**吉林大学**

《c#程序设计》

结课作业

学院：软件学院

年级：2015级

成员与分工

|  |  |  |  |
| --- | --- | --- | --- |
| 成员 | 学号 | 工作 | 工作量 |
| 李祥锐 | 54150423 | 爬虫与报告编写 | 40% |
| 宋欢丞 | 54150221 | 词库,网页与报告编写 | 40% |
| 夏志龙 | 54150519 | 报告编写 | 10% |
| 向海东 | 54150317 | 报告编写 | 10% |

# Abstract

爬虫技术是当前获取网页数据最简单有效的手段之一，c#编程语言也是实现爬虫技术的中主流编程语言之一。本文描述了基于c#编程语言爬取吉大校园网新闻信息的爬虫实现过程、爬取数据、数据分析结果和用于展示结果的词云，并将结果展示在html网页中。

**Keyword:**c#; spider; website; wordcloud

Table of Contents

[Chapter One - Introduction](#_Toc5992_WPSOffice_Level1) [4](#_Toc5992_WPSOffice_Level1)

[1. 主页，通过日期排序展示所有新闻](#_Toc10050_WPSOffice_Level1) [4](#_Toc10050_WPSOffice_Level1)

[2. 展示页面，点击新闻后有新闻的关键词的词云及新闻分词的结果](#_Toc19236_WPSOffice_Level1) [4](#_Toc19236_WPSOffice_Level1)

[Chapter Three - Methods](#_Toc17519_WPSOffice_Level1) [5](#_Toc17519_WPSOffice_Level1)

[Chapter Four - Data Analysis and Results](#_Toc11692_WPSOffice_Level1) [7](#_Toc11692_WPSOffice_Level1)

[1.统计结果](#_Toc9926_WPSOffice_Level1) [7](#_Toc9926_WPSOffice_Level1)

[2.程序运行截图](#_Toc16499_WPSOffice_Level1) [8](#_Toc16499_WPSOffice_Level1)

[3.Jieba分词](#_Toc14896_WPSOffice_Level1) [9](#_Toc14896_WPSOffice_Level1)

[4.网页设计](#_Toc14239_WPSOffice_Level1) [10](#_Toc14239_WPSOffice_Level1)

[Chapter Five - Conclusion](#_Toc654_WPSOffice_Level1) [12](#_Toc654_WPSOffice_Level1)

[References](#_Toc3711_WPSOffice_Level1) [13](#_Toc3711_WPSOffice_Level1)

