

校园搜索引擎

计 41 张盛豪 2014011450

计 43 李明杰 2014011351

2017-06-03

目录

1 内容	1
2 开源搜索引擎工具资源:	2
3 实验要求	2
4 实现流程	2
4.1 爬虫爬取清华校内网页数据	2
4.1.1 Heritrix 抓取对象	2
4.1.2 Heritrix 抓取过程中遇到的问题	3
4.2 数据清洗及 PageRank 计算	5
4.3 构建索引及倒排索引	8
4.3.1 文档解析	8
4.3.2 Html 解析	8
4.3.3 PDF 解析	8
4.3.4 Doc 解析	8
4.4 检索	8
4.4.1 修改图片搜索框架	8
4.4.2 使用 MultiFieldQueryParser	8
4.4.3 分域权重	9
4.4.4 文档摘要	9
5 实验结果	11
6 心得体会	13

1 内容

综合运用搜索引擎体系结构和核心算法方面的知识，基于开源资源搭建搜索引擎

2 开源搜索引擎工具资源：

- Heritrix 1.14.4
- Lucene 4.0
- 分词工具：IK Analyzer 2012
- Html 解析：Jsoup 1.7.2
- PDF 解析：pdfbox 1.8.1
- Doc 解析：poi 3.16
- 前端服务：apache+tomcat (<http://tomcat.apache.org/>)

3 实验要求

- 抓取清华校内绝大部分网页资源以及大部分在线万维网文本资源（含 M.S.office 文档、pdf 文档等，约 20-30 万个文件）
- 实现基于概率模型的内容排序算法；
- 实现基于 HTML 结构的分域权重计算，并应用到搜索结果排序中；
- 实现基于 PageRank 的链接结构分析功能，并应用到搜索结果排序中；
- 采用便于用户信息交互的 Web 界面。

4 实现流程

4.1 爬虫爬取清华校内网页数据

4.1.1 Heritrix 抓取对象

Heritrix 环境搭建教程

- 清华新闻网网页（不包括图书馆）资源
- 种子 <http://news.tsinghua.edu.cn/>
- 使用正则表达式对 URL 进行过滤
 - 过滤无关页面，过滤无关格式文件，，保留 html 页面和 pdf,word 文档，去除奇怪链接：

```
.*(?:i)\.(ms|tar|txt|asx|asf|bz2|lpe?g|MPE?G|tiff?|gif|GIF|png|PNG|ico|ICO|css|sitleps|wmf|zip|pptx?|xlsx?|gz|rpm|tgz|mov|MOV|exe
```
 - 禁止抓取图书馆资源：`[\\S]*lib.tsinghua.edu.cn[\\S]*`；`[\\S]*166.111.120.[\\S]*`
 - 只抓取清华新闻网的数据 `[\\S]*news.tsinghua.edu.cn[\\S]*`；
- Module 设置，参考 PPT 中的设置进行

