

文章编号: 1003-0077(2023)10-0016-10

基于依存句法的中文语义模型及语义提取方法

王佳琦, 韩 军, 孙启童

(北京航空航天大学 软件开发环境国家重点实验室, 北京 100191)

摘 要: 意图识别与槽填充是语义提取的常用方法, 其存在如下两个问题: 依赖训练数据, 需要标注大量的数据用于训练模型; 可迁移性差, 训练得到的模型难以复用。针对上述问题, 该文对于不同的语义提取场景, 提出了四种不同的语义模型。同时, 该文研究了汉语语法的特点, 基于依存句法, 提出了不同的语义提取算法, 解决了模型难以复用的问题。该方法对数据集的要求较低, 节省了成本。最后, 设计了语义提取实验, 验证了在样本规模小且分布不均匀的数据集下, 语义提取算法相比于部分中文文本分类算法有更高的准确率。该文提出的模型和算法具有一般性, 对于文本分类、人机对话等不同语义提取场景具有较强的指导意义。

关键词: 意图识别; 依存句法; 语义模型; 语义提取

中图分类号: TP391

文献标识码: A

Chinese Semantic Model and Semantic Extraction Method Based on Dependency Syntax

WANG Jiaqi, HAN Jun, SUN Qitong

(State Key Laboratory of Software Development Environment, Beihang University, Beijing 100191, China)

Abstract: Intention recognition and slot filling are common tasks in semantic extraction, which rely on large amount of annotated training data. This paper proposes four different semantic models for different semantic extraction scenarios based on dependency syntax. This method requires less data set and saves cost. Experiments are designed to verify that the proposed methods achieve higher accuracy than some Chinese text classification algorithms in the case of small sample size and uneven distribution of data sets.

Keywords: intention recognition; dependency syntax; semantic model; semantic extraction

0 引言

自然语言处理包含众多的研究和应用领域, 如机器翻译、人机对话、文本分类、信息检索等。不同的领域, 对于语义的定义有所不同, 也对语义提取的粒度及方式有着不同的需求。本文针对不同的语义提取场景, 设计了不同的语义提取算法, 将语句按照不同的语义模型进行提取。本文所提出的基于依存句法的中文语义模型及语义提取方法, 对语义进行了结构化的表述, 很好地地区分了不同的语义提取场景, 并在各个场景中取得了预期的效果。

1 相关研究

1.1 语义模型

国内对于语义模型的研究, 主要有如下两种。基于词组本位语法的语义模型和基于依存句法树的语义模型。

基于词组本位语法的语义模型是由刘群^[1]等人提出的, 目的是实现一个汉英机器翻译系统。他们从配价语法、格语法和论元结构理论入手, 在对这些理论进行了分析研究之后, 构建出了这种通用的语

义模型。1959 年,法国语言学家 Lucien T 提出了配价语法^[2-3]这一概念。配价语法以动词为中心,对不同的动词标以不同的价值,以此来描述不同实体之间的关系。所谓格语法^[4],又被称为语法关系理论。“格”指的是不同词语,在不同语态环境中发生的变化,如常常被提起的“第三人称单数”,指的就是“格”的变化。所谓论元结构理论^[5],则是注重于

语句的语义处理,同时兼顾语句的语法处理。

基于依存句法树的语义模型,是直接使用依存句法分析^[6]生成的树状结构作为语义模型。依存句法分析指的是,在分词和词性标注的基础上,识别语句中各个词语之间的依存关系。以“帮我买张从北京到上海的机票”为例,其依存句法树如图 1 所示。

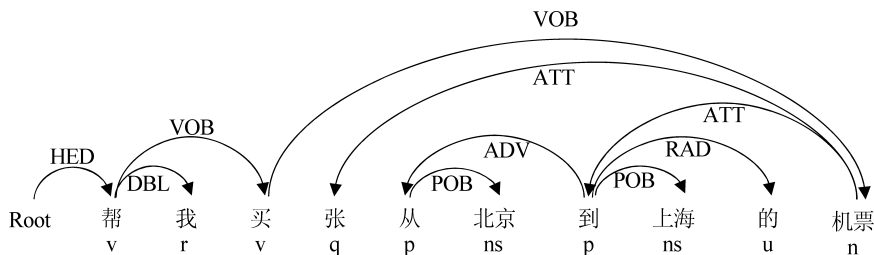


图 1 依存句法树示意图

近些年,哈尔滨工业大学在依存句法分析方面取得了突破性的进展。哈工大社会计算与信息检索研究中心花费数年搭建的中文自然语言处理平台 LTP^[7],同时兼具分词、词性标注、依存句法分析等功能。在 Chinese Dependency Treebank (CDT) 数据集^[8]上,依存句法分析的准确率高达 89.5%。

对于上述两种语义模型,都存在其各自的缺陷。基于词组本位语法的语义模型,其目的是实现汉英翻译系统,其对于语义提取的帮助,只能对单一场景下的单句有效,迁移性较差。基于依存句法树的语义模型,在假设依存句法分析正确的前提下,可以保证包含该语句的所有信息。但这样也会带来另一个问题,即语义模型所包含的信息太多,调用者难以分辨哪些是自己需要的,哪些是自己不需要的。同时,依存句法树的生成只依赖于自然语言处理工具,而其效果是由训练集和训练算法决定的,缺乏一定的泛化能力,对边缘数据的处理不够完善,对语义的描述存在一定的偏差。

