

## 面向软件模糊自适应的语音式任务目标识别与结构化转换

张晓冰, 杨启亮, 邢建春, 韩德帅

(解放军理工大学 国防工程学院, 南京 210007)

**摘 要:** 已有语音识别方法将用户用英文语音表达的任务目标直接施加到模糊自适应环中, 采取直接将识别结果匹配规则前件的方法, 限制了系统的识别能力。为此, 提出一种语音式任务目标的结构化转换方法。对于语音式任务目标进行句法分析和关键成分提取, 对关键成分进行语义关联拓展, 建立与任务目标等价的语义关联集合, 基于集合完成面向模糊规则前件的结构化转换。通过搭建任务机器人实验系统, 验证了该方法具有较好的语音式任务目标识别能力。

**关键词:** 自适应软件系统; 软件模糊自适应; 目标识别; 结构化转换; 自然语言处理

**中文引用格式:** 张晓冰, 杨启亮, 邢建春, 等. 面向软件模糊自适应的语音式任务目标识别与结构化转换[J]. 计算机工程, 2018, 44(4): 59-65.

**英文引用格式:** ZHANG Xiaobing, YANG Qiliang, XING Jianchun, et al. Recognition and Structured Conversion of Phonetic Form Task Object for Software Fuzzy Self-adaptation[J]. Computer Engineering, 2018, 44(4): 59-65.

## Recognition and Structured Conversion of Phonetic Form Task Object for Software Fuzzy Self-adaptation

ZHANG Xiaobing, YANG Qiliang, XING Jianchun, HAN Deshuai

(College of Defense Engineering, PLA University of Science and Technology, Nanjing 210007, China)

**[Abstract]** The existing voice-identification methods directly apply English voice-based task goals into the fuzzy self-adaptation loop. However, the ability of recognition is limited, as the methods directly matched the raw recognized phrases with rules. In order to solve this problem, a structural transformation approach is proposed. Firstly, voice-based task goals are analyzed through the syntax and their key components are extracted. Then, semantic-equivalence sets are established by expanding semantic-relevance words. Based on these keyword sets of task goals, structural transformation orienting to fuzzy rules' pre-component is finally completed. By constructing task-oriented robot system, it is verified the approach has better recognition ability of voice-based task goals.

**[Key words]** self-adaptive software system; Software Fuzzy Self-adaptation (SFSA); object recognition; structured conversion; Natural Language Processing (NLP)

**DOI:** 10.3969/j.issn.1000-3428.2018.04.010

### 0 概述

自适应软件近十几年一直是软件工程领域的研究热点<sup>[1]</sup>。其能够根据环境变化和用户任务目标来调整自身的参数和行为<sup>[2]</sup>以应对变化。在前期工作中, 课题组提出了软件模糊自适应 (Software Fuzzy Self-adaptation, SFSA) 范型<sup>[3]</sup>以及相应的建模<sup>[4]</sup>和实现工具<sup>[5]</sup>; 基于模糊控制理论构造模糊自适应环, 以消除模糊不确定性对自适应过程的影响。在运行过程中, 模糊自适应环在用户设定的目标制导下, 基于模糊的、不

完备的感知信息进行推理和决策。

软件模糊自适应系统是目标驱动的系统, 与用户持续性的交互是其重要特征。特别是在任务型自适应系统中 (例如送餐机器人系统), 由用户事先设定任务目标到系统中, 系统基于目标自动驱动自适应过程进行感知、决策和施动, 以确保任务目标的完成。在传统方式下, 用户大多根据任务要求先将意图转化为文本形式的任务目标, 进而输入到自适应系统中。这种方法显然存在实时性不强、交互性弱等问题。随着语音识别技术的发展, 用户越来越倾

**基金项目:** 江苏省自然科学基金面上项目 (BK20151451)。

**作者简介:** 张晓冰 (1992—), 男, 硕士研究生, 主研方向为自适应软件、模糊控制; 杨启亮, 副教授、博士; 邢建春, 教授、博士; 韩德帅, 博士研究生。

**收稿日期:** 2017-02-22

**修回日期:** 2017-04-19

**E-mail:** 635768948@qq.com

向通过语音表达意图,即直接将语音式的任务目标输入到系统,从而提高任务目标设定的实时性、便捷性。但是,这种非形式化、非结构化特征的语音式任务目标给系统的语音识别和处理带来了极大挑战。

为能直接捕获语音式任务目标,文献[6]提出直接将(英语)语音式任务目标施加到模糊自适应环。然而此方法限制用户只能按照模糊规则库中预设的词组或短语表达需求,缺乏语义的包容性和拓展性处理,限制了用户对任务目标的灵活性表达。例如要求任务以“quickly”(快速)执行,当表述为“speedily, rapidly”时,虽含义相同但难仍难以被系统识别。该方法对语音输入的强制性,限制了系统对任务目标的识别能力:如果语音式任务目标不够标准,表达不在规则集合内就会导致识别失败。

