

# X<sub>Y</sub>L<sup>A</sup>T<sub>E</sub>X 指南

在 *Ubuntu* 下排版专业的 *pdf* 文章

万泽<sup>1</sup> | 万泽<sup>2</sup>

版本： 1.11

---

<sup>1</sup>作者：万泽，湖南常德人氏。

<sup>2</sup>编者：万泽，邮箱：[a358003542@gmail.com](mailto:a358003542@gmail.com)。

# 前言

我试图收集整理各方面来的 `xelatex` 相关的信息，包括自己的实践经验总结，作为后学者的指南手册。

一切在尊重版权的基础上出于爱好兴趣和相信自由分享的价值原则上进行。

全书主要分为五大块：

## 1. ubuntu 知识

主要是一些最基本的 `ubuntu` 知识或者和本文其他内容相关的知识。当然有时出于全面性考虑会加入一点其他的 `linux` 知识进来，不过这部分需要做到不过分扩充内容。

## 2.xelatex 基础

这部分内容会就 `xelatex` 或者 `latex` 的基础知识内容做全面的整理归类排版，争取做到言简意赅并面面俱到和实用性很强的手册性质。

## 3.latex 高级篇

这部分内容会就 `latex` 的一些较高级的知识做出说明，有时就会 `tex` 的一些命令做出说明。这部分内容主要以各个不同的专题议题的形式展开，不求全，但求就某一个实用的问题解释清楚。

#### **4. 其他问题的讨论**

这部分如标题所言，是我根据实际需要接触到的很有用的技巧，以问题的形式展开。这部分内容能够做到分解成为前面三部分内容的将会分解，觉得不好分解的就放到这里，这部分内容不应过分扩充。

#### **5. 后面的珍宝**

这部分会将本书涉及到的有用的资源收集起来给出下载链接或者打印出来。

# 目 录

前言	i
目录	iii
<b>I ubuntu 知识</b>	<b>1</b>
<b>1 准备工作</b>	<b>2</b>
1.1 通过 U 盘安装 ubuntu . . . . .	2
1.2 初入 ubuntu . . . . .	3
1.3 linux 最基本的命令介绍 . . . . .	4
1.4 通过 ppa 安装软件 . . . . .	5
1.5 ubuntu 软件推荐 . . . . .	6
1.6 ubuntu 下好的输入法 . . . . .	8
1.7 rime 输入法的备份和还原 . . . . .	9
1.8 把 QQ 拼音上你的词库导入 rime . . . . .	9
1.9 定制 rime 输出特殊符号 . . . . .	10
1.9.1 rime 基本情况说明 . . . . .	10
1.9.2 开始 DIY . . . . .	11
1.9.3 symbols.yaml 文件的说明 . . . . .	13
1.9.4 重新部署 rime . . . . .	13
1.9.5 具体输入符号的命令 . . . . .	14
1.10 配置 L <sup>A</sup> T <sub>E</sub> X 编写环境 . . . . .	14

1.11 ubuntu 下 $\text{\LaTeX}$ 新宏包的安装 . . . . .	14
1.12 texmaker 技巧 . . . . .	15
1.12.1 自动补全命令 . . . . .	15
1.12.2 自定义命令 . . . . .	15
1.12.3 保存你的设置文件 . . . . .	16
1.13 用 rime 快速输出 $\text{\LaTeX}$ 命令 . . . . .	16
1.14 其他 $\text{\TeX}$ 编辑器评测 . . . . .	16
1.15 gedit 的一些技巧简要说明 . . . . .	17
1.16 结合 github 进行项目管理 . . . . .	18
1.16.1 基本命令 . . . . .	18
1.16.2 日常改动提交流程 . . . . .	20
1.17 安装字体 . . . . .	20
1.17.1 找字体文件 . . . . .	20
1.17.2 放置字体文件 . . . . .	20
1.17.3 命令行安装字体 . . . . .	21
1.17.4 通过 fontmanager 安装字体 . . . . .	21
1.18 ubuntu 系统字体的配置 . . . . .	22
1.19 ubuntu 的备份和还原 . . . . .	23
 插图目录	 26
 表格目录	 27
 参考文献	 28

ubuntu 知识

# 准备工作

## 通过 U 盘安装 ubuntu

1. 在安装之前请先把硬盘中的资料做一些调整，空出一个大于 20G 的硬盘做将来要安装 **ubuntu** 根目录的地方。然后还需要一个大约为你内存两倍的硬盘分区等下要作为 **linux** 系统的 **swap** 交换分区。
2. 到 **ubuntu** 的官网上下载系统的光盘映像。
3. 用 **ultriso** 软件（其他具有类似功能的软件也行）（注意 **ultriso** 最好选择最新的版本，**ubuntu10.10** 之后的一定要用 9.3 版本之上的）将该光盘映像写入到你的 U 盘中去。
4. 重启计算机，**BIOS** 稍作改动使计算机变成从 U 盘启动。
5. 进入安装过程，其他过程都比较直观，就是硬盘分区设置上我们选择高级手动，然后将你分出来的那个 20G 硬盘作为／的加载点，并设置格式化成 **ext4** 日志系统（其他文件系统如 **ext3** 等也行）。然后设置交换分区，安装完成。

