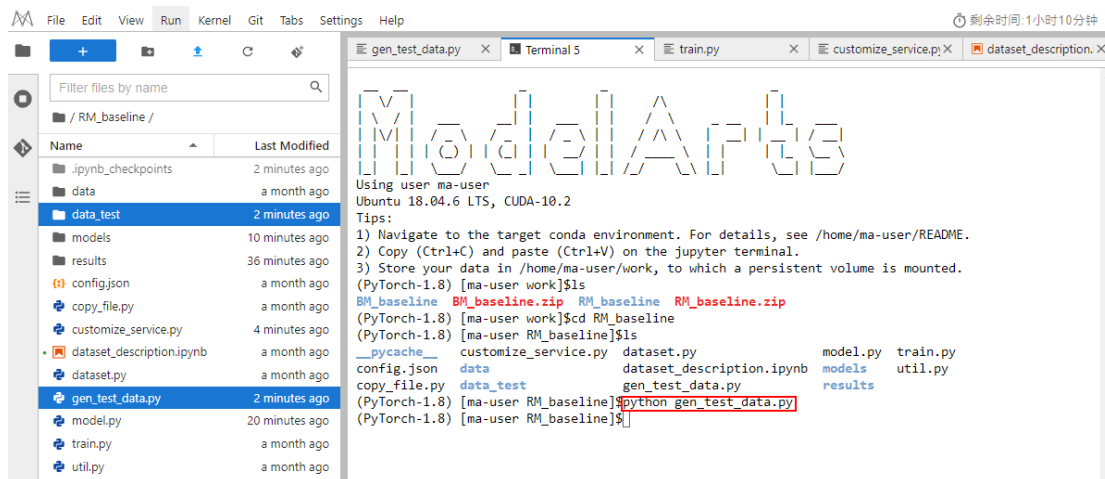


```
48 data_tmp = data[self.file_name]
49 data_tmp = data_tmp[0]
50 pos = torch.Tensor(data_tmp['pos'])
51 result = self.model(pos).to(device)
52
53 print(f'result:{type(result)}--{result}')
54 pathloss = result.detach().cpu().numpy().tolist()
55
56 ## Here to change Codes to get the following pathloss uplink_loss downlink_loss ##
57
58 uplink_loss = 5000
59 downlink_loss = 5000
60 results_fin = {'pathloss': pathloss, 'uplink_loss': uplink_loss, 'downlink_loss': downlink_loss}
61
62 #-----END-----##
63
64 print(f'result_fin={results_fin}')
65 return results_fin
66
67 def _postprocess(self, data):
68     print('-----postprocess-----')
69
70     return data
71
```

在示例代码(model.py)中,模型的输出pathloss为预测的信号强度,上下行的通信开销需要选手自行统计给定。为此,我们需要增加上述红色方框的代码来得到最终的结果result_fin。

5 生成测试文件

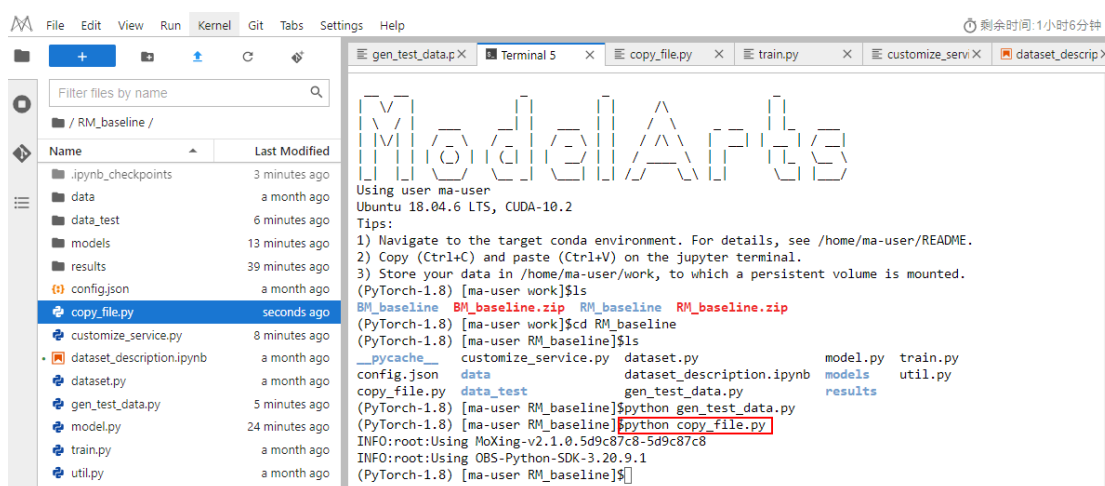
选手无法得到测试数据集,但是选手可以使用训练数据集来进行模型的在线测试。在《华为云大赛平台使用说明文档》中我们提及的模型在线服务测试,需要上传单个测试文件(单个样例)来进行模型包的测试,这里的测试文件需要写成json格式。为简便起见,我们提供了使用训练集中的数据来生成测试文件。选手可以直接运行脚本文件gen_test_data.py来得到测试文件集合data_test,如下图所示:



