

语音语法在 VoiceXML 中的应用

李永涛 汪林林

(重庆邮电学院软件学院 重庆 400065)

摘要 语音语法是 ASR 的重要参考依据,在基于 VoiceXML 的语音增值业务中应用越来越广泛。本文首先介绍 VoiceXML 与语法的关系,然后描述了语音识别原理,并推导出常用的 Nuance 语法形式逻辑表达式。在此基础上,我们成功构建了一个极具应用价值和市场前景的 VAD 系统,并着重阐述了其中语音语法的应用。

关键词 VoiceXML,DTMF,ASR,VAD,语音语法

The Application of Speech Grammar in VoiceXML

LI Yong-Tao WANG Lin-Lin

(Software College,Chongqing University of Post and TeleCOM, Chongqing 400065)

Abstract Speech Grammar is the important reference frame of ASR, and it is more and more popular in the Voice value-added services based on VoiceXML. The paper introduces the relationship between VoiceXML and Grammar at first, then describes the principle of ASR, and deduces the common formal logic of Nuance speech grammar. Having these, we construct a VAD System, which is very useful and has a very broad market prospect, and emphasis on the using of speech grammar in it.

Keywords VoiceXML,DTMF,ASR,VAD,Speech grammar

1 VoiceXML 与语法

VoiceXML 是由 IBM、Lucent、Motorola、AT&T 四家公司于 2000 年提出的一种专门设计用来支持语音业务的可扩展标记语言,它的主要目标是将基于 Web 的开发内容移植到 IVR(Interactive Voice Response,交互式语音应答)应用中。

基于 VoiceXML 的语音增值业务通过电话终端实现与用户的交互,在交互过程中语法起着非常重要的作用,它用于规定用户的输入范围,并以一定的规则产生输出结果。VoiceXML 语法包括 DTMF(Dual Tone Multi-frequency,双音多频收号器)语法和语音语法。

DTMF 语法是将用户的按键输入产生输出结果集的算法,用户可以通过电话按键输入数字来进行菜单选择,按键信息由 CTI(Computer Telephony Integration,计算机电话集成)服务器收集,这是当前语音业务中应用比较广泛的一种语法。语音语法用于规范用户可能的语音输入,是 ASR(Automatic Speech Recognition,自动语音识别)服务器的重要参考依据,同时也是产生输出结果集的算法,语音输入信息由 ASR 服务器收集。

就人的自身习惯来看,通过言语的交流,利用听和说是人们更愿意接受的交流和获取信息的方式,通过 ASR,用户只需要“说”出想做什么,系统就会做出响应,不必再局限于数字输入,这大大地扩展了用户信息输入范围,提高了业务应用的灵活性。用户语音输入的信息及相应的输出信息由语音语法文件定义,ASR 服务器收集用户语音输入,并以语音语法文件为依据对语音输入进行识别,对与语法文件匹配的输入,返回文件中定义的返回值。

目前,W3C 已经制定了语音识别语法规范,VoiceXML 平台必须支持该规范,同时可以支持其他语音识别规范。Nuance 语法支持 VoiceXML 平台定义的规范,是当前 VoiceXML 中应用比较广泛的一种语音语法。

2 语音识别原理及语法规则

2.1 语音识别原理

语音的识别匹配过程实际上就是一个将声音特征集转换成文本信息的过程,在 ASR 领域里,识别系统已逐渐从整个词的模仿匹配,向音素层次的识别系统方向发展,整个词的模仿匹配系统,或多或少要依赖讲话者,而基于音素层次的识别系统则大大地减轻了 ASR 对讲话者的依赖性,增强了语音识别的智能化。

当 VoiceXML 流程进入收集用户输入阶段,ASR 服务器启动录音,对用户的语音信息进行检测并提取特征,基于音素的识别系统其具体识别匹配过程分为三个阶段:

- 1)音素匹配 根据语音模型集对输入的音素进行识别匹配。音素识别是上下文相关的,即当前音素的识别和之前之后的音素相关。
- 2)音素序列构成单词 根据字典中的单词的定义,找到音素序列对应的单词。
- 3)由单词生成词组或者句子

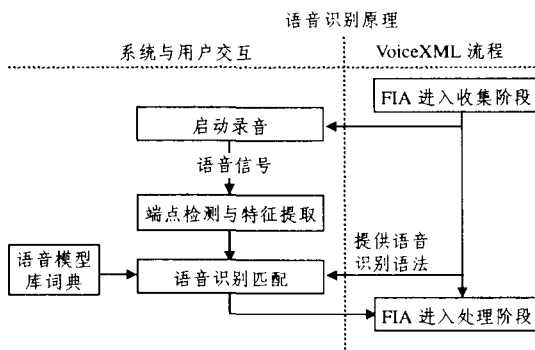


图1 基于音素的语音识别原理图

为应用程序定义的语法决定了应用程序可理解的、能生成词组或句子的单词序列集。如用户的输入与语法匹配成功,ASR 将根据程序定义的语法提供识别结果返回给应用程序, VoiceXML 流程再根据识别结果进行后续处理。在语音识别过程中,匹配识别过程和自然语言理解结果输出这两个过程,都涉及到了语音的识别语法,作为 ASR 的重要参考依据,语音语法在语音识别过程中起着非常重要的作用。