## 初入 ubuntu

- 设置 ubuntu 的 root 用户，这个以后可能会有用。按住 `ctrl+alt+t` 弹出终端，然后输入：

```
sudo passwd
```

设置好密码之后，以后输入：

```
su
```

就可以进入 root 账户了。

- 安装一个有用的工具：

```
sudo apt-get install nautilus-open-terminal
```

这个工具可以让你在视窗背景下鼠标右键之后打开当前目录下的终端，有时还是很快捷的，尤其是遇到一些怪怪的文件名的时候。

- 系统安装好之后第一件要做的事是选一个好的源，然后安装更新。我的 ubuntu 版本是 12.04，在右上角的哪里，软件更新，软件源设置，在下载自的哪里就是软件源的服务商，你最好还是自己搜索一个速度最快的源。然后在终端中执行以下命令来升级系统软件包：

```
sudo apt-get update （更新源）
```

```
sudo apt-get upgrade （更新源下已经安装了的软件）1
```

- 一些闭源的有用的东西：

```
sudo apt-get install ubuntu-restricted-extras
```

这个安装到 `adobe-flashplugin` 的时候会有点卡住，最好还是耐心点吧。或者在 ubuntu 软件中心（Ubuntu 额外的版权受限程序）中安装也是可以的。

---

<sup>1</sup>如果有很多软件需要升级的时候推荐使用命令：`sudo apt-get dist-upgrade` 这样不容易出错些。



- ubuntu 系统清理，运行以下两个命令即可：  
`sudo apt-get autoremove` （清理软件残余）  
`sudo apt-get autoclean` （清理缓存）。
- 删除通过 `apt-get` 安装的软件：  
`sudo apt-get remove` 要删除的软件名  
删除相应的软件配置文件：  
`sudo apt-get purge` 软件名
- 长按 `super` 键或者说是 `win` 键会弹出一些 ubuntu 桌面常用的快捷键，比如说 `ctrl+win+d` 是显示桌面，按住 `win+f` 搜索文件，`win+a` 搜索程序等。

## linux 最基本的命令介绍

**\$ 和 ~** 当我们打开终端的时候，看到一个美元 `$` 符号，如果我们输入 `su` 命令，然后进入 `root` 账户，看到开头有一个 `#` 符号，其中 `$` 表示普通用户，`#` 表示现在是超级用户。然后我们看到一个波浪号 `~`，这个波浪号的意思就是当前用户的个人主文件夹，比如现在我这里 `~` 的意思就表示目录 `/home/wanze`。

**cd** 一般 `dir` 或者 `ls` 之后列出该目录所包含的文件夹或者文件，然后执行如下命令：

`cd` 某个文件夹名

就进入这个文件夹了。如果 `cd` 一个文件就会报错。然后特殊的有：

`cd ~` 回到个人主文件夹

`cd .` 其中点表示当前目录

`cd ..` 表示返回上一级目录

**cd** 某个目录，比如 **cd / etc** 那么就直接跳到系统的 **/ etc** 目录下了。

**cp** 复制文件命令：

**cp** 要复制的文件 目标文件目录

**which** 查看系统某个命令的位置

**touch** 创建某个文件或者对已存在的文件更新时间戳

**rm** 删除某个文件，一些具体选项我就不罗嗦了，如果有不明了的请自己用 **-- help** 来查看帮助信息。比如这里的 **-r** 选项就可以强制删除一个目录下所有的文件。

**mkdir** 创建文件夹

**rmdir** 删除文件夹

**mv** 移动某个文件或者重命名某个文件

## 通过 ppa 安装软件

ppa 也就是软件源，一般推荐 ppa 安装软件即该软件还处于活跃开发期中，所以推荐尽快更新到官方最新的版本。过程就是添加 ppa 源，然后用前面讲的 **apt -get upgrade** 来更新源下的软件。添加 ppa 源的格式如下：

**sudo add-apt-repository ppa:** 后面就是什么官方源地址了

然后更新源下的软件列表：

**sudo apt-get update**

接下来就是通过 **apt-get install** 来安装该软件了。**upgrade** 是更新源下已经安装了的软件，所以第一次装还是要用 **install** 命令。<sup>2</sup>

---

<sup>2</sup>[how-install-any-software-ubuntu-ppa](#)

