

# 人机对话中自然语言任务命令的识别和计划<sup>①</sup>



王 杉, 丁 磊, 王晓旭

(云南大学滇池学院 理工学院, 昆明 650228)

通信作者: 王 杉, E-mail: 52298256@qq.com

**摘 要:** 随着智能手机和智能系统的飞速发展, 使用自然语言对话的人机交互方式也成为了流行趋势. 但是如果该对话系统需要处理多功能任务类型, 那么将产生复杂的任务命令, 问题的维度也会增加. 尽管目前的 NLP 技术能提供一些解决方案, 但在动态范围内实现动态任务命令识别与处理的能力仍然有限, 解决复杂问题的效果还有待提高. 因此, 在本项工作中, 提供了一种结合 NLP 引擎和任务计划单元的方法, 根据自然语言的指令来设定任务计划, 以便对话系统能较准确地识别命令任务和相关参数, 并为任务生成相应的合理计划. 同时, 为解决自然语言对话中信息的歧义或遗漏, 还研究了一种对话策略, 在必要时能以最少的问题迭代收集对话信息.

**关键词:** NLP; 人机对话; 命令识别; 任务系统; 语音识别

引用格式: 王杉, 丁磊, 王晓旭. 人机对话中自然语言任务命令的识别和计划. 计算机系统应用. <http://www.c-s-a.org.cn/1003-3254/8627.html>

## Recognition and Schedule for Natural Language-based Task Commands in Human-machine Conversation

WANG Shan, DING Lei, WANG Xiao-Xu

(Institute of Technology, Dianchi College of Yunnan University, Kunming 650228, China)

**Abstract:** The rapid development of smartphones and smart operating systems boosts the prevalence of natural language conversations in human-machine interactions. In the case of multiple-function tasks, however, the conversation system will generate a complex task command, and a variety of problems will arise. The current NLP technology can provide some solutions, but its capability to dynamically recognize and process task commands is insufficient in solving complex problems. In this study, we propose a solution that combines the NLP engine and task scheduling unit. Specifically, natural language commands are used for task scheduling, and thus the conversation system can accurately recognize command tasks and related parameters and generate a rational schedule for the tasks. In addition, a conversation strategy is proposed to address ambiguity or information omission in the natural language conversation, by which conversation information can be collected with minimum question-answering iterations when necessary.

**Key words:** NLP; human-machine conversation; command recognition; task system; speech recognition

随着智能设备和人机交互技术的发展, 如“Siri”“小度”等使用自然语言的人机对话系统逐渐出现在各类型智能设备中, 给人们的生活带来了便利<sup>[1]</sup>. 近年来, 得益于 NLP 技术的发展, 在人机对话系统中能够使用自然语言进行表达和交互. 然而, 自然语言中存在的歧义

使对话系统很难完全理解命令任务并执行符合人类意图的任务. 因为人们日常习惯的交谈大多使用短句, 对任务的上下文通常隐含着许多的假设, 对话系统难以准确地识别人类所传达的命令任务. 为了解决此类问题, 一些研究提出了基于受限的自然语言交互<sup>[2]</sup>. 可以

<sup>①</sup> 基金项目: 云南省教育厅科学研究基金 (2021J0797)

收稿时间: 2021-11-01; 修改时间: 2021-12-08, 2021-12-16; 采用时间: 2021-12-21; csa 在线出版时间: 2022-05-30

在一定程度上减轻交互对话中可能出现的歧义,而且也能适当简化计划任务的生成过程,和执行任务的过程。但是,在多功能的人机对话系统中,功能集可能非常大,针对每个任务的编程会带来巨大的工作量。另外,还有一些研究表明对自然语言的限制会降低人机对话系统的可用性和可接受性,尤其是在问询、聊天等日常的对话环境中<sup>[3,4]</sup>。

