[python学习] 简单爬取维基百科程序语言消息盒

文章主要讲述如何通过Python爬取维基百科的消息盒(Infobox)，主要是通过正则表达式和urllib实现；后面的文章可能会讲述通过BeautifulSoup实现爬取网页知识。由于这方面的文章还是较少，希望提供一些思想和方法对大家有所帮助。如果有错误或不足之处，欢迎之处；如果你只想知道该篇文章最终代码，建议直接阅读第5部分及运行截图。

一. 维基百科和Infobox

你可能会疑惑Infobox究竟是个什么东西呢？下面简单介绍。

维基百科作为目前规模最大和增长最快的开放式的在线百科系统,其典型包括两个网状结构:文章网络和分类树(以树为主体的图)。该篇博客主要是对维基百科“程序语言”结构进行分析，下载网页后提取相关消息盒(Infobox)中属性和对应的值。

Infobox是模板(一系列的信息框)，通常是成对的标签label和数据data组成。参考：http://zh.wikipedia.org/wiki/Template:Infobox

下图是维基百科“世界政区索引”中“中国”的部分Infobox信息和“程序设计语言列表”的“ACL2”语言的消息盒。

自然地理（实际管辖区）	
面积	
• 国土面积：9,634,057 [注 1]平方公里（世界第3/4名）	
• 水域率：2.8%	
首都	北京市
中央政府所在地	北京市西城区西长安街街道中南海
最大行政区	新疆维吾尔自治区
最大城市	重庆市[1]（按照行政区域内人口计算）
地理最高点	珠穆朗玛峰
最长河流	长江
最大湖泊	青海湖
海岸线	32000公里[2]
时区	北京时间，UTC+8

ACL2	
编程范型	函数式编程，元编程
发行时间	1990（内部发布） 1996（公开发布）
设计者	Robert S. Boyer J Strother Moore Matt Kaufmann
实作者	Matt Kaufmann J Strother Moore
最新发行时间	6.4 / 2014年1月9日，13个月前
型态系统	动态类型
启发语言	Nqthm, Common Lisp
操作系统	跨平台
许可证	BSD License
网站	ACL2

二. 爬虫实现

1. python下载html网页

首先需要访问维基百科的“程序设计语言列表”，并简单讲述如何下载静态网页的代码。在维基百科中输入如下URL可以获取所有程序语言列表：

http://zh.wikipedia.org/wiki/程序设计语言列表

你可以看到从A到Z的各种程序语言，如A# .NET、ActionScript、C++、易语言等，当然可能其中很多语言都没有完善或没有消息盒Infobox。同样如果你想查询世界各个国家的信息，输入URL如下：

http://zh.wikipedia.org/wiki/世界政区索引

通过如下代码可以获取静态的html页面：

```
# coding=utf-8
import urllib
import time
import re

#第一步 获取维基百科内容
#http://zh.wikipedia.org/wiki/程序设计语言列表
keyname="程序设计语言列表"
temp='http://zh.wikipedia.org/wiki/'+str(keyname)
content = urllib.urlopen(temp).read()
open('wikipedia.html','w+').write(content)
print 'Start Crawling pages!!!'
```

获取的本地wikipedia.html界面如下图所示：

2. 正则表达式获取URL超链接

现在需要通过Python正则表达式获取所有语言的超链接URL。

网页中创建超链接需要使用A标记符,结束标记符为</A>.它的最基本属性是href,用于指定超链接的目标,通过href属性指定不同的值,可以创建不同类型的超链接.

```
href = '<p><a href="www.csdn.cn" title="csdn">CSDN</a></p>'
link = re.findall(r"(?<=href=\").+?(?=\")|(?<=href=\').+?(?=\')", href)
print link
```

上面是获取网页URL的正则表达式代码，输出结果是：['www.csdn.cn']。

但是获取“程序设计语言列表”中所有语言时，我是通过人工确定起始位置“0-9”和结束位置“参看”进行查找的，代码如下：

```
# coding=utf-8
import urllib
import time
import re

#第一步 获取维基百科内容
#http://zh.wikipedia.org/wiki/程序设计语言列表
keyname="程序设计语言列表"
```

```
temp='http://zh.wikipedia.org/wiki/'+str(keyname)
content = urllib.urlopen(temp).read()
open('wikipedia.html','w+').write(content)
print 'Start Crawling pages!!!'

#第二步 获取网页中的所有URL
#从原文中"0-9"到"参看"之间是A-Z各个语言的URL
start=content.find(r'0-9')
end=content.find(r'参看')
cutcontent=content[start:end]
link_list = re.findall(r"(?<=href=\").+?(?=\")|(?<=href=\').+?(?=\')", cutcontent)
fileurl=open('test.txt','w')
for url in link_list:
    print url
```