## ubuntu 软件推荐

以下编号：Ⓐ 表示可以从 **ubutnu** 软件中下载到，Ⓑ 表示可以从终端 **apt-get** 命令中下载到，Ⓒ 表示推荐从 **ppa** 安装。

**新立得软件包管理器** 这个有时安装一套软件组合很有用的。Ⓐ

**chromium** 不清楚和 **google-chrome** 的区别。它的插件很方便，有的时候网页显示相比较于 **firefox** 不会出错些。Ⓐ

**chromium** 插件推荐：

**Daum Equation Editor** 本文不会讨论数学公式的输入问题，这里有个简单的解决方案，就是在这个插件里面编辑公式，下面就有对应的 **L<sup>A</sup>T<sub>E</sub>X** 的代码。

**代理助手** 好吧，我讲这个完全是因为天朝的防火墙，不学会翻墙严重干扰我们搜索信息。具体设置是 **http** 代理填写相关地址，**china list** 这个选项也勾上吧。下面还有详细说明。

**翻墙软件** 要搜东西真的要用 **google**，在天朝你懂的，因为要跑防火墙慢死了。轮子的自由门还可以。<sup>3</sup>翻完墙之后到弹出来的那个网站或者 **voa** 网站哪里多下几个类似的翻墙软件吧，有备无患。**freegate** 在 **ubuntu** 下需要使用 **wine** 模拟，在模拟前先用 **winetricks** 装好 **mfc42.dll**，然后就行了。**freegate** 进去之后在通道哪里选择经典模式。**google-chrome** 那边的设置在代理助手哪里，就是使用 **freegate** 提供的那个 **http** 地址加上端口号即可——在内容那一页。一般代理助手就点系统自动连接吧，毕竟是免费代理，别想质量有多好。

---

<sup>3</sup>[下载 freegate 的链接](#)

**texmaker** 写  $\text{\LaTeX}$  非常好的界面环境，和 windows 下的 **texstudio** 类似，但是感觉界面更加的漂亮。④

**shutter** 一个非常好的截图软件。**ubuntu** 系统自带的截图软件截完图不能拖动改变大小，这个可以。④

**ubuntune** 在 **ubuntu** 系统下很好的文档同步软件，在手机上安装之后可以方便进行某些文档的传输。④

**wine** 虽然 **wine** 软件一直在更新，但是老实说总感觉不是十分令人满意。这里提到 **wine** 是因为前面谈到的翻墙软件需要用 **wine** 来模拟。**wine** 的安装不同于其他软件，尽量下载最新版本的吧。到官网上查看最新的稳定版本号，比如 **wine1.6**。然后在终端上通过 **ppa** 来安装，**ppa** 地址见脚注。④

**workrave** 人们对久坐的危害总是低估了，这个软件帮助提醒人们做一段时间之后最好起来活动活动。④

**Gcolor2** 这个小软件取色，RGB 转换等很有用。④

**wps for linux** 目前写毕业论文不得已还是要用 **doc** 格式，而在 **ubuntu** 下我试过很多解决方案，比如 **wine**，**libreoffice**，或者网上的在线编辑。结果都不尽如人意。因为我需要的是最好完全全兼容 **microoffice2003** 的格式。金山公司的这个 **wps for linux** 目前还在开发中，但是效果还是可以了。下载地址见脚注。⑤

其他还有很多有用的软件看官自己探索吧。

---

<sup>4</sup>[ppa:ubuntu-wine/ppa](#)

<sup>5</sup>[wps for linux](#)

## ubuntu 下好的输入法

更多信息请参考 [rime 官方 wiki \(1\)](#)。还有一种 [ppa 安装方法](#)，[ppa 地址](#)见尾注 [\(2\)](#)。

1. 安装 `ibus-rime`:

```
sudo apt-get install ibus-rime
```

2. `ibus` 输入框不见了:

```
ibus-daemon -drx
```

3. 重启 `ibus`: 有时 `ibus` 会出现一些奇怪的问题，在终端运行下面命令重启 `ibus` 也许会解决:

```
killall ibus-daemon
```

```
ibus-daemon -d
```

4. `Rime` 输入法的其他设置: 在输入框调到 `Rime` 的状态的时候，按一下 `F4`，就弹出一些设置选项。

5. 用 `Rime` 输入常见的特殊符号: 比如我要输入 `*`，按下 `shift` 和数字 `8` 按键，就出来一些选项供你选择。其他很多符号类似，具体请参考官网上的说明。

6. `Rime` 输入法是架构在 `ibus` 框架之上的，所以在系统输入法首选项哪里，可以设置 `Rime` 候选项是横着显示或者竖着显示，还有下面可以自定义显示的字体和字体大小。推荐用 `sans-serif` 字体。然后字体大小要比你输出之后的效果稍微大一点，这样醒目些。

