基于错误驱动的语义文法自动扩展学习方法研究

王东升[1],王卫民[1],祁云松[1],王 石[2],曹存根[2]
(1. 江苏科技大学计算机学院,江苏镇江 212003;
2. 中国科学院计算技术研究所中科院智能信息处理重点实验室,北京 100190)

摘 要: 面向领域的自然语言理解技术是垂直搜索引擎、领域相关问答系统等应用的核心技术之一. 本文在已构建的基于本体和语义文法的自然语言理解系统的基础上,提出一种基于错误驱动的语义文法自动扩展学习方法,对于解析错误的句子,利用核心文法生成部分解析树,按照打分函数选择一组最佳的部分解析树,利用预测模型预测部分解析树的上层节点并试图构建完整的解析树,从而学习得到新的文法规则,对于学习得到的不同类型的规则进行验证并更新核心文法库,通过对句子的可学习性度量来筛选学习对象,从而提高文法扩展学习的整体质量和效率. 分别在两个不同规模的领域数据集进行了测试,在交互式学习范式下,测试对比了学习算法在不同规模领域的学习效率,在批量学习范式下,测试对比了更新后的文法和核心文法在两个领域数据集上的准确率和识别率等性能指标. 实验结果表明,本文所提出的方法是有效的.

关键词: 语义文法; 文法扩展; 自然语言理解; 领域; 本体
中图分类号: TP311 **文献标识码:** A **文章编号:** 0372-2112 (2021)02-0248-12
电子学报 URL: http://www.ejournal.org.cn **DOI:** 10.12263/DZXB.20190159

Automatic Error-driven Learning Method of Semantic Grammar

WANG Dong-sheng[1], WANG Wei-min[1], QI Yun-song[1], WANG Shi[2], CAO Cun-gen[2]
(1. *School of Computer Science, Jiangsu University of Science of Technology, Zhenjiang, Jiangsu* 212003, *China*;
2. *Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing* 100190, *China*)

Abstract: Domain-specific natural language understanding technology is one of the core technology of vertical search engines, domain-specific question answering system and other applications. This research focus on a novel constrained semantic grammar and its automatic learning methods based on an existing domain-specific question answering system. An error-driven learning method of semantic grammar is proposed. The method first partially parses the ungrammatical sentence based on the core semantic grammar, then it attempts to build a complete parse tree, including predicting the top-level node of the partial parsing tree, generating and verifying hypotheses of new grammar rules. Learnability metrics is used to filter sentences in the training corpus to improve the overall quality and efficiency of grammar extending algorithm. The proposed algorithm is applied to two domains of different scales. In the interactive learning paradigm, learning efficiency are compared in different domains. In the batch learning paradigm, the paper compares the accuracy, MRR and recognition rate of the extended grammar and core grammar on twodatasets. The test results show that the proposed method is effective.

Key words: semantic grammar; grammar extending; natural language understanding; domain specific; ontology

1 引言

短文本理解在很多应用中都有很急迫的需求,比如web搜索、广告匹配、智能客服等. 与长文本相比,很多场景下要处理的短文本通常不符合书面语的语法,意味着传统的NLP方法不能直接使用到短文本理解中. 另外,短文本的语境信息通常比较有限,比如在web搜索场景下,大部分的查询不超过5个词[1]. 因此,短文

收稿日期:2019-01-28;修回日期:2020-11-03;责任编辑:王天慧
基金项目:国家自然科学基金(No. 61702234);科技部重点研发项目(No. 2017YFC1700302)

本中通常缺乏一些统计信息来支持一些统计文本处理技术如主题建模等[2].

文本理解方法大致可以分为浅层语义分析方法和深层语义分析方法[3],浅层语义分析通常会标注与句子中谓词有关成份的语义角色,如施事、受事、时间和地点等,比如在检索型问答系统中,可采用该技术对句子进行标注,并利用数据源中的冗余信息,通过检索和匹配等技术来定位答案.

深层语义分析方法大致可以分为两类. 第一类方法是基于近年来发展较快的深度学习技术. 这些方法一般通过嵌入学习技术、深度学习技术等学习语料库、知识库和问句的语义表示及它们相互之间的语义映射关系[4],从而将用户的自然语言问题、知识库中的实体、概念、类别以及关系等转换为低维稠密的数值向量,并将知识库问答等任务看成是语义向量之间的相似度计算或分类过程. 基于深度学习的语义分析方法的优点是鲁棒性较强,并且由于采用端到端的学习策略,可以部分消除错误累积问题. 但目前这类方法还存在一些问题:(1)深度学习方法通常需要大量的训练语料,而对于一些领域来说(如医疗领域),在系统建设初期,获取大量的高质量训练语料仍然是个瓶颈[5];(2)已有的基于深度学习的文本理解方法多是针对简单问题,对于复杂问题的理解能力尚且不足[6];(3)在实际系统出现问题时,无法进行及时有效的人工干预[7].

第二类方法是基于符号逻辑的语义分析方法,即基于符号化的文法对用户的自然语言问句进行分析并转化成结构化的语义表示. 在这一大类方法中,一类是采用级联式的自然语言处理方法,即先对句子进行词法、句法分析,再依据句法规则与领域语义的对应关系,对句法分析结果进行语义解释. 级联式的自然语言处理方法对于短文本理解来说,通常是低效的,特别是一些口语问答系统 、社交媒体等,用户的问题通常是口语化、不规范的短文本,导致语言分析过程的前两步常常就不能产生正确的结果[8]. 另一类通过将语义附着于词汇或文法规则上,实现语法和语义一体化分析,比如语义文法(Semantic Grammar)[9]、SC 文法(Sub-Category grammar)[10]、组合范畴语法(Category Compositional Grammar,CCG)[11]、依存组合语法(Dependency-based Compositional Semantics,DCS)[12]等. 这类方法的优点是可解释性较好,产生的是一个有层次的解析结果,缺点是起到重要作用的文法一般主要由人工生成,比如 CCG 中的词汇表和规则集,在进行领域转换和扩展时,需要耗费大量的人力和时间来生成或扩展规则库,所以目前基于有监督[12](如提供自然语言问题-语义表示对,自然语言问题-答案对)或无监督[13]的自动文法学习成为了研究人员探索的重点.

通过对上述几类方法的分析发现,基于符号逻辑的语义分析方法具有分析结果层次性较丰富、可解释性较好等特点,缺点是一些文法形式过复杂或过简单,导致解析效率低或存在过生成等问题,比如,目前语义解析中采用较多的组合范畴语法 CCG[14],在某些场景下,附着于 CCG 文法规则的逻辑表达式(logic form)会非常复杂,导致解析 CCG 时搜索空间巨大[12]. 而另外一些文法形式如 FunQL,则过于简单,导致表达能力不足,存在过生成、易产生解析歧义等问题[15].