Chapter One - Introduction

本项目爬取了吉大新闻网的校园快讯，日期为2017-1至2018-5。爬取的信息有新闻的标题，发布时间，发布部门/作者，文章内容。

并对数据进行了简单分析。分析内容包括，爬取新闻的总量，每周的新闻总量，每天平均新闻数量，每周平均新闻数及每个部门的平均新闻数，将其保存在spider文件夹中

对数据的分词分为两部分。

1. 对于每天新闻的分词，采用精确模式，并对其进行了删除停止符。
2. 对于每天新闻的关键词提取，采用基于 TF-IDF 算法的关键词抽取

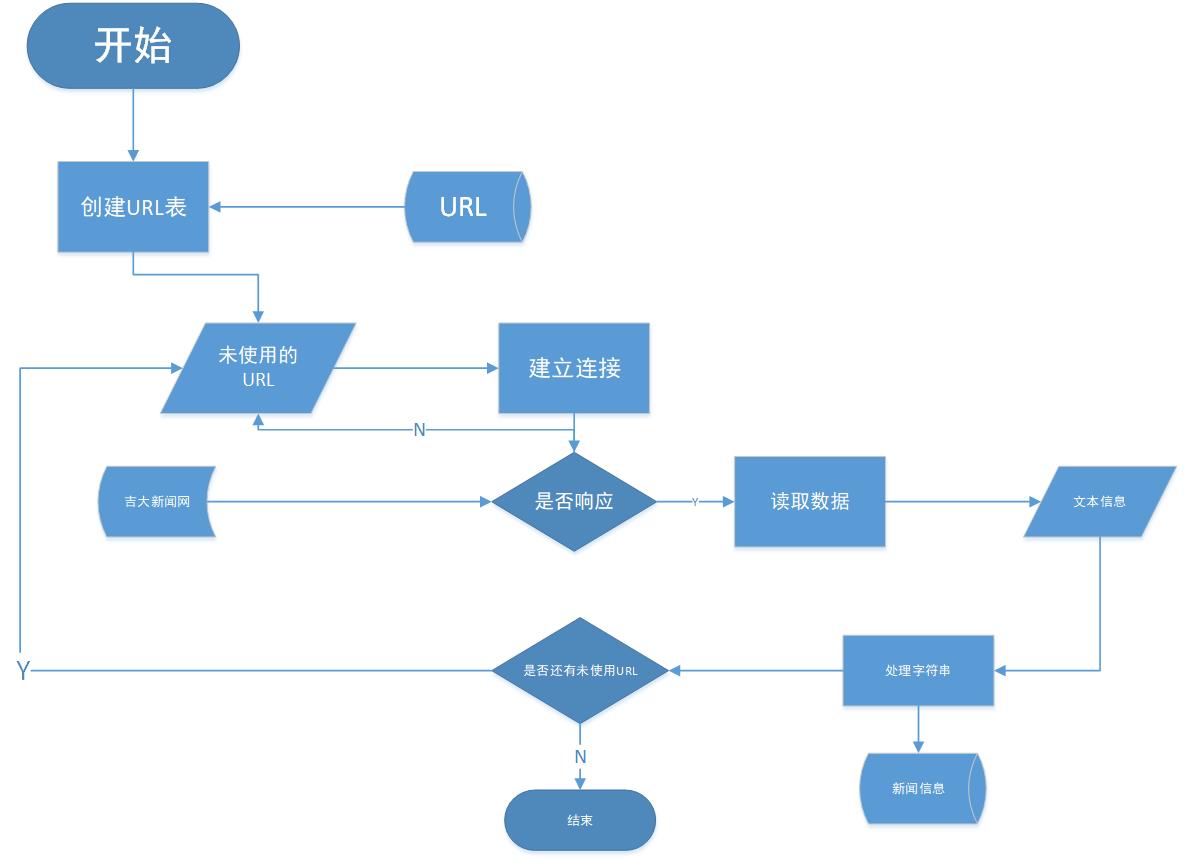
将结果保存在output文件夹中

网页的展示分为两部分

