

Java网页正文提取工具



泊牧

□ 已关注

2017.09.18 15:05* 字数 533 阅读 117 评论 0 喜欢 0

最近做一个项目，其中涉及到网页信息采集，随后对相关的技术进行了学习与研发，网页正文提取技术常用的有joyhtml、boilerpipe、cx-extractor下面将对其做一个简单的对比说明，和大家进行下分享。

相关技术

1. joyhtml: <http://code.google.com/p/joyhtml/>

JoyHTML的目的是解析HTML文本当中的链接和正文，利用超链接密度法为主要判断依据的标记窗算法，采用DOM树解析模式。

2. boilerpipe: <http://code.google.com/p/boilerpipe/>

这个Java类库提供算法来探测和删除在一个网页中主文本内容旁多余的重复内容。它已经有提供特殊的策略来处理一些常用的功能如：新闻文章提取

依赖的lib:

[lib.rar](#)

[boilerpipe-1.2.0-bin.tar.gz](#)

使用示例:

```
public static void main(String[] args) throws Exception {
    String url = "http://finance.people.com.cn/n/2013/1011/c66323-23157265.html";
    TextDocument doc = new BoilerpipeSAXInput(new InputSource(new URL(url).openStream()))
        .getTextDocument();
    BoilerpipeExtractor extractor = CommonExtractors.ARTICLE_EXTRACTOR;
    extractor.process(doc);
    System.out.println("title:" + doc.getTitle());
    System.out.println("content:" + doc.getContent());
}
```

3. cx-extractor: <http://code.google.com/p/cx-extractor/>

本算法首次将网页正文抽取问题转化为求页面的行块分布函数，并完全脱离HTML标签。通过线性时间建立行块分布函数图，由此图可以直接高效、准确的定位网页正文。同时采用统计与规则相结合的方法来解决系统的通用性问题。

4. WebCollector/ContentExtractor: <https://github.com/CrawlScript/WebCollector>

WebCollector的正文抽取API都被封装为ContentExtractor类的静态方法。可以抽取结构化新闻，也可以只抽取网页的正文（或正文所在Element）。

正文抽取效果指标：

比赛数据集CleanEval P=93.79% R=86.02% F=86.72%

常见新闻网站数据集 P=97.87% R=94.26% F=95.33%

算法无视语种，适用于各种语种的网页。

标题抽取和日期抽取使用简单启发式算法。

调用方法：

```
News news = ContentExtractor.getNewsByHtml(html, url);
News news = ContentExtractor.getNewsByHtml(html);
News news = ContentExtractor.getNewsByUrl(url);

String content = ContentExtractor.getContentByHtml(html, url);
String content = ContentExtractor.getContentByHtml(html);
String content = ContentExtractor.getContentByUrl(url);

Element contentElement = ContentExtractor.getContentElementByHtml(html, url);
Element contentElement = ContentExtractor.getContentElementByHtml(html);
Element contentElement = ContentExtractor.getContentElementByUrl(url);
```

最终选择WebCollector

理由如下：

简单：java代码不超过400行

准确率高：>95%

算法时间复杂度为线性

小礼物走一走，来简书关注我

赞赏支持

[自然语言处理](#)

[举报文章](#) [© 著作权归作者所有](#)



泊牧 [👤](#)

写了 20154 字，被 15 人关注，获得了 11 个喜欢

[👤 已关注](#)

喜欢



更多分享



评论

智慧如你，不想[发表一点想法](#)咩~

被以下专题收入，发现更多相似内容

[收入我的专题](#)

掘金 **Android** 文章精选合集

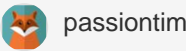
用两张图告诉你，为什么你的 App 会卡顿？ - Android - 掘金Cover 有什么料？从这篇文章中你能获得这些料：知道setContentView()之后发生了什么？ ... Android 获取 View 宽高的常用正确方式，避免为零 - 掘金...



掘金官方

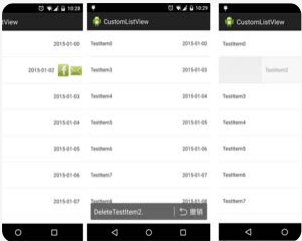
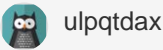
Android - 收藏集

用两张图告诉你，为什么你的 App 会卡顿？ - Android - 掘金 Cover 有什么料？ 从这篇文章中你能获得这些料： 知道setContentView()之后发生了什么？ ... Android 获取 View 宽高的常用正确方式，避免为零 - 掘金...



