基于网页分类的RESTful Web服务识别系统*

董 丽[1] 吴晓蕊[2] 马戎燕[2] 杨 飞[2] 薛 凯[2] 赵 耀[1]

([1]北京邮电大学 网络与交换技术国家重点实验室 北京 100876
[2]北京临近空间飞行器系统工程研究所 北京 100076)

摘要:随着RESTful Web服务的飞速发展,如何在互联网中有效识别RESTful服务文档,成为Web服务发现领域面临的一个重要问题。本文设计并实现了一种基于网页分类的RESTful服务识别系统,系统主要包括网页预处理模块、分类器训练模块和分类器识别模块,并提出了基于朴素贝叶斯分类器和向量空间模型的识别方法。服务识别系统在实际的RESTful服务集上进行了测试,得到了较高准确率、召回率,表明系统能够有效识别RESTful服务。

关键词:**RESTful Web服务,网页分类,朴素贝叶斯,向量空间模型,服务识别**

Design of A RESTful Web Service Identification System Based on Page Classification

DONG Li[1], WU Xiaorui[2], MA Rongyan[2], YANG Fei[2], XUE Kai[2], ZHAO Yao[1]

([1]State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, 100876, China,
[2]Beijing Institute of Nearspace Vehicle's Systems Engineering, Beijing, 100076, China)

Abstract: With the rapid development of RESTful Web services, how to effectively identify RESTful Web services has become an important issue in service discovery. In this paper we design and implement a RESTful service identification system based on web page classification, which includes page preprocessing module, classifier training module and classifier identifying module. The identification strategy of the system is based on a hybrid framework we proposed, combining Naive Bayes Classifier and Vector Space Model. Experiments with real - world RESTful services prove that our system works effectively with high precision and recall rate.

Keywords: RESTful Web service, Web page classification, Naive Bayes, Vector Space Model, Service identification

1 引言

RESTful Web服务[1]是一类遵循REST[2]风格的Web服务。与传统的WSDL服务相比,RESTful Web服务在灵活性、扩展性和安全性等方面都有更大的优势,并逐渐成为Web服务的主流技术[3]。知名的服务门户网站ProgrammableWeb(http://www.programmableweb.com/apis)的最新统计显示,RESTful Web服务在所有类型的服务中占比达到了70%,且这一比例还有逐渐增加的趋势。

RESTful Web服务的开发是自治的,没有事实上的统一标准或规范,目前虽然有研究者提出了WADL[4]、WSDL2.0等一些规范用于RESTful Web服务的接口描述,但尚未得到推广[5],因此我们通常看到

本文于2014-10-09收到。
*基金项目:国家高技术研究发展计划863项目(2011AA01A102)和中央高校基本科研业务费专项资金(2012RC0606)。

的 RESTful 服务描述文档只是一个普通的 HTML 网页。也有研究学者提出对 RESTful 服务文档添加语义标签,采用 SA - REST[6] 和 hREST[7] 等语义化方法,虽然有较好的发展前景,但目前并未得到广泛的使用。而传统的面向 WSDL 服务的搜索引擎(例如 Seekda① 网站)难也以自动化地发现这类服务。RESTful 服务发布站点 ProgrammableWeb 网站目前已列出了 6000 多个 RESTful 服务,但对互联网上的 RESTful Web 服务的覆盖面仍然较小,存在服务更新不及时、部分服务连接失效等问题,若要获取服务信息仍需要大量人工操作。

因此,如何有效地识别和发现 RESTful Web 服务,已经成为制约进一步推广和使用这类服务的一个重要研究问题。

目前相关的研究工作正逐渐从 WSDL 服务的有效识别[8] 扩展到对语义 Web 服务和 RESTful 服务的识别。针对 RESTful 服务的识别,有研究者提出使用机器学习和文本分类的方法,如基于 SVM 的识别模型[9]、基于特征项抽取的识别方法[10]、基于朴素贝叶斯分类器的识别方法[11],但他们只考虑了服务页面的文本内容,因此识别效果仍有待提高,而且这种方法难以区别服务页面和服务相关的其他页面(如服务相关的博客、论坛、FAQ),后者的文本内容中因为包含与服务相关或相似的文本词语而具有很大的混淆性。

研究发现,RESTful 服务页面往往具有相似的文档结构[12]。例如,服务页面中往往都包含服务调用、服务输入参数、服务输出信息和返回值等部分,这些部分具有各自的结构特征,如 URL、表格、XML/JSON 代码等。因此,如果在识别过程中,综合考虑服务页面的结构特征,将有助于提高识别效果。