## rime 输入法的备份和还原

输入法用久了，你的自造词还有你输入相关词的词频这些信息是非常有用的，能够极大地提高你的工作效率。rime 备份的核心词是命令：**rime\_dict\_manager**。你需要在 rime 的用户配置文件夹里面打开终端，然后输入这个命令，你就可以看到提示信息了。

首先输入 **-l** 选项，你可以看到现在已经有的词典，比如 **luna\_pinyin**。然后用 **-s** 同步，这样你就会看到一个 **sync** 文件夹，里面放着快照文件。**-b** 是选择性备份某一个词典。**-r** 是还原快照文件。**-e**，**-i** 是 **txt** 格式操作，**txt** 保存数据会有点损失。将用户 **rime** 配置文件夹整个复制到同步盘即可。

## 把 QQ 拼音上你的词库导入 rime

1. 在 windows 下在 QQ 拼音设置里面找到那个导出词库的命令。就是一个 **txt** 文件。
2. 将这个 **txt** 文件另存为 **utf-8** 编码
3. 进入 **linux**（以下 **windows** 用户原理类似）找个在线的简繁转换网站，如果你使用的是简体略过这步。然后转换成为繁体，由于词库比较大，建议你耐心点，推荐笨笨网站简繁转换，不容易卡死。实在不行就只好下载软件了。我的一万多条还是能够转换出来。
4. 打开 **libreoffice**，将前面的 **txt** 文件，复制，粘贴。然后会弹出一个选项，分隔符注意勾选空格。
5. 前三列有用，后面的意义不大，统统删除。

6. 按 `ctrl+f`，下面弹出查找于替换。输入，后面输入空格全部替换。输入' 后面输入空格全部替换。
7. 将第一列第二列位置互换，就是剪切粘贴操作。
8. 全部选择前三列，复制，粘贴到文本。
9. `rime_dict_manager -i luna_pinyin` 上面谈及的你创建的文本命令。
10. 注意上面要在 `rime` 用户目录下操作即`.config/ibus/rime`。
11. 查看 `dict_manager -b luna_pinyin`

## 定制 rime 输出特殊符号

我希望通过 `rime` 输入法能够快速地将 `Unicode` 中的符号打出来。本文主要关注这个问题，其他定制请参阅官网 [wiki](#)。

### rime 基本情况说明

`Rime` 的 `ubuntu` 版本叫做中州韵，不管这么多，一般称作 `ibus-rime`，下面简称 `rime`。下面的讨论只使用于 `ubuntu`，我目前的版本是 13.04，12.10 试过没有问题。

`rime` 的程序在哪里我不关心，在 `ubuntu` 下所有程序的配置是用户配置优先级高于程序自带的配置，所以我们在 `rime` 的用户文件夹里面 `DIY` 就是了。`rime` 的用户资料夹在：

`~/.config/ibus/rime`

进入这个文件夹之后，我们可以点击看以下里面的内容，这些文件情况说明如下：

**cangjie5.schema.yaml** 仓颉五代配置文件。我用的是朙 (míng) 月拼音，这个文件就不管了。

**default.yaml** 这个文件存放着输入法的一些全局设定，重新部署 rime 时会重新生成。

**installation.yaml** 这个文件存放着 rime 的安装信息，不用管它。

**luna\_pinyin.schema.yaml** 这个是默认的朙月拼音的配置文件，要仔细看看。

**luna\_pinyin\_fluency.schema.yaml** 这个是朙月拼音语句流的配置文件，我不关心。

**luna\_pinyin\_simp.schema.yaml** 这个是明月拼音简化字的配置文件，也就是输入法切换到简体字输入之后起作用的。

**luna\_pinyin\_tw.schema.yaml** 这个是朙月拼音台湾正体的配置文件，没用过。

**symbols.yaml** 这个是 rime 自带的一个输出 symbol 的配置文件，但是奇怪的是刚开始并没有配置好。

**user.yaml** 这个是用户状态信息。

## 开始 DIY

其中官网 wiki 中讲了一些 default.custom.yaml 的定制，那些我似乎都没有需要，然后略过。现在主要讨论 luna\_pinyin.custom.yaml 的配置问题。具体配置如下：



```

1 patch:
2     punctuator:
3         import_preset: symbols
4     recognizer:
5         import_preset: default
6         patterns:
7             reverse_lookup: "'[a-z]*'?$"
8             punct: "^/[a-z\\[\\]]*$"

```

---

我想你已经看到了这些配置文件都是.yaml 后缀，它们是用一种什么 yaml 数据描述语言写的，类似于 XML 的标记语言。然后里面还有一些正则表达式的东西。关于这些内容我是能够略过就略过吧。

