快速构建 PHP 全文检索

xapian/xunsearch 实战分享

By: hightman/2013.6

认识全文检索

- 搜索 — 通往信息海洋之门
- 在“海量”信息中快速、准确地根据关键词/句返回所需的信息
- 专业称谓：信息检索 (Information Retrieval)
- 全文检索 ≠ LIKE ‘%keywords%’

检索比较

	全文检索	数据库 LIKE
索引	使用事先建好的全文索引	用不到索引，只能遍历匹配
匹配效果	通过分词器切割匹配，良好支持中文、英文词干	%eight% 也会匹配 height "%com%net%"：就不能匹配颠倒的xxx.net..xxx.com
相关度	基于概率模型的相关性算法，越相关的排在越前面	无相关算法，匹配一次或多次无明显区别
可定制	通过定制分词器，实现不同索引规则	难以定制
结论	支持大数据，性能高效果好	效率低，相关性差，模糊检索效果差，适合小规模

IR基础：术语

- 反向索引概念 (类似字典检字表)
- Document, Term, Posting

  要检索的对象称之为 document，通常可以认为是一块文本或一条数据库记录；而 term 则是一个词或短词用于描述 document，每一个 document 包含若干个 term；posting 就是包含了 term 在 document 所处位置的 term，用于相关度检索。

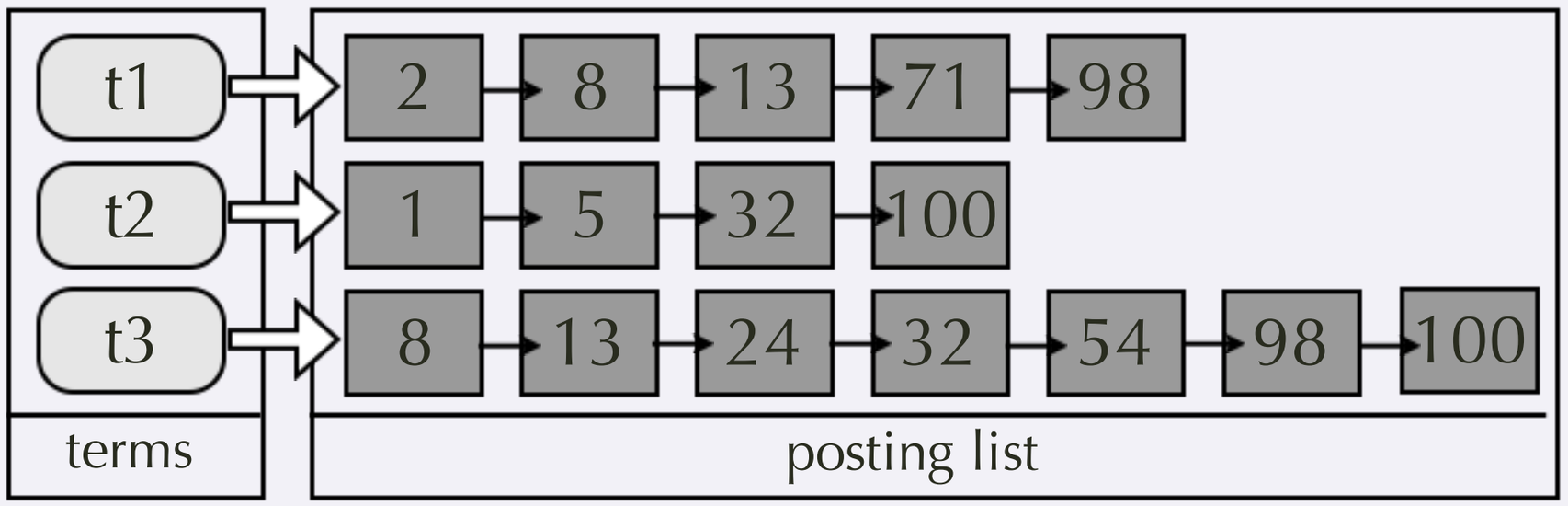
  在全文检索中，如果名为 D 的 document 被一个名为 t 的 term 所描述，那么称之为 t 索引了 D。因此，term 和 document 是一个多对多的关系。每一个 term 存储着被它索引的 documents 列表，称之为 posting list。