FredKarlsson 等提出的一种形式简单并可利用语境信息进行歧义消解的带约束文法[16],可有效限制文法的生成能力,针对不同的需求,可在文法规则中加入对当前匹配所对应上下文的词、词性、句法等进行约束[17],最早用于词形(Morphology)分析、句法分析等,近年来有研究者将其用于问答、对话系统等[18]. 本文在约束文法的基础上,提出一种基于本体的带约束语义文法及其理解方法,通过在文法约束中融合词汇、句法、语义(领域本体)知识以及匹配控制等,从而对文法解析过程进行多层次约束,可以有效提高文法解析的鲁棒性及降低匹配歧义问题.

2　相关工作

总体来说,文法更新及学习的主要方法包括:(1)收集不能被现有文法解析或解析错误的句子,并提交给文法设计人员进行文法更新,方法缺点是需要非常有经验的人员且需要较长的时间;(2)采用机器学习方法如贝叶斯分类器、深度学习方法等预测句子的顶层概念类型,方法缺点是产生的结果是一个无层次的分类,并不能产生一个带有嵌入变元的结构[19];(3)采用基于最小距离的解析法,对于没有解析结果的句子,从解析成功的句子集中找一个最相似的句子,方法缺点是用于查找在语法上最相似句子的算法具有指数级的时间和空间复杂性[20];(4)利用模式推断[21]或归纳学习算法自动学习文法,这类方法学习到的文法可理解性较差,所生成文法树不适合直接用于生成句子的语义表示[22];(5)利用构造的树库学习文法规则,此类方法可用于构造初始的核心文法,不适合扩展文法以使其覆盖不合语法的句子.

很多研究者试图将自动学习方法引入到设计文法的各个阶段,以提高系统的整体效率及文法的覆盖度. Rosé[23]提出一种通过交互方式处理不合语法的现象,当遇到不合语法句子且句子至少产生了两棵部分解析子树时,通过组合以及交互两个步骤构造句子的完整解析树,这一方法对交互的人员有较高的要求. Portele[24]提出了一种面向特定用户的自适应扩展核心文法的方法,针对不同的解析失败原因(包括删除、插入、

替换、位置交换等),分别制定了文法的扩展动作. 这一方法面向特定用户扩展文法,使得学习出的文法有可能不具有普适性. Wang [25]将基于语料的统计方法与基于规则的方法相结合,当对一个句子解析失败时,首先由基于规则的部件产生所有可能的关于现有文法缺陷的假设,再由基于语料的统计方法从中找出最有可能的一个,并据此对原有文法进行改造. Gil[26]首先由设计人员生成一个简单的领域模型和一个核心语法,在运行阶段,通过一个交互式界面来动态扩展核心语法. 该方法只采用与用户交互的方式进行文法扩展学习,效率较低. 研究者[27]利用包含领域中相关概念的语义模式对语料进行标注,同时将语义模式自动转换成 CFG 模板,最后依据 CFG 模板、标注的语料库、通用文法库、语法限制规则等自动扩展学习新的文法规则.

通过对相关研究工作的总结和比较,我们认为:(1)采用基于机器学习的自动文法归纳学习方法,可以大大降低人工成本,领域迁移性较好,但通常产生的是一个无层次的分类[28],学习到的文法可理解性较差[29],用其所生成文法树不适合直接用于生成句子的语义表示[30];(2)人工设计的文法可理解较好,可解释性较强,但对于大规模领域来说,手工设计文法效率较低.

因此,本文对上述两类方法进行了融合,在已建立的基于领域本体和语义文法的自然语言理解系统基础上[31],提出一种基于错误驱动的自动文法学习方法,针对不同的自然语言理解失败原因分别进行文法规则自动扩展学习,以解决人工构造文法的效率低和过生成等问题.

3 本文方法

本文提出一种基于错误驱动的文法学习方法,解析错误主要包括:(1)由于现有文法规则不全,导致无法对句子产生一棵完整解析树;(2)使用现有文法产生了错误的解析树. 根据不同的解析失败原因自动扩展文法,总体流程如图 1 所示.

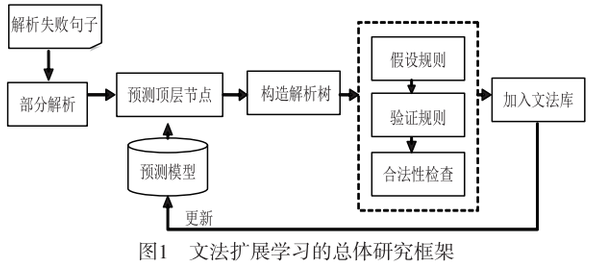

图1 文法扩展学习的总体研究框架

传统的文法解析器对句子生成一棵只有一个根节点的解析树,对于不合文法的句子,通常不能生成一棵完整的解析树. 本文首先利用核心语义文法对句子进行部分解析,尽可能多地识别出句子中的语义单元,再利用构建的预测模型预测句子的未能解析部分的上层节点,并逐步构建句子的完整解析树,在构建解析树的过程中,对引入的新规则进行合法性检查并更新文法,再对预测模型进行更新.

3.1 基于本体的带约束语义文法

本体作为语义文法的“骨架”,在文法扩展学习时,领域本体将是对现有文法进行扩展学习的重要“知识源”. 领域本体用一个有向无环图表示 $DAG=(N,E,M)$,其中:(1)N 为有向图中的节点集合,包括领域相关概念(关键概念、辅助概念)、语义文法中的非终结符等;(2)E 为节点之间的有向边的集合,$E\subseteq N\times N$;(3)M 是一个映射函数,$M:E\rightarrow SM$,其中 $SM=\{\text{ISA},\text{REQ},\text{OPT}\}$,ISA 表示 N 中节点间的上下位关系,REQ 表示 N 中节点在语义文法中的“总是必须”关系,OPT 表示 N 中节点在语义文法中的“总是可选”关系.

定义 1 文法约束 文法约束(constraints)是一种逻辑表达式. 为了便于文法分析和文法学习,本文采用一种有限的约束形式,并采用析取范式(DNF)表示. 文法约束可以包括词汇级(词汇依存)、语义级(语义依存)约束等,可以弥补传统 PCFG 的上下文无关的不足. 典型的文法约束谓词包括:

(1)匹配控制约束,表示规则匹配有序或无序:Control(< *s*0 > , < *control-str* >).

(2)词汇或语义约束,表示匹配成分语境的词汇或本体类别约束,比如 Followed-by(< *s*0 > , < *s*1 >).

(3)语义约束,表示对匹配成分的本体类别约束,比如 ISA(< *s*0 > , < *s*1 >).