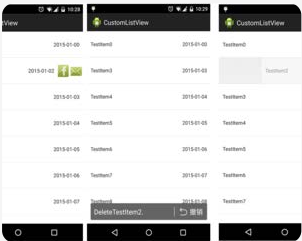
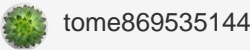
2017-05-30

【转】Android 开源项目分类汇总 旭川君已关注 2017.08.15 16:49*字数 29527阅读 1795评论 1喜欢 35 来源：<https://github.com/Trinea/android-open...>



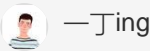
自定义控件收藏

Android 开源项目分类汇总，更全更新可见codekk.com目前包括： Android 开源项目第一篇——个性化控件(View)篇 包...



即使漂泊，也别让思想流浪

文/熊赳赳 其实，关于接下来要写点什么，其实我也没有想好。这个号很早之前就开了，那还是我上一次创业，和朋友开了个影视公司叫“上集”。结果做了上...



一个守山人眼中的绝美三清山

三清山又名少华山、丫山，位于中国江西省上饶市玉山县与德兴市交界处。因玉京、玉虚、玉华三峰宛如道教玉清、上清、太清三位尊神列坐山巅而得名。 ...




操蛋的人生 总会总会越来越好的吧

近来，很多杂事萦绕心头，感觉自己被自我压得喘不过气了。每每心情不太好的时候就喜欢写写文字，文笔拙劣，却又依然死皮赖脸地赖着它。感觉自己...



微商如何在洗牌浪潮中独善其身

朋友圈的创富传奇正在像多米诺骨牌一样倒塌。 从今年五月份开始，持续火爆了近一年的微商市场骤然降温，微商靠朋友圈晒图成交的单渠道销售模式已经不再奏效，绝大多数的微商都遭遇到了业绩的下滑，甚...

 极客志强

使用Python进行网页正文提取



泊牧

□ 已关注

2017.09.18 11:25* 字数 451 阅读 159 评论 0 喜欢 0

1. Goose Extractor

1.1 Python Goose介绍

[Goose Extractor](#)是一个Python的开源文章提取库。可以用它提取文章的文本内容、图片、视频、元信息和标签。[Goose](#)本来是由Gravity.com编写的Java库，最近转向了scala。

Goose Extractor网站是这么介绍的：

Goose Extractor完全用Python重写了。目标是给定任意资讯文章或者任意文章类的网页，不仅提取出文章的主体，同时提取出所有元信息以及图片等信息。

Goose Extractor基于[NLTK](#)和[Beautiful Soup](#)，分别是文本处理和HTML解析的领导者。用Python进行文章提取可以使用Python Goose。

Goose目前只支持Python2

1.2 安装Python Goose

```
pip install goose-extractor
```

直接使用Url链接抽取示例：

```
from goose import Goose

url = 'https://www.fireeye.com/blog/executive-perspective/2017/08/fireeye-provides-update-on-allegations-of-breach.html'
```



```
g = Goose()
article = g.extract(url=url)
print article.title
print article.meta_description
print article.cleaned_text[:150]
print article.top_image.src
```

使用**Html**文档抽取示例：

```
# -*- coding: utf-8 -*-
import goose,urllib2,sys

reload(sys)
sys.setdefaultencoding("utf-8")

#url = "https://www.fireeye.com/blog/executive-perspective/2017/08/anti-encryption-and-
cyber-sovereignty.html"

url = "https://krebsonsecurity.com/2017/09/equifax-hackers-stole-200k-credit-card-
accounts-in-one-fell-swoop/"
opener = urllib2.build_opener(urllib2.HTTPCookieProcessor())
response = opener.open(url)
raw_html = response.read()
g = goose.Goose()
article = g.extract(raw_html=raw_html)
print article.title.encode('gbk', 'ignore')
print article.meta_description.encode('gbk', 'ignore')
print article.cleaned_text.encode('gbk', 'ignore')
```

1.3 urllib2获取的HTML网页乱码问题

网页可能是压缩了，看里面是不是有 Content-Encoding:xxx

如果是压缩了，需要手动解压，urllib是不会帮你解压的

解决代码：

```
# -*- encoding: utf-8 -*-
import urllib2,gzip,StringIO

url = r'https://krebsonsecurity.com/2017/09/equifax-hackers-stole-200k-credit-card-
accounts-in-one-fell-swoop/'
response = urllib2.urlopen(url)
stream = StringIO.StringIO(response.read())
with gzip.GzipFile(fileobj=stream) as f:
    data = f.read()
print(data)
```

附一篇文章谈Python编码：[也谈Python的中文编码处理](#)

2. Boilerpipe

Github开源代码：[Boilerpipe](#)

在开源系统里Boilerpipe的precision和recall都好过Goose，甚至比收费的Alchemy API还要好。Boilerpipe是Java的，在Python里调用需要用python-boilerpipe这个包装，它底层用的是jpye。也可以用JCC来调。代码如下：

