

mindspore 环境配置

CPU 版本配置

根据对应平台直接使用 pip/conda 安装：

- windows x64, python==3.9, mindspore==2.2.14

```
pip install https://ms-release.obs.cn-north-4.myhuaweicloud.com/2.2.14/MindSpore/cpu/x86_64/mindspore-2.2.14-cp39-cp39-win_amd64.whl --trusted-host ms-release.obs.cn-north-4.myhuaweicloud.com -i https://pypi.tuna.tsinghua.edu.cn/simple
```

- Linux x86_64, python==3.9, mindspore==2.2.14

```
pip install https://ms-release.obs.cn-north-4.myhuaweicloud.com/2.2.14/MindSpore/unified/x86_64/mindspore-2.2.14-cp39-cp39-linux_x86_64.whl --trusted-host ms-release.obs.cn-north-4.myhuaweicloud.com -i https://pypi.tuna.tsinghua.edu.cn/simple
```

如果你使用的是其他的 python 版本，请参考 [mindspore 官网安装指南](#)。

GPU 版本

GPU 版本只支持 Linux, **windows 下只能使用 CPU 版本**。

安装 conda

conda 是一个开源的软件包管理系统和环境管理系统，用于安装多个版本的软件包及其依赖关系，并在它们之间轻松切换。

常见的 conda 的版本有 Anaconda 和 Miniconda，这两者在使用上没有区别。miniconda 只有命令行工具，而 anaconda 会提供一些图形化界面、开发工具等。这里我们使用 miniconda 进行试验。

在 Linux 下安装 miniconda3：

```
mkdir -p ~/miniconda3
wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh -O ~/miniconda3/miniconda.sh
bash ~/miniconda3/miniconda.sh -b -u -p ~/miniconda3
rm -rf ~/miniconda3/miniconda.sh
```

初始化 shell 环境：

```
# 使用 bash
~/miniconda3/bin/conda init bash
```

如果你使用的是 `zsh`，则执行：

```
~/miniconda3/bin/conda init zsh
```

可以使用 `echo $SHELL` 命令查看当前使用的是哪个 shell，例如，若当前使用 bash，则一般情况下，会输出

```
echo $SHELL
# /bin/bash # 输出当前 shell 路径
```

创建 conda 环境

推荐使用 conda 创建一个独立的虚拟环境（可参考 [Linux 下 miniconda 安装](#)），以 python==3.9, cuda==11.6 为例。

```
conda create --name ms python==3.9
conda activate ms

# 安装 cuda
conda install nvidia/label/cuda-11.6.2::cuda

# 安装 cudnn
conda install esri::cudnn

# 安装 mindspore
export MS_VERSION=2.2.14
pip install https://ms-release.obs.cn-north-4.myhuaweicloud.com/${MS_VERSION}/MindSpore/unified/x86_64/mindspore-${MS_VERSION}/-/-cp39-cp39-linux_x86_64.whl --trusted-host ms-release.obs.cn-north-4.myhuaweicloud.com -i https://pypi.tuna.tsinghua.edu.cn/simple
```

使用时，需在 shell 环境中设置 `LD_LIBRARY_PATH` 环境变量：

```
export ENV_PATH=~/.miniconda3/envs/ms
export LD_LIBRARY_PATH=$ENV_PATH/lib:$LD_LIBRARY_PATH
python your_program.py
```

此处将 `ENV_PATH` 设置为你的 conda 环境的路径。

或者，也可以在 python 程序中设置 `LD_LIBRARY_PATH` 变量：

```
import os

# 假设你的用户名为 user，conda 环境路径为 ~/.miniconda3/envs/ms
LIB_PATH = "/home/user/miniconda3/envs/ms/lib"
os.environ["LD_LIBRARY_PATH"] = LIB_PATH + ":" + os.environ["LD_LIBRARY_PATH"]

# 必须在引用 mindspore 库之前设置环境变量
import mindspore as ms
from mindspore import ops

# ...
```

可能遇到的问题

mpi4py 无法安装

```
# 推荐安装 mpi4py==3.0.3
pip install mpi4py==3.0.3
```

安装 mindspore 过程如果遇到 `mpi4py` 无法安装的问题，可以参考以下流程：

1. 报错：gcc 版本过高：

可以通过 conda 安装 gcc-4.8.5：

```
# gcc-4.8.5
conda install -c free gcc
```

或者，也可以通过[编译源码安装 gcc-8](#)。

2. 报错：找不到 `x86_64-conda_cos6-linux-gnu-cc` 执行程序：

如果你使用的是 python 3.8 且使用了 conda 虚拟环境，可能需要先备份一个文件：

```
# 缺失该文件会导致 pip 无法正常运行
cp $ENV_PATH/lib/python3.8/_sysconfigdata_x86_64_conda_cos7_linux_gnu.py ~/
# 安装该包后应当有 x86_64-conda_cos7-linux-gnu-cc 可执行程序
conda install gcc_linux-64 gxx_linux-64

# 如果 pip 无法正常运行了，说明 _sysconfigdata_x86_64_conda_cos7_linux_gnu.py 被
删除了，从备份恢复这个文件
cp ~/_sysconfigdata_x86_64_conda_cos7_linux_gnu.py
$ENV_PATH/lib/python3.8/_sysconfigdata_x86_64_conda_cos7_linux_gnu.py

# 将 x86_64-conda_cos6-linux-gnu-cc 链接到 x86_64-conda_cos7-linux-gnu-cc
export bin=$(dirname $(which x86_64-conda_cos7-linux-gnu-cc))
ln -s $(which x86_64-conda_cos7-linux-gnu-cc) $bin/x86_64-conda_cos6-linux-
gnu-cc
```

其他包的安装

实验中所需的其他 python 第三方库直接通过 pip 或 conda 安装即可：

```
pip install numpy
# matplotlib 库用以绘制图表
pip install matplotlib
# tqdm 库用以打印进度条，以检测程序当前运行进度
pip install tqdm
```