定义 2 基于本体的带约束产生式规则 带约束产生式规则表示形式如下:

< production > : : = < head > → < body > [< constraints >]

< head > : : = < non-terminal >

< body > : : = < non-terminal >
| < terminal >
| < non-terminal > < body >
| < terminal > < body >
| [< non-terminal >] < body >

< non-terminal > : : = < semantic class >
| < Intent-non-terminal >
| < basic non-terminal >
| < ANY >

< semantic class > : : = < Principal-concept >
| < Auxiliary-concept >

< Intent-non-terminal > : : = ‘BUY-TICKETS’
| ‘QUERY-FOR-TRAFFIC’
| …

在上述定义中,引入了通配型非终结符(< ANY >)以提高语义文法的鲁棒性,为了避免“过匹配”问题,引入的文法约束可以对通配型非终结符的匹配进行“约束”,检查通配符的匹配是否满足约束. 文法约束还可以对其他的一些基本型非终结符等进行限制.

定义 3　带约束语义文法　语义文法 G 为一个 4 元组 $G=(VT,VN,S,R)$,其中:(1) VT 为语义文法中的终结符;(2) VNT 表示语义文法中的非终结符. 包括基本型非终结符集、通配型非终结符、本体中的关键概念、辅助概念等;(3) S 表示语义文法的开始符集合 $S=\{S1,\cdots,Sn\}$. 开始符与领域本体中的查询意图非终结符相对应(< Intent-non-terminal >);(4) R 为语义文法中的带约束产生式集合.

语义文法的构建主要包括两个部分:(1)本体构建,构建领域相关的概念及其相互关系,本体是语义文法的骨架;(2)核心语义文法规则设计,将一些非终结符或终结符相组合得到文法模式.

下面给出一个语义文法示例(如表 1 所示),此语义文法主要用于处理用户提交的关于购票的查询句子. 在下文的语义文法扩展学习中,将使用此语义文法进行举例. 在文法中,带有前缀“_”的非终结符表示是领域的一些关键概念,也称“语义概念槽”,如“_WEEKLY_DATE”、“_CITY”等,语义抽取模块将以此为标识生成句子的语义表示.

表 1　语义文法规则举例

No.	语义文法规则
1	< QUERY_FOR_BUY_TICKET > →[< HOW >][< BUY >] < DATE_TIME > < ROUTE > < TICKET >
2	< QUERY_FOR_BUY_TICKET > → < DATE_TIME > < ROUTE > < FLIGHT > [< QUERY_WHAT >]
3	< DATE_TIME > → < _WEEKLY_DATE >
4	< ROUTE > → < DEPARTURE_INFO > < ARRIVAL_INFO >
5	< DEPARTURE_INFO > →从 < _CITY > @Control('ordered')
6	< ARRIVAL_INFO > →[到] < _CITY > @Control('ordered')
7	< HOW > →怎么|怎样|如何
8	< BUY > →买|购|购买 |订购
9	< _WEEKLY_DATE > →周一|周二|周三|周四|周五|周六|周日
10	< _CITY > →北京|上海|深圳
11	< FLIGHT > →航班|航线
12	< TICKET > →机票|飞机票
13	< QUERY_WHAT > →有哪些|有什么|哪些|…
14	< HAVE > →有|拥有|包括|…

图 2 是句子“怎么购买周五从北京到深圳机票?”利用上述语义文法得到的解析结果.

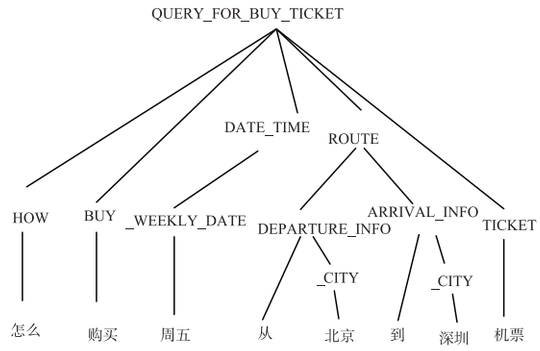

图2　语义文法分析结果示例

3.2　基于核心文法的部分解析

基于核心文法的部分解析建立在已有工作的基础上,即基于带约束语义文法的 STM 文法分析算法[24],通过引入文法约束检查及解析树评分机制,依据语义文法对句子进行解析,从中选择得分最高的解析树作为结果. 为了支持文法规则扩展学习,本文对不合文法的句子采用部分解析模式,即文法规则中的所有非终结符都可以作为树的根节点,一个句子可能产生多组部分解析结果,每组解析结果由多棵部分解析子树组成. 例如一个句子为:

请问周五从北京到深圳飞机有哪些?

在解析器的精确解析模式下,由于规则不全,不能对上述句子生成一棵完整的解析树. 启用部分解析模式,得到部分解析结果如图 3 所示.

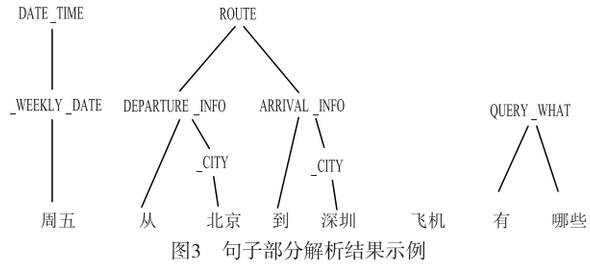

图3　句子部分解析结果示例

部分解析模式可能生成多组解析结果,每一组由子树序列和未能解析的词构成. 本文设计了部分解析结果的打分模型,模型中考虑了以下特征.

特征 1　部分解析树对句子的覆盖度(*PCoverage*)　部分解析树对句子的覆盖度是指解析树所包含的词数与句子中所有词的个数之比. 部分解析树对句子的覆盖度越大,越优先.

特征 2　部分解析分支数(*NFrage*)　一组部分解析结果中,所有子树的数目. 一组部分解析树中的分支数越少,越优先.

特征 3　解析树中使用的通配符数(*NAny*)　通配符数越少,越优先.

部分解析的打分函数为:

$$XMatchScore = w_1 \times MatchScore_{STM} + w_2 \times PCoverage + w_3 \times NFrage + w_4 \times NAny \quad (1)$$

其中$MatchScore_{STM}$表示由文献[31]提出的打分函数得到的分值. 权重w_i表示各个特征的权重. 权重采用 Percy Liang 等提出的有监督随机梯度下降算法 SGD[34]训练得到. 训练算法中的训练数据 $<x,y>$,其中 x 表示解析成功的句子,y 表示该句子所对应的解析树及其子树集合,具体训练算法,请参见相关论文,不再赘述.

3.3 预测模型构建

有两类依赖关系需要使用预测模型来刻画,对应的模型分别称为横向预测模型和纵向预测模型,本文使用 N-Gram 语言模型构建预测模型.

(1)纵向预测模型用来刻画解析树中节点间的嵌套关系. 比如,DEPARTURE_INFO 倾向于作为 ROUTE 的子节点,而 ROUTE 倾向于作为 QUERY_FOR_BUY_TICKET 的子节点. 预测模型的词典包含树库中所有的终结符和非终结符. 事件包括从树库的解析树中抽取的自底向上的所有路径.

