

OpenKE 环境搭建及操作教程

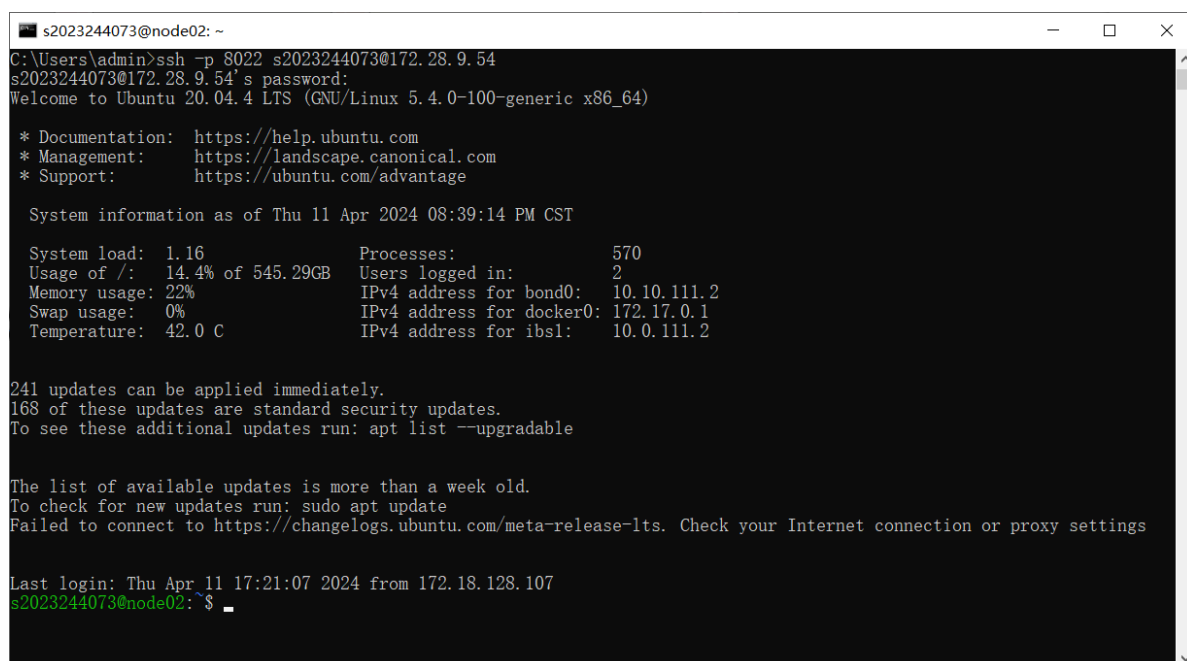
OpenKE 是 THUNLP 基于 TensorFlow、PyTorch 开发的用于将知识图谱嵌入到低维连续向量空间进行表示的开源框架。OpenKE需要在Linux服务器环境下运行，其中提供了快速且稳定的各类接口，也实现了诸多经典的知识表示学习模型。该框架易于扩展，基于框架设计新的知识表示模型也十分的方便。

一、智算集群环境配置

智算集群地址为172.28.9.54，端口为8022，需要在校园网环境下登入，也可以使用天津大学 VPN服务从校外访问集群。

登陆集群可使用 Putty、SecureCRT、SSHClient 等软件，在安装有SSH组件的命令行中，也可使用如下命令进行登陆：

```
ssh -p 8022 username@172.28.9.54
```



```
s2023244073@node02: ~
C:\Users\admin>ssh -p 8022 s2023244073@172.28.9.54
s2023244073@172.28.9.54's password:
Welcome to Ubuntu 20.04.4 LTS (GNU/Linux 5.4.0-100-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Thu 11 Apr 2024 08:39:14 PM CST

System load:  1.16               Processes:    570
Usage of /:   14.4% of 545.29GB  Users logged in: 2
Memory usage: 22%               IPv4 address for bond0: 10.10.111.2
Swap usage:   0%                 IPv4 address for docker0: 172.17.0.1
Temperature: 42.0 C              IPv4 address for ibs1: 10.0.111.2

241 updates can be applied immediately.
168 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings

Last login: Thu Apr 11 17:21:07 2024 from 172.18.128.107
s2023244073@node02:~$
```

