

C# 网络编程之豆瓣OAuth2.0认证详解和遇到的各种问题及解决

原创 Eastmount 最后发布于2014-05-25 17:49:34 阅读数 6500 ☆ 收藏

展开



Python+TensorFlow人工智能

该专栏为人工智能入门专栏,采用Python3和TensorFlow实现人工智能相关算法。前期介绍安装流程、基础语法...



Eastmount

¥9.90

去订阅

最近在帮人弄一个豆瓣API应用,在豆瓣的OAuth2.0认证过程中遇到了各种问题,同时自己需要一个个的尝试与解决,最终完成了豆瓣API的访问.作者这里就不再吐槽豆瓣的认证文档了,毕竟人家也不容易.但是作者发现关于豆瓣OAuth认证过程的文章非常之少,所以想详细写这样一篇文章方便后面要做同样东西的人阅读.希望文章对大家有所帮助,尤其是想做豆瓣API开发的初学者.

(文章中蓝色字表示官方文档引用,红色字是可能遇到问题及注意,黑色字是作者叙述)

一.误区OAuth1.0认证过程

你遇到的第一个问题可能就是还在阅读"豆瓣 API OAuth认证"这篇官方文档,并且在疯狂的尝试.我就是这样花费了很长时间研究了OAuth1.0认证过程,但总是错误,因为他已经过时了.你需要使用OAuth2.0完成认证过程.阅读"使用OAuth2.0访问豆瓣API"官方文档.

但是OAuth1.0提供的思想还是可以的(安慰自己),而且很多关于豆瓣博客文档都是基于1.0的认证过程,其实已经过时.这里也简单讲讲它的基本过程和原理:(OAuth原理和过去Auth1.0认证成果案例)

它主要是通过Google-OAuth项目提供的C#语言的OAuth库,在自定义OAuth类中有计算签名值oauth_signature方法,签名方法HMAC-SHA1,还有计算时间戳oauth_timestamp等方法,然后在参照DoubanOAuthBasicSample项目完成它的认证流程,主要方法是getRequestToken、authorization和getAccessToken.

豆瓣OAuth1.0官方文档给出的具体四个步骤:获取未授权的Request Token、请求用户授权Request Token、使用授权后的Request Token换取Access Token、使用 Access Token 访问或修改受保护资源.具体豆瓣 OAuth1.0认证代码:OAuth类和DoubanOAuthBasicSample

其实,你只要知道它已经过时,不要在使用该方法即可.下面才是我想讲述的具体如何通过C#程序完成豆瓣的OAuth认证并访问需要授权的数据.

二.获取authorization_code

首先在你需要参考的具体豆瓣官方文档就是:<http://developers.douban.com/wiki/?title=oauth2>

使用OAuth2.0流程具体如下:(官方文档)

- 1.应用向豆瓣请求授权
- 2.豆瓣为用户显示一个授权页面,用户在此页面确认是否同意授权
- 3.如果用户同意授权,应用会获取到一个访问令牌(access_token),通过此令牌用户可以访问授权数据
- 4.如果访问需要授权的Api,请使用https协议,加上access_token的Header.

下面详细讲解

首先你需要申请API Key,在完成申请过程中你需要注意3个值:API Key\Secret\回调地址.后面的程序需要应用,当使用的时候我会详细介绍.由于我是桌面客户端应用,授权流程为native-application flow.

根据它的具体流程我设计的界面如下图所示:



获取authorization_code步骤(官方文档)

通过在浏览器中访问下面地址,来引导用户授权并获取authorization_code.

<https://www.douban.com/service/auth2/auth>

参数:

client_id 必选参数,应用的唯一标识,对应于APIKey

redirect_uri 必选参数,用户授权完成后的回调地址,通过回调地址获得用户的授权结果.与注册一致.

response_type 必选参数,此值可以为code或者token.在本流程中此值为code

scope 可选参数,申请权限的范围,如果不填,则使用缺省的scope.如果申请多个scope,使用逗号分隔

state 可选参数,用来维护请求和回调状态的附加字符串,授权完成回调时会附加此参数,应用根据此字符串来判断上下文关系.

