

# 数据采集的另一种思路 - 浏览器脚本注入

作者: [maixiaojie](#)

原文链接: <https://ld246.com/article/1559269309863>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



昨天想去极客时间把购买的一个专栏里的数据扒下来，发现之前写的python脚本不能用了，原因是他们网站做了限流、也加了http时间戳的一些校验。我们可以将之前的python脚本进行改进，用ip代理来处理限流，寻找时间戳验证的规则就可以解决。

但是这次我们用了另外一种爬虫的思路，就是我们直接写一些js脚本，在对方的网站里运行，去请相应的接口，从而得到想要的数据库。

这种思路其实见过很多例子，之前有一个很火的，qq空间自动点赞的脚本，看过它的源码，其实很简单，就是直接去操作dom，然后触发一些事件。

另外一个很火的例子，github上很火的一个repo，fuckZhihu，据说是winter当年退知乎时写的，自己在知乎的数据保存下来。

下面是这次实践的内容：

## 获取文章id集合

刚进入专栏的时候会有一个获取左侧文章列表集合的请求，在这个接口里，我们就能获取到当前专栏所有请求。

```
/**
 * 运行在浏览器控制台中
 * @type {Array}
 */

// 要爬取的文章id集合 geek网站做了限流，所以分两次请求。当然你也可以用代理什么的
var ids = [77345, 77749, 77804, 78158, 78168, 78884, 79319, 79539, 80011, 80021, 80042, 80240, 80260, 80311, 81730, 82111, 82112, 82113, 82114, 82115, 82116, 82117, 82118, 82119, 82120, 82121, 82122, 82123, 82124, 82125, 82126, 82127, 82128, 82129, 82130, 82131, 82132, 82133, 82134, 82135, 82136, 82137, 82138, 82139, 82140, 82141, 82142, 82143, 82144, 82145, 82146, 82147, 82148, 82149, 82150, 82151, 82152, 82153, 82154, 82155, 82156, 82157, 82158, 82159, 82160, 82161, 82162, 82163, 82164, 82165, 82166, 82167, 82168, 82169, 82170, 82171, 82172, 82173, 82174, 82175, 82176, 82177, 82178, 82179, 82180, 82181, 82182, 82183, 82184, 82185, 82186, 82187, 82188, 82189, 82190, 82191, 82192, 82193, 82194, 82195, 82196, 82197, 82198, 82199, 82200, 82201, 82202, 82203, 82204, 82205, 82206, 82207, 82208, 82209, 82210, 82211, 82212, 82213, 82214, 82215, 82216, 82217, 82218, 82219, 82220, 82221, 82222, 82223, 82224, 82225, 82226, 82227, 82228, 82229, 82230, 82231, 82232, 82233, 82234, 82235, 82236, 82237, 82238, 82239, 82240, 82241, 82242, 82243, 82244, 82245, 82246, 82247, 82248, 82249, 82250, 82251, 82252, 82253, 82254, 82255, 82256, 82257, 82258, 82259, 82260, 82261, 82262, 82263, 82264, 82265, 82266, 82267, 82268, 82269, 82270, 82271, 82272, 82273, 82274, 82275, 82276, 82277, 82278, 82279, 82280, 82281, 82282, 82283, 82284, 82285, 82286, 82287, 82288, 82289, 82290, 82291, 82292, 82293, 82294, 82295, 82296, 82297, 82298, 82299, 82300, 82301, 82302, 82303, 82304, 82305, 82306, 82307, 82308, 82309, 82310, 82311, 82312, 82313, 82314, 82315, 82316, 82317, 82318, 82319, 82320, 82321, 82322, 82323, 82324, 82325, 82326, 82327, 82328, 82329, 82330, 82331, 82332, 82333, 82334, 82335, 82336, 82337, 82338, 82339, 82340, 82341, 82342, 82343, 82344, 82345, 82346, 82347, 82348, 82349, 82350, 82351, 82352, 82353, 82354, 82355, 82356, 82357, 82358, 82359, 82360, 82361, 82362, 82363, 82364, 82365, 82366, 82367, 82368, 82369, 82370, 82371, 82372, 82373, 82374, 82375, 82376, 82377, 82378, 82379, 82380, 82381, 82382, 82383, 82384, 82385, 82386, 82387, 82388, 82389, 82390, 82391, 82392, 82393, 82394, 82395, 82396, 82397, 82398, 82399, 82400, 82401, 82402, 82403, 82404, 82405, 82406, 82407, 82408, 82409, 82410, 82411, 82412, 82413, 82414, 82415, 82416, 82417, 82418, 82419, 82420, 82421, 82422, 82423, 82424, 82425, 82426, 82427, 82428, 82429, 82430, 82431, 82432, 82433, 82434, 82435, 82436, 82437, 82438, 82439, 82440, 82441, 82442, 82443, 82444, 82445, 82446, 82447, 82448, 82449, 82450, 82451, 82452, 82453, 82454, 82455, 82456, 82457, 82458, 82459, 82460, 82461, 82462, 82463, 82464, 82465, 