

如何在算力云上部署Stable Diffusion

算力云网址 <https://www.autodl.com/home>

这回真的是无任何阉割的版本了，截至4月19日，所有的webui最新版功能都能正常用

零、基础环境

算力云环境

镜像

Miniconda conda3

Python 3.10(ubuntu22.04)

Cuda 11.8

GPU

Tesla T4(16GB) * 1升降配置

CPU8 vCPU Intel Xeon Processor (Skylake, IBRS)

内存56GB

硬盘

系统盘:25 GB

数据盘:免费:50GB SSD

目前服务器是包天租的，一天19.13，包周1周124.42，包月的话，一个月463.48。试验阶段可以租按小时算的，我这个配置1小时0.83元。

一、初始准备，用户权限的设置

1.创建非root管理员用户并改主目录为数据盘

使用以下步骤创建一个名为"autodl"的管理员用户，并将其主目录设置为 /root/autodl-tmp。

1. 以具有管理员权限的用户身份登录。
2. 打开终端窗口。
3. 输入以下命令以创建一个新用户：

```
sudo adduser autodl  
...
```

这将提示您设置新用户的密码和其他详细信息。

4. 更改新用户的主目录：

```
sudo usermod -d /root/autodl-tmp autodl
...
```

2.删除lock锁

```
sudo rm -rf /var/lib/dpkg/lock

sudo rm -rf /var/cache/apt/archives/lock
```

3.更改这两个目录权限

```
sudo chmod 777 /root /root/autodl-tmp
sudo chown -R root:autodl /root /root/autodl-tmp
```

4.改python的用户权限

可以按照以下步骤将其添加到 autodl 用户的环境变量中：

1. 找到Python的安装路径。在终端中输入以下命令来查找：

```
which python
...
```

这将输出Python的路径，例如`/root/miniconda3/bin/python`。

2. 将Python的路径添加到 autodl 用户的环境变量中。在终端中输入以下命令：

```
echo 'export PATH="$PATH:/root/miniconda3/bin"' >>
/home/autodl/.bash_profile
...
```

这会将Python的路径添加到`/home/autodl/.bash_profile`文件中，该文件是`autodl-tmp`用户的默认bash shell配置文件。

3. 使新的环境变量生效。在终端中输入以下命令：

```
source /home/autodl-tmp/.bash_profile
...
```

这将使新的环境变量立即生效，您现在应该可以在`autodl-tmp`用户的终端中使用Python了。

请注意，您需要在 autodl-tmp 用户下运行 `source /home/autodl-tmp/.bash_profile` 命令以使环境变量生效。如果您使用了不同的shell或终端，可能需要调整上述步骤以适应您的系统和配置。

5.打开目录

```
cd /root/autodl-tmp
```

二、安装

1.下载webui包

```
git clone http://ghproxy.com/https://github.com/AUTOMATIC1111/stable-diffusion-webui.git
```

2.webui.sh

把有github网址的地方加上ghproxy, 第151行

```
"${GIT}" clone http://ghproxy.com/https://github.com/AUTOMATIC1111/stable-diffusion-webui.git "${clone_dir}"
```

install路径也改了,改成

```
# Install directory without trailing slash
if [[ -z "${install_dir}" ]]
then
    install_dir="/root/autodl-tmp"
fi
```

