基于OWLReady2的智能人机对话系统

**宋丛威**1

(浙江工业大学之江学院 理学院 绍兴 310302)1

**摘要**OWL是本体论知识管理描述语言，OWLReady2是它的Python接口，OWLReady2封装了推理引擎HermiT, 可以进行基于描述逻辑的推理。利用Python的动态编程功能，实现具有推理能力的人机对话系统。

**关键词** OWLReady2，描述逻辑，人机对话系统

**中图法分类号：**（3位数字）**文献标识码：**A

**Human-Machine Dialogue System based on OWLReady2**

SONG Cong-wei

(Department of Sience，Zhijiang College Zhejiang University of Technology， Shaoxing 310302，China)1

(School of xx，xx University，City ZipCode， Country)2

**Abstract**

**Keywords**  OWLReady，DLs

引言

研究的最终目的是建立一个具备逻辑推理能力和知识存储的人机对话系统。这种系统已经被创造出来，并可以被更多的软件或计算机语言实现，比如Prolog，CLIPS,以及Python实现的PyKE。不过使用它们，需要学会一种计算机语言和现代逻辑学原基本理。有一些网络机器人，能够实现自然语言交互，但是不具有逻辑推理能力。人们自然会设计一套解析程序，将自然语言翻译成计算机语言，然后交给软件进行推理。本文的特点是用OWLReady2（Python对OWL的封装）实现自然语言人机交互。我们会写一个解析器，将自然语言直接翻译成OWLReady2对象(本质上是Python对象)，然后交给一个对话流程，在这个流程中，读取并处理这些对象。每一次交互，程序应该完成下面几个任务：

1. 解析陈述句，并作为知识存储，如果包含未知信息，应该反问用户。还要检验与原有知识的一致性。
2. 解析一般疑问句，判断真假。
3. 解析特殊疑问句，计算疑问词对应的对象。

描述逻辑DLs、OWL与OWLOready2简介

描述逻辑DLs一般被认为是一阶逻辑的子集，最大的特点是可判定性。

OWL是一种本体标记语言，用于构建本体论知识库。它和网络标记语言HTML/XML风格上非常相似，其实就是XML的一种应用。OWL可以表示DLs的每一种表达式，而OWL推理器也能实现DLs的推理。

OWLReady2是用Python实现的OWL封装。熟悉Python的读者来说，可以少学一种新语言。当然，它也实现了DLs的推理，以及“封闭世界假说”。它的 推理引擎是HermiT和Pellet（基于Java）。即使不能保证OWLReady2实现了DLs的所有推理形式，也因为它是用Python写的开源软件，我们可以进行功能上的几乎无限制的扩展。本文就可以看做是一个扩展，即增加一个解析模块和动态化本体信息存储模块。动态化本体信息存储模块负责信息的同态读取，因为信息都是在人机交互中读取、修改和添加的；解析模块可以让我们采用自然语言与机器沟通，而不是直接用计算机语言或逻辑公式。

DLs和OWLReady2中的表达式（函数、类、运算符）都是一一对应的。因此，如果用户熟悉DLs，一种更为简单的选择是把DLs翻译成OWLReady2对象。此外，OWLReady2已经支持SWRL规则，因此

人机对话系统框架

在引言中，我们已经提到了建立对话系统的思路和任务，本节讨论技术细节。

最困难的是解析过程。我们采用纯Python实现的文法解析库pyparsing，它具有良好的扩展性，可以根据需要改造解析结果。

形式上，每一次输入的句子都被理解成是本体知识库（描述逻辑表达式的集合）到自身的变换。对于陈述句，这个变换就是加入新的表达式，但不必有返回值；而对于疑问句，这个变换是恒等变换，同时返回推理引擎给出的该问题的答案。本文的目的是建立一个实用的人机对话程序，并不打算对这个变换进行严谨的形式化的讨论，只是给出变换的大致形式。

1. 陈述句：

如果进行一致性检验，则定义如下，

1. 一般疑问句：
2. 特殊疑问句：

其中是本体知识库,是的解析结果，即描述逻辑表达式。如果采用封闭世界假设，那么一般疑问句的回答只会是真或假。特殊疑问句牵涉变量，而变量的取值不一定是唯一的。更严格的形式化理论会在以后的研究中完成。

在这个变换过程中，我们可以设计各种扩展程序，记录元信息，丰富推理系统。

对人机对话系统实现

演示，基本实现了第2节的框架。

除此之外，对话系统还具有如下功能

历史记录，提供元信息。

命令交互。

源代码已经上传至GitHub，供分享和交流。

1. 总结

通过测试，该系统能够顺利完成几个对话任务。

目前，该系统还比较简单，主要受限于自然语言解析。

以后，我们研制更强大的语言解析器，尤其是增加复合语句的解析。此外，我们也正在开发适合编程的形式语言，避免引入自然语言中非逻辑因素。人机互动过程编程这种语言的读写循环（REPL）。

参考文献

[]Lamy J B. Owlready: Ontology-oriented programming in Python with automatic classification and high level constructs for biomedical ontologies[J]. Artificial Intelligence in Medicine, 2017, 80.