安装：

```
git clone https://github.com/misja/python-boilerpipe.git
cd python-boilerpipe
pip install -r requirements.txt
python setup.py install
```

使用：

```
from boilerpipe.extract import Extractor

url = "https://krebsonsecurity.com/2017/09/equifax-hackers-stole-200k-credit-card-accounts-in-one-fell-swoop/"
extractor = Extractor(extractor='ArticleExtractor', url=url)
print extractor.getText().encode('gbk', 'ignore')
```

或传入一个HTML文本作为参数：

```
extractor = Extractor(extractor='ArticleExtractor', html=myWebPage)
```

用getText() or getHTML() 拿回处理过的纯文本或加亮了正文的HTML

```
processed_plaintext = extractor.getText()
highlighted_html = extractor.getHTML()
```

也可以用JCC把Java的包编译成Python可以调用的包

```
wget http://boilerpipe.googlecode.com/files/boilerpipe-1.2.0-bin.tar.gz
tar xvzf boilerpipe-*.tar.gz
cd boilerpipe-1.2.0
sudo python -m jcc \ --jar boilerpipe-1.2.0.jar \ --classpath lib/neohtml-1.9.13.jar \ -
--classpath lib/xerces-2.9.1.jar \ --package java.net \ java.net.URL \ --python boilerpipe
```



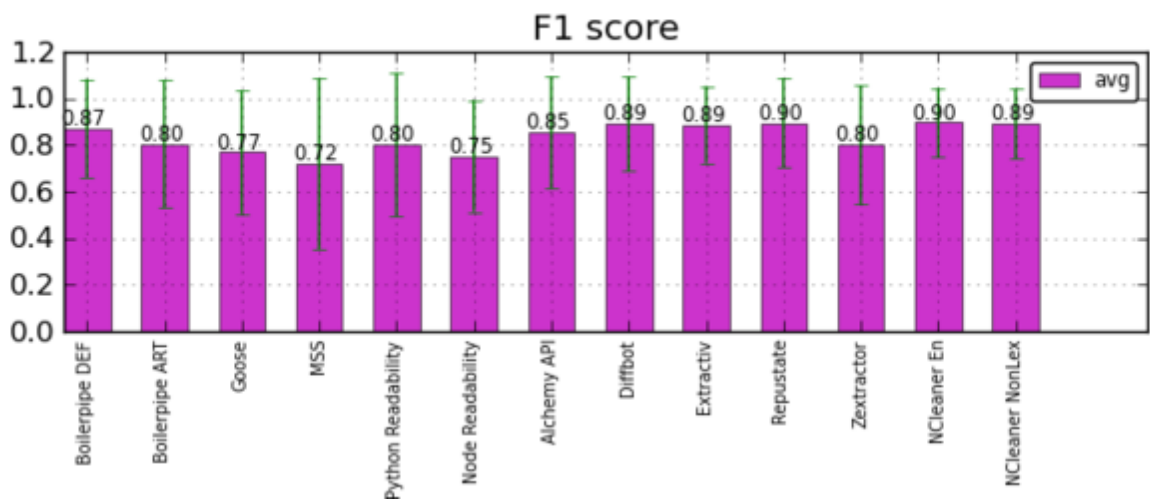
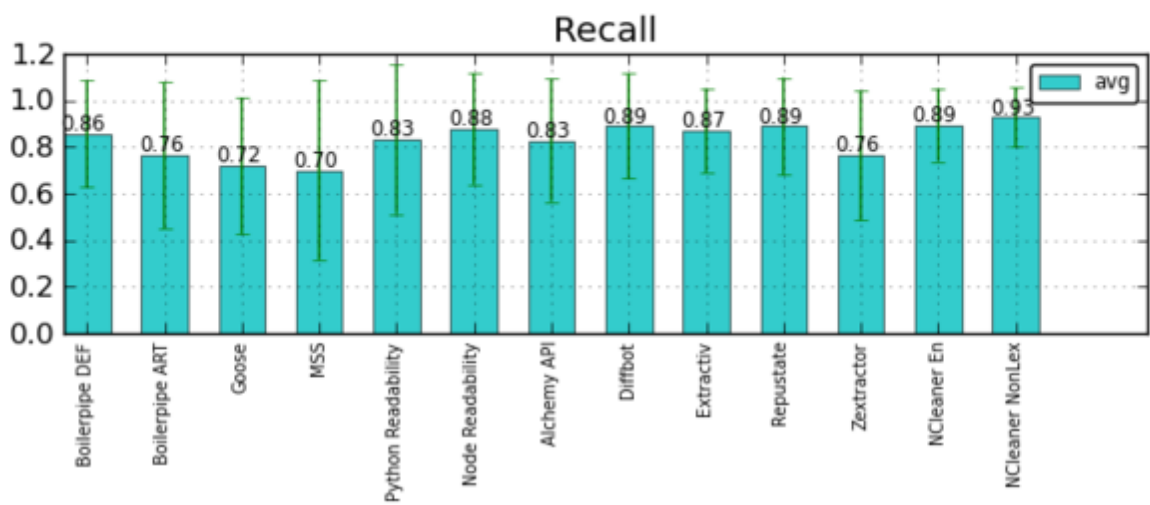
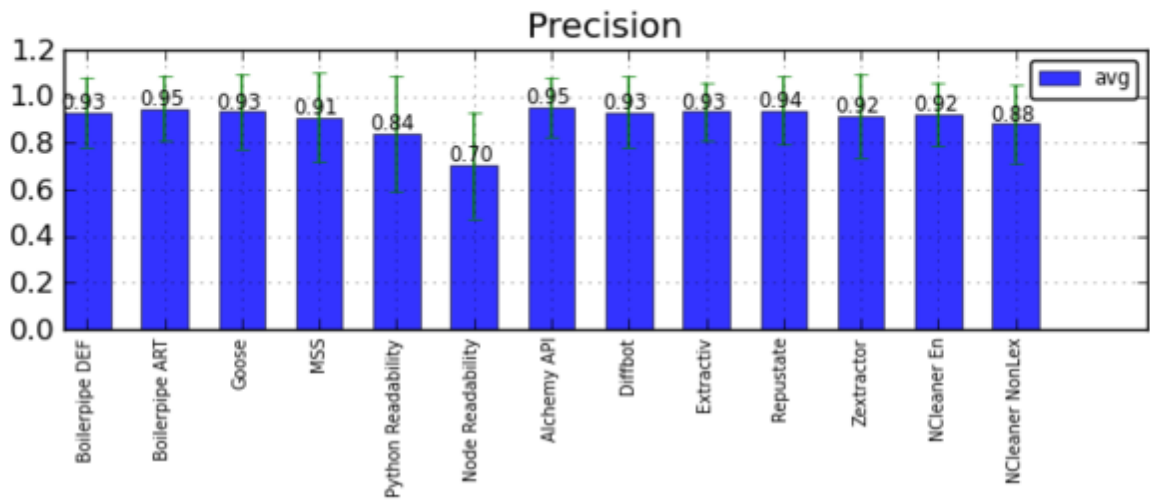
```
--build --install
```