4.1.2 Heritrix 抓取过程中遇到的问题

4.1.2.1 页面剔除问题

在一开始的抓取过程中发现了部分奇怪的页面，如图 [Heritrix 异常页面] 所示，发现了很多以yyyy.MM.dd、yyyy.M.d、MM.dd.yyyy、a.nivo-nextNav 为结尾的奇怪链接，后来经过分析清华新闻网的源码，发现这些链接都是 js 代码里的内容，虽然已经在配置中设定为不从 css,js 等域中获取超链接，但是仍然会得到这样的 url，因此后来在正则表达式中加上了dlddlyyylnivo-nextNavlxiao_ming字段进行过滤，最终得到的页面有 52769 个，共 2.3G

```
2017-05-24T11:57:19.800Z 404 405 http://news.tsinghua.edu.cn/publish/thunews/9673/2011/20110110112743271522690/yyyy.MM.dd.XLLL
2017-05-24T11:57:19.829Z 200 8951 http://news.tsinghua.edu.cn/publish/thunews/9673/2011/20110110113018439726389/201101101130184
2017-05-24T11:57:19.829Z 404 405 http://news.tsinghua.edu.cn/publish/thunews/9664/2014/20140912114543278616330/yyyy.MM.dd.XLLLLL
2017-05-24T11:57:19.829Z 404 405 http://news.tsinghua.edu.cn/publish/thunews/9664/2014/20140912114543278616330/a.xiao_ming.XLLLLL
2017-05-24T11:57:19.830Z 404 405 http://news.tsinghua.edu.cn/publish/thunews/9664/2014/20140912114543278616330/a.nivo-nextNav.XL
2017-05-24T11:57:19.846Z 404 405 http://news.tsinghua.edu.cn/publish/thunews/9664/2014/20140912114543278616330/MM.dd.yyyy.XLLLLL
2017-05-24T11:57:19.846Z 404 405 http://news.tsinghua.edu.cn/publish/thunews/9673/2011/20110110113018439726389/yyyy.MM.dd.XLLL
2017-05-24T11:57:19.847Z 404 405 http://news.tsinghua.edu.cn/publish/thunews/9673/2011/20110110113018439726389/MM.dd.yyyy.XLLL
2017-05-24T11:57:21.809Z 404 405 http://news.tsinghua.edu.cn/publish/thunews/9673/2011/20110110113018439726389/yyyy.M.d.XLLLLL
2017-05-24T11:57:21.812Z 404 405 http://news.tsinghua.edu.cn/publish/thunews/9673/2011/20110110104920184291986/vvvv.MM.dd.XLLL
```

图 1: Heritrix 异常页面

4.1.2.2 Heritrix 加速问题

在抓取页面过程中，第一次尝试时发现抓取速度特别慢，爬取整整一个晚上只能获取 300MB 的数据，通过查询相关资料尝试了很多加速方法，最终在博客Heritrix 提高抓取效率的若干尝试 {http://blog.csdn.net/yangding_/article/details/41122977} 找到 Heritrix 抓取速度特别慢的原因：heritrix 在抓取时一般只运行了一个线程。这是因为在默认的情况下，Heritrix 使用HostnameQueueAssignmentPolicy来产生 key 值，而这个策略是用 hostname 作为 key 值的，因此一个域名下的所有链接都会被放到同一个线程中去。如果对 Heritrix 分配 URI 时的策略进行改进，利用 ELFHash 算法把 url 尽量平均分部到各个队列中去，就能够用较多的线程同时抓取一个域名下的网页，速度将得到大大的提高。

@Override

//重写 getClassKey()方法

```
public String getClassKey(CrawlController controller, CandidateURI cauri) {
    String uri = cauri.getURI().toString();
    long hash = ELFHash(uri); //利用 ELFHash 算法为 uri 分配 Key 值
    String a = Long.toString(hash % 50); //取模 50，对应 50 个线程
    return a;
}


public long ELFHash(String str)
{
    long hash = 0;
    long x = 0;
    for(int i = 0; i < str.length(); i++) {
        hash = (hash << 4) + str.charAt(i); //将字符中的每个元素依次按前四位与上
        if((x = hash & 0xF0000000L) != 0) //个元素的低四位想与
```

```

    {
        hash ^= (x >> 24); //长整的高四位大于零，折回再与长整后四位异或
        hash &= ~x;
    }
}
return (hash & 0x7FFFFFFF);
}

```

按照该教程完成代码的修改之后，重新运行，抓取速度得到了大幅提升，4 个半小时的时间完成了清华新闻网的页面抓取工作。



Status as of **五月. 23, 2017 17:26:52 GMT**
Alerts: **6 (5 new)**

Admin Console
CRAWLING JOBS
RUNNING job: *News*

0 jobs pending, 1 completed
85 URIs in 4m13s (0.35/sec)

[Console](#)
[Jobs](#)
[Profiles](#)
[Logs](#)
[Reports](#)
[Setup](#)
[Help](#)

Crawler Status: **CRAWLING JOBS** | [Hold](#)

Jobs
Running: *News*
0 pending, 1 completed
Alerts: [6 \(5 new\)](#)

Memory
21713 KB used
65536 KB current heap
932352 KB max heap

Job Status: **RUNNING** | [Pause](#) | [Checkpoint](#) | [Terminate](#)