  全文索引要就是存储 term 与 document 的关系，并有序的组织着 terms。term 并不一定要求必须在 document 中出现，通常也会做词干修剪处理。

- Tokenizer (分词器：Document/Query -> term)

IR基础：索引组织

- 以索引 100 个 documents 为例：



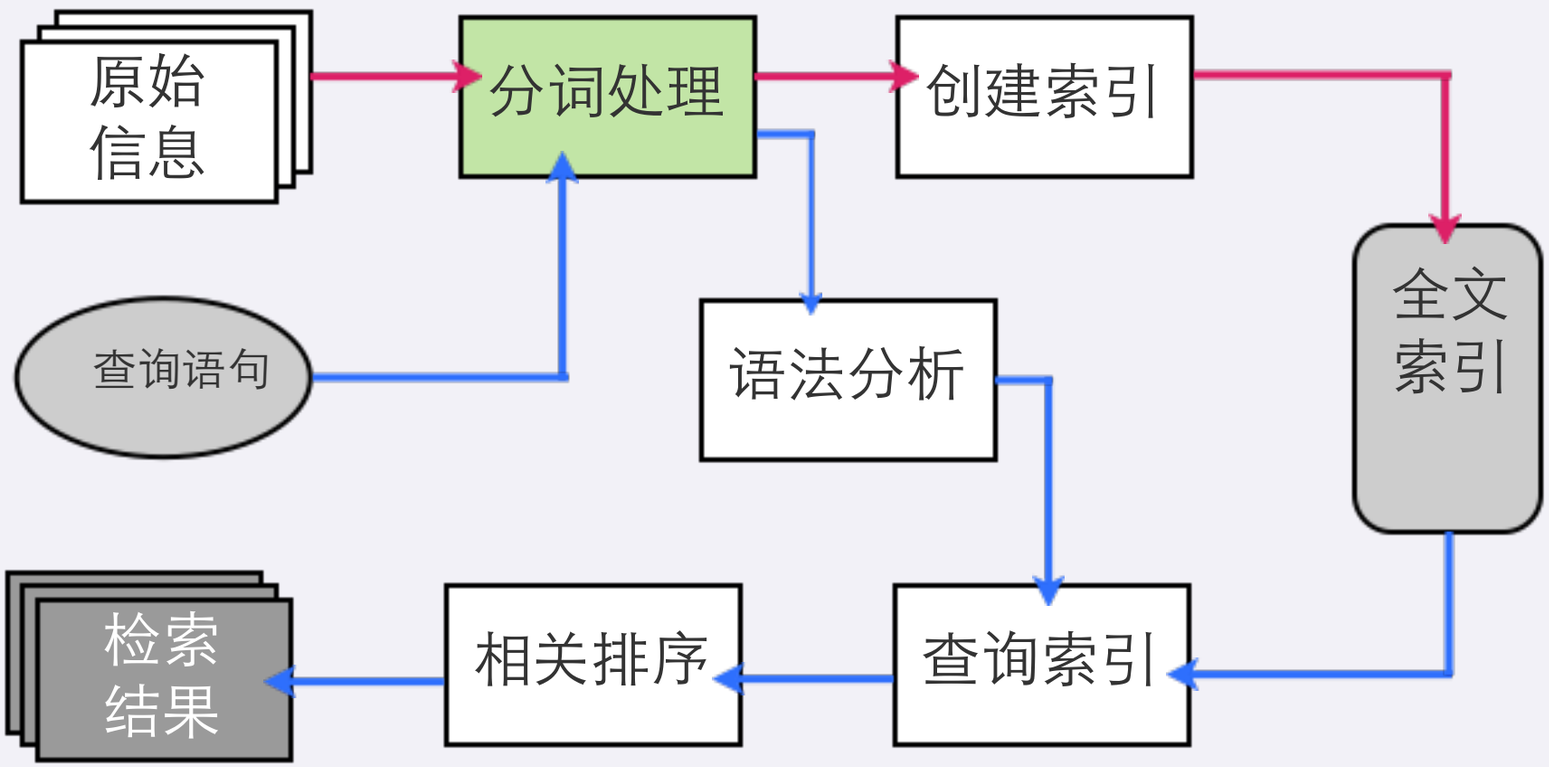
- 能否检索到某一个 Document 取决于是否建立了相应的 Term 索引

- 中文分词的重要性

IR基础：相关性

- 概率模型的核心概 (Relevance)
- wdp：Term 在 document 中的出现位置
- wdf：Term 在 document 中的出现次数
- ndl：当前文档长度/平均长度
- wqf, wqp, nql ……
- BM25 (Best Match，used by xapian, sphinx ...)

全文检索过程



开源方案

- Lucene：Java 界最有名的检索程序库，相关应用方案 nutch, solr ...

- Sphinx： C++ 编写的依赖于 SQL 的搜索服务器

- **Xapian**：发音 /zap-ian/

  - 近 30 年悠久历史，类似 Lucene 纯工具库
  - C++ 编写，跨平台支持，支持大量脚本语言绑定
  - Unicode 支持，索引数据统一采用 UTF-8 存储
  - 概率搜索排序，默认采用 BM25 算法，越相关的结果排在越前面
  - 全方位的布尔查询解析器；词干修剪 (支持英语等数 10 种语言)
  - 支持实时搜索，同义词、拼写纠错、精确搜索等
  - 单库支持最高 40 亿条数据；单写多读，原子性修改
  - 大量采用 B-tree 存储，索引写入速度相对较慢，约 500 条/s
  - 搜索性能佳，官方宣称 1.5TB/5亿网页 < 1 秒，实测百万级均 0.0x 秒