为此,本文提出一种面向软件模糊自适应的语音式任务目标识别与结构化转换方法。对语音式任务目标进行词法、句法分析,提取依赖关系;根据依赖关系识别任务关键成分进行语义关联拓展;按照模糊规则前件的格式,实现对任务目标的结构化转换;将结构化转换后的语音式任务目标输入软件模糊自适应系统,进行规则匹配,从而实现对话音式任务目标识别与结构化转换的目的。

## 1 相关工作

目前还缺少将语音式任务目标进行自然语言处理后施加到模糊自适应软件系统的针对性工作。然而在软件工程与自然语言处理交叉领域,以及指令抽取与自然语言处理交叉等领域已有不少相关成果:

1) 已经有许多学者尝试将自然语言处理技术应用在软件工程尤其是形式化领域,提出了自动或半自动抽取和建模的相关方法和技术:文献[7]将自然语言处理和 KAOS 需求工程融合(NLP-KAOS),实现了文档集的目标抽取和建模;文献[8-9]提出了将文本需求自动转化为 UML 和 SysML 形式化描述模型。

2) 借助自然语言处理技术实现指令和语义抽取的相关工作有:

(1) 文献[10]运用自然语言理解、知识库、组合范畴语法理论知识,对中文语音指令的解析方法进行了研究,利用条件随机场和知识库结合进行句法分析识别实体,利用支持向量机学习未知指令。虽实现了语言指令解析系统但未能构造完整的语音指令服务机器人平台。

(2) 文献[11]对车辆行驶指令进行抽取和结构化,提出语义分类方法并利用 K-means 聚类对动词进行归类,然而没有实现语音接口。

(3) 文献[12]提出基于语义的写作辅助方法,支持按照词性、搭配和概念扩展等语义条件来检索

短语和自动推荐搭配。

本文借鉴两方面研究成果,实现将语音式任务目标经过自然语言处理后直接施加到模糊自适应软件系统,提高了模糊自适应软件任务目标输入的快速性、灵活性和包容性。

## 2 预备知识

### 2.1 软件模糊自适应

软件模糊自适应是前期研究中提出的一种自适应软件范性<sup>[3]</sup>,能对环境 and 用户任务目标含有的模糊性信息进行处理,采用模糊数学工具来处理自适应系统所面临的模糊性,其概念模型如图 1 所示。

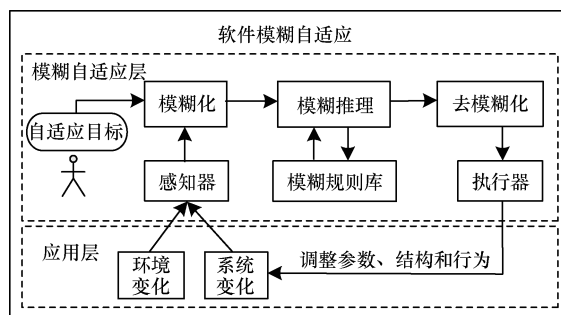


图 1 软件模糊自适应概念模型

如图 1 所示,模糊自适应软件系统根据软件自适应外部机制<sup>[3]</sup>要求,划分为模糊自适应层和应用层。模糊自适应层本质是目标驱动下的“感知-决策-执行”自适应环,其基于用户设定的自适应目标,由感知器感知变化,由模糊推理机制作出合理决策并通过执行器自动调整参数、结构和行为。模糊化、模糊自适应推理和去模糊化是自适应决策的主要环节。

模糊自适应规则库由一套事先固化的规则“if A is  $A_1$  then B is  $B_1$ ”(“A is  $A_1$ ”为规则前件)构成。任务目标与感知信息经模糊化后输入到规则库进行匹配、推理,完成自适应决策。在前期研究中主要考虑了任务目标以文本的形式预先输入的情形,而本文研究工作主要关注如何支持用户灵活而直接的语言(自然语言)形式来输入任务目标,并提取其中的关键成分,把不规则的自然语言输入转换成结构化的表示,从而实现高效精准的匹配推理。

### 2.2 自然语言处理

自然语言作为信息沟通的最主要方式,可为人机交互时提供其他方式不可替代的直接和便利,缺点在于其描述的不充分性、语言歧义性(语法和语义歧义)<sup>[13]</sup>。处理口语和书面语(统称为“语言”)的计算机技术称为自然语言处理<sup>[14]</sup>,其基础研究内容主要包括词法分析、句法分析、语义分析与篇章分析等研究<sup>[15]</sup>。基于自然语言处理(Natural Language Processing, NLP)的技术应用日益广泛,涌现了许多优秀的处理工具如 NLTK、CoreNLP 等。