(2)横向预测模型用来刻画解析树中节点间的邻接关系. 比如,在作为 ROUTE 的子节点时,ARRIVAL_INFO 总是与 DEPATURE_INFO 邻接. 每一个非终结符对应一个横向预测模型. 横向预测模型的词典为文法中的所有终结符和非终结符,事件为从树库中抽取出的给定非终结符下的子节点有序序列.

预测模型使用树库进行训练,树库通过以下两种方式构建:(a)利用核心语义文法的生成能力,产生树库;(b)在与用户的交互过程中,解析正确的句子,以及初始解析失败,但经过文法学习后,解析成功的句子,记录下这些句子所对应的解析树,形成用户树库. 基于抽取出的事件库分别训练出 N-Gram 模型.

具体来说,对于给定的部分解析结果序列 $e = <e_1, e_2, \cdots, e_m>$ $e = <e_1, e_2, \cdots, e_m>$($e_i$为终结符或解析子树),每一个问题本体中的概念(即顶层节点)使用纵向预测模型可计算出一个分值,使用向量$\boldsymbol{V}_H$表示,定义如下[32]:

$$P_{Hypo}(NT_i \mid e) = \sum_{j=1}^{m} P_{Hypo}(NT_i \mid e_j) \quad (2)$$
$$NT_i \in \{\text{concepts in Ontology}\}$$

其中,$NT_i \in \{\text{conceptsinOntology}\}$(下同),根据横向预测模型,每一个问题本体中的概念(即顶层节点)也可计算出一个分值,定义如下:

$$P_{Para}(NT_i \mid e) = \sum_{j=1}^{m} P_{Para,NT_i}(r(e_j) \mid r(e_{j-(n-1)}) \cdots r(e_{j-1})) \quad (3)$$
$$NT_i \in \{\text{concepts in Ontology}\}$$

其中,

$$r(e_j) = \begin{cases} e_j, & e_j \text{ is a terminal} \\ root(e_j), & e_j \text{ is a tree} \end{cases} \quad (4)$$

上述公式的含义是利用每一个NT_iNT_i的横向预测模型,计算窗口大小为 k(实验表明,窗口大小 k 取值为 4 为最佳)由终结符和子树组成的子序列概率,将这些概率累加得到子节点为$e_{j-(k-1)}, \cdots, e_{j-1}e_{j-(k-1)}, \cdots e_{j-1}$, $1 \leqslant j \leqslant k$,父节点为$NT_i$的概率. 类似可求得所有$NT_i$的概率,并使用向量$\boldsymbol{V}_p$表示.

最终,可融合上述两种模型的预测结果,定义如下:

$$v(i) = \lambda \cdot v_H[i] + (1-\lambda) \cdot v_P[i]$$
$$\forall i: 1 \leqslant i \leqslant |\{\text{concepts in Ontology}\}| \quad (5)$$

其中,参数 $0 \leqslant \gamma \leqslant 1$ 通过有监督随机梯度下降算法[33]训练得到,$0 \leqslant \gamma \leqslant 1$ 训练算法中的训练数据 $<x,y>$,其中 x 表示解析成功的句子,y 表示该句子所对应的解析树的顶层概念节点.

3.4 解析树重构

在上述算法中,利用部分解析结果及预测模型,预测最有可能的顶层概念节点. 在选择待扩展规则时,本文采用层次式策略进行选择:

(I)若某规则的 RHS(Right-Hand-Side,规则的结论部分,下同)中存在一个或多个关键概念与部分解析结果中的一个或多个概念相匹配,那么优先选择该规则作为待扩展规则.

(II)若某规则的 RHS 中存在一个或多个动词性非终结符与部分解析结果中的一个或多个动词性节点相匹配,那么优先选择该规则作为待扩展规则.

(III)规则的 RHS 中的非终结符或终结符与部分解析结果匹配的数量越多,那么优先选择该规则作为待扩展规则.

构建解析树的过程包括自上而下和自下而上两个子过程,主要包括:

(I)自上而下过程. 依据建立的领域本体及其它信息,对于句子中未能解析的部分,为父节点选择合适的子节点以覆盖这些部分.

(II)自下而上过程. 利用未能解析部分的语境信息、预测模型及领域本体,预测可能覆盖此部分的中间节点.

在构建解析树时,主要包括以下两种类型规则的扩展.

定义 4 Pre-terminal 型规则 指 RHS 为具体的词条,LHS(Left-Hand-Side,规则的条件部分,下同)为语义词类所对应的非终结符的规则.

定义 5 Non-terminal 型规则 指 RHS 由非终结符、终结符等组成,LHS 为非 pre-terminal 的非终结符的规则.

扩展 pre-terminal 型规则的时机主要包括:

(1)核心文法库中某条规则的 RHS 中存在某个 pre-terminal 型非终结符在句子中没有找到匹配部分,且句子中存在 OOV 词(集外词).

(2)核心文法库中某条规则的 RHS 需要插入一个已有的 pre-terminal 型非终结符,且句子中存在 OOV 词.

在扩展 pre-terminal 型规则时,采用以下启发式策略:

(I) 对于概念节点,优先匹配名词或名词性实体,名词性实体可由 NP chunker 标注得到.

(II)对于动词性非终结符,优先匹配动词. 若未匹配部分有动词,且存在动词性非终结符未找到匹配成分,则动词性非终结符优先匹配动词.

(III)对于其他非终结符,也按照词性进行匹配. 如对于介词性非终结符、形容词性非终结符等分别匹配具有相同性质的词或短语.

扩展 Non-terminal 型规则的时机包括:

(1)若句子中有成分未能建立到顶层概念节点的连通路径,且解析树中已扩展的规则的 RHS 中不存在覆盖这些成分的非终结符,则需在解析树中的某条规则的 RHS 中插入非终结符,以便能够覆盖未匹配的部分.

(2)若某条引入的规则的 RHS 还存在必选成分没能匹配句子中的成分,则需扩展此条规则:重新设置原有规则中成分的性质(可选/必选),在扩展的规则中将此成分设置为可选.

在已有规则的 RHS 中插入非终结符时,主要采用有以下策略:

(I)必须原则,若某个非终结符与当前考察的规则的 LHS 在领域本体中的关系为"总是必须",且在当前所考察的规则中没有出现该非终结符,则将该非终结符插入规则中. 这样的非终结符可能有多个,多个候选非终结符对应着规则的多种扩展方式.

(II)动词原则,即若未匹配成分为动词,则选择具有动词性质的非终结符插入.

(III)非必须原则,即若某个非终结符与当前考察的规则的 LHS 的在领域本体中的关系为非"总是可选",且在当前所考察的规则中没有出现该非终结符,则将该非终结符插入规则中.

(IV)利用语境预测,即利用未匹配成分周围已匹配的信息预测可能的非终结符(横向预测模型),将最有可能的非终结符插入到当前考察的规则的 RHS 中.

(V)总是可选原则,若某个非终结符与当前考察的规则的 LHS 的在领域本体中的关系为"总是可选",且在当前所考察的规则中没有出现该非终结符,则将该非终结符插入规则中.