本文针对 RESTful Web 服务的识别和抓取问题,提出同时从 Web 网页的文本内容和文档结构两方面进行分析,基于网页分类[13] 的方法进行识别。网页分类方法可以对网页进行归类[14]、选择不同的分类算法进行某类网页的识别[15]。在我们的工作中引入了基于朴素贝叶斯分类器和向量空间模型的网页分类算法,进行 RESTful 服务的自动化识别。此外,我们还设计了一个面向 Web 服务的爬虫引擎 WSCE(Web Service Crawler Engine),并在此基础上,开发了一个专用的服务搜索引擎 WebSuite②,可以提供 WSDL 服务、RESTful 服务及移动 App 的个性化搜索。

本文后续内容:第 2 部分介绍了相关背景,包括朴素贝叶斯算法和向量空间模型;第 3 部分详细介绍了 RESTful 服务识别系统;第 4 部分基于真实 RESTful 服务和普通页面组成的数据集,全面分析和论证了多种实验方案和对比结果,验证了识别系统的有效性。最后,第 5 部分总结全文并分析下一步工作方向。

2　相关背景

2.1　朴素贝叶斯算法

朴素贝叶斯(NB)分类器[16] 是一种基于独立假设的贝叶斯定理的简单概率分类器,它在有监督学习的样本集中能取得很好的分类效果,广泛应用于文本分类、垃圾邮件的监测和过滤等,而且表现出很强的健壮性[17]。

将一个网页文档用 d 表示,文档中的特征项用 t_i 表示,则 d 可以表示为一个特征向量,即 $d=(t_1,t_2,\cdots,t_i,t_m)$,$1\leqslant i\leqslant m$。根据独立假设,各特征项相互独立,因此,定义了两个类别和,分别代表 RESTful 服务页面和普通 Web 网页,则根据贝叶斯定理的公式[18],如式(1)

$$P(C\mid d)=P(C)\frac{P(d\mid C)}{P(d)}=P(C)\frac{\Pi_{i=1}^{m}P(t_i\mid C)}{P(d)},\quad (C=C_{rest},C_{non-rest}) \tag{1}$$

其中,$P(C)$ 表示先验概率,为预估的该类别占所有类别的比例;$P(d)$ 是一个常数,可以不必计算;$P(t_i\mid C)$ 是特征项 t_i 在该类别 C 下的条件概率,可通过训练得到。我们采用词频统计方式进行训练,这样对高频

① http://webservices.seekda.com.

② http://www.websuite.info/.

词和低频词的区分度明显,选择出的特征项更具有代表性。计算公式为

$$P(t_i \mid C) = tf_i = \frac{n_i}{s_c}, (C = C_{rest}, C_{non-rest}) \tag{2}$$

其中 n_i 为该类别下 t_i 出现的次数,S_c 为 C 类别下所有单词出现的次数。最后通过比较文档 d 在两个类别中的后验概率,可以判断出该文档所属的类别。

2.2 向量空间模型

向量空间模型[19](VSM)是由 Salton 等人于 60 年代末首先提出的,在信息检索、文本分类和信息过滤等领域有着广泛的应用。向量空间模型通过把文本内容的处理转化为向量空间中的向量运算,简化了文本处理的复杂度,并有效提高了文本处理的速度。

向量空间模型中将特征项 $t_1', t_2', \cdots, t_i', \cdots, t_m'$ 作为一个 m 维的坐标系,每个特征项 t_i' 对应一个权值 w_i,表示它在文档中的重要程度。常用的权值计算方式有词频 TF、文档频率 DF、TF - IDF 等[20]。文档 d 可以用特征向量表示为 $d = (w_1, w_2, \cdots, w_i, w_m)$,两个文档的相似度通常采用特征向量间夹角的余弦值来计算。向量余弦值的计算公式为

$$\text{sim} = (d_i, d_j) = \cos(d_i, d_j) = \frac{\sum_{k=1}^{m}(w_{ik} \cdot w_{jk})}{\sqrt{\sum_{k=1}^{m} w_{ik}^2 \cdot \sum_{k=1}^{m} w_{jk}^2}} \tag{3}$$

3 RESTful 服务识别系统

我们将 RESTful 服务网页的识别问题视为网页分类的二分类问题,即判断一个 HTML 网页属于服务类网页还是非服务类网页。基于 RESTful 服务的内容特征和网页结构特征,我们提出一种 RESTful 服务识别方法,利用朴素贝叶斯分类器对服务网页内容进行识别,以及利用向量空间模型对服务网页结构特征进行识别,然后为两个识别结果评分并计算加权总分,当总分超过给定阈值时,则认为该网页是 RESTful 服务网页。

RESTful 服务识别系统主要包括分类器训练模块和识别模块两部分,其中两个模块都包含了网页预处理模块,整体的流程图如图 1。在训练阶段,对训练集中的 RESTful 服务页面和非 RESTful 服务页面进行朴