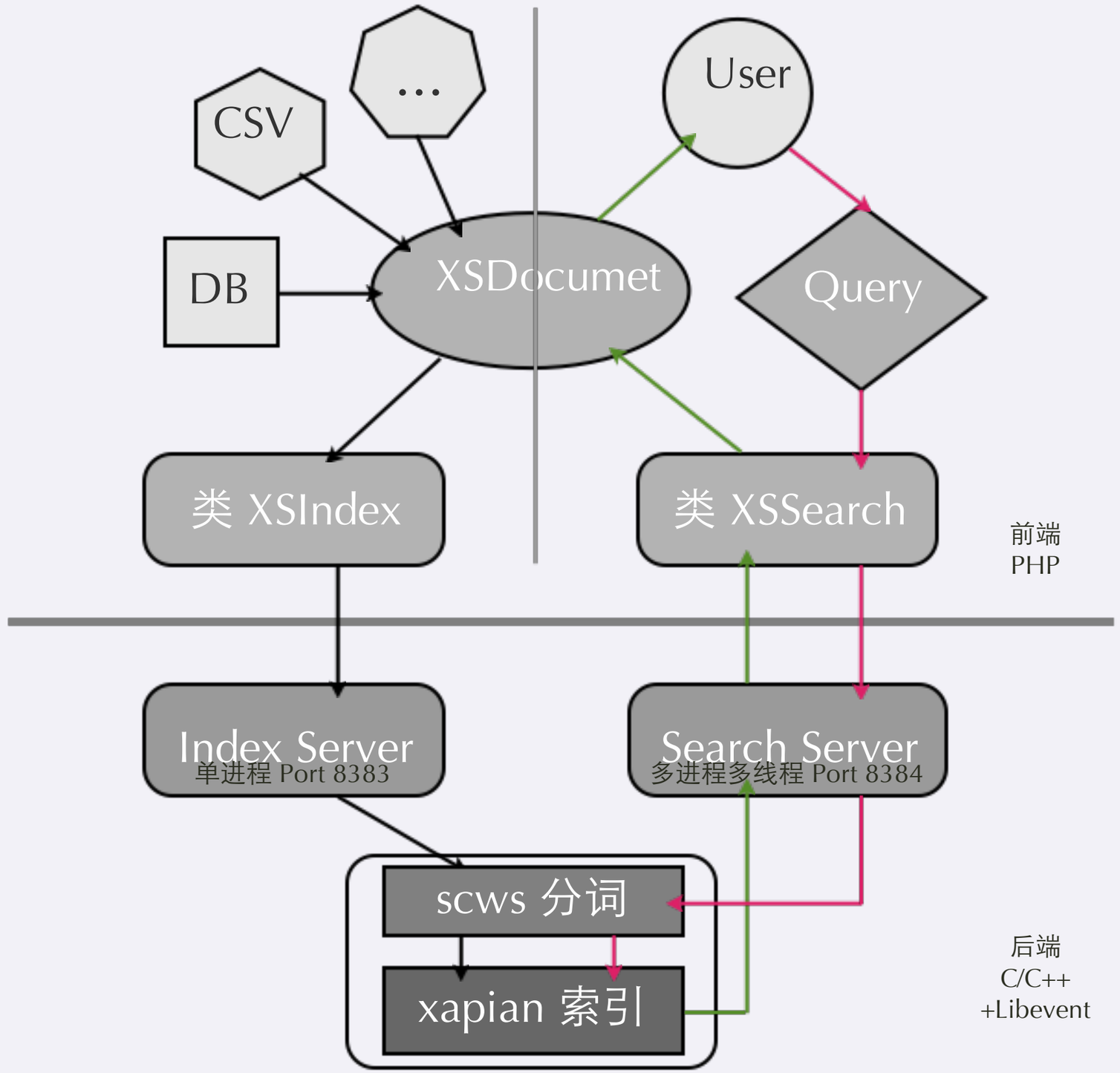
Xapian 实战缺陷

- 并不是完整的应用程序
- 缺少字段概念，缺少中文分词支持
- 英文资料本身就不多，更缺少中文资料
- 缺少统一服务端来管理单写多读机制
- API 接口繁多、复杂，使用门槛较高

xunsearch 诞生

- 整合 xapian 和 scws 中文分词，优化中文处理
- API 简单清晰，附带中
- 支持 255 个字段，高亮
- GPL 协议，2011.9 首次
  前稳定版本 1.4.6，已被
- scws 同样开源，支持复

XS 架构


CSV
...
User
DB
XSDocumet
Query
类 XSIndex
类 XSSearch
前端
PHP
Index Server
单进程 Port 8383
Search Server
多进程多线程 Port 8384
scws 分词
xapian 索引
后端
C/C++
+Libevent

