

# Linux基本操作学习

## 1. 介绍

Linux系统在我们正式转到算法这个行业的时候，就成为了我们必须会用到的一种系统了。

与Windows的图形化界面不同，Linux更多的是使用**命令行**的形式来进行操作，比如我们新建一个"a.txt"的文件，在windows中，我们主要是使用鼠标->右击->新建文件等步骤。在linux下，我们直接运行命令 **touch a.txt**即可。

那么为什么我们要学习linux系统呢？与windows相比，linux有哪些优势呢？我们常用的linux又有哪些版本呢？

### 1. 为什么我们需要学习linux系统？

这个问题的答案是因为：**有的公司在用**。你不知道公司是用windows还是linux系统，但是以我目前遇到的几个公司，都是使用的linux服务器进行训练以及开发、部署等工作。



**我能怎么办啊  
我也很绝望啊**

### 2. 为什么公司用linux服务器，它有什么优点呢？

- Linux免费，并且开源，相对于免费，开源更加重要。
- 开源意味着更加安全。
- 开源意味着可以优化。

开源透明对真正懂linux内核的人而言，可以自定义linux，如加入自己的功能，修改潜在的bug，或者对某些模块功能进行优化。若你只会linux命令，那么其实你都不清楚你的机器有没有被植入木马。因此，选择linux或是windows进行个人机器，每个人的想法是不一样的。不过，如Pytorch等深度学习框架，在一开始的时候是先开出来linux版本，后来再推出windows的版本的，可能对于他们来说，默认程序员们，都是使用linux的吧。或许，这就是人生吧！



### 3. 常用Linux版本介绍

目前主要在生产环境中使用的linux服务器有centos、Ubuntu等linux系统。

对于我们个人而言，以ubuntu为例来学习linux的基本操作，或者打开你的mac也可以来进行linux的命令学习。对于安装ubuntu系统而言，可以在网上下载镜像，再安装即可。

可以参考我们提供一个安装ubuntu双系统的教程！

## 2. Linux的系统目录结构

在安装完ubuntu之后，我们首先查看下linux下的系统目录结构。

```
1 | ls /
```

运行完之后，则我们可以看到类似下面的页面：

```
[root@localhost ~]# ls /
bin    dev    home   lost+found  mnt    proc    sbin    srv    tmp    var
boot   etc    lib    media      opt    root    selinux sys    usr
```

我们可以先来学习这些文件夹是干啥的：

- /bin：  
bin 是 Binaries (二进制文件) 的缩写，这个目录存放着最经常使用的命令。
- /boot：  
这里存放的是启动 Linux 时使用的一些核心文件，包括一些连接文件以及镜像文件。
- /dev：  
dev 是 Device(设备) 的缩写，该目录下存放的是 Linux 的外部设备，在 Linux 中访问设备的方式和访问文件的方式是相同的。
- /etc：  
etc 这个目录用来存放所有的系统管理所需要的配置文件和子目录。
- /home：  
用户的主目录，在 Linux 中，每个用户都有一个自己的目录，一般该目录名是以用户的账号命名的。
- /lib：  
lib 是 Library(库) 的缩写这个目录里存放着系统最基本的动态连接共享库，其作用类似于 Windows 里的 DLL 文件。几乎所有的应用程序都需要用到这些共享库。
- /lost+found：