1. 主页，通过日期排序展示所有新闻
2. 展示页面，点击新闻后有新闻的关键词的词云及新闻分词的结果

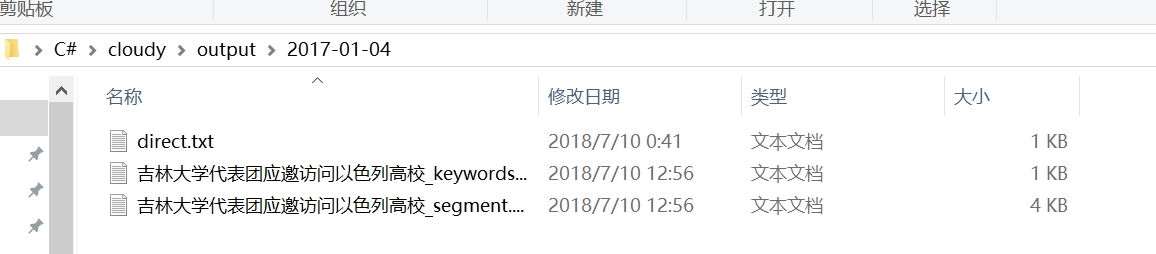
Chapter Three - Methods

## 爬虫原理

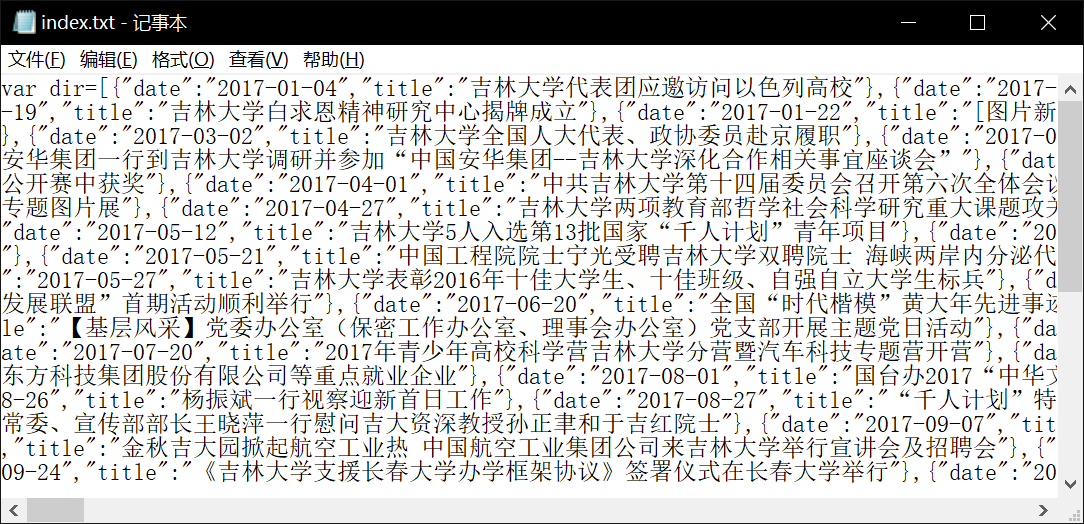


## Jieba分词

1. 提取所有爬出的数据，并保存在数组中
2. 对于数组中的每条数据，分别进行jieba的分词，关键字提取（本项目采用TF-INF提取，TextRank的代码在最后有注释），并对分词进行删除停止符。并保存到output文件夹中。\_keywords保存关键字，\_segment保存分词。