XS 安装

- 下载&编译

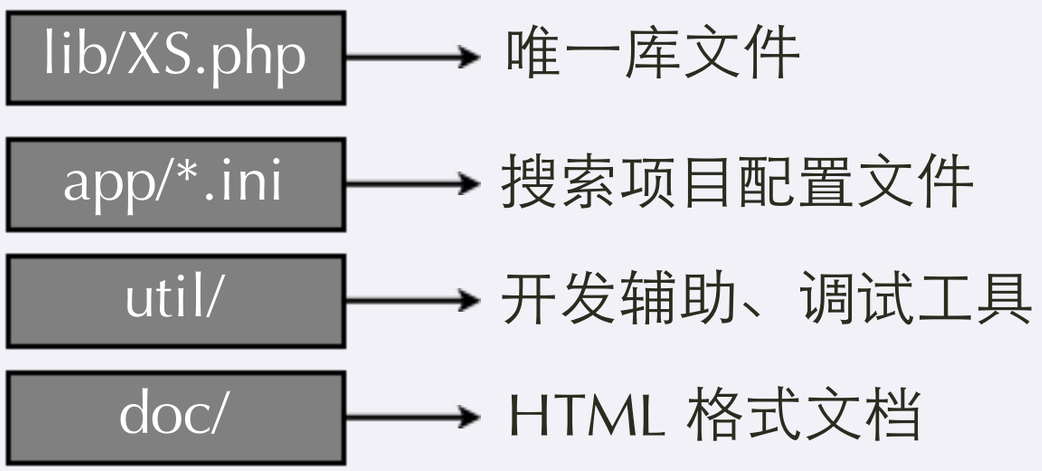
```
wget http://www.xunsearch.com/download/xunsearch-full-latest.tar.bz2
tar -xjf xunsearch-full-latest.tar.bz2
cd xunsearch-full-1.4.6
sh setup.sh
```

- 启动服务

```
$prefix/bin/xs-ctl.sh start
$prefix/bin/xs-ctl.sh -b inet start
$prefix/bin/xs-ctl.sh restart
```

通讯无加密和验证，需配合 iptables 等工具辅助设置

- SDK 组成：$prefix/sdk/php



XS 初识

- 多项目支持，独立INI
- 三步开发流程
- 了解 XS 对象
  - XS
  - XSDocument
  - XSException
  - XSIndex
  - XSSearch

```
require '$prefix/sdk/php/lib/XS.php';
try {
    $xs = new XS('demo');
    $index = $xs->index;
    $search = $xs->search;
    // ... other codes ...
} catch (XSException $e) {
    echo $e;
}
```

项目INI文件

- 以段落描述字段
- 字段类型 (type)
- 索引方式 (index)
- 分词器 (tokenizer)
  - 类型误区、修改注意

```
; sdk/php/app/demo.ini
project.name = demo

[pid]
type = id

[subject]
type = title

[message]
type = body

[cid]
index = self
tokenizer = full

[chrono]
type = numeric
```

创建索引

- 创建 XSDocument
- 添加/更新数据 (区别)
- 删除数据
- 批量处理 openBuffer
- 同步机制 flushIndex

```
$doc = new XSDocument();
$doc->pid = 1;
$doc['message'] = 'hello content';
$doc->setFields(array(
    'subject' => 'hello title',
    'cid' => 10,
    'chrono' => time(),
));
$index->add($doc);
// $index->update($doc);

$index->del('1');
$index->del(array('3', '51'));

$index->openBuffer();
// ... more operators here ...
$index->closeBuffer();

$index->flushIndex();
```

搜索

I. 构建 Query

```
$query = '杭州西湖';
$query = 'subject:杭州 西湖';
$query = '杭州 OR 西湖'; // 缺省AND
$query = '西湖 -杭州'; // - 紧贴关键词
$query = '(杭州 OR 西湖) NOT 广场';
```

II. 设置搜索参数

```
$search->setFuzzy(false)
        ->setAutoSynonyms(true)
        ->setLimit(20, 80) //limit, offset
        ->setSort('chrono')
        ->setQuery($query);

$search->setFuzzy(true)
        ->setCutOff(70); // cut percent
```

III. 获取结果/匹配数

```
// 获取结果文档
$docs = $search->search();
foreach ($docs as $doc) {
  $subject = $search->highlight($doc->subject); // 高亮标题
  echo $doc->rank() . '. ' . $subject . ' [' . $doc->percent() . '%] - ' . date('Y-m-d', $doc->chrono) . "\n";
  echo $doc->message . "\n\n";
}

// 获取匹配数量估算值
$count = $search->getLastCount();
$count = $search->count($query);
$total = $search->getDbTotal();
```

特色搜索

- 基于搜索日志自动分析 (Seach Log)
- 热门：XSSearch::getHotQuery([int $limit = 10])
- 相关：XSSearch::getRelatedQuery()
- 纠错：XSSearch::getCorrectedQuery()
- 展开：XSSearch::getExpandedQuery(string $prefix)

