[Day 13] 實戰：Scrapy爬PTT文章

2019鐵人賽

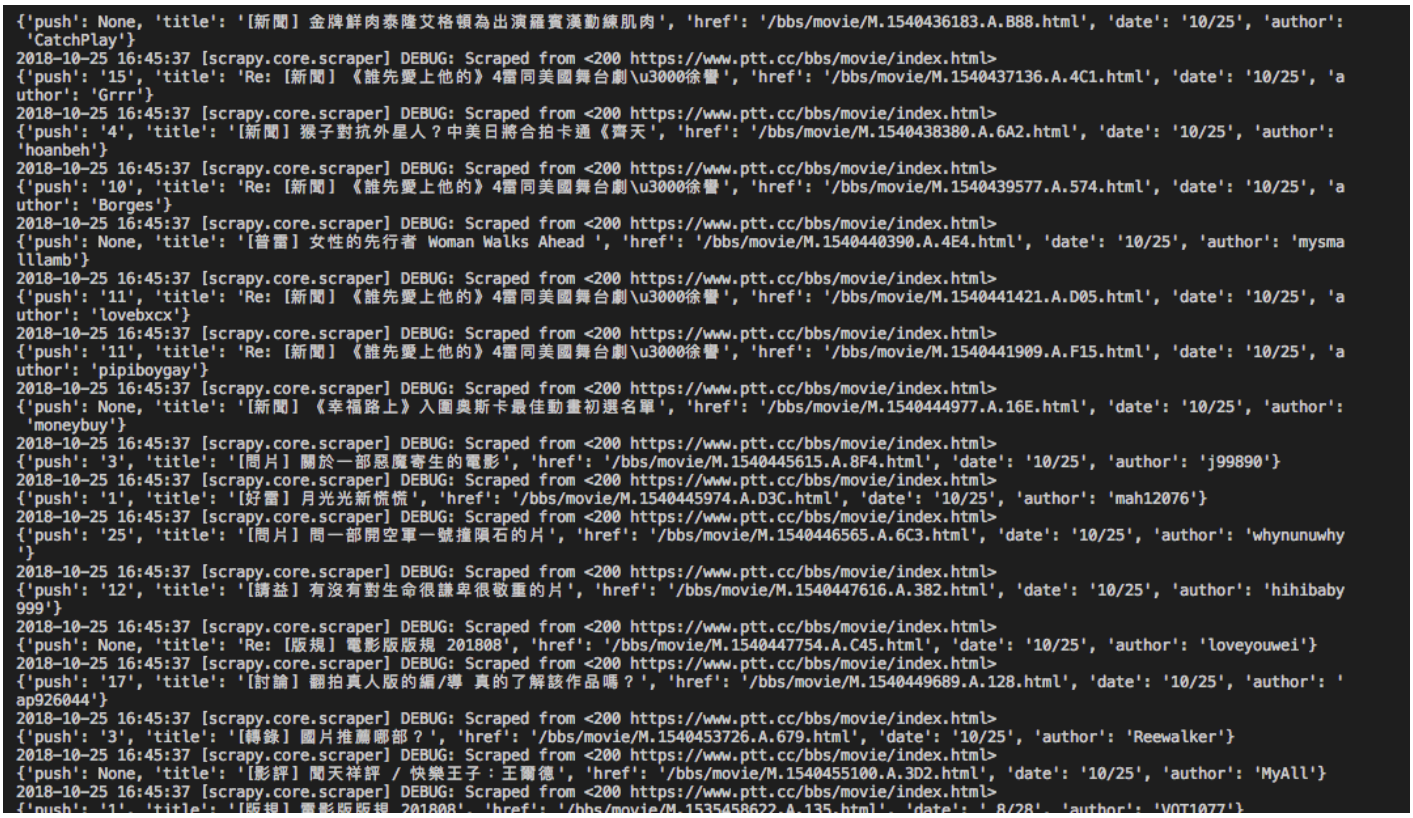
plusone
團隊 NUTC_imac
2018-10-28 07:09:34

早安，昨天我們介紹了 spider 的基本架構，今天會介紹 spider 實現ptt的爬蟲，透過 Scrapy 框架可以減少很多程式碼。
因為我們對於爬蟲的流程已經有稍微地瞭解了，我就直接貼上全部的程式碼了：

```
import scrapy
from scrapy.exceptions import CloseSpider

class PttSpider(scrapy.Spider):
    count_page = 1
    name = 'ptt'
    allowed_domains = ['www.ptt.cc/']
    start_urls = ['https://www.ptt.cc/bbs/movie/index.html']
    def parse(self, response):
        for q in response.css('div.r-ent'):
            item = {
                'push':q.css('div.nrec > span.hl::text').extract_first(),
                'title':q.css('div.title > a::text').extract_first(),
                'href':q.css('div.title > a::attr(href)').extract_first(),
                'date':q.css('div.meta > div.date ::text').extract_first(),
                'author':q.css('div.meta > div.author ::text').extract_first(),
            }
            yield(item)
        next_page_url = response.css('div.action-bar > div.btn-group > a.btn::a
ttr(href)')[3].extract()
        if (next_page_url) and (self.count_page < 10):
            self.count_page = self.count_page + 1
            new = response.urljoin(next_page_url)
        else:
            raise  CloseSpider('close it')
        yield scrapy.Request(new, callback = self.parse, dont_filter = True)
```