1.2 语义提取方法

语义特征指的是语言中构成意义的基本单位。1955 年, Bloomfield 提出了语义特征这一概念^[9]。在汉语领域,朱德熙于 1956 年将语义特征用于分析汉语的语法研究^[10]。

近些年,国内对于语义特征提取方法的研究,逐渐趋于统一。根据邵敬敏、周芍的研究^[11],语义特征提取方法可以总结归纳为如下四种方法:内省概括法、组合分析法、对立比较法和变换分析法。①内省分析法适用一个固定的语法结构,当想要了解其中某

个词语的语义时,可以尝试将其替换为词性相同、含义相近的词语,然后对整个语句作出分析理解,进而概括总结该词语的语义。②组合分析法指的是,将两个或两个以上的词语进行组合后,可产生新的语义。新的语义一般与组合词的词性相关,不同的词性对应着不同的语义。同时,新的语义也会因为组合词所蕴含的意义太多,而呈现出复杂、多义的特点。③对立比较法,指的是针对一词多义这种现象,将该词的所有含义列举出来,然后分别举例,进行分组比较。④变化分析法一般适用于非正常语序的句子,可以将其转变成正常语序,然后通过分析得到语义特征。

语义特征的分析 and 提取,是直接建立在语言学基础上的,是自然语言最直接、最底层的分析方式^[12]。由此可见,如何分析提取语义特征,对于研究语义提取方法有重要的指导作用。但是,上述方法更注重理论,并且均未定义语义模型。对于最终的语义表达结果,没有给出明确的定义。因此,上述方法并不能直接用作语义提取算法。

2 语义模型的构建

要构建语义模型,首先要定义何为语义。广义上来讲,语义指的是语言所蕴含的含义。不同的自然语言,基于不同的符号,这些符号起初不包含信息。人们对于相同的符号,有着相同的理解,此时这些符号便蕴含了信息。

语义模型的作用就是,从自然语言中提取信息,并对这些信息进行结构化的表述。因此,语义模型

的构建,需围绕信息提取来展开。

2.1 键值对语义模型

精简语义提取场景,指的是用户在处理语句时,使用的语义数量相对较少,且语义均为普遍的社会常识,如目的地,出发地等。其特点是,数量较少的语义即可完成用户对语义提取的需求。

针对上述场景,本文提出了键值对语义模型。此模型使用 key-value 键值对的形式表示语义。其定义了所有语义项的 key,并定义了其对应 value 所应满足的各种条件。例如,“目的地”是一种语义项的 key,其 value 的定义是:词性必须为 ns,其在依存句法树中的父节点必须为“到”“至”等词语。对于例句“帮我买张从北京到上海的机票”,其键值对语义模型如图 2 所示。

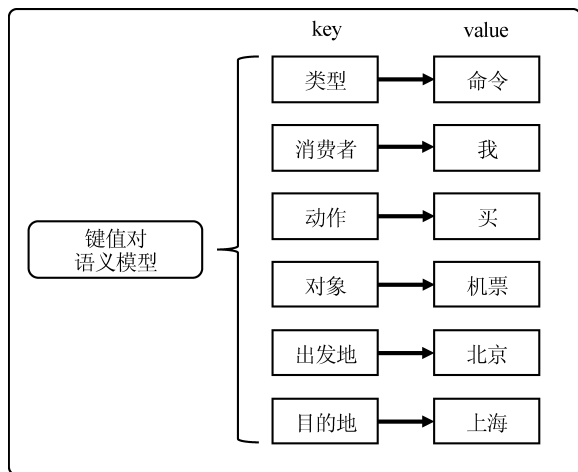


图 2 键值对语义模型示例

键值对语义模型依赖于汉语语法。本文从汉语词典中遍历了所有的汉语字词,将这些字词作为知识库的基本词汇,对它们的重要程度、语义本体、使用频率等属性进行总结。节选如表 1 所示。其中,“重要程度”值越小,该语义越重要;“使用频率”值越大,该语义出现频率越高。

表 1 语义本体总结节选

词语	重要程度	语义本体	使用频率
同意	1	同意	3
撒谎	2	撒谎	2
撒娇	2	撒娇	1
丧气	1	沮丧	1

本文对日常生活中经常使用的 2 513 个字词做

了统计,针对字词的重要程度、使用频率对其进行划分,并将其映射到语义本体。其中,语义本体是一种 key-value 键值对。key 的定义,主要是使用已有的、具有普适性的词语,去描绘其他使用频率较低的词语。例如,用“选择”作为“筛选”的语义本体,用“讨论”作为“商议”“商榷”的语义本体。其次,是使用意图明显的短语来描绘这些词语。例如,用“对人好”作为“善待”“善意”的语义本体,用“想便宜”作为“砍价”的语义本体。

键值对语义模型要求用户对不同的词语进行总结,存在大量的重复性劳动。同时,语义项中 key 的定义没有明确的规范,导致语义模型对于同一语义的描述缺乏标准。例如,在“从北京到上海”这个短语中,提取出的语义是“出发地:北京,目的地:上海”,而在“今天谁值日”这个句子中,提取出的语义是“询问:谁”。其中,“出发地”“目的地”“询问”等 key 的定义,受主观因素影响,其本质是使用另一字符串来消除歧义。

