

文章编号: 1003-0077(2009)03-0095-08

## 隐喻字面语义表示与生成

王金锦<sup>1</sup>, 杨芸<sup>1</sup>, 周昌乐<sup>1,2</sup>

(1. 厦门大学 人工智能研究所, 福建 厦门 361005; 2. 浙江大学 语言与认知中心, 浙江 杭州 310028)

**摘要:** 在隐喻理解中, 隐喻字面语义表示是隐喻深层语义表示的前提; 确切地说, 隐喻字面语义表示语言作为隐喻计算的输入语言直接影响到隐喻的最终释义, 因此隐喻字面语义表示对隐喻的机器理解有着重要的影响作用。但在国内学术界, 还鲜有这方面的研究。鉴于此, 该文结合汉语隐喻特点, 从隐喻字面语义表示的角度出发, 将汉语隐喻分为无嵌套隐喻和嵌套隐喻两种。并在分析隐喻字面语义(浅层语义信息和隐喻信息)的基础上, 提出了隐喻角色依存表示语言作为隐喻字面表示语言, 最后给出隐喻角色依存表示语言生成算法。实验表明, 该方法引入到汉语隐喻解释机制中是富有成效的。

**关键词:** 计算机应用; 中文信息处理; 隐喻字面语义; 隐喻计算; 无嵌套隐喻; 嵌套隐喻; 隐喻角色依存表示语言  
**中图分类号:** TP391      **文献标识码:** A

## Metaphor Literal Meaning Representation and Generation

WANG Jinjin<sup>1</sup>, YANG Yun<sup>1</sup>, ZHOU Changle<sup>1,2</sup>

(1. Institute of Artificial Intelligence Xiamen University, Xiamen, Fujian 361005, China;

2. Center for the Study of Language and Cognition, Zhejiang University, Hangzhou, Zhejiang 310028, China)

**Abstract** The representation of the metaphor literal meaning is the premise of deeper semantic representation and the further processing of metaphor. In fact, as the input language of the metaphor computation system, the metaphor literal meaning representation language would influence the final semantic calculation results. In this paper, we first classify the Chinese Metaphor into two categories—the Un-Nested Metaphor and the Nested Metaphor, based on the analysis of Chinese metaphor characteristics. We further design the metaphor role dependency representation language to describe the metaphor literal meaning in terms of its shallow semantic and metaphor information. Finally, the experiments show that the proposed method is quite effective in interpreting Chinese metaphor.

**Key words:** computer application; Chinese information processing; metaphor literal meaning; metaphor computation; un-nested metaphor; nested metaphor; metaphor role dependency representation language

### 1 引言

隐喻形式化表示是隐喻计算研究的一方面。戴帅湘等人首次提出汉语隐喻形式化办法<sup>[1]</sup>。但是在本质上却无法从计算机角度推导出隐喻的真正含义。隐喻理解首先要获取隐喻的字面语义, 然后才能进一步转译到真实意义。而隐喻字面语义表示是

隐喻识别后, 对隐喻句子特别是句中的隐喻成分进行字面上的表示。显然从隐喻识别到隐喻理解需要有隐喻字面语义表示这一中间过程。

隐喻识别是隐喻字面语义乃至隐喻深层语义表示的前提。杨芸等人提出了基于隐喻角色依存模式的汉语隐喻计算分类体系<sup>[2]</sup>, 将隐喻划分为两大类: “非常规指称型隐喻”和“非常规搭配型隐喻”, 并将这两大类进一步细化为 32 种子类, 分别给出类

收稿日期: 2008-07-21 定稿日期: 2008-11-10

基金项目: 国家自然科学基金资助项目(60373080); 福建省青年科技人才创新资助项目(2008F3105)

作者简介: 王金锦(1983—), 女, 硕士生, 研究方向为自然语言处理, 隐喻计算; 杨芸(1980—), 女, 博士, 研究方向为自然语言处理, 隐喻计算; 周昌乐(1959—), 男, 教授, 博导, 主要从事人工智能领域的研究工作, 研究方向为计算语言学, 理论脑科学和认知逻辑学。

别标记,对可能的隐喻成分给出了量化判断方法,并首次尝试了对大规模不同类型的汉语隐喻的自动识别。在此基础上,李剑锋<sup>[3-4]</sup>针对搭配型隐喻大类——以动词为中心的隐喻句子进行了识别,取得了良好的效果。本文在此隐喻识别基础上,将32种隐喻类别信息引入到隐喻字面语义表示语言中,并从隐喻字面语义表示的角度出发,将汉语隐喻分为无嵌套隐喻和嵌套隐喻两种。