自然语言处理工具箱 NLTK (Natural Language Toolkit) 在利用 Python 处理自然语言的工具中处于领先地位。NLTK 包含大量的软件、数据供直接使用,例如 WordNet。WordNet<sup>[16]</sup> 是一基于认知语言学的词汇语义网络系统。其构造核心在于根据单词的语义来组织词汇信息:将词汇组织为同义词集合 (synset), 每个集合表示一个词汇概念,每个单词对应一个或多个同义词集;同时概念之间建立不同指针,表达上下位、反义等不同的关系,可以有效表达词汇语义。

CoreNLP<sup>[17]</sup> 提供了一系列集成的自然语言分析工具,能分析语句中每个词语的组成与语法,并且用短语和词汇间的依赖关系来标记出语句的组成结构。

综上,NLTK 和 CoreNLP 都是功能强大的自然语言处理工具,本文主要选取这些工具构建原型系统来验证方法。

### 2.3 应用案例

在生活中逐步开始应用的送餐任务机器人作为应用案例:送餐机器人需要经常与用户交互,获取任务目标,将它们输入到自适应环完成指定的任务。前期研究中构造了一种模糊自适应送餐机器人,如图2所示。

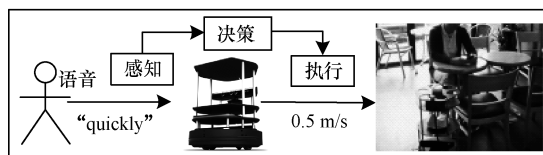


图2 模糊自适应送餐机器人案例

客户要求 (“convey the milk to the restaurant quickly”, “return to the charger” 等), 机器人识别语音, 根据语音直接决策任务动作和目的。例如, 识别得到语音要求速度 “quickly”, 系统综合感知器感知的外界障碍物拥堵程度, 根据模糊推理机制得出速度目标 “speed is fast”, 再经过过去模糊化得到任务速度的具体值 (如 0.5 m/s) 并将其发送到应用层, 机器人将按照指令速度完成任务。

然而研究中仍存在不足, 系统对语音输入的强制性限制了其任务目标识别能力: 虽仍是同义任务目标但难以被识别, 如果语义表达不够标准, 捕获到的词组不在规则集合内就会导致识别失败。例如输入 “quickly” 的同义词 “speedily, rapidly” 等, 由于库涵规则只有 “quickly”, 系统不会识别用户的任务目标。为此本文试图实现语音任务目标输入自适应环的直接性, 并提高模糊自适应软件系统对任务目标的识别能力。

## 3 语音任务目标的识别与结构化转换方法

本文提出的面向软件模糊自适应的语音式任务目标识别与结构化转换方法的总体框架如图3所示。其中, 模糊自适应软件系统包含感知-决策-执行

的自适应环, 其控制模块基于模糊规则构造。通过自然语言处理, 识别非形式化、非结构化的语音式任务目标, 并将其直接作用于自适应环。

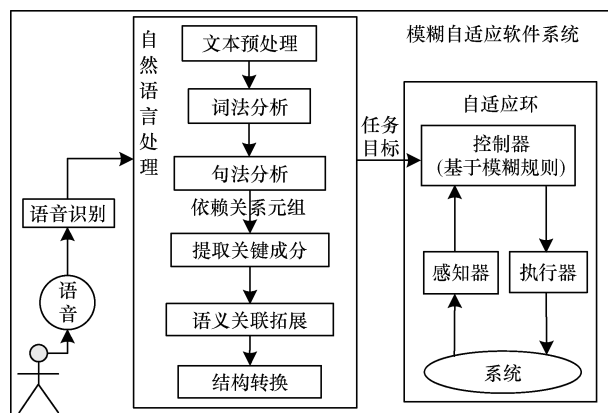


图3 语音任务目标的识别与结构化转换方法

语音任务目标的识别与结构化转换方法主要包括以下关键步骤: 文本预处理, 词法句法分析, 语义关联拓展和结构化转换。

系统经过语音识别, 得到文本形式的语音式任务目标再进行处理。首先定义任务目标的句子成分:

$$S = (w_1 w_2 \cdots w_n), i = 1, 2, \dots, n \quad (1)$$

其中, 句子  $S$  由多个单词  $w_i$  组成。

### 3.1 文本预处理

语音指令通常是简单句或短语, 系统将语音指令转换为文本后, 首先进行预处理工作将句子分割为单独的词汇, 为后续工作打下基础。

分词预处理的形式表述为:

$$S' = f(S), f = [(w_i w_{i+1} \cdots) \rightarrow (w_i, w_{i+1}, \cdots)] \quad (2)$$

