

爬虫实战【3】Python-如何将html转化为pdf(PdfKit)

前言

前面我们对博客园的文章进行了爬取，结果比较令人满意，可以一下子下载某个博主的所有文章了。但是，我们获取的只有文章中的文本内容，并且是没有排版的，看起来也比较费劲。。。咋办？一个比较好的方法是将文章的正文内容转化成pdf，就不要考虑排版的事情了，看起来比较美观，也不会丢失一些关键信息。

python中将html转化为pdf的常用工具是Wkhtmltopdf工具包，在python环境下，pdfkit是这个工具包的封装类。如何使用pdfkit以及如何配置呢？分如下几个步骤。

1、下载wkhtmltopdf安装包，并且安装到电脑上，在系统Path变量中添加wkhtmltopdf的bin路径，以便于pdfkit的调用。

下载地址：<https://wkhtmltopdf.org/downloads.html>

请根据自己的系统版本，选择合适的安装包。如果没有装C语言库，建议选择Windows下的第二种。

【插入图片 pdf1】

Flavor	Version	Downloads	Comments
Windows (MSVC)	0.12.4	32-bit / 64-bit	for Windows Vista/2008 or later; bundles VC++ Runtime 2015
Windows (MinGW)	0.12.4	32-bit / 64-bit	for Windows XP/2003 or later; standalone
Linux	0.12.4	32-bit / 64-bit	depends on: zlib, fontconfig, freetype, X11 libs (libX11, libXext)
OS X	0.12.4	32-bit / 64-bit	(has regression) for OS X 10.6 or later
Others	0.12.4	source code	read INSTALL.md for compilation instructions

2、在pycharm中安装pdfkit库，过程就不介绍啦，前面讲过类似的内容。

3、在pycharm中安装whtmltopdf库。

这个和第一步中的安装包是两个东西，请区别开来。

用法简介

对于简单的任务来说，代码很easy，比如：

```
import pdfkit
pdfkit.from_url('http://baidu.com', 'out.pdf')
pdfkit.from_file('test.html', 'out.pdf')
pdfkit.from_string('Hello!', 'out.pdf')
```

pdfkit包含的方法很少，主要用的就是这三个，我们简单看一下每个函数的API：

from_ulr()

```
def from_url(url, output_path, options=None, toc=None, cover=None,
            configuration=None, cover_first=False):
    """
    Convert file or files from URLs to PDF document

    :param url: url可以是某一个url也可以是url的列表,
    :param output_path: 输出pdf的路径, 如果设置为False意味着返回一个string

    Returns: True on success
    """

    r = PDFKit(url, 'url', options=options, toc=toc, cover=cover,
               configuration=configuration, cover_first=cover_first)

    return r.to_pdf(output_path)
```

from_file()

```
def from_file(input, output_path, options=None, toc=None, cover=None, css=None,
              configuration=None, cover_first=False):
    """
    Convert HTML file or files to PDF document

    :param input: 输入的内容可以是一个html文件, 或者一个路径的list, 或者一个类文件对象
    :param output_path: 输出pdf的路径, 如果设置为False意味着返回一个string

    Returns: True on success
    """

    r = PDFKit(input, 'file', options=options, toc=toc, cover=cover, css=css,
               configuration=configuration, cover_first=cover_first)

    return r.to_pdf(output_path)
```

from_string()

```
def from_string(input, output_path, options=None, toc=None, cover=None, css=None,
                configuration=None, cover_first=False):
    #类似的, 这里就不介绍了
    r = PDFKit(input, 'string', options=options, toc=toc, cover=cover, css=css,
               configuration=configuration, cover_first=cover_first)
    return r.to_pdf(output_path)
```

举几个栗子

我们可以传入列表:

```
pdfkit.from_url(['google.com', 'yandex.ru', 'engadget.com'], 'out.pdf')
pdfkit.from_file(['file1.html', 'file2.html'], 'out.pdf')
```

我们可以将一个打开的文件对象传进去:

```
with open('file.html') as f:
    pdfkit.from_file(f, 'out.pdf')
```

如果我们想继续操作pdf，可以将其读取成一个变量，其实就是一个string变量。

```
# Use False instead of output path to save pdf to a variable
pdf = pdfkit.from_url('http://google.com', False)
```

指定pdf的格式

我们可以指定各种选项，就是上面三个方法中的options。

具体的设置可以参考<https://wkhtmltopdf.org/usage/wkhtmltopdf.txt> 里面的内容。

我们这里只举个栗子：

```
options = {
    'page-size': 'Letter',
    'margin-top': '0.75in',
    'margin-right': '0.75in',
    'margin-bottom': '0.75in',
    'margin-left': '0.75in',
    'encoding': "UTF-8",
    'custom-header' : [
        ('Accept-Encoding', 'gzip')
    ]
    'cookie': [
        ('cookie-name1', 'cookie-value1'),
        ('cookie-name2', 'cookie-value2'),
    ],
    'no-outline': None
}

pdfkit.from_url('http://google.com', 'out.pdf', options=options)
```