2.2 槽式语义模型

随着语义的增多,键值对语义模型对于“键”的定义,有如下要求:(1)唯一性。在定义一个全新的“键”时,需对比所有已定义的“键”,避免产生歧义。(2)可读性。对于所有已定义的“键”,需配以相应的说明,阐述其具体含义。(3)扩展性。添加新键时,不与旧键产生冲突。这种对于“键”有较高要求的场景,本文将其定义为复杂语义提取场景。其特点是,该场景所需的语义较为复杂,对于“键”有唯一性、可读性、扩展性的需求。复杂语义提取场景是为了区分简单语义提取场景,指的是该语义模型中,含有更多的“键”,这些“键”是由后续开发者来定义的。而该语义模型的用户,只能使用已定义的语义,不能添加语义。

针对上述场景,本文提出了槽式语义模型。此模型对于“键”有了规范化的定义,可以表述任意语法特征。在语句中,根据汉语语言学的研究可知,句式和语义存在映射关系。因此,本文采用保留关键词的形式,对模型有如下定义:(1)保留汉语中的虚词,抽取实词;(2)对于被抽取的词,组织成数组的形式;(3)将被抽取后留下的空位,用“\$+词性”的方式进行填充,得到的字符串作为“键”。对于例句“帮我买张从北京到上海的机票”,可以得到如图 3 所示的槽式语义模型。

槽式语义模型,使用 key-value 键值对的形式

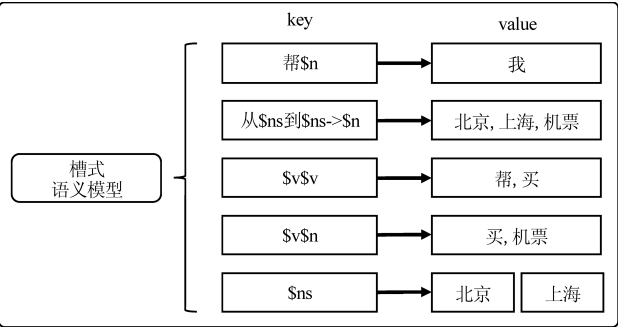


图 3 槽式语义模型示例

表示语义。其中它的 key 是一个字符串,使用“\$”作为一个槽(slot)的占位符,后面跟着的英文字母是这个 slot 应填入词的词性。value 是一个二维数组,内层数组对应着各个 slot 所取出的值,外层数组是考虑到同一语句中,可能存在着多个相同的语义而设计的。

那么,对于语义模型的 key 来说,哪些字词该保留下来,哪些字词该换成 slot,就成了至关重要的问题。在这个问题上,槽式语义模型给出了如下方案,即保留介词、关联词等结构性的虚词,将名词、代词等有着大量同属性的实词做为 slot。语义模型的 key 值,使用“关键词+slot”的形式,实现了语义的唯一性、可读性及扩展性。只需将 value 中的值逐一填入 slot 中,就可得到语序正常、表意清晰的短语。后续开发者可以依据此规则,定义新的语义。

2.3 树状语义模型

分词、词性标注及依存句法分析,包含了语句的全部信息。当语义提取对信息完备性有需求时,需要定义新的语义模型,来确保上述信息不会丢失。

针对上述场景,本文提出了树状语义模型。此模型定义了如表 2 所示的五种基本句式及三种修饰关系,用来描述语句所有的句法关系。

表 2 树状语义模型句法表

类别	结构形式	含义
句式	s+v	主语+谓语
	s+v+o	主语+谓语+宾语
	s+v+c	主语+谓语+补语
	s+v+o+o	主语+谓语+宾语+宾语
	s+v+o+c	主语+谓语+宾语+补语
修饰	dec+word	表任意修饰
	v+不/没+v	表选择
	v+了/着/过	表时态

此模型的特点是,保留语句中的大部分信息,对依存句法树进行重构,通过递归的方式,构建语义树。对于例句“我不知道他有没有去过故宫。”,其树状语义模型如图 4 所示。

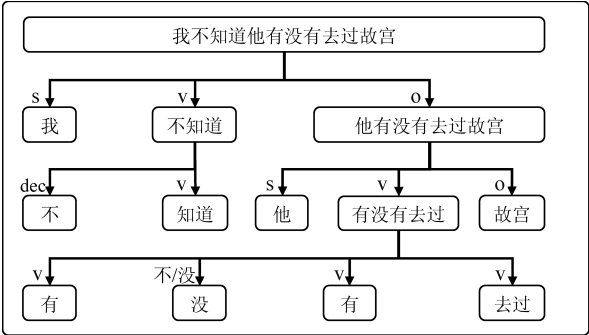


图 4 树状语义模型示例

同时,树状语义模型也可用于多粒度语义的提取。在图 4 中,例句的第一层是“s+v+o”的句式,对应的例句则是“我|不知道|他有没有去过故宫”。用户可以通过查询知识库的方式来分别获取 s、v、o 的具体含义。若知识库中包含该词语,则此时就已经完成了对语义的提取;若知识库不包含该词语,则需查询该节点下是否还有子节点,而它的子节点,也符合给定的句式。因此,用户只需对语义树进行遍历,并与知识库中的信息进行匹配,即可提取出语义信息。

2.4 路径式语义模型

自然语言是动态的,当语义提取对语义有新的要求时,需对语义进行扩充。此时,语义模型的扩展性会显得格外重要。本文将这种场景定义为扩展性语义提取场景。该场景允许用户设计并添加语义,与复杂语义提取场景有着本质的区别。

针对上述场景,本文提出了路径式语义模型。用户可以通过添加语义提取模板的方式,添加语义的提取规则,其示意图如图 5 所示。

所谓路径,指的是在依存句法树中从根节点到叶子节点的路径。依存句法树示例如图 1 所示,对于这棵依存句法树,可以通过深度优先遍历,得到如表 3 所示的中间结果。

路径式语义模型,需要定义语义提取模板,然后将多个模板组合在一起。只有当语句满足所有的模板时,才会完成匹配。规定每个语义提取模板必须为奇数个元素,且奇数位元素形式为“词语+空格+词性”的形式,偶数位元素为依存句法关系。为了增加模板的灵活性,可以使用“\$”符作为通配符。