系统在集群后端共享存储上为每个账户创建了自己的用户目录，并挂载到全部节点，方便在计算任务执行过程中读写数据，可以使用专用文件传输软件或远程访问客户端软件自带文件传输模块进行文件传输，如 FileZilla, Xftp, Putty 的 Psftp 等。在安装有 SSH 组件的命令行中，也可使用如下命令进行文件传输：

```
scp -P 8022 files username@172.28.9.54:filepath
```

在本教程中使用 Xftp 进行文件传输，端口属性如下图所示，填入用户名与密码即可连接

由于流量受限，集群默认不开放外网数据访问，需要进行如下操作才可以使用 pip 更新 python 软件包。集群配置了 mirrors.aliyun.com 的名称映射，所以目前只支持阿里软件源，具体方法如下：

1. 确定代理端口，可选较大端口号（1024-49151），以下示例中选择 35800
2. 在本机的命令行终端中使用 SSH 命令建立端口映射：

```
ssh -p 8022 username@172.28.9.54 -NR 35800:mirrors.aliyun.com:80
```

其中，35800 为选择的代理端口（端口冲突时需要重新选择端口），mirrors.aliyun.com:80 为阿里软件源的域名和端口，172.28.9.54 和 8022 为集群 ssh 服务的 IP 和端口号，username 为用户名。运行此命令后输入密码即可，此命令运行成功后，不返回命令行，也没有任何消息，会一直等待，如下图所示：

```
命令提示符 - ssh -p 8022 s2023244073@172.28.9.54 -NR 35800:mirrors.aliyun.com:80
C:\Users\admin>ssh -p 8022 s2023244073@172.28.9.54 -NR 35800:mirrors.aliyun.com:80
s2023244073@172.28.9.54's password:
```

3. 登陆集群，在当前用户下创建 ~/.pip/pip.conf，内容如下：

```
[global]
index-url=http://mirrors.aliyun.com:35800/pypi/simple/

[install]
trusted-host=mirrors.aliyun.com
```

之后可以使用 pip 命令进行 python 软件包的安装和更新。完成后，可退出一直等待的代理服务。再次使用时，如果端口不变，运行第 2 步的命令即可。如果端口改变，还需重新配置第 3 步的设置。

二、Python 软件包安装

OpenKE使用到的软件包主要包括 pytorch、sklearn 以及 tqdm，**推荐根据附录在虚拟环境中进行操作，避免用户之间可能发生的冲突**

sklearn 和 tqdm 可以通过以下命令安装：

```
pip install scikit-learn
pip install tqdm
```

如果直接使用 pip 通过国内源安装 pytorch 可能会替换为CPU版本，为了利用集群提供的 GPU，可以通过以下方式安装 GPU 版本的 pytorch。

首先，在命令行输入 nvidia-smi 命令，得到 cuda 版本为12.0

```
s2023244073@node02:~$ nvidia-smi
Fri Apr 12 13:07:46 2024
```

NVIDIA-SMI 525.60.13		Driver Version: 525.60.13		CUDA Version: 12.0	
GPU	Name	Persistence-M	Bus-Id	Disp. A	Volatile Uncorr. ECC
Fan	Temp	Perf	Memory-Usage	Memory-Usage	GPU-Util Compute M.
		Pwr:Usage/Cap			MIG M.
0	Tesla V100-PCIE...	Off	00000000:D8:00:0	Off	0
N/A	38C	P0 26W / 250W	0MiB / 16384MiB		Default N/A

Processes:						
GPU	GI	CI	PID	Type	Process name	GPU Memory Usage
	ID	ID				
No running processes found						

查看 pytorch 官网 (<https://pytorch.org/get-started/previous-versions/>) 可知，最高可安装 11.8 版本，同时我们能得到对应的三个组件的版本对应关系

Linux and Windows

```
# CUDA 11.8
conda install pytorch==2.2.1 torchvision==0.17.1 torchaudio==2.2.1 pytorch-cuda=11.8 -c pytorch -c nvidia
# CUDA 12.1
conda install pytorch==2.2.1 torchvision==0.17.1 torchaudio==2.2.1 pytorch-cuda=12.1 -c pytorch -c nvidia
# CPU Only
conda install pytorch==2.2.1 torchvision==0.17.1 torchaudio==2.2.1 cpuonly -c pytorch
```