刚开始我写这个 patch 的时候总是不成功，然后看到佛振的那个帖子，链接到另外一个源码上。[请参看这个网站](#)。然后直接修改 luna\_pinyin.schema.yaml 文件，就是把 default 改成 symbols，也就是指向那个 symbols.yaml 文件，然后将 patterns 的 punct: 那一行复制过去。就发现可以了。

现在 patch 可以正常工作了，原因是之前我的缩进不正确，由于我对 yaml 语言不清楚，就此打住。

其中 punct 那一行涉及到的正则表达式知识有符号 ^ 表示匹配开始，\$ 表示匹配结束，[] 和里面的内容一起表示一个字符，这个字符可能是 a-z 所有的小写字母，然后我想加上符号 [和]，因为我想新建一个命令/[ ] 就能输入很多形式的括号。最后发现这个问题还有点小麻烦，[请参看这个网站](#)。他解释了最好前面加上两个 \，然后才能更好的工作。最后那个 \* 表示重复零次 h 或者更多次。

(3)

## symbols.yaml 文件的说明

我在 github 网站上新建了一个项目。里面有我编辑的 symbols.yaml 文件。现在将我所做的修改工作简单说明如下：

前面是全角和半角部分，如果你有需要可以自己定制下，这里我略过。然后看到 symbols: 哪里，开始自己定制命令。前面那个代码我们看到了匹配是以/开始的，所以下面所有的命令都要以符号/开始，如果你想有其他模式修改正则表达式就可以了，比如说可以换成命令以符号\开始。

yaml 的缩进表达我不太清楚，实际上我不太喜欢那种缩进的语法。yaml 还提供了另外一种语法形式。就是一个系列可以用一个方括号明确标识出来。<sup>6</sup>在这里格式很简单，就是建立了很多符号调出命令，对应的后选词选项用方括号包围起来。

然后还需要一提的就是 yaml 语言支持锚点，意思就是某一个变量，跟着 & 任意的名字，后面就可以用 \* 那个名字来表示那个词条了，这个有点类似于取别名的意思。anyway，用这样的方法可以给你建立的命令取几个别名。

## 重新部署 rime

最后将上面两个文件放入用户 rime 配置文件夹哪里，然后删除 default.yaml 文件，然后运行ibus-daemon -drx。等着 rime 重新部署完毕即可。

---

<sup>6</sup>请参看[这个网站](#)

## 具体输入符号的命令

现在我在 `rime` 下输入 `/gcs`，就会弹出很多符号。类似的还有很多，就不一一介绍了。

## 配置 $\text{\LaTeX}$ 编写环境

1. `sudo apt-get install texlive` (下载安装 `ubuntu` 下有名的 `texlive`)
2. `sudo apt-get install texlive-full` (下载安装 `texlive` 的各个包)
3. 在 `ubuntu` 软件中心中下载安装 `texmaker` 软件。
4. `ibus` 似乎在这里面有点小问题，有点不稳定。在终端中运行 `sudo apt-get install ibus-qt4` 则问题解决。
5. 在 `texmaker` 的选项和配置 `texmaker` 里面，设置快速构建，点上用户自定义那一栏，然后输入如下命令：  
`xelatex -interaction=nonstopmode %.tex|`  
这是使用 `xelatex` 来对目标 `tex` 文件进行编译，而不是传统的 `latex` 或者 `pdflatex` 方式，之所以这样是因为多方对比之后，觉得其在字体处理方面是未来的趋势。

## ubuntu 下 $\text{\LaTeX}$ 新宏包的安装

安装 `texlive-full` 之后，如果还遇到没有的宏包，可以先到 CTAN 官网上下载到这个宏包之后，然后将这个宏包解压到系统目

录：

`/usr/share/texmf/tex/latex` 里面即可。

当然你也可以在另外一个文件夹里面，这里必须是你的主文件夹下，新建一个文件夹 `texmf`，然后里面新建一个 `tex`，然后再新建一个目录 `latex`，然后在这个 `latex` 里面放着你下载下来的宏包。具体比如说有一个宏包名字叫做 `config`，那么 `latex` 下面就是 `config` 文件夹，然后里面就是 `config.sty` 文件。你自己写的宏包扔进去一样有效。唯一要额外做的操作就是在 `texmf` 目录之下运行命令：

```
sudo texhash
```

让 `texlive` 把这个目录也加入搜索范围。

## texmaker 技巧

### 自动补全命令

在菜单里找到用户自定义的 `customize completion` 也就是自动完成，里面加入你想要的命令。比如：

`\textbackslash` 然后点击 `add`，这样以后想输入显示命令前面的那个斜线的时候会方便点。如果括号里面加入 `@` 符号，那么就会出现类似系统自带命令 `\section{•}` 的那个黑点 `•`。