可得:

$$S' = (w_1, w_2, \dots, w_n) \quad (3)$$

其中,  $f$  函数实现句子分词,  $n$  为单词个数, 例如 “convey the milk to restaurant speedily”, 经过分词处理后得到 “convey, the, milk, to, restaurant, speedily”。

### 3.2 词法分析

词法分析的主要任务是词性标注 (part of speech tagging), 即对词汇按语法范畴判定其词性进行分类并加以标注的过程。对于多义词, 一个词可以表达多个意义, 但根据具体上下文语境, 可以确定其含义<sup>[13]</sup>。词性标注的形式化表述为:

$$S'' = f'(S'), f' = (w_i | Pos_i), Pos_i \in \{NN, VB, RB, \dots\} \quad (4)$$

其中,  $f'$  标注单词  $w_i$  的词性  $Pos_i$  (part of speech), 词性标注为名词 NN, 动词 VB, 修饰词 RB 等, 可得:

$$S'' = ('w_1, NN', 'w_2, VB', 'w_m, RB', \dots) \quad (5)$$

例如如图4中的步骤1所示:  $S'' = ('convey, VB', 'milk, NN', 'to, TO')$ 。

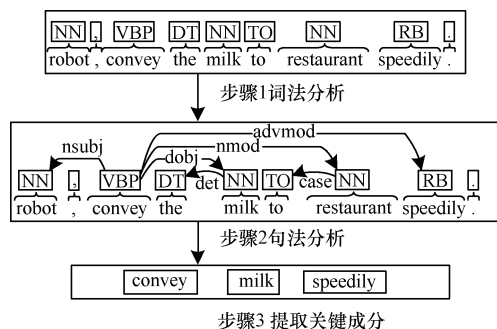


图 4 任务目标的识别

### 3.3 句法分析

句法分析用于对句子语法功能进行分析,确定句子的句法结构。在上一步词法分析句子成分后,明确其之间的相互关系,进而根据其句法结构提取出关系到任务目标的关键成分。

用户给出的语音式任务目标一般由关键成分(动作成分(多为谓语)、目的成分(多为宾语)和修饰成分)构成。任务目标中的关键成分决定自适应软件执行的任务目的和要求,如执行何种任务(“express drink”),其中的修饰成分决定任务指标,如以何种速度(“quickly”)。而在模糊自适应软件中,模糊自适应规则前件一般也由动作成分、目的成分和修饰成分组成,因此要有选择地提取这些任务目标关键成分,便于下一步语义关联拓展。

任务目标的依赖关系可形式表达为:

$$S''' = f''(S''), f'' = [(w_i, v_j) \rightarrow R_m], m = 1, 2, \dots, n \quad (6)$$

可得到包含多个元组  $W_m$  表示的依赖关系  $S'''$ :

$$S''' = [W_1, W_2, \dots, W_n], \\ W_m = (w_i, NN', R_m, w_j, VB'), \\ (R_m \in (dobj, nsubj, advmod, det \dots)) \quad (7)$$

其中,  $W_m$  中  $R$  表示 2 个单词  $w_m$  之间的关系,包括直接宾语关系 dobj, 名词性主语 nsubj, 副词修饰 advmod, 冠词 det 等。例如如图 4 步骤 2 所示,着色的是词性,连接线是词汇之间的依赖关系。元组(“convey, VB”, dobj, “milk, NN”)等抽象了句子中动宾语法关系,根据依赖关系,提取得关键成分(convey, milk, speedily),如图 4 步骤 3 所示。

如图 4 所示,经过以上步骤,实现对用户语音式任务目标的识别。然而仍无法灵活地与模糊规则进行匹配。需要下一步进行语义关联拓展进而实现结构化转换,使得任务目标能较好地匹配模糊规则前件进而驱动软件模糊自适应过程。

### 3.4 语义关联拓展

语义关联<sup>[18]</sup>是描述 2 个词语之间存在的关联关系(包括同义关系、上下位关系、反义关系等)。在分析句法得到任务目标后,为尽可能实现语义关联拓展并减少非必需的计算,选择任务目标的任务关

键成分进行语义关联拓展。把不规则的自然语言输入转换成结构化的表示以更好的匹配规则,(其本质为将与 A 语义关联(如  $\tilde{A}, \hat{A}$ )的任务目标等价收敛为 A)如图 5 所示。关联词汇(细字体)就会统一收敛对应规则库中定义的词汇(粗字体),以利于模糊自适应规则的匹配与推理。