## 2 隐喻字面语义表示

### 2.1 隐喻字面语义界定

本文是面向隐喻理解来研究隐喻字面语义表示语言。因此首先对隐喻字面语义进行界定。我们认为隐喻的字面语义包括两个方面:浅层语义信息和隐喻信息。

#### 2.1.1 浅层语义信息

本文以句子为单位展开研究。句子由词构成,部分词组合成短语。其中句子成分包括词和短语。一个隐喻句子可以形式化为一棵树,如图1所示,词节点表示词,依存关系描述句子组织结构。

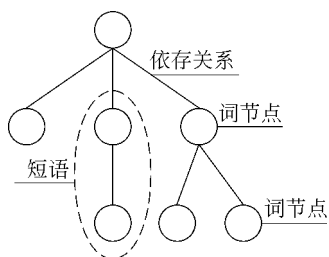


图1 隐喻句子树状图

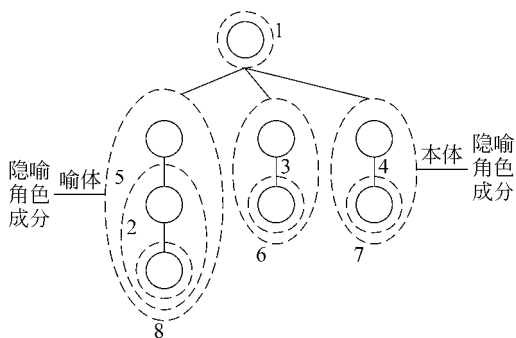


图2 隐喻角色成分

在树状图中,我们使根节点只代表词,而叶子节点是由一个词构成的特殊短语。以内部节点为根节点构成的子树代表短语。值得注意的是,依存关系表现形式上体现为词与词之间的关系,实质上却是

词节点所代表的词与其子树所代表的短语之间的关系。由于汉语本身语言特性,一个词可以由好几个句子成分加以修饰,对应树状图中则表现为,一个词节点(叶子节点除外)可以有一个或多个不同依存关系的子树,且这些子树相对该词,在句中又有不同的位置关系。

一个词节点语义信息可以是词本身及词性。因此,一个隐喻句的浅层语义信息包括词节点语义信息,该词节点与其所有不同子树之间的依存关系,及这些子树在句中相对于该词的不同位置关系等。

#### 2.1.2 隐喻信息

一个具体的隐喻话语的句子成分个数是固定的。如图2中实线圆表示句中的8个词,虚线圆则为8个句子成分。从结构上看,句子成分1,2,3,4表示词语,5,6,7,8表示短语。部分句子成分包含隐喻角色<sup>[3]</sup>:本体,喻体,喻底,喻词。我们把含有隐喻角色的句子成分称为隐喻角色成分。图2中句子成分7,8分别是本体和喻体,两者都是隐喻角色成分。至此句子成分和隐喻角色成分已确定,并且从根本上迎合了语言隐喻成分取值的不确定性难点<sup>[5]</sup>,即隐喻角色成分可以标识任何形式的语言隐喻成分,包括任何单个词语,任何形式的短语,甚至句子。

无嵌套隐喻是指一个隐喻话语只包含一个隐喻单元,其隐喻信息为隐喻角色成分。隐喻话语中的嵌套隐喻是指一个语言隐喻单元包含另一个语言隐喻单元。两者之间是包含与被包含的关系。如图3中隐喻角色成分8,其本身又是一个隐喻。我们把具有包含关系的隐喻单元称为嵌套主隐喻,被包含关系的隐喻单元称为嵌套次隐喻。本质上,嵌套主隐喻中的隐喻角色成分的语义需要通过解释嵌套次隐喻的意义来获得,这种机制我们称为嵌套语义映射。因此,要实现嵌套主隐喻从字面语义过渡到真实语义,首先要解释嵌套次隐喻的实际意义。嵌套隐喻的隐喻信息除隐喻角色成分外,还包括嵌套主隐喻,嵌套次隐喻和嵌套语义映射。