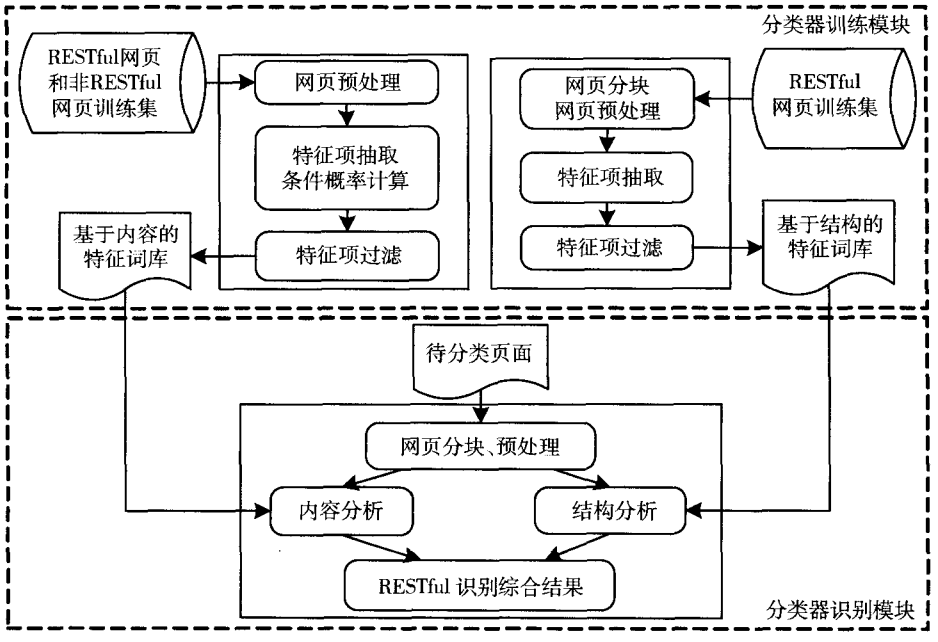

图 1 RESTful 服务识别系统流程图

素贝叶斯分类训练,得到基于网页内容的特征词库,同时对 RESTful 服务训练集进行向量空间模型训练,得到基于网页结构的特征词库;在识别阶段,利用两个词库分别对测试页面的内容和结构的分析,从而判断该页面是否是一个服务页面。

以下对网页预处理模块、特征词库训练模块和分类器识别模块这三个模块分别进行介绍。

3.1 网页预处理

通过对 HTML 网页的预处理,我们将得到从该网页中提取出的特征项及对应的出现频率。网页预处理模块主要包括网页去噪、网页提取和网页分块三个部分。

网页去噪部分将剔除导航条、版权、广告等噪音信息,同时 img、script、style、! doctype 等标签的内容与我们的识别所需内容无关,也进行了剔除。另外对于 title、h1 到 h6、strong 等重要标签的网页内容赋予较高权重(通过给提取的内容乘以一定的增倍因子)。

网页提取部分将提取出网页的纯文本内容,并对文本内容进行分词、去停用词处理,统计单词频率和文档频率,保存在我们的语料库中。

网页分块部分主要对网页的正文部分进行处理。首先提取出正文部分的文本节点,并以 h1 ~ h6 的标题标签进行分块和分层(网页正文通常会以相应的标题划分结构体),当遇到标题节点时,将标题节点后的兄弟节点和标题节点本身都包装到一个分块节点中,然后根据标题节点的嵌套关系将分块分成不同的嵌套等级,并标记到 HTML 页面中。然后对于网页中的每一个分块,依次提取分块节点的 HTML 标签,统计包含该标签的分块数,然后提取分块节点的文本内容,进行 XML 和 JSON 格式的代码文本的探测,统计包含该格式文本的分块数,进行分词、去停用词并提取关键词,统计单词数。

3.2 分类器训练

系统的训练模块对人工获取的服务训练集进行一系列处理,得到两份特征词库,分别是通过朴素贝叶斯算法训练得出的基于网页文本内容的特征词库,和通过向量空间模型训练出的基于网页结构特征的特征词库。

3.2.1 基于文本内容的特征词库

根据朴素贝叶斯的分类原理,在训练阶段我们对 RESTful 服务页面训练集和非 RESTful 服务页面训练集分别进行学习,得到两个类别的特征词表,然后将两个特征词表进行整合,选择区分度高的具有代表性的单词作为最终的特征词,生成 RESTful 网页内容的特征词库。