上述几种策略的优先级从高到低,在选择插入非终结符时,当高优先级的策略使用失败时,才考虑下一种策略.

3.5　解析树寻优及规则扩展

对于每一棵候选的解析树,须经过以下两个步骤:(1)产生规则假设集(Generating hypothesis),即在重建解析树过程中新扩展的规则,形成规则假设集合,集合中主要包括 pre-terminal 类型规则和 non-terminal 类型规则;(2)验证规则假设集(Verifying hypothesis),即对规则假设集合中的所有规则进行验证,若规则集中的任一条规则验证失败,则认为整棵树构建不合理,验证失败. 本文针对两种类型的规则分别采用不同的验证方法.

3.5.1　Pre-terminal 型规则的验证方法

具有相同 LHS 的 pre-terminal 型规则的 RHS 通常是一些同义词,这些词具有相同或相似的领域含义. 引入相似性函数计算一个词与一个词的集合(已存在的同义词类下的词)的相似程度. 若不相似,则新扩展的规则无效,反之,则认为验证成功. 一个词与一个词集合的相似性定义如下.

$$\mathrm{sim}(W,w_t) = \sum_{w_i \in W} \mathrm{f}(\mathrm{sim}(w_i,w_t))/|W| \tag{6}$$

其中:

$$\mathrm{f}(t) = \begin{cases} 0, & t > threshold \\ 1, & t \geqslant threshold \end{cases} \tag{7}$$

其中,W 为已有的同义词集合,w_t 为新增规则的 RHS. 相似度度量方法可基于词的语境分布,计算词之间的 KL 距离(距离越小,相似程度越高),如式(8):

$$\mathrm{D}(p_1 \| p_2) = \sum_{i=1}^{V} p_1(i) \log \frac{p_1(i)}{p_2(i)} \tag{8}$$

其中,p_1,p_2 分别对应着两个词的语境概率分布. 为了使得距离对于两个词具有对称性,修改距离公式为:

$$\mathrm{Div}(p_1 \| p_2) = \mathrm{D}(p_1 \| p_2) + \mathrm{D}(p_2 \| p_1) \tag{9}$$

语义相似度定义如下:

$$\mathrm{sim}(e_1,e_2) = 1/\mathrm{Div}(e_1 \| e_2) \tag{10}$$

根据语义相似度式(10),计算词项之间的语义相似度.

3.5.2　Non-terminal 型规则的验证方法

对于规则 $L \rightarrow R$,主要验证学习到的规则是否冗余或歧义,关于冗余和歧义的定义如下.

定义 6　文法歧义　假设新加入的规则为 $L \rightarrow R$,由 R 生成的子语言为:Gen(R),若对于任何 s 属于 Gen(R),s 可以由现有文法解析,假设解析的文法形式为:$L1 \rightarrow R1$,则认为新加入的文法引入了歧义.

定义 7　文法冗余　假设新加入的规则为 $L \rightarrow RL \rightarrow$

R,由 R 生成的子语言为:Gen(R),对于现有文法库中的任意规则 $L1\rightarrow R1L1\rightarrow R1$,若 Gen($R$)属于 Gen($R1$),并且 $L1=LL1=L$,则认为新加入的规则是冗余的.

定义 8 扩展集 exp(R) 扩展集 exp(R)是定义在$(\sum\cup V)^*$上的子语言,即不须将 R 中的非终结重写为终结符,只需要根据规则的约束,生成的所有符号串由非终结符或终结符组成.

在下文用 P(s,L) 表示符号串 s 可以由 LHS 为 L 的规则匹配.

歧义检测:对于 R 的所有扩展集 exp(R),利用现有文法对其使用解析器的文法检查解析模式进行解析,若对于任何 r 属于 exp(R),可以由现有文法解析,假设解析的文法形式为:$L1\rightarrow R1$,若 $L1\neq L$,则认为新加入的文法引入了歧义. 这一方法的优点是 exp(R)的规模较 Gen(R)小很多,提高了效率.

冗余检测:对于 R 的所有扩展集 exp(R),利用现有文法对其使用解析器的文法检查解析模式进行解析,若对于所有 r 属于 exp(R),都可以由现有文法解析,假设解析的文法形式为:$L1\rightarrow R1$,若 $L1=L$,则认为新加入的文法引入了冗余.

Non-terminal 型规则的验证算法如算法 1 所示.

算法 1 Validation algorithm of Non-terminal rules

输入:Rule R;
输出:*FLAG*:Redundant | AMbiguous | OK;
PASS:True | False;
Begin
(1) Generate exp(R);
(2) Set grammar-check mode, and exp(R) as input;
(3) For every $r\in$ exp(R)
If (Exist $L\in G$, $P(r,L1)$ and $L1==L$)
{
FLAG = Redundant; *PASS* = False;
}
(4) If (Exist $r\in$ exp(R), exist $L\in G$, $P(r,L1)$ and $L1!=L$)
{
FLAG = Ambiguous; *PASS* = False;
}
(5) *FLAG* = OK; *PASS* = True;
(6) Return *FLAG*, *PASS*;
(9) Return *ListIntentions*; *ListSemRep*;
End.

3.6 基于可学习性度量的文法扩展批量学习算法

本文采用可学习性度量对所有文法分析失败的句子进行可学习性评价以提高学习效率,得到一个按照句子的评分高低进行排序的队列,并依次作为文法扩展学习的输入,直到满足系统设定的学习目标或条件.

定义 9 可学习性 对句子作为学习对象的一种度量,可学习性高的句子能够以较高的效率学习到新文法规则,反之,可学习性低的句子一般用来学习文法规则的效率较低,句子的可学习性基于以下特征进行计算.

特征 1 句子的复杂性

一个句子越复杂,基于这样的句子进行文法扩展学习时,学习到的规则将越不具有一般性. 句子的复杂性函数$f_{\text{complex}}(s,G)$ 定义如下为:

$$f_{\text{complex}}(s,G)=\text{length}(s)+\text{branch}(s,G) \tag{11}$$

其中,length(s)表示句子的长度,branch(s,G)表示利用核心文法 G 对句子 s 进行部分解析时,具有最高得分的部分解析树的分支数目.

特征 2 预测顶层节点的概率

利用部分解析结果预测的顶层概念节点的最大概率值可作为句子可学习性的依据,概率值越小,说明系统对于预测的概念节点可信度越低,基于此扩展的规则的可信度也越低,用下式表示:

$$\text{P}_{\text{top}}(s,G)=\text{MAX}\{\text{p}(root)\mid root\in predictlist_{s,G}\} \tag{12}$$

其中,$predlist_{s,G}$表示系统依据当前的核心文法 G 预测的句子 s 的所有顶层概念节点集合,$p(root)$表示概念节点 *root* 的预测概率值.