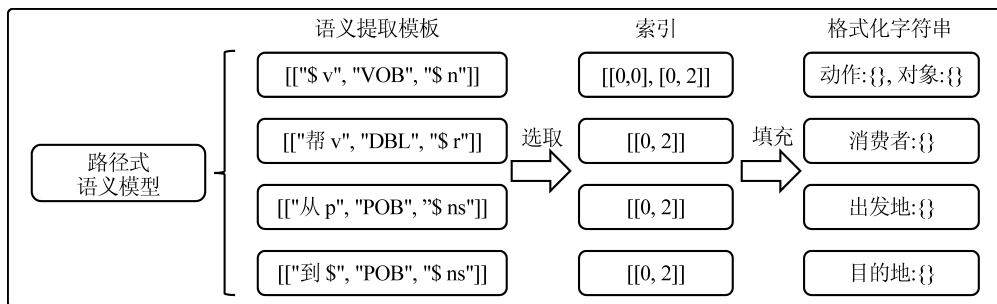


图 5 路径式语义模型示例

表 3 依存句法树深度优先遍历结果

序号	路径信息
1	["帮 v", "DBL", "我 r"]
2	["帮 v", "VOB", "买 v", "VOB", "机票 n", "ATT", "张 q"]
3	["帮 v", "VOB", "买 v", "VOB", "机票 n", "ATT", "到 p", "ADV", "从 p", "POB", "北京 ns"]
4	["帮 v", "VOB", "买 v", "VOB", "机票 n", "ATT", "到 p", "POB", "上海 ns"]
5	["帮 v", "VOB", "买 v", "VOB", "机票 n", "ATT", "到 p", "RAD", "的 u"]

路径式语义模型的存储结构如表 4 所示。

表 4 路径式语义模型存储结构

Semantics	Indexes	Formatting
["到 p", "ADV", "从 \$", "\$", "\$ ns"], ["到 p", "POB", "\$ ns"]]	[[0, 4], [1, 2]]	出发地 {} 目的地 {}

Semantics 为语义提取模板，只有满足模板，才可以说语句满足这一语义。Indexes 为用户自定义要提取位置的索引。在用语义提取模板对语句进行比对后，如果语句满足所有的模板，就可以匹配出相应的值。但是，用户需要的不一定是所有的值，因此需要用户自己定义提取哪个索引对应的词。Formatting 是预设字符串，用户可以根据索引提取出来的词语填入该预设字符串中，得到用户自定义的语义。

引提取出来的词语填入该预设字符串中，得到用户自定义的语义。

3 语义提取算法

3.1 基于关键词的语义提取算法

基于关键词的语义提取算法使用了键值对语义模型和槽式语义模型，通过对词语的分类，首先确定语句中的关键词，然后对关键词的父节点、子节点、兄弟节点、祖先节点、子孙节点等进行逻辑判断，依次筛选出语句的语义项，组合生成最终的语义。

如图 6 所示，以语句“帮我买张从北京到上海的机票”为例，先使用自然语言处理工具对其进行分词、词性标注和依存句法分析，得到如图 2 所示的依存句法树。因为语句中存在“帮”和动宾短语，因此可以判断此语句是一个命令句。然后，确定关键词“帮”，其 DBL 边所指向的子节点是该命令的作用对象，此句中可定义为消费者，得到语义项“消费者：我”；确定关键词“从”，其 POB 边所指向的子节点是“北京”，其词性为 ns 地点，得到语义项“出发地：北京”；确定关键词“到”，其 POB 边所指向的子节点是“上海”，其词性为 ns 地点，得到语义项“目的地：上海”。以此类推，得到最终的语义。

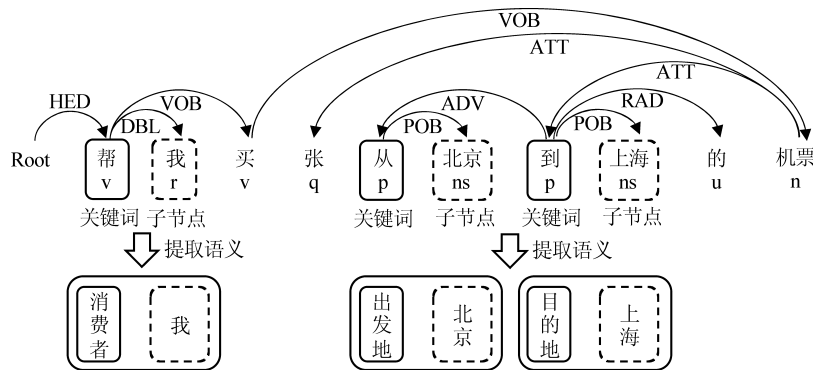


图 6 基于关键词的语义提取算法示例

该算法的伪代码如表 5 所示。

表 5 基于关键词的语义提取算法

Algorithm 1: 基于关键词的语义提取算法	
输入：分词 words, 词性标注 postags, 依存句法 arcs, 长度 len, 关键词词典 map	
输出：该语句中基于关键词提取到的语义 Y	
1	begin
2	for i $\leftarrow$ 0 to len-1 do
3	if words[i] == kw and postags[i] == kp and arcs[i] == ka then //找到关键词
4	Y.append(map[word[i]]) //添加语义
5	end if
6	end for
7	end

