

[文件](#) » 使用Firebug进行刮擦

使用Firebug进行抓取

❗ 注意

Google Directory（本指南中使用的示例网站）已不再可用，因为它已被Google关闭。但是，本指南中的概念仍然有效。如果您想更新本指南以使用新的（工作）网站，您的贡献将非常受欢迎！有关如何操作的信息，请参阅[Contributing to Scrapy](#)。

介绍

本文档介绍了如何使用[Firebug](#)（一个Firefox附加组件）来使抓取过程更轻松，更有趣。对于其他有用的Firefox附加组件，请参阅[有用的Firefox附加组件以进行抓取](#)。使用Firefox附加组件检查页面有一些注意事项，请参阅[使用检查实时浏览器DOM的注意事项](#)。





在此示例中，我们将展示如何使用[Firebug](#)从[Google Directory](#)中抓取数据，该[目录](#)包含与本[教程](#)中使用的[Open Directory Project](#)相同的数据，但具有不同的面。


Firebug附带了一个非常有用的功能，称为[Inspect Element](#)，它允许您通过将鼠标悬停在它们上来检查不同页面元素的HTML代码。否则，您必须通过HTML正文手动搜索标记，这可能是一项非常繁琐的任务。

在下面的屏幕截图中，您可以看到[Inspect Element](#)工具的运行情况。

Related Categories:

- [Business > Business Services > Consulting > Medical and Life Sciences](#) (321)
- [Kids and Teens > Health](#) (1160)
- [Recreation > Humor > Medical](#) (26)
- [Science > Social Sciences > Communication > Health Communication](#) (3)
- [Shopping > Health](#) (7391)
- [Society > Issues > Health](#) (2592)

Web Pages	Viewing in Google PageRank order	View
 WebMD - http://www.webmd.com/	A health resources for consumers, physicians, nurses, and educators. Includes news, chat forums, health quizzes and consumer product updates.	View
 Health On the Net Foundation - http://www.hon.ch/	Guides lay persons and non-medical users and medical practitioners to useful and reliable online medical and health information.	View
 BBC Health - http://www.bbc.co.uk/health/	Features current news plus archives, guides by subject, "Ask a Doctor" inquiry feature, a searchable conditions database, medical advice, and more.	View
 AOL Health - http://www.aolhealth.com/		View


Inspect
Edit
⌵
a
< font
< td
< tr
< tbody
< table
< form
< body
< html

Console

HTML

CSS

Script

DOM

Net

```

<table width="100%" cellspacing="0" cellpadding="1" border="0">
  <tbody>
    <tr valign="top">
      <td width="6%">
        <td>
          <font face="arial,sans-serif">
            <a href="http://www.webmd.com/">WebMD</a>
            <font size="-1" color="#6f6f6f">
              <br/>
              <font size="-1"> A health resources for consumers, physicians, nurses, and educators. Includes news, chat forums, health quizzes and consumer product updates. </font>
            </font>
          </td>
        </tr>
      </tbody>
    </table>
  
```

Done

正如所料，子类别包含指向其他子类别的链接，还包含指向实际网站的链接，这是目录的目的。

获取链接

通过查看类别URL，我们可以看到它们共享一种模式：

http://directory.google.com/Category/Subcategory/Another_Subcategory

一旦我们知道了，我们就可以构建一个正则表达式来跟随这些链接。例如，以下一个：

```
directory\.google\.com/[A-Z][a-zA-Z_/_]+&
```

因此，基于该正则表达式，我们可以创建第一个爬网规则：

```
Rule(LinkExtractor(allow='directory.google.com/[A-Z][a-zA-Z_/>+$', ),
      'parse_category',
      follow=True,
    ),
```

该 `Rule` 对象指示 `CrawlSpider` 基于蜘蛛如何遵循类别链接。 `parse_category` 将是蜘蛛的一种方法，它将处理和提取这些页面中的数据。

这就是蜘蛛到目前为止的样子：

```
from scrapy.linkextractors import LinkExtractor
from scrapy.spiders import CrawlSpider, Rule

class GoogleDirectorySpider(CrawlSpider):
    name = 'directory.google.com'
    allowed_domains = ['directory.google.com']
    start_urls = ['http://directory.google.com/']

    rules = (
        Rule(LinkExtractor(allow='directory\google\.com/[A-Z][a-zA-Z_/>+$', ),
              'parse_category', follow=True,
            ),
    )

    def parse_category(self, response):
        # write the category page data extraction code here
        pass
```