综上所述,一个隐喻句的字面语义包括浅层语义信息和隐喻信息。隐喻字面语义形式化表示是本文的重点和难点。

### 2.2 隐喻的字面语义表示语言: 隐喻角色依存表示语言

使用隐喻角色依存表示语言是对隐喻字面语义进行形式化的一种方法。隐喻字面语义是隐喻分类

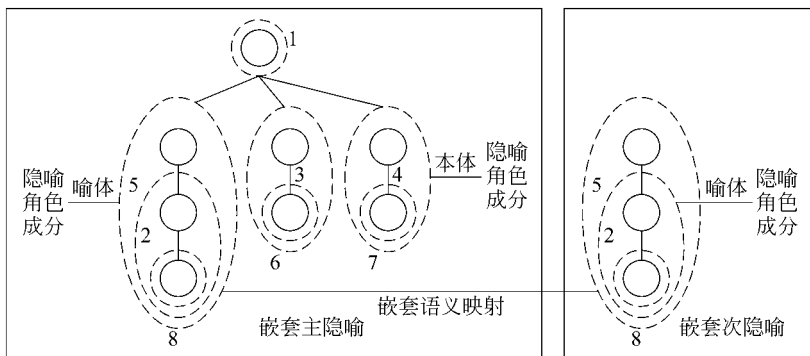


图 3 嵌套隐喻

识别<sup>[2-3]</sup>的结果,同时也是隐喻分析和解释程序的输入。本节定义了隐喻角色依存表示语言 (Metaphor Role Dependency Representation Language, MRDRL) 来表示隐喻字面语义。MRDRL 基于依存文法<sup>[6-9]</sup>, 又新增隐喻信息。句子关系除依存结构外,在嵌套隐喻中,又引入嵌套语义映射。所以隐喻角色依存表示语言不仅是依存语言的扩充,而且是适合隐喻计算的语言。

**定义 2.2.1** 隐喻角色依存表示语言 (Metaphor Role Dependency Representation Language, 简称为 MRDRL)

隐喻角色依存表示语言用来描述一个隐喻性话语的字面语义,即描述隐喻分类识别程序识别出的由词节点、依存关系有向弧和隐喻角色构成的隐喻角色依存模式。

本文主要讨论用 MRDRL 描述无嵌套隐喻和嵌套隐喻的字面语义。

### 2.3 无嵌套隐喻的字面语义表示

MRDRL 表示无嵌套隐喻的举例如下:

**例 1** 脸蛋漂亮得像三色紫罗兰。〈EOS〉

```

Class 14
{脸蛋} Met_t {像} Met_m {三色紫罗兰} Met_v
[6]色 [5]三(QUN)
[7]紫罗兰 [6]色(ATT)
[4]像 [7]紫罗兰(VOB)
[3]得 [4]像 (DEI)
[2]漂亮 [3]得(CMP)
[2]漂亮 [1]脸蛋(SBV)
[8]〈EOS〉 [2]漂亮(HED)

```

图 4 例句 1 隐喻识别结果

图 4 为例 1 句隐喻识别结果: 隐喻角色标注和词与词之间的依存关系,其中每个词前面的数字表示该词在原句中的位序。图 5 是 MRDRL 描述无

```

(Class 14: <EOS>: Met==1
(HED : 漂亮, a, Met_Role(Met_g) : Left(1)
(SBV : 脸蛋, n, Met_Role(Met_t) : Left(1)
(CMP : 得, uf, Met_Role(NULL) : Right(1)
(DEI : 像, vx, Met_Role(Met_m) : Right(1)
(VOB : 紫罗兰, n, Met_Role(Met_v) : Right(1)
(ATT : 色, q, Met_Role(NULL) : Left(1)
(QUN : 三, m, Met_Role(NULL) : Left(1))))))

```

图 5 例句 1 无嵌套隐喻字面语义表示

嵌套隐喻句子字面语义的一个实例。下面给出无嵌套隐喻角色依存表示语言的形式化定义。

一个隐喻话语的字面语义由若干语义项表示,并用依存关系组织。每一行代表一个语义项,每一个语义项如图 5 中以行单位描述,依存结构用括号表示。

**定义 2.3.1** 无嵌套隐喻语义 (Un-Nested Metaphor Semantic, 简称为 UNMS)

无嵌套隐喻语义是一个集合,  $UNMS = \{UNMFT, UNMHT, UNMOT\}$ 。每一个无嵌套隐喻语义项 (Un-Nested Metaphor Semantic Term, 简称为 UNMST) 由三部分组成,用两个冒号“:”间隔,每一部分又有多个属性时,用“,”间隔。记做 ( : ... , ... , ... : )。

◦ 无嵌套隐喻首项 UNMFT,  $UNMFT = (Class N : \langle EOS \rangle : Met == i)$ , 标识该隐喻的类型及隐喻嵌套的层数信息, Class N 表示最外层类别代号,  $N = 1, 2, \dots, 32$ , 分别代表 32 类隐喻句<sup>[2]</sup>。〈EOS〉为话语结束符, Met 为隐喻嵌套变量, i 表示该话语中含有隐喻的个数,  $i = 1, 2, \dots$ ,  $Met == 1$  表示该隐喻为无嵌套隐喻,  $Met == 2$  表示该隐喻中还嵌套了另一层隐喻,以此类推。

◦ 无嵌套隐喻中心项 UNMHT,  $UNMHT = (HED : WORD, POS, Met\_Role(x) : LOC)$ , 标

识隐喻句的中心词节点。HED 这一依存关系名, 标识为隐喻句中心节点。WORD 描述词语本身, POS 为 WORD 的词性, 词性标注使用的是“863”的标注体系的词性标准<sup>[10]</sup>, Met\_Role(x) 为 WORD 的隐喻角色<sup>[2]</sup>, 其中 x 取值除了本体角色标记 Met\_t, 喻体角色标记 Met\_v, 喻底角色标记 Met\_g 和喻词角色标记 Met\_m 四个值外, 还有非隐喻角色标记 NULL。LOC 表示当 WORD 为受支配词时, 与它的中心词(上一个语义项)在句中的位置关系, 由 Left(i) 和 Right(i) 表示, i=1, 2, ..., 如 Left(1) 表示 WORD 靠近其依存中心词左边最近的第一个位置, 以此类推……