输出的结果HTML源码主要包括以下几种形式：

```
<a href="#A">A</a>
<a href="/wiki/C%EF%BC%83" title="C#" class="mw-redirect">C#</a>
<a href="/w/index.php?title=..." class="new" title="A Sharp (.NET)（页面不存在）">A# .NET</a>
输出：
#A
/wiki/C%EF%BC%83
/w/index.php?title=A%2B%2B&amp;action=edit&amp;redlink=1.
```

此时获取了href中URL，很显然“http://zh.wikipedia.org”加上获取的后缀就是具体的一门语言信息，如：
http://zh.wikipedia.org/wiki/C#
http://zh.wikipedia.org/wiki/A_Sharp_(.NET)
它会转换成C%EF%..等形式。而index.php?此种形式表示该页面维基百科未完善，相应的Infobox消息盒也是不存在的。下面就是去到每一个具体的URL获取里面的title信息，同时下载相应的静态URL。

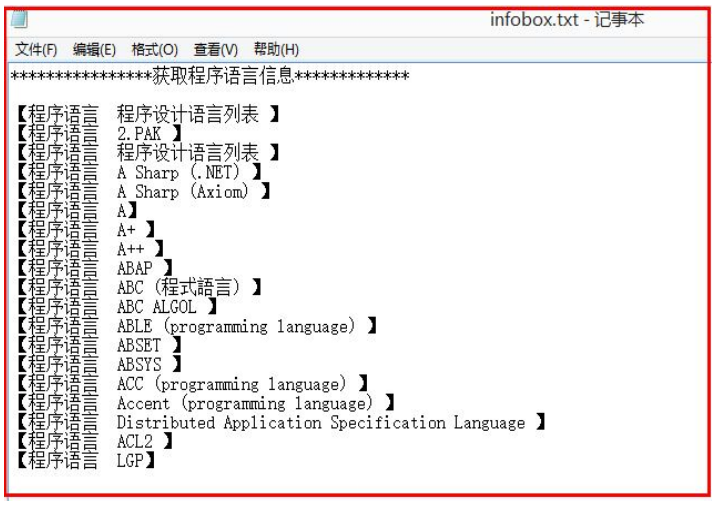
3. 获取程序语言title信息及下载html

首先通过拼接成完整的URL，在通过open函数下载对应的程序语言html源码至language文件夹下；再通过正则表达式r'(?<=<title>).*?(?=</title>)'可以获取网页的title信息。代码如下：

```
# coding=utf-8
import urllib
import time
import re

#第一步 获取维基百科内容
#http://zh.wikipedia.org/wiki/程序设计语言列表
keyname="程序设计语言列表"
temp='http://zh.wikipedia.org/wiki/'+str(keyname)
content = urllib.urlopen(temp).read()
open('wikipedia.html','w+').write(content)
print 'Start Crawling pages!!!'

#第二步 获取网页中的所有URL
#从原文中"0-9"到"参看"之间是A-Z各个语言的URL
start=content.find(r'0-9')
end=content.find(r'参看')
cutcontent=content[start:end]
link_list = re.findall(r"(?<=href=\").+?(?=\")|(?<=href=\').+?(?=\')", cutcontent)
fileurl=open('test.txt','w')
for url in link_list:
```

```
    #字符串包含wiki或/w/index.php则正确url 否则A-Z
 if url.find('wiki')>=0 or url.find('index.php')>=0:
        #print url
        num=num+1
fileurl.close()
print 'URL Successed! ',num,' urls.'

#第三步 下载每个程序URL静态文件并获取Infobox对应table信息
#国家：http://zh.wikipedia.org/wiki/阿布哈兹
#语言：http://zh.wikipedia.org/wiki/ActionScript
info=open('infobox.txt','w')
info.write('****************获取程序语言信息*************\n\n')
j=1
for url in link_list:
    if url.find('wiki')>=0 or url.find('index.php')>=0:
        #下载静态html
        wikiurl='http://zh.wikipedia.org'+str(url)
        print wikiurl
        language = urllib.urlopen(wikiurl).read()
        name=str(j)+' language.html'
        #注意 需要创建一个country的文件夹 否则总报错No such file or directory
        open(r'language/'+name,'w+').write(language) #写方式打开+没有即创建
        #获取title信息
        title_pat=r'(?<=<title>).*?(?=</title>)'
        title_ex=re.compile(title_pat,re.M|re.S)
        title_obj=re.search(title_ex, language) #language对应当前语言HTML所有内容
        title=title_obj.group()
        #获取内容'C语言 - 维基百科，自由的百科全书' 仅获取语言名
        middle=title.find(r'-')
        info.write('【程序语言  '+title[:middle]+'】\n')
        print title[:middle]

        #设置下载数量
        j=j+1
        time.sleep(1)
        if j==20:
            break;
    else:
        print 'Error url!!!'
else:
    print 'Download over!!!'
        fileurl.write(url+'\n')
```