3.launch.py

把所有git链接前面加上<http://ghproxy.com/>

```
def prepare_environment():
    global skip_install

    torch_command = os.environ.get('TORCH_COMMAND', "pip install
torch==1.13.1+cu117 torchvision==0.14.1+cu117 --extra-index-url
https://download.pytorch.org/whl/cu117")
    requirements_file = os.environ.get('REQS_FILE', "requirements_versions.txt")

    xformers_package = os.environ.get('XFORMERS_PACKAGE',
'xformers==0.0.16rc425')
    gfpgan_package = os.environ.get('GFPGAN_PACKAGE',
"git+http://ghproxy.com/https://github.com/TencentARC/GFPGAN.git@8d2447a2d918f8e
ba5a4a01463fd48e45126a379")
    clip_package = os.environ.get('CLIP_PACKAGE',
"git+http://ghproxy.com/https://github.com/openai/CLIP.git@d50d76daa670286dd6cac
f3bcd80b5e4823fc8e1")
    openclip_package = os.environ.get('OPENCLIP_PACKAGE',
"git+http://ghproxy.com/https://github.com/mlfoundations/open_clip.git@bb6e834e9
c70d9c27d0dc3ecedeebeaeb1ffad6b")

    stable_diffusion_repo = os.environ.get('STABLE_DIFFUSION_REPO',
"http://ghproxy.com/https://github.com/Stability-AI/stablediffusion.git")
    taming_transformers_repo = os.environ.get('TAMING_TRANSFORMERS_REPO',
"http://ghproxy.com/https://github.com/CompVis/taming-transformers.git")
    k_diffusion_repo = os.environ.get('K_DIFFUSION_REPO',
'http://ghproxy.com/https://github.com/crowsonkb/k-diffusion.git')
    codeformer_repo = os.environ.get('CODEFORMER_REPO',
'http://ghproxy.com/https://github.com/sczhou/CodeFormer.git')
    blip_repo = os.environ.get('BLIP_REPO',
'http://ghproxy.com/https://github.com/salesforce/BLIP.git')
```

```
stable_diffusion_commit_hash =
os.environ.get('STABLE_DIFFUSION_COMMIT_HASH',
"cf1d67a6fd5ea1aa600c4df58e5b47da45f6bdbf")
taming_transformers_commit_hash =
os.environ.get('TAMING_TRANSFORMERS_COMMIT_HASH',
"24268930bf1dce879235a7fddd0b2355b84d7ea6")
k_diffusion_commit_hash = os.environ.get('K_DIFFUSION_COMMIT_HASH',
"5b3af030dd83e0297272d861c19477735d0317ec")
codeformer_commit_hash = os.environ.get('CODEFORMER_COMMIT_HASH',
"c5b4593074ba6214284d6acd5f1719b6c5d739af")
blip_commit_hash = os.environ.get('BLIP_COMMIT_HASH',
"48211a1594f1321b00f14c9f7a5b4813144b2fb9")
```

到这里跑一下webui.sh

到了install gfpgan

4.GFPGAN安装

```
cd /root/autodl-tmp/stable-diffusion-webui/venv/lib64/python3.10/site-packages
git clone https://github.com/TencentARC/GFPGAN.git
pip install cpython
python -m pip install basicsr facexlib
python -m pip install -r requirements.txt
```

5.多运行几遍webui.sh,

因为网络问题，这玩意有时候是个玄学。

一次往往不能完整的运行完。

多运行几次，就可以了

四、打包，装model模型之前打个包

先把它默认的v1-5模型删了，再打包，那个模型比较老了，其实

```
tar -zcvf stable-diffusion-webui.tar.gz stable-diffusion-webui/
```

打好包后下载下来

这样下次安装比较方便

然后把自己准备好的safetensor或cpkg模型上传到models/stable-diffusion/目录下。

五、快速启动设置

修改webui-user.sh

```
export COMMANDLINE_ARGS="--share --gradio-auth username:password"
```

--share 是出一个临时公网ip, --gradio-auth username:password 是设置登录口用户名和密码。

六、使用

再次开机，是在root用户下

```
su autodl
source /home/autodl/.bash_profile
cd /root/auto-tmp/stable-diffusion
./webui.sh
```

七、安全提示

- 1、临时公网ip最好不要到处说
- 2、自己创建的autodl（那个非root用户），密码要设成强密码！
- 3、登录口密码，最好也不要弱口令。

第二个尤其重要，要不然别人就能凭借ssh弱口令用你的显卡服务器了。ssh弱口令是一个高危漏洞。

八、后续SD教程

常用插件和脚本、webui-user.sh的常用参数讲解、模型种类的讲解、模型从哪里找。

下一节教程计划讲双语对照插件和格式转换插件、还有补间动画脚本。这几个都是比较基本简单的插件和脚本。