1. 对每条数据分别提取日期和表题，保存在index.txt中。



## 网页设计

1.Index.html中有所有新闻的标题和时间，并在底部设置了一个table可供跳转。具体css参照学校网站设置。

2.showNews.html中为点开新闻后的展示页面，将keywords展示为词云，并将segment直接输出（只将内容展示，并未关注页面布局和css）

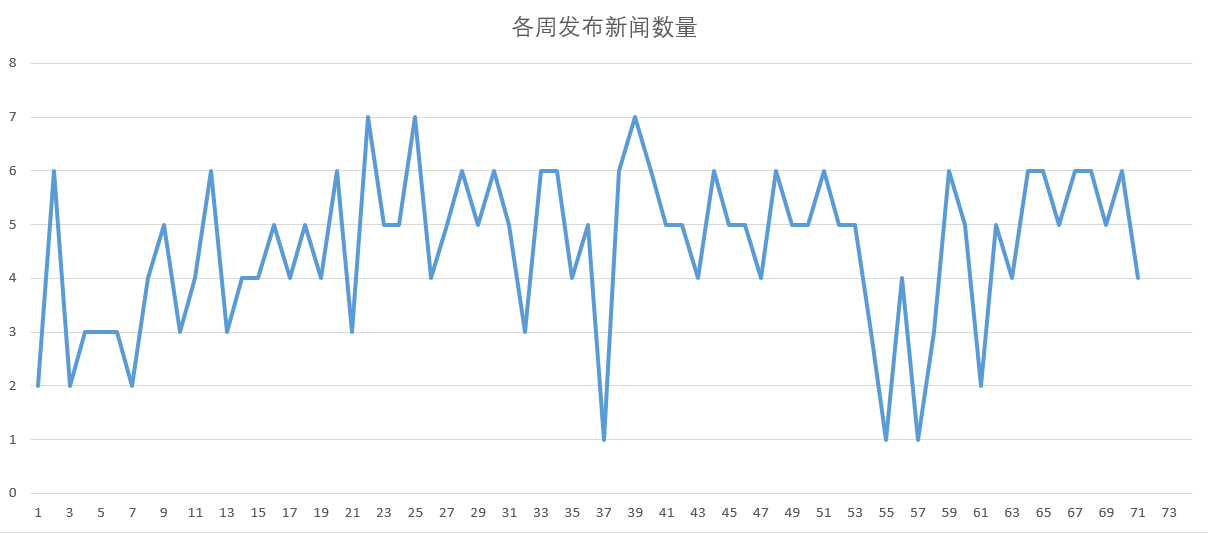
3.所有文档中数据的提取均通过script标签中的src来引入

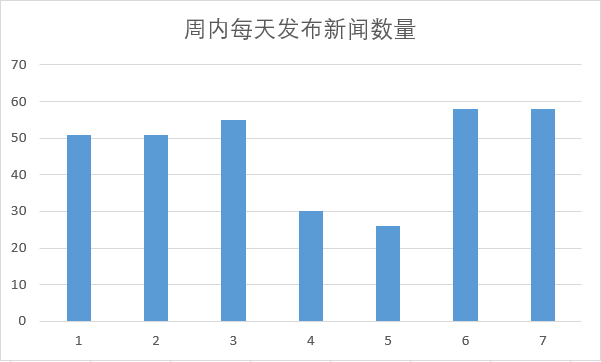
Chapter Four - Data Analysis and Results

## 1.统计结果

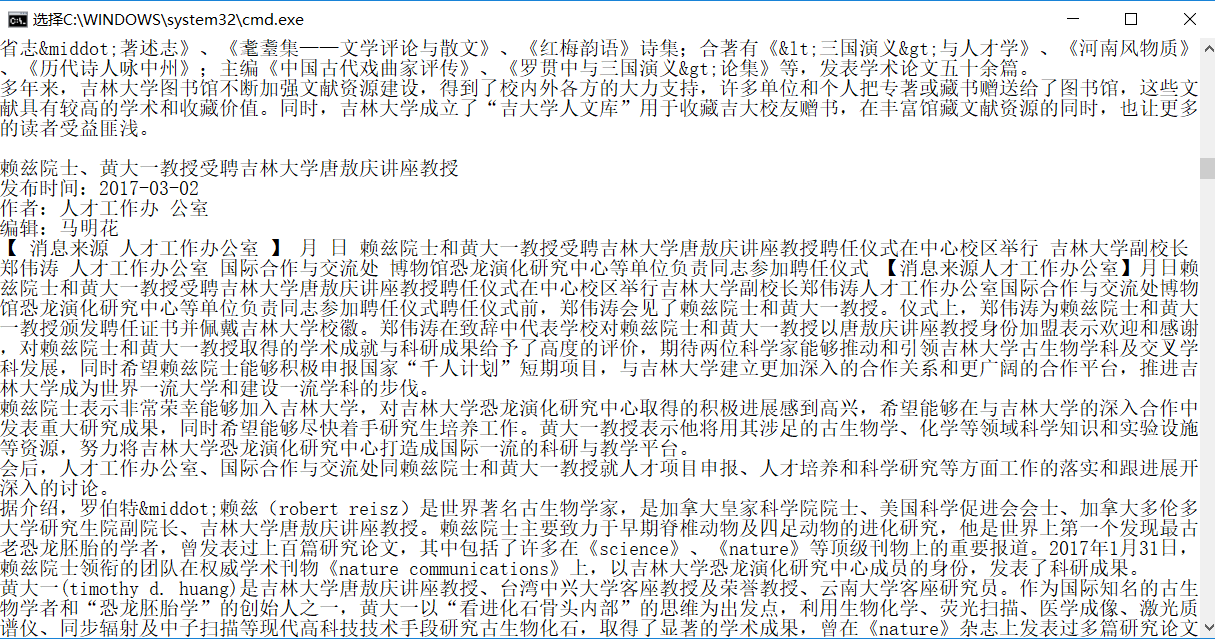
|  |  |
| --- | --- |
| 属性 | 数量 |
| 信息总数 | 567 |
| 统计天数 | 330 |
| 统计周数 | 74 |
| 统计部门（包含个人） | 140 |
| 平均每天 | 1.7条 |
| 平均每周 | 7.66条 |
| 平均每个部门 | 4.05条 |