2.2 语法规则

Nuance 识别系统,是一个基于音素匹配、采用有限词汇集的语音识别系统,对于每次识别,限制其识别词汇的范围,就限定了识别过程中的搜索匹配空间的复杂度,能有限缩短匹配过程,提高识别结果的可信度。由于其支持 VoiceXML 平台规范、识别不依赖于说话人且识别率高,因此在语音增值业务中应用非常广泛,我们所构建的 VAD 系统也正是基于 Nuance 识别系统。

在 Nuance 识别系统中,语法定义了可以识别和可以解释的用户语音的输入集合,并定义了识别结果的最终输出形式,其描述语法的语言规范称为 GSL (Grammar Specification Language)。根据实践中常用到的语法文件格式及 GSL 规范,我们推导出了 Nuance 语法文件一般的形式逻辑表达式:

```
<NUANCE_GRAMMAR> ::= <GRAMMARNAME> <GRAMMARRULE>*
<GRAMMARNAME> ::= <TOKEN>
<GRAMMARRULE> ::= <TOKEN> <ITEMS>
<TOKEN> ::= ('_' | [A-Z] | [a-z]) ('_' | [0-9] | [A-Z] | [a-z])*
<ITEMS> ::= OR_ITEMS | SEQ_ITEMS | OPT_ITEMS | GRP_ITEMS | TOKEN | ALIAS | {'<REFER>'} | {'<RETURN>'}
<OR_ITEMS> ::= ' '
[ ' <ITEMS> ( ' ' <ITEMS> ) * ' ' ]
<SEQ_ITEMS> ::= <ITEMS> <ITEMS> *
<OPT_ITEMS> ::= ' ? ' <ITEMS>
<GRP_ITEMS> ::= ' ( ' <ITEMS> ' ) '
<ALIAS> ::= <TOKEN> [ ' ' <TOKEN> ]
<REFER> ::= ' <' <TOKEN> > ' $ ' <TOKEN> > ' > ' *
<RETURN> ::= return ' ( ' <' <TOKEN> > ' ' ' ' ) ' ' ' ' '
```

每个语法文件由文件名和语法规则构成,文件名命名规则由 TOKEN 描述,一个语法内有零或多条语法规则,每条语法规则由规则名及语法项所组成。语法项由各种运算符及表达式构成。OR_ITEMS 表示“或”,其内容由 [] 封装,SEQ_ITEMS 代表顺序关系,OPT_ITEMS 表示可选,可选内容由 ? 进行限定,GRP_ITEMS 代表“与”,其内容封装于 () 中。

ALIAS 用于定义语法规则名的别名,规则名与别名中间通过冒号分隔,语法规则名后紧跟输入输出匹配表达式,而其别名则用于填充应用程序变量,应用程序变量与别名中间通过 \$ 分隔,由 REFER 表示,如 (Name; acct) { username \$ acct > }, 上述语法通过填充方式把语法 Name 的返回值通过变量 acct 返回给应用程序变量 username, 当用户的输入匹配语法文件时, 将从 Name 语法规则中返回相应的值填充应用程序变量 username。RETURN 用于把一个特定的值和一条语法关联起来, 如 Name [(ma3 bol1) { return ("ma3bol1") }], 当用户说“马波”时, Name 语法将返回“ma3bol1”, 并以填充方式赋值给应用程序变量。

通过变量填充的方式, 可以很容易实现自然语言识别, 即用户在表达同样一个意思的时候, 会有多种表达方式, 而识别结果是对该句子的一个解释, 只提取出其中最关键的信息。

如一个银行系统提示用户“您想得到什么服务”, 用户可能会说:

"Withdraw fifteen hundred dollars from savings"

"Take fifteen hundred out of saving"

"Give me one thousand five hundred dollars form my savings account"

表达的是同一个意思, 其中有用的信息也是相同的, 即 action=withdraw amount=1500 accout=savings。应用层关注的就是 action, amount 和 account, 在语法描述中, 就应该分别定义对应于这三个语义单元的语法匹配关系, 并填充三个相应的应用程序变量, 这样, 匹配方式对用户是透明的, 这大大增加了用户语音输入的灵活性。