82466, 82467, 82468, 82469, 82470, 82471, 82472, 82473, 82474, 82475, 82476, 82477, 82478, 82479, 82480, 82481, 82482, 82483, 82484, 82485, 82486, 82487, 82488, 82489, 82490, 82491, 82492, 82493, 82494, 82495, 82496, 82497, 82498, 82499, 82500, 82501, 82502, 82503, 82504, 82505, 82506, 82507, 82508, 82509, 82510, 82511, 82512, 82513, 82514, 82515, 82516, 82517, 82518, 82519, 82520, 82521, 82522, 82523, 82524, 82525, 82526, 82527, 82528, 82529, 82530, 82531, 82532, 82533, 82534, 82535, 82536, 82537, 82538, 82539, 82540, 82541, 82542, 82543, 82544, 82545, 82546, 82547, 82548, 82549, 82550, 82551, 82552, 82553, 82554, 82555, 82556, 82557, 82558, 82559, 82560, 82561, 82562, 82563, 82564, 82565, 82566, 82567, 82568, 82569, 82570, 82571, 82572, 82573, 82574, 82575, 82576, 82577, 82578, 82579, 82580, 82581, 82582, 82583, 82584, 82585, 82586, 82587, 82588, 82589, 82590, 82591, 82592, 82593, 82594, 82595, 82596, 82597, 82598, 82599, 82600, 82601, 82602, 82603, 82604, 82605, 82606, 82607, 82608, 82609, 82610, 82611, 82612, 82613, 82614, 82615, 82616, 82617, 82618, 82619, 82620, 82621, 82622, 82623, 82624, 82625, 82626, 82627, 82628, 82629, 82630, 82631, 82632, 82633, 82634, 82635, 82636, 82637, 82638, 82639, 82640, 82641, 82642, 82643, 82644, 82645, 82646, 82647, 82648, 82649, 82650, 82651, 82652, 82653, 82654, 82655, 82656, 82657, 82658, 82659, 82660, 82661, 82662, 82663, 82664, 82665, 82666, 82667, 82668, 82669, 82670, 82671, 82672, 82673, 82674, 82675, 82676, 82677, 82678, 82679, 82680, 82681, 82682, 82683, 82684, 82685, 82686, 82687, 82688, 82689, 82690, 82691, 82692, 82693, 82694, 82695, 82696, 82697, 82698, 82699, 82700, 82701, 82702, 82703, 82704, 82705, 82706, 82707, 82708, 82709, 82710, 82711, 82712, 82713, 82714, 82715, 82716, 82717, 82718, 82719, 82720, 82721, 82722, 82723, 82724, 82725, 82726, 82727, 82728, 82729, 82730, 82731, 82732, 82733, 82734, 82735, 82736, 82737, 82738, 82739, 82740, 82741, 82742, 82743, 82744, 82745, 82746, 82747, 82748, 82749, 82750, 82751, 82752, 82753, 82754, 82755, 82756, 82757, 82758, 82759, 82760, 82761, 82762, 82763, 82764, 82765, 82766, 82767, 82768, 82769, 82770, 82771, 82772, 82773, 82774, 82775, 82776, 82777, 82778, 82779, 82780, 82781, 82782, 82783, 82784, 82785, 82786, 82787, 82788, 82789, 82790, 82791, 82792, 82793, 82794, 82795, 82796, 82797, 82798, 82799, 82800, 82801, 82802, 82803, 82804, 82805, 82806, 82807, 82808, 82809, 82810, 82811, 82812, 82813, 82814, 82815, 82816, 82817, 82818, 82819, 82820, 82821, 82822, 82823, 82824, 82825, 82826, 82827, 82828, 82829, 82830, 82831, 82832, 82833, 82834, 82835, 82836, 82837, 82838, 82839, 82840, 82841, 82842, 82843, 82844, 82845, 82846, 82847, 82848, 82849, 82850, 82851, 82852, 82853, 82854, 82855, 82856, 82857, 82858, 82859, 82860, 82861, 82862, 82863, 82864, 82865, 82866, 82867, 82868, 82869, 82870, 82871, 82872, 82873, 82874, 82875, 82876, 82877, 82878, 82879, 82880, 82881, 82882, 82883, 82884, 82885, 82886, 82887, 82888, 82889, 82890, 82891, 82892, 82893, 82894, 82895, 82896, 82897, 82898, 82899, 82900, 82901, 82902, 82903, 82904, 82905, 82906, 82907, 82908, 82909, 82910, 82911, 82912, 82913, 82914, 82915, 82916, 82917, 82918, 82919, 82920, 82921, 82922, 82923, 82924, 82925, 82926, 82927, 82928, 82929, 