此时，我们还需要通过命令行查看 python 版本，才能最终确定需要下载的文件

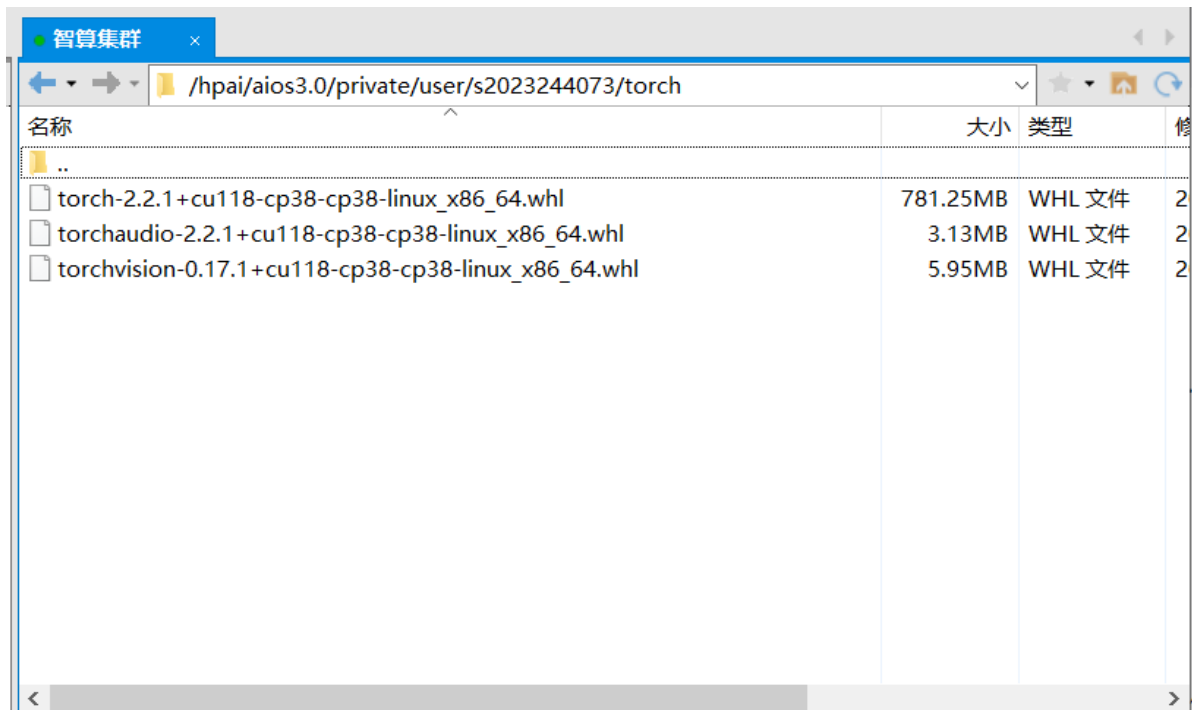
```
s2023244073@node02:~$ python3 --version
Python 3.8.10
```

打开网址<https://download.pytorch.org/whl/cu118>

首先选择torch, 搜索 cu118-cp38-cp38-linux, 这里 cu118 指我们下载的 CUDA 11.8 版本, cp38-cp38 则指 Python 版本是 3.8, 找到 2.2.1 版本的 pytorch, 点击即可下载

```
torch-2.2.1+cu118-cp312-cp312-win_amd64.whl
torch-2.2.1+cu118-cp38-cp38-linux_x86_64.whl
torch-2.2.1+cu118-cp38-cp38-win_amd64.whl
torch-2.2.1+cu118-cp38-cp38-linux_x86_64.whl
```

同样, 我们可以在网站下载相应版本的 torchvision 和 torchaudio, 最终在服务器中建立 torch 文件夹并将上传三个文件



在命令行依次输入以下命令即可完成安装, 注意替换为自己文件的储存路径

```
pip install torch/torch-2.2.1+cu118-cp38-cp38-linux_x86_64.whl
pip install torch/torchaudio-2.2.1+cu118-cp38-cp38-linux_x86_64.whl
pip install torch/torchvision-0.17.1+cu118-cp38-cp38-linux_x86_64.whl
```

通过以下命令即可查看pytorch版本以及GPU是否可用

```
s2023244073@node02:~$ python3
Python 3.8.10 (default, Mar 13 2023, 10:26:41)
[GCC 9.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import torch
>>> print(torch.__version__)
2.2.1+cu118
>>> print(torch.version.cuda)
11.8
>>> print(torch.cuda.is_available())
True
>>> exit()
s2023244073@node02:~$
```

