**云数据管理II 《检索结果分类/聚类》**

2015080121 软工52 李在弦

2015013161 软工51 魏佳辰

**实验环境**

操作系统：Windows

IDE：PyCharm

系统结构：b/s 结构

Web框架：Django 2.1.2

后端编程语言：Python 3.6

数据库：MongoDB

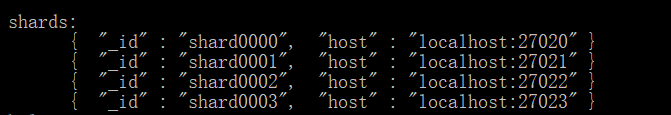
**数据库**

* ***MongoDB介绍***

MongoDB 是一个基于分布式文件存储的数据库，它是一个面向文档存储的数据库，操作起来比较简单和容易。如果负载的增加（需要更多的存储空间和更强的处理能力） ，它可以分布在计算机网络中的其他节点上这就是所谓的分片。

* ***分布式存储***

当MongoDB存储海量的数据时，一台机器可能不足以存储数据，也可能不足以提供可接受的读写吞吐量。这时，我们就可以通过在多台机器上分割数据，使得数据库系统能存储和处理更多的数据。分片结构端口分布如下：



Config server: 27100

Route Process: 40000

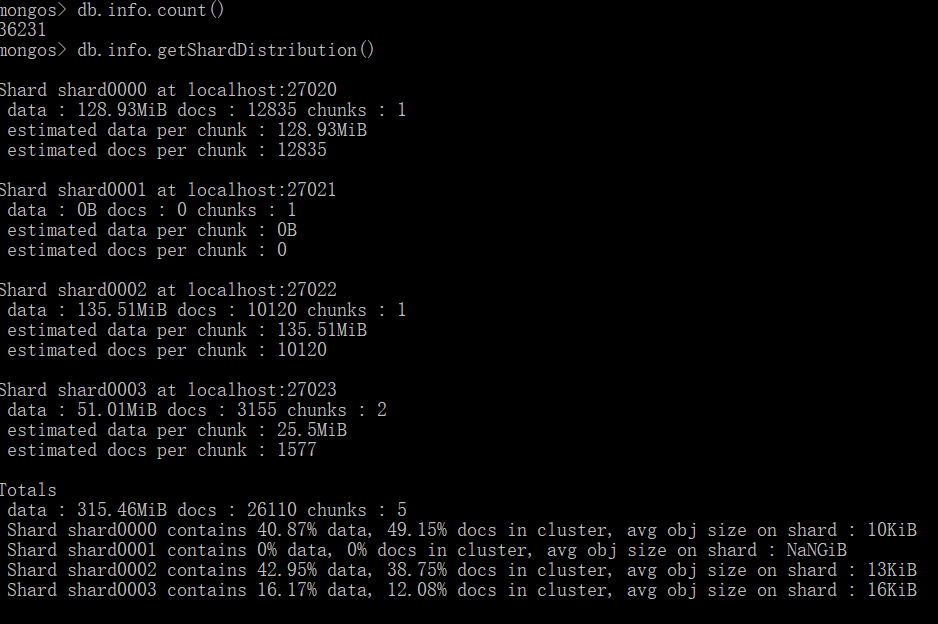
启动shard server, config server和route process之后，我们对分片进行配置