特征 3 未登录词数

若句子的未登录词数越多,对于句子的理解难度越大,用式(13)来表示:

$$\text{N}_{\text{oov}}(s,G)=\sum_{w\in s}\text{OOV}(w,G) \tag{13}$$

其中,当词 w 是一个集外词时,OOV(w,G) = 1,反之,OOV(w,G) = 0.

特征 4 句子部分解析树分值(XMatchScore)

详细定义见 3.2 节.

基于以上四个特征,评价一个句子的可学习性的函数$\text{f}_{\text{learnable}}(s,G)$定义如下:

$$\begin{aligned}\text{f}_{\text{learnable}}(s,G)=&w1\times\text{f}_{\text{complex}}(s)\\&+w2\times\text{P}_{\text{top}}(s,G)\\&+w3\times\text{N}_{\text{oov}}(s,G)\\&+w4\times\text{XMatchScore}(s,G)\end{aligned} \tag{14}$$

权重参数通过有监督随机梯度下降算法[33]训练得到,$0\leqslant\gamma\leqslant1$ 训练算法中的训练数据 $<x,y>$,$0\leqslant\gamma\leqslant1$ 其中 x 表示句子,y 表示由人工标注的可学习性高低,按照 1 ~ 5 分进行标注,1 表示可学习性最低,5 表示最高.

基于可学习性的文法扩展批量学习算法如算法 2 所示.

算法 2 Grammar Batch-Extending algorithm (GBE)

输入:U:使用现有核心文法分析失败的句子集合;

L:经人工标注的标准测试集;
G:系统的核心文法
输出:扩展后的文法 G;
Begin
(1) Repeat Times = 1000;　　//学习迭代次数阈值
(2) Repeat:
(3)　N← select(U,G,flearnable);　//选择"可学习性"最好的句子
(4)　G1←SGE(N,G);　　//文法扩展学习 SGE
(5)　U←U - N;
(6)　G←G + G1;
(7)　*RepeatTimes* - -;
(8) Until *TEST* (L,G) > *THRESHOLD* or (U is empty) or (*RepeatTimes* <0);
(9) Return G;
End.

批处理的方式优点是不需要人工的参与,学习效率较高,但带来的结果是会损失一部分准确性. 在实际应用中,为了确保更新后的文法库的可靠性,可以再进行一些后处理操作,如可将新添加的规则交由人工检查并确认.

4　实验

本文使用两种范式(交互式和批量式测试范式)在两个应用领域进行测试. 在交互式测试中,让 10 个用户分别与系统进行交互,在交互过程中记录相关日志数据. 在 3.2 节中,在确定一组部分解析结果后,应用训练好的预测模型,得到按可能性排序的根节点列表,若根节点的置信度大于设定阈值,则不与用户进行交互,否则,将列表展示给用户,由用户来选择其一作为最佳的根节点. 在批量学习范式下,对语料中的例子进行可学习性度量,选择可学习性较高的句子进行文法扩展学习,并比较核心语义文法和扩展后的语义文法在测试集上的理解性能.

4.1　实验数据描述

用于文法扩展学习的实验数据集(各包含 10w 条用户咨询)包括如下.

数据集 1:BSC Data Set,数据集中的句子是关于某个银行的产品或业务的咨询,比如关于如何办理信用卡或汇款手续费等.

数据集 2:MSC Data Set,数据集中的句子是关于某个通信公司的产品或业务的咨询,比如关于手机归属地查询或办理通信套餐业务等.

在文法扩展学习时,随机抽取其中的一半作为训练集,另一半作为测试集(T1). 同时,为了验证文法扩展学习得到文法的整体精度,将全部数据集作为测试集(T).

4.2　测试指标

在交互式学习模式下,主要从下面的几个指标来考察系统的学习行为. (1)输入的总句子数(*NQuery*):用户提交到系统的总查询数;(2)学习次数(*NLearning*):用户提交到系统的查询句子,激起系统学习的次数;(3)用户与系统平均交互次数(*NInteraction*):此数目越小,说明系统的自学习能力越强;(4)平均给用户的选项数(*NChoice*):此数目越少,说明系统的自学习能力越强;(5)学习到的规则数(*NRules*):学习到的新规则数.

在批量学习测试中,用于评价基于语义文法的 NLU 系统性能的指标包括:准确率、MRR、识别率,这些指标的定义详见文献[31].

4.3　测试结果

(1)交互式测试范式

表 2 至表 3 分别为方法应用到 MSC 和 BSC 领域,十个用户与系统的交互的相关数据统计. 在 MSC 领域共学习到了 52 个新规则(其中有 4 个重复的规则). 在 BSC 领域共学习到了 38 个新规则(其中有 9 个重复的规则). 另外,学习到的规则数目与学习次数、交互次数等没有直接关系,其原因是可能一次学习过程不会学习到任何新的规则,也有可能一次学习过程学习到多条新规则.

表 2　通信业务信息咨询领域用户交互数据统计

	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10	Total
NQuery	17	13	11	15	21	19	16	25	9	13	159
NLearning	4	3	3	5	9	6	5	8	3	4	50
NInteraction	3	2	2	3	4	2	2	5	2	2	3.0
NChoice	5.3	4.0	3.2	4.30	5.2	4.5	3.2	4.4	4.1	4.2	4.4
NRules	3	4	2	7	8	6	8	6	3	5	52

由前述可知,MSC 领域相比于 BSC 领域来说,领域概念较多,概念的属性及关系较复杂,领域规模较大,初始的核心文法较难完全覆盖用户咨询所表达的概念的属性或语义概念间的关系. 所以,应用于 MSC 领域,在相近数目的用户查询语句时,学习次数较多. 由于领域规模较大,与用户交互时,呈现给用户的选项数也较

多. 而在较小的应用领域,由于领域概念较少,概念的属性或概念间的关系较简单,初始的核心文法相对较容易总结全面,在文法扩展学习时,与用户交互次数较少,系统具有较强的自学习能力. 以上分析说明,本文所提出的文法扩展学习的效率、学习质量与初始的核心文法是密切相关的.

表 3 银行业务信息咨询领域用户交互数据统计

	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10	Total
NQuery	17	21	15	10	13	12	18	17	14	16	153
NLearning	5	4	4	3	2	3	6	4	3	4	38
NInteraction	5	3	2	1	1	2	3	2	1	1	2.3
NChoice	4.3	3.8	4	3	4	3.5	4.2	2	3.0	2.0	3.4
NRules	3	5	3	4	3	2	8	3	4	3	38

(2)批量式测试范式

对于两个领域的数据集,在更新文法之前,先使用核心文法对数据集进行测试,以与更新后的文法的测试结果进行比较. 本文提出的解析算法采用 C++语言编写,并且可以运行于 windows 平台和 Unix 平台,表 4 列出了所测试的两个领域的初始核心文法的统计数据. 测试机的配置为 Pentium IV 3.2GHz,测试语料的平均词数为 8.5 个,最大长度为 25 个词,其中,MSC 领域的平均解析时长为 312.83ms,最长处理时长为 1321ms,BSC 领域的平均解析时间为 23.2ms,最长解析时长为 52ms.