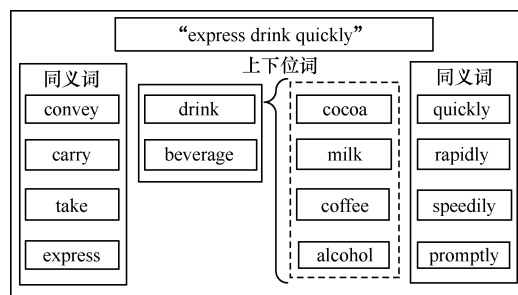


图 5 语义关联拓展示例

由于模糊自适应规则通常采取“if A is  $A_1$  then B is  $B_1$ ”的形式,例如简单的双输入单输出模糊规则如表 1 所示。在规则 1 中,当语音任务目标表述食物类型是饮品以及速度目标为快时,规则设定速度为慢。

表 1 模糊规则示例

| 序号     | if-then 规则  |
|--------|---|
| rule 1 | if food type is drink and speed goal is quickly then speed is slow. |
| rule 2 | if food type is solid and speed goal is quickly then speed is fast. |
| ⋮      | ⋮   |
| rule n | if ... then ...   |

拓展主要分为同义词拓展和上位词(上级概念与从属概念的关系)拓展。语义关联拓展可形式化表示为:

$$W' = W_m | w \rightarrow (W_s, W_h) \\ W_s = \{w_{sim1}, w_{sim2}, \dots, w_{simi}\}, i = 1, 2, \dots, n \\ W_h = \{w_{hyp1}, w_{hyp2}, \dots, w_{hypm}\}, m = 1, 2, \dots, n \quad (8)$$

其中,  $W'$  为语音式任务目标中的任务关键成分  $w_m$  拓展后的关键词集。  $w_m$  可以被拓展为同义词集  $W_s$  和上位词集  $W_h$  ( $w_{simi}$  为第  $i$  个同义词,  $w_{hypm}$  为第  $m$  个上位词)。

#### 3.4.1 同义词拓展

以表 1 中规则为例,若输入“convey the milk speedily”,如果不经语义关联拓展处理,则此时系统不会匹配到任何规则。然而如图 5 所示,“quickly”的同义词是:“speedily, rapidly”等,若进行语义关联同义拓展,可以增加规则前件匹配的灵活性,会匹配到 rule 1。

#### 3.4.2 上位词拓展

仍以上述情景为例,为使输入运送物品为 drink 的下级词汇时可以被系统识别,将词汇进行上位词汇拓展。如图 5 所示,“drink”包括:“cocoa, coffee, milk”等,当输入“convey milk speedily”时,通过上位词联想使原

任务目标被拓展,仍会成功匹配规则 rule 1。

以上各步骤只以例句作为简单示例,事实上,语义关联词库是根据规则库中的规则关键词汇自动构造而成,并不局限于图 5 示例。

### 3.4.3 结构化转换

将提取并且转换的任务关键成分(动作成分、目的成分和修饰成分)转换为系统相统一的规则前件。若输入:“convey this milk to the restaurant speedily”,经过上述步骤将被识别为“convey milk speedily”;随即语义关联等价拓展为“express drink quickly”;再进行结构化转换得“food type is drink and speed goal is quickly”。使得原始的语音式任务目标被转换为与库涵规则前件相统一的表示(如表 1 中 rule 1)。

最后进行规则匹配:将拓展后的语音式任务目标  $W'$  与规则前件进行匹配,若成功则激活相应规则,系统进入模糊决策及后续动作环节。

## 4 语音式任务目标识别与结构化转换算法

根据前文提出的语音式任务目标识别与结构化转换方法,本文设计其实现算法如下所示。

```

输入 识别到的原始语音式任务目标 S
输出 匹配得的规则 rulei

1  S1 ← Tokenize ( S ) //对句子进行单词分割
2  S2 ← Part-of-speech_tagging ( S1 ) //词性标注
3  S3Parser ( S2 ) //句法分析
4  tuple ← Tuple ( S3 ) //以元组形式提取依赖关系
5  for 'relationship' in tuple: //遍历依赖成分
6      return keywordi ∈ List ( keywordi ) //得到目
//标关键成分
7  if keywordi ! = rule in Rule_Base: //若匹配不到库
//涵规则
8      set1 = synonyms ( keywordi ) //同义词拓展
9      set2 = hypernysms ( keywordi ) //上位词拓展
10 keywordi * = ( keywordi + set1 + set2 ) //输出语
//义关联词集
11 trans-goal ← structuring ( keywordi * ) //得到结构
//化转换的任务目标
12 if trans-goal matches rulei in Rule_Base: //匹配规
//则库
13 return rulei //输出匹配得到的规则

```

首先对句子分词(第 1 行)、词法和句法分析(第 2 行、第 3 行)取得依赖关系,进而取得任务关键成分(第 4 行~第 6 行),并进行语义关联拓展(第 7 行~第 10 行),实现任务目标的结构化转换(第 11 行)并匹配规则(第 12 行、第 13 行)。