一个人机对话系统在获得自然语言指令后,通常会按以下的步骤进行任务识别和处理,如图1所示。

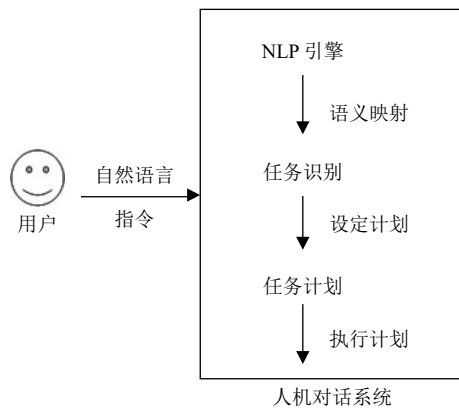


图1 自然语言人机对话系统基本构架

图1表述了基于自然语言的人机对话系统的基本构架,其中NLP引擎是重要的组成部分。除此之外,还有一些问题需要解决。首先,通用的NLP引擎往往仅提供自然语言指令的语法细节,不处理句子的归类和分级,也不提供任务类型以及归纳隐藏在句子中的参数集的机制。因为这需要对系统所运行的领域和对话系统所处领域的功能集有所了解。其次,在自然语言对话中,单向互动会产生歧义,系统不能很准确识别人类的意图。因此,需要基于双向对话进行设计,在对话系统上有许多工作<sup>[5-7]</sup>进行了尝试,但在有效率和准确性方面仍存在不足。针对以上问题和不足,本次研究提出了一个“对话任务代理”器模型,以不受限制的自然语言接收用户发出的任务指令,并将其转换为计划程序。该模型主要包括有以下模块:对话引擎、基础NLP工具集、任务分类器、计划器。对话引擎和基础NLP工具集用于对接收到的自然语言指令和对话进行语法分析和任务识别。任务分类器使用一组已知的任务进行训练,并把这些任务映射到对话系统的任务功能中。任务分类器使用了“Temporal Grounding Graph”模型<sup>[8]</sup>,提供了类似的知识库和任务模板,每个任务都与一个模

板相关联,这个模板会由前置条件和后置条件组成。在给出自然语言指令的情况下,编码生成一组基础逻辑命题,这些命题代表任务预期范围内的初始状态和最终状态,然后基于逻辑的符号状态进行推理。最后,提出了任务计划问题,并设计一个“计划器”以解决该问题。

## 1 相关工作

在自然语言理解方面已经有了大量研究和工作,在这里着重探讨关于人机交互指令的理解,以及在处理模棱两可或不完整任务指令的问题。

在早期的一些研究中,Lauria等人尝试了自然语言给出的导航指令,其中将任务转换为一阶逻辑形式和语义树结构,以提取动作过程和相应的参数<sup>[9]</sup>。之后,有研究将监督学习方法用于学习语义解析器,该语义解析器将路由指令映射到原始动作和控制表达式<sup>[4]</sup>。这些研究在理解自然语言指令方面提供了有价值的指导,但仅通过导航操作却有着很大的限制。

将基于丰富词法资源的语义解释器用于理解通用的自然语言指令也有尝试,例如FrameNet<sup>[10]</sup>。FrameNet通过框架元素、词元、词义类型等各项信息表达了词汇的本质属性,并结合深层句法和语义分析,以实现自然语言指令的解析<sup>[11]</sup>。但是,因为仅将语言信息作为输入,而没有考虑用世界模型的细节去设置上下文,所以这些解析器很难在人机系统执行指令之前完全消除歧义。判别重排序模型(DRMs)在一定程度上解决了歧义问题<sup>[12]</sup>,但是在实际的人机对话场景中,机器所定义的动作程序相对“低级”,而自然的人类命令通常包含较高级的任务目标,这使得此类映射的学习比较困难。在不同的上下文中,很难预测任务的所有原始动作的顺序。因此,将非结构化的自然语言输入生成计划器仍然是研究的焦点。

Thomas等人提出了RoboFrameNet<sup>[13]</sup>,该框架可以将人类指令解析为FrameNet的语义框架,然后生成原始动作序列。他们使用手工制作词汇资源来预测任务,并使用通用的依赖解析器来提取动作参数。但会使语言的理解能力受到限制,而且使用随机搜索来找到动作序列相对低效,并且较难扩展到真实场景。