使用:

```
import boilerpipe

jars = ':'.join(('lib/nekohtml-1.9.13.jar', 'lib/xerces-2.9.1.jar'))
boilerpipe.initVM(boilerpipe.CLASSPATH+':'+jars)
extractor = boilerpipe.ArticleExtractor.getInstance()
url = boilerpipe.URL('http://readthedocs.org/docs/jcc')
extractor.getText(url)
```

3. 各种Python正文抽取工具比较



各种Python正文抽取工具比较


小礼物走一走，来简书关注我

赞赏支持

 自然语言处理

[举报文章](#) © 著作权归作者所有



泊牧 

写了 20154 字，被 15 人关注，获得了 11 个喜欢

 已关注

喜欢




更多分享



评论

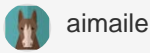
智慧如你，不想[发表一点想法](#)咩~

 被以下专题收入，发现更多相似内容

 收入我的专题

Python--Flask Django等常用库总结

Python 资源大全中文版 我想很多程序员应该记得 GitHub 上有一个 Awesome - XXX 系列的资源整理。[\[awesome-python\]\(https://link.jianshu.com?t=https%3A%2F%2Fgithub.com%2Fvin...\)](#)



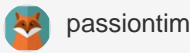
Python 资源大全中文版

GitHub 上有一个 Awesome - XXX 系列的资源整理,资源非常丰富, 涉及面非常广。awesome-python 是 vinta 发起维护的 Python 资源列表, 内容包...



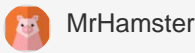
Android - 收藏集

用两张图告诉你, 为什么你的 App 会卡顿? - Android - 掘金 Cover 有什么料? 从这篇文章中你能获得这些料: 知道setContentView()之后发生了什么? ... Android 获取 View 宽高的常用正确方式, 避免为零 - 掘金...



python库收集贴

环境管理管理Python版本和环境的工具。p-非常简单的交互式python版本管理工具。pyenv-简单的Python版本管理工具。Vex-可以在虚拟环境中执行命令。virtualenv-创建独立Python环境的工具。virtualenvwrapp...