表 4 核心文法统计数据及解析效率

Index\Domain	MSC Grammar	BSC Grammar
Non-terminals	6823	582
Terminals	9540	2230
Rules	12961	3520
Intent-Non-terminals	520	132
Average parsing time	312.83ms	23.2ms
Maximal parsing time	1321ms	52ms

表 5 和表 6 列出了更新后的文法以及核心文法在两个测试集上的测试结果.

从表 5 和表 6 可以看出,更新后的文法在测试语料上的相关指标均取得了一定提高,特别是在规模较大的 MSC 领域,更新后的文法能够较大程度地提高文法对领域概念及关系的覆盖度. 但是由于领域概念较多,关系较复杂,核心文法本身就具有较大的歧义性,扩展后的文法也具有较大的歧义,所以扩展后的文法对测试问题集 T1 和 T 的准确率只提升了 1.1%和 1.5%. 而对于在规模较小的 BSC 领域,初始的核心文法本身对领域就具有较高的覆盖度(表现为具有较高的识别率),对测试语料的识别率和MRR值提升幅度较小. 而由于领域概念较少,关系较简单,所以文法规则本身具有较小的歧义,经文法扩展学习,更新后的文法能够较大的提升文法对测试问题集 T1 和 T 的准确率分别提高了 2.3%和 2.6%.

表 5 MSC Data Set 测试结果

性能指标	核心文法/%(T1)	更新后的文法/%(T1)	核心文法/%(T)	更新后的文法/%(T)
准确率	81.5	82.6	81.3	82.8
MRR	90.2	92.3	90.3	92.6
识别率	91.9	94.4	91.7	94.8

表 6 BSC Data Set 测试结果

性能指标	核心文法/%(T1)	更新后的文法/%(T1)	核心文法/%(T)	更新后的文法/%(T)
准确率	85.3	87.6	85.2	87.8
MRR	92.7	93.5	92.5	93.6
识别率	94.1	95.6	93.7	95.8

对批量文法学习算法中的可学习性度量进行了测试. 比较测试了不采用可学习性度量(baseline)、以及在可学习性度量中采用不同的特征组合对语义理解的影响. 表 7 和表 8 分别给出了在两个不同规模数据集上的准确率、识别率和 MRR 值. 其中f_{complex}表示句子复杂性(f_1),P_{top}表示预测顶层节点的概率(f_2),N_{oov}表示未登录词数(f_3),XMatchScore 表示句子部分解析树分值(f_4). 从表中可以看出,采用句子复杂性特征的可学习度量相对于不采用可学习度量选择学习对象时,识别率具有较大的提升. 在可学习性度量中逐一增加特征时,准确率、识别率和 MRR 值均有提高,说明采用基于可学习性度量的文法扩展学习方法学习到的文法能够较准确地刻画领域语义,同时具有较高的领域覆盖度.

表 7　多个特征对比结果(MSC 数据集)

特征\指标	Accuracy(T1)	MRR(T1)	识别率(T1)	Accuracy(T)	MRR(T)	识别率(T)
Baseline	78.1%	87.2%	89.3%	78.6%	88.1%	90.1%
f_1	78.4%	88.7%	92.4%	79.6%	89.4%	93.3%
f_1+f_2	80.5%	90.4%	93.2%	80.7%	90.7%	93.8%
$f_1+f_2+f_3$	81.4%	91.3%	93.9%	81.8%	91.8%	94.1%
$f_1+f_2+f_3+f_4$	82.6%	92.3%	94.4%	82.8%	92.6%	94.8%

表 8 多个特征对比结果(BSC 数据集)

特征\指标	Accuracy(T1)	MRR(T1)	识别率(T1)	Accuracy(T)	MRR(T)	识别率(T)
Baseline	83.8%	89.2%	91.8%	84.3%	89.4%	91.2%
f_1	84.9%	91.1%	93.2%	85.1%	90.7%	94.1%
f_1+f_2	85.6%	91.6%	94.1%	85.7%	91.8%	94.6%
$f_1+f_2+f_3$	86.5%	92.6%	95.2%	86.3%	92.1%	95.1%
$f_1+f_2+f_3+f_4$	87.6%	93.5%	95.6%	87.8%	93.6%	95.8%

4.4　实验分析

通过上述实验结果分析,本文提出的文法规则自动扩展学习方法主要存在以下问题和不足:(1)可以考虑更多的特征对句子进行“可学习性”的度量以排除超出领域范围的句子;(2)在解析过程中,对跳过的句子成分进行“重要性”检测,包括从词法、句法、语义角度来对跳过的成分进行打分,若跳过的成分比较重要,则可判定跳过这些成分生成的解析树无效或分值降低;(3)可以考虑增加全新规则的阈值条件以避免语意偏移问题;(4)可以考虑先对句子进行语义块识别,以避免在分词阶段将一些有意义的成分拆分开;(5)可以考虑将“重要性”高的词加入到文法中以提高规则覆盖度;(6)考虑对文法约束进行学习以提高文法规则的准确性以及降低人工干预程度.

5　结束语

本文研究了一种基于错误驱动的语义文法规则自动扩展学习方法,首先通过核心文法对解析失败句子进行部分解析,在此基础上,方法试图构建句子的完整解析树,包括预测部分解析结果的顶层节点、生成新扩展文法规则假设、验证假设等,并对扩展学习到的文法规则进行冗余检测等. 提出了通过句子的可学习性度量来筛选学习对象,以提高文法扩展学习的整体质量和效率. 分别测试对比了更新后的文法和核心文法在两个领域数据集上的相关性能指标,试验结果表明,本文所提出的方法是有效的.

参考文献

[1] Hua W, Wang Z, Wang H, et al. Short text understanding through lexical-semantic analysis[A]. 2015 IEEE 31st International Conference on Data Engineering (ICDE). Piscataway[C]. NJ:IEEE,2015. 495 – 506.

[2] Khanam S A, Liu F, Chen Y P P. Comprehensive structured knowledge base system construction with natural language presentation[J]. Human-centric Computing and Information Sciences, 2019, 9(1): 23 – 30.

[3] 周国栋,李军辉. 中文信息处理发展报告(2016)[R]. 北京:中国中文信息学会,2016,15 – 16.

[4] Kunneman F, Ferreira T C, Krahmer E, et al. Question similarity in community question answering: A systematic exploration of preprocessing methods and models[A]. Proceedings of the International Conference on Recent Advances in Natural Language Processing[C]. Varna, Bulgaria: INCOMA Ltd., 2019. 593 – 601.

[5] Choi E, Bahadori M T, Song L, et al. GRAM: Graph-based attention model for healthcare representation learning[J]. arXiv Preprint, 2016, arXiv: 1611.07012.

[6] Cui W, Xiao Y, Wang H, et al. KBQA: learning question answering over QA corpora and knowledge bases[J]. arXiv Preprint, 2019, arXiv: 1903.02419.