在近期的一些研究中,Antunes等人提出了一种基于概率计划的体系结构,以尝试针对子任务失败以及子目标的重新排序而采取不同的措施,从而生成复杂任务的计划<sup>[14]</sup>。Misra等人也提出了一种通过在相似情

况下从学习样本中添加、删除和替换动作来实现动作序列接地的方法<sup>[15]</sup>。这些研究虽然较之前的方法有了进步,但是因为需要大量的精力来创建新的数据集和训练数,所以这种学习给定任务的原始动作序列的方法,并不能很好地在新的人机对话领域和具有不同功能的对话系统中应用。

在实际应用中,经常会基于 WordNet 通过单词相似性度量来处理理解包含新动词的指令<sup>[16]</sup>或通过使用特定于环境的训练数据<sup>[17]</sup>消除歧义。但我们的模型提出了一种对话策略来理解新词汇和模棱两可的指令,以实现更好的指令解释和更短的交互。并且系统以 PDDL 形式语言生成输入,减少了在新的对话域中重构系统的工作量。我们还在语义解析信息转换计划器正式语言方法<sup>[18]</sup>的基础上,将其扩展为一个完整的框架,使用人机对话和上下文相关的任务计划,将自然语言指令更准确地转换为计划的原始动作序列。

## 2 系统概述

“对话任务代理”器模型由 4 个主要部分组成。

- (1) 处理对话系统与用户之间双向交互的对话引擎;
- (2) 任务标识符,该标识符标识预期任务和交互中的相关参数;
- (3) 一个计划生成器,用于为给定任务生成有效的计划;
- (4) 知识库,其中包含对话系统正在运行的世界规则模型以及用于计划生成的任务模板。

### 2.1 对话引擎

对话引擎由一组通用的 NLP 工具集组成(可以使用基于深度学习的 TensorFlow 或者 PaddlePaddle 等框架),该工具集对用户发出的自然语言指令进行分词和功能提取,并且设定一个问题框架制定相关问题,以解决任务理解中的歧义。例如,如果用户说“播放一首王某的歌”,则 NLP 工具会对其进行分词处理,得到以下具有语法特征的输出:

(播放, 动词, 中心词), (一首, 数量词, 定语), (王某, 名词, 定语), (的, 助词, 定语), (歌, 名词, 宾语)

### 2.2 任务标识符

为了消除歧义并理解 NLP 工具集提供的语义信息,对话系统和人类必须形成顺畅的沟通,即使人类用自然语言发出指令,也可以将其转换为计算机可以存储和处理的中间表示形式。我们使用了基于 FrameNet 框

架<sup>[19,20]</sup>关系来实现任务标识符的建模。框架语义把现实世界中的事件建立为框架模型,模型使用事件的参与实体(称为框架元素)全面地描述事件,框架元素包括主题和来源,主题表示受事件影响的对象,来源表示存在主题的位置。因此,当任务标识符处理 NLP 工具的输出时,将产生以下输出:

[播放]<sub>激活</sub>[一首王某的]<sub>来源</sub>[歌]<sub>主题</sub>

同时可以设计一个分类器来识别框架和框架元素。如果分类器无法获取足够的判别信息,则会向用户询问相关问题,以解决歧义和信息不足的问题。

### 2.3 计划生成器

在人类预期的任务指令与计算机所能支持的原始动作之间通常无法进行一对一映射,因为某些高级任务目标会包含一系列子任务。所以,为实现任务计划,必须让计算机理解用户真实意图,使任务能够以初始状态迁移至不同的目标状态。

规划域定义语言 (PDDL)<sup>[21]</sup>,是一种标准化的人工智能规划语言,提供了一种以形式化的语言对世界的初始状态和目标状态进行编码的方法。为了能够执行计划的动作序列,每个动作都要映射到 PDDL 状态转换运算符。在 PDDL 中,把谓词、参数变量和状态转换运算符集定义为域,而基础的初始条件和目标条件集称为问题。这样形式化的计划器就可以解决给定领域中的计划问题,从而找到满足人类预期任务目标的机器动作序列。