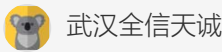
2017-11-29

2017.11.28 星期三 第4篇 看见孩子这几天学习又用心了, 做父母心里非常高兴, 现在非常喜欢看书, 也不太跑神了, 我担心的是孩子的眼睛, 应该看多长时间才合适, 因为她现在一本书最多天就看完, 只要回来就...



【震惊】北京伽途ix系列官降来袭 或将掀起MPV市场...

今年“十一”黄金周期间, 游客纷纷选择阖家出行, 自驾游取代团队游成为主流, MPV成了多数家庭的选择。对于消费者来说, 花合适的钱, 买到一款够...



晨读感悟：北有大张伟，南有薛之谦——这么折腾才能...

今天被大张伟和薛之谦的一首《意外》扎了心, 咱们就来聊聊这两个娱乐圈的“意外”。“北有大张伟, 南有薛之谦”这句话早就在娱乐圈流传, 这两个被歌...



2016 年有哪些值得一看的纪录片？

抓取了哔哩哔哩**站2016年所有的纪录片，下面按照播放量、收藏数量以及硬币数量分别进行排名，排名由高到低，每一项取前50名，已经剔除只有会员才能看的视频。 1、按照播放量从高到低（格式：播放量 纪...



不稚名

你是我流浪的地方

我醒来的时候 祈祷声早已停止 白色的你也悄无声息 虔诚者手捧经文路过 他此生幸福，来世安康 而我一无所有 窗前甚至没有荫凉 太阳依旧比血液滚烫 到哪里都一样 燃烧它生长的一切 而血 因为出生的时候见过 所...



中习习



基于机器学习的网页抽取



泊牧

□ 已关注

2017.07.12 16:41* 字数 3448 阅读 1534 评论 2 喜欢 4

由于最近在做一个项目，给了36个安全网站相关的博客网站，需要将其中的博客正文都抽取出来，而且需要满足以后添加一个博客网站的链接，就可以自动完成正文的抽取工作。

以前写过的爬虫是正则或CSS选择器(或xpath)的网页抽取都基于属于基于包装器(wrapper)的网页抽取，但是这类抽取算法有一个通病，对于不同结构的网页，要制定不同的抽取规则。如果一个安全态势感知系统需要获取1000个异构网站的博客正文，就需要编写并维护1000套抽取规则，这太恶心了，根本就是不想完成的任务。

从2000年左右就开始有人研究如何用机器学习的方法，让程序在不需要人工制定规则的情况下从网页中提取所需的信息。从目前的科研成果看，基于机器学习的网页抽取的重心偏向于新闻网页内容自动抽取，即输入一个新闻网页，程序可以自动输出新闻的标题、正文、时间等信息。新闻、博客、百科类网站包含的结构化数据较为单一，基本都满足{标题，时间，正文}这种结构，抽取目标很明确，机器学习算法也较好设计。

题外话：这种正文提取算法可以帮助提取安全博客网站的正文，但是一些电商、求职等类型的网页中包含的结构化数据非常复杂，有些还有嵌套，并没有统一的抽取目标，针对这类页面设计机器学习抽取算法难度较大。

下面主要描述如何设计机器学习算法抽取新闻、博客、百科等网站中的正文信息，后面简称为网页正文抽取(Content Extraction)。

基于机器学习的网页抽取算法大致可以分为以下几类：

- 基于启发式规则和无监督学习的网页抽取算法
- 基于分类器的网页抽取算法
- 基于网页模板自动生成的网页抽取算法

三类算法中，第一类算法是最好实现的，也是效果最好的。



下面简单描述一下三类算法，如果你只是希望在工程中使用这些算法，只要了解第一类算法即可。

下面会提到一些论文，但请不要根据论文里自己的实验数据来判断算法的好坏，很多算法面向早期网页设计（即以表格为框架的网页），还有一些算法的实验数据集覆盖面较窄。有条件最好自己对这些算法进行评测。

1. 基于启发式规则和无监督学习的网页抽取算法

基于启发式规则和无监督学习的网页抽取算法（第一类算法）是目前最简单，也是效果最好的方法。且其具有较高的通用性，即算法往往在不同语种、不同结构的网页上都有效。

早期的这类算法大多数没有将网页解析为DOM树，而是将网页解析为一个token序列，例如对于下面这段html源码：

```
<body>
  <div>广告...(8字)</div>
  <div class="main">正文...(500字)</div>
  <div class="foot">页脚...(6字)</div>
</body>
```

程序将其转换为token序列：

```
标签(body), 标签(div), 文本, 文本...(8次), 标签(/div), 标签(div), 文本, 文本...(500次), 标签(/div), 标签(div), 文本, 文本...(6次), 标签(/div), 标签(/body)
```

早期有一种MSS算法(Maximum Subsequence Segmentation)以token序列为基础，算法有多个版本，其中一个版本为token序列中的每一个token赋予一个分数，打分规则如下：

