# scGPT - user guide

Qi Liu

Data Science Group, GHDDI

qi.liu@ghddi.org

2023.08.29

## Get Start

先下载VS Code，远程连接cluster，步骤如下：

1. 下载VS Code: https://code.visualstudio.com/

2. 打开VS Code, 点击侧边栏的扩展

* 搜索框中输入ssh, 下载Remote SSH，并配置服务器[2],
* .ssh/config内容参考Figure 2

3. 在VS Code中连接远程服务器

* 点击侧边栏的"远程资源管理器"
* 连接刚才配置好的服务器，选择在当前/新的窗口打开
* 左下角显示已经连接后，选择侧边栏"打开文件夹"，打开cluster中想要创建项目的路径，或者已经存在的路径

后续就可以直接在cluster上进行开发了.

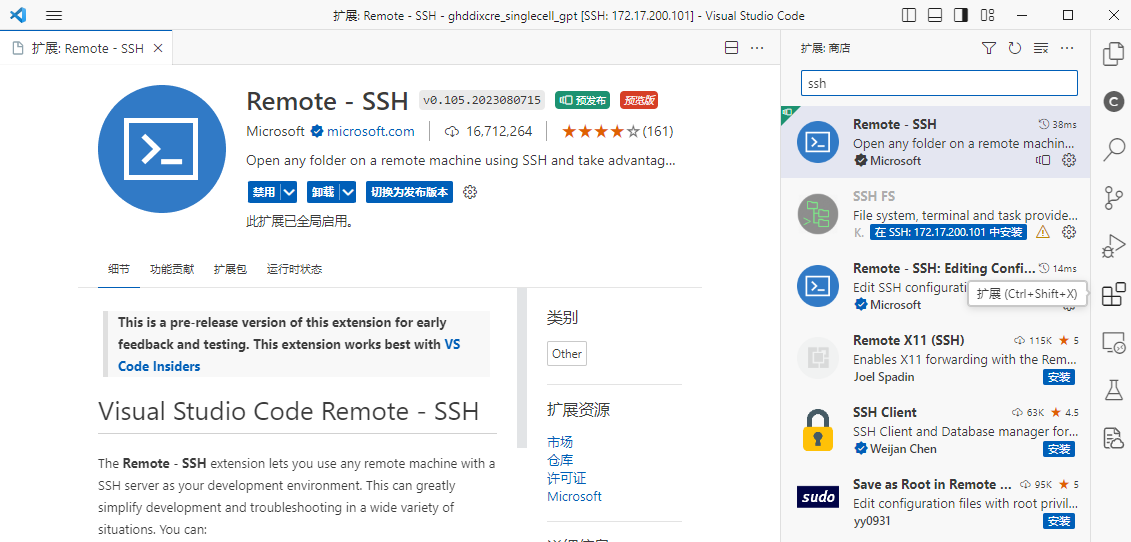


Figure 1 安装Remote SSH扩展

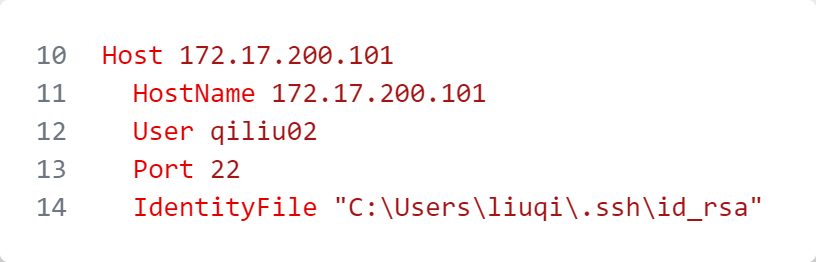
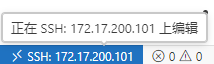


Figure 2 remote ssh的config

 (Figure 3-1)



（Figure 3-2)

Figure 3 (1) 连接server，（1）打开文件夹

## Conda env

安装好VS Code和远程连接好Cluster后，下面安装conda，并激活scGPT的Conda env.

假如当前个人账号没有安装Conda的话，建议先安装[3]，如使用下面命令：

wget https://repo.anaconda.com/archive/Anaconda3-2023.07-2-Linux-x86\_64.sh

bash ./Anaconda3-2023.07-2-Linux-x86\_64.sh

101服务器上已经安装好了用于使用scGPT的环境, 可选Chen Liang (梁忱) 提供的环境：

Python解释器： /home/cliang02/work/bin/cre-python

Conda环境：/home/cliang02/work/software/common/proglang/mambaforge/envs/cre

或者Qi Liu (刘奇）提供的环境：

Conda环境：/home/qiliu02/miniconda3/envs/single\_cell\_gpt

新建Terminal后，可以输入命令激活：

conda activate /home/cliang02/work/software/common/proglang/mambaforge/envs/cre

# or

conda activate /home/qiliu02/miniconda3/envs/single\_cell\_gpt

就切换到了我们运行scGPT的环境了.

