

C# 网络编程之webBrowser乱码问题及解决知识

原创 Eastmount 最后发布于2013-09-23 21:15:45 阅读数 12373 ☆ 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

在使用PHP+MySQL编写网页时，曾近就因为显示中文乱码“口口口???”困扰我很长时间,没想到在C#制作浏览器或获取XML页面时也经常会遇到显示中文乱码的问题，可想而知怎样解决编码问题或统一编码问题是非常严重的问题。下面就讲讲我的一些理解及解决方法吧！

一.浏览器实现

前面我已经介绍了使用webBrowser控件实现“最简单的浏览器”基本代码如下所示：

```
// 命名空间
using System.Net;
using System.IO;
// 点击"浏览"按钮
private void button1_Click(object sender, EventArgs e)
{
    // 输入URL
    string url = textBox1.Text;
    var request = (HttpWebRequest)WebRequest.Create(url); //HTTP请求
    var response = (HttpWebResponse)request.GetResponse(); //HTTP应答
    // 显示webBrowser中
    Stream stream = response.GetResponseStream(); // 获取应答流
    StreamReader sr = new StreamReader(stream); // 从字节流中读取字符
    string content = sr.ReadToEnd();
    webBrowser1.DocumentText = content;
}
```

该方法通过获取相应URL的应答内容,通过赋值数据流,再从字节流中读取内容赋值给webBrowser控件中实现最简单的浏览器;但通过该方法常常会遇到现实中文字符乱码问题,或者是显示格式错误等问题.例如访问google等.



PS:这里有另外一种方法调用webBrowser的Navigate方法将指定位置的文档加载到控件中,其中一种重载方法Navigate(string)将制定的统一资源定位符URL处的文档加载到WebBrowser控件中替换上一个文档,而且实现该方法不会出现乱码问题、排版问题、缺少http报错问题.其实我很想知道封装的该函数是如何实现的.

```
private void button1_Click(object sender, EventArgs e)
{
    webBrowser1.Navigate(textBox1.Text.Trim());
}
```

二.乱码问题

通过获取网页的响应字符集string str = response.CharacterSet(只读属性)可以发现,当字符集为utf-8时才能正常显示,否则其他编码方式都会出现中文乱码;这里最常出现的乱码编码方式为ISO-8859-1,big5,gbk,gb2312等;而utf-8通常能显示中文.

ISO-8859-1:又称Latin-1或“西欧语言”,是单字节编码,自身不能显示中文,配合gbk或utf-8显示中文,通常以byte[]形式存储,以ISO-8859-1格式解码会是乱码,通常采用gb2313解码;

big5:通行于台湾、香港地区的一个繁体字编码方案,俗称“大五码”.上面访问香港google就是big5出现的乱码.

utf-8:是一种针对Unicode的可变长度字符编码,又称万国码.它可以用来表示Unicode标准中的任何字符,且其编码中的第一个字节仍与ASCII兼容,逐渐成为电子邮件、网页及其他存储或传送文字的应用中,优先采用的编码.

gb2312:是中华人民共和国国家汉字信息交换用编码,全称“信息交换用汉字编码字符集”,基本集共收入汉字6763个和非汉字图形字符682个.gbk亦汉字编码标准.

出现编码方式的根本原因是在解析时使用的字符编码和网页的编码方式不同,所以采用的解决方法通常是:

- 1.首先利用HttpWebResponse.CharacterSet属性获取字符集;
- 2.在根据不同的字符集设置相应的Encoding来避免乱码.

三.解决方法

其中最简单的方法是先获取其指定网页的字符集,在根据它的字符集采用相应的编码方式进行解码读取.我们采用下面代码获取该URL的字符集为"ISO-8859-1"

```
string str = response.CharacterSet;
        MessageBox.Show(str);
```

在设置其对应的编码方式,通过定义Encoding enc字符编码方式,其方法GetEncoding("相应编码方式")设置字符编码,然后在StreamReader(stream,enc)中采用对应设置的编码方式从字节流中读取内容.

```
private void button1_Click(object sender, EventArgs e)
{
    // 获取输入的URL
    string url = textBox1.Text;
    var request = (HttpWebRequest)WebRequest.Create(url); //HTTP请求
    var response = (HttpWebResponse)request.GetResponse(); //HTTP应答
    // 显示响应字符集
    string str = response.CharacterSet;
    MessageBox.Show(str);
    // 设置ISO-8859-1字符编码方式
    Encoding enc;
    if (response.CharacterSet != "ISO-8859-1")
    {
        enc = Encoding.GetEncoding(response.CharacterSet);
    }
    else
    {
        enc = Encoding.GetEncoding("GBK");
    }
    // 显示webBrowser中
    Stream stream = response.GetResponseStream(); // 获取应答流
    StreamReader sr = new StreamReader(stream, enc); // 从字节流中读取字符
    string content = sr.ReadToEnd();
    webBrowser1.DocumentText = content;
}
```