- 一个标签给 -3.25 分
- 一个文本给 +1 分

根据打分规则和上面的token序列，我们可以获取一个分数序列：

```
-3.25, -3.25, 1, 1, 1...(8次), -3.25, -3.25, 1, 1, 1...(500次), -3.25, -3.25, 1, 1, 1...(6次), -3.25, -3.25
```

- **MSS算法**

MSS算法认为，找出token序列中的一个子序列，使得这个子序列中token对应的分数总和达到最大，则这个子序列就是网页中的正文。从另一个角度来理解这个规则，即从html源码字符串中找出一个子序列，这个子序列应该尽量包含较多的文本和较少的标签，因为算法中给标签赋予了绝对值较大的负分(-3.25)，为文本赋予了较小的正分(1)。

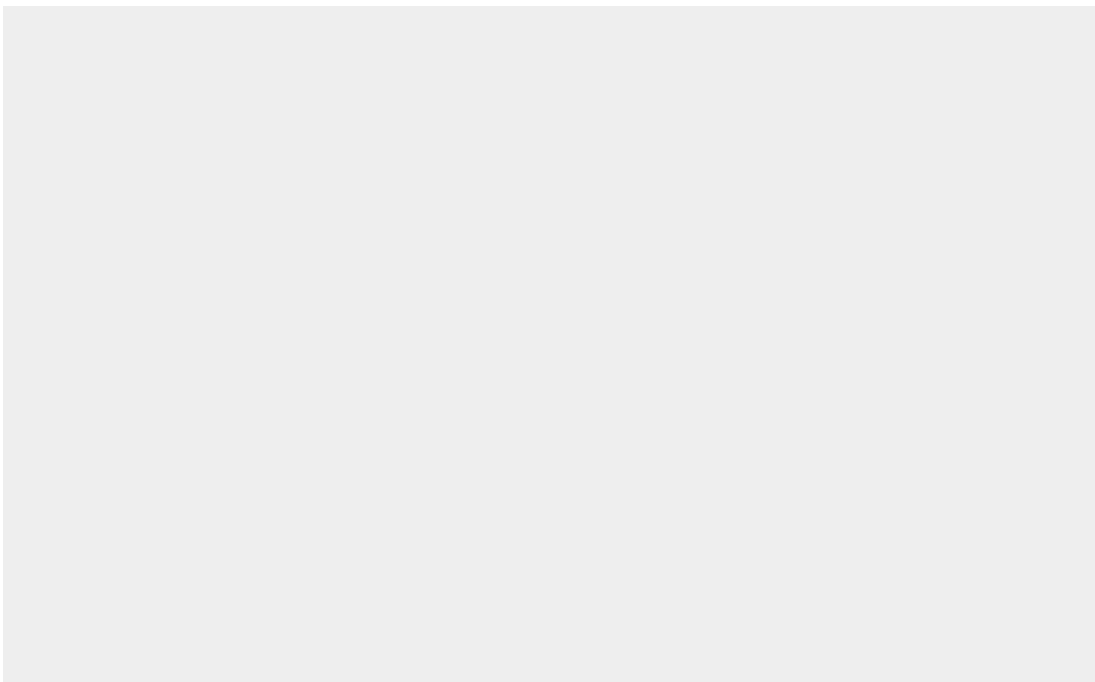
如何从分数序列中找出总和最大的子序列可以用动态规划很好地解决，这里就不给出详细算法，有兴趣可以参考《[Extracting Article Text from the Web with Maximum Subsequence Segmentation](#)》这篇论文，MSS算法的效果并不好，但本文认为它可以代表早期的很多算法。

- **MSS算法（朴素贝叶斯）**

MSS还有其他的版本，我们上面说算法给标签和文本分别赋予-3.25和1分，这是固定值，还有一个版本的MSS（也在论文中）利用朴素贝叶斯的方法为标签和文本计算分数。虽然这个版本的MSS效果有一定的提升，但仍不理想。

- **利用聚类的方法**

无监督学习在第一类算法中也起到重要作用。很多算法利用聚类的方法，将网页的正文和非正文自动分为2类。例如在《[CETR - Content Extraction via Tag Ratios](#)》算法中，网页被切分为多行文本，算法为每行文本计算2个特征，分别是下图中的横轴和纵轴，红色椭圆中的单元（行），大多数是网页正文，而绿色椭圆中包含的单元（行），大多数是非正文，使用k-means等聚类方法，就可以很好地将正文和非正文分为两类，然后再设计一些启发式算法，即可区分两类中哪一类是正文，哪一类是非正文。