### 自定义命令

在菜单哪里用户自定义，你看到可以用户自定义命令，填好之后就是快速构建下面那些备用的 `1:2:3:` 哪里将成为有意义的命令。

## 保存你的设置文件

在选项哪里有保存设置文件的功能，主要是自定义命令，自动补全命令等可以保存下来。

## 用 rime 快速输出 $\text{\LaTeX}$ 命令

具体效果就是我按下 `/tex` 等命令，就会弹出很多  $\text{\LaTeX}$  命令，这个也是修改 `symbols.yaml` 文件来达到的，也是在我新建的那个 [github 项目](#) 里面，我做了很多优化工作，有兴趣的可以研究下。

(4)

## 其他 $\text{\TeX}$ 编辑器评测<sup>7</sup>

**emacs** 安装 `emacs24` 之后，用 `package manager` 安装 `auctex` 宏包。然后一些基本的 `latex` 编辑器功能都有了。稍微熟悉下就行了。唯一遗憾的是没有那种左侧很方便的目录导航功能。然后 `emacs` 的各种快捷键让我压力好大。

**winefish latex** 太高端了，刚进去一片空白？

**gedit** `gedit` 安装好 `latex` 的插件之后也很好。因为 `latexila` 编辑器在编译大文件的时候速度好慢，而 `gedit` 是直接调用终端模式所以没有损伤的。值得一提的是 `latexila` 编辑器的一些功能在 `gedit` 里面可以通过外部工具这个插件自己编写终端命令来实现。

**texworks** 没怎么评测。

---

<sup>7</sup>虽然 `texmaker` 软件也有些小缺陷，但综合起来我觉得是最棒的。

**kile** 没怎么评测。

**texstudio** 我刚进去什么都没动，它好像中毒一样到处乱操作。。

**gummi** 打开 tex 文档或者做些小的测试还是可以的，但是似乎不支持 include 文档。

**latexila** 这个软件刚进去我就感觉设计理念非常的简洁，和 gedit 一样。然后我特别在意的文档结构图显示，语法染色都有。

(5)

## gedit 的一些技巧简要说明

在编辑 → 首选项那里请安装好 latex 插件和外部工具插件，恩，代码注释插件和文件浏览器插件推荐。

我在这里重点介绍一下外部工具插件的用法。看到工具 → 外部工具。请选择最下面的那个管理外部工具，看到在此处打开终端，gnome-terminal 后面去除掉，然后就能正常工作了。(6)

现在我们新建一个命令，名字叫做 xelatex，具体内容如下：

---

```
1#!/bin/sh
2filename=$GEDIT_CURRENT_DOCUMENT_NAME
3shortname='echo $filename | sed 's/\(.*\)\\.tex$/\1/'
4xelatex -interaction batchmode -src $filename
```

---

那么这个小工具就实现 xelatex 编译功能了。

然后我们也可以再新建一个小工具，名字叫清理：

---

```
1#!/bin/sh
2filename=$GEDIT_CURRENT_DOCUMENT_NAME
3shortname='echo $filename | sed 's/\(.*\)\\.tex$/\1/'
4rm -f $shortname.aux $shortname.ent $shortname.out
5$shortname.lot $shortname.idx $shortname.lof
6$shortname.ilg $shortname.ind $shortname.log
7$shortname.toc $shortname.bbl $shortname.blg
```

---

这样运行它就可以清理临时文件了。

## 结合 github 进行项目管理

以下主要参考[这个网站 \(7\)](#) 到 github 上注册, 创建新的项目, 还有安装 git 软件 (ubuntu12.10 自带的有) 都很简单的, 我就不多说了。下面就 git 命令使用的基本流程说明如下:

### 基本命令

**远程仓库文件到本地** 网上创建项目之后, 你需要将网上的存档下载到本地, 在你希望下载的地点, 打开终端:

```
git clone https://github.com/a358003542/xelatex-guide-book.git8
```

git init 命令用于本地创建的文件夹上传到远程仓库, 我估计 git clone 下来的仓库文件已经索引了, 但 git init 命令要不要用还不确定。这里认为不需要用。 (8)

---

<sup>8</sup>后面的这个链接地址在你创建的项目的右下角哪里, 写着 HTTPS clone URL

**本地仓库文件进入索引** 下载下来的本地仓库文件进入 git 的索引，该文件夹内的所有文件都进入索引则在终端中输入如下命令：

```
git add .
```

因为我们在 github 创建项目的时候已经创建了一个配置文件，比如我选择的是 latex 语言，然后它会自动会处理将某些文件不上传。

**将索引中改动的文件提交到 head** 不太清楚这个索引，head 具体是什么意思，anyway，过程就是这样的。

```
git commit -m '2013-08-25:19:00'
```

后面的文字串等下在 github 网站中会看到的，表示这个文件的标示符吧，你也可以取其他的名字。我就喜欢取当时的日期和时间了。

**将 head 中改动的文件更新到远程仓库** 第一次你需要给你的远程服务器取个简单点的名字：

```
git remote add origin https://github.com/a358003542/xelatex-guide-book.git9
```