注意:此请求必须是HTTP GET方式

返回结果:

当用户拒绝授权时,浏览器会重定向到redirect_uri,并附加错误信息

https://www.example.com/back?error=access_denied

当用户同意授权时,浏览器会重定向到redirect_uri, 并附加authorization_code

<https://www.example.com/back?code=9b73a4248>

首先,你需要定义变量如下,这些变量都是我们需要使用的.其中APIkey\secret\回调地址填写你自己的,下面的被我改过不是源数据 © (担心丢失)

```
#region 定义变量
// 申请的API Key
public string apiKey = "00489f145c2576bd00d9dd3d147064";
// 申请的API 密钥
public string apiKeySecret = "72c36131ace8ea";
// 申请的回调地址 我的应用URL
public string myurl = "http://www.baidu.com/";
// 获取authorization_code
public Uri GetAuthorizationCode = new Uri("https://www.douban.com/service/auth2/auth");
// 获取access_token
public Uri GetAccessToken = new Uri("https://www.douban.com/service/auth2/token");
// 浏览器返回authorization_code
public string authorizationCode = "";
// AccessToken授权成功后返回的Json数据
public string accessToken = "";
public string userName = "";
public string userId = "";
public string expiresIn = "";
public string refreshToken = "";
#endregion
```

然后,点击button1(浏览)按钮,同时设计视图中为webBrowser1控件添加DocumentCompleted事件(右键->属性->'闪电图标'事件).同时引用命名

空间:using System.Web;\using System.Net;\using System.IO;

```
#region 第一步 浏览 获取authorization_code
// 点击"浏览"显示豆瓣登录界面
private void button1_Click(object sender, EventArgs e)
{
    // 获取authorization_code构造URL
    StringBuilder url = new StringBuilder(GetAuthorizationCode.ToString()); // 可变字符串
    // 追加组合格式字符串
    url.AppendFormat("?client_id={0}&", apiKey);
    url.AppendFormat("redirect_uri={0}&", myurl);
    url.AppendFormat("response_type={0}", "code");
    // url.AppendFormat("scope={0}", "scope=shuo_basic_r,shuo_basic_w");
    // 显示输入URL
    textBox1.Text = url.ToString();
    string Input = url.ToString();
    // 将指定URL加载到WebBrowser控件中
    webBrowser1.Navigate(Input);
}

// 文件加载完成后发生事件
private void webBrowser1_DocumentCompleted(object sender, WebBrowserDocumentCompletedEventArgs e)
{
    // 获取当前文档URL
    string reUrl = this.webBrowser1.Url.ToString();
    textBox1.Text = reUrl;
    // 获取https://www.example.com/back?code=9b73a4248中code值
    if (!string.IsNullOrEmpty(reUrl))
    {
        // 获取问号后面字符串
        string LastUrl = reUrl.Substring(reUrl.LastIndexOf("?") + 1, (reUrl.Length - reUrl.LastIndexOf("?") - 1));
        // 根据参数间的&号获取参数数组 可以获取多个参数此时只需一个code
        string[] urlParam = LastUrl.Split('&');
        foreach (string s in urlParam)
        {
            // 将参数名与参数值赋值给数组 value[0] 参数名称 value[1] 参数值
            string[] value = s.Split('=');
            // MessageBox.Show("参数名称为:" + value[0] + " 参数值为:" + value[1]);
            if (value[0] == "code")
            {
                {
                    authorizationCode = value[1];
                }
            }
        }
    }
    // 输出获取code值
    if (authorizationCode != "") MessageBox.Show(authorizationCode);
}
#endregion
```

运行结果如下图所示,填写完用户邮箱和密码,点击授权后获取authorization_code如下:



这里你需要注意的有以下几点:

1.为什么我的回调地址设置为"http://www.baidu.com/"?

最初我设置的回调地址为自定义的一个网址,但是输入邮箱和密码后点击授权,总是HTTP 400错误,啥都不能获取.使用IE浏览器也是一样,但是google能获取.后来经过同学提醒(他做腾讯API,资料多,官方文档规范)可能回调地址需要能访问的网址,所以改成了百度,并且成功获取.因为这一步只需要获取code值,网址并不关键.(可能也有没考虑到地方)

2.你需要知道该程序中必须使用GET方法=浏览可以直接访问.(除设置header,后面讲述)

在程序中我是直接调用webBrowser1.Navigate(Input);函数访问的,也可以使用GET方法访问.GET和POST的区别是你需要知道的,简单来说就是GET后面给的网址参数显示,而POST是隐式的.