如下圖，可以看到我們順利的抓取到內容了！



```
{'push': None, 'title': '[新聞] 金牌鮮肉泰隆艾格頓為出演羅賓漢勤練肌肉', 'href': '/bbs/movie/M.1540436183.A.B88.html', 'date': '10/25', 'author':
 'CatchPlay'}
2018-10-25 16:45:37 [scrapy.core.scraper] DEBUG: Scraped from <200 https://www.ptt.cc/bbs/movie/index.html>
{'push': '15', 'title': 'Re: [新聞] 《誰先愛上他的》4雷同美國舞台劇\u3000徐譽', 'href': '/bbs/movie/M.1540437136.A.4C1.html', 'date': '10/25', 'a
uthor': 'Grrr'}
2018-10-25 16:45:37 [scrapy.core.scraper] DEBUG: Scraped from <200 https://www.ptt.cc/bbs/movie/index.html>
{'push': '4', 'title': '[新聞] 猴子對抗外星人？中美日將合拍卡通《齊天', 'href': '/bbs/movie/M.1540438380.A.6A2.html', 'date': '10/25', 'author':
'hoanbeh'}
2018-10-25 16:45:37 [scrapy.core.scraper] DEBUG: Scraped from <200 https://www.ptt.cc/bbs/movie/index.html>
{'push': '10', 'title': 'Re: [新聞] 《誰先愛上他的》4雷同美國舞台劇\u3000徐譽', 'href': '/bbs/movie/M.1540439577.A.574.html', 'date': '10/25', 'a
uthor': 'Borges'}
2018-10-25 16:45:37 [scrapy.core.scraper] DEBUG: Scraped from <200 https://www.ptt.cc/bbs/movie/index.html>
{'push': None, 'title': '[普雷] 女性的先行者 Woman Walks Ahead ', 'href': '/bbs/movie/M.1540440390.A.4E4.html', 'date': '10/25', 'author': 'mysma
lllamb'}
2018-10-25 16:45:37 [scrapy.core.scraper] DEBUG: Scraped from <200 https://www.ptt.cc/bbs/movie/index.html>
{'push': '11', 'title': 'Re: [新聞] 《誰先愛上他的》4雷同美國舞台劇\u3000徐譽', 'href': '/bbs/movie/M.1540441421.A.D05.html', 'date': '10/25', 'a
uthor': 'lovebxcx'}
2018-10-25 16:45:37 [scrapy.core.scraper] DEBUG: Scraped from <200 https://www.ptt.cc/bbs/movie/index.html>
{'push': '11', 'title': 'Re: [新聞] 《誰先愛上他的》4雷同美國舞台劇\u3000徐譽', 'href': '/bbs/movie/M.1540441909.A.F15.html', 'date': '10/25', 'a
uthor': 'pipiboygay'}
2018-10-25 16:45:37 [scrapy.core.scraper] DEBUG: Scraped from <200 https://www.ptt.cc/bbs/movie/index.html>
{'push': None, 'title': '[新聞] 《幸福路上》入圍奧斯卡最佳動畫初選名單', 'href': '/bbs/movie/M.1540444977.A.16E.html', 'date': '10/25', 'author':
 'moneybuy'}
2018-10-25 16:45:37 [scrapy.core.scraper] DEBUG: Scraped from <200 https://www.ptt.cc/bbs/movie/index.html>
{'push': '3', 'title': '[問片] 關於一部惡魔寄生的電影', 'href': '/bbs/movie/M.1540445615.A.8F4.html', 'date': '10/25', 'author': 'j99890'}
2018-10-25 16:45:37 [scrapy.core.scraper] DEBUG: Scraped from <200 https://www.ptt.cc/bbs/movie/index.html>
{'push': '1', 'title': '[好雷] 月光光新慌慌', 'href': '/bbs/movie/M.1540445974.A.D3C.html', 'date': '10/25', 'author': 'mah12076'}
2018-10-25 16:45:37 [scrapy.core.scraper] DEBUG: Scraped from <200 https://www.ptt.cc/bbs/movie/index.html>
{'push': '25', 'title': '[問片] 問一部開空軍一號撞隕石的片', 'href': '/bbs/movie/M.1540446565.A.6C3.html', 'date': '10/25', 'author': 'whynunuwhy
'}
2018-10-25 16:45:37 [scrapy.core.scraper] DEBUG: Scraped from <200 https://www.ptt.cc/bbs/movie/index.html>
{'push': '12', 'title': '[請益] 有沒有對生命很謙卑很敬重的片', 'href': '/bbs/movie/M.1540447616.A.382.html', 'date': '10/25', 'author': 'hihibaby
999'}
2018-10-25 16:45:37 [scrapy.core.scraper] DEBUG: Scraped from <200 https://www.ptt.cc/bbs/movie/index.html>
{'push': None, 'title': 'Re: [版規] 電影版版規 201808', 'href': '/bbs/movie/M.1540447754.A.C45.html', 'date': '10/25', 'author': 'loveyouwei'}
2018-10-25 16:45:37 [scrapy.core.scraper] DEBUG: Scraped from <200 https://www.ptt.cc/bbs/movie/index.html>
{'push': '17', 'title': '[討論] 翻拍真人版的編/導 真的了解該作品嗎？', 'href': '/bbs/movie/M.1540449689.A.128.html', 'date': '10/25', 'author': '
ap926044'}
2018-10-25 16:45:37 [scrapy.core.scraper] DEBUG: Scraped from <200 https://www.ptt.cc/bbs/movie/index.html>
{'push': '3', 'title': '[轉錄] 國片推薦哪部？', 'href': '/bbs/movie/M.1540453726.A.679.html', 'date': '10/25', 'author': 'Reewalker'}
2018-10-25 16:45:37 [scrapy.core.scraper] DEBUG: Scraped from <200 https://www.ptt.cc/bbs/movie/index.html>
{'push': None, 'title': '[影評] 聞天祥評 / 快樂王子：王爾德', 'href': '/bbs/movie/M.1540455100.A.3D2.html', 'date': '10/25', 'author': 'MyAll'}
2018-10-25 16:45:37 [scrapy.core.scraper] DEBUG: Scraped from <200 https://www.ptt.cc/bbs/movie/index.html>
{'push': '1', 'title': '[版規] 電影版版規 201808', 'href': '/bbs/movie/M.1535458622.A.135.html', 'date': ' 8/28', 'author': 'VOT1077'}
```