然后以后都可以用这个简单的命令来更新了：

```
git push origin master10
```

**远程仓库的改动更新到本地** 需要说的是 git 对文件的操作是合并式的，也就是只是替换最新改动的文件。

```
git pull origin master
```

---

<sup>9</sup>这里的地址和前面的一样

<sup>10</sup>这里的 origin 的意思前面说了，master 是远程仓库默认的一个分支，后面会讲到你可以创建其他的分支。



## 日常改动提交流程

一般情况下就用前面讲的的三步，`add .` `commit -m` 然后 `push`。这是基本的日常维护提交流程。

如果你在网站上对远程仓库做了一些修改，记得先用 `pull` 命令将远程仓库的改动更新到本地。

## 安装字体

### 找字体文件

如果你装了 windows 系统，那么你可以到 windows 下 copy 这些字体文件。比如 windows 常用的宋体，times new roman 等，在 C 盘的 windows 的 fonts 文件夹里面。本文用的就是 adobe 中文系列：adobe 宋体 std，adobe 黑体 std，adobe 楷体 std。很奇怪，在 pdf 上我觉得这几个字体感觉很好，但是在屏幕上就觉得不太好了。

### 放置字体文件

推荐都放在 ubuntu 的主目录的 `.fonts` 文件夹里面<sup>11</sup>，如果没有请新建一个。这是通常默认用户新加字体放置的目录。当然你也可以放在其他目录里面，比如你的同步盘里面，然后用 `font-manager` 安装字体也是可以的。<sup>12</sup>

---

<sup>11</sup>这是一个隐藏目录

<sup>12</sup>这是本文推荐的方式，安装卸载字体都方便些。

## 命令行安装字体

运行命令：

```
fc-cache -f -v
```

字体就安装好了，如果你要看现在你的系统上有那些可用的中文字体，在终端运行命令<sup>13</sup>：

```
fc-list :lang=zh | sort >ziti.txt
```

打开 `ziti.txt`，里面就是你的可用中文字体的信息，比如：

```
/home/wanze/.fonts/simsun.ttc:
```

```
宋体,SimSun:style=Regular
```

其中第一个是字体文件所在的目录，第二列信息是可以调用的名字，有宋体和 SimSun。

## 通过 fontmanager 安装字体

你可以安装其他软件来安装和管理字体，比如 `fontmanager`：

```
sudo apt-get install font-manager
```

这个软件查看安装卸载或者禁用某些字体都很方便的，需要提醒的是这个软件占用了默认的用户配置文件 `.fonts.conf`。然后你的字体 DIY 需要到 `~/.config/font-manager/local.conf` 哪里去设置。这个下面会讲到。

---

<sup>13</sup>| 表示 linux 命令中的通道，第一个命令的输出信息流会流向 `sort` 命令，排序之后重定向到 `ziti.txt` 文件里面。然后终端的数据就保存在这个文件里面了。

## ubuntu 系统字体的配置

不是特别难看的情况就没必要改动系统字体，因为我们不要低估了人眼的适应能力。这里的配置主要是指由于系统升级带来的字体的改变，特别是中文字体的改变，手动配置将其固定下来。

(9)

---

```

1 <?xml version="1.0"?>
2 <!DOCTYPE fontconfig SYSTEM "fonts.dtd">
3 <fontconfig>
4
5 <match target="font">
6     <edit name="rgba" mode="assign"><const>rgb</const></edit>
7 </match>
8
9 <match>
10     <test name="lang" compare="contains"><string>zh</string></test>
11     <test name="family"><string>serif</string></test>
12     <edit name="family" mode="prepend"><string> 微软雅黑</string></edit>
13 </match>
14
15 <match>
16     <test name="lang" compare="contains"><string>zh</string></test>
17     <test name="family"><string>sans-serif</string></test>
18     <edit name="family" mode="prepend"><string> 微软雅黑</string></edit>
19 </match>
20

```

```

21 <match>
22     <test name="lang" compare="contains"><string>zh</string></test>
23     <test name="family"><string>monospace</string></test>
24     <edit name="family" mode="prepend"><string> 微软雅黑</string></edit>
25 </match>
26
27 </fontconfig>