### 2.4 知识库

知识库存储世界规则模型和任务模板。与“Temporal Grounding Graph”模型的世界状态的感知基础模型相似<sup>[8]</sup>,世界规则模型包含了对话系统正在操作的世界规则和环境,并表示为当前状态的符号。知识库还提供了在任务计划生成器生成计划问题时需要考虑的任务上下文。并在机器执行动作后,还会考虑更新知识库。另一方面,任务模板用于存储生成任务计划问题时所需要的前提条件和后置条件,这些条件与 FrameNet 框架语义相符。如果对于给定的任务指令难以判别,则对话系统将会发起与用户的对话以收集更多信息。

## 3 系统的核心功能

为了实现对任务命令识别和计划的目标,以及保持技术的先进性,本次研究提出了 3 个核心的组成功能。



### 3.1 任务和命令识别

任务标识符的作用是在给定一段句法标记的文本后, 识别预期的任务及其相应的参数, 并将单词或短语分类为任务类型或任务参数. 为此, 我们参考了“条件随机场”(CRF) 模型, 该模型结合了最大熵模型和隐马尔可夫模型的特点, 近年来在分词、词性标注和命名实体识别等序列标注任务中取得了很好的效果<sup>[22]</sup>. 解析过程分为两个连续的阶段——任务类型预测和参数提取. 任务类型预测阶段预测给定文本中可能的词序列. 然后, 在给出前一阶段可能出现的词的预测情况下, 参数提取阶段将预测文本中词元素的序列. 通过预测文本中每个词标记的 IOB 标签来标记预测的序列, 表示是在内部 (I), 外部 (O) 还是在开头 (B).

在任务类型识别阶段, 训练数据如下表示:

$$[z_j = [c_i, rt_i]_{i=1}^M]_{j=1}^N$$

句子  $z_j$  由一个  $M \times 2$  的矩阵构成,  $c_i$  表示句子中的单词,  $rt_i$  表示单词所对应的任务类型的 IOB 标签,  $N$  表示训练数据中此类句子的数量.

在参数提取阶段, 训练数据如下表示:

$$[z_j = [c_i, T_i, ct_i]_{i=1}^M]_{j=1}^N$$

其中,  $T_i$  表示与单词  $c_i$  关联的预测任务类型, 而  $ct_i$  表示参数类型的 IOB 标签.

用于任务类型识别的 CRF 模型定义了如下的条件概率分布:

$$P(rt_{1:M}|c_{1:M}) = \tau \cdot \exp\left(\sum_M \sum_K W_k \phi_k(rt_{i-1}, rt_i, c_i)\right)$$

其中,  $\phi_k$  是特征函数的第  $k$  项分量,  $k$  是特征数,  $W_k$  是第  $k$  项特征的权重,  $\tau$  是归一化因子. 使用梯度下降优化从训练数据中学习权重.

对于参数提取阶段, CRF 模型定义以下条件概率分布:

$$P(ct_{1:M}|c_{1:M}) = \tau \cdot \exp\left(\sum_M \sum_K W_k \phi_k(ct_{i-1}, ct_i, c_i, T_i)\right)$$

最后, 使用词汇和语法特征来预测任务和参数类型. 词汇特征包括单词本身、词根以及单词的上下文. 语法特征包括词和上下文词的词性以及句法依赖性.

### 3.2 基于对话理解任务

即使准确定义了任务标识符, 但当用户以自然语

言进行指令传递时, 仍有可产生歧义而导致识别失败或错误预测. 现有的任务识别模型通常使用指令中存在的动词和动词周围的语言特征进行训练<sup>[15,18]</sup>. 但是当用户为非专业人士时, 用户可能会使用未登记词汇, 那么任务识别模型可能会无法进行有效地预测, 导致错误地预测任务类型. 由于任务类型识别处于处理流程的前期阶段, 因此需要解决其故障和错误, 才能够往后执行指令. 通常在这种情况下, 对话系统会在与人类的对话中进行互动交流, 以能在错误预测的情况下确定指令的含义或正确的任务类型<sup>[3]</sup>. 但是, 非专业用户可能不知道对话系统使用的术语, 所以也不能直接给出提问的正确回答. 例如对话系统如果提问“这是什么类型的任务?”, 用户是完全无法回答的, 因为用户根本不知道系统是如何对指令归类. 为此, 对话系统可以采用一些提示问题, 例如“这是类似于……吗?”, 然后用户给出肯定或否定的回答. 然而, 这又会引发两个新问题. 首先, 对话系统储存的任务类型数量通常有很多, 如果机器人一一提出建议, 则会严重降低用户体验. 其次, 非专业用户可能仍然无法理解该建议的意思, 需要对对话系统进一步解释.