这个目录一般情况下是空的，当系统非法关机后，这里就存放了一些文件。

- /media:  
linux 系统会自动识别一些设备，例如U盘、光驱等等，当识别后，Linux 会把识别的设备挂载到这个目录下。
- /mnt:  
系统提供该目录是为了让用户临时挂载别的文件系统的，我们可以将光驱挂载在 /mnt/ 上，然后进入该目录就可以查看光驱里的内容了。
- /opt:  
opt 是 optional(可选) 的缩写，这是给主机额外安装软件所摆放的目录。比如你安装一个ORACLE数据库则就可以放到这个目录下。默认是空的。
- /proc:  
proc 是 Processes(进程) 的缩写，/proc 是一种伪文件系统（也即虚拟文件系统），存储的是当前内核运行状态的一系列特殊文件，这个目录是一个虚拟的目录，它是系统内存的映射，我们可以通过直接访问这个目录来获取系统信息。  
这个目录的内容不在硬盘上而是在内存里，我们也可以直接修改里面的某些文件，比如可以通过下面的命令来屏蔽主机的ping命令，使别人无法ping你的机器：  
`echo 1 > /proc/sys/net/ipv4/icmp_echo_ignore_all`
- /root:  
该目录为系统管理员，也称作超级权限者的用户主目录。
- /sbin:  
s 就是 Super User 的意思，是 Superuser Binaries (超级用户的二进制文件) 的缩写，这里存放的是系统管理员使用的系统管理程序。
- /selinux:  
这个目录是 Redhat/CentOS 所特有的目录，Selinux 是一个安全机制，类似于 windows 的防火墙，但是这套机制比较复杂，这个目录就是存放selinux相关的文件的。
- /srv:  
该目录存放一些服务启动之后需要提取的数据。
- /sys:  
这是 Linux2.6 内核的一个很大的变化。该目录下安装了 2.6 内核中新出现的一个文件系统 sysfs 。
- /tmp:  
tmp 是 temporary(临时) 的缩写这个目录是用来存放一些临时文件的。
- /usr:  
usr 是 unix shared resources(共享资源) 的缩写，这是一个非常重要的目录，用户的很多应用程序和文件都放在这个目录下，类似于 windows 下的 program files 目录。
- /usr/bin:  
系统用户使用的应用程序。
- /usr/sbin:  
超级用户使用的比较高级的管理程序和系统守护程序。
- /usr/src:  
内核源代码默认的放置目录。
- /var:  
var 是 variable(变量) 的缩写，这个目录中存放着在不断扩充着的东西，我们习惯将那些经常被修改的目录放在这个目录下。包括各种日志文件。
- /run:  
是一个临时文件系统，存储系统启动以来的信息。当系统重启时，这个目录下的文件应该被删掉或清除。如果你的系统上有 /var/run 目录，应该让它指向 run。

## 3. 基本命令

linux的命令遵循一个格式，就是**命令 -选项 参数**(选项与参数可以为空)  
如：

```
1 | ls -la /home #其中的ls为命令 选项为l与a 参数就是/home
```

### 3.1 不同版本linux的安装命令

对于不同的linux版本，安装包的命令主要就是yum或者apt这两个命令。

- yum

在 Fedora 和 RedHat 以及 SUSE 中的 Shell 前端软件包管理器,实际中，主要是centos系统，我们采用yum命令来安装包。其实不用管那么多，centos系统，我们缺啥，用yum来装就好。yum常用的命令如下：

完整命令	解释说明
yum check-update	列出所有可更新的软件清单命令
yum update	更新所有软件命令
yum install <package_name>	仅安装指定的软件命令
yum update <package_name>	仅更新指定的软件命令
yum list	列出所有可安装的软件清单命令
yum remove <package_name>	删除软件包命令
yum search <package_name>	查找软件包命令
yum clean packages	清除缓存目录下(/var/cache/yum)的软件包
yum clean headers	清除缓存目录下的 headers
yum clean oldheaders	清除缓存目录下旧的 headers
yum clean, yum clean all (= yum clean packages; yum clean oldheaders)	清除缓存目录下的软件包及旧的 headers

**注意：**yum 会把下载的软件包和header存储在cache中，而不会自动删除。假如我们觉得他们占用了磁盘空间，能够使用yum clean指令进行清除，更精确 的用法是yum clean headers清除header，yum clean packages清除下载的rpm包，yum clean all一股脑儿端。

而类似python的pip包，会有国内的源，配置国内的源，下载肯定会快很多。对于yum，同样是在一些包的服务器上下载安装，外国的服务器我们直接下载肯定慢的一批，所以我们需要配置成我们国内的源,当然这些是我们可以联网的情况下，我们可以用网上的源，如果我们服务器不能联网怎么办？那么你就需要下载rpm包上传进行安装，或者你本地搭载一个yum源。我们默认你可以联网，那么可以按照下面的步骤往下走。