```

---

有很多内容没有深究，第一个是打开 **rgba** 模式，优化液晶显示的。然后下面就是对三大字族设置，如果是 **zh** 中文的话那么就 **prepend** 也就是插入微软雅黑，也就是在搜索队列中微软雅黑优先级最高。这个可以通过 **fc-match -s serif** 等来查看。然后其他字体设置不想涉及了，只希望他们能够稳定下来。感觉设置雅黑字体了，系统的主题换为 **Radiance** 更好看些。然后用 **font-manager** 针对微软雅黑高级设置加上 **AA** 和 **AH**。一个是反锯齿一个是自动粗细设置吧。就这样了。 [\(10\)](#)

## ubuntu 的备份和还原<sup>14</sup>

首先是利用系统自带的备份软件将 **home** 文件夹里面的一些内容备份好，注意配置文件夹排除法则。

备份步骤简介如下：

1. 备份你的 **PPA**，也就是你通过 **PPA** 装的软件。有的可能都没有通过 **PPA** 装过软件，那么这一步和下面的备份 **PPA** 的 **key** 都

---

<sup>14</sup>以下主要参考了[这个网站](#)

可以省略。命令如下：

```
sudo cp -r /etc/apt/sources.list.d ~/Nutstore/ubuntu-config
```

2. 备份你的 PPA 的 key:

```
sudo apt-key exportall > ~/Nutstore/ubuntu-config/myppakey
```

3. 通过新力得软件包管理器生成你安装的所有软件包的列表：选择文件 F→ 将标记的项目另存为 A→ 然后下面保存全部状态，不仅仅是变更选项勾上 → 然后选择一个地址取一个有意义的名字。

如果重装系统了，还原步骤相当于前面备份步骤的反向操作吧：

1. 还原你的 PPA: 这里就不写命令了，和上面类似也是 `cp -r`，不同的是现在将你保存的 PPA 还原到新系统的 `etc` 的 `apt` 目录下。

2. 还原你的 PPA key，假设对照上面的情况，我们有命令：

```
sudo apt-key add ~/Nutstore/ubuntu-config/myppakey
```

3. 通过新力得软件包全部安装你之前保存的软件包列表：选择文件 F→ 读取标记的项目 R→ 然后类似正常的通过新力得安装软件包的步骤，点击应用即可。
4. 通过上面的步骤，你的新系统的所有软件全新安装好了，至于其他配置文件前面说了一些你在意的配置保存，然后选择相应的位置，一般在 `.config` 文件夹里面，复制进去就行了。

## 注释和参考资料

- (1) [rime 官方 wiki](#)
- (2) ppa 地址是: [ppa:lotem/rime]
- (3) 你可以到这个网站继续学习正则表达式: [正则表达式 30 分钟入门教程](#)
- (4) 在 rime 中文模式下, 直接按 enter 键输出的也是英文, 所以一些简单地命令直接输入也是很快捷的。
- (5) latexila 编辑器使用 xelatex 引擎生成文档, 一样要自己编写一个生成命令, 在创建 → 首选项那里 → 标签就写 xelatex, 写其他的也可以。扩展名写.tex, 然后命令和 texmaker 一样写上这个: `xelatex -interaction=nonstopmode $filename`。
- (6) 本信息来自 ubuntu13.04, gedit 版本号为 3.6.2。
- (7) 在项目网站右下角 settings 哪里进去有很多项目管理内容, 其中最下面有删除项目的功能, 请慎重使用。
- (8) 参考了[这个网站](#)
- (9) 这个代码主要参考了[这个网站](#)
- (10) chromium 字体也都设置成为微软雅黑吧, 然后我感觉页面稍微放大点更好看。

# 插图目录

# 表格目录



原谅我，我对参考文献的格式美化比较困惑。

## 参考文献

- [1] 有名的《一份不太简短的  $\text{\LaTeX}2\text{e}$  介绍》，原版作者：Tobias Oetiker, Hubert Partl, Irene Hyna 等，版本：2001-08-09，中文翻译：CTeX 用户小组，版本：2002-05，pdf 下载链接：<http://www.ctan.org/tex-archive/info/lshort/chinese>。
- [2] 《大家来学  $\text{\LaTeX}$ 》，原版作者：李果正，版本：2004-03-08，网站链接：<http://edt1023.sayya.org/tex/latex123/>。
- [3] 维基图书 latex 篇英文版，版本：2013-08，网站链接：<http://en.wikibooks.org/wiki/LaTeX>。这本书甚至都没有 cite 它一下，因为不知道插在哪里，几乎到处都有对它的引用，谢谢所有 wikibook 的编写者。
- [4] using boxes and glue in  $\text{\TeX}$  and  $\text{\LaTeX}$ ，原版作者：Nelson H. F. Beebe，版本：2009-03-28，pdf 下载链接：链接地址不好复制，请 google 搜索书名之。
- [5] The Latex Companion