注意：conda env中没有安装scgpt package，而是在项目中源码安装scGPT，这是为了随时保持最新和方便修改源码. 后面的预处理脚本用sys.path.append的方式引用了源码包，若有相关问题，可以联系Qi Liu.



Figure 4 新建终端

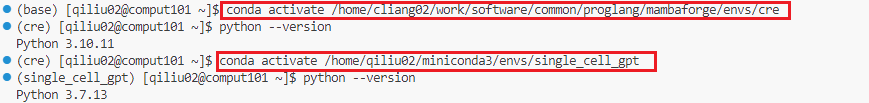


Figure 5 激活scgpt运行的conda环境

## Prepare data

使用我们数据预处理脚本来将.h5ad数据处理成scGPT pretraining需要的input embedding数据集。

首先安装git [4], 终端输入`git --version`测试，已安装的话跳过.

使用git clone, 将预处理脚本clone到自己的路径：

git clone https://github.com/qiliu-ghddi/singlecell\_gpt

clone后，进入到`data`路径下，将我们的要处理的数据放（或者软连接）到`data/raw`，

运行`build\_large\_scale\_data.py`, 给定`\*.h5ad`文件的路径，将其用scgpt.scbank处理为便于大规模数据处理的格式；

接着`binning\_mask\_allcounts.py`, 将上一步的输出，处理为能够作为pretraining scGPT的embedding数据集。

命令如下：

cd data

conda activate <env> # 参考教程里的激活conda env，激活运行我们的scgpt环境

python build\_large\_scale\_data.py --input-dir "raw/" --output-dir "./preprocessed"

python binning\_mask\_allcounts.py --data\_source "./preprocessed/all\_counts/"

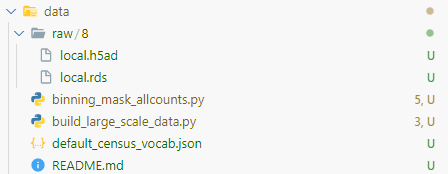


Figure 6-1

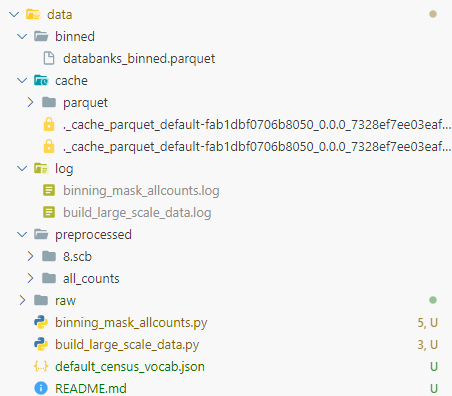


Figure 6-2

Figure 6 (6-1) 处理前的数据，(6-2)处理后的数据和结果

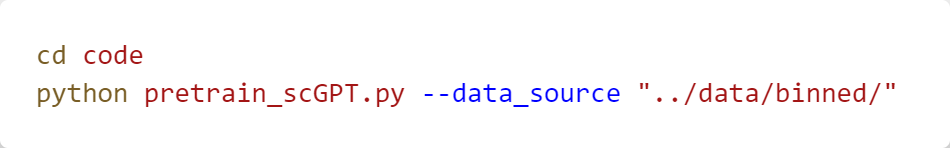
## Pretrain scGPT

我们conda env环境中没有直接安装scGPT[5], 下面源码安装scGPT，注册wandb[7]，随后预训练scGPT.

clone scGPT源，将其中的`scGPT/scgpt`源码放在我们的项目根路径下（和code, data平齐）.

由于scGPT使用了wandb作为日志记录的工具，所以建议去wandb上注册一个账号，激活后，新建终端，输入`wandb login`, 输入账号和密码即可，只需要配置一次，后续就不用管了.

进入到`code`路径下，运行`python pretrain\_scGPT.py`，将`data\_source`配置为我们前面处理得到的`binned`数据集的路径，即可开始训练. 更多参数，请输入`python pretrain\_scGPT.py --help`查看.



训练完后，在`save/xxx-<datetime>`路径下，会保存`checkpoint`即`best\_model`，及参数`args.json`和`vocab.json`文件，可以作为后面`finetune`环节的输入.

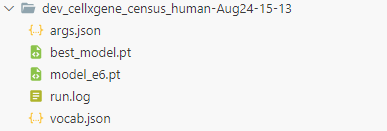


Figure 7 保存得到的checkpoint和结果

## Down-stream tasks and evaluation

scGPT 提供了一些下游finetune任务的jupyter notebooks [8]，将我们训练得到的结果，作为参数，可以运行这些下游任务.