程序和文档中说道的采用GET的方法,并带有参数的都是这样连接的.url+? +参数=参数值&参数=参数值...正如获取code中的URL,它通过浏览器是可以直接访问的.

https://www.douban.com/service/auth2/auth?client_id=00489f145c2576bd00d9dd3d147064&redirect_uri=http://www.baidu.com/&response_type=code

3.你可能会遇到的错误是113 缺少参数required_parameter_is_missing.

此时你在设置url时,通过StringBuilder可变量字符串增加参数时,需要把所有的必须的参数填写.同时,注意你的&和?是否正确填写.你可以参考的官方文档错误报告如下:

<http://developers.douban.com/wiki/?title=oauth2>

http://developers.douban.com/wiki/?title=api_v2

此时,你的第一步获取code已经完成,接下来是获取access_token的过程.

三.获取access_token

官方文档接受的获取access_token如下:

<https://www.douban.com/service/auth2/token>

参数:

client_id 必选参数,应用的唯一标识,对应于APIKey

client_secret 必选参数,应用的唯一标识,对应于豆瓣secret

redirect_url 必选参数,用户授权后的回调地址

grant_type 必选参数,此值可为authorization_code或者refresh_token,此时为authorization_code

code 必选参数,上一步中获取的authorization_code

注意:此请求必须是HTTP POST方式

返回结果为Json格式数据如下图所示

```
// 获取的数据转换为Json格式 此时返回的json格式的数据
{
  "access_token": "0e63c03dfb66c4172b2b40b9f2344c",
  "douban_user_name": "Eastmount",
  "douban_user_id": "57279898",
  "expires_in": 604800,
  "refresh_token": "84406d40cc58e0ae8cc147c2650aa2",
}
```

其中你需要注意的是四个地方:

1.如何使用POST方法发送请求获取应答,这再也不是通过浏览器就能直接访问的问题.

2.获取的JSON格式数据如何通过C#解析.

3.可能你的访问不成功,你需要在"我的应用"中添加测试用户.看看自己的是否添加.

4.你可能遇到101 错误的请求方法,invalid_request_method: GET因为你需要采用POST方法.

我讲解如何解析JSON格式的数据,这里参照很多人可能都使用过的方法.

你需要下载Newtonsoft.Json.dll文件并添加引用它,参考百度"C#使用json字符串"方法,最好先把该dll放置到C:\WINDOWS\Microsoft.NET\Framework文件夹中,里面有很多dll(我的程序是VS2012 .NET4.5).

在"解决方案"中鼠标右键引用->添加引用->浏览->添加该dll.然后添加命名空间using Newtonsoft.Json.Linq;将POST方法获取应答string转换为JSON格式转换赋值即可.具体代码如下,同时POST方法发送HTTP请求我是通过自定义函数sendMessage实现的.加载button2(授权)按钮事件.

```
#region 第二步 授权 获取access_token
private void button2_Click(object sender, EventArgs e)
{
    // 显示输入URL
    textBox1.Text = GetAccessToken.ToString();
    // 获取access_token构造POST参数
    StringBuilder url = new StringBuilder(""); // 可变字符串
    // 追加组合格式字符串
    url.AppendFormat("client_id={0}&", apiKey);
    url.AppendFormat("client_secret={0}&", apiKeySecret);
    url.AppendFormat("redirect_uri={0}&", myurl);
    url.AppendFormat("grant_type={0}&", "authorization_code");
    url.AppendFormat("code={0}", authorizationCode);
    // 获取POST提交数据返回内容
    string AccessContent = sendMessage(GetAccessToken.ToString(), url.ToString());
    webBrowser1.DocumentText = AccessContent;
    // 获取的数据转换为Json格式 此时返回的json格式的数据
    JObject obj = JObject.Parse(AccessContent);
    accessToken = (string)obj["access_token"]; //access_token
    userName = (string)obj["douban_user_name"]; //豆瓣用户名
    userId = (string)obj["douban_user_id"]; //用户id
    expiresIn = (string)obj["expires_in"]; //生命周期 604800秒=7天
    refreshToken = (string)obj["refresh_token"]; //刷新令牌
    MessageBox.Show(accessToken + "\n" + userName + "\n" +
        userId + "\n" + expiresIn + "\n" + refreshToken);
}
```