生成特征词库的系统流程如图 2 所示。经过网页预处理之后,得到每一个单词的词频和文档频率,然后进行特征词抽取,这里用修正的 TF - IDF 算法计算各个单词的权值,进行特征词过滤,使得过滤出来的特征词更具有代表性,而且防止高维诅咒的发生。TF - IDF 的计算公式为:

$$tf - idf_i = tf_i \cdot \log\left(\frac{|D|}{m_i + 1} + 0.01\right) \tag{4}$$

其中 $|D|$ 表示文档集中的文档总数,m_i 表示包含关键词的文档的总数。

对于过滤出的特征词,构建两个类别各自的特征词库,计算词频 TF(式 2)。然后根据该特征词在 RESTful 服务集和在非 RESTful 服务集的 TF 进行特征词库更新,计算条件概率为各自的词频在总的词频中所占的比例,即该特征词在两类中的条件概率之和应为 1。最后将特征词及条件概率以"word#$P(t_i|C_{rest})$#$P(t_i|C_{non-rest})$"形式写入特征词库。

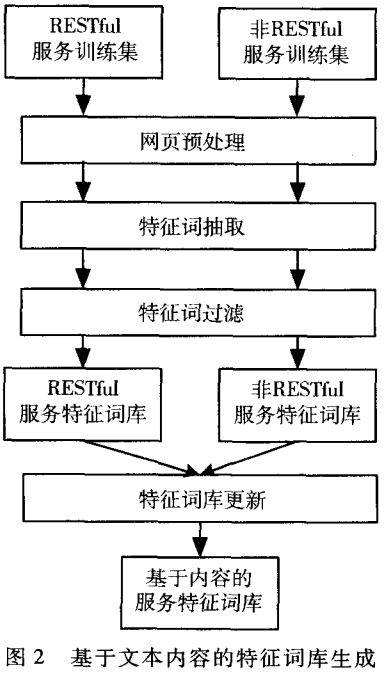

图 2　基于文本内容的特征词库生成

3.2.2 基于网页结构的特征词库

通过对多个 RESTful 服务的分析,我们发现服务文档具有一定的通

用的结构特征，主要集中在 Endpoint、Input 和 Output 这三个结构。Endpoint 为服务调用信息，通常有一个容易区别于普通文本的 URL 形式文本；Input 为服务输入信息，通常以 table 的形式对输入参数进行说明；Output 为服务输出信息，同样会以 table 的格式给出返回的各个域的描述，另外还会有一段 XML 或者 JSON 格式的示例代码说明返回的数据格式[9]。同时各个结构体的标题文字及文本内容的关键词也具有明显特征，如 URL、request、parameters、response 等。

以上这些结构特征以及相应的关键词可以作为各个结构的特征项。我们通过对每个特征项赋予一定的权重值，就可以利用向量空间模型来进行网页结构特征的识别，并用于 RESTful 服务识别。

生成基于结构的特征词库的流程如图 3 所示。我们以 Endpoint、Input、Output 这三类主要的结构特征进行服务识别的基础和依据，特征项主要包括关键词，URL 形式的文本，HTML 的格式标签（table、ul、pre 等）和代码形式的文本（XML、JSON）等。对 RESTful 服务训练集中的页面进行网页分块和网页预处理，并对分块人工标注出的 Endpoint、Input、Output 三类结构，然后针对各个结构提取出特征项，统计的分块数或词频，并计算权重值，选择权值较高的特征项写入对应结构的特征词库中。

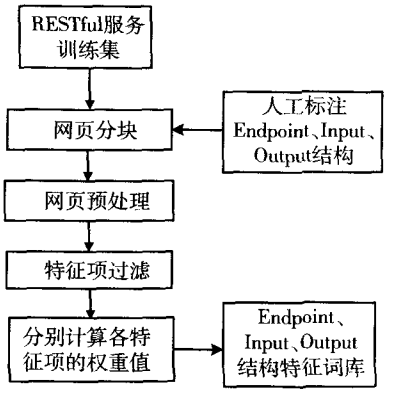

图 3 基于结构的特征词库生成

特征项权重值的计算针对特征项有不同的计算方式。对于 HTML 标签和格式文本，采用包含该特征项的分块数 k_i 与所有特征项总的分块数 K_C 的比值作为权重值，即为 k_i/K_C；对于关键词，采用词频 TF 的计算方式（式 2），计算关键词的出现次数 n_i 与总的关键词出现次数 n_c 的比值。其中 C 分别代表 $C_{endpoint}$，C_{input} 和 C_{output}。

3.3 RESTful 服务识别

RESTful 服务识别模块将根据训练模块中得到的内容特征词库和结构特征词库，分别对网页内容和网页结构特征进行识别，然后将得到网页内容的得分和网页结构的得分进行加权综合，得到的总分作为我们最终判断的依据。算法内部的阈值以及加权系数的设定由我们进行多次实验来确定，在实验部分将会具体说明。识别方法的总体流程图如图 4 所示。