Rates
0.35 URIs/sec (0.34 avg)
5 KB/sec (5 avg)

Time
4m13s elapsed
20m2s remaining (estimated)

Totals

downloaded 85

17%

403 queued

488 total downloaded and queued
1.3 MB crawled (1.3 MB novel)

[Refresh](#)

[Shut down Heritrix software](#) | [Logout](#)

图 2：未修改 hash 函数时爬取速度

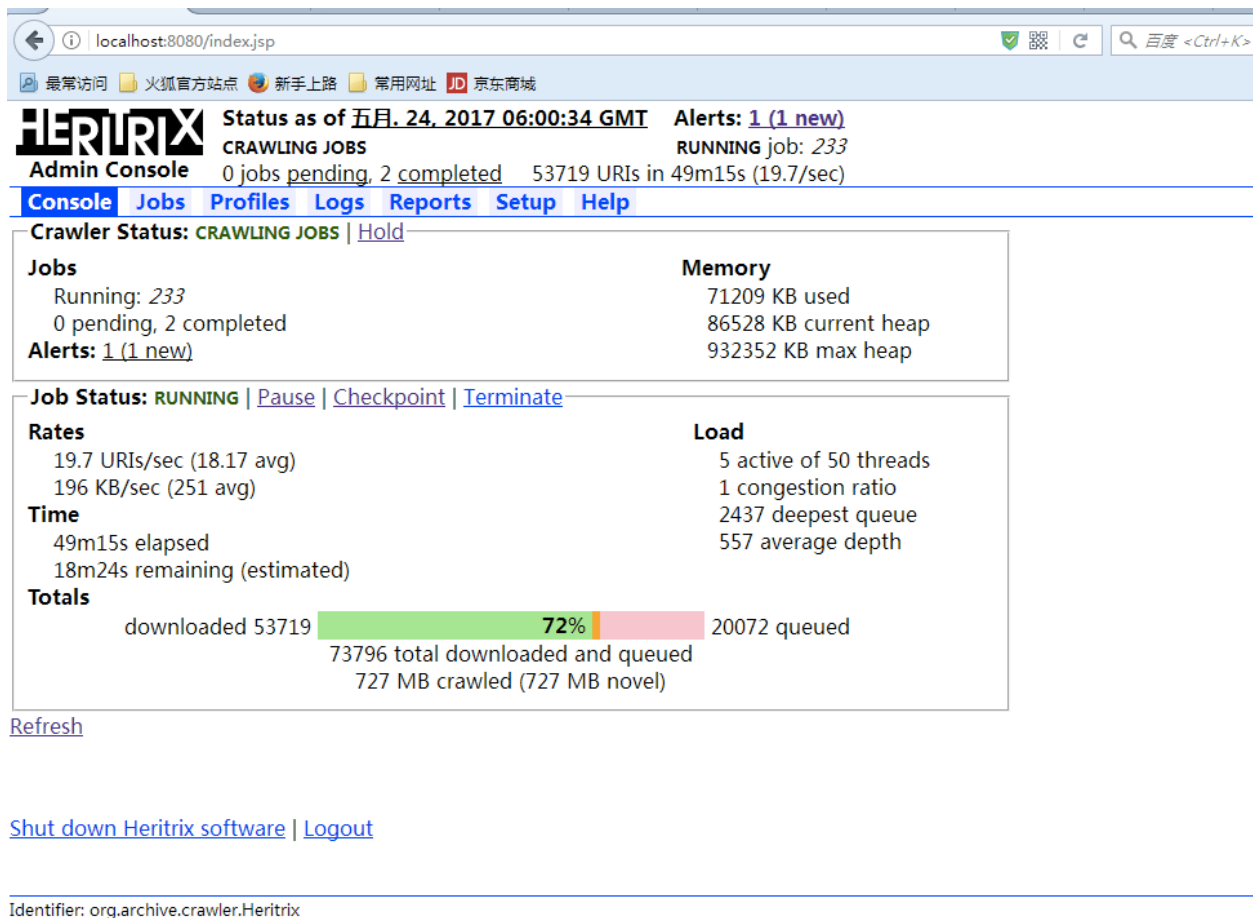


图 3：修改代码后的爬取速度明显加快

4.2 数据清洗及 PageRank 计算

在实验过程中我们需要使用页面的 PageRank 值来对网页进行打分，因此需要对抓取的数据计算其 PageRank 值，由于 Heritrix 在抓取时就已经有了 crawl.log 文件用于记录抓取到的网页链接及访问信息，因此在计算 PageRank 时直接使用了该文件的记录信息。