这里建议使用网易的源，其实别的源也是同样可以的，如中国科大的源，清华的源，修改过程如下：

1. 备份原先的地址

```
1 | mv /etc/yum.repos.d/CentOS-Base.repo /etc/yum.repos.d/CentOS-Base.repo.backup
```

2. 下载对应版本的repo文件，放入/etc/yum.repos.d/

- CentOS5 : <http://mirrors.163.com/.help/CentOS5-Base-163.repo>
- CentOS6 : <http://mirrors.163.com/.help/CentOS6-Base-163.repo>
- CentOS7 : <http://mirrors.163.com/.help/CentOS7-Base-163.repo>

```
1 | wget http://mirrors.163.com/.help/CentOS6-Base-163.repo
2 | mv CentOS6-Base-163.repo CentOS-Base.repo
```

### 3. 生成缓存

```
1 | yum clean all
2 | yum makecache
```

- apt

apt是一个在 Debian 和 Ubuntu 中的 Shell 前端软件包管理器，apt 命令提供了查找、安装、升级、删除某一个、一组甚至全部软件包的命令，而且命令简洁而又好记。

apt 命令执行需要超级管理员权限(root)，这里默认你是有root权限，如果没有权限，后面我们会搭建一个无root无网搭建caffe环境的教程。

#### apt常用命令

sudo apt update

sudo apt upgrade

apt list --upgradable

sudo apt full-upgrade

sudo apt install <package\_name>

sudo apt install <package\_1> <package\_2>  
<package\_3>

sudo apt update <package\_name>

sudo apt show <package\_name>

sudo apt remove <package\_name>

sudo apt purge <package\_name>

sudo apt search

sudo apt autoremove

apt list --installed

apt list --all-versions

#### 解释说明

列出所有可更新的软件清单命令

升级软件包

列出可更新的软件包及版本信息

升级软件包，升级前先删除需要更新软件包

安装指定的软件命令

安装多个软件包

更新指定的软件命令

显示软件包具体信息,例如：版本号，安装大小，依赖关系等等

删除软件包命令

移除软件包及配置文件

查找软件包命令

清理不再使用的依赖和库文件

列出所有已安装的包

列出所有已安装的包的版本信息

同理，修改源：

1. sudo vim /etc/apt/sources.list删除其中所有内容
2. 添加以下内容：

```
1 deb http://mirrors.aliyun.com/ubuntu/ trusty main multiverse restricted
  universe
2 deb http://mirrors.aliyun.com/ubuntu/ trusty-backports main multiverse
  restricted universe
3 deb http://mirrors.aliyun.com/ubuntu/ trusty-proposed main multiverse
  restricted universe
4 deb http://mirrors.aliyun.com/ubuntu/ trusty-security main multiverse
  restricted universe
5 deb http://mirrors.aliyun.com/ubuntu/ trusty-updates main multiverse
  restricted universe
6 deb-src http://mirrors.aliyun.com/ubuntu/ trusty main multiverse restricted
  universe
7 deb-src http://mirrors.aliyun.com/ubuntu/ trusty-backports main multiverse
  restricted universe
8 deb-src http://mirrors.aliyun.com/ubuntu/ trusty-proposed main multiverse
  restricted universe
9 deb-src http://mirrors.aliyun.com/ubuntu/ trusty-security main multiverse
  restricted universe
10 deb-src http://mirrors.aliyun.com/ubuntu/ trusty-updates main multiverse
    restricted universe
```