## 5 原型实现

在前期工作<sup>[6]</sup>中构建了语音驱动的模糊自适应送餐机器人系统。在此基础上,根据前文提出的语音任务目标识别与结构化转换方法,利用自然语言处理工具,对原平台进行改进,构建支持语音式任务

目标识别与结构化转换的自适应送餐机器人原型系统 VoiceGuider。

### 5.1 实现平台

本文以 Turtlebot 为平台构建了模糊自适应送餐机器人原型系统。Turtlebot 是一款低成本机器人,硬件包括 Kobuki 底盘、Kinect 传感器等,由于其设计的大型开源机器人操作系统架构 ROS<sup>[19]</sup>,成为一款理想的开发和测试平台。运行在 Linux 平台的 ROS 类似于分散的进程框架,提供了一个基于操作系统之上的结构化的通信层。软件功能有实时定位构图、图像处理等。

由于机器人操作系统需要 Linux 运行环境,因此选择以 Python 作为编译语言的 NLTK,并将 CoreNLP 中的 Parser 加入到 NLTK 中,利用 Parser 提取依赖关系,由 WordNet 进行语义关联拓展。其模型如图 6 所示,利用语音识别工具,在语言处理功能基础上,集成机器人原有的自适应定位、避障导航、跟随等功能,实现语音式任务目标驱动的模糊自适应送餐机器人的平台搭建。

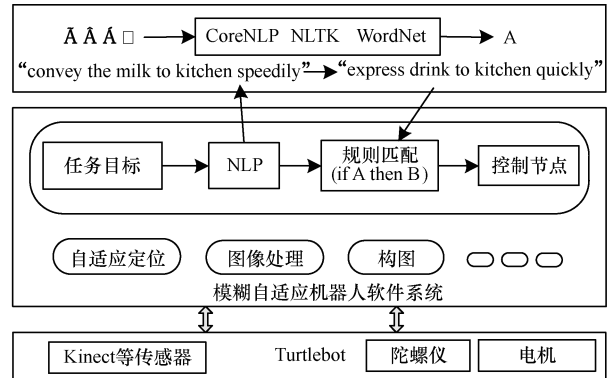


图 6 模糊自适应送餐机器人原型系统 VoiceGuider

### 5.2 模糊自适应送餐机器人软件系统组件交互关系

序列图如图 7 所示,描绘原型系统内部不同软件实体之间的交互关系。

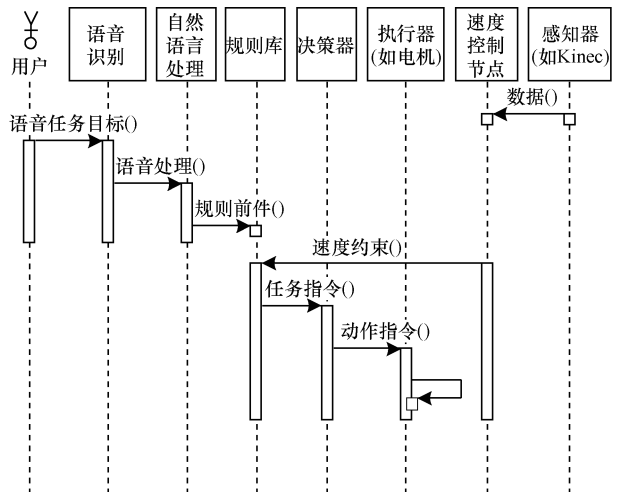


图 7 VoiceGuider 软件系统序列图

语音识别模块持续识别人类的声音并发送回调给语言处理模块。与此同时传感器模块持续发送信息到速度控制节点。语言处理模块将语音式任务目标进行结构化转换,与规则库规则前件匹配后然后发送到决策器。决策器下达动作指令到执行器控制机器人完成所要求任务。

## 6 实验结果与分析

下面以 VoiceGuider 进行实验,验证方法是否实现语音任务目标输入自适应环的直接性,是否提高了对语音任务目标的识别能力。

### 6.1 实验设计

选取 2 台具有模糊自适应能力的送餐任务机器人,分别命名为 V1 和 V2。其中,V1 仍然采用任务目标识别后直接匹配的方法,或称之为任务目标未处理机器人,而 V2 则采用本文所提出的语音式任务目标处理方法,或称之为任务目标识别与结构化转换机器人,其实际情况如图 8 所示。

