哲学与逻辑新探

从类型逻辑语法角度审视DRT的组合性问题*

石运宝

【提　要】刻画自然语言的语义可以先将自然语言翻译为逻辑语言，然后再给出模型，如蒙塔古的PTQ（1973）以及本文关注的DRT；也可以直接为自然语言给出模型。鉴于前者的翻译介于自然语言和模型之间，于是被称为“中间层”，此层面为句法—语义接口的语义部分。所谓组合性，是指从自然语言到该中间层，从中间层再到模型都存在“句法—语义”对应。关于组合性问题，学界已有的解决方案是从自然语言的句法结构和中间层的句法结构连同中间层的语义解释入手，规范地梳理句法—语义对应问题。本文介绍一种经典方案，即将范畴语法与DRT相结合，之后给出改进思路。

【关键词】标准DRT　组合性　类型逻辑
〔中图分类号〕B81〔文献标识码〕A〔文章编号〕1000－2952（2017）02－0021－08

一、问题的提出

标准DRT虽然相对于谓词逻辑和蒙塔古语法在处理驴子句时显得优越，但是随着人们研究发现，标准DRT其实设计比较“粗糙”，其句法和语义都不够完善。泽万特（Zeevat，1989）指出，虽然DRT解决了驴子句等问题，但初创期的DRT的句法系统没法像蒙塔古语法和其他表征自然语言意义的方案那样，存在一个精确的“语法”。这里的语法是指符合组合性的、具有数理精确的形式化特征的系统，即像蒙塔古语法那样兼具句法代数和语义代数，且句法代数和语义代数之间存在同态。甘莫特（Gamut，1991）认为，标准DRT采取真值条件语义学显得语义过于狭窄，导致DRT框架出现违反组合性的实例。[①]缪斯肯斯（Muskens）[②]认为改良标准DRT是为了使得DRT与（蒙塔古式）组合性相结合，使得组合性DRT既具有标准DRT的优势，又兼具组合性的特质。范·艾杰克（Jan van Eijk）和汉斯·坎普（Hans Kamp）则提到，标准DRT经常被批评不符合组合性，这种批评有两层含义，一是指标准DRT无法为自然语言提供直接的组合语义学，

* 该论文受到重庆市社会科学规划项目“范畴语法与双指针投射DRT相结合的理论研究”（2015BS089）资助。

① L. T. F. Gamut, *Logic, Language and Meaning, vol. I: Introduction to Logic, vol. II: Intensional Logic and Logical Grammar*, Univ. of Chicago Press: 1991, p. 287.

② R. A. Muskens, Combing Montague Semantics and discourse Representation, *Linguistics and Philosophy*, vol. 19, 1996, pp. 143－186.

还有一层含义是指，标准DRT本身的形式不符合组合性。范·艾杰克和汉斯认为，鉴于标准DRT最初的设计，前一种质疑是有道理的，而后一种质疑则是无道理的。他们认为，是出于标准DRT本身的问题，需要进行改良或修正以符合组合性，来应对所谓的组合性问题。他们确实给出了有力的回应，并且持续关注这个问题，分别在1996年和2010年给出两个版本的阐述来回应这个问题。①

从自然语言到逻辑表达式这一层面是否存在严格的"句法—语义"对应而言，蒙塔古语法已做得很好，逐步建立起每个句法成分的语义，然后将逐个成分的语义组合起来，构成更大成分的语义。DRT则无法为每个句法成分都提供一个语义，然后将这些成分的语义组合起来。具体来说，在句子层面之下，不是每个表达式都对应特定的DRS，有的对应DRS条件。从计算的角度来讲，如果不能为每个成分指派特定意义，那么便无法从较小成分计算出较大成分的意义。在句子层面之上，即文本或话语层面，自然语言往往以句子序列方式呈现，如何确认句法上的"部分"，如何从句法—语义对应的角度为句法上确认的成分指派相应的DRS，标准DRT都无法回应。标准DRT展示了它动态的一面，却忽视了语义可计算等重要性质。严格从组合性角度审视，话语或篇章分别缺乏特定句法成分和语义类型。具体来说，针对已经处理过的较小话语或篇章，没有相对应的DRS作为语义，也没有相应的句法成分来区分不同范畴的成分。就另外一个层面的对应而言，即从DRS到DRS本身的解释上，问题更多。首先，由于DRT最初的句法构造的特殊性，分为DRS和DRS条件两部分，DRS与DRS条件不像谓词逻辑合式公式那样，由初始合式公式生成复杂合式公式。DRS不是由简单的DRS组成的，而是由DRS条件组成的，DRS条件由原子DRS条件和逻辑联结词加DRS构成的复杂DRS条件构成。这样的DRS和DRS条件二分的情形导致DRS本身的构造无法满足组合性。