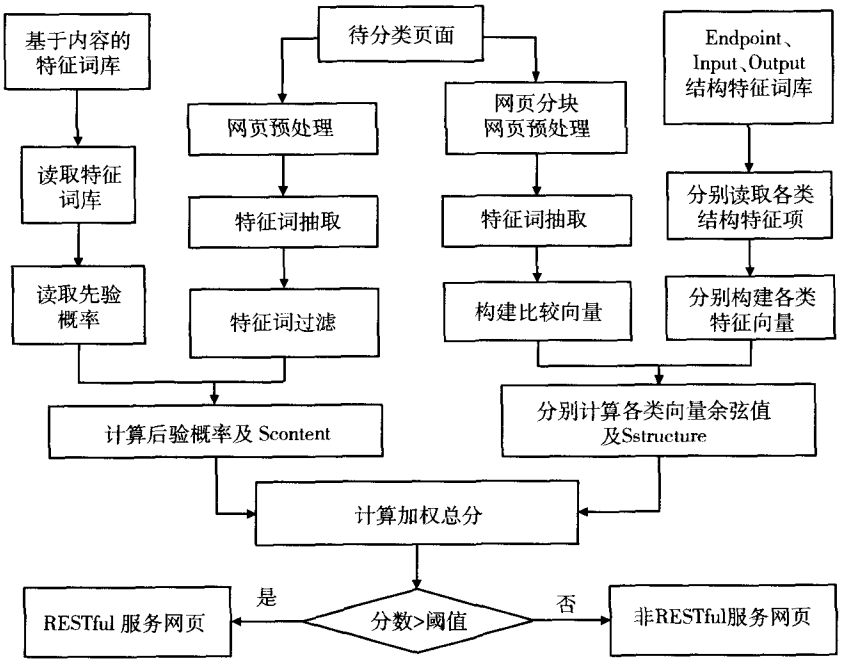

图 4 服务识别模块流程

RESTful 网页内容的识别过程为首先将待分类页面进行网页预处理,得到特征词并统计词频,根据词频进行过滤;同时获取我们训练得到的 RESTful 网页特征词库以及先验概率,根据贝叶斯公式(1)计算分别属于 RESTful 网页类和非 RESTful 网页类的后验概率和,并计算得分。通常取的值作为的识别得分,但实验中我们发现该得分呈指数变化(从 e^{-58} 到 e^{110}),范围跨度很大。为便于我们进行阈值的调整和设置,将该比值取对数,因此最终的网页内容识别分数计算式为

$$S_{content} = \log\left(\frac{P_{rest}}{P_{non-rest}}\right) \tag{5}$$

RESTful 网页结构特征的识别流程为分别获取 Endpoint、Input、Output 结构特征词表,建立结构特征向量;同时将待分类页面进行网页分块处理,针对每一个分块提取出分块中的 HTML 标签和纯文本,建立比较向量,其中关键词的权重值采用词频 TF(式 2),标签的权重值采用 0 - 1(即出现为 1,不出现为 0),并分别计算三类结构特征的相似度(式 6),乘以相应权重系数加入到各自的得分 $S_{endpoint}$,S_{input} 及 S_{output};对 $S_{endpoint}$ 再加上 URL 文本的得分 f_{url},对 S_{output} 再加上 XML/JSON 格式文本的得分;对该页面的所有分块进行以上的计算过程,最终保留各结构的最高得分分别作为该页面的三类结构得分,并将结果与相应阈值的比值进行加权求和,加权系数分别设为和,计算文档结构特征的综合得分 $S_{structure}$。相应的阈值和权重系数均由实验确定,将在 4.2 节中进行说明。$S_{structure}$ 计算式为

$$S_{structure} = \alpha \cdot \frac{S_{endpoint}}{T_{endpoint}} + \beta \cdot \frac{S_{input}}{T_{input}} + \gamma \cdot \frac{S_{output}}{T_{output}}$$
$$(\text{其中 } 0 < \alpha < 1, 0 < \beta < 1, 0 < \gamma < 1, \alpha + \beta + \gamma = 1) \tag{6}$$

图 5 对网页结构特征的识别进行了说明,其中 E、I、O 分别为 Endpoint、Input、Output 的结构特征向量。

最后我们将服务网页内容得分和文档结构得分进行加权,加权系数分别设为 θ 和 μ,计算网页内容和文档结构的综合得分 S,计算式为

$$S = \theta \cdot S_{content} + \mu \cdot S_{structure},$$
$$(\text{其中 } 0 < \theta < 1, 0 < \mu < 1, \theta + \mu = 1) \tag{7}$$

