

rar 解压：

```
sudo apt-get update
```

```
sudo apt-get install unrar
```

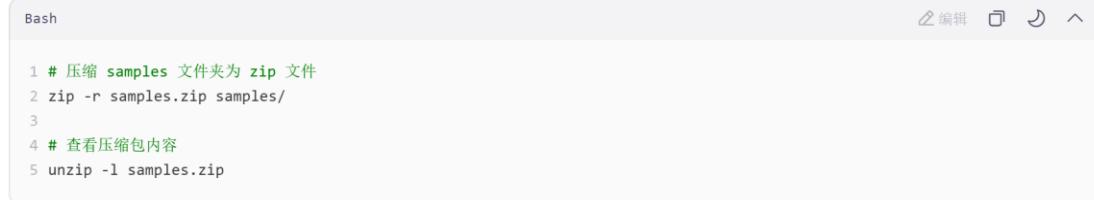
安装完成后，可以使用 unar 命令来解压 Rar 文件：

```
root@dsw-470039-74f5f4b59c-89vs6:/mnt/workspace/user/zhongyurong/FSDrive/LLaMA-Factory/data/nuscen...# unrar x CAM_FRONT.rar  
UNRAR 7.00 freeware Copyright (c) 1993-2024 Alexander Roshal
```

zip 压缩

```
sudo apt update
```

```
sudo apt install zip unzip
```



```
Bash  
编辑 □ ⌂ ^  
1 # 压缩 samples 文件夹为 zip 文件  
2 zip -r samples.zip samples/  
3  
4 # 查看压缩包内容  
5 unzip -l samples.zip
```

tar 压缩

### tar

- 命令选项：

1	-z(gzip)	用gzip来压缩/解压缩文件
2	-j(bzip2)	用bzip2来压缩/解压缩文件
3	-v(verbose)	详细报告tar处理的文件信息
4	-c(create)	创建新的档案文件
5	-x(extract)	解压缩文件或目录
6	-f(file)	使用档案文件或设备，这个选项通常是必选的。

- 命令举例：

```
1 #压缩  
2 [root@localhost tmp]# tar -zvcf buodo.tar.gz buodo  
3 [root@localhost tmp]# tar -jvcf buodo.tar.bz2 buodo  
4  
5 #解压  
6 [root@localhost tmp]# tar -zvxf buodo.tar.gz  
7 [root@localhost tmp]# tar -jvxf buodo.tar.bz2
```

<https://blog.csdn.net/capecape/article/details/78548723>

## rm

- 删除文件或目录

使用 rm 命令要小心，因为文件删除后不能恢复

选项	含义
-f	强制删除，忽略不存在的文件，无需提示
-r	递归地删除目录下的内容，删除文件夹时必须加此参数

删除当前目录下所有以"CAM"开头且后缀为"rar"的文件： rm CAM\*.rar

### 1、mv 命令

mv是"move"的缩写，用于 移动文件或目录到另一个位置。

将文件 a.txt 移动到 目录home下

```
bash                                     ☺ AI生成项目 | ⚡ 复制
1 | mv a.txt home
```

将 目录cl 移动到 目录home下

```
bash                                     ☺ AI生成项目 | ⚡ 复制
1 | mv cl home
```

将 a.txt 重命名为 b.txt

```
bash                                     ☺ AI生成项目 | ⚡ 复制
1 | mv a.txt b.txt
```

将 目录cl 重命名为 目录cl2

```
bash                                     ☺ AI生成项目 | ⚡ 复制
1 | mv cl cl2
```

### 2、cp 命令

cp是"copy"的缩写，用于 复制文件或目录到另一个位置。

复制 文件a.txt 到 目录home下

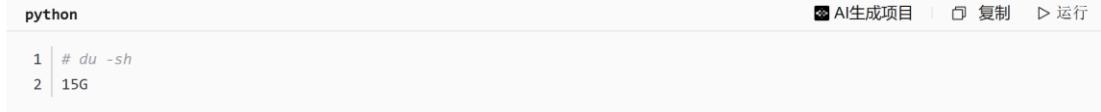
```
bash                                     ☺ AI生成项目 | ⚡ 复制
1 | cp a.txt home
```

复制 目录cl 到 目录home下

```
bash                                     ☺ AI生成项目 | ⚡ 复制
1 | cp -r cl home   (-r选项用于递归复制，复制整个目录树)
```

将 user/zhong1 目录下的所有文件（包括隐藏文件）拷贝到 user/zhong2 目录下的正确命令是： cp -r user/zhong1/. user/zhong2/

1、查看当前目录总共占的容量，而不单独列出各子项占用的容量，在所在目录输入du -sh：



```
python
1 | # du -sh
2 | 15G
```

The screenshot shows a terminal window with a light gray background. At the top, there's a header bar with the word "python" on the left and three icons on the right: "AI生成项目", "复制" (Copy), and "运行" (Run). Below the header, the terminal prompt is "# du -sh". The output of the command is shown in two lines: "1" and "2", followed by a vertical pipe symbol, then "# du -sh" and "15G" respectively.

wget [https://d36yt3mvayqw5m.cloudfront.net/public/v1.0/v1.0-trainval07\\_blobs.tgz](https://d36yt3mvayqw5m.cloudfront.net/public/v1.0/v1.0-trainval07_blobs.tgz)

项目 TPVFormer\_NeRF 所要使用的数据应存放在服务器的 /mnt/workspace/user/zhongyurong/dz/TPVFormer\_NeRF/data/nuscenes 这个目录下，但目前这些数据都存放 在 服 务 器 的 /mnt/workspace/user/zhongyurong/dz/TPVFormer/data/nuscenes 这个目录下，因为 /mnt/workspace/user/zhongyurong/dz/TPVFormer/data/nuscenes 这个目录下所存放的数据所需存储空间太大了，因此如何在不迁移或者复制数据到 /mnt/workspace/user/zhongyurong/dz/TPVFormer\_NeRF/data/nuscenes 目录下的前提下让项目 TPVFormer\_NeRF 能够正常使用 /mnt/workspace/user/zhongyurong/dz/TPVFormer/data/nuscenes 这个目录下所存放的数据：

在 Linux 系统中，可以这样操作：

```
cd /mnt/workspace/user/zhongyurong/dz/TPVFormer_NeRF
# 1. 创建 data 目录
mkdir -p data
# 2. 创建 nuscenes 符号链接
ln -s /mnt/workspace/user/zhongyurong/dz/TPVFormer/data/nuscenes data/nuscenes
```

或者

```
ln -s ../../TPVFormer/data/nuscenes data/nuscenes
# 验证
ls -la data/
```