三、OpenKE 安装及运行

OpenKE可以通过 github 官方页面进行下载 (<https://github.com/thunlp/OpenKE/>) , 此外OpenKE-PyTorch 版的相关文件也已经上传到知谱空间。

进入目录:

```
cd OpenKE-OpenKE-PyTorch/openke/
```

编译C++文件

```
bash make.sh
```

回到OpenKE目录并创建checkpoint文件夹

```
cd ..  
mkdir checkpoint
```

我们选择使用 train_transe_FB15K237.py 作为示例, 该程序是做链接预测 (Link Prediction) 任务, 即已有三元组 (h, r, t) 中去掉头实体 h 或尾实体 t 预测的准确度, 使用了 TransE 算法和 FB15K237 数据集, 可以根据实际条件修改训练参数

```
# train the model  
trainer = Trainer(model = model, data_loader = train_dataloader, train_times = 1000, alpha = 1.0, use_gpu = True)  
trainer.run()  
transe.save_checkpoint('./checkpoint/transe.ckpt')  
  
# test the model  
transe.load_checkpoint('./checkpoint/transe.ckpt')  
tester = Tester(model = transe, data_loader = test_dataloader, use_gpu = True)  
tester.run_link_prediction(type_constrain = False)
```

拷贝程序

```
cp examples/train_transe_FB15K237.py ./
```

执行程序, 开始训练

```
python3 train_transe_FB15K237.py
```

```
s2023244073@node02:~/OpenKE-OpenKE-PyTorch$ python3 train_transe_FB15K237.py  
Input Files Path : ./benchmarks/FB15K237/  
The toolkit is importing datasets.  
The total of relations is 237.  
The total of entities is 14541.  
The total of train triples is 272115.  
Input Files Path : ./benchmarks/FB15K237/  
The total of test triples is 20466.  
The total of valid triples is 17535.  
Finish initializing..  
Epoch 17 | loss: 5.597768: 2% | 18/1000 [00:33<29:58, 1.83s/it]
```

等待程序运行完毕即可得到结果

```

s2023244073@node02:~/OpenKE-OpenKE-PyTorch$ python3 train_transe_FB15K237.py
Input Files Path : ./benchmarks/FB15K237/
The toolkit is importing datasets.
The total of relations is 237.
The total of entities is 14541.
The total of train triples is 272115.
Input Files Path : ./benchmarks/FB15K237/
The total of test triples is 20466.
The total of valid triples is 17535.
Finish initializing...
Epoch 999 | loss: 2.373240: 100%| 1000/1000 [30:56<00:00, 1.86s/it]
100%| 20466/20466 [00:25<00:00, 799.65it/s]
no type constraint results:
metric:      MRR          MR          hit@10      hit@3       hit@1
l(raw):      0.088456      570.228516  0.202238   0.084433    0.033666
r(raw):      0.250027      165.870132  0.438239   0.273380    0.157090
averaged(raw): 0.169241      368.049316  0.320238   0.178906    0.095378

l(filter):    0.189245      315.479767  0.360647   0.211815    0.104759
r(filter):    0.388160      139.764481  0.592739   0.439607    0.280905
averaged(filter): 0.288703      227.622131  0.476693   0.325711    0.192832
0.476693
0.4766930341720581

```

下图使用了虚拟机 CPU 进行训练，可以发现训练速度的差距十分显著

```

Epoch 49 | loss: 8.460692: 100%| 50/50 [1:33:27<00:00, 112.16s/it]
100%| 3134/3134 [11:04<00:00, 4.72it/s]
no type constraint results:

```

输出结果中的l和r表示left和right，即去掉头实体h和尾实体t的预测结果。raw和filtered分别表示未修复数据和修复了的数据的预测结果。

附：python 纯净环境下虚拟环境配置

安装virtualenv

```
pip install virtualenv
```

在创建虚拟环境前，首先创建一个文件夹用于存储虚拟环境的文件信息，推荐使用学号命名，避免冲突。随后进入到该文件夹下

```
mkdir ke
cd ke
```

使用virtualenv命名来创建虚拟环境，这里可随意对虚拟环境命名

```
virtualenv ke
```

使用source命令激活虚拟环境

```
source ke/bin/activate
```