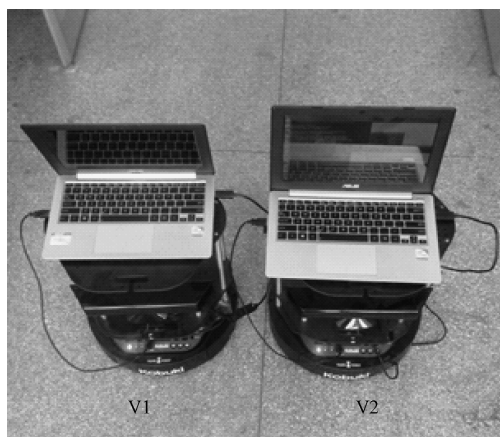


图 8 自适应送餐机器人实物图

在 2 种实验场景下,通过对 2 台机器人下达相同形式的任务目标,记录机器人任务完成情况(实验中不考虑语音识别失败或错误的情况,即假设成功识别所有语音)。

模糊规则库预先定义好,以速度变量为例,其隶属度函数由 5 个模糊区间构成,如图 9 所示(模糊自适应决策验证部分实验详见前期工作<sup>[6]</sup>,在此不做赘述)。

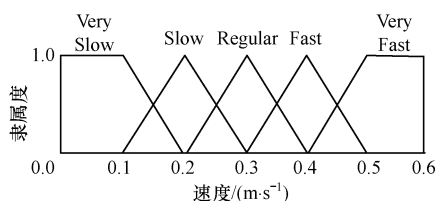


图 9 速度变量的隶属度函数

**实验 1** 语音输入规则库库涵任务目标(库涵指规则库内涵盖的规则)如表 2 所示,规则以动作成分

(express, follow, ...)、目的成分(drink, somewhere, someone, ...)和修饰成分(slowly, quickly, ...)叠加,随机不重复组合查询  $3 \times 3 \times 3 = 27$  次。(动作成分表示执行何种动作;目的成分表示任务的目的或目标;修饰成分决定任务指标,如以何种速度)。

表 2 库涵语音指令

| 动作成分    | 目的成分                   | 修饰成分    |
|---------|------------------------|---------|
| follow  | solid/mixture/drink... | slowly  |
| express | somewhere/kitchen...   | quickly |
| ⋮       | ⋮                      | ⋮       |
| move    | someone/me...          | regular |

**实验 2** 输入非库涵语义关联任务目标,任务目标来自于规则库涵词汇的同义词和下级词:例如“move to the kitchen rapidly”,“express the hot coffee slowly”,“follow me”,任务目标按照动作成分、目的成分和修饰成分,随机不重复组合查询 100 条。

通过实验 1、实验 2,验证本文方法是否实现了语音任务目标输入自适应环的直接性;是否提高了对语音任务目标的识别能力。

### 6.2 结果分析

实验 1 中 V1、V2 都对语音进行了成功处理,识别成功率达到 100% (成功率 = 指令匹配成功数/总数),这是由于两者都直接进行 if-then 规则匹配。

在实验 2 中,在输入非库涵但语义关联任务目标时 V1 匹配率为 0, V2 则维持在很高的识别成功率(分别为 93%、89%、100%),如图 10 所示。

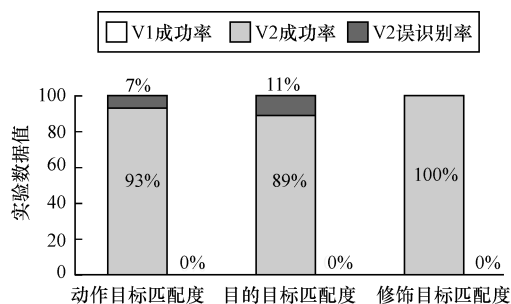


图 10 实验 2 运行结果

通过上述实验,可以判断:

1) 由实验 1 可知,V1、V2 都实现了对任务目标的直接捕获,并将其直接注入模糊自适应环驱动机器人,从而完成任务。

2) 由实验 2 可知,V2 具有更强的任务目标识别能力,针对规则库中未定义然而语义等价的任务目标,V2 显著提高了识别成功率;模糊自适应决策验证部分可见前期工作<sup>[6]</sup>。

3) 在实验过程中,发现本文方法会产生误识别。这是由于在规则过多时,语义关联拓展会对应多个前件,导致规则匹配冲突。

## 7 结束语

本文提出一种面向软件模糊自适应的语音式任务目标的识别与结构化转换方法。对用户语音描述进行句法分析后将关键成分提取并以元组形式化表述,把不规则的语音式任务目标转换成类规则的结构化表示,利用语义关联拓展降低对语音输入的要求。通过构建模糊自适应送餐机器人进行实验验证,结果表明,所述方法有效实现了语音任务目标输入自适应性,提高了系统对语音式任务目标的识别能力和系统可用性。

为克服本文方法存在的误识别缺点,下一步计划利用单词之间的相似度<sup>[20]</sup>来定量计算单词之间的语义关联程度。在匹配到的多个规则前件发生冲突时,通过比较关联词的关联程度提高方法的匹配成功率。