预处理及 PageRank 计算

- └─ clean_page.py : 页面清洗,
- └─ parse_graph_title_anchor.py :
- └─ share.py : 一些共有的文件名记录, html 页面处理函数
- └─ tsinghua_rank.py : 计算 PageRank 值, 结果保存到 pagerank.txt 中
- └─ crawl.log : Heritrix 抓取的结果记录

需要注意的是，由于 Heritrix 在抓取带 GET 请求的网页时，存储文件的文件名和网址 URL 并不能一一对应（其去掉了问号，挪动了文件类型的位置），且单从文件名并不能找到对

应的 URL，所以第一步分析 Heritrix 爬取日志是必要且是必须的。通过分析日志，得到了 URL 到文件名的双向映射，同时删除了 404 网页，将网页个数减少到 52205 个页面。

在提取链接、标题、anchor 时一开始使用的是 BeautifulSoup 进行提取，后来发现这种方式太慢，于是手动使用正则表达式进行提取。

```
href_pattern = re.compile(r'<a  
    href=[\"'\"]([^\\"'\"]*?\.(html|pdf|doc|docx))([\"'\"].*?>(.+?)</a>', re.S)  
html_pattern = re.compile(r'<a href=[\"'\"]([^\\"'\"]*?\.(html))([\"'\"]', re.S)  
title_pattern = re.compile(r'<title>(.*?)</title>', re.I | re.M | re.S)
```

计算完成之后的 PageRank 值如下所示：

1. 排名前 10 的页面

/publish/thunews/index.html	0.0563122679453	首页	清华大学新闻网
/publish/thunews/9652/index.html	0.0448806058384	更多 ›	清华大学新闻网 - 图说清华
/publish/thunews/en/index.html	0.0384040176156	ENGLISH Tsinghua University News	
/publish/thunews/9650/index.html	0.0257207878312	媒体清华	清华大学新闻网 - 媒体清华
/publish/thunews/10303/index.html	0.0257202750476	综合新闻	清华大学新闻网 - 综合新闻
/publish/thunews/9656/index.html	0.0250196913473	清华人物	清华大学新闻网 - 清华人物
/publish/thunews/9649/index.html	0.02500709101	要闻聚焦	清华大学新闻网 - 要闻聚焦
/publish/thunews/9657/index.html	0.0249803912605	新闻合集	清华大学新闻网 - 新闻合集
/publish/thunews/10304/index.html	0.0249798815124	新闻排行	清华大学新闻网 - 新闻排行
/publish/thunews/9655/index.html	0.0243158953181	专题新闻	清华大学新闻网 - 专题新闻

2. 新闻页前 10

/publish/thunews/9648/2017/20170520203232435687344/20170520203232435687344_.html	0.00197507591613	邱勇出席第二届中以创新论坛：畅谈国际创新创业教育合作
/publish/thunews/9648/2017/20170518115011788320647/20170518115011788320647_.html	0.00197507591613	清华医学院程功研究组揭示寨卡病毒感染暴发机制
/publish/thunews/9648/2017/20170519190126950804131/20170519190126950804131_.html	0.00197507591613	邱勇会见以色列总统鲁文·里夫林·接受以色列特拉维夫大学荣誉博士学位
/publish/thunews/9648/2017/20170522184445768862282/20170522184445768862282_.html	0.00197507591613	清华大学新闻与传播学院举办纪念成立15周年系列活动