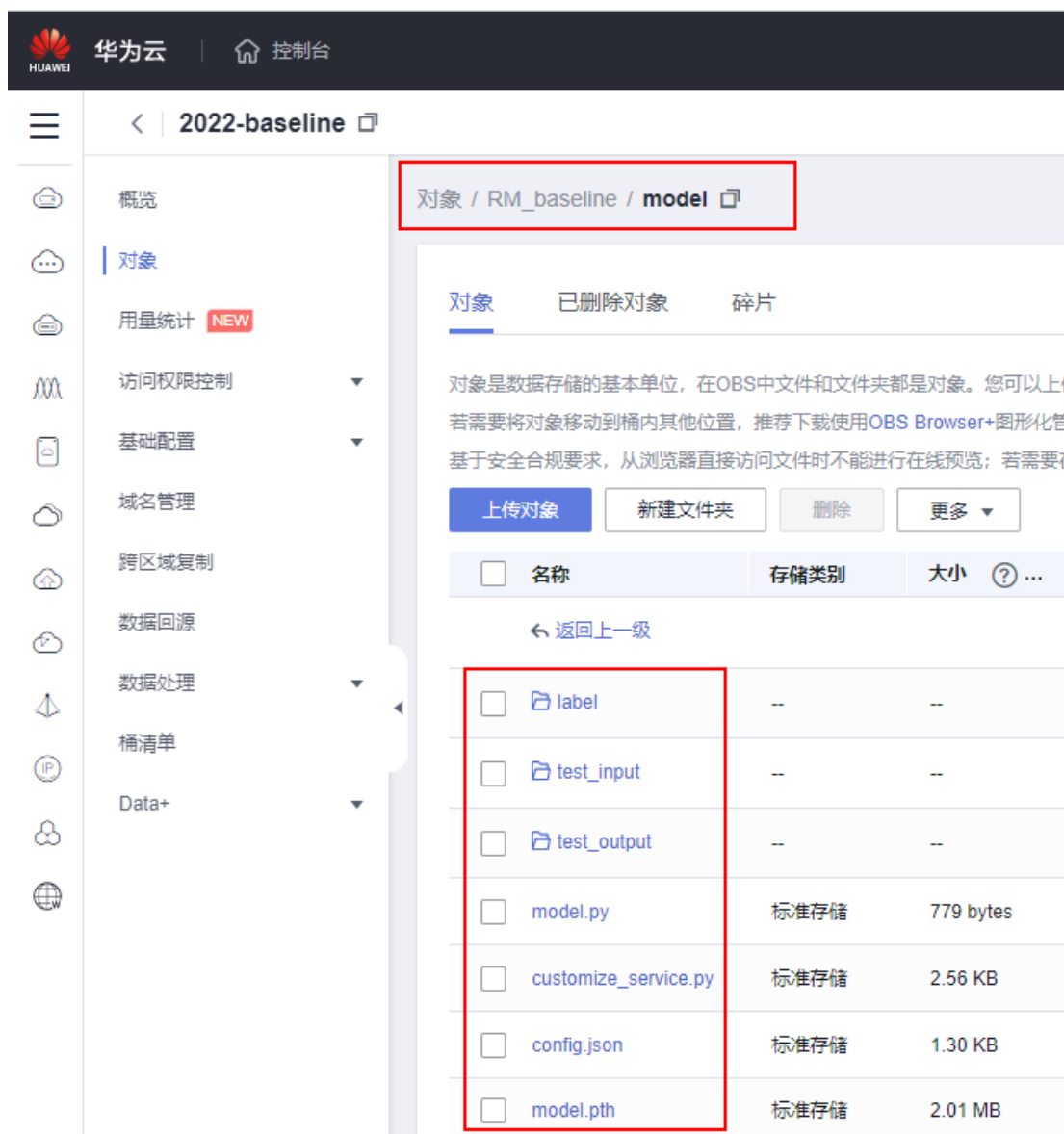
在data_test文件夹中包含三个子文件夹label、test_input和test_output。其中，label中的每个文件夹为一个样例的信号强度pathloss、上行通信开销uplink_cost和下行通信开销downlink_cost，test_input为测试文件，每个文件为一个测试样例（用户的位置信息，json格式数据）。test_output为空文件夹，用于保存在线服务时模型的输出结果。

6 生成模型包

至此，我们已经在Notebook中完成所有文件的生成。接下来，我们需要将这些文件上传到OBS桶中构成模型包，用于后续的在线服务以及发布到比赛中。首先，我们在OBS中新建一个桶（命名为2022-baseline，选手可自定义），接着在该桶中新建目录RM_baseline和其子文件夹model（model文件夹名不可修改）。最后，只需要在Notebook的终端中运行如下命令即可：



最后，我们就可以在OBS桶中得到我们的模型包：



注意：copy_file.py文件中的地址需要改为选手自己的OBS桶地址。

7 提交模型包

阅读《华为云大赛平台使用说明文档》第5章和第6章，将模型提交给大赛平台，用于自动判分。需等待一定时间（判分系统进行判分需一定时间，运行时长与选手提交的模型有关），判分系统完成判分后，可在大赛平台“提交作品”页面查看得分。