比起之前用 `requests` 與 `BeautifulSoup` 實作是不是方便又簡單很多？現在來說明上面的程式碼，我們總共要爬 標題，連結，日期，作者，推文數 等資訊欄位。

介紹程式碼：

- 爬取的標籤如何找這裡就不多做說明了，之後應該也會再找機會說明各種解析的方式。
- `CloseSpider` 這個方法用來關閉spider，我們放在if判斷內，若超過頁數了，就關閉 `spider` ，停止的時候也會看到 `[scrapy.core.engine] INFO: Closing spider (close it)` 的內容。舉例來說可以：

```
def parse_page(self, response):
    if 'Bandwidth exceeded' in response.body:
        raise CloseSpider('bandwidth_exceeded')
```

- `q.css()` 為 `css` 選擇器用來解析網頁資訊。
- `count_page` 用來計算總共要爬取幾頁內容。
- `urljoin(url)` 用於建置絕對url，當傳入的參數為一個相對位址href時，會根據 `response.url` 組合成對應的url。這裡用來產生 上一頁 的 `url` 給 `new` 變數。

- `scrapy.Request(new, callback = self.parse, dont_filter = True)` ，傳入參數 `new` 便為上一頁的url， `callback` 後為頁面解析函數， `Requests` 請求的頁面下載完後由該參數指定的頁面解析函數被呼叫。
- `yield Request()` ：用 `yield` 函數不會一次把所有值返回給你，會幫你在每次調用 `next()` 返回值， `Scrapy` 內部會處理 `next` ，所以若用 `return` 則會直接結束函數不會增加新的 `url` 。 `yield` 我理解為是一個迭代器，返回可執行的next函數，進行下一個 `url` 的爬取。

```
for q in response.css('div.r-ent'):
            item = {
                'push':q.css('div.nrec > span.hl::text').extract_first(),
                'title':q.css('div.title > a::text').extract_first(),
                'href':q.css('div.title > a::attr(href)').extract_first(),
                'date':q.css('div.meta > div.date ::text').extract_first(),
                'author':q.css('div.meta > div.author ::text').extract_first(),
            }
            yield(item)
```

可以看到上面的例子中，我們使用Python的 `dictionary` 方式存資料，不過這樣可能會有缺點， `dictionary` 雖然方便卻缺少結構性，容易打錯字或者回傳不一致的數據，特別是在多個 `Spider` 的專案中，所以明天我們會說明 `Item` 類別，用來封裝爬取到的資料，以及說明為什麼要用 `Item` ！

好的，那今天就到這啦！明天見！