清华大学新闻与传播学院举办纪念成立15周年系列活动
 /publish/thunews/9648/2017/20170515184412579525281/20170515184412579525281_.html
 0.00197507591613 清华大学全球可持续发展研究院正式揭牌成立
 清华大学全球可持续发展研究院正式揭牌成立
 /publish/thunews/9648/2017/20170516121327519550489/20170516121327519550489_.html
 0.00197507591613
 清华微电子所钱鹤、吴华强课题组在基于新型忆阻器阵列的类脑计算取得重大突破
 清华微电子所钱鹤、吴华强课题组在基于新型忆阻器阵列的类脑计算取得重大突破
 /publish/thunews/9652/2017/20170307133841881904789/20170307133841881904789_.html
 0.00193350555685 【组图】最美三月女生节 浪漫创意盈满“幅”
 【组图】最美三月女生节 浪漫创意盈满“幅”
 /publish/thunews/9652/2017/20170314142112142471080/20170314142112142471080_.html
 0.00193350555685 【组图】春到绿茵场 马杯足球赛正酣 【组图】春到绿茵场
 马杯足球赛正酣
 /publish/thunews/9945/2017/20170524164751321376872/20170524164751321376872_.html
 0.00143223231223
 5月18日，以“一带一路低碳前行”为主题的第十二届世界低碳城市联盟大会暨低碳城市发展论坛在三
 ... 2017-05-24 清华共同主办第十二届世界低碳城市联盟大会
 /publish/thunews/9945/2017/20170524113049223303463/20170524113049223303463_.html
 0.00143197048315
 2017年“共和国的脊梁——科学大师名校宣传工程”汇演在重庆大学启动。清华大学原创话剧《马兰花
 ... 2017-05-24 清华原创话剧《马兰花开》在科学大师名校宣传工程汇演上首演
 /publish/thunews/9945/2017/20170522171134178522272/20170522171134178522272_.html
 0.00143197048315
 5月19日晚，清华大学巅峰对话第二十期物理分论坛在清华大学举行。本次活动邀请了2015年诺贝尔物
 ... 2017-05-23 诺贝尔物理学奖得主梶田隆章做客“巅峰对话”
 /publish/thunews/9945/2017/20170524093408535456498/20170524093408535456498_.html
 0.00143197048315
 5月18日—21日，清华大学第一附属医院党委书记类延旭和副院长朱栓立带领一附院医疗分队走进昆明
 ... 2017-05-24 清华大学第一附属医院走进昆明健康义诊

由 PageRank 计算结果可以发现，正常的新闻页面的 PageRank 值大多在 10^{-6} 10^{-4} 之间，如果直接将该 PageRank 值与 BM25 算法的得分相乘会导致 PageRank 值高的页面，即使关键词出现次数少，也会在最终排名中特别靠前，为了减少其影响，在将 PageRank 值应用到 Score 计算时对 PageRank 值进行压缩

$$newPageRank = 16 + \ln(PageRank)$$

4.3 构建索引及倒排索引

4.3.1 文档解析

4.3.2 Html 解析

网页文件元素十分丰富。实验中使用 Jsoup 工具包解析网页，抽取 title 标签的文本内容作为文档的标题域；抽取 p、span、td、div、li、a 标签的文本内容作为文档的内容域；a 标签的内容表示页面链出的内容，也作为一个域单独索引；h1-h6 标签的文本内容表示页面内的小标题，拿出来作为一个域；此外，进入页面的链接有着和页面标题相似的作用，单独成为一个域。

4.3.3 PDF 解析

PDF 的元素不易区分，实验中使用 pdfbox 解析文件获得内容域，直接以文件名作为标题域。

4.3.4 Doc 解析

Doc 文件与 PDF 文件类似，实验中使用 POI 包解析文件获得内容域，直接以文件名作为标题域。需要注意的是，POI 工具解析 .doc 文件 .docx 文件的方法并不一样，在实验中，我们为此耽误了不少时间。

4.4 检索

4.4.1 修改图片搜索框架

在实验开始，我们修改了图片搜索的框架，进行如下操作。

1. 对查询进行分词后获得 token 列表。
2. 对每一个 token 的倒排索引，只需满足一个域的文档即认为是属于该 token 的文档
3. 满足所有 token 的文档才能作为整个查询的文档进行评分
4. 对每个 token 的每个域计算 BM25 并求和，最后加上页面的 PageRank 值。加上 PageRank 值而非相乘，可以避免索引页面总排在最前面而在评分相差不大时获得优势