保存退出

### 3. 更新

```
1 | sudo apt-get update
```

## 3.2 操作文件与目录命令

命令	选项	完整命令	解释说明
ls		ls	显示当前目录下文件
	目录名	ls /home	显示指定目录下文件
	-l	ls -l	长格式显示目录文件
	-l	ls -l 1.py	长格式显示指定文件
	-a	ls -a /home	显示所有文件(包含隐藏文件)
	-al	ls -al /home	长格式显示当前目录下所有文件
	-h	ls -h 1.py	文件大小显示为常见大小单位 B KB MB ...
	-d	ls -d	显示目录本身，而不是里面的子文件
cat		cat 1.py	显示1.py的所有内容
more		more 1.py	分页显示文本文件的内容，可以前后翻页，空格表示向后翻页，b向前翻页
less		less 1.py	分页显示文本文件的内容，可以前后翻页，空格表示向后翻页，b向前翻页,支持底行模式
head		head 1.py	显示文件的开头内容，默认为10行内容
	-[num]	head -20 1.py	显示文件前20行内容
tail		tail 1.py	显示文件的末尾内容，默认为10行内容
	-[num]	tail -20 1.py	显示文件末尾20行内容
	-f	tail -f 1.py	可循环读取文件，并动态显示在屏幕上
	-F	tail -F 1.py	循环读取文件，并动态显示在屏幕上
wc		wc 1.py	统计该文件的行数、字数、字符数
	-m	wc -m 1.py	字符数
	-w	wc -w 1.py	文本字数
	-m	wc -l 1.py	文本行数
cd		cd /home	切换到/home目录下
pwd		pwd	打印当前路径

### 3.3 文件权限简介

其中，对于使用-l参数的长格式显示项，如下：

```
1 | ls -l a.py
```

得到的结果如下：

```
1 | -rw-r--r-- 1 root root 1078 12月 10 17:56 a.py
2 | #让我们做个标记再进行讲解
3 | # (1)      (2)  (3)  (4)  (5)  (6)  (7)
```

针对做的标记进行讲解：

- (1) 为权限位，下面再进行详细说明解释
- (2) 引用计数
- (3) 所有者(属于哪一个用户)

- (4) 所属组(属于哪一个用户组)
- (5) 文件大小
- (6) 最后一次修改时间
- (7) 文件名

对于权限位，详细参考下图：



- 其中为r,w,x,分别代表可读权限，可写权限以及可执行权限(也能够以二进制表示，其中r为100 w为010 x为001)。同时我们一个文件前三位表示这个文件的所属用户，即创建它的人有哪些权限，如上面例子种就有rw的权限，换成数字就是(4+2=6)。
- 其中第一位d表示目录，-位普通文件，l为链接文件
- 第2-4位：所属用户权限，用u(user)表示
- 第5-7位：所属组权限，用g(group)表示
- 第8-10位：其它用户权限，用o(other)表示
- 第2-10位：表示所有的权限，用a(all)表示

对于权限管理，相关的命令主要是chmod与chown。  
具体的用法如下：

命令	选项	完整命令	解释说明
chmod		chmod u + r 1.py	修改文件或者目录的权限。该命令是给当前用户添加可执行的权限
	-R	chmod -R u+r /home/demo	修改/home/demo这个文件夹所有内容的权限，给这些文件的所属用户添加执行权限
		chmod 755 run.sh	7 = 4 + 2 + 1 表示所属用户用户读、写、执行文件的权限。 5=4+1 表示所属组中的用户有读与执行的权限 同理，其它用户也有读与执行的权限
chown		chown user1:group1 1.py	修改文件的所属用户与组。例：将1.py文件的所属用户指定为用户1，组为group1
	-R	chown -R user1:group1 /home/demo	修改/home/demo这个文件夹下所有文件及子目录的所属用户与组，可以用数字表示权限(r=4,w=2,x=1)

### 3.4 压缩与解压缩文件

在我们训练模型的时候，我们一般数据集都是以压缩包的形式保存。在linux服务器中，我们会涉及到数据集的解压缩操作,其中常用的命令为zip、zip、unzip。



- tar命令

目前在使用tar命令来进行**压缩操作**，常用的压缩方式有以下三种：

- 1. gzip：是公认的压缩这速度最快，压缩大文件的时候与其他的压缩方式相比更加明显，历史最久，应用最广泛的压缩方式
- 2. bzip：压缩形成的文件小，但是可用性不如gzip
- 3. xz：是最新的压缩方式，可以自动提供最佳的压缩率

