搜索引擎架构

离线部分

1、词典文件、词典索引文件

可以作为一个单独的程序

输入的是语料，输出的就是词典文件与词典索引文件

生成4个文件

词典文件的产生，词典索引文件的产生，封装成类

英文

停用词　去掉停用词　得到英文词典文件

26个字母建立索引　得到英文索引文件

中文

分词，cppjieba　cppjieba的安装与使用（重点）　cppjieba只需要初始化一次，初始化很浪费时间

停用词　去掉停用词　得到中文词典文件

以每个汉字建立索引　得到中文索引文件

可以将文件名字存起来　opendir/chdir/readdir

如果没有索引的话，就需要遍历词典文件，然后比较两个词之间的相似度(最小编辑距离)

2、网页库、网页偏移库、倒排索引库

可以作为一个单独的程序

输入的是网页信息，输出的是网页库、网页偏移库、倒排索引库文件　生成3个文件

网页库、网页偏移库、倒排索引库的创建，封装成类

生成网页库，也就是之前rss作业使用，但是这里会涉及一个问题，文件名字很多（注意）　可以将文件名字存起来　opendir/chdir/readdir

在线部分

1、作为一个独立的程序

2、需要将离线部分的7个文件作为在线部分的输入

3、提供关键字推荐服务

多个与查询词相关联的词组的集合

先走缓存，缓存没有的话走查询模块

客户端发送关键字过来之后，再去索引文件中进行查找　最小编辑距离算法（难点）

英文一个字符是一个字节

utf8下面，一个字符是三个字节，注意处理

如果查询的时候，既有中文又有英文，怎么处理?

选取完成之后，需要将满足条件的几个候选词放在优先级队列里面进行存储

4、提供网页搜索服务

网页标题

网页摘要

URL，链接

通过关键词，找到符合条件的一篇篇网页信息

可能有很多篇符合要求的文章，可以做分页功能

也可以设置缓存系统

5、序列化与反序列化

C++中是使用对象存储　序列化　字符串json

发送到客户端之后，语言不同　反序列化　js对象

客户端与服务器使用json进行传输

协议解析类

到底走的是关键字推荐模块　按照PDF中的解析，自己定义协议

还是走网页搜索模块

客户端

实现客户端，与服务器之间进行交互

可以直接使用nc进行测试，或者将之前的客户端程序拿过来直接用

手机APP，Web