3.2 树状语义提取算法

树状语义提取算法是依据树状语义模型,对依存句法树进行整合,得到全新的树状结构的一种算法。树状语义模型包含“简单句”和“修饰关系”这两种句

型结构,这两种结构又细分为“主语+谓语”“主语+谓语+宾语”“主语+谓语+补语”等句型结构。这里的句型结构并没有严格的规定,取决于用户对于句型结构的理解,用户也可以根据需求自定义句型结构。

如图 7 所示,以“我不知道他有没有去过故宫。”为例,介绍树状语义提取算法。首先构造依存句法树,然后使用层序遍历的方式遍历这棵树。这里以“主语+谓语+宾语”为例,当遍历到节点“知道 v”时,发现该节点的词性是动词,此时,已经找到了“谓语”,需要再找到“主语”和“宾语”。于是,递归遍历该节点的所有子节点,判断其子节点的 arc 值是否为“SBV”,是的话,则找到该动词的主语;同时判断其子节点的 arc 值是否为“VOB”,是的话,则找到该动词的宾语。找到“主语”“谓语”和“宾语”的节点之后,以这些节点分割句子,即可得到主语是“我”,谓语是“不知道”,宾语是“他有没有去过故宫”。递归时,“他有没有去过故宫”也会得到对应的结果,即主语是“他”,谓语是“有没有去过”,宾语是“故宫”。以此类推,得到最终结果。

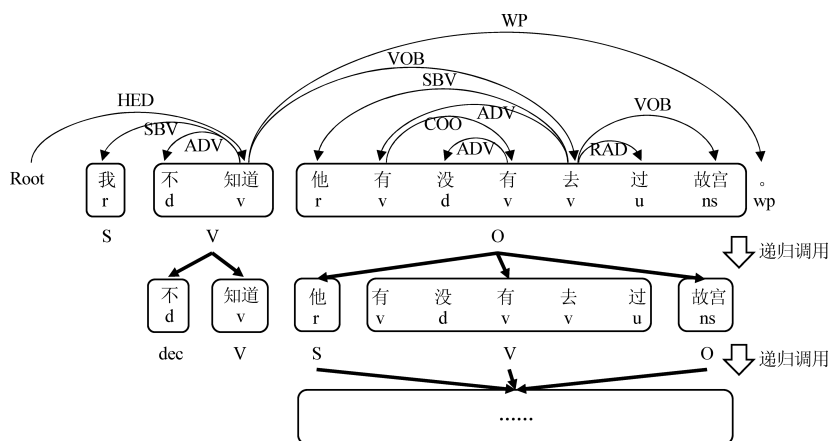


图 7 树状语义提取算法示意图

该算法的伪代码如表 6 所示。

表 6 树状语义提取算法

Algorithm 2: 树状语义提取算法	
输入：语句 s, 依存句法树层序遍历结果 level_list, 行索引 x, 列索引 y	
输出：该语句中提取到的树状语义 Y	
1	begin
2	function helper(s, level_list, x, y)
3	node $\leftarrow$ level_list[x][y]
4	if node.pos != "v" or x + 1 >= len(level_list) then
5	return
6	end if

续表

7	si $\leftarrow$ -1, vi $\leftarrow$ node.index, nr $\leftarrow$ level[x+1]
8	for i $\leftarrow$ 0 to len(nr)-1 do
9	if nr[i].arc == "SBV" then
10	si $\leftarrow$ nr[i].index
11	end if
12	if nr[i].arc == "VOB" and nr[i].pos == "v" then
13	helper(s, level_list, x+1, i) //递归调用
14	end if
15	end for
16	Y.append(s.words[si], s.words[si+1:vi+1], s.words[vi+1:]) //格式化并添加该层的树状语义
17	end