对tar命令，常见的操作如下表：

参 数	解 释
-z	是否同时具有gz属性
-j	是否同时具有bz2属性
-J	是否同时具有xz属性
-x	解压缩、提取打包的内容
-t	查看压缩包内容
-c	建立一个压缩，打包文档
-C	切换到指定目录，表示指定解压缩包的内容和打包的内容存放的目录
-v	显示压缩或者打包的内容
-f	使用文件名，在f后面要接压缩后的文件的名字，只要用到tar命令，-f选项是必须要用的，-f参数在使用的时候一定排在其他参数的后面，在最右边
-p	保留备份数据的原本权限与属性，常用于备份（-c）重要的配置文件
-P	保留绝对路径
-z	用于gzip压缩方式
-j	bzip2
-J	用于xz压缩方式
-r	表示追加的意思

常用的实际操作中的命令如下表：

命令	选项	完整命令	解释说明
tar	-cvf	tar -cvf data.tar 1.txt 2.txt 3.txt	将1.txt,2.txt,3.txt打包成data.tar
	-tf	tar -tf data.tar	查看data.tar里面的内容
	-xf	tar -xf data.tar -C /home/demo	提取data.tar文件的内容到/home/demo的目录下
	-zcvf	tar -zcvf data.tar.gz data	将data文件夹打包成data.tar.gz,使用-z表示按照gzip压缩方式
	-jcvf	tar -jcvf data.tar.bz2 data	将data文件夹打包成data.tar.bz2,使用-j表示按照bzip2压缩方式
	-Jcvf	tar -Jcvf data.tar.xz data	将data文件夹打包成data.tar.xz,使用-J表示按照xz压缩方式
	-rf	tar -rf data.zip label.txt	将label.txt追加进data.zip压缩包中

而如果在运行过程中，遇到问题，可能需要安装bzip2等包，在centos中，我们可以使用：

```
1 | yum -y install gzip2
```

而如果是ubuntu等系统，我们可以使用：

1 | `sudo apt-get install gzip2`

- zip与unzip命令  
顾名思义，zip与unzip是压缩与解压缩。

1. zip的常用参数：

参数	解释
-m	将文件压缩后，删除原文件
-o	将压缩文件内的所有文件的最新变动时间设为压缩的时间
-q	安静模式，在压缩的时候不显示指令执行的过程
-r	递归压缩，将自定目录下的所有子文件以及文件一起处理
-x	”文件列表“，压缩时排除文件列表中的文件

常用的实际操作中的命令如下表：

命令	选项	完整命令	解释说明
zip	-r	zip -r demo.zip demo	将demo压缩成demo.zip压缩包，其中显示压缩过程
zip	-q -r	zip -q -r demo.zip demo	将demo压缩成demo.zip压缩包，不显示压缩过程
zip	-q -r	zip -q -r demo.zip demo run.sh 1.txt	将demo文件夹，run.sh，1.txt压缩成demo.zip

2. unzip的常用参数

参数	解释
-c	将解压缩的结果显示到屏幕上（显示每一个目录下的每一个文件的内容），同时对字符做适当的转换，但是并没有解压压缩包
-l	显示压缩文件内所包含的文件
-t	检查压缩文件是否正确
-v	执行时显示压缩文件的详细信息
-q	安静模式，执行时不显示任何信息
-d	指定文件解压后存储的目录
-x	指定不要处理压缩文件中的那些文件

常用的实际操作中的命令如下表：

命令	选项	完整命令	解释说明
unzip	-c	unzip -c data.zip	将data.zip压缩包解压缩的结果显示出来
unzip	-l	unzip -l data.zip	不解压，显示data.zip压缩包里面所包含的文件
unzip	-t	unzip -t data.zip	不解压，检查data.zip压缩文件是否正确
unzip	-v	unzip -v data.zip	不解压，查看压缩包里面的内容（查看显示的文件列表还包含压缩比率，其内容比使用-l显示的信息要更加详细）
unzip	-q -d	unzip -q data.zip -d /home/	将data.zip解压到/home文件夹下