4.4.2 使用 MultiFieldQueryParser

通过修改框架的方式获得了很大的自由空间，但实现上效率很低，搜索结果用时很长。由此，我们使用 MultiFieldQueryParser 替代自己实现的 SimpleQuery、SimpleSimilarity、SimpleScorer 等类。为了使用 BM25 评分，Lucene 也改为 4.0 版本，相应地，IK Analyzer 也修改了版本。至此，我们使用 Lucene 提供的 BM25Similarity 计算 BM25 评分。

4.4.3 分域权重

我们抽取 1000 个文档进行测试，给各个域赋予不同的权重，作为 boosts 参数传给 MultiFieldQueryParser。在使用整体数据进行测试的过程中，我们也进行了相应调整，最后确定了 100、25、35、1、0.001 的一组权重。

4.4.4 文档摘要

呈现文档时，我们对文档内容抽取摘要进行展示。建立所有 token 在文档内容中的位置构成的集合，从前开始，并呈现 token 前后的 30 个字符；若两个 token 临近则连续输出。

```
public static String genAbstract(List<String> tokens, String content) {
    int maxLength = 300;
    int range = 30;
    String result = "";
    content = content.trim();
    List<Integer> startPositions = new ArrayList<Integer>();
    List<Integer> endPositions = new ArrayList<Integer>();
    for (String t : tokens) {
        String token = new String(t);
        int colonIndex = token.indexOf(':');
        if (colonIndex >= 0) {
            token = token.split(":")[1];
        }
        int pos = 0;
        Pattern pattern = Pattern.compile(token, Pattern.CASE_INSENSITIVE);
        Matcher matcher = pattern.matcher(content);

        int num = 0;
        while (matcher.find(pos) && ++num < maxLength / range) {
            pos = matcher.start();
            startPositions.add(pos);
            endPositions.add(pos + token.length());
            ++pos;
        }
    }
    Collections.sort(startPositions);
    Collections.sort(endPositions);
    int i = 0;
    int size = startPositions.size();
    while (i < size) {
        int pos = startPositions.get(i);
        int end = endPositions.get(i);
        int ptr;
        for (ptr = pos; ptr >= pos - range; --ptr) {
```

```

        if (ptr < 0 || stopChar.contains(content.charAt(ptr))) {
            ++ptr;
            break;
        }
    }
    result += content.subSequence(ptr, pos);
    result += "<em>";
    result += content.subSequence(pos, end);
    result += "</em>";
    ++i;
    while (i < size) {
        pos = startPositions.get(i);
        if (end > pos) {
            result += "<em>";
            result += content.subSequence(end, endPositions.get(i));
            result += "</em>";
            pos = end;
            end = endPositions.get(i);
            ++i;
        } else {
            if (pos == end) {
                result += content.subSequence(end, pos);
                end = endPositions.get(i);
                result += "<em>";
                result += content.subSequence(pos, end);
                result += "</em>";
                ++i;
            } else if (pos - end < range) {
                result += content.subSequence(end, pos);
                end = endPositions.get(i);
                result += "<em>";
                result += content.subSequence(pos, end);
                result += "</em>";
                ++i;
                if (result.length() > maxLength - range) {
                    break;
                }
            } else {
                break;
            }
        }
    }
    for (ptr = end; ptr < end + range; ++ptr) {
        if (ptr >= content.length()
            || stopChar.contains(content.charAt(ptr))) {

```

```

        break;
    }
}
result += content.subSequence(end, ptr) + "... ";

if (result.length() > maxLength) {
    break;
}
}
return result;
}

```

但这种方法并不总能正确呈现查询词的位置。之后，我们使用 Lucene 自带的 highlight 进行处理，一些原来认为毫无关系的文档也能看到相关性。

5 实验结果



The screenshot shows the Tsinghua University News website with a search bar containing '刘奕群 陈旭'. Below the search bar, there are four search results listed:

- 1. 校党委书记陈旭参加计算机系网络所教职工党支部组织生活**
智能支撑的互联网搜索技术及其应用”获得一等奖（技术发明类）。清华大学方面的主要负责人包括计算机系副教授刘奕群、副教授张敬、教授马少平、博士生王超和金奕江工程师。2015.08 31 副校长姜胜耀... 9月24日下午，校党委书记陈旭参加其定点联系的基层单位——计算机系网络所教职工党支部的组织生活，与支部成员一同围绕“三严三实”专题教育以及如何在学校育人的大环境下加强创新创业教育进行交流和讨论...，分别就以三严三实精神开创事业新局面、如何平衡创新创业与学生教育教学、如何平衡技术研发与科学探索的关系、如何有效实现技术成果转移等问题发表了各自观点并展开讨论。 陈旭也交流了她对上述问题的理解和看法
[快照](#)
- 2. 【走进清华辅导员】刘奕群：学生工作是有出息和有帮助的“负担”**
他是严谨而谆谆善诱的良师，在一次又一次谈话中与同学们推心置腹，为他们指点迷津；他是贴心而明言直谏的益友，会细心地了解同学心中的烦恼，用涓涓细流为他们排解忧愁。他就是计算机系学生工作组组长刘奕群。 从年级辅导员到计算机系团委书记，再到校TMS协会会长，刘奕群一路走来，在多个重要的学生工作岗位经受了锻炼。2006年至今，刘奕群担任计算机系学生工作组组长，面对更繁忙的工作，更重的责任，更大的压力，刘奕群笑称“人只有在压力下才会成长”。 迄今为止，刘奕群已发表学术论文20余篇，申请专利3项，博士论文被评为清华大学优秀博士论文，比预定计划提前一年完成博士阶段学习。与此同时，计算机系
[快照](#)
- 3. “强队”夺得2010年清华大学校史知识竞赛冠军**
果于近日揭晓，清华大学计算机系作为第一完成单位，与搜狗公司合作完成的项目“群体智能支撑的互联网搜索技术及其应用”获得一等奖（技术发明类）。清华大学方面的主要负责人包括计算机系副教授刘奕群、副教授张敬...面前。 2015.09 27 校党委书记陈旭参加计算机系网络所教职工党支部的组织生活，与支部成员一同围绕“三严三实”... 2012.04 20 校长陈吉宁到材料系现场办公 校长陈吉宁到材料系现场办公清华大学新闻网4月20日电（记者刘蔚如）4月18日上午，校长陈吉宁到材料系现场办公，就学科发展、人才队伍建设等问题与材料系教师代表
[快照](#)
- 4. 清华对口支援青海大学再谱新篇**
“获得一等奖（技术发明类）。清华大学方面的主要负责人包括计算机系副教授刘奕群、副教授张敬、教授马少平、博士生王超和金奕江工程师。 2015.11 03 陈旭、姜胜耀出席对口支援青海大学工作专题... 10月31日，对口支援青海大学工作专题会议在西北农林科技大学召开。清华大学党委书记陈旭，党委副书记、副校长姜胜耀出席。西北农林科技大学校长孙其信首先致欢迎辞。会议听取了青海大学党委书记俞红波、校长王...新疆大学召开。清华大学党委书记胡和平、常务副书记陈旭，西安交通大学党委书记王建华，武汉大学校长李晓红，中南大学党委书记高文兵，北京师范大学党委书记刘川生，新疆自治区教育厅副厅长宋毅，新疆大学党委书记李中耀
[快照](#)

图 4：刘奕群陈旭

1. 清华超算团队包揽2015年三大国际超算竞赛总冠军

（记者）美国当地时间11月19日在德克萨斯州奥斯汀市举办的SC15国际大学生超算竞赛中，代表清华大学参赛的计算机系超算团队斩获总冠军，这是中国大陆高校第一次在该项赛事中获得总冠军。清华获奖团队合影，从左到右依次为：梁盾、鲁逸沁、梁俊邦、翟季冬老师、裘捷中、卓有为、王懿、李恺威。清华大学超算团队由计算机系师生组成，包括6名本科生：裘捷中、王懿、卓有为、梁盾、梁俊邦、鲁逸沁，以及教练李恺威和指导教师翟季冬。至此，在今年三大国际大学生超算竞赛ASC15、ISC15、SC15中，清华大学超算团队包揽了三项国际赛事的冠军。SC与ISC和ASC并列为国际三大大学生超级计算机赛事，每年举办