针对以上情况, 这里提出一种对话策略以解决上述问题. 对于给定的指令, 识别任务可以通过动词和充当动词自变量的名词短语共同表征, 并利用数据集中存在的这些任务与参数关系 (使用训练解析器的同一数据集) 来估计已知任务被传递的可能性. 给定一个句子  $Z$  和一组所有可能的任务类型  $T = \{T_1, T_2, \dots, T_n\}$ , 然后定义一个  $n$  元组  $T'$ , 每个元素  $T'_i$  表示  $T$  中的一个任务类型,  $T'_i$  成为给定句子的真实任务类型的可能性决定了  $T'$  的顺序, 即  $P(T'_i|C)$ .

可以使用以下过程估算  $T'$ : 首先, 使用参数类型预测模型, 该模型用于查找指令中可能存在的参数类型. 参数类型预测模型也使用 CRF 来实现:

$$P(ct'_1|c_{1:M}) = \tau \cdot \exp\left(\sum_M \sum_K W_k \phi_k(ct'_{i-1}, ct'_i, c_i)\right)$$

这个模型与参数提取模型的不同之处在于, 该模型可预测每个单词 ( $c_i$ ) 的参数 ( $ct'_i$ ) 的 IOB 标签, 而不用考虑句子的任务类型.

然后, 将预测的  $ct'_i$  里转换为句子中的一个参数类型集合, 并定义为  $CT_P$ . 再定义另一个集合  $CT_D$  表示一个任务类型的参数类型, 该类型出现于训练数据集中.

接下来对训练数据集所有指令中, 满足 $CT_P \subset CT_D$ 的任务类型 $T_i$ 的实例数量进行统计, 生成一个  $n$  元组, 其中元素来自集合  $T$ , 由相应的计数排序. 将计数归一化并将其转换到概率分布, 然后按概率对  $n$  元组进行排序, 最终得到 $T'$ .

为了回答用户有关任务类型定义的问题, 模型为每种任务类型设置一个问题模板, 可以将参数类型预测模型预测的结果填充到模板中. 例如, 如果要求非专业用户清楚查询任务的含义, 则使用模板“是否要我在 [ 主题 ] 中查询 [ 目标 ]”. 例如:

询问 1: 查询张三

回答 1: 这个任务类似于查找吗?

询问 2: 我不明白

回答 2: 你要我在通讯录中查询张三名片吗?

询问 3: 是的

回答 3: 明白了

当确定对话缺少参数时, 就可以使用类似的模板来提问. 非专业用户指定的高级任务也可以由已知任务组成, 这时模型具有与用户进行交互以提取已知任务序列的能力. 在这种对话情况下, 系统会要求用户说出执行高级任务的步骤. 然后, 将响应视为包含多个序列化任务的单个指令, 将其作为输入来生成计划.

### 3.3 计划问题的形式化

一个任务其实就是自然语言指令表示的一个可被实现的目标, 任务也可被假定为由任务的初始状态表示的世界状态给出. 为了更好表达和实现任务计划, 我们采用 PDDL 进行任务计划的描述, 并且按 PDDL 规则将每种任务类型的初始状态和目标状态的模板存储在知识库中. 表 1 显示了此类模板的一些实例.

表 1 在 PDDL 中生成初始状态和目标状态的模板示例

任务	初始状态模板	目标状态模板
bring	hasobject (源, 主题)	hasobject (目标, 主题)
move	$\emptyset$	at-room (目标)
pick	free (主题)	hasobject (目标, 主题)
drop	hasobject (源, 主题)	free (主题)

在此类模板中, 可以通过使用从指令中提取的参数将抽取的变量进行定义, 然后再用世界模型进行验证, 从而生成计划问题的初始状态和目标状态. 通常基于 PDDL 域的谓词并根据建模任务的框架的定义来创建模板.