在`examples`中提供了`finetune\_integration`作为例子，修改脚本中的`load\_model`的值，为我们上面训练得到的`checkpoint`所在的路径，即可运行. 其会下载PBMC 10K`数据，并用我们得到的模型进行`batch integration`的finetuning，最后将结果保存到`wandb`的`run`中.

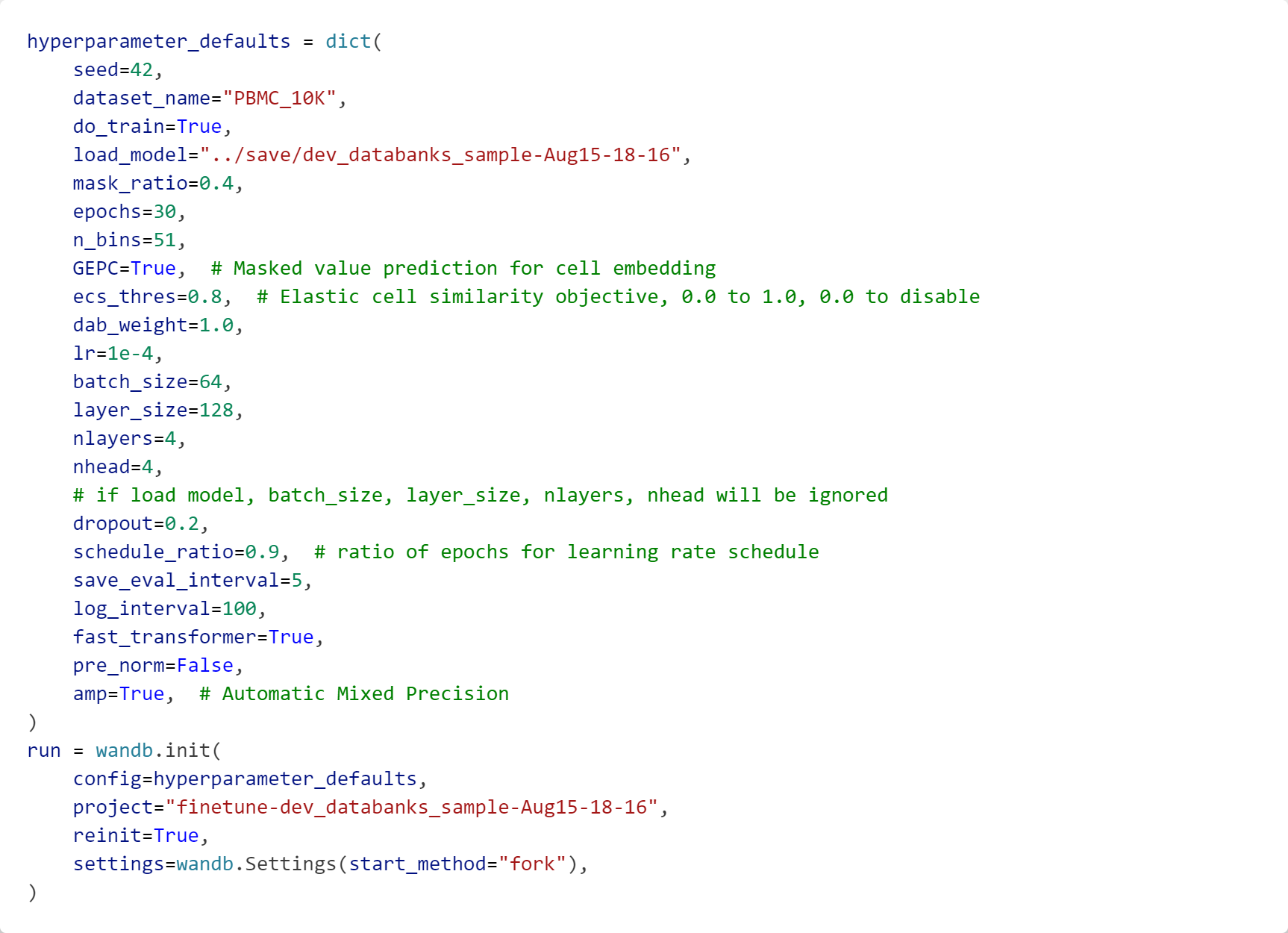


Figure 8 `finetune\_integration.py`中要关注的参数

**参考**

[1] [VS Code](https://code.visualstudio.com/)

[2] [Remote Development using SSH](https://code.visualstudio.com/docs/remote/ssh#\_getting-started)

[3] [Anaconda Download](https://www.anaconda.com/download/)

[4] [Git Download](https://git-scm.com/downloads )

[5] [scGPT](https://github.com/bowang-lab/scGPT)

[6] [scGPT tutorials](https://github.com/bowang-lab/scGPT/tree/main/tutorials)

[7] [Wandb](https://wandb.ai/)

[8] [Wandb login](https://docs.wandb.ai/ref/cli/wandb-login)