输出结果如下图所示，其中获取20个程序语言URL的标题输入infobox.txt如下：



然后是获取每门语言HTML下载至本地的language文件夹下，需要自己创建一个文件夹。其中一门语言代码如下，标题就是下图左上方的ACL2：

4. 爬取class=Infobox的table信息

获取Infobox的table信息，通过分析源代码发现“程序设计语言列表”的消息盒如下：

<table class="infobox vevent" ..><tr><th></th><td></td></tr></table>

而“世界政区索引”的消息盒形式如下：

<table class="infobox"><tr><td></td></tr></table>

具体的代码如下所示：

```
# coding=utf-8
import urllib
import time
import re

#第一步 获取维基百科内容
#http://zh.wikipedia.org/wiki/程序设计语言列表
keyname="程序设计语言列表"
temp='http://zh.wikipedia.org/wiki/'+str(keyname)
content = urllib.urlopen(temp).read()
open('wikipedia.html','w+').write(content)
print 'Start Crawling pages!!!'

#第二步 获取网页中的所有URL
#从原文中"0-9"到"参看"之间是A-Z各个语言的URL
start=content.find(r'0-9')
end=content.find(r'参看')
cutcontent=content[start:end]
link_list = re.findall(r"(?<=href=\").+?(?=\")|(?<=href=\').+?(?=\')", cutcontent)
fileurl=open('test.txt','w')
for url in link_list:
    #字符串包含wiki或/w/index.php则正确url 否则A-Z
    if url.find('wiki')>=0 or url.find('index.php')>=0:
        fileurl.write(url+'\n')
        #print url
        num=num+1
fileurl.close()
print 'URL Successed! ',num,' urls.'

```

```
#第三步 下载每个程序URL静态文件并获取Infobox对应table信息#国家：http://zh.wikipedia.org/wiki/阿布哈兹
#语言：http://zh.wikipedia.org/wiki/ActionScript
info=open('infobox.txt','w')
info.write('*****************获取程序语言信息*************\n\n')
j=1
for url in link_list:
    if url.find('wiki')>=0 or url.find('index.php')>=0:
        #下载静态html
        wikiurl='http://zh.wikipedia.org'+str(url)
        print wikiurl
        language = urllib.urlopen(wikiurl).read()
        name=str(j)+' language.html'
        #注意 需要创建一个country的文件夹 否则总报错No such file or directory
        open(r'language/'+name,'w+').write(language) #写方式打开+没有即创建
        #获取title信息
        title_pat=r'(?<=<title>).*?(?=</title>)'
        title_ex=re.compile(title_pat,re.M|re.S)
        title_obj=re.search(title_ex, language) #language对应当前语言HTML所有内容
        title=title_obj.group()
        #获取内容'C语言 - 维基百科，自由的百科全书' 仅获取语言名
        middle=title.find(r'-')
        info.write('【程序语言  '+title[:middle]+'】\n')
        print title[:middle]

        #第四步 获取Infobox的内容
        #标准方法是通过<table>匹配</table>确认其内容，找与它最近的一个结束符号
        #但此处分析源码后取巧<p><b>实现
        start=language.find(r'<table class="infobox vevent"') #起点记录查询位置
        end=language.find(r'<p><b>'+title[:middle-1])    #减去1个空格
        infobox=language[start:end]
        print infobox


        #设置下载数量
        j=j+1
        time.sleep(1)
        if j==20:
            break;
    else:
        print 'Error url!!!'
else:
    print 'Download over!!!'
```