当 S 大于设定的阈值 T 时,则认为该网页为 RESTful 服务网页。阈值经过多次实验测试和分析设定,在下一章中会进行具体说明。

Name: *IdentifyServicePageStructure*
Input: *page, E, I, O*
Output: $S_{structure}$
divide the page into blocks
foreach b_i **in** blocks
　get HTML tag list;
　get text content T;
　if T has URL text
　　$S_{endpoint} += f_{url}$;
　end if
　if T has XML/JSON text
　　$S_{output} += f_{format} * w_{format}$;
　Partition of T and remove stop words;
　foreach $word_i$ **in** tokens
　　n_i = the count of $word_i$;
　　$n_c += n_i$;
　end
　Build V_{word} (weight=tf_i) and V_{tag} (weight=0/1);
　$S_{endpoint} += Ef_{word} * sim(V_{word}, E_{word}) + Ef_{tag} * sim(V_{tag}, E_{tag})$;
　$S_{input} += If_{word} * sim(V_{word}, I_{word}) + If_{tag} * sim(V_{tag}, I_{tag})$;
　$S_{output} += Of_{word} * sim(V_{word}, O_{word}) + Of_{tag} * sim(V_{tag}, O_{tag})$;
end
Get $max(S_{endpoint})$,$max(S_{input})$ and $max(S_{output})$;
$S_{structure} = \alpha \cdot \frac{S_{endpoint}}{T_{endpoint}} + \beta \cdot \frac{S_{input}}{T_{input}} + \gamma \cdot \frac{S_{output}}{T_{output}}$;
return $S_{structure}$

图 5　服务结构特征识别算法

4　实验结果及评价

本文实验验证了 RESTful 服务识别系统的识别效果,并与单独使用朴素贝叶斯、向量空间模型算法以及这两种算法的交集融合和并集融合结果进行对比,给出了实验结果。实验采用的评价标准为服务识别的召回率、准确率[21]。当召回率和准确率都较高的时候,认为该算法的识别效果较好。

4.1　数据集

我们从互联网上收集了 562 个 RESTful 服务页面和 867 个普通 Web 页面,作为服务识别算法实现和验证的训练集和测试集,其中训练集为 80 个 RESTful 服务页面和 120 个普通 Web 页面,其余为测试集页面。RESTful 服务页面来自 ProgrammbleWeb 网站提供的服务文档页面,经过人工的筛选和判断,将网页分为 RESTful 页面或者非 REST-

ful 页面保存到本地。实验中我们只关注直接描述服务详细信息的页面，而与服务相关的网页，例如服务的相关博客、论坛、FAQ 等都作为非 RESTful 页面。

4.2 参数设定

(1)朴素贝叶斯、向量空间模型内部参数设定。朴素贝叶斯识别算法的内部参数为后验概率的阈值。后验概率是判断一个网页是否是服务网页的直接依据，当网页内容识别得分 $S_{content}$(式(5))大于阈值 $T_{content}$ 时则认为该网页的文本内容是与 RESTful 服务相关的。从实验结果(图 6)来看，随着阈值的增大，召回率下降而准确率上升，我们综合考虑召回率和准确率的最优表现，选择了 $T_{content}$ 为 1，此时召回率为 82.71%，准确率为 81.12%。

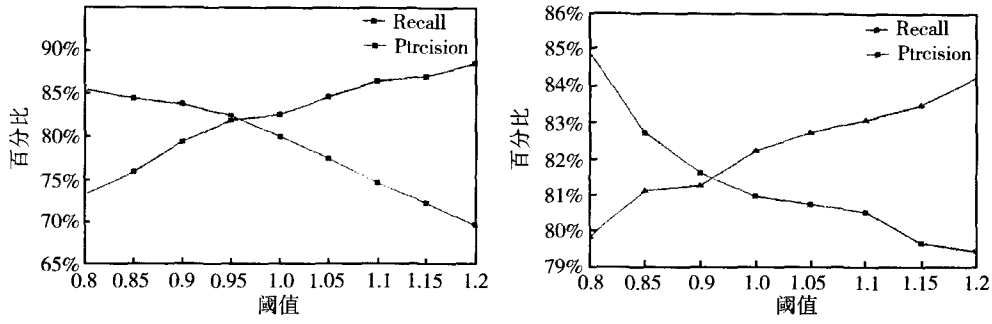

图 6 NB 及 VSM 阈值设定实验

基于向量空间模型的识别算法涉及到 Endpoint、Input、Output 三类结构特征的识别，我们将对三类结构特征分别进行识别和计算得分，最终进行加权得到一个总分 $S_{structure}$(式 6)，并设置一个阈值 $T_{structure}$，当 $S_{structure}$ 大于 $T_{structure}$ 时则认为该网页的结构特征是与 RESTful 服务相关的。算法的内部参数见图 5 中所介绍的参数，通过分别测试不同的权重系数和阈值下的服务网页的召回率和准确率，达到最好的效果时所用的参数就设定为该类结构计算式中的参数。其中 $S_{structure}$ 计算式中三类结构特征的加权系数 α 为 0.2，β 为 0.3，γ 为 0.5，阈值 $T_{structure}$ 为 1，此时的召回率为 80.09%，准确率为 82.62%。