提取数据

现在我们要编写代码来从这些页面中提取数据。

在Firebug的帮助下，我们将查看包含指向网站链接的页面（例如 <http://directory.google.com/Top/Arts/Awards/>），并了解如何使用选择器提取这些链接。我们还将使用Scrapy shell来测试那些XPath，并确保它们按预期工作。

Web Pages	Viewing in Google PageRank order	View in alph
WebMD - http://www.webmd.com/ A health resources for consumers, physicians, nurses, and educators. Includes news, chat forums, health quizzes and consumer product		
Health On the Net Foundation - http://www.hon.ch/ Guides lay persons and non-medical users and medical practitioners to useful and reliable online medical and health information. Provide		
BBC Health - http://www.bbc.co.uk/health/ Features current news plus archives, guides by subject, "Ask a Doctor" inquiry feature, a searchable conditions database, message board		
AOL Health - http://www.aolhealth.com/ Find advice, information about diseases and drugs, fitness tips, and news items.		

Inspect Edit td <tr <tbody <table <form <body <html

Console HTML CSS Script DOM Net Options

```

<table width="100%" cellspacing="0" cellpadding="0" border="0">
  <table width="100%" cellspacing="0" cellpadding="1" border="0">
    <tbody>
      <tr valign="top">
        <td width="6%">
          <nobr>
            <a href="http://www.google.com/intl/en/dirhelp.html#pagerank">
          </nobr>
        </td>
      </tr>
    </tbody>
  </table>
</table>

```


Done

```

[<HtmlXPathSelector (a) xpath="//td[descendant::a[contains(@href, '#pagerank')]]/following-sibling::td//a>,
<HtmlXPathSelector (a) xpath="//td[descendant::a[contains(@href, '#pagerank')]]/following-sibling::td//a>,
<HtmlXPathSelector (a) xpath="//td[descendant::a[contains(@href, '#pagerank')]]/following-sibling::td//a>,
<HtmlXPathSelector (a) xpath="//td[descendant::a[contains(@href, '#pagerank')]]/following-sibling::td//a>,
<HtmlXPathSelector (a) xpath="//td[descendant::a[contains(@href, '#pagerank')]]/following-sibling::td//a>,
<HtmlXPathSelector (a) xpath="//td[descendant::a[contains(@href, '#pagerank')]]/following-sibling::td//a>]

In [5]: hxs.x('//td[descendant::a[contains(@href, '#pagerank')]]/following-sibling::td//a').extract()
Out[5]:
[u'<a href="http://www.webmd.com/">WebMD</a>',
 u'<a href="http://www.hon.ch/">Health On the Net Foundation</a>',
 u'<a href="http://www.bbc.co.uk/health/">BBC Health</a>',
 u'<a href="http://www.aolhealth.com/">AOL Health</a>',
 u'<a href="http://www.intelihealth.com/">InteliHealth</a>',
 u'<a href="http://www.judgehealth.org.uk/">Judge: Web Sites for Health</a>']

In [6]:

```

正如您所看到的，页面标记不是很具描述性：元素不包含 `id`，`class` 或者任何明确标识它们的属性，因此我们将使用排名栏作为参考点来选择我们构建时要提取的数据XPath的。

使用FireBug后，我们可以看到每个链接都在一个 `td` 标记内，该标记本身位于一个 `tr` 标记中，该标记还包含链接的排名栏（在另一个标记中 `td`）。

所以我们可以选择排名栏，然后找到它的父级（the `tr`），最后找到链接 `td`（包含我们要抓取的数据）。

这导致以下XPath：

```
//td[descendant::a[contains(@href, '#pagerank')]]/following-sibling::td//a
```

使用Scrapy shell测试这些复杂的XPath表达式并确保它们按预期工作非常重要。

基本上，该表达式将查找排名栏的

当然，这不是唯一的XPath，也许不是选择数据的简单方法。例如，另一种方法可能是找到任何font具有链接灰色的标签，

最后，我们可以编写我们的parse_category()方法：

```
def parse_category(self, response):
    # The path to website links in directory page
    links = response.xpath('//td[descendant::a[contains(@href, "#pagerank")]]/following-sibling::td/font')

    for link in links:
        item = DirectoryItem()
        item['name'] = link.xpath('a/text()').extract()
        item['url'] = link.xpath('a/@href').extract()
        item['description'] = link.xpath('font[2]/text()').extract()
        yield item
```

请注意，您可能会发现某些元素出现在Firebug中，但不会出现在原始HTML中，例如<tbody>元素的典型情况。

或者在页面上的其他HTML源代码可以在Firebug上检查实时DOM