82930, 82931, 82932, 82933, 82934, 82935, 82936, 82937, 82938, 82939, 82940, 82941, 82942, 82943, 82944, 82945, 82946, 82947, 82948, 82949, 82950, 82951, 82952, 82953, 82954, 82955, 82956, 82957, 82958, 82959, 82960, 82961, 82962, 82963, 82964, 82965, 82966, 82967, 82968, 82969, 82970, 82971, 82972, 82973, 82974, 82975, 82976, 82977, 82978, 82979, 82980, 82981, 82982, 82983, 82984, 82985, 82986, 82987, 82988, 82989, 82990, 82991, 82992, 82993, 82994, 82995, 82996, 82997, 82998, 82999, 83000, 83001, 83002, 83003, 83004, 83005, 83006, 83007, 83008, 83009, 83010, 83011, 83012, 83013, 83014, 83015, 83016, 83017, 83018, 83019, 83020, 83021, 83022, 83023, 83024, 83025, 83026, 83027, 83028, 83029, 83030, 83031, 83032, 83033, 83034, 83035, 83036, 83037, 83038, 83039, 83040, 83041, 83042, 83043, 83044, 83045, 83046, 83047, 83048, 83049, 83050, 83051, 83052, 83053, 83054, 83055, 83056, 83057, 83058, 83059, 83060, 83061, 83062, 83063, 83064, 83065, 83066, 83067, 83068, 83069, 83070, 83071, 83072, 83073, 83074, 83075, 83076, 83077, 83078, 83079, 83080, 83081, 83082, 83083, 83084, 83085, 83086, 83087, 83088, 83089, 83090, 83091, 83092, 83093, 83094, 83095, 83096, 83097, 83098, 83099, 83100, 83101, 83102, 83103, 83104, 83105, 83106, 83107, 83108, 83109, 83110, 83111, 83112, 83113, 83114, 83115, 83116, 83117, 83118, 83119, 83120, 83121, 83122, 83123, 83124, 83125, 83126, 83127, 83128, 83129, 83130, 83131, 83132, 83133, 83134, 83135, 83136, 83137, 83138, 83139, 83140, 83141, 83142, 83143, 83144, 83145, 83146, 83147, 83148, 83149, 83150, 83151, 83152, 83153, 83154, 83155, 83156, 83157, 83158, 83159, 83160, 83161, 83162, 83163, 83164, 83165, 83166, 83167, 83168, 83169, 83170, 83171, 83172, 83173, 83174, 83175, 83176, 83177, 83178, 83179, 83180, 83181, 83182, 83183, 83184, 83185, 83186, 83187, 83188, 83189, 83190, 83191, 83192, 83193, 83194, 83195, 83196, 83197, 83198, 83199, 83200, 83201, 83202, 83203, 83204, 83205, 83206, 83207, 83208, 83209, 83210, 83211, 83212, 83213, 83214, 83215, 83216, 83217, 83218, 83219, 83220, 83221, 83222, 83223, 83224, 83225, 83226, 83227, 83228, 83229, 83230, 83231, 83232, 83233, 83234, 83235, 83236, 83237, 83238, 83239, 83240, 83241, 83242, 83243, 83244, 83245, 83246, 83247, 83248, 83249, 83250, 83251, 83252, 83253, 83254, 83255, 83256, 83257, 83258, 83259, 83260, 83261, 83262, 83263, 83264, 83265, 83266, 83267, 83268, 83269, 83270, 83271, 83272, 83273, 83274, 83275, 83276, 83277, 83278, 83279, 83280, 83281, 83282, 83283, 83284, 83285, 83286, 83287, 83288, 83289, 83290, 83291, 83292, 83293, 83294, 83295, 83296, 83297, 83298, 83299, 83300, 83301, 83302, 83303, 83304, 83305, 83306, 83307, 83308, 83309, 83310, 83311, 83312, 83313, 83314, 83315, 83316, 83317, 83318, 83319, 83320, 83321, 83322, 83323, 83324, 83325, 83326, 83327, 83328, 83329, 83330, 83331, 83332, 83333, 83334, 83335, 83336, 83337, 83338, 83339, 83340, 83341, 83342, 83343, 83344, 83345, 83346, 83347, 83348, 83349, 83350, 83351, 83352, 83353, 83354, 83355, 83356, 83357, 83358, 83359, 83360, 83361, 83362, 83363, 83364, 83365, 83366, 83367, 83368, 83369, 83370, 83371, 83372, 83373, 83374, 83375, 83376, 83377, 83378, 83379, 83380, 83381, 83382, 83383, 83384, 83385, 83386, 83387, 83388, 83389, 83390, 83391, 83392, 83393, 