对于DRT的组合性问题，表述不尽相同。综上所述，由于DRT在自然语言和模型论解释之间存在一个DRS层，即话语表现结构，所以DRT的组合性问题归结为从自然语言到DRS要存在句法—语义对应，然后从DRS到模型论语义学也要遵循句法—语义对应。这便是标准DRT的问题所在，即两方面都需要从组合性角度进行审视。这里面还有一个非常重要的问题，即彻底遵循组合还是只是做到某一层面句法语义对应即可。所谓的彻底遵循组合性，是说从句子层面之下便要遵循组合性，句法上从短语等成分组合成句子，然后由句子和话语组合成更大的话语；而语义上，从短语等对应的语义一直计算出话语的语义。无论是彻底的组合还是仅仅从句子层面审视组合性问题，标准DRT都没做到。DRS结构跟蒙塔古语法的内涵逻辑表达式和类型逻辑语法中的加标公式功能有些相似，都是语义表征层，但是DRS却没有后两者组合得那么好。虽然后两者也存在差异，即蒙塔古语法是句法规则和语义规则一前一后，而类型逻辑语法则平行推演，但是他们二者的组合性都较DRT优良。在后两者的对比中发现，由于标准DRT缺乏组合性，进而无法得到一些重要的性质，比如自下而上的组合，以及构建一个完善的推理系统等就无法实现。

二、一种经典方案解读

缪斯肯斯给出了一个模式，并非像蒙塔古语法那样庞大、细致地分析诸多现象。缪斯肯斯的方案是遵循组合性的典范，主要体现在句法—语义对应这层含义，该方案较为完善地实现了二者的并行推演。通过DRS到类型逻辑公式的转化，巧妙将DRS合并转化为λ—演算。

缪斯肯斯（1994）② 的工作表明，简单文本句法上按照兰贝克（Lambek）范畴语法进行分析，那么针对每个句法范畴都存在一个DRS语

① J. V. Eijck and H. Kamp. Representing discourse in context. In J. van Benthem and A. T. Meulen, editors, *Handbook of Logic and Language*, Amsterdam: Elsevier, 1997, pp. 179—237.

② R. A. Muskens, Categorial Grammar and Discourse Representation Theory. *Proceedings of COLING* 94, Kyoto, Japan, 1994, pp. 508—514.

义与之对应。缪斯肯斯为文本进行意义指派使用的是柯里—霍华德—范·本特姆对应定理。

范·本特姆（1986）[①] 的工作证明柯里—霍华德对应（每个句法证明操作匹配 λ 词项），可以用于为兰贝克范畴语法和蒙塔古语义之间“搭建桥梁”。兰贝克演算的每个证明都配上一个 λ 词项，并且范·本特姆阐明了这个 λ 词项作为一个“清单”详细记载这个证明结论的语义是如何一步步根据其组成部分的语义计算得来。

通常，以上述方式获得的语义是蒙塔古（1973）工作的外延性扩充（亨德里克斯（1993）将这种方法进行了推广，以适用于整个内涵片段）。在回指照应等现象提出后，蒙塔古框架缺乏动态性等来应对这类问题。针对这些问题，汉斯等提出 DRT 语义理论，该理论提供了不同于蒙塔古语法的理论方法，来处理驴子句等涉及的回指照应现象。此后，许多学者采用该理论作语义研究框架。但这种理论的问题也逐渐呈现出来。巴维斯（1987）和鲁思（1987）观察到 DRT 未像蒙塔古语法那样将名词统一处理为广义量词，并且汉斯等（1993）给出标准 DRT 无法像蒙塔古语法那样漂亮地处理任意范畴的并列等问题。

缪斯肯斯在 DRT 理论框架和蒙塔古语法框架外提出第三个语义框架，[②] 即合并这两大语义框架。柯里—霍华德—范·本瑟姆对应方法为兰贝克证明提供 λ 词项作为证明的语义。由于标准 DRT 句法并未与兰贝克范畴语法形成句法—语义接口（interface），即在处理自然语言语句法时标准 DRT 并未采取兰贝克范畴语法作为生成方式，标准 DRT 自然不像柯里—霍华德—范·本瑟姆对应那样形成一个范畴语法的推演伴随一个 λ 演算进而形成并行推演。

泽万特（1989），格伦内捷克和斯托克霍夫（1990，1991）所做的工作便是在缪斯肯斯所说的第三个语义框架内所做的工作。但是这些工作都没有将 DRT 与类型逻辑相结合，而缪斯肯斯通过结合 DRT 与类型逻辑使得 DRT 与兰贝克演算结合。缪斯肯斯之所以做到这些，是因为他观察到，DRSs 的意义是一阶可定义的，因此 DRSs 可以用类型逻辑的一阶部分进行表达。这意味着，可以将名词短语处理为广义量词，并且可以统一处理任意范畴的并列问题。

缪斯肯斯（1994）的主要工作是将 DRT 与兰贝克范畴语法相结合。下面分五个主要部分介绍他的工作。先介绍英语这门自然语言与兰贝克证明之间的关联，接着为兰贝克证明匹配 λ 词项；第三部分用类型逻辑定义 DRSs；第四部分用定义出的 DRSs 去替换兰贝克证明的语义（即 λ 词项），最后为这些 DRSs 提供真值条件。

（1）从英语片段到兰贝克证明

兰贝克演算不做具体介绍，这里只提及用到的部分。

定义 1：范畴是通过下列条款得到的最小集合：

＜1＞基础范畴为 txt（表示“文本”），s（表示“句子”），n（表示“名词”）和 cn（表示“通名”）。

＜2＞派生范畴可由下述步骤获得：若 A 和 B 为范畴，则（A/B）和（B \ A）都是一个范畴。

与＜1＞和＜2＞对应的句法规则为＜1＞′和＜2＞′：

＜1＞′