3 VAD 中语音语法的应用

基于 Nuance 语音识别系统及 VoiceXML, 我们构建了一个极具实用价值及市场前景的 VAD 系统。VAD (Voice Activated Dialing, 语音拨号) 业务就是用户通过语音直接说出被叫的姓名, 系统识别用户的语音输入, 然后接通被叫。使用 VAD 业务的用户, 可以快速便捷地查找好友的各种联系方式, 并可以通过语音输入快速便捷地拨打好友电话号码, 免除了用户记忆繁琐的电话号码, 实现了智能化的电话呼出。

在 VAD 业务中, 我们构造了如下语法结构:

```
PHONEBOOK PHONEBOOK
<Header
(COMMID1; commid ? del ? COMMTYPE1; commtype) { <Callee $
commid> (PhoneType $ commtype> )
}
Header
[
(? qing3(zhao3 yi1 xia4))
]
COMMID1[
(ma3 bol1) { return ("ma3bol1") }
(li3 yong3 tao1) { return ("li3yong3tao1") }
(zuo3 yang2 guang1) { return ("zuo3yang2guang1") }
]
COMMTYPE1
((ban4 gong1)? (dian4 hua4)) { return ("OfficePhone") }
((jia1 ting2)? (dian4 hua4)) { return ("HomePhone") }
(shou3 ji1) { return ("MobilePhone") }
```

上述语法通过变量填充的方式实现了自然语言识别, 通过 COMMID1 及 COMMTYPE1 的返回值分别填充应用程序变量 Callee 和 PhoneType, Header 语法规则为可选内容, 只用于扩展用户语音输入的范围, 而不会被应用程序所使用。Callee 通过 commid 返回用户的好友姓名, 其输入输出匹配关系由 COMMID1 定义, PhoneType 则通过 commtype 返回用户好友的电话类型, 其输入输出匹配关系由 COMMTYPE1 定义, 电话类型定义为可选。上述语法就告诉识别器准备去识别用户说: “请找一下马波的办公电话”, 其中只有马波和办公电话为有用信息, 即会返回给应用程序使用, 而只有好友姓名“马波”为必填项, 其它都为可选, 用户只说“马波”也可以匹配此语法文件。当用户的输入与语法相匹配时返回对应的值给应用程序变量 Callee 和 PhoneType, 即 Callee = "ma3bol1" PhoneType = "HomePhone"。

VAD 业务中最关键、重要的就是语音语法文件的生成与加载, 由于用户可能会频繁地增加、删除或修改好友的相关信息, 且操作逻辑比较繁杂, 我们可借助 Web 来对语法文件的生成和修改进行管理, 当用户对好友的信息进行维护如增加、删除或修改了好友姓名或联系方式等信息时, 需要对语法文件进行及时更新, 语法文件的基本结构维持不变, 只需要对输入输出的匹配信息进行更新, 我们一般只需要对 COMMID1 后的匹配项进行增删除操作, 即当用户维护好友信息时, 我们只需要读取此用户对应的语法文件, 并对其进行修改即可。

在具体业务应用中考虑到用户量大, 且增删操作比较多,

而 VoiceXML 频繁地加载语法文件会对系统性能造成很大的影响,因此,对 Web 端维护的语法文件,我们可以通过一个定时任务在一段时间后对所修改的语法进行统一加载,这样就不会影响系统性能。

语法文件在 VoiceXML 中被加载后进入语音信息收集阶段,我们通过事先定义的 Callee 和 PhoneType 字段来收集用户的语音输入,当用户的输入匹配语法文件时,返回对应的字符串给应用程序,应用程序通过数据库操作查询出用户好友的对应对话号码,如马波的手机,然后上报给智能网设备进行后续接续。

下面代码演示了我们所构建的 VAD 系统最基本的呼出流程:

```
.....
<grammar src="<%=sFinalGrammarPath%>" />
<field name="Callee">
<prompt bargain="true"><audio src="<%=sFinalAudioPath%>
1.wav">请问您要找哪位? 比如:找张三的手机。查询好友电话号码
请按 1,维护好友电话号码请按 2.....</audio></prompt>
</field>
<field name="PhoneType">
<filled namelist="Callee">
<if cond="PhoneType==undefined">
<assign name="PhoneType" expr=" MobilePhone "/>
<if cond="Callee==1">
<submit next="queryfriendphone.jsp"/>
.....
<else/><!-- 如果是语音输入 -->
<clear namelist="getfriendphone"/>
</if>
.....
<subdialog name="getfriendphone" src="getfriendphone.jsp"
namelist="CallerNo Callee PhoneType" expr=true >
<filled>
<if cond="getfriendphone.Ret==0">
<submit next="returnevent.jsp" namelist="CallerNo
CalleeNo"/>
<else/>
<prompt><audio src="<%=sFinalAudioPath%>2.wav">对不起,您未设定该电话号码。</audio></prompt>
<clear namelist="Callee PhoneType"/><reprompt/>
</if>
</filled>
</subdialog>
```