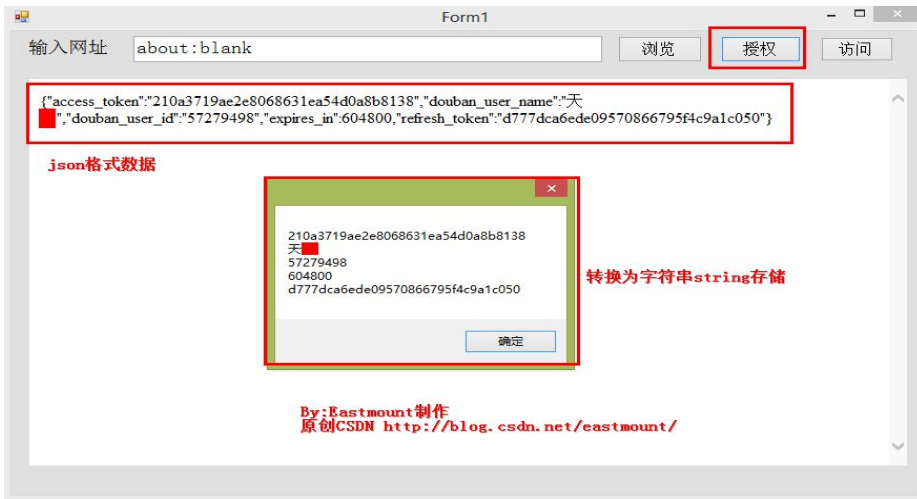
```

//发送消息Post方法
public static string sendMessage(string strUrl, string PostStr)
{
    try
    {
        //设置消息头
        CookieContainer objCookieContainer = null;
        HttpRequest request = (HttpRequest)WebRequest.Create(strUrl);
        request.Method = "Post";
        request.ContentType = "application/x-www-form-urlencoded";
        request.Referer = strUrl;
        if (objCookieContainer == null)
            objCookieContainer = new CookieContainer();

        request.CookieContainer = objCookieContainer;
        byte[] byteData = Encoding.UTF8.GetBytes(PostStr.ToString().TrimEnd('&'));
        request.ContentLength = byteData.Length;
        using (Stream reqStream = request.GetRequestStream())
        {
            reqStream.Write(byteData, 0, byteData.Length);
        }
        //Response应答流获取数据
        string strResponse = "";
        using (HttpWebResponse res = (HttpWebResponse)request.GetResponse())
        {
            objCookieContainer = request.CookieContainer;
            using (Stream resStream = res.GetResponseStream())
            {
                using (StreamReader sr = new StreamReader(resStream, Encoding.UTF8)) //UTF8
                {
                    strResponse = sr.ReadToEnd();
                }
            }
            // res.Close();
        }
        return strResponse;
    }
    catch (Exception e)
    {
        Console.WriteLine(e.ToString());
        Console.Read();
    }
    return null;
}
#endregion

```

运行程序,点击"浏览"授权成果后在点击"授权"的运行结果如下图所示:



四.使用access_token

通过前面的步骤我们已经获取了access_token数据,那么怎样使用它呢?当我看到豆瓣给出的使用官方文档如下时:

```
curl "https://api.douban.com/v2/user/~me"
-H "Authorization: Bearer a14afef0f66fcfcfce3e0fcd2e34f6ff4"
```

我很头痛啊!那么,怎样通过C#实现使用access_token访问要授权的数据呢?