(2)加权算法参数设定。服务识别系统中所用的加权算法的参数主要为朴素贝叶斯和向量空间模型两个算法的加权系数以及阈值。在之前的实验中分别确定了两个算法的内部参数和阈值，两者的阈值都设置为 1，但由于前者的得分变化范围较大，而后者的得分变化范围小，在进行加权的时候为了平衡两个算法对加权总分的影响，各自增加了一个平衡因子，最后实验所用的计算式为

$$S_{content} = 99 + \log_{10}\left(\frac{P_{rest}}{P_{non-test}}\right)$$

$$S_{structure} = 100 \times \left(\alpha \cdot \frac{S_{endpoint}}{T_{endpoint}} + \beta \cdot \frac{S_{input}}{T_{input}} + \gamma \cdot \frac{S_{output}}{T_{output}}\right) \tag{8}$$

增加了平衡因子之后并不会影响两个算法各自的结果，平衡因子的设定只是辅助我们更好的进行加权融合。我们通过实验发现不同的平衡因子下将会得到对应的加权系数(θ,μ)和阈值 T，但最终实验得到的最优的召回率和准确率非常接近，因此在本文中我们采用以上这一组平衡因子。

根据 S 的计算式(7)以及新的 Scontent 计算式和 Sstructure 计算式(8)，我们进行加权系数和阈值的实验，计算服务网页的召回率和准确率，实验的结果如图 7。我们综合选择召回率和准确率均较好时的参数，最终取 $\theta = 0.9$，$\mu = 0.1$，阈值 T 为 102，此时的召回率为 87.75%，准确率为 84.42%。

4.3 对比实验

这一部分将我们的识别方法的效果与 NB 算法、VSM 算法进行了对比，同时将 NB 和 VSM 两种算法的识别结果分别取交集和并集作为共同结果，也加入对比实验中。其中交集是指将 NB 和 VSM 两种算法均识别为 RESTful 服务网页的判定为 RESTful 服务网页，表示为 NBVSM；并集是指将 NB 和 VSM 两种算法的其

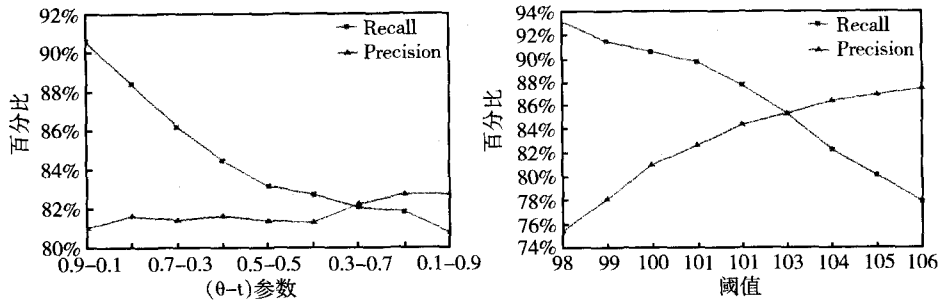

图 7　加权算法参数调整

中一个识别为 RESTful 服务网页的判定为 RESTful 服务网页,表示为 NB + VSM。为更加精确的进行对比实验,每次实验时,我们随机从测试集中抽取出 60 个 RESTful 页面和 90 个非 RESTful 页面,共进行了 20 次测试,记录每次的测试结果及平均值。

从实验数据来看,NB + VSM 的召回率最高,达到 94%,而且稳定性好,这是由于 NB + VSM 只要满足其中一个识别条件就认为是 RESTful 服务网页。这样虽然提高了召回率,但同时也集中了两种识别方法的误差,它的准确率比 NB 和 VSM 都要低。NBVSM 算法的准确率最高,因为它的识别条件最为严格,当两个识别条件分别都满足时才识别服务。这样也导致了它的召回率是最低的,只有 68%,因此不利于我们进行大范围的 RESTful 服务的抓取。

相比之下,本文的识别方法召回率约 88%,比 NB 算法提高了约 6%,比 VSM 算法提高了约 8%;准确率约 85%,比 NB 算法提高了约 3.5%,比 VSM 算法提高了约 1%。因此同时保证了召回率和准确率,理论上来看,由于采用了加权融合的方式,将两种识别方法进行了平衡,能够让两个算法对服务的识别都做出一定贡献,使得它们综合发挥的效果达到最好。