首先通过db.runCommand({addShard:”localhost:27020”})等命令添加分片

然后通过db.runCommand({enablesharding:”docdb”})来设置分片存储的数据库

再通过db.runCommand({shardcollection:”docdb.info”,key:{time:1}})来分片info这集合

然后我们通过mongoimport导入处理好的json文件来插入数据，最后数据库自动分片结果如下



**算法说明**

* ***网络爬虫***

从10万多个单词词库中选取1700多个单词，通过request库访问百度首页，输入单词，爬取前三个搜索结果的URL，标题等信息。

访问每个URL，通过BeautifulSoup库分析HTML信息，排除不需要的标签，获取文本信息。

最后把URL，标题，文本内容等信息保存到txt文件。

最终爬取到5222个文档信息。

* ***分词，提取关键词，获取词干，建立“单词-文档权值矩阵”和倒排索引***

读取刚刚保存的所有文档的文本信息，提取关键词，进行分词（只对关键词进行分词）。

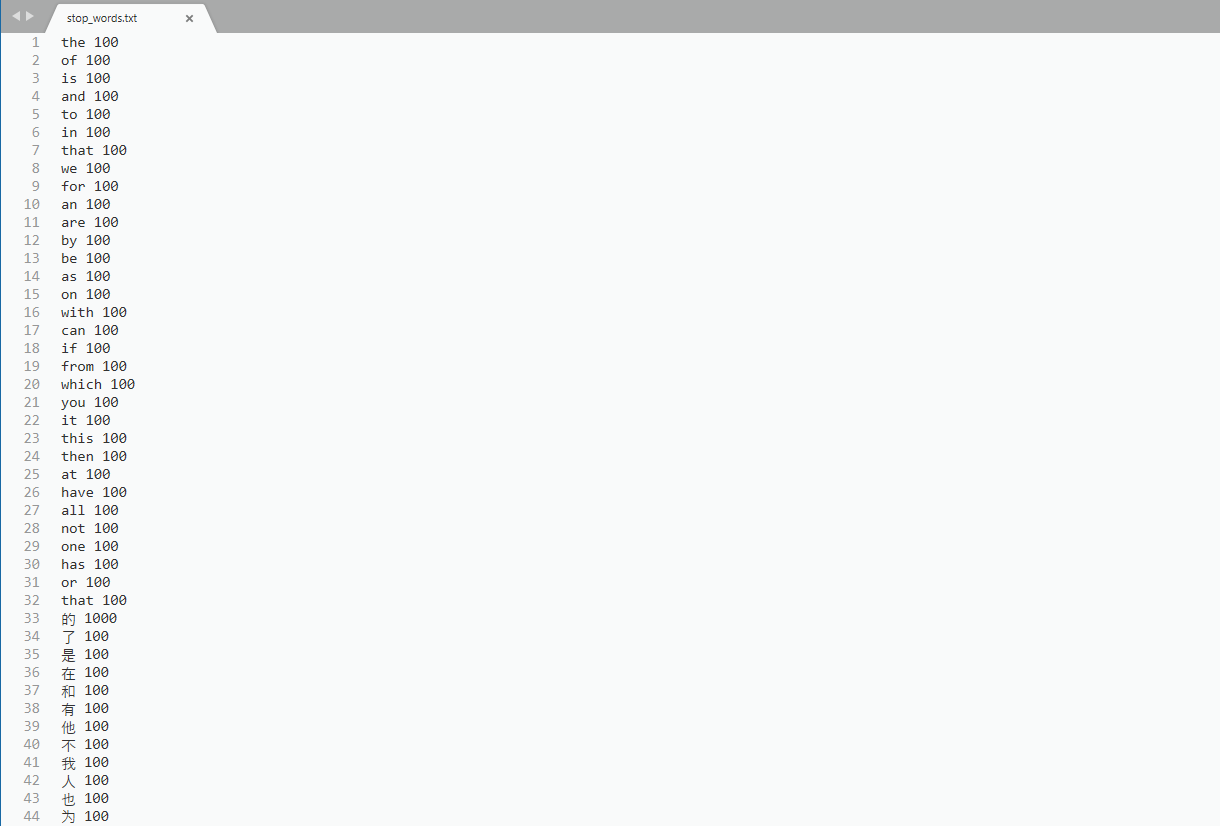
进行分词的时候，为了更好地对英文单词进行分词，用nltk开源库获取每个关键词的词干，然后用这些词干建立“单词-文档权值矩阵”并保存关键词在文档中出现的次数。