◦ 无嵌套隐喻其他项 UNMOT, UNMOT = (DR: WORD, POS, Met\_Role(x): LOC), DR = {ATT, SVB ... POB}, 表示依存关系。特别地, Met\_Role(x) 表示为以 WORD 为根的子树所属的隐喻角色, 这与 UNMHT 中所定义的 Met\_Role(x) 不同。其他定义如同 UNMHT。

用 MRDRL 描述无嵌套隐喻具有以下特点:

◦ 所有的 UNMST 之间通过依存关系组成一个隐喻句子结构, 即 MRDRL 是基于依存文法的。具有依存结构所具有的一切特性。

◦ UNMST 描述了词本身的属性, 包括词, 词性。同时还描述了词, 短语的隐喻角色成分。从而从字面上表示了隐喻成分。

◦ 基于以上两种特性, MRDRL 从字面上解决了隐喻的复杂性: 包括隐喻成分的不确定性, 隐喻的递进复用等。能够包含语言学上隐喻的众多现象, 而不仅仅拘泥于 A VX B 等形式。同时 MRDRL 是面向计算机的语言, 方便计算机处理。

### 2.4 嵌套隐喻的字面语义表示

嵌套隐喻比较复杂, 因此其字面语义的形式化表示也更为复杂。由 MRDRL 表示嵌套隐喻字面语义的实例如图 6 所示。

例 2 教师是人类灵魂的工程师。〈EOS〉

例 2 是 2 层嵌套隐喻, 隐喻识别结果如图 7 所示。其中嵌套主隐喻是“教师是人类灵魂的工程师”, 嵌套次隐喻是“人类灵魂的工程师”。嵌套主隐喻的理解是建立在嵌套次隐喻的理解基础之上。两者之间的处理为嵌套语义映射, 如图 6 中椭圆部分所示。通过引进新的符号“?”来标识。显然这两者之间的关系不同于其他行之间的依存关系, 而是语义映射关系。

```
(Class 0: <EOS>: Met==2
(HED: 是, vx, Met_Role[1](Met_m): Left(1)
(SBV: 教师, n, Met_Role[1](Met_t): Left(1))
(VOB: ?, n, Met_Role[1](Met_v): Right(1))
(Class 24: ? : Met==1
(HED: 工程师, n, Met_Role[2](Met_v): Left(1)
(ATT: 的, ue, Met_Role[2](NULL): Left(1)
(DE: 灵魂, n, Met_Role[2](Met_t): Left(1)
(ATT: 人类, n, Met_Role[2](NULL): Left(1))))))
```

图 6 例句 2 的嵌套隐喻字面语义表示

```
1. class 0{教师}Met_t(是)Met_m(人类灵魂的工程师)Met_v
2. class 24 {人类灵魂}Met_t{工程师}Met_v
[4]灵魂_人类[3](ATT)
[5]的_[4]灵魂(DE)
[6]工程师_[5]的(ATT)
[2]是_[6]工程师(VOB)
[2]是_[1]教师(SBV)
[7]<EOS>_[2]是(HED)
```

图 7 例句 2 的隐喻识别结果

嵌套隐喻字面语义由若干语义项表示。嵌套主隐喻的结构关系除了依存关系外, 还有语义映射关系。下面就嵌套隐喻语义项及嵌套语义映射机制进行阐述。

定义 2.3.2 嵌套隐喻语义(Nested Metaphor Semantic, 简称为 NMS)也是一个集合,  $NMS = \{NMFT, NMHT, NMOT\}$ 。每一个嵌套隐喻语义项(Nested Metaphor Semantic Term, 简称为 NMST)也由三部分组成, 如同 UNMST, 见定义 2.3.1。

◦ 嵌套隐喻首项 NMFT,  $NMFT = \{(Class N; \langle EOS \rangle; Met == i), (Class N; ? : Met == i)\}$ 。(Class N;  $\langle EOS \rangle; Met == i$ ) 见定义 UNFT。(Class N; ? : Met == i) 标识嵌套次隐喻的类型及层数信息。“?”用来匹配嵌套主隐喻的隐喻角色成分和嵌套次隐喻。

◦ 嵌套隐喻中心项 NMHT,  $NMHT = (HED; WORD, POS, Met\_Role[i](x); LOC)$ 。与 UNMHT 大致相同。唯一区别是引入了隐喻角色数组 Met\_Role[i](x), 区分不同嵌套层上的隐喻角色。