[快照](#)

2. 首届ASC超算大赛落幕 力促超算技术的普及应用

ASC超算大赛落幕 力促超算技术的普及应用 中国新闻网 2013-4-22 日前，首届ASC（Asia Student Supercomputer challenge）亚洲大学生超级计算机竞赛...。据悉，该竞赛是与美国SC、德国ISC大学生超算大赛并驾齐驱的全球三道超算赛事之一。大赛规则要求各大学生参赛队在3000瓦的功耗限制下自行设计搭建超级计算机系统，并进行HPL、GROMACS、OPENCFD、WRF、BSDE等5项应用优化，参赛队还需要将各自的方案和优化策略对评审委员会现场呈现，这对各参赛队伍的超算应用能力提出了全面挑战。进入决赛的十支队伍是从报名参加初赛的亚洲各国和地区

[快照](#)

3. 清华学生夺国际“超算”总冠军

举办一次，吸引着世界各个国家和地区的众多高校参与。报名参加超算大赛并不是一件容易的事情，首先必须要有足够支撑高速运算的设备，也就是超级计算机。“这和我们平时用的笔记本、台式机家用计算机完全不是一个概念...根本就不算熬夜。”王懿说。150多个日日夜夜的辛勤备战最终迎来考验。11月13日，清华超算团队经旧金山飞往比赛地——得克萨斯州奥斯汀市。裘捷中说，在候机时队员们仍然抱着笔记本进行最后的测试优化...，而一旦不能多机运行，之前的付出将功亏一篑。他开始了紧急处理，“通俗的说就是一方面让已经驶上高速公路的车速度慢下来，一方面利用时间赶紧抢通高速公路。”15分钟后故障终于被顺利排除。最终，清华大学超算

[快照](#)

4. 清华学生超算团队获世界大学生超级计算机竞赛总冠军

28日，世界大学生超级计算机竞赛（ASC17）总决赛在国家超级计算无锡中心落下帷幕，清华学生超算团队夺得总冠军，并同时获得“e Prize计算挑战奖”。至此，在世界大学生超级计算机竞赛过去7届比赛中，清华共有4次获得总冠军。本次大赛获冠军的同学。（左二起：叶方柯、刘家昌、王懿、李北辰、李宇轩、冯冠宇）本次竞赛要求各参赛队伍在3千瓦功率的限制条件下利用组委会提供的浪潮超算节点搭建计算机集群系统，考察内容包括数值计算、基因拼接、流体力学、分子动力学、海洋模拟、深度学习等方面的应用。计算机系组织的学生超算团队从去年秋季学期开始进行队员选拔和培训，队员们前期对各个应用进行了细致研究，做了

[快照](#)

图 5：超算



图 6：长文本搜索

6 心得体会

实验开始，我们完成了对图像搜索框架的修改，并实现了摘要提取，费尽周章进行调试，但是效果始终不理想。一个是响应速度慢，需要十几秒，一个是显示的摘要大多时候表现不出和查询的相关性。

这个时候，我们采用了 Lucene 自带的 MultiFieldQueryParser 进行查询，显著缩短了查询时间，增强了查询体验；同时，Lucene 框架内的 highlights 提供了更加友好的摘要，使得一些标题好像根本不相关的页面也表现出了相关性。与此同时，原有的代码被删减近半。

痛惜之余，我们也感受到了开源社区的优越性，反复造轮子的过程是对时间的浪费，多使用已有的工具包可以获得更好的效果。回过头来看，似乎所有的工作在最后一天下午又重新做起了。

在重新构建框架的过程中也进一步发现了很多有用的开源工具，也发现了很多可进一步扩展的功能，不过由于前期反复造轮子耗时过多导致最后时间不够没能进一步实现。