如果人机对话系统能自主工作, 那么它将持续接

受来自人类用户的新任务并执行这些任务. 在为新任务生成计划问题时, 将使用世界模型来验证假定的初始状态, 并能使计划中的状态转换算子后置条件建模和世界状态的编码更加流畅.

算法 1 显示了生成计划任务的完整过程. 对于一个已序列化, 且包含要实现的多个任务目标说明, 可以通过保留指令中所有任务的世界状态上下文来处理此类复合指令. 存储的模板机制还可以对新任务的集成或迁移进行简化, 并通过引入新模板以及 PDDL 域中的谓词和状态转换运算符来添加新任务.

算法 1. 生成 PDDL 计划任务

```

1) 输入: 目标状态模板:  $T_s$ , 初始状态模板:  $T_0$ , 世界状态:  $H$ , 已解析的参数:  $K$ 
2) 初始化: 初始状态= $\emptyset$ , 目标状态= $\emptyset$ 
3) while( $T_0 \neq \emptyset$ ){
    从  $T_0$  中获得  $c$ ;
    grounded_atom = ground(template_atom,  $K$ );
    while( $H \neq \emptyset$ ){
        从  $H$  中获得 fluent;
        if(fluent 与 grounded_atom 矛盾){
            新增 fluent 到初始状态;
        }
        else
        {
            新增 grounded_atom 到初始状态;
        }
    }
    while( $T_s \neq \emptyset$ ){
        从  $T_s$  中获得 template_atom;
        grounded_atom = ground(template_atom,  $K$ );
        新增 template_atom 到目标状态;
    }
}
4) 输出: 初始状态, 目标状态
    
```

## 4 测试与评估

为了验证和评估系统, 我们使用了中文公开对话语料库 chinese\_chatbot\_corpus<sup>[23]</sup> 的部分数据进行测试. 该库对目前市面上已有的开源中文对话语料进行了整理, 包含 8 组公开闲聊与对话的常用语料. 在测试中, 选取 4 组较有代表性的语料库中的部分数据作为输入, 以测试和评估系统对任务的理解和计划能力. 并与仅使用基本功能的基准系统、以及加入部分功能的试验系统进行比较, 以显示系统在解决问题时的能力. 表 2 显示了使用不同的方法模块的不同系统.

通过与中文公开对话语料库数据集中的注解进行比较,系统的框架语义解析器可以从389个类指令语句中识别到436个任务,且经过比对,其中412个任务符合正确含义,平均总正确率为94.49%(如表3所示),所以系统解析器从自然语言指令中预测任务类型是较为准确的。

表2 任务识别和计划生成的不同方法

系统	任务识别方法	计划生成方法
基准系统	仅使用语义解析器	使用静态模板
交互任务理解	使用解析器的同时,加入交互信息的对话	使用静态模板
使用上下文规划的交互任务理解	使用解析器的同时,加入交互信息的对话	使用世界模型更新模板
完整的计划生成模型	使用共引解析器和对话解析器	使用世界模型更新模板

表3 系统在中文公开对话语料库上进行任务识别的性能

组别	指令数	任务数	正确的预测数	预测率(%)
chatterbot	143	162	150	93.75
青云语料	114	127	120	94.48
小黄鸡语料	112	126	122	96.82
微博语料	20	21	20	95.23
总计	389	436	412	94.49

在任务识别的基础上,我们进一步进行了任务计划生成的测试(如表4所示)。

表4 系统在中文公开对话语料库上进行计划生成的性能

系统(平均生成率)	生成计划数	生成率(%)	组别
基准系统(41.01%)	94	58.02	chatterbot
	52	40.94	青云语料
	34	26.98	小黄鸡语料
	8	38.09	微博语料
交互任务理解(73.02%)	123	75.93	chatterbot
	92	72.44	青云语料
	90	71.43	小黄鸡语料
	16	76.19	微博语料
使用上下文规划的交互任务理解(81.96%)	131	80.86	chatterbot
	101	79.52	青云语料
	103	81.74	小黄鸡语料
	18	85.71	微博语料
完整的计划生成模型(92.01%)	150	92.59	chatterbot
	119	93.7	青云语料
	109	86.5	小黄鸡语料
	20	95.24	微博语料