### 参考文献

- [1] KRUPITIAER C, ROTH F M, VANSYCKEL S, et al. A survey on engineering approaches for self-adaptive systems[J]. *Pervasive & Mobile Computing*, 2014, 17: 184-206.
- [2] LEMOS R D, GIESE H, MULLER H A, et al. Software engineering for self-adaptive systems: a second research roadmap[J]. *Lecture Notes in Computer Science*, 2011, 7475: 1-32.
- [3] YANG Qiliang, XING Jianchun, TAO Xianping, et al. Fuzzy self-adaptation of mission-critical software under uncertainty[J]. *Journal of Computer Science and Technology*, 2013, 28(1): 165-187.
- [4] HAN Deshuai, YANG Qiliang, XING Jianchun, et al. FAME: a UML-based framework for modeling fuzzy self-adaptive software[J]. *Information & Software Technology*, 2016, 76: 118-134.
- [5] YANG Qiliang, TAO Xianping, XIE Hui, et al. FuAET: a tool for developing fuzzy self-adaptive software systems[C]//*Proceedings of Asia-Pacific Symposium on Internetware on Internetware*. New York, USA: ACM Press, 2014: 54-63.
- [6] ZHANG Xiaobing, YANG Qiliang, XING Jianchun, et al. Recognizing voice-based requirements to drive self-adaptive software systems[C]//*Proceedings of Computer Software and Application Conference*. Washington D. C., USA: IEEE Press, 2016: 417-422.
- [7] CASSAGRANDE E, WOLDEALAK S, WEI L W, et al. NLP-KAOS for systems goal elicitation: smart metering system case study[J]. *IEEE Transactions on Software Engineering*, 40(10): 941-956.
- [8] DRECSLER R, SOEKEN M, WILLE R. Formal specification level: towards verification-driven design based on natural language processing[C]//*Proceedings of Specification and Design Languages*. Washington D. C., USA: IEEE Press, 2012: 53-58.
- [9] ALKHADER Y, HUDAIB A, HAMMO B. Experimenting with extracting software requirements using NLP approach[C]//*Proceedings of International Conference on Information and Automation*. Washington D. C., USA: IEEE Press, 2006: 349-354.
- [10] 王 聪. 家庭服务机器人中文语音指令解析器的研究[D]. 秦皇岛: 燕山大学, 2015.
- [11] 袁树明. 基于自然语言理解的车辆行驶指令抽取[D]. 北京: 北京邮电大学.
- [12] 朱叶霜, 喻 纯, 史元春. 基于语义的英文短语检索与搭配推荐及其在辅助 ESL 学术写作中的应用[J]. *计算机学报*, 2016, 39(4): 822-834.
- [13] POHL K. Requirements engineering: fundamentals, principles, and techniques[M]. Berlin, Germany: Springer-Verlag, 2010.
- [14] JURAFSKY D. Speech and language processing[M]. London, UK: Pearson Education India, 2000.
- [15] 李 生. 自然语言处理的研究与发展[J]. *燕山大学学报*, 2013(5): 377-384.
- [16] MILLER G A. WordNet: a lexical database for english[J]. *Communications of the ACM*, 1995, 38(11): 39-41.
- [17] MANNING C D, SURDEANU M, BAUER J, et al. The stanford CoreNLP natural language processing toolkit[C]//*Proceedings of Meeting of the Association for Computational Linguistics: System Demonstrations*. Baltimore, USA: ACL, 2014: 55-60.
- [18] GRACIA J, MENA E. Web-based measure of semantic relatedness[C]//*Proceedings of International Conference on Web Information Systems Engineering*. Berlin, Germany: Springer-Verlag, 2008: 136-150.
- [19] QUIGLEY M, CONLEY K, GERKEY B, et al. ROS: an open-source robot operating system[EB/OL]. (2010-11-21). <http://www.andrewng.org/portfolio/ros-an-open-source-robot-operating-system/>.
- [20] 张志昌, 周慧霞, 姚东任, 等. 基于词向量的中文词汇蕴涵关系识别[J]. *计算机工程*, 2016, 42(2): 169-174.

编辑 刘 冰

(上接第45页)

- [13] CURTIS A R, KIM W, YALAGANDULA P. Mahout: low-overhead datacenter traffic management using end-host-based elephant detection[C]//*Proceedings of IEEE INFOCOM'11*. Washington D. C., USA: IEEE Press, 2011: 1629-1637.
- [14] TEAM M. Mininet: An instant virtual network on your laptop[EB/OL]. [2016-10-21]. <http://www.mininet.org/>.
- [15] 张 白, 宋安军. 基于 Iperf 的网络性能测量研究[J]. *电脑知识与技术*, 2009, 36(5): 10227-10229.
- [16] ZAHAVI E, KESLASSY I, Kolodny A. Distributed adaptive routing convergence to non-blocking DCN routing assignments[J]. *IEEE Journal on Selected Areas in Communications*, 2014, 32(1): 88-101.

编辑 索书志