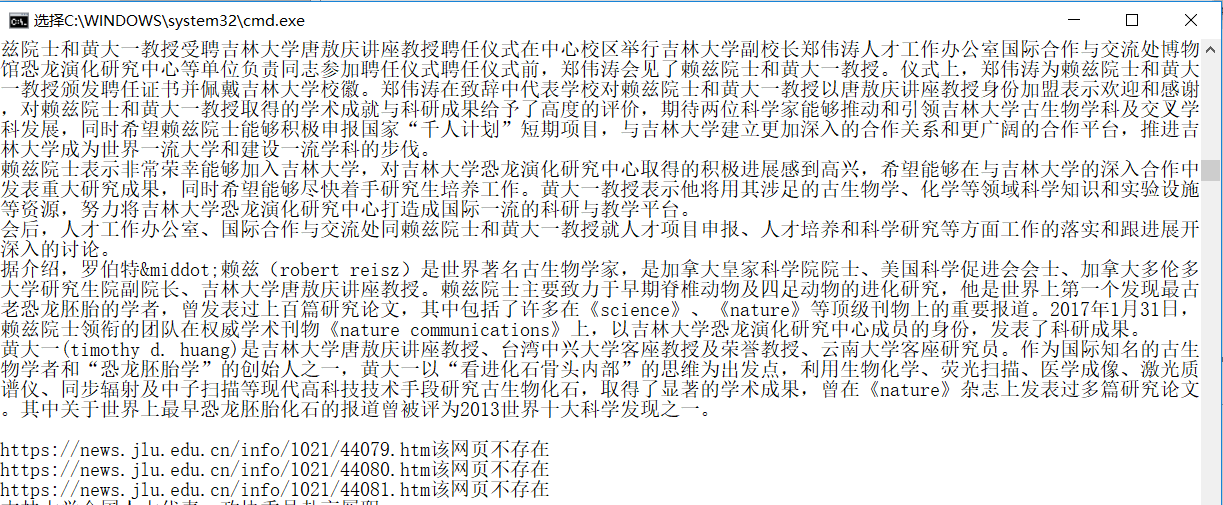
注：并没有用boxplot，而是将数据保存在txt文件中，详请可见压缩包

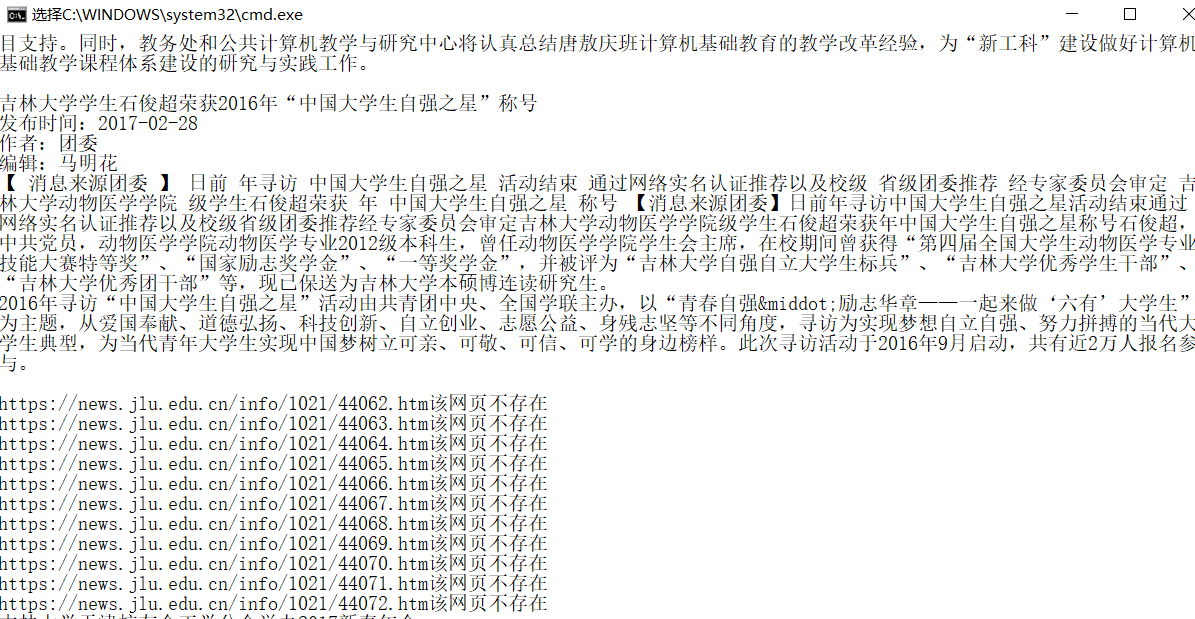




## 2.程序运行截图

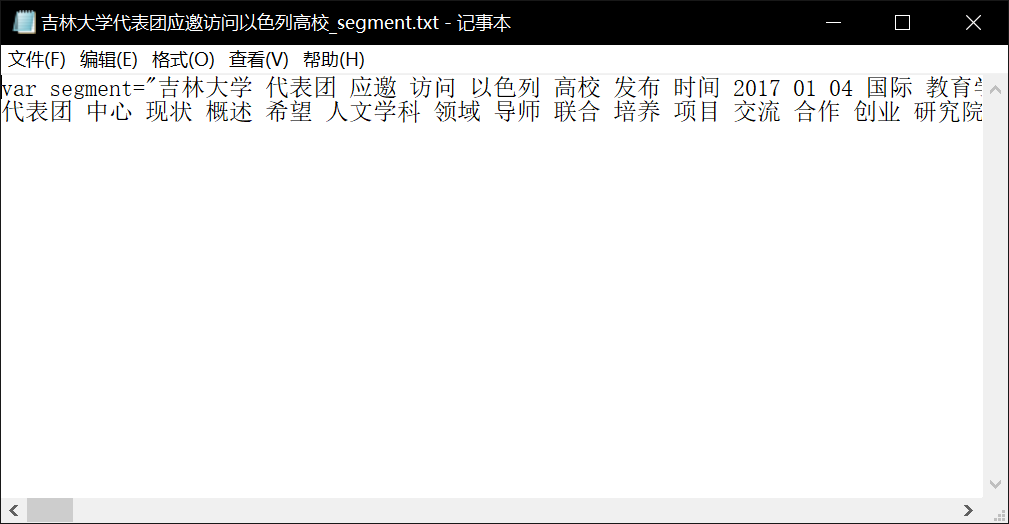




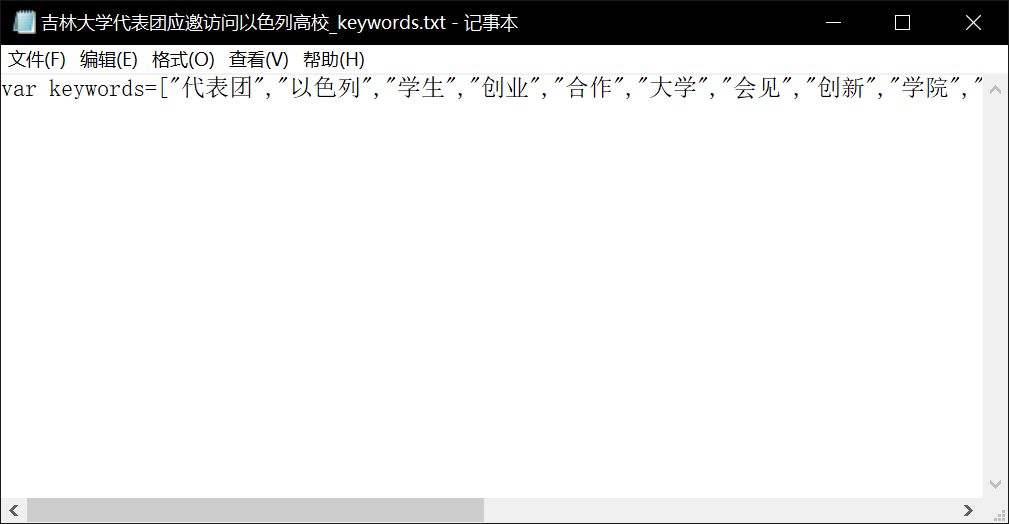


## 3.Jieba分词

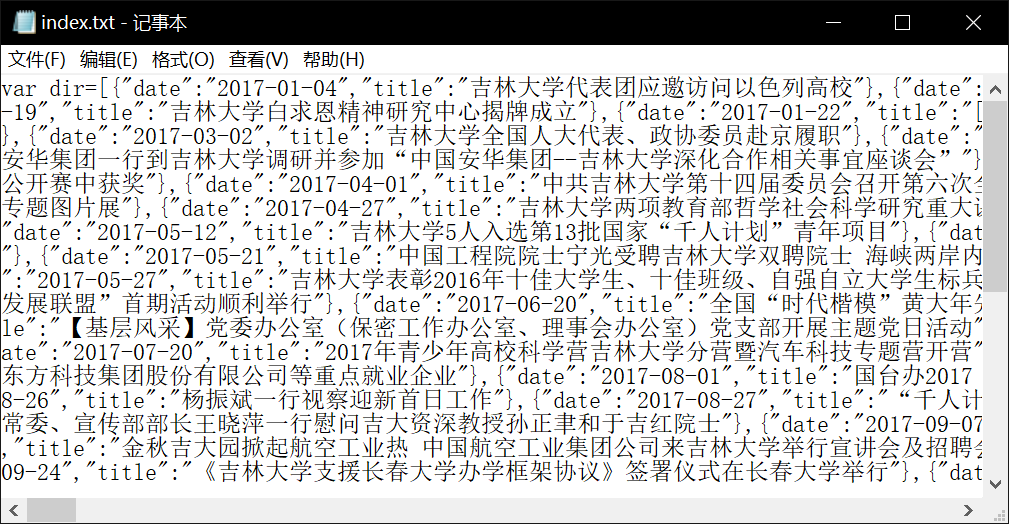
分词：



关键词：



新闻目录:



## 4.网页设计

主页：



展示页面：



Chapter Five - Conclusion

本次课程作业令我们小组成员获益良多。我们更加深刻地熟悉了C#开发，体验了爬虫的制作过程，了解了jieba,词云,及网页的相关知识。并在制作过程中体会到了小组合作的重要性。