同时，为了以后实现命中词突出显示的功能，把每个关键词在文档中出现的位置存储到倒排索引。

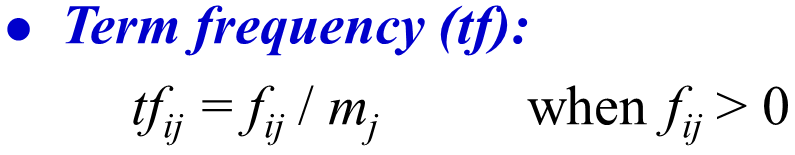
* ***筛选关键词***

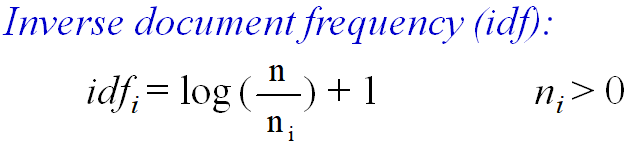
删除关键词中的不需要的字符

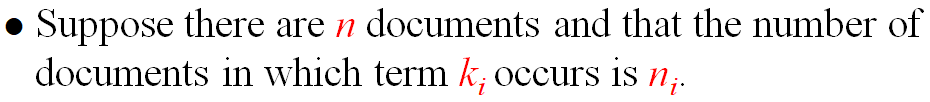
* ***停用词处理***

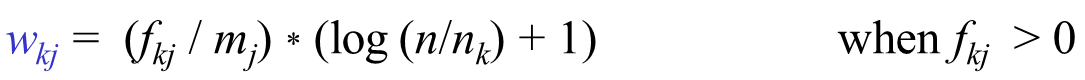
设置几十个常见的停用词，然后修改“单词-文档权值矩阵”中每个停用词出现次数。

* ***使用 Tf.idf 扩展布尔模型，计算 mj，tf，dj，nk，idf，Wkj***

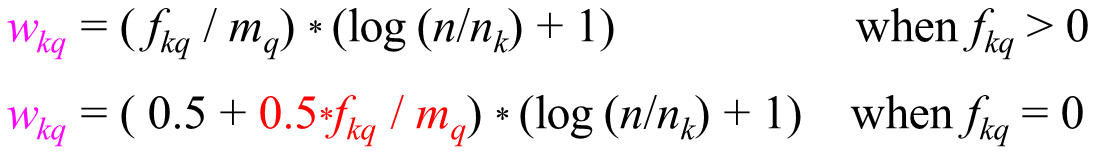
 



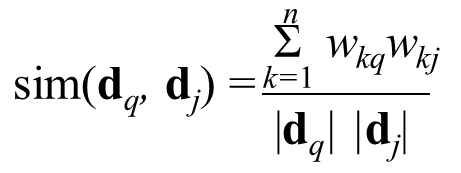




* ***检索，相关性计算***

收到查询语句之后对它进行分词，记录每个单词出现次数，计算mq，Wkq，dq。

然后计算查询语句和每个文档之间的相关度，

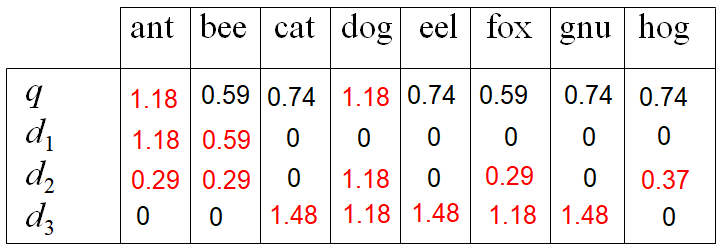


最后按相关度排序结果。

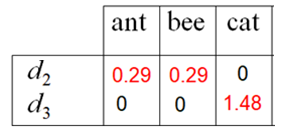
* **聚类**

获取所有搜索结果文档的关键词，从“Tf.idf 扩展布尔模型矩阵”获取各个文档和关键词相应的权值。用这些文档编号列表，关键词列表和权值来建立新的“Tf.idf 扩展布尔模型矩阵”。

Ex）假设下边图是原来的“Tf.idf 扩展布尔模型矩阵”



其中，如果跟搜索语句有相关的文档是 d2和d3，它们两个的关键词并集是“ant，bee，cat”，那么新的“Tf.idf 扩展布尔模型矩阵”是



用这些权值来进行动态聚类。

聚类部分用了动态聚类方法中的重心法。首先计算全部聚类文档权值的重心，建立新凝聚点的最小临界距离d，依次逐个输入全部文档权值，如果输入文档与已有凝聚点的距离大于d，则将它作为新的凝聚点；否则不作为凝聚点；