### 3.5 磁盘管理

Linux磁盘管理好坏直接关系到整个系统的性能问题，在linux的磁盘管理中，常用的三个命令为df、du、fdisk。其中：

- df: 列出文件系统的整体磁盘使用量
- du: 检查磁盘空间使用量
- fdisk: 用于磁盘分区

1. df  
常见选项与参数：

参数	解释
-a	列出所有的文件系统，包括系统特有的 /proc 等文件系统
-k	以 KBytes 的容量显示各文件系统
-m	以 MBytes 的容量显示各文件系统
-h	以人们较易阅读的 GBytes, MBytes, KBytes 等格式自行显示
-H	以 M=1000K 取代 M=1024K 的进位方式
-T	显示文件系统类型, 连同该 partition 的 filesystem 名称 (例如 ext3) 也列出；
-i	不用硬盘容量，而以 inode 的数量来显示

实际中常用的命令：

命令	选项	完整命令	解释说明
df		df	将系统内所有的文件系统列出来！
	-h	df -h	将容量结果以易读的容量格式显示出来
	-h	df -h /etc	将 /etc 底下的可用的磁盘容量以易读的容量格式显示
	-aT	df -aT	将系统内的所有特殊文件格式及名称都列出来

2. du

Linux du命令也是查看使用空间的，但是与df命令不同的是Linux du命令是对文件和目录磁盘使用的空间的查看，还是和df命令有一些区别的  
常见选项与参数：

参数	解释
-a	列出所有的文件与目录容量，因为默认仅统计目录底下的文件量而已。
-h	以人们较易读的容量格式 (G/M) 显示；
-s	列出总量而已，而不列出每个各别的目录占用容量
-S	不包括子目录下的总计，与 -s 有点差别
-k	以 KBytes 列出容量显示
-m	以 MBytes 列出容量显示

实际中常用的命令如下：

命令	选项	完整命令	解释说明
du		du	只列出当前目录下的所有文件夹容量（包括隐藏文件夹）：
	-h	du -h	将文件的容量也列出来
	-sh	df -sh *	将当前文件夹下所有文件与大小显示出来

### 3. fdisk

fdisk 是 Linux 的磁盘分区表操作工具。分区的意思就是类似windows系统中的C盘，D盘一样。基本上我们fdisk使用的就是-l命令。

命令	选项	完整命令	解释说明
fdisk	-l	fdisk -l	系统将会把整个系统内能够搜寻到的装置的分区均列出来

其次，还有一些磁盘挂载，包括磁盘格式化的内容，在这里，我们就暂时不深入讲解了。

## 3.6 其它常用快捷键

命令	选项	完整命令	解释说明
ctrl + c			停止进程
ctrl + l			清屏
ctrl + r			搜索历史命令
ctrl + q			退出
tab			自动补全
echo		echo "hehe!"	在屏幕上输出"hehe"字符
>		echo "hehe!" > 1.txt	先清空1.txt当中的内容，再将"hehe!"输入到1.txt文件中
>>		echo "hehe!" >> 1.txt	不清空1.txt当中的内容，保留原先的内容，再将"hehe!"以追加的形式输入到1.txt文件中
		ls   wc -l	统计当前的文件数
nohup		nohup python -u train.py > train.log 2>&1 &	在后台运行train.py，同时将打印的内容输入至train.log文件中 去