3.3 可扩展的语义提取算法

可扩展的语义提取算法,是基于路径式语义模型而构建的,其示意图如图 8 所示。

图 8 以语句“帮我买张从北京到上海的机票”为例,先对依存句法树进行深度优先遍历,得到依存句

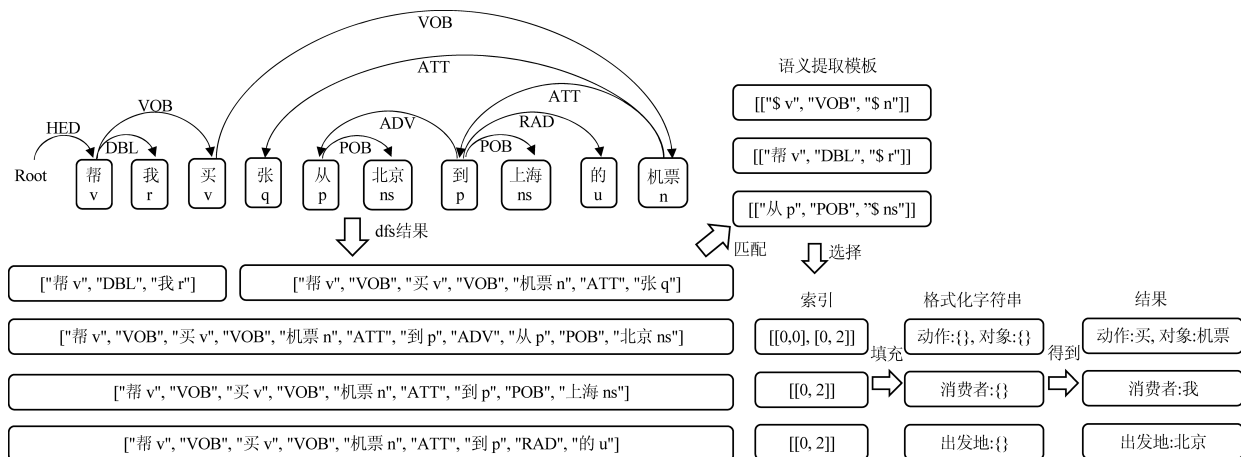


图 8 可扩展的语义提取算法示意图

其扩展性主要表现在两个方面。首先,其没有针对某一个词语的复杂逻辑结构,而是由用户直接定义语义模板,用来对语句进行处理。当语句与所有模板都匹配成功时,则提取语义。其次,用户可以根据实际需求,控制语义提取的粒度。若用户对某个语义的准确率要求很高,那么只需设置多个语义模板,或者定义更长的语义模板,即可达到要求。若用户对某个语义的准确率要求不高,那么可以设置较短的语义模板。

该算法的伪代码如表 7 所示。

表 7 可扩展的语义提取算法

Algorithm 3: 可扩展的语义提取算法

输入: 语句路径 s , 语义匹配模板 $templates$

输出: 是否与该模板匹配 $flag$, 匹配成功时对应的词语 $words$

```

1 begin
2   foreach  $t \in templates$  do //预处理,给每个词语加上小括号
3      $c.append(add\_bracket(t))$ 
4   end foreach
5    $flag \leftarrow false$ 
6   foreach  $cl \in c$  do
7      $flag \leftarrow false$ 
8     foreach  $sl \in s$  do
9       if  $len(cl) > len(sl)$  then
10        continue

```

续表

```

11    end if
12     $ij\_flag, ij\_list \leftarrow list\_contain(cl, sl) //cl$  中
    若含有  $sl$ ,返回  $bool$  及索引
13    if  $ij\_flag$  then
14       $words.append(ij\_list)$ 
15       $flag \leftarrow true$ 
16      break
17    end if
18  end foreach
19  if !  $flag$  then
20    break
21  end if
22 end foreach
23 return  $flag, words$ 
24 end

```

4 实验与分析

4.1 实验数据集

本节以“怎么”“什么”“哪”“多少”“谁”这五个疑问代词为例,展示了如何通过语义模型及语义提取算法来提取语义。之后,使用北京语言大学提供的 BCC 语料库^[12]进行实验,其中每个疑问代词对应的例句有 1 000 句,共 5 000 句例句。例句的实际含义为人工标注,各个实验的设计与结果如下文所述。

4.2 语义提取实验

疑问代词“怎么”，在现代汉语词典中有五种含义，分别是“询问”“泛指”“虚指”“程度”“否定”。由于泛指和虚指对于语义提取来说，不具有明确的意义，而有些特殊语句可能包含两种以上的含义，如例句“大家怎么会没有意见呢？”，其中“怎么”的含义，既表示了询问，又表示了否定。因此，将泛指和虚指筛选后，将剩下的含义组合，给出了本文对于“怎么”的六种含义的分类，分别是“询问”“程度”“否定”“询问|程度”“询问|否定”“程度|否定”。

关于疑问代词“怎么”，本文仅通过对 50 条包含“怎么”的语句进行总结后，就可定义出 20 个语义提取模板，节选如表 8 所示。

表 8 “怎么”语义提取模板节选

模板	索引	格式化	例句
[[" \$ v", "ADV", "会 v", "ADV", "怎么 r"]]	[[0, 0]]	惊讶：怎么会{ }	怎么会变成这个样子
[[" \$ \$", "ADV", "怎么 r"]]	[[0, 0]]	询问 程度：怎么{ }	他怎么受得了这份苦
[[" \$ a", "ADV", "不怎么 d"]]	[[0, 0]]	程度：不太{ }	春天的黄昏还并不怎么悲凉
[[" \$ v", "ADV", "怎么 r"], [" \$ v", "ADV", "不 d"]]	[[0, 0]]	表否定：怎么不{ }	你怎么不来我家玩了

然后，本文使用表 8 所示的语义提取模板对 BCC 语料库中的 1 000 条包含“怎么”的语句，使用定义了 20 个语义提取模板的可扩展的语义提取算法进行了语义提取测试，结果节选如表 9 所示。

表 9 “怎么”语义提取结果节选

句子	提取结果	含义	T/F
他最近怎么了？	["询问：他怎么了"]	询问	T
狐狸怎么会变成人呢？	["询问 否定：怎么会变成", "否定：不会"]	询问 否定	T
我们怎么做呢？	["询问 程度：怎么做", "否定：不做"]	询问	F
怎么不见了？	["询问：怎么不见", "否定：不见"]	询问	T

由表 8、表 9 可知，当语义提取模板设计过短时，可以与更多的语句进行匹配，得到更多的语义提取结果。这些结果并没有优先级的设定，会被填入一个二维数

组中。这也印证了汉语的复杂性。例如，同一句话，既可以是询问，也可以是否定。对于这种情况，用户可以在设定格式化字符串时，传入不同的数字，代表不同的优先级，然后对得到的语义结果进行排序和取舍。

对于基于关键词的语义提取算法及树状语义提取算法，这两种算法不支持额外的配置，因此直接对实验数据集中的语句进行了语义提取实验。与使用了 20 个语义提取模板的可扩展的语义提取算法进行对比，其准确率如表 10 所示。

表 10 三种语义提取算法准确率对比

(单位：%)