最后，把每个类的关键词权值分别加，然后用权值最高的关键词来代表每个类

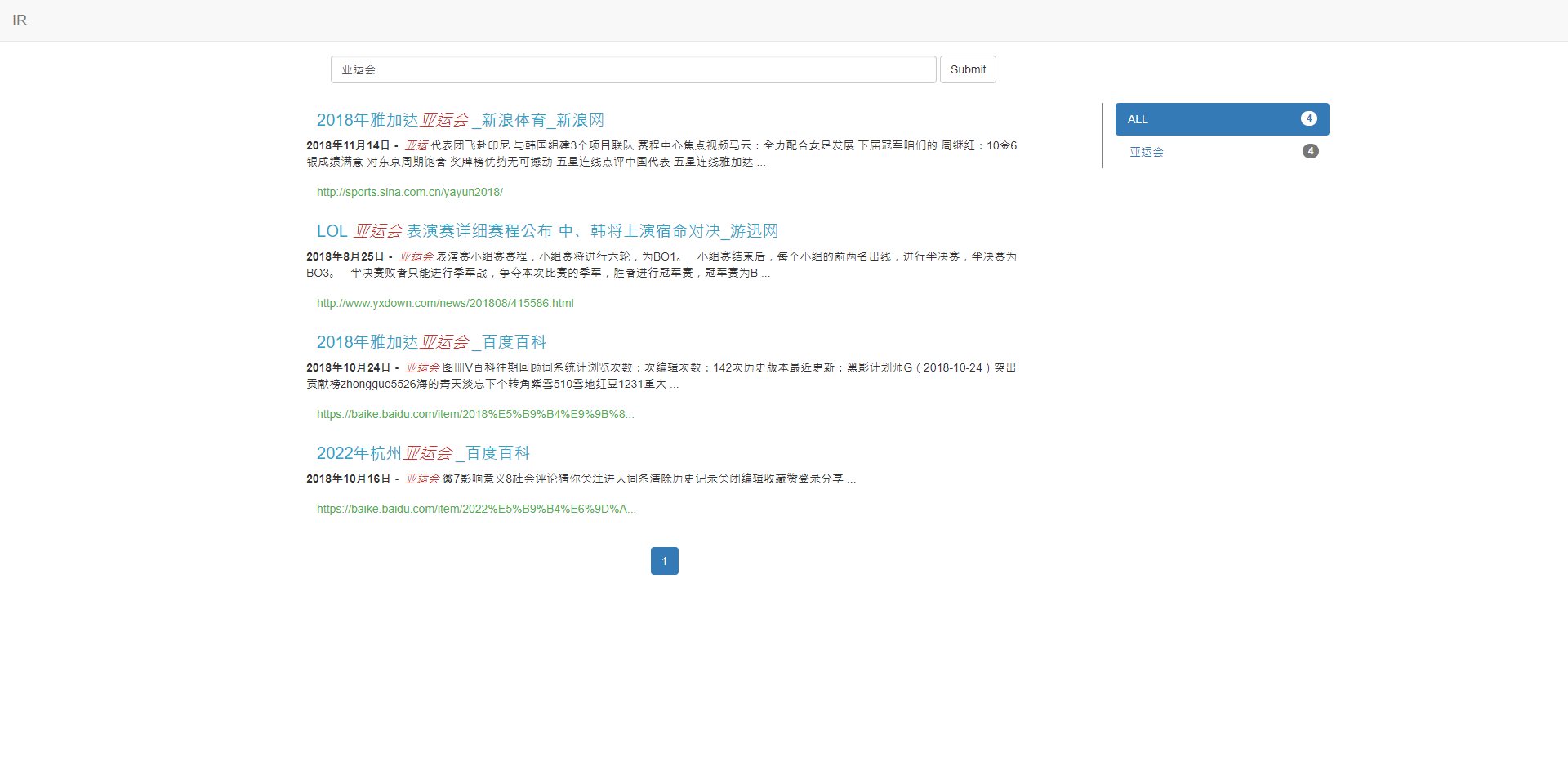
Ex) ant = 0.29 + 0 = 0.29

bee = 0.29 + 0 = 0.29

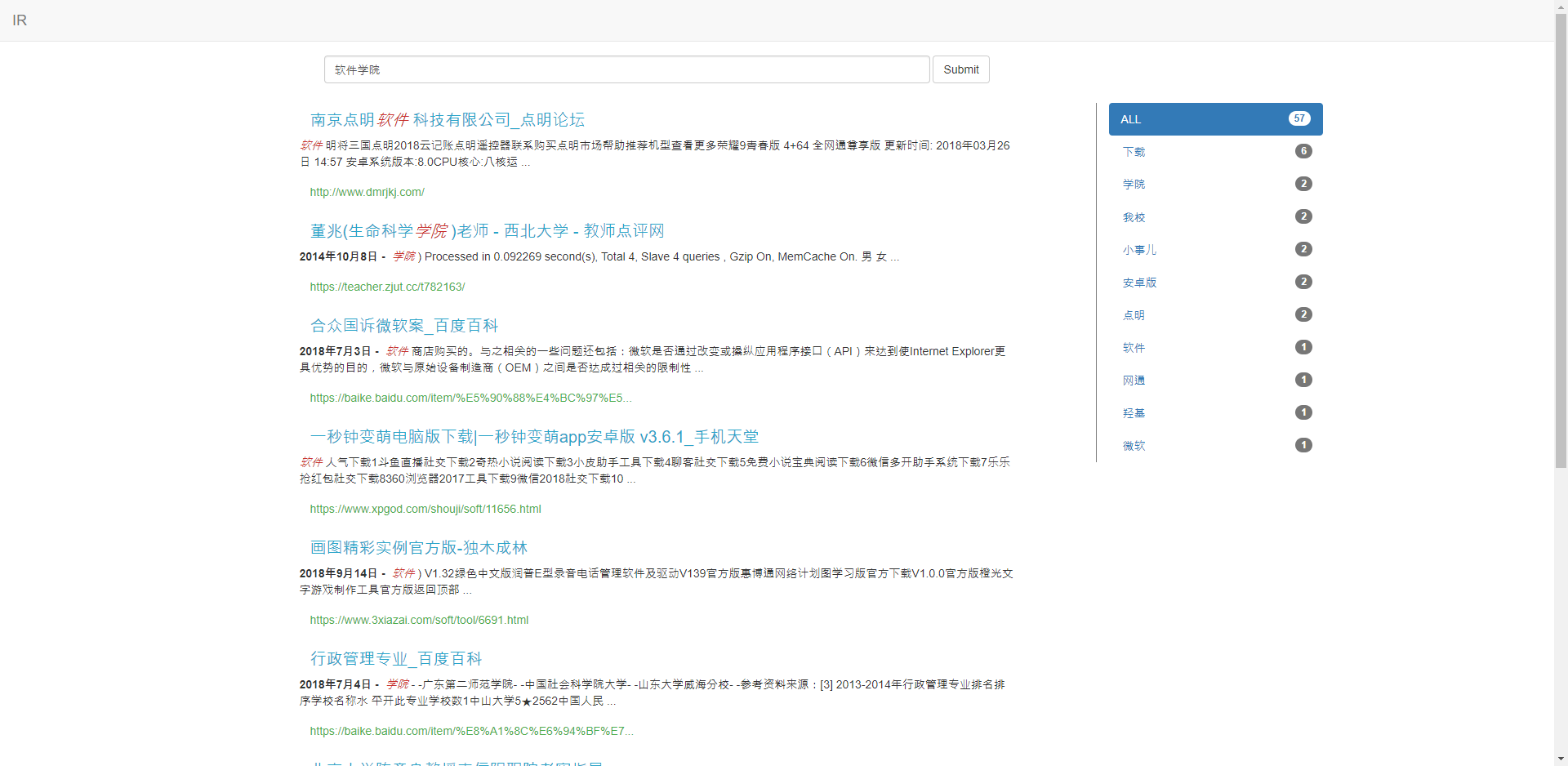
cat = 0 + 1.48 = 1.48

所以权值最高的关键词是“cat”，所以用它代表类名

**可视化**



《输入“亚运会”的搜索结果》



《输入“软件学院”的搜索结果》



《输入“清华大学”的搜索结果》



《输入“python”的搜索结果》

**使用说明**

（先安装好 nltk库，pymongo库，jieba库，requests库，bs4库，hashlib库，MongoDB， django ）

1. 执行src/preprocess目录下的pachong.py文件（爬虫）

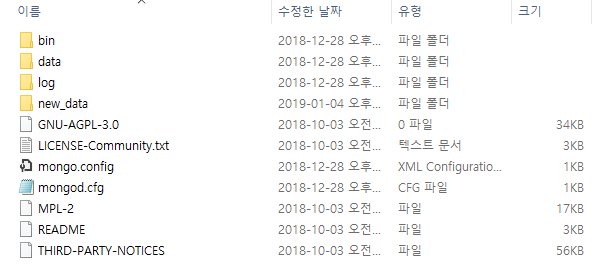
修改pachong.py文件的188行就能调整爬虫文档数。

（修改65）

（因为爬虫需要两三个小时，为了让助教老师更方便运行我们的程序，我把5222个文档打包之后放在src/preprocess/pachong目录下。）

1. 启动MongoDB

打开cmd，进入MongoDB目录，

创建新的文件夹（比如，new\_data）

进入MongoDB/bin目录

然后运行 mongod --dbpath (new\_data文件夹的路径)



就可以启动MongoDB。

1. 执行src/preprocess目录下的fenci.py文件

（注意！运行程序的时候必须从pachong文件夹删除打包文件，pachong.zip）

运行fenci.py文件做

“分词”,“ 提取关键词”,“ 获取词干”,

“ 建立单词-文档权值矩阵和倒排索引”,“筛选关键词”,“ 停用词处理”,

“ 使用 Tf.idf 扩展布尔模型，计算 mj，tf，dj，nk，idf，Wkj”

这些工作。

做完这些工作之后会把数据保存到MongoDB。

1. 打开django服务器

打开cmd，进入src目录，执行 python manage.py runserver 0:8000 --noreload

每次打开服务器的时候需要从数据库获取 tf.idf扩展布尔模型矩阵，所以可能需要2~3分钟的时间。

1. 进入网站

打开浏览器，访问 127.0.0.1:8000