从测试可以发现即使有较高的任务预测率,但是由于基准系统不使用对话来获取缺少的信息,并且使用了静态模板,不会为多个目标状态冲突的任务指令

生成计划.基所以准系统的计划生成性能较低(平均生成率仅41.01%)。

在交互任务理解系统中,通过添加对话模块以获取缺少的参数可以在一定程度上解决了指令不完整的问题,系统使用对话策略来引导模糊指令的识别分析,生成计划的有效率得到了提高(平均生成率为58.75%),一定程度上解决了指令不完整的问题.但是使用静态模板无法处理需要上下文相关计划的复杂指令,于是在增加了上下文规划的交互任务理解模型中,并使用了基于世界模型的动态更新模板,使得计划生成率进一步提高(平均生成率为73.04%)。

最后,通过在系统中使用了共引解析,把指代相同对象的名词归类指向同一个实体.系统生成了353个任务计划,获得了较好的计划生成率(平均生成率为92.01%),这也与正确的任务识别率(94.49%)相接近。

我们也寻求了正确预测任务但未生成有效计划的原因,主要是包含在测试数据中已经存在的参数未能有效地析出;其次在某些情况下,由于错误地假定上下文,任务的计划失败会导致同一指令中的从属任务也失败。

## 5 结论与展望

在本次工作中,研究了人机对话系统的任务对话代理模型,该代理根据自然语言指令执行任务计划,并提出了一种方法来整合自然语言解析器和对话代理的自动计划器.为了实现该方法,我们开发了一种语言理解模型来识别预期的任务和相关的参数,同时研究了一种对话策略,对模棱两可的指示进行识别,以完成与人机对话的互动.测试和评估表明系统能基于知识库的上下文感知推理完成与生成计划的对话,并且能以较少的询问获得有意义的问题.在后续的研究中,可以对该系统进行进一步的扩展,例如加入概率和分层计划,还可以基于更丰富的知识库进行推理,以期获得更好的对话效果和任务计划识别。

## 参考文献

- 车万翔,张伟男.人机对话系统综述.人工智能,2018,(1):76-82.
- Matuszek C, Herbst E, Zettlemoyer L, et al. Learning to parse natural language commands to a robot control system. In: Desai JP, Dudek G, Khatib O, et al. eds. Experimental