疑问代词	关键词	树状	可扩展
怎么	87.9	86.8	86.3
什么	94.0	90.2	93.5
哪	93.1	87.5	96.0
多少	87.4	83.6	85.5
谁	94.4	89.4	97.9

由表 10 可知，对于定义了 20 个语义提取模板的可扩展的语义提取算法，其准确率与基于关键词的语义提取算法各有优劣。这种情况主要由模板数量及质量决定。由 4.3 节实验可知，当模板数量逐渐增加时，可扩展的语义提取算法效果会越来越好。同时，当定义的模板适用于更多语句时，其提取效果会更好。对于树状提取算法，其准确率均处于 80% 以上，不符合正常语序的句子，如倒装句等，会造成树状语义提取出错。

4.3 扩展模板实验

针对语义提取方式的可行性与扩展性，本文通过扩充模板数量的方式，分别使用了 10 个、15 个、20 个、25 个、30 个语义提取模板，对“怎么”、“什么”、“哪”、“多少”、“谁”五种疑问代词进行了语义提取实验。实验所使用的数据取自北京语言大学的 BCC 语料库，每个疑问代词有 1 000 条语句。最终的结果如表 11 所示。

表 11 语义提取准确率 (单位：%)

序号	代词	类别	10 个模板	15 个模板	20 个模板	25 个模板	30 个模板
1	怎么	6	83.3	85.2	86.3	88.6	89.8
2	什么	4	86.2	90.4	93.5	95.2	96.0
3	哪	3	91.7	93.5	96.0	97.3	98.3
4	多少	6	78.2	82.8	85.5	88.2	90.1
5	谁	2	93.6	95.1	97.9	98.6	99.1

由表 11 数据可以绘制出如图 9 所示的模板数量对于语义提取效果影响折线图。

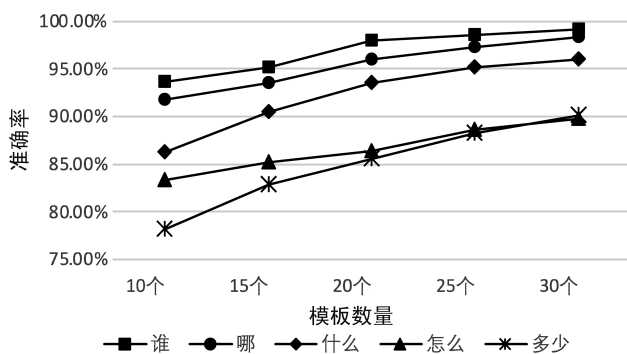


图 9 模板数量对于语义提取效果影响折线图

由图 9 可知,随着模板配置数量的提升,各个疑问代词的语义提取准确率也逐渐升高,且总体上有放缓的趋势。从 20 个模板以后,折线都趋于平缓。这是因为,模板的数量已经足够大,足以正确提取 90%左右的语句。而提取错误的语句,通常是一些边缘数据,针对这些边缘数据设计的语义提取模板,泛用性相对较低,因此折线会趋于平缓。

另外,图中不同折线的高度略有不同。“怎么”有六个类别,“什么”有四个类别,“哪”有三个类别,“多少”有六个类别,“谁”有两个类别。一般来说,类别多的疑问代词,语义提取效果会更好。但在 30 个模板时,“多少”的准确率超过了“怎么”,这是由于语义提取模板的设定和数据标注不准确导致的。语句中词语的含义无法量化,存在一定的主观误差。

4.4 与深度学习算法的对比实验

本节采用包含“多少”的语句,共计 1 000 条,语料来自于北京语言大学提供的 BCC 语料库,类型为人工标注,该数据集的详细信息如表 12 所示。

该数据集有以下特点:

- (1) 样本规模小;
- (2) 数据分布不均匀;
- (3) 不同含义的语句句法结构类似。

表 12 数据集详情表

编号	个数	含义	例句
0	836	询问数量	你吃了多少面包
1	11	不定数量	母亲没有多少文化
2	22	稍微	你多少安点心吧
3	20	或多或少	多少总学了几手
4	105	数量大小	没有多少喜剧色彩
5	6	其他含义	多少能得好

中文文本分类任务,现阶段的主流解决方案是深度学习。于是,本文与基于深度学习的中文文本分类算法 TextCNN^[13],TextRNN^[14],FastText^[15],TextRCNN^[16],TextRNN_Att^[17],DPCNN^[18],Transformer^[19]做了对比。训练集、验证集和测试集的比例为 3 : 1 : 1,保证在各个集合中每一类所占的比例相同。使用上述方法,对本节给出的数据集进行训练测试,各模型的准确率如表 13 所示。

表 13 文本分类算法准确率

(单位: %)

TC	TR	TRA	TRC	FT	DPC	Trans
85.94	86.72	85.94	85.94	85.16	85.94	85.94

由表 13 可知,TextCNN,TextRNN,TextRCNN,DPCNN 和 Transformer 的准确率均为 85.94%。TextRNN 准确率为 86.72%,FastText 准确率为 85.16%。

