

# 信息论及其应用实验一：数据压缩

## 使用手册

史泽宇

2020 年 4 月 3 日

## 1 概述

本实现使用 Rust 程序设计语言完成了 LZ 编码, Huffman 编码以及不完美的算术编码, 总共代码行数为 1012 行。实现包括了一个简单易用的命令行接口, 可以在命令行下调用本程序, 以下是使用说明。

## 2 使用手册

### 2.1 主界面

直接使用命令 `assignment_big`, 或者提供 `-h, -help` 参数可以获取程序的使用说明

```
assignment_big 0.1.0
yuyy <lancelot96@protonmail.ch>
```

USAGE:

```
assignment_big <SUBCOMMAND>
```

FLAGS:

```
-h, --help      Prints help information
-V, --version   Prints version information
```

SUBCOMMANDS:

```
arithmetic
help      Prints this message or the help of the given subcommand(s)
huffman
lz
```

主要提供了三个子命令, `lz`, `huffman`, `arithmetic`, 分别表示使用哪种编码算法。每个子命令后都带有几个选项, 其中共有的几个选项是 `-source <source>`, `-target <target>`, `<-encode|-decode>`。source

表示输入文件路径，target 表示输出文件路径，encode 表示对输入文件进行编码，decode 表示对输入文件进行解码。

## 2.2 lz 子命令

使用 assignment\_big lz 调用 lz 子命令，表示使用 LZ 编码算法

error: The following required arguments were not provided:

```
--alphabet <alphabet>
--letter-code-length <letter-code-length>
--prefix-code-length <prefix-code-length>
--source <source>
--target <target>
<--encode|--decode>
```

USAGE:

```
assignment_big lz --alphabet <alphabet> --base <base> --letter-code-length
<letter-code-length> --prefix-code-length <prefix-code-length> --source <source>
--target <target> <--encode|--decode>
```

For more information try --help

对于编码模式，只需要另外提供字母表参数即可，如

```
assignment_big lz --encode --source data.txt --target lz_coded \
--alphabet alphabet.txt
```

对于解码模式，除了需要提供字母表参数，还需要提供编码长度

```
assignment_big lz --decode --source lz_coded.txt --target lz_decoded.txt \
--alphabet alphabet.txt --letter-code-length 7 --prefix-code-length 11
```

## 2.3 huffman 子命令

使用 assignment\_big huffman 调用 huffman 子命令，表示使用 Huffman 编码算法

error: The following required arguments were not provided:

```
--codebook <codebook>
--source <source>
--target <target>
<--encode|--decode>
```

USAGE:

```
assignment_big huffman --base <base> --codebook <codebook> --source <source>
```

```
--target <target> <--encode|--decode>
```

For more information try --help

对于编码模式，不需要提供任何额外参数

```
assignment_big huffman --encode --source ./data.txt --target huffman_coded
```

对于解码模式，还需要提供编码之后产生的编码映射

```
assignment_big huffman --decode --source ./huffman_coded.txt --target \
huffman_decoded.txt --codebook huffman_coded.cb
```

## 2.4 arithmetic 子命令

使用 assignment\_big arithmetic 调用 arithmetic 子命令，表示使用算术编码算法

```
error: The following required arguments were not provided:
```

```
--distribution <distribution>
--source <source>
--target <target>
<--encode|--decode>
```

USAGE:

```
assignment_big arithmetic --base <base> --distribution <distribution>
--source <source> --target <target> <--encode|--decode>
```

For more information try --help

对于编码模式，不需要提供任何额外参数

```
assignment_big arithmetic --encode --source data.txt --target arithmetic_coded
```

对于解码模式，还需要提供信源的概率分布

```
assignment_big arithmetic --decode --source ./arithmetic_coded.txt \
--target arithmetic_coded_decoded.txt --distribution distribution.prob
```

但是需要注意，本程序实现的算术编码不能处理较长内容的输入，具体可以参考测试文件中的内容。