“print infobox”输出其中一门语言ActionScript的InfoBox消息盒部分源代码如下：

```
<table class="infobox vevent" cellspacing="3" style="border-spacing:3px;width:22em;text-align:left;font-size:small;line-height:1.5em;">
<caption class="summary"><b>ActionScript</b></caption>
<tr>
<th scope="row" style="text-align:left;white-space:nowrap;;;">发行时间</th>
<td style=";;">1998年</td>
</tr>
<tr>
<th scope="row" style="text-align:left;white-space:nowrap;;;">实现者</th>

<td class="organiser" style=";;"><a href="/wiki/Adobe_Systems" title="Adobe Systems">Adobe Systems</a></td>
</tr>
<tr>
<tr>
<th scope="row" style="text-align:left;white-space:nowrap;;;">启发语言</th>

<td style=";;"><a href="/wiki/JavaScript" title="JavaScript">JavaScript</a>、<a href="/wiki/Java"
```

```
 title="Java">Java</a></td>
15 </tr> 16 </table>
```

5. 爬取消息盒属性-属性值

爬取格式如下：

```
<table>
    <tr>
        <th>属性</th>
        <td></td>
    </tr>
</table>
```

其中th表示加粗处理，td和th中可能存在属性如title、id、type等值；同时<td></td>之间的内容可能存在或或
等值，都需要处理。下面先讲解正则表达式获取td值的例子：

参考：http://bbs.csdn.net/topics/390353859?page=1

```
1 <table>
2 <tr>
3 <td>序列号</td><td>DEIN3-39CD3-2093J3</td>
4 <td>日期</td><td>2013年1月22日</td>
5 <td>售价</td><td>392.70 元</td>
6 <td>说明</td><td>仅限5用户使用</td>
7 </tr>
8 </table>
```

Python代码如下：

```
1  s = '''<table>
2  ....
3  </table>''' #对应上面HTML
4  res = r'<td>(.*?)</td><td>(.*?)</td>'
5  m = re.findall(res,s,re.S|re.M)
6  for line in m:
7      print unicode(line[0],'utf-8'),unicode(line[1],'utf-8') #unicode防止乱码
8
9  #输出结果如下:
10 #序列号 DEIN3-39CD3-2093J3
11 #日期 2013年1月22日
12 #售价 392.70 元
13 #说明 仅限5用户使用
```

如果<td id="">包含该属性则正则表达式为r'<td id=.*?>(.*?)</td>'；同样如果不一定是id属性开头，则可以使用正则表达式r'<td .*?>(.*?)</td>'。