[7] Ferrone L, Zanzotto F M. Symbolic, Distributed and distributional representations for natural language processing in the era of deep learning: a survey[J]. arXiv Preprint, 2017, arXiv: 1702.00764.

[8] Fernández A M. Closed-domain Natural Language Approaches: Methods and Applications[M]. España, ES: Editorial de la Universidad de Granada, 2014, 19 – 23.

[9] R. R. Burton. Semantic Grammar: An Engineering Technique for Constructing Natural Language Understanding Systems[R]. Cambridge, Mass, UK: BBN Report #3453, Bolt, Beranek, and Newman, 1976.

[10] 陈肇雄. SC 文法功能体系[J]. 计算机学报, 1992, 15

(11):801 - 808.
Chen Zhaoxiong. A new context-sensitive subcategory (SC) grammar for machine translation[J]. Chinese Journal of Computers,1992,15(11):801 - 808. (in Chinese)
[11] Steedman M. 14 Combinatory categorial grammar[J]. Current Approaches to Syntax: A Comparative Handbook,2019,3:389 - 416.
[12] Liang P,Jordan M I,Klein D. Learning dependency-based compositionalsemantics[A]. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics[C]. Portland, Oregon, USA: Association for Computational Linguistics,2011. 590 - 599.
[13] Titov I,Klementiev A. A Bayesian model for unsupervised semantic parsing[A]. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics[C]. Portland, Oregon, USA: Association for Computational Linguistics,2011. 1445 - 1455.
[14] P. Liang,Learning executable semantic parsers for natural languageunderstanding[J]. Commun ACM,2016,59(9): 68 - 76.
[15] Kate R J,Wong Y W,Mooney R J. Learning to transform natural to formal languages[A]. Proceedings of the National Conference on Artificial Intelligence[C]. Pittsburgh, Pennsylvania, USA: AAAI Press, 2005. 1062 - 1068.
[16] Karlsson F. Constraint grammar as a framework for parsing running text[A]. Proceedings of the 13th Conference on Computational Linguistics[C]. Finland: Association for Computational Linguistics,1990. 168 - 173.
[17] Bick E, Didriksen T. CG-3—beyond classical constraint grammar[A]. Proceedings of the 20th Nordic Conference of Computational Linguistics (NODALIDA 2015)[C]. Vilnius,Lithuania:Linköping University Electronic Press, 2015. 31 - 39.
[18] L. Antonsen, S. Huhmarniemi, T. Trosterud. Constraint grammar in dialogue systems[A]. Proceedings of the NODALIDA 2009 Workshop Constraint Grammar and Robust Parsing[C]. Odense,Denmark:NEALT,2009. 13 - 14.
[19] Mitchell T, Cohen W, Hruschka E, et al. Never-ending learning[J]. Communications of the ACM,2018,61(5): 103 - 115.
[20] Ibañez R, Soria Á, Teyseyre A, et al. Approximate string matching: A lightweight approach to recognize gestures with Kinect[J]. Pattern Recognition,2017,62:73 - 86.
[21] 郑黎晓,王成. XML 模式推断研究综述[J]. 电子学报,2016,44(2):461 - 471.
Zheng Xiao-li,Wang Cheng. Schema inference from XML data:a review[J]. Acta Electronica Sinica,2016,44(2): 461 - 471. (in Chinese)
[22] Dahl V,Tessaris S,Bispo M D S. Parsing as semantically guided constraint solving: the role of ontologies[J]. Annals of Mathematics and Artificial Intelligence,2018,82(1 - 3):1 - 25.
[23] Rosé C P. A framework for robust semantic interpretation[A]. Proceedings of the 1st North American Chapter of the Association for Computational Linguistics Conference[C]. Minneapolis, Minnesota: Association for Computational Linguistics,2000. 311 - 318.
[24] Portele T. Data-driven classification of linguistic styles in spoken dialogues[A]. Proceedings of the 19th International Conference on Computational Linguistics[C]. Taipei, China: Association for Computational Linguistics, 2002. 1 - 7.
[25] Wang Y Y,Acero A,Chelba C,et al. Combination of statistical and rule-based approaches for spoken language understanding[A]. Proceedings of Seventh International Conference on Spoken Language Processing[C]. Valletta,Malta:European Language Resources Association (ELRA),2002. 609 - 612.
[26] Gil Y. Human tutorial instruction in the raw[J]. ACM Transactions on Interactive Intelligent Systems (TiiS), 2015,5(1):2 - 8.
[27] Wang Y Y,Deng L,Acero A. Semantic frame-based spoken language understanding[J]. Spoken Language Understanding: Systems for Extracting Semantic Information from Speech,2011,22(5):41 - 91.
[28] Lake B M,Ullman T D,Tenenbaum J B,et al. Building machines that learn and think like people[J]. Behavioral and Brain Sciences,2017,40:1 - 70.
[29] Bąk K,Zayan D,Czarnecki K,et al. Example-driven modeling:model = abstractions + examples[A]. Proceedings of the 2013 International Conference on Software Engineering[C]. New York,USA:IEEE Press,2013. 1273 - 1276.
[30] López-Fernández J J,Cuadrado J S,Guerra E,et al. Example-driven meta-model development[J]. Software & Systems Modeling,2015,14(4):1323 - 1347.
[31] Dongsheng Wang. Answering contextual questions based on ontology and question templates[J]. Frontier of Computer Science in China,2011,5(4):405 - 418.
[32] M. Gavaldà and A. Waibel. Growing semantic grammars[A]. Proc 36th Ann Meeting of the Assoc[C]. Computational Linguistics, Montreal, Quebec, Canada: Association for Computational Linguistics,1998. 451 - 456.
[33] Liang P,Potts C. Bringing machine learning and compositional semantics together[J]. Annu Rev Linguist,2015,1 (1):355 - 376.

作者简介

王东升　男,1982 年生,江苏盐城人,现为江苏科技大学计算机学院副教授. 主要研究方向为问答系统、知识图谱和自然语言处理.
E-mail:jsjxy_wds@ just. edu. cn

王卫民　男,1977 年生,浙江绍兴人,现为江苏科技大学计算机学院讲师、博士. 主要研究方向为问答系统和自然语言处理.
E-mail:wangweimin@ just. edu. cn

祁云嵩　男,1967 年生,江苏如皋人,博士,现为江苏科技大学计算机学院教授. 主要研究方向机器学习理论与应用,装备综合保障.
E-mail:mailqys@ 163. com

王　石　男,1981 年生,山东博兴人,现为中国科学院计算技术研究所智能信息处理重点实验室副研究员. 主要研究方向为问答系统、知识图谱和自然语言处理.
E-mail:wangshi@ ict. ac. cn

曹存根　男,1964 年生,江苏东台人,现为中国科学院计算技术研究所智能信息处理重点实验室研究员. 主要研究方向为大规模知识工程.
E-mail:cgcao@ ict. ac. cn