默认的，pdfkit会show出所有的output，如果你不想使用，可以设置为quite：

```
options = {
    'quiet': ''
}

pdfkit.from_url('google.com', 'out.pdf', options=options)
```

我们还可以传入任何html标签，比如：

```
body = """
<html>
  <head>
    <meta name="pdfkit-page-size" content="Legal"/>
    <meta name="pdfkit-orientation" content="Landscape"/>
  </head>
  Hello World!
</html>
"""

pdfkit.from_string(body, 'out.pdf') #with --page-size=Legal and --
orientation=Landscape
```

改进


有了上面的知识之后，我们大可以尝试一下，如果将之前的**save_file**方法做一些改变，就能够实现我们下载PDF的目标啦。

我们将方法名改成**save_to_pdf**，并且在**get_body**方法中直接返回**str(div)**，而不是**div.text**。代码如下：

```
def save_to_pdf(url):
    '''
    根据url，将文章保存到本地
    :param url:
    :return:
    '''
    title=get_title(url)
    body=get_Body(url)
    filename=author+'-'+title+'.pdf'
    if '/' in filename:
        filename=filename.replace('/', '+')
    if '\\' in filename:
        filename=filename.replace('\\', '+')
    print(filename)
    options = {
        'page-size': 'Letter',
        'encoding': "UTF-8",
        'custom-header': [
            ('Accept-Encoding', 'gzip')
        ]
    }
    #本来直接调用pdfkid的from方法就可以了，但是由于我们的wkhtmltopdf安装包有点问题，一直没法搜
    到，所以只能用本办法，直接配置了wk的地址
    #尴尬了，主要是一直没法下载到最新的wk，只能在网上down了旧版本的。有谁能下到的话发我一份。。。
    config=pdfkit.configuration(wkhtmltopdf=r'C:\Program
Files\wkhtmltopdf\bin\wkhtmltopdf.exe')
    pdfkit.from_string(body, filename, options=options, configuration=config)
    print('打印成功!')
```

【插入图片，pdf2】



 注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)， [访问](#) 网站首页。

【福利】个推四大热门移动开发SDK全部免费用一年，限时抢！

昵称: xingzhui
园龄: 3年3个月
粉丝: 28
关注: 4
[+加关注](#)

2019年10月						
日	一	二	三	四	五	六
29	30	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31	1	2
3	4	5	6	7	8	9

搜索

找找看

谷歌搜索

常用链接

[我的随笔](#)

[我的评论](#)

[我的参与](#)

[最新评论](#)

[我的标签](#)

我的标签

[Python\(29\)](#)

[爬虫\(24\)](#)

[Java基础\(13\)](#)

JSP基础(9)

C#基础(6)

Selenium(4)

AJAX(3)

MongoDB(3)

PyQuery(2)

PySpider(2)

更多

随笔档案

2019年7月(1)

2017年12月(12)

2017年11月(17)

2017年8月(1)

2017年3月(1)

2016年12月(4)

2016年11月(1)

2016年8月(14)

2016年7月(15)

相册

公众号(2)

最新评论

1. Re:爬虫实战【4】Python获取猫眼电影最受期待榜的50部电影

为了看你解决字体反爬，特意进来看的，裤子都脱了你就给我看这个

--Stubbron

2. Re:爬虫实战【1】使用python爬取博客园的某一篇文章

博主，看了您的文章，学习了很多。我也仿照您的程序实现了一下，但是保存在txt文档中没有格式，全是堆在一起的，请问，这个格式应该怎么处理呢？

--Andrew_qian

3. Re:Servlet3.0 jsp跳转到Servlet 出现404错误的路径设置方法

所以jsp文件没法找到servlet生成的class文件
应该是在——项目名/build/classes

--jueye

4. Re:Servlet3.0 jsp跳转到Servlet 出现404错误的路径设置方法

请问下，单独访问servlet也报404是什么问题

--programmer_yan

5. Re:C# 网络通信基础 总结

是一边学习一边总结的内容，希望能够坚持每天更新，记录自己的成长！

--xingzhui

阅读排行榜

1. DAO接口及实现类(15015)
2. 爬虫入门【1】urllib.request库用法简介(14972)
3. Java 在指定目录建立指定文件名的文件 并输入内容(13697)
4. CSS+DIV 设计一个简单的个人网页界面(13021)
5. Java ArrayList的使用方法(11504)

评论排行榜

1. Servlet3.0 jsp跳转到Servlet 出现404错误的路径设置方法(2)
2. 爬虫实战【4】Python获取猫眼电影最受期待榜的50部电影(1)
3. 爬虫实战【1】使用python爬取博客园的某一篇文章(1)
4. C# 网络通信基础 总结(1)

推荐排行榜

1. Java 在指定目录建立指定文件名的文件 并输入内容(2)
2. JSP基础——属性保存范围和request对象(1)
3. Servlet3.0 jsp跳转到Servlet 出现404错误的路径设置方法(1)
4. Java 设计一个贷款计算器 简易(1)
5. 【穿插】Python基础之文件、文件夹的创建，对上一期代码进行优化(1)