显示结果如下:其中CharacterSet采用ISO-8859-1编码方式,但从网页源代码中发现它的charset=gb2312所以我设置的Encoding.GetEncoding("GBK或GB2312").能正确显示中文汉字:



其实当获取指定网页字符集时,采用指定编码方式对其进行解码的核心代码就是几句:(同样可设置webBrowser.DocumentStream)

```
Stream stream = response.GetResponseStream();
StreamReader sr = new StreamReader(stream, System.Text.Encoding.GetEncoding("gb2312"));
string content = sr.ReadToEnd();
```

也可以采用获取到的文章内容content通过**byte[] utf8Bytes = System.Text.Encoding.Convert(iso_8859_1, utf_8, isoBytes)**;这样的语句转换为相应内容显示;经常能看到这样的通过byte[]转换ISO-8859-1的方法,但本人没有尝试过.个人认为在读取时就采用相对应的编码方式比较好.

由于webBrowser是简单的浏览器,肯定不能使用每一个页面都去找相应的characterSet字符集,因此我们可以设置相应的函数,直接调用函数实现显示内容:(代码感谢一位博主,<http://blog.csdn.net/lemonay/article/details/8865939>)

```
private static string GetHTMLbyWebRequest(string url)
{
    // 获取输入的URL
    var request = (HttpWebRequest)WebRequest.Create(url); //HTTP 请求
    var response = (HttpWebResponse)request.GetResponse(); //HTTP 应答
    Encoding encoding = System.Text.Encoding.Default; // 当前字符编码方式
    // 响应状态为OK
    if (response.StatusDescription.ToUpper()=="OK") // 大写
    {
        // 设置获取链接中网页的编码格式
        switch (response.CharacterSet.ToLower()) // 小写
        {
            case "gbk":
                encoding = Encoding.GetEncoding("GBK");
                break;
            case "gb2312":
                encoding = Encoding.GetEncoding("GB2312");
                break;
            case "utf-8":
                encoding = Encoding.UTF8;
                break;
            case "iso-8859-1":
                encoding = Encoding.GetEncoding("GBK"); //GB2312
                break;
            case "big5":
                encoding = Encoding.GetEncoding("Big5");
                break;
            default:
                encoding = Encoding.UTF8;
                break;
        }
        // 流操作
        Stream stream = response.GetResponseStream();
        StreamReader sr = new StreamReader(stream, encoding);
        string content = sr.ReadToEnd();
        File.WriteAllText("1.html", content, Encoding.UTF8);
        // 关闭释放资源
        stream.Close();
        sr.Close();
        response.Close();
        return content;
    }
    else
    {
        MessageBox.Show("响应失败!");
        return string.Empty;
    }
}
```

然后在点击按钮事件中调用该函数即可:`webBrowser1.DocumentText = GetHTMLbyWebRequest(textBox1.Text.Trim());`就能实现访问乱码的网站,但网站还是有一个问题:在访问sohu时是乱码,其他网站基本都能正常访问.这让我有陷入思考中.下面是访问google,同时在该函数中最后添加`File.WriteAllText("text.html", content, Encoding.UTF8);`还能获取保存静态页面.



同时也可以采用动态方法获取网页的字符集,在采用对应的编码方式进行读取.可以参看下面文章:<http://blog.csdn.net/xx530713660/article/details/6310121>其核心代码是:

```
// 动态获取网页编码方式并读取
Encoding encoding = Encoding.GetEncoding(webBrowser.Document.Encoding);
StreamReader stream = new StreamReader(webBrowser.DocumentStream, encoding);
string conten = stream.ReadToEnd();
```

四.总结

文章主要是针对我在采用WebBrowser编写简单浏览器时遇到的中文乱码问题,通常会显示为"口口口"或"???",不同的编码方式ISO-8859-1、GBK、Big5、utf-8采用相应的编码方式即可避免.文章以PHP+MySQL遇到的中文乱码开头,这里也以它结尾,在PHP+MySQL中需要注意两个方面:

(1).PHP网页|MySQL|Apache|浏览器中|服务器对应的编码方式一致,就会避免乱码问题,其中utf-8对应utf-8,gb2312(国标码)对应txt中ANSI编码方式;


(2).注意有无BOM问题(为识别Unicode文件,以U+FEFF字符开头,作为字节顺序标记byte-order mark,BOM来识别文件中使用的编码和字节顺序),通常Apache中charset设置为utf-8,所以采用UltraEdit设置文件格式为utf-8无BOM另存为即可.

希望文章能帮助到大家,如果文章中有错误或不足之处,请大家海涵!

(By:Eastmount 2013-9-23 21点 <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