◦ 嵌套隐喻其他项 NMOT,  $NMOT = (DR; WORD, POS, Met\_Role[i](x); LOC)$ 。与 UNMOT 基本相似。其中 Met\_Role[i](x) 为第 i 层上以 WORD 为根的子树所属的隐喻角色。WORD = {词本身, ?}。当 WORD = “?” 时, 其语义未知, 需要解释含有“?” 的 NMFT, 即解释嵌套次隐喻。下面我们给出嵌套语义映射在数学形式上的定义:

定义 2.3.3 嵌套语义映射:  $F(OS, TS)$ , 其中 OS 是嵌套主隐喻中隐喻角色成分的语义, TS 是嵌套次隐喻的语义, OS 和 TS 具有相同的句子成分。则  $(OS)_E(TS)$  表示 OS 通过 TS 间接释义, 对应在嵌套隐喻字面语义表示中体现为 (含有“?”的 NMOT) $_E$ (含有“?”的 NMFT)。

用 MRDRL 描述嵌套隐喻句子时新增以下几个特点:

- NMST 之间的结构组织不仅有依存关系, 还有语义映射关系。依存关系保持了句子结构, 语义映射关系映射了嵌套次隐喻和嵌套主隐喻中的隐喻角色成分。

- MRDRL 语言不仅从字面上可以表示 2 层隐喻嵌套, 同时适用于多层隐喻嵌套。嵌套的解决办法采用嵌套语义映射机制。

MRDRL 解决了隐喻字面语义表示。既不乏隐喻语言特性, 又便于计算机处理。下节将重点阐述隐喻角色依存表示语言生成算法。

### 3 隐喻角色依存表示语言生成算法

隐喻角色依存表示语言生成算法基于中心词驱动(Head-Driven)及合一(Unification)的思想。基于中心驱动的推导主要完成隐喻依存树的整体框架构成, 合一部分则是填补框架中必要信息。两者可以同时进行。

隐喻依存树构建包括两部分: 构建依存对模板和推导合一。以“脸蛋漂亮得像三色紫罗兰”这个句子所生成的隐喻角色依存表示语言(图 5)为例来说明。

#### 3.1 无嵌套的隐喻角色依存表示语言生成算法

构建依存对模板是推导合一的前提, 如图 8 中共有 7 对依存关系对, 所以可以构建 7 对依存对模板, 如图 8 所示。依存对模板给出了隐喻角色依存表示语言的框架。



图 8 依存对模板

#### 3.1.2 推导合一

基于中心词驱动和合一的思想, 根据中心词(Head)按深度优先扩展子树, 边扩展边合一。扩展的过程是句子结构生成的过程, 如图 9 所示。

合一的内容除了填充词, 词性, 句子成分相对位置关系外, 还需要填充隐喻角色值。在隐喻角色依存表示语言中, 合一的信息来自依存对模板和隐喻角色识别的结果。

无嵌套的隐喻角色依存表示语言生成算法描述如下:

- 先找出含有句子结束标志符 <EOS> 的依存对, 扩展子树, 并合一。
- 记录新的中心词, 按深度优先策略, 扩展新

中心词的子树, 并合一。

- 重复步骤 2, 直到无法扩展新的子树。

#### 3.2 嵌套的隐喻角色依存表示语言生成算法

嵌套隐喻角色依存表示语言生成算法的核心思想基本一样。所不同的是隐喻角色的标注, 以及语义映射关系的建立。图 10 给出了一个示例。

算法主要包括以下三个部分:

- 构建嵌套主隐喻依存对模板, 包括嵌套次隐喻依存对模板。
- 嵌套次隐喻的推导合一。
- 嵌套主隐喻的推导合一。

其中, 第 1 部分构建隐喻依存对模板的方法与

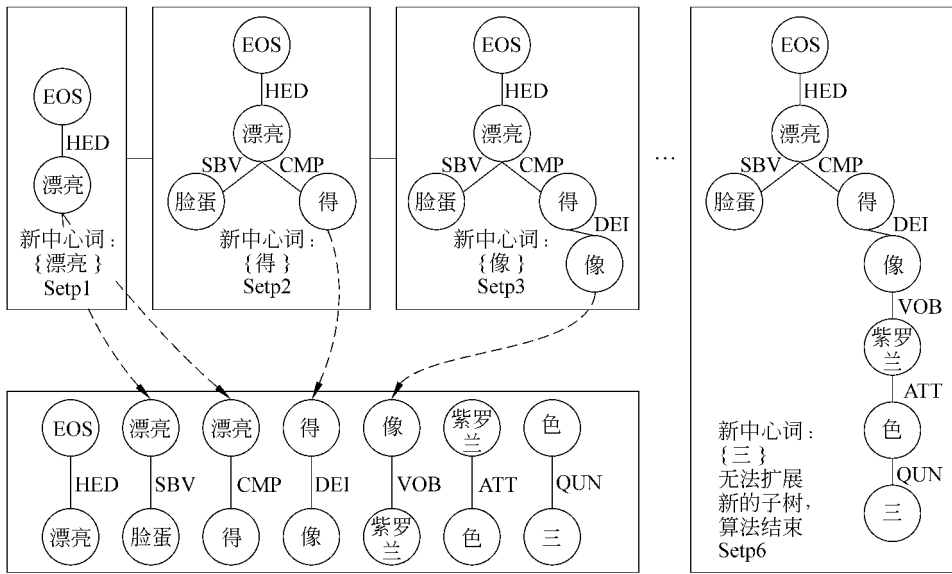


图9 句子结构生成示例

1. 教师是人类灵魂的工程师。 <EOS>class0{教师}Met\_t{是}Met\_m{人类灵魂的工程师}Met\_v  
 2. 人类灵魂的工程师 class24{人类灵魂}Met\_t{工程师}Met\_v

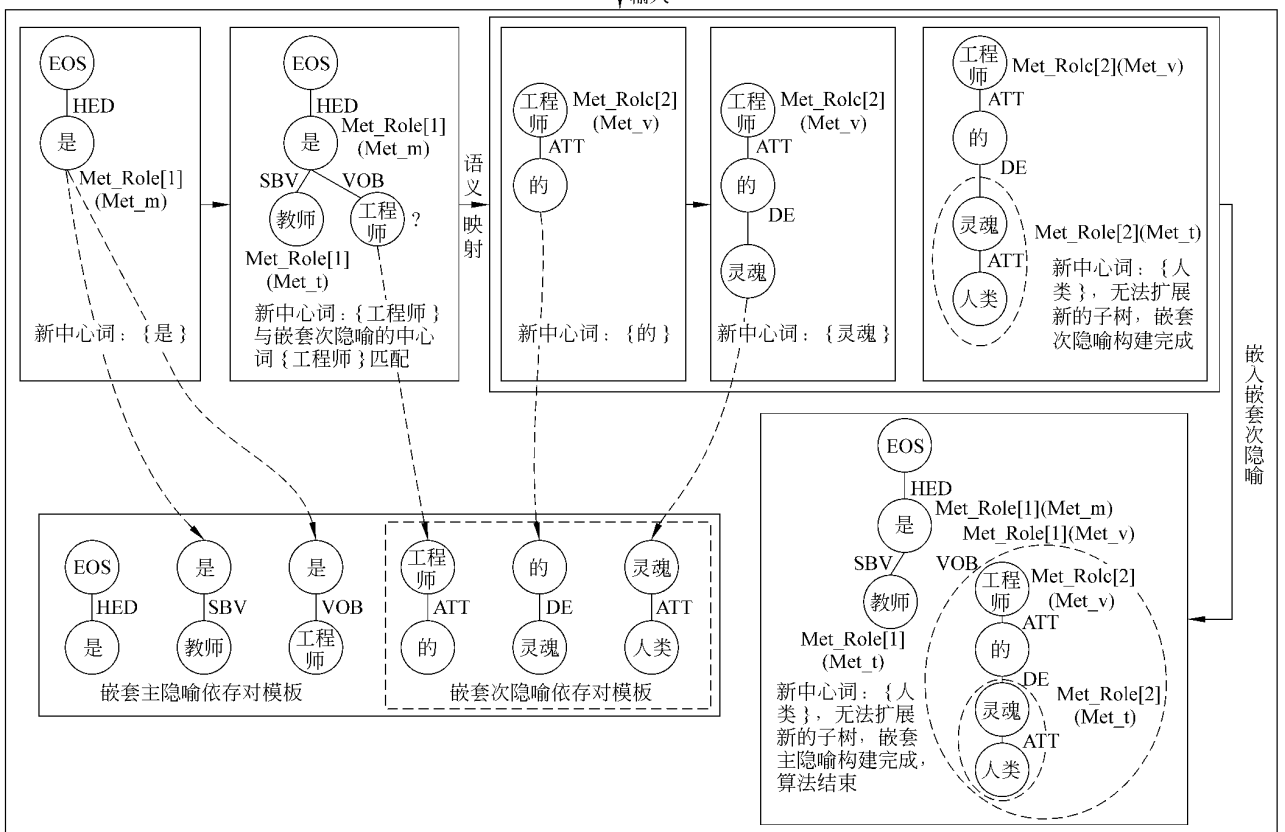


图10 嵌套的隐喻角色依存表示语言生成示例

无嵌套隐喻角色依存表示语言生成算法的方法一样。第2部分嵌套次隐喻的推导合一生成算法如下：

- 找出嵌套次隐喻的中心词节点，从依存模板

中找出以该中心词为首的依存对，扩展子树，并合一。

- 记录新的中心词，按深度优先策略，扩展新中心词的子树，并合一。

◦ 重复步骤 2, 直到无法扩展新的子树。

可见嵌套次隐喻的推导合一的方法与无嵌套隐喻的推导合一除了第一步有差别外, 其他并无区别。

嵌套主隐喻的推导合一, 生成算法如下:

◦ 先找出含有句子结束标志符 <EOS> 的依存对, 扩展子树, 并合一。

◦ 记录新的中心词, 按深度优先策略方法进行扩展。

如果新的中心词不是嵌套次隐喻的中心词,

则扩展新中心词的子树, 并合一。

否则

建立语义映射关系, 如图 6 中椭圆所示部

分, 嵌入嵌套次隐喻。

◦ 重复步骤 2, 直到无法扩展新的子树。

### 3.3 实验结果与讨论

实验从《读者》隐喻句语料库<sup>[10]</sup>中抽取了 500 个隐喻句进行隐喻字面语义表示的实验。其中一半为无嵌套隐喻句, 另一半为嵌套隐喻句。实验表明隐喻角色依存表示语言生成算法是一种精确算法, 对所有正确识别出来的隐喻句子, 即识别出来的可理解或者是可接受的隐喻句, 都能精确地表达其字面语义信息。表 1 给出了部分隐喻句子的字面语义表示, 其中 1, 2 句子是无嵌套隐喻句, 3, 4 句子是嵌套隐喻句。

表 1 隐喻字面语义表示实验示例

序号	例句	隐喻识别结果	字面语义表示结果
1	祖国像母亲。	Class 0{祖国}Met_t{像}Met_m {母亲}Met_v [ 2 ] 像 [ 3 ] 母亲 (VOB) [ 2 ] 像 [ 1 ] 祖国 (SBV) [ 4 ] <EOS> [ 2 ] 像 (HED)	(Class 0; <EOS>: Met== 1 (HED; 像, vx, Met_Role(Met_m); Left(1)) (SBV; 祖国, n, Met_Role(Met_t); Left(1)) (VOB; 母亲, n, Met_Role(Met_v); Right(1))))
2	眼泪像大堤崩溃。	Class 5{眼泪}Met_t{像}Met_m {大堤崩溃}Met_v [ 4 ] 崩溃 [ 3 ] 大堤 (SBV) [ 2 ] 像 [ 4 ] 崩溃 (VOB) [ 2 ] 像 [ 1 ] 眼泪 (SBV) [ 5 ] <EOS> [ 2 ] 像 (HED)	(Class 5; <EOS>: Met== 1 (HED; 像, vx, Met_Role(Met_m); Left(1)) (SBV; 眼泪, n, Met_Role(Met_t); Left(1)) (VOB; 崩溃, vg, Met_Role(Met_v); Right(1)) (SBV; 大堤, n, Met_Role(NULL); Left(1))))
3	她们是乐器的翅膀。	1 Class 0{她们}Met_t{是}Met_m {乐器的翅膀}Met_v 2 Class 24{乐器}Met_t{翅膀}Met_v [ 4 ] 的 [ 3 ] 乐器 (DE) [ 5 ] 翅膀 [ 4 ] 的 (ATT) [ 2 ] 是 [ 5 ] 翅膀 (VOB) [ 2 ] 是 [ 1 ] 她们 (SBV) [ 6 ] <EOS> [ 2 ] 是 (HED)	(Class 0; <EOS>: Met== 2 (HED; 是, vx, Met_Role[ 1 ] (Met_m); Left(1)) (SBV; 她们, n, Met_Role[ 1 ] (Met_t); Left(1)) (VOB; ?, n, Met_Role[ 1 ] (Met_v); Right(1)) (Class 24; ?: Met== 1 (HED; 翅膀, n, Met_Role[ 2 ] (Met_v); Left(1)) (ATT; 的, ue, Met_Role[ 2 ] (NULL); Left(1)) (DE; 乐器, n, Met_Role[ 2 ] (Met_t); Left(1))))))
4	坦率是批评最灿烂的宝石。	1 Class 0{坦率}Met_t{是}Met_m {批评最灿烂的宝石}Met_v 2 Class 28{批评}Met_t{宝石}Met_v {最灿烂}Met_g [ 5 ] 灿烂 [ 4 ] (ADV) [ 6 ] 的 [ 5 ] 灿烂 (DE) [ 7 ] 宝石 [ 6 ] 的 (ATT) [ 7 ] 宝石 [ 3 ] 批评 (ATT) [ 2 ] 是 [ 7 ] 宝石 (VOB) [ 2 ] 是 [ 1 ] 坦率 (SBV) [ 8 ] <EOS> [ 2 ] 是 (HED)	(Class 0; <EOS>: Met== 2 (HED; 是, vx, Met_Role[ 1 ] (Met_m); Left(1)) (SBV; 坦率, n, Met_Role[ 1 ] (Met_t); Left(1)) (VOB; ?, n, Met_Role[ 1 ] (Met_v); Right(1)) (Class 28; ?: Met== 1 (HED; 宝石, n, Met_Role[ 2 ] (Met_v); Left(1)) (ATT; 批评, n, Met_Role[ 2 ] (Met_t); Left(1)) (ATT; 的, ue, Met_Role[ 2 ] (NULL); Left(2)) (DE; 灿烂, n, Met_Role[ 2 ] (Met_g); Left(1)) (ADV; 最, u, Met_Role[ 2 ] (NULL); Left(1))))))