GET https://api.douban.com/v2/user/~me

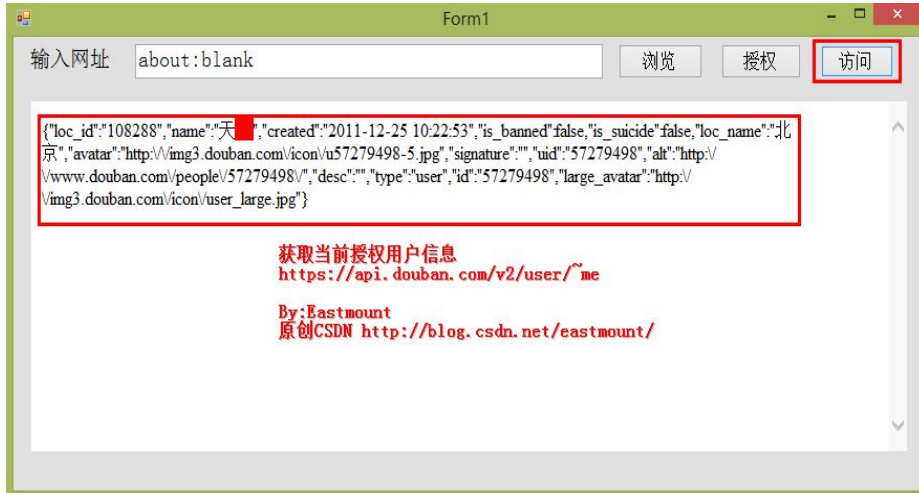
该URL是获取当前授权用户信息的,需要必须先进行API认证授权,返回的是当前授权用户信息.因此,使用它完全可以验证授权成果后的操作.具体代码如下:

```
#region 第三步 豆瓣访问授权数据
private void button3_Click(object sender, EventArgs e)
{
    try
    {
        // 获取当前授权用户信息 需要必须先进行API认证授权,返回当前授权的UserInfo
        string Input = "https://api.douban.com/v2/user/~me";
        textBox1.Text = Input;
        //HttpWebRequest对象实例:该类用于获取和操作HTTP请求
        var request = (HttpWebRequest)WebRequest.Create(Input); //Create:创建WebRequest对象
        //设置请求方法为GET
        request.Headers.Add("Authorization", "Bearer " + accessToken);
        request.Method = "GET";
        //HttpWebResponse对象实例:该类用于获取和操作HTTP应答
        var response = (HttpWebResponse)request.GetResponse(); //GetResponse:获取答复
        //构造数据流对象实例
        Stream stream = response.GetResponseStream(); //GetResponseStream:获取应答流
        StreamReader sr = new StreamReader(stream); //从字节流中读取字符
        //从流当前位置读取到末尾并显示在WebBrowser控件中
        string content = sr.ReadToEnd();
        webBrowser1.DocumentText = content;
        //关闭响应流
        stream.Close();
        sr.Close();
        response.Close();
    }
    catch (Exception msg) //异常处理
    {
        MessageBox.Show(msg.Message);
    }
}
```



```
} }  
#endregion
```

运行结果如下图所示:



不要以为这简单的几句代码就很容易实现了访问数据,其实你需要注意以下几点:

1.你可能会遇到1000错误 需要权限need_permission?

那时你需要在访问时设置header,添加-H "Authorization: Bearer a14afef0f66fcfce3e0fcd2e34f6ff4".采用C#设置标题头的具体代码如下:

```
request.Headers.Add("Authorization", "Bearer " + accessToken);
```

其中你需要注意Bearer后面的空格.而且有同学说他在使用网盘认证时,获取的方法有两种,一种是设置header,一种是在URL后面加上?access_token=值即可.但我测试了下豆瓣只有设置header可以.

2.认证过程已经详细叙述了,增删改查其他数据影片信息、用户信息、评论、收藏、关注等方法都类似,感兴趣的可以自己完成.它的具体设置参数参照豆瓣文档:http://developers.douban.com/wiki/?title=api_v2

五.总结

最后总结下它的具体步骤,其实它就是按照豆瓣的文档完成的,三个步骤:获取authorization_code、获取access_token和使用access_token.你也不能说豆瓣文档讲得不好,其实质东西它都讲述清楚了.需要的只是你自己的探索,可能刚接触比较新鲜和难,但其实完成后就发现它很简单.

希望该文章对大家有所帮助,尤其是想做豆瓣API开发的并且使用C#的,这方面资料比较少,基本都是php和java的.更希望同学能从该文章中学到一下几个东西:

- 1.如何通过官方文档完成一个东西,可能遇到的问题都需要自己解决,而不是复制粘贴.
- 2.如何使用C#网络编程GET和POST两种方法HTTP请求并获取应答.
- 3.如何使用C#解析Json格式的数据.
- 4.如何使用OAuth认证API开发,这方面腾讯、新浪等比较完善.

最后也感谢豆瓣网带给我很多知识,推荐程序员也看看文学书籍,如《文学回忆录》和《季羡林 清华园日记》,生活不只有编程啊.如果有错误或不足之处,还请海涵.

源代码免费下载地址:<http://download.csdn.net/detail/eastmount/7399075>

(By:Eastmount 2014-5-25 下午6点 原创CSDN<http://blog.csdn.net/eastmount/>)

👍 点赞 2 ☆ 收藏 📄 分享 ...



Eastmount 博客专家

发布了446 篇原创文章 · 获赞 6052 · 访问量 489万+

他的留言板

关注