- Robotics. Berlin, Heidelberg: Springer, 2013. 403–415.
- 3 Macglashan J, Gao R, Winner K, *et al.* Muresan: Learning to interpret natural language instructions. Montreal: Association for Computational Linguistics, 2012. 1–6.
  - 4 高志伟, 何伟. 特定任务域中基于自然语言的人机对话模型研究. 石家庄铁道学院学报, 2000, 13(2): 44–47.
  - 5 Lam C, Hannah MA. The social help desk: Examining how Twitter is used as a technical support tool. *Communication Design Quarterly*, 2016, 4(2): 37–51.
  - 6 Patidar M, Agarwal P, Vig L, *et al.* Automatic conversational helpdesk solution using Seq2Seq and slot-filling models. *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. Atlanta: ACM, 2018. 1967–1975.
  - 7 刘娇, 李艳玲, 林民. 人机对话系统中意图识别方法综述. *计算机工程与应用*, 2019, 55(12): 1–7, 43. [doi: [10.3778/j.issn.1002-8331.1902-0129](https://doi.org/10.3778/j.issn.1002-8331.1902-0129)]
  - 8 Paul R, Barbu A, Felshin S, *et al.* Temporal grounding graphs for language understanding with accrued visual-linguistic context. arXiv: 1811.06966, 2018.
  - 9 Lauria S, Bugmann G, Kyriacou T, *et al.* Converting natural language route instructions into robot executable procedures. *Proceedings of the 11th IEEE International Workshop on Robot and Human Interactive Communication*. Berlin, Heidelberg: IEEE, 2002. 223–228.
  - 10 Fillmore CJ, Johnson CR, Petruck MRL. Background to *framenet*. *International Journal of Lexicography*, 2003, 16(3): 235–250. [doi: [10.1093/ijl/16.3.235](https://doi.org/10.1093/ijl/16.3.235)]
  - 11 贾君枝, 邵杨芳. *FrameNet* 的语义类型研究. *情报理论与实践*, 2007, 30(5): 689–692. [doi: [10.3969/j.issn.1000-7490.2007.05.032](https://doi.org/10.3969/j.issn.1000-7490.2007.05.032)]
  - 12 Dinarelli M, Moschitti A, Riccardi G. Discriminative reranking for spoken language understanding. *IEEE Transactions on Audio, Speech, and Language Processing*, 2012, 20(2): 526–539. [doi: [10.1109/TASL.2011.2162322](https://doi.org/10.1109/TASL.2011.2162322)]
  - 13 Thomas BJ, Jenkins OC. *RoboFrameNet: Verb-centric semantics for actions in robot middleware*. 2012 IEEE International Conference on Robotics and Automation. Saint Paul: IEEE, 2012. 4750–4755.
  - 14 Antunes A, Jamone L, Saponaro G, *et al.* From human instructions to robot actions: Formulation of goals, affordances and probabilistic planning. 2016 IEEE International Conference on Robotics and Automation (ICRA). Stockholm: IEEE, 2016. 5449–5454.
  - 15 Misra DK, Sung J, Lee K, *et al.* Tell me Dave: Context-sensitive grounding of natural language to manipulation instructions. *The International Journal of Robotics Research*, 2016, 35(1–3): 281–300. [doi: [10.1177/0278364915602060](https://doi.org/10.1177/0278364915602060)]
  - 16 Lu DC, Chen XP. Interpreting and extracting open knowledge for human-robot interaction. *IEEE/CAA Journal of Automatica Sinica*, 2017, 4(4): 686–695. [doi: [10.1109/JAS.2017.7510628](https://doi.org/10.1109/JAS.2017.7510628)]
  - 17 Misra DK, Tao KJ, Liang P, *et al.* Environment-driven lexicon induction for high-level instructions. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Beijing: Association for Computational Linguistics, 2015. 992–1002.
  - 18 Lu DC, Zhou Y, Wu F, *et al.* Integrating answer set programming with semantic dictionaries for robot task planning. *Proceedings of the 26th International Joint Conference on Artificial Intelligence*. Melbourne: AAAI Press, 2017. 4361–4367.
  - 19 Johnson CR, Fillmore CJ, Wood EJ, *et al.* The *FrameNet* project: Tools for Lexicon Building. Manuscript, Berkeley: International Computer Science Institute, 2001.
  - 20 张鹏, 李国臣, 李茹, 等. 基于 *FrameNet* 框架关系的文本蕴含识别. *中文信息学报*, 2012, 26(2): 46–50. [doi: [10.3969/j.issn.1003-0077.2012.02.009](https://doi.org/10.3969/j.issn.1003-0077.2012.02.009)]
  - 21 Ghallab M, Howe A, Knoblock C, *et al.* PDDL—The planning domain definition language. Technical Report, Yale: Yale Center for Computational Vision and Control, 1998.
  - 22 洪铭材, 张阔, 唐杰, 等. 基于条件随机场 (CRFs) 的中文词性标注方法. *计算机科学*, 2006, 33(10): 148–151, 155. [doi: [10.3969/j.issn.1002-137X.2006.10.040](https://doi.org/10.3969/j.issn.1002-137X.2006.10.040)]
  - 23 codemayq. 中文公开聊天语料库 *chinese\_chatbot\_corpus*. [https://github.com/codemayq/chinese\\_chatbot\\_cor](https://github.com/codemayq/chinese_chatbot_cor). (2019-04-15)[2021-01-14].

(校对责编: 孙君艳)