此外, 隐喻识别的结果显示了一定的有效性, 其中指称型隐喻识别封闭测试的性能接近 80%, 动词为中心的搭配型隐喻识别封闭测试的性能也接近 70%<sup>[2]</sup>。在此基础上, 我们用准确率和召回率来评价隐喻字面语义表示结果:

$$\text{准确率} = \frac{\text{正确表示隐喻字面语义的隐喻句总数}}{\text{识别出的隐喻句总数}}$$

$$\text{召回率} = \frac{\text{正确表示隐喻字面语义的隐喻句总数}}{\text{实际隐喻句总数}}$$

$$F = \frac{2 \times \text{准确率} \times \text{召回率}}{\text{准确率} + \text{召回率}}$$

表 2 结果评价

		准确率	召回率	F 值
无嵌套隐喻	(250 句)	77.77%	70.00%	73.68%
嵌套隐喻	(250 句)	76.40%	65.00%	70.20%
隐喻句	(500 句)	77.17%	67.50%	72.01%

从实验结果看, 该实验具有一定的有效性。其中无嵌套隐喻句字面语义表示效果接近 75%, 嵌套隐喻接近 70%, 两者之间没有很大差别。很显然, 如果在保证句法分析器高效性的基础上, 隐喻识别的性能进一步提高, 将会扩大字面语义表示的隐喻句子的覆盖面, 隐喻理解的性能也将会进一步提高。

#### 4 总结

本文在全面分析汉语隐喻特点的基础上, 对汉语隐喻字面语义进行了界定, 指出汉语隐喻字面语义包括浅层语义信息和隐喻信息, 并给出计算机形式化表示: 隐喻角色依存表示语言, 分别对无嵌套隐喻和嵌套隐喻进行描述, 并提出了隐喻角色依存表示语言生成算法。

隐喻字面语义的形式化表示是计算机隐喻理解的前提。只有对隐喻字面语义进行了形式化定义, 才能有效地进行下一步工作, 为将隐喻的字面语义转译到真实意义, 实现隐喻的机器理解奠定良好的

工作基础。

本文提出的汉语隐喻角色依存表示语言已应用到 CHMeta-汉语隐喻识别与解释系统中, 实验表明该方法具有切实可行性。这种以文法来组织句子结构, 以隐喻信息来标注隐喻成分的字面语义表示方法稍做修改也可适用于其他文法和汉语隐喻识别算法。不过, 由于该方法是建立在依存标注和隐喻识别基础之上, 故其准确率受两者影响较大。如何提高隐喻字面语义获取的正确率将是我们今后研究工作的一个重点。

#### 参考文献:

- [1] 戴帅湘, 周昌乐. 隐喻计算模型及其在隐喻分类上的应用[J]. 计算机科学, 2005, 32(05): 159-163, 166.
- [2] 杨芸. 汉语隐喻识别与解释计算模型研究[D]. 博士学位论文, 厦门: 厦门大学, 2008.
- [3] 李剑锋. 面向隐喻计算的汉语语义超常搭配识别模型研究[D]. 硕士学位论文, 厦门: 厦门大学, 2008.
- [4] 李剑锋, 杨芸, 周昌乐. 一种基于汉语隐喻依存句法树的嵌入式树匹配算法[J]. 厦门大学学报, 2008, 47(04): 500-504.
- [5] 杨芸, 周昌乐. 隐喻的语言形式特征及其对隐喻机器理解研究的影响[J]. 心智与计算, 2007(04): 458-464.
- [6] Peter Hellwig. Extended Dependency Unification Grammar[J]. Functional Description of Language, 1993: 67-84.
- [7] Joakim Nivre. Dependency grammar and dependency parsing[R]. Technical Report MSI report 05133, Vaxjo University: School of Mathematics and Systems Engineering, 2005.
- [8] Debusmann, R. An introduction to dependency grammar[J]. University of the Saarland, 2000.
- [9] Kirvinen, Timo and Pasi Tapanainen. Towards an implementable dependency grammar[C] // Proceedings of COLING/ACL'98 Workshop on Processing Dependency-Based Grammars. 1998: 1-10.
- [10] 李剑锋, 杨芸, 周昌乐. 面向隐喻计算的语料库研究和建设[J]. 心智与计算, 2007(01): 142-146.