# Lucene倒排索引

搜索引擎是什么？

搜索引擎（Search Engine）是指根据一定的策略、运用特定的计算机程序从互联网上搜集信息，在对信息进行组织和处理后（结构化处理，parser），为用户提供检索服务，将用户检索相关的信息展示给用户的系统。

搜索引擎包括全文索引（百度，谷歌）、目录索引、元搜索引擎、垂直搜索引擎（电商类的，专业领域类等）、集合式搜索引擎、门户搜索引擎与免费链接列表等

搜索引擎能解决什么问题？

高效:查询效率特别高。

适合大数据：千万级。

Solr es：elk，分布式存储。lucene

Lucene与搜索引擎的关系：

Lucene是检索系统的核心框架，它类似spring Jar包。只要把Jar引入自己的工程就可以用，不依赖es 和 solr，单机版 -> 集群版。

特别适用于小规模的检索，比如10万左右的数据，用mysql达不到效果，用es和solr等成本太高，就可以直接用Lucene。直接嵌入到你的工程，

不需要部署任何服务。

Mysql like ‘%四川省%’

Solr: 他们是基于Lucene封装的一个成熟的产品，提供很多的api尤其是提供了分布式索引，大大的提高了存储能力和搜索效率。适用于千万级以上分单机版和集群版：最大的区别

zookeeper:协调 管理服务 分布式锁 节点 统一动态配置的功能。分布式

Es:跟solr一样他也是基于Lucene开发的一款分布式搜索引擎。分单机版和集群版 这个集群版不需要zk;

倒排索引原理：

倒排索引的创建过程 倒排索引的创建其核心就会用到分词。比如下面这篇文章我们要建立倒排索引，其具体是怎么操作的呢

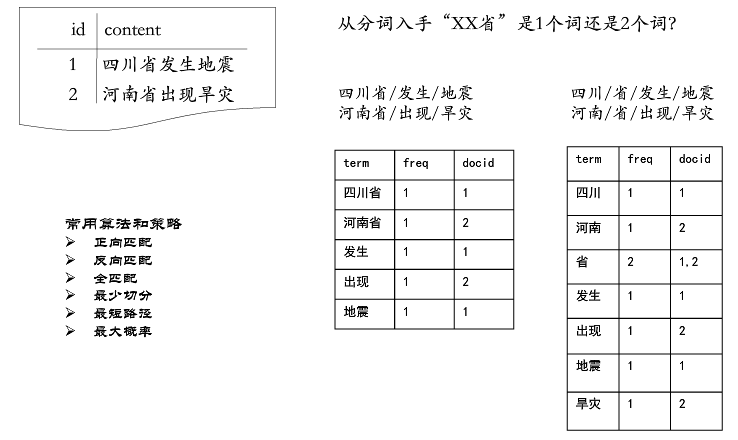
Doc(id = 1)： 四川省发生地震

第一步：分词

第二步：将分词后的词建立一个映射关系

即 词->docId，如右图所示

这种结构为什么叫倒排索引呢？



TF-IDF：大家看百度的结果 他是根据什么进行排序的呢？当然你要忽略一些广告，这些就是order by money。如果我们看谷歌 那就更好了，这里同样的文章凭什么你就能排在我前面呢？

“数据结构”这个词：

“数据结构”在所有的文档中出现了多少次。如果有10篇文章，都出现了”数据结构”。就表示没有什么区分度

TF：词频 一篇doc中包含了多少这个词，包含越多表明越相关。只计算一篇文档的数量。

DF：文档频率 包含这个词的文档总数，DF在一篇文档算一次。

IDF：DF取反 也就是 1/DF;如果包含该词的文档越少，也就是DF越小，IDF越大，则说明词对这篇文档重要性就越大。

TFIDF: TF\*IDF 的主要思想是：如果某个词或短语在一篇文章中出现的频率TF高，并且在其他文章中很少出现，则认为此词或者短语具有很好的类别区分能力，这篇文章的得分也就越高。

有什么问题？长度的有大有小，一篇文章就2个词，搜其中一个讲道理是50%的权重，有一篇文章有100个词，48词。

归一化处理：主要是对TF-IDF做处理，会根据文章的长度。

打分的定制加成：