由于每个用户都有其相应的语法文件,因此语法文件的信息我们存放于数据库中,当用户拨打电话进入 VAD 系统,通过用户鉴权后从数据库中取出该用户对应的语法文件路径,然后通过 grammar 元素的 src 属性指定该用户对应的语

法文件路径,并对此语法文件进行加载,此语法文件中包含该用户好友及其各种联系方式,语法的作用域覆盖 Callee 和 PhoneType 两个字段。业务播放相应的提示音后,开始收集用户输入,当用户的输入不匹配语法时,提示用户继续输入,如用户的输入匹配所加载的语法文件,则从语法文件返回定义的对应值给变量 Callee 和 PhoneType,在程序中,我们对 PhoneType 字段进行了灵活处理,用户可以只说好友姓名而说不说联系方式,这时,可以把好友的某种联系方式当成默认的查找类型,如上述代码中,当用户不说联系电话类型时,我们默认处理为手机,然后流程根据 Caller(用户电话号码,从电话交换设备传进业务中)、Callee 和 PhoneType 字段的值进行数据库操作,查找该用户所查询的好友的相应的联系方式,如果查找失败,则进行后续错误处理,如提示用户继续查找或播放提示音后退出系统等。如查找成功,则上报给智能网,智能网根据业务中上报的好友电话号码和主叫号码进行接续和计费操作。

总结 随着基于 VoiceXML 的语音增值业务的规模逐步扩大,应用领域越来越广泛,用户对语音业务的需求也越来越多元化、个性化,传统的电话按键输入已越来越不满足语音业务发展的需要,通过 ASR 实现语音输入将成为语音增值业务发展的新方向。

我们基于 Nuance 识别系统所构建的 VAD 系统,具有实用、方便、易操作等特点,在国内正处于推广和试用阶段,部分地区已开始商用,特别在面向企业和集团的应用中,其效益和价值十分明显,这也是我们推广业务的一个重要方向。我们相信:在不久的将来,VAD 业务必将因其便利性、实用性和效益得到更多用户和运营商的青睐,从而得到越来越广泛的应用,成为语音增值业务新的亮点。

参考文献

- 1 Voice Extensible Markup Language (VoiceXML) Version 2.0. <http://www.w3.org/>
- 2 Nuance Grammar Developer's Guide. <http://www.nuance.com/>
- 3 VoiceXML Programmer's Guide. <http://cafe.bevocal.com/docs/vxml/index.html>
- 4 Beasley R. Voice XML 语音应用程序开发[M]. 北京:机械工业出版社,2002

(上接第 96 页)

的直接启动应用接口。使用应用代理会为 workflow 管理系统的开发和集成都带来很大方便。应用代理与 workflow 引擎之间的数据交互和消息传递都可以使用标准化的 API 和数据格式完成。而应用代理和直接启动的应用之间的数据交互和消息传递利用专门的面向特定应用的集成接口来完成。这样可以大大提高 workflow 系统的柔性。

结束语 随着高校规模的不断扩大和校园各部门对应用软件要求的不断提高,校园日常办公的业务流程日趋复杂,基于分布式 workflow 技术的应用软件集成系统是对高校现有信息系统资源融合的有效方式,必将对高校各处室、各院系等部门之间业务流程的自动化或半自动化、提高工作效率发挥积极的作用。本文利用 Java 和 CORBA 技术提出了基于分布式 workflow 技术的校园应用集成模型,在此模型的基础上并遵循 workflow 系统相关标准,我们正在开发的校园应用软件集成系统,具有良好的开放性和一定的灵活性,可应用于目前高校的办公自动化系统中。其主要优点如下:

(1)可扩展性好:应用扩展时,只需添加相应的分布式处

理结点即可;

(2)健壮性较强:当分布式环境中的某台处理结点出现错误后,只有那些流经该结点的过程的运行受到影响,而不至于使整个系统陷于瘫痪;

(3)能够自然地应用在大规模、分散性强的企事业环境中。

参考文献

- 1 Hollingsworth D. Workflow Management Coalition; The Workflow Reference Model Document Number TC00-1003 Document Status-Issuel. 1, Jan. 1995
- 2 Cichocki A, Helal A S, Rusinkiewicz M, Woelk D. Workflow and Process Automation-Concepts and Technology. Kluwer Academic Publishers, 1998
- 3 CORBA. The OMG Management Architecture. The Object Management Group. <http://www.omg.org>
- 4 The IBM Corporation. WebSphere MQ Workflow. <http://www-306.ibm.com/software/integration/wmqwf/>
- 5 MIMC 接口规范, URL: <http://www.mimc.org>
- 6 (美)Justin Couch, 等著, 考迟, 马琳, 等译. J2EE 宝典. 北京:电子工业出版社, 2002