聚类

- 使用**DOM**树的**Node**作为特征计算的基本单元

早期的算法往往将token序列、字符序列作为计算特征的单元，从某种意义上来说，这破坏了网页的结构，也没有充分利用网页的特征。在后来的算法中，很多使用**DOM**树的**Node**作为特征计算的基本单元，例如《[Web news extraction via path ratios](#)》、《[Dom based content extraction via text density](#)》，这些算法仍然是利用启发式规则和无监督学习，由于使用**DOM**树的**Node**作为特征计算的基本单元，使得算法可以获取到更好、更多的特征，因此可以设计更好的启发式规则和无监督学习算法，这些算法在抽取效果上，往往远高于前面所述的算法。由于在抽取时使用**DOM**树的**Node**作为单元，算法也可以较容易地保留正文的结构（主要是为了保持网页中正文的排版）。

我们在WebCollector(1.12版本开始)中，实现了一种第一类算法，可以到[官网](#)直接下载源码使用。

2. 基于分类器的网页抽取算法（第二类机器学习抽取算法）

实现基于分类器的网页抽取算法（第二类算法），大致流程如下：

- 找几千个网页作为训练集，对网页的正文和非正文（即需要抽取和不需要抽取的部分）进行人工标注。
- 设计特征。例如一些算法将**DOM**树的标签类型(div,p,body等)作为特征之一（当然这是一个不推荐使用的特征）。
- 选择合适的分类器，利用特征进行训练。

对于网页抽取，特征的设计是第一位的，具体使用什么分类器有时候并不是那么重要。在使用相同特征的情况下，使用决策树、**SVM**、神经网络等不同的分类器不一定对抽取效果造成太大的影响。

从工程的角度来说，流程中的第一步和第二步都是较为困难的。训练集的选择也很有讲究，要保证在选取的数据集中网页结构的多样性。例如现在比较流行的正文结构为：

```
<div>
  <p>xxxx</p>
  <p>xxxxxxxxxx</p>
  <span>xxx</span>
  <p>xxxxx</p>
  <p>xxxx</p>
</div>
```

2.1 eager learning

基于分类器的网页抽取算法，算法通过训练集产生了模型（如决策树模型、神经网络模型等）

如果训练集中只有五六个网站的页面，很有可能这些网站的正文都是上面这种结构，而恰好在特征设计中，有两个特征是：

- 节点标签类型(div,p,body等)
- 孩子节点标签类型频数(即孩子节点中，div有几个，p有几个...)

假设使用决策树作为分类器，最后的训练出的模型很可能是：

如果一个节点的标签类型为div，且其孩子节点中标签为p的节点超过3个，则这个节点对应网页的正文。

虽然这个模型在训练数据集上可以达到较好的抽取效果，但显而易见，有很多网站不满足这个规则。因此训练集的选择，对抽取算法的效果有很大的影响。

网页设计的风格一致在变，早期的网页往往利用表格(table)构建整个网页的框架，现在的网页喜欢用div构建网页的框架。如果希望抽取算法能够覆盖较长的时间段，在特征设计时，就要尽量选用那些不易变化的特征。标签类型是一个很容易变化的特征，随着网页设计风格的变化而变化，因此前面提到，非常不建议使用标签类型作为训练特征。

2.2 lazy learning

事先不通过训练集产生模型的算法，比较有名的KNN就是属于lazy learning。

一些抽取算法借助KNN来选择抽取算法，可能听起来有些绕，这里解释一下。假设有2种抽取算法A、B，有3个网站site1,site2,site3。2种算法在3个网站上的抽取效果（这里用0%-100%的一个数表示，越大说明越好）如下：

| 网站 | A算法抽取效果 | B算法抽取效果 |
|----|---------|---------|
| | | |

| | | |
|-------|-----|-----|
| site1 | 90% | 70% |
| site2 | 80% | 85% |
| site3 | 60% | 87% |

可以看出来，在site1上，A算法的抽取效果比B好，在site2和site3上，B算法的抽取效果较好。在实际中，这种情况很常见。所以有些人就希望设计一个分类器，这个分类器不是用来分类正文和非正文，而是用来帮助选择抽取算法。例如在这个例子中，分类器在我们对site1中网页进行抽取时，应该告诉我们使用A算法可以获得更好的效果。

举个形象的例子，A算法在政府类网站上抽取效果较好，B算法在互联网新闻网站上抽取效果较好。那么当我对政府类网站进行抽取时，分类器应该帮我选择A算法。

这个分类器的实现，可以借助KNN算法。事先需要准备一个数据集，数据集中有多个站点的网页，同时需要维护一张表，表中告诉我们在每个站点上，不同抽取算法的抽取效果（实际上只要知道在每个站点上，哪个算法抽取效果最好即可）。当遇到一个待抽取的网页，我们将网页和数据集中所有网页对比（效率很低），找出最相似的K个网页，然后看着K个网页中，哪个站点的网页最多（例如k=7,其中有6个网页都是来自CSDN新闻），那么我们就选择这个站点上效果最好的算法，对这个未知网页进行抽取。

