

NJUNLP-SMT-WEBDEMO

南京大学自然语言处理组

version 0.1

邹远航

目 录

前 言	1
1 怎样解析 WSDL	2
1.1 Wsdl 及 Web serviceL 简介	2
1.1.1 Web service 介绍.....	2
1.1.2 WSDL 简介	2
1.2 基本步骤.....	3
1.2.1 C# 语言怎样生成可被调用的 WSDL	3
1.2.2 java 语言怎样生成可被调用的 WSDL	3
1.2.3 使用 python 语言解析生成的 WSDL.....	4
参考文献	6

插图目录

1.1 修改后的配置文件	3
1.2 SMT WSDL 文件	4

前 言

NJUNLP-SMT-DEM 是南京大学自然语言处理组的机器翻译的 Web Demo。

主要由以下几部分组成

第一部分将本地的程序做成 WEB Service 的方式供调用

第二部分利用 bootstrap+python 构造 Web Demo (TODO)

1 怎样解析 WSDL

下面首先将介绍 WSDL 及 Web Service 的概念。然后阐述怎样生成可以使用的 WSDL，主要以 c# 语言和 java 语言 (TODO) 为例介绍。最后以 python 语言为例，介绍怎样解析生成的 WSDL 及调用其中的服务。

1.1 Wsdl 及 Web serviceL 简介

在了解 WSDL 之前，需要首先了解 Web Service，Web Service 是一种服务导向架构的技术，通过标准的 Web 协议提供服务，目的是保证不同平台的应用服务可以互操作。

1.1.1 Web service 介绍

通俗的说，我们把计算机后台程序提供的功能，称为服务。根据来源的不同，服务可以分为两种，一种是”本地服务”(使用同一台机器提供的服务，不需要网络), 另一种是”网络服务”(使用另一台计算机提供的服务，必须通过网络才能完成)。

wikipedia 上关于 Web Service 的描述如下

http://zh.wikipedia.org/wiki/Web_service

(1) 本地服务的缺点

a) 可移植性差

如果你想把本机的服务，移植到其他机器上，往往很困难，尤其是在跨平台的情况下。

(2) Web Service 的优势

a) 平台无关

不管使用什么系统，都可以使用 Web Service。

b) 编程语言无关

使用规范的格式调用，就可以使用任意编程语言。

1.1.2 WSDL 简介

WSDL(Web Services Description Language) 指网络服务描述语言。WSDL 是使用 XML 编写的文档，可描述某个 Web service，用来规定服务的位置及提供的操作。

wikipedia 上关于 WSDL 的描述如下

http://en.wikipedia.org/wiki/Web_Services_Description_Language

WSDL 文档利用下面的元素来描述某个 Web service 的

元素	定义
<portType>	web service 执行的操作
<message>	web service 使用的消息
<types>	web service 使用的数据类型
<binding>	web service 使用的通信协议

1.2 基本步骤

1.2.1 C# 语言怎样生成可被调用的 WSDL

- (1) 如何：在托管 Windows 服务中承载 WCF 服务

参考 <http://msdn.microsoft.com/zh-cn/library/ms733069.aspx>

- (2) 生成 WSDL 时需要注意的问题

不要直接粘贴配置文件，防止生成不合法字符

配置文件中的 localhost 改为提供服务的服务器 IP 地址

修改配置文件中的 wsHttpBinding 为 basicHttpBinding

- (3) 修改后的配置文件

```

1 <?xml version="1.0" encoding="utf-8" ?>
2 <configuration>
3   <system.serviceModel>
4     <services>
5       <!-- This section is optional with the new configuration model
6            introduced in .NET Framework 4. -->
7       <service name="Microsoft.ServiceModel.Samples.CalculatorService"
8             behaviorConfiguration="CalculatorServiceBehavior">
9         <host>
10          <baseAddresses>
11            <add baseAddress="http://localhost:8000/ServiceModelSamples/service"/>
12          </baseAddresses>
13        </host>
14        <!-- this endpoint is exposed at the base address provided by host: http://localhost:8000/ServiceModelSamples/service -->
15        <endpoint address=""
16              binding="wsHttpBinding"
17              contract="Microsoft.ServiceModel.Samples.ICalculator" />
18        <!-- the mex endpoint is exposed at http://localhost:8000/ServiceModelSamples/service/mex -->
19        <endpoint address="mex"
20              binding="mexHttpBinding"
21              contract="IMetadataExchange" />
22      </service>
23    </services>
24    <behaviors>
25      <serviceBehaviors>
26        <behavior name="CalculatorServiceBehavior">
27          <serviceMetadata httpGetEnabled="true"/>
28          <serviceDebug includeExceptionDetailInFaults="False"/>
29        </behavior>
30      </serviceBehaviors>
31    </behaviors>
32  </system.serviceModel>
33 </configuration>