83394, 83395, 83396, 83397, 83398, 83399, 83400, 83401, 83402, 83403, 83404, 83405, 83406, 83407, 83408, 83409, 83410, 83411, 83412, 83413, 83414, 83415, 83416, 83417, 83418, 83419, 83420, 83421, 83422, 83423, 83424, 83425, 83426, 83427, 83428, 83429, 83430, 83431, 83432, 83433, 83434, 83435, 83436, 83437, 83438, 83439, 83440, 83441, 83442, 83443, 83444, 83445, 83446, 83447, 83448, 83449, 83450, 83451, 83452, 83453, 83454, 83455, 83456, 83457, 83458, 83459, 83460, 83461, 83462, 83463, 83464, 83465, 83466, 83467, 83468, 83469, 83470, 83471, 83472, 83473, 83474, 83475, 83476, 83477, 83478, 83479, 83480, 83481, 83482, 83483, 83484, 83485, 83486, 83487, 83488, 83489, 83490, 83491, 83492, 83493, 83494, 83495, 83496, 83497, 83498, 83499, 83500, 83501, 83502, 83503, 83504, 83505, 83506, 83507, 83508, 83509, 83510, 83511, 83512, 83513, 83514, 83515, 83516, 83517, 83518, 83519, 83520, 83521, 83522, 83523, 83524, 83525, 83526, 83527, 83528, 83529, 83530, 83531, 83532, 83533, 83534, 83535, 83536, 83537, 83538, 83539, 83540, 83541, 83542, 83543, 83544, 83545, 83546, 83547, 83548, 83549, 83550, 83551, 83552, 83553, 83554, 83555, 83556, 83557, 83558, 83559, 83560, 83561, 83562, 83563, 83564, 83565, 83566, 83567, 83568, 83569, 83570, 83571, 83572, 83573, 83574, 83575, 83576, 83577, 83578, 83579, 83580, 83581, 83582, 83583, 83584, 83585, 83586, 83587, 83588, 83589, 83590, 83591, 83592, 83593, 83594, 83595, 83596, 83597, 83598, 83599, 83600, 83601, 83602, 83603, 83604, 83605, 83606, 83607, 83608, 83609, 83610, 83611, 83612, 83613, 83614, 83615, 83616, 83617, 83618, 83619, 83620, 83621, 83622, 83623, 83624, 83625, 83626, 83627, 83628, 83629, 83630, 83631, 83632, 83633, 83634, 83635, 83636, 83637, 83638, 83639, 83640, 83641, 83642, 83643, 83644, 83645, 83646, 83647, 83648, 83649, 83650, 83651, 83652, 83653, 83654, 83655, 83656, 83657, 83658, 83659, 83660, 83661, 83662, 83663, 83664, 83665, 83666, 83667, 83668, 83669, 83670, 83671, 83672, 83673, 83674, 83675, 83676, 83677, 83678, 83679, 83680, 83681, 83682, 83683, 83684, 83685, 83686, 83687, 83688, 83689, 83690, 83691, 83692, 83693, 83694, 83695, 83696, 83697, 83698, 83699, 83700, 83701, 83702, 83703, 83704, 83705, 83706, 83707, 83708, 83709, 83710, 83711, 83712, 83713, 83714, 83715, 83716, 83717, 83718, 83719, 83720, 83721, 83722, 83723, 83724, 83725, 83726, 83727, 83728, 83729, 83730, 83731, 83732, 83733, 83734, 83735, 83736, 83737, 83738, 83739, 83740, 83741, 83742, 83743, 83744, 83745, 83746, 83747, 83748, 83749, 83750, 83751, 83752, 83753, 83754, 83755, 83756, 83757, 83758, 83759, 83760, 83761, 83762, 83763, 83764, 83765, 83766, 83767, 83768, 83769, 83770, 83771, 83772, 83773, 83774, 83775, 83776, 83777, 83778, 83779, 83780, 83781, 83782, 83783, 83784, 83785, 8
```

```

/**
 * 初始化 文档中注入FileSaver.js 并执行请求开始的方法
 * @return {[type]} [description]
 */
function init() {
  var src = 'https://cdn.bootcss.com/FileSaver.js/2014-11-29/FileSaver.js';
  var script = document.createElement('script');
  script.src = src;
  var heads = document.getElementsByTagName("head");
  if (heads.length)
    heads[0].appendChild(script);
  else
    document.documentElement.appendChild(script);
  script.onload = function() {
    console.log('script loaded')
    start()
  }
}