3. 基于网页模板自动生成的网页抽取算法

基于网页模板自动生成的网页抽取算法（第三类算法）有很多种。这里例举一种。在《[URL Tree: Efficient Unsupervised Content Extraction from Streams of Web Documents](#)》中，用多个相同结构页面（通过URL判断）的对比，找出其中异同，页面间的共性的部分是非正文，页面间差别较大的部分有可能是正文。这个很好理解，例如在一些网站中，所有的网页页脚都相同，都是备案信息或者版权申明之类的，这是页面之间的共性，因此算法认为这部分是非正文。而不同网页的正文往往是不同的，因此算法识别出正文页较容易。这种算法往往并不是针对单个网页作正文抽取，而是收集大量同构网页后，对多个网页同时进行抽取。也就是说，并不是输入一个网页就可以实时进行抽取。

小礼物走一走，来简书关注我

赞赏支持



泊牧

写了 20154 字，被 15 人关注，获得了 11 个喜欢

已关注

喜欢 | 4



更多分享

被以下专题收入，发现更多相似内容

收入我的专题

机器学习(Machine Learning)&深度学习(Deep Learning)资料(Ch...

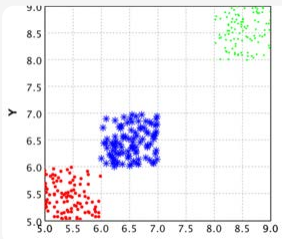
机器学习(Machine Learning)&深度学习(Deep Learning)资料(Chapter 1) 注:机器学习资料篇目一共500条,篇目二开始更新 希望转载的朋友, 你可以不用联系我. 但是一定要保留原文链接, 因为这个项目还在继续也...

Albert陈凯

面向开发人员的机器学习指南

首页 资讯 文章 资源 小组 相亲 登录 注册 首页 最新文章 IT 职场 前端 后端 移动端 数据库 运维 其他技术 - 导航条 - 首页最新文章IT 职场前端- JavaScript-...

Helen_Cat



机器学习与深度学习资料

《Brief History of Machine Learning》 介绍:这是一篇介绍机器学习历史的文章, 介绍很全面, 从感知机、神经网络、决策树、SVM、Adaboost到随机森林、Deep Learning. 《Deep Learning in Neural Netw...

JasonDing

掘金 Android 文章精选合集

用两张图告诉你, 为什么你的 App 会卡顿? - Android - 掘金Cover 有什么料? 从这篇文章中你能获得这些

料：知道setContentView()之后发生了什么？ ... Android 获取 View 宽高的常用正确方式，避免为零 - 掘金...



掘金官方

掘金 Java 文章精选合集

Java 基础思维导图，让 Java 不再难懂 - 工具资源 - 掘金思维导图的好处 最近看了一些文章的思维导图，发现思维导图真是个强大的工具。了解了思维导图的作用之后，觉得把它运用到java上应该是个不错的想法...



掘金官方

精英分享：专升本学习时的小问题你及时改正了吗？

距离2018年安徽专升本还有半年时间，在专升本复习过程当中，很多时候你认为是对的学习方法，可能思路不一定正确，甚至会误导你，从而浪费更多的复习时间，精英专升本整理了几个复习易犯的错误给同学们...



安徽精英专升本学校

扶贫打假，既要看统计也要问生计

统计数据显示，西部某深度贫困县自来水入户率达到了85%以上，然而一位领导在该县下乡调研时发现，供水设施虽然接到了群众家里，但是他走访中拧了几十个水龙头，却只有一个出水，原因是山区村落缺乏二...



这个_aef4

寺中一日

是谁举起双手，打出第一片碎屑 是谁殚精竭虑，让佛拈花微笑，俯视众生 又是谁开凿山道，引领众生的脚步 大殿巍峨，香烛燃烧，梵音唱起 我沿街而上时...



半虹骑士



何必如此追求完美

在如今的社会中，人们学会了享受，反而缺失了初心，曾经的我们满足于简单的快乐，如今的我们更急于追求虚有的快乐，家家都有本难念的经，不要觉得老天不公，不公平的事多了老天才懒的理你，你的付出会...



愈丹

我们疯过爱过憧憬过的——印尼

说起来和印尼的相遇还是上个暑假了，照片还躺在手机里，时不时还能翻到。彩色的天空和飞驰的机车，破旧的小村和红白两色的国旗——记不住太多细节...



赛赛赛琳娜