## 4. Linux下的文本编辑

### 4.1. 图形化文本编辑

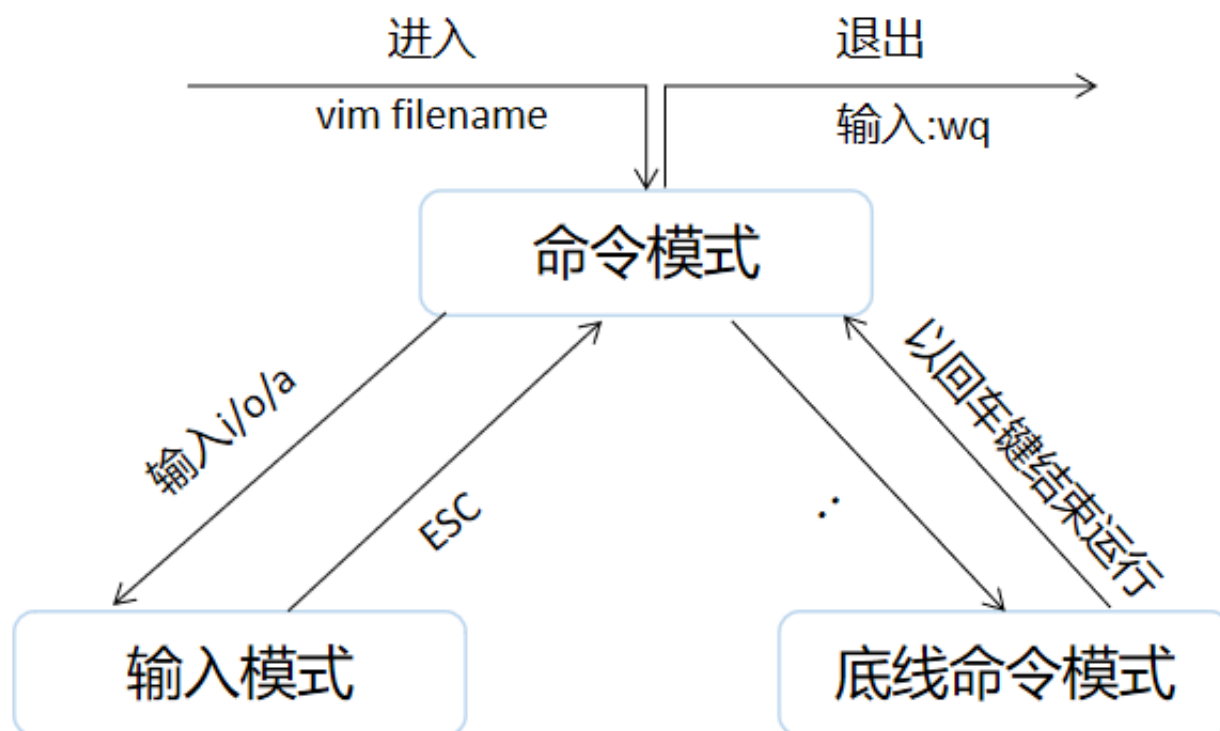
对于图像化文本，那么对于文件的修改，十分简单。类似于windows，我们可以右击使用gedit打开，或者再terminal中使用gedit train.py 命令，就可以打开这个文件，进行修改，再使用ctrl+s进行保存,这样的流程，与windows下面基本上是一样的。

## 4.2. vim

vim是一个大杀器，在linux下，新手一般都不喜欢vim，首先快捷键很多，不一定记得住，而老手们，喜欢vim的就会喜欢的不得了，极大的提高了工作效率。

本文中，主要也是介绍我在实际中使用vim的常用快捷键，如果你想深入学习vim，那么可以自行查阅资料！

vim/vi是linux中常用的文本编辑器，只有命令，没有菜单。vim中主要有以下三种模式：



- 命令模式

当用户输入vim之后，就进入了命令模式。此状态下敲打键盘的动作会被vim识别成命令，而非字符。比如我们此时按下i，这时并不会输入一个字符，而是被当成了一个命令，切换到输入模式。而vim很多的快捷键就是在命令模式下使用，比如连续按两下d就会删除当前行的内容。这些常用快捷键，在下面我们再详细展开说。

- 输入模式

当我们从命令模式通过i，o，a等按键从命令模式变成了输入模式，此时我们就可以正常对文档进行修改了。如我们使用Enter进行换行，DEL删除光标后一个字符，方向键进行上下左右操作等。