命令行工具

- util.Indexer
- util.Quest
- util.SearchSkel
  - 作为学习范例

```
# 各指令均可通过 --help 查看详细帮助
php util/Indexer.php demo --clean  # 清空索引
localhost:php hightman$ php util/Indexer.php demo --source=mysql://ro
lhost/hightman_cn --sql="SELECT * FROM mybb_posts"
初始化数据源 ... mysql://root@localhost/hightman_cn
开始批量导入数据（请直接输入数据）...
完成索引导入：成功 2506 条，失败 0 条
刷新索引提交 ...
localhost:php hightman$ php util/Quest.php demo hightman --limit=2
在 2,508 条数据中，大约有 274 条包含 hightman，第 1-2 条，用时：0.08

1.  #594# [100%,3.88]
...12:25

[img]http://www.hightman.cn/bbs/images/attachicons/image.gif[/img] 过
http://www.hi...
Chrono:

2. 回复 #1 hightman 的帖子 #1529# [98%,3.84]
...地址：[url=http://www.hightman.cn/down/scws-1.0.0.tar.gz]http://w
tman.cn/down/scws-1.0.0.t...
Chrono:

相关搜索：enable-hightman-mbft hightman mbft hightman twomice
```

技巧：树型分类

- 案例分析

- 网店商品表使用字段 cid 表示分类，分类数据表使用字段 parent_id 表示所属父类
- 问题/需求：根据分类id 检出商品以及子孙分类商品

- 递归求出所有子孙分类ID：xxx, xxx2, xxx3, ...
- SQL 数据库检索：WHERE cid IN (xxx, xxx2, xxx3 ...)
- 未处理的 XS 检索： cid:xxx OR cid:xxx2 OR cid:xxx3 ...

- XSDocument::addTerm(string $field, string $term)

Tips: 可以在创建索引时，建立所有祖先分类的索引关联

```
$doc = new XSDocument($fields);
for ($parent_id = $doc->cid; $parent_id = get_parent_id($parent_id); ) {
    $doc->addTerm('cid', $parent_id);
}
$index->update($doc);
```

技巧：同义词

- 案例分析

- 为什么文章中明明包含“刘诗诗”，可搜索“刘”却什么也检索不到?
- 希望在网店搜索“鞋”时，可以检索到 “鞋子”，“皮鞋” 等商品?
- 怎么能让用户搜索 “范爷” 时也能搜索到 “范冰冰” 的信息?

根源都在于分词，只有建立了相应的索引才能检索到!

- 最简单解决方案：补充相关数据进行索引
- 同义词用法
- 自定义分词

```
// 添加同义词
$index->addSynonym(‘鞋’, ‘鞋子’);
$index->addSynonym(‘鞋’, ‘皮鞋’);
$index->addSynonym(‘范爷’, ‘范冰冰’);
$index->addSynonym(‘范冰冰’, ‘范爷’);
// 获取同义词列表
$search->getAllSynonyms([int $limit [, int $offset = 0]]);
```

技巧：自定义分词

- 实现接口或继承现有分词器

  $prefix/sdk/php/lib/XSTokenizerXyz.class.php

```
// class XSTokenizerXyz extends XSTokeinizerScws
class XSTokenizerXyz implements XSTokenizer
{
    public function getTokens($value, XSDocument $doc = null) {
        return empty($value) ? array() : explode('-', $value);
    }
}
```

- 在 INI 文件中指定自定义分词器

```
[some_field]
index = self
tokenizer = xyz
```

联系我们

- 源码托管：https://github.com/hightman/xunsearch
- 项目网站：http://www.xunsearch.com
- 社区支持：http://bbs.xunsearch.com
- QQ群支持：14413875
- 高级定制：亿级数据、分布式架构

Thanks