最终代码如下：

```
1  # coding=utf-8
2  import urllib
3  import time
4  import re
5
6  #第一步 获取维基百科内容
7  #http://zh.wikipedia.org/wiki/程序设计语言列表
8  keyname="程序设计语言列表"
9  temp='http://zh.wikipedia.org/wiki/'+str(keyname)
10 content = urllib.urlopen(temp).read()
```

```
open('wikipedia.html','w+').write(content) 12 print 'Start Crawling pages!!!'

#第二步 获取网页中的所有URL
#从原文中"0-9"到"参看"之间是A-Z各个语言的URL
start=content.find(r'0-9')
end=content.find(r'参看')
cutcontent=content[start:end]
link_list = re.findall(r"(?<=href=\").+?(?=\")|(?<=href=\').+?(?=\')", cutcontent)
fileurl=open('test.txt','w')
for url in link_list:
    #字符串包含wiki或/w/index.php则正确url 否则A-Z
    if url.find('wiki')>=0 or url.find('index.php')>=0:
        fileurl.write(url+'\n')
        #print url
        num=num+1
fileurl.close()
print 'URL Successed! ',num,' urls.'

#第三步 下载每个程序URL静态文件并获取Infobox对应table信息
#国家：http://zh.wikipedia.org/wiki/阿布哈兹
#语言：http://zh.wikipedia.org/wiki/ActionScript
info=open('infobox.txt','w')
info.write('*****************获取程序语言信息*************\n\n')
j=1
for url in link_list:
    if url.find('wiki')>=0 or url.find('index.php')>=0:
        #下载静态html
        wikiurl='http://zh.wikipedia.org'+str(url)
        print wikiurl
        language = urllib.urlopen(wikiurl).read()
        name=str(j)+' language.html'
        #注意 需要创建一个country的文件夹 否则总报错No such file or directory
        open(r'language/'+name,'w+').write(language) #写方式打开+没有即创建
        #获取title信息
        title_pat=r'(?<=<title>).*?(?=</title>)'
        title_ex=re.compile(title_pat,re.M|re.S)
        title_obj=re.search(title_ex, language) #language对应当前语言HTML所有内容
        title=title_obj.group()
        #获取内容'C语言 - 维基百科，自由的百科全书' 仅获取语言名
        middle=title.find(r'-')
        info.write('【程序语言  '+title[:middle]+'】\n')
        print title[:middle]

        #第四步 获取Infobox的内容
        #标准方法是通过<table>匹配</table>确认其内容，找与它最近的一个结束符号
        #但此处分析源码后取巧<p><b>实现
        start=language.find(r'<table class="infobox vevent"') #起点记录查询位置
        end=language.find(r'<p><b>'+title[:middle-1])    #减去1个空格
        infobox=language[start:end]
        #print infobox

        #第五步 获取table中属性-属性值
        if "infobox vevent" in language: #防止无Infobox输出多余换行
            #获取table中tr值
            res_tr = r'<tr>(.*?)</tr>'
            m_tr =  re.findall(res_tr,infobox,re.S|re.M)
            for line in m_tr:
                #print unicode(line,'utf-8')

                #获取表格第一列th 属性
                res_th = r'<th scope=.*?>(.*?)</th>'
```

```
                m_th = re.findall(res_th,line,re.S|re.M)
                for mm in m_th:
                    #如果获取加粗的th中含超链接则处理
                    if "href" in mm:
                        restr = r'<a href=.*?>(.*?)</a>'
                        h = re.findall(restr,mm,re.S|re.M)
                        print unicode(h[0],'utf-8')
                        info.write(h[0]+'\n')
                    else:
                        #报错用str()不行 针对两个类型相同的变量
                        #TypeError: coercing to Unicode: need string or buffer, list found
                        print unicode(mm,'utf-8') #unicode防止乱
                        info.write(mm+'\n')

                #获取表格第二列td 属性值
                res_td = r'<td .*?>(.*?)</td>'
                m_td = re.findall(res_td,line,re.S|re.M)
                for nn in m_td:
                    if "href" in nn:
                        #处理超链接<a href=../rel=..></a>
                        res_value = r'<a .*?>(.*?)</a>'

                        m_value = re.findall(res_value,nn,re.S|re.M) #m_td会出现TypeError: expected string or
buffer
                        for value in m_value:
                            print unicode(value,'utf-8'),
                            info.write(value+' ')
                        print ' ' #换行
                        info.write('\n')
                    else:
                        print unicode(nn,'utf-8')
                        info.write(nn+'\n')
            print '\n'
            info.write('\n\n')
        else:
            print 'No Infobox\n'
            info.write('No Infobox\n\n\n')

        #设置下载数量
        j=j+1
        time.sleep(1)
        if j==40:
            break;
    else:
        print 'Error url!!!'
else:
    print 'Download over!!!'
```

输出结果是自定义爬取多少门语言，其中Ada编程语言如下图所示：

最初我采用的是如下方法，维基百科需要中文繁体，需要人工标注分析HTML再爬取两个尖括号(>...<)之间的内容：

```
#启发语言
start=infobox.find(r'啟發語言')
end=infobox.find(r'</tr>',start)
print infobox[start:end]
info.write(infobox[start:end]+'\n')
```

当然代码中还存在很多小问题，比如爬取的信息中含<a href>超链接时只能爬取含超链接的信息，而没有超链接的信息被忽略了；如何删除或
等信息。但是我希望自己能提供一种爬取网页知识的方法给大家分享，后面可能会讲述如何通过Python实现BeautifulSoup爬取网页知识以及如何爬取图片，很多时候我们在野网站浏览图片都需要不断点击下一张。

（By：Eastmount 2015-3-18 深夜4点 http://blog.csdn.net/eastmount/）