- 底线命令模式

在底线命令模式下按照:就会进入底线命令模式。在底线命令模式可以输入单个或者多个字符的命令。

## 4.2.1 编辑文本

命令	解释
i	在光标前插入
I	在光标当前前行插入
a	在光标后插入
A	在光标当前行末尾插入
o	在光标当前行的下一行插入新行
O	在光标当前行的上一行插入新行
:wq	保存并退出
:q!	强制不保存，并退出
yy	复制光标所在行
nyy	n为数字，复制光标所在的向下n行，如5yy表示复制5行
y1G	复制当前行到首行的所有数据
yG	复制当前行到最后一行的所有数据
y0	复制光标所在的那个字符到该行行首的所有数据
y\$	复制光标所在的那个字符到该行行尾的所有数据

p 为将已复制的数据在光标下一行贴上，P 则为贴在光标上一行！举例来说，我目前光标在第 20 p, P 行，且已经复制了 10 行数据。则按下 p 后，那 10 行数据会贴在原本的 20 行之后，亦即由 21 行开始贴。但如果是按下 P 呢？那么原本的第 20 行会被推到变成 30 行。

## 4.2.2 快捷定位

命令	解释
:set nu	显示行号
:set nonu	取消行号
gg	跳到首行
G	取消末行
:n	跳到第n行
/word	向光标之下寻找word字符串，例如需要寻找python这个字符串，那么可以输入/python即可。
?word	向光标之上寻找word的字符串

## 4.2.3 内容替换与取消

命令	解释
u	取消上一步操作
r	替换光标所在处的字符
R	从光标所在处开始替换，按ESC键结束
:n1,n2s/word1/word2/g	n1 与 n2 为数字。在第 n1 与 n2 行之间寻找 word1 这个字符串，并将该字符串取代为 word2
:1,\$s/word1/word2/g 或 :%s/word1/word2/g	从第一行到最后一行寻找 word1 字符串，并将该字符串取代为 word2
:1,\$s/word1/word2/gc 或 :%s/word1/word2/gc	从第一行到最后一行寻找 word1 字符串，并将该字符串取代为 word2！且在取代前显示提示字符给用户确认 (confirm) 是否需要取代！

## 4.2.4 内容删除命令

命令	解释
x	删除光标所在处字符
nx	删除光标所在处后的n个字符
dd	删除光标所在行。n dd删除n行
dG	删除光标所在行到末尾行的所有内容
d1G	删除光标所在到第一行的所有数据
d\$	删除光标所在处，到该行的最后一个字符
d0	那个是数字的 0，删除光标所在处，到该行的最前面一个字符
D	删除光标所在处到行尾的内容
:i, jd	删除i到j，指定范围的行

## 4.2.5 其它快捷键

命令	解释
Shift + zz	保存退出，与":wq"作用相同
v	进入字符可视模式
V	进入行可视模式
Ctrl + V	进入块可视模式
:w	将编辑的数据写入硬盘档案中(常用)
:w!	若文件属性为『只读』时，强制写入该档案。不过，到底能不能写入，取决于有没有权限！
:q	离开vi
:q!	若曾修改过档案，又不想储存，使用！为强制离开不储存档案。

## 5. 总结

Linux的常用命令，差不多就这些啦！那么其实还有一些可以提高我们工作效率的很多东西，比如awk,这种来处理文本，但是这些我们暂时用不到，等后面有机会的时候，再进行补充，基本上我们百分之90的常用Linux命令都在这里面了。

## 6. 关于我

“大家好！我是灿视，百面AI号主。94年奶爸一枚，华五软工硕士毕业，转行的算法工程师。我一路从月薪3k到年薪30w，从双非材料到转行算法，从农村孩子到华五院校，从孩子到人父，这一路上我有很多话想跟大家分享！

欢迎各位加我微信，讨论转行，求职，面经甚至带娃的经验哦！”







欢迎加我微信，每天16个小时在线