以模型的最高准确率 86.72%为基准,与不同模板数量对应的分类效果做对比,可得到如图 10 所示的折线图。

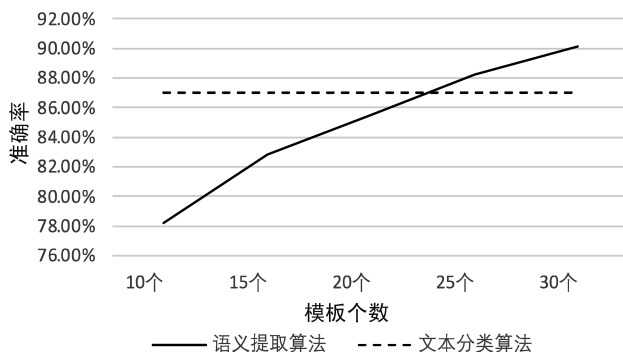


图 10 语义提取算法与文本分类算法对比折线图

由图 10 可知,当模板数量少于 25 个时,各模型的准确率都高于语义提取算法。当模板数量达到 25 以上时,语义提取算法的准确率会超过上述模型,但仍保持了增长的趋势。

综上所述,当样本规模小,分布不均匀,且各个分类的区别不明显时,随着模板数量的增加,语义提取算法的准确率会越来越高,最终超过上述模型。因此,语义提取算法在处理上述数据时,有明显的优势。

5 结论

本文主要通过对语言学理论体系的研究,总结归纳汉语的语法特点,使用自然语言处理工具的分

词、词性标注及依存句法分析的结果,提出了基于依存句法的语义模型,提出了相应的语义提取算法。

本文充分地分析研究了汉语的语法特点,在此基础上,提出了四种不同的语义模型,分别是键值对语义模型、槽式语义模型、树状语义模型及路径式语义模型,这些模型分别适用于不同的语义提取场景,然后提出了对应的语义提取算法,分别是基于关键词的语义提取算法、树状语义提取算法及可扩展的语义提取算法。最后,使用 BCC 语料库的 5 000 条语句,进行了语义提取实验,验证了上述语义提取方法的可行性与优越性。

上述模型与算法,均适用于训练数据少的应用场景。用户只需总结特定场景中常见的语法特点,即可制定出相应的语义提取规则,从而完成对语义的提取。其次,对于跨场景的情况,存在大量的可重用的语义提取规则,可以直接用于不同场景下的语义提取。因此,该方法对于训练数据少的应用场景,有着更好的适用性,且弥补了传统算法可迁移性差等缺点。

参考文献

- [1] 刘群,詹卫东,常宝宝,等. 一个汉英机器翻译系统的计算模型与语言模型[C]//智能计算机接口与应用进展: 第二届中国计算机智能接口与智能应用学术会议论文集. 北京: 电子工业出版社, 1997: 253-258.
- [2] FINK S R. Aspects of a pedagogical grammar based on case grammar and valence theory[M]. Max Niemeyer Verlag, 2011.
- [3] 徐凯.现代汉语配价语法研究综述[J].今古文创,2021, 1(31): 108-109.
- [4] HARRIS M. Some problems for a case grammar of Latin and early Romance[J]. Journal of Linguistics, 1975, 11(2): 183-194.
- [5] BACKLEY P. Introduction to element theory[M]. Edinburgh University Press, 2011.
- [6] NIVER J. Dependency parsing[J]. Language and Linguistics Compass, 2010, 4(3): 138-152.
- [7] CHE W, LI Z, LIU T. LTP: A Chinese language technology platform[C]//Proceedings of the Coling: Demonstrations, 2010: 13-16.
- [8] LIU T, MA J, LI S. Building a dependency treebank for improving Chinese parser[J]. Journal of Chinese Language and Computing, 2006, 16(4): 207-224.
- [9] BLOOMFIELD L. Linguistics and reading[J]. Language Learning, 1955, 5(3-4): 94-107.
- [10] 朱德熙. 现代汉语形容词研究[J]. 语言研究, 1956, 1(1): 1-37.
- [11] 邵敬敏,周芍. 语义特征的界定与提取方法[J]. 外语教学与研究(外国语文双月刊), 2005, 37(1): 21-28.
- [12] 荀恩东,饶高琦,肖晓悦,等. 大数据背景下 BCC 语料库的研制[J]. 语料库语言学, 2016, 3(1): 93-118.
- [13] CHEN Y. Convolutional neural network for sentence classification[D]. University of Waterloo, 2015.
- [14] LIU P, QIU X, HUANG X. Recurrent neural network for text classification with multi-task learning [C]//Proceedings of the 25th International Joint Conference on Artificial Intelligence, 2016: 2873-2879.
- [15] ZHOU P, SHI W, TIAN J, et al. Attention based bidirectional long short-term memory networks for relation classification[C]//Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, 2016: 207-212.
- [16] LAI S, XU L, LIU K, et al. Recurrent convolutional neural networks for text classification[C]//Proceedings of the 29th AAAI Conference on Artificial Intelligence, 2015: 2267-2273.
- [17] JOULIN A, GRAVE É, BOJANOWSKI P, et al. Bag of tricks for efficient text classification[C]//Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, 2017: 427-431.
- [18] JOHNSON R, ZHANG T. Deep pyramid convolutional neural networks for text categorization[C]//Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, 2017: 562-570.
- [19] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]//Advances in Neural Information Processing Systems, 2017: 5998-6008.
- [20] PULVERMÜLLER F. Neurobiological mechanisms for semantic feature extraction and conceptual flexibility[J]. Topics in Cognitive Science, 2018, 10(3): 590-620.



王佳琦(1996—),硕士研究生,主要研究领域为自然语言处理。

E-mail: wangjq@buaa.edu.cn



孙启童(1994—),博士研究生,主要研究领域为自然语言处理。

E-mail: sunqt@buaa.edu.cn



韩军(1969—),通信作者,博士,教授,主要研究领域为人工智能、网络综合优化、现代计算方法、服务计算等。

E-mail: jun_han@buaa.edu.cn