```

我们在这里创建一个script标签，并将这个标签插入到文档中。

```

// 将数据转存为json文件
function downloadJson(data) {
  var blob = new Blob([JSON.stringify(data)], { type: "" });
  saveAs(blob, "data.json");
}

```

我在这里写了一个方法downloadJson，我们将等会获取到的数据传到这里来，就可以下载这个json文件了。

## 创建请求

创建ajax请求，请求文章详情的接口。

```

// 爬取一篇文章 就是一个ajax请求
function fetch(id) {
  var data = JSON.stringify({
    "include_neighbors": "false",
    "id": id
  });
  var xhr = new XMLHttpRequest();
  xhr.withCredentials = true;
  xhr.addEventListener("readystatechange", function() {
    if (this.readyState === 4) {
      var res = JSON.parse(this.responseText);
    }
  });
  xhr.open("POST", "https://time.geekbang.org/serv/v1/article");
  xhr.setRequestHeader("content-type", "application/json");
  xhr.send(data);
}

```

这里我们用原生的js来写的，是一个post请求，res就是我们得到这个接口的返回值，我们将需要的数从这个返回值中取出来就可以了。

上面说的是单个请求的实现。多个请求的实现如下图所示。

```
function start() {
  // 依次请求
  for (var i = 0; i < ids2.length; i++) {
    (function(i) {
      setInterval(fetch(ids2[i]), 3000)
    })(i)
  }
}
```

然后将数据保存一下：

```
var res = JSON.parse(this.responseText);
if (res.code == 0) {
  var data = res.data;
  var item = {
    id: data.id,
    pid: data.cid,
    article_content: data.article_content,
    article_cover: data.article_cover,
    article_ctime: data.article_ctime,
    article_title: data.article_title,
    audio_download_url: data.audio_download_url,
    audio_size: data.audio_size,
    audio_time: data.audio_time,
    audio_url: data.audio_url
  }
  rs.push(item);
}
```

所有的结果都放在rs这个数组中了。

## 下载数据

我们将所有数据放在了一个数组中，在最后一次请求结束的时候，执行我们写好的downloadJson方进行下载就可以了。

```
// 如果当前是最后一个就下载文件
if (id == ids2[ids2.length - 1]) {
  downloadJson(rs)
}
```

## 导入数据库

json文件导入数据库网上有很多的工具，我这次是用之前写好的脚本。

这个脚本在我的github上面，是用nodejs写的，地址：[tomysql.js](#)

## 最后

我们这次没用通用的做法，模拟请求，或者模拟浏览器，而是直接利用浏览器来采集数据，当然也要根据实际情况去选择用哪种做法。

完整的脚本：[geek.js](#)