我们的作业基本完成了作业的要求，部分内容可能由于理解原因有所出入。以下是我们制作过程中感受到的不足

字符串处理的不好，一部用正则表达式，一部分用string.split，有些文档中</strong>加粗 没有取掉

2. 存在有两个文件相同内容，可能是同一条新闻发了两次

3. 有些文件名有一点问题，仅仅包含.txt，是我在字符串处理时，标题没有截好。

4. 数据统计，没有实现使用boxplot，而是将数据存储在txt文件中，并用c#和matlab一起分析。

5. 一开始由于python版本问题和空格等浪费了很长时间

6. output文件夹的输出花费时间较长，且方法比较笨，将所有目录通过‘\’分开后一点一点创建

7. 在文件的读写上花费了大量时间，尤其是java script中的文件读写，操作较为困难，最后无奈选择通过script标签引入，这样做会降低网页的速度。

8. 词云展示时仅将keywords展示，并为根据其score区分字体大小，而是随机分配

9. 网页设计美观度不够，并未太注重布局和css的书写。

References

[1]<https://blog.csdn.net/tindoc/article/details/53648189>

[2]<https://github.com/fxsjy/jieba>

[3]<https://blog.csdn.net/wireless_com/article/details/60571394>

[4]<https://www.jasondavies.com/wordcloud/>

[5]<https://d3js.org/>

[6]<https://blog.csdn.net/u010533386/article/details/51458591>

[7]<https://blog.csdn.net/gy_grace/article/details/72863646>