```

这里的localhost需要修改为提供服务的服务器IP地址，假设提供某项服务的服务器，IP地址是114.212.189.224。
需要修改为：
<http://114.212.189.224:8000/ServiceModelSamples/service>

wsHttpBinding修改为basicHttpBinding

图 1.1 修改后的配置文件

1.2.2 java 语言怎样生成可被调用的 WSDL

TODO

1.2.3 使用 python 语言解析生成的 WSDL

首先查看需要调用的 WSDL 文件，这里以提供统计机器翻译的 WSDL 为例。
地址如下

<http://114.212.189.224:9000/NJUPhraZTranslationService?wsdl>

(1) SMT WSDL 文件



```
<?xml version='1.0'?>
<wsdl:definitions xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns:wsx="http://schemas.xmlsoap.org/ws/2004/09/mex"
  xmlns:wsa10="http://www.w3.org/2005/08/addressing" xmlns:tns="http://tempuri.org/"
  xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
  wssecurity-utility-1.0.xsd" xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
  xmlns:wsap="http://schemas.xmlsoap.org/ws/2004/08/addressing/policy"
  xmlns:msc="http://schemas.microsoft.com/ws/2005/12/wsdl/contract" xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
  xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata" xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:i0="http://Xin.PhraZ.PhraZDecoder"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" name="TranslationService"
  targetNamespace="http://tempuri.org/">
  <wsdl:import namespace="http://Xin.PhraZ.PhraZDecoder"
    location="http://114.212.189.224:9000/NJUPhraZTranslationService/service?wsdl=wsdl0"/>
  <wsdl:types/>
  <wsdl:binding name="BasicHttpBinding_ITranslator" type="i0:ITranslator">
    <soap:binding transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="Translate">
      <soap:operation soapAction="http://Xin.PhraZ.PhraZDecoder/ITranslator/Translate" style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:service name="TranslationService">
    <wsdl:port name="BasicHttpBinding_ITranslator" binding="tns:BasicHttpBinding_ITranslator">
      <soap:address location="http://114.212.189.224:9000/NJUPhraZTranslationService/service"/>
    </wsdl:port>
  </wsdl:service>
</wsdl:definitions>
```

这里给出了Web Service提供的操作，
即可以被调用的操作名称为Translate

图 1.2 SMT WSDL 文件

(2) 使用 SUDS 解析 WSDL

suds 的使用说明如下

<https://fedorahosted.org/suds/wiki/Documentation>

解析程序如下：

```
from suds.client import Client
# The url below links to NJUNLP SMT wsdl and
# This wsdl provides Translate operation
SMTUrl="http://114.212.189.224:9000/NJUPhraZTranslationService?wsdl"
SMTClient=Client(SMTUrl)

# The url below links to NJUNLP WordSegment wsdl and
# This wsdl provides RunICTCLASSegment operation
WordSegUrl="http://114.212.189.224:9444/WCFICTCLASService?wsdl"
```



```
WordSegClient=Client(WordSegUrl)

testString=unicode("南京大学","utf-8")

WordSegClient.Client(testString)
print SMTClient.service.Translate(testString)
```

(3) 需要注意的问题

在使用 SMT 提供的服务时候，解析服务的文件时需要将文件的编码设置为 **utf-8**. 如果设置编码后，还出现服务不能使用的情况，可以进行牵制转换，这里以 **python** 语言为例：

```
#以南京大学为例
unicode("南京大学","utf-8")
```