5　结论及将来工作

本文针对 RESTful 服务的识别和抓取问题,提出了基于网页分类方法的 RESTful 服务识别系统。由于 RESTful 服务网页具有共性的结构特征,因此这些结构特征可以作为服务识别的一个重要因素。通过我们对真实服务和易混淆页面进行的多次实验,实验结果证明该识别系统能够有效地提高服务识别的召回率和准确率,说明综合考虑网页文本内容和网页结构特征的网页分类方法能够更好的进行服务识别。

在将来的工作中需要考虑对识别算法的改进,包括对 RESTful 服务网页的结构特征项的进一步优化,对权重值、加权系数采用更合理计算方式以及对朴素贝叶斯分类器和向量空间模型算法的改进和完善。

参　考　文　献

[1] Richardson, L. & Ruby, S. RESTful Web Services[M]. O'Reilly Media, Inc., 2007.

[2] R. Fielding, Architectural Styles and The Design of Network - based Software Architectures, PhD thesis[D], University of California, 2000.

[3] Xinyang Feng, Jianjing Shen, Ying Fan. REST: An alternative to RPC for Web services architecture. Future Information Networks, 2009. ICFIN 2009. First International Conference on. 2009. 7 - 10

[4] Marc Hadley. Web Application Description Language. World Wide Web Consortium, Member Submission SUBM - wadl - 20090831, August 2009.

[5] Mangler, J., Schikuta, E., Witzany, C. Quo vadis interface definition languages? Towards a interface definition language for RESTful services. Service - Oriented Computing and Applications (SOCA), 2009. 1 - 4

[6] Sheth, A., Gomadam, K., and Lathem, J. 2007. SA - REST: Semantically Interoperable and Easier - to - Use Services and Mashups[J]. Internet Computing, IEEE, 2007, 11(6): 91 - 94

[7] Kopecky, J., Vitvar, T., Pedrinaci, C., and Maleshkova. M. REST: From Research to Practice[M]. Springer. Chapter

RESTful Services with Lightweight Machine - readable Descriptions and Semantic Annotations. 2011.

[8] Hatzi Ourania, Batistatos Georgios, Nikolaidou Mara. A Specialized Search Engine for Web Service Discovery[C]. Web Services (ICWS). 2012 IEEE 19th International Conference, 2012. 448 - 455

[9] Nathalie Steinmetz, Holger Lausen, Manuel Brunner. Web Service Search on Large Scale[C]. ICSOC - ServiceWave, 2009. 437 - 444

[10] Qing Liu, Chenhe Liu, Huian Li, Xiang Xu, Lexi Gao. Towards Automatic Discovering for a Real - World RESTful Web Service [C]. Web Information Systems and Applications Conference (WISA), 2012. 39 - 42

[11] Carlos Pedrinaci, Dong Liu, Chenghua Lin. Harnessing the Crowds for Automating the Identification ofWeb APIs. AAAI Spring Symposium - Technical Report, 2012. 58 - 63

[12] Maleshkova, M. Pedrinaci, C., Domingue, J. Investigating Web Services on the World Wide Web. European Conference on Web Services (ECOWS), 2010. 107 - 114

[13] Xiaoguang Qi and Brian D. Davison. Web Page Classification: Features and Algorithms[J]. ACM Computing Surveys, 2009, 41 (2): 12(1 - 31)

[14] Moradi, P., Abdollahzadeh, A., Shiri, M. I. Novel Method for Improving Web Text Classifiers Performance Through Machine Learning. International Conference on Information and Communication Technologies, (ICTTA'06) 2006. 534 - 539

[15] Kennedy, A. ; Shepherd, M. Automatic Identification of Home Pages on the Web. System Sciences, HICSS '05. 2005. 99

[16] Wang Ding, Songnian Yu, Qianfeng Wang. A Novel Naive Bayesian Text Classifier. 2008 International Symposiums. Information Processing (ISIP), 2008. 78 - 82

[17] Dominigos, P.. PaZZani M. On the optimality of the simple Bayesian classifier under Zero - one loss[J]. Machine Learning. 1997, 29(2): 103 - 130

[18] J. Han, MichelineKamber. Data Mining Concepts and Techniques[M]. China Machine Press, 2000.

[19] Dik L. LEE and Huei Chuang. Document Ranking and the Vector - Space - Model[J]. IEEE SOFTWARE, 1997, 14(2): 67 - 75

[20] Yiming Yang. A Comparative Study on Feature Selection in Text Categorization. Proceedings of the 14th International Conference on Machine Learning (ICML'97) 1997. 412 - 420

[21] Olson, David L., Delen, Dursun. Advanced Data Mining Techniques[M], Springer, 1st edition February 1, 2008. 138.

作者简介

董丽,(1990 -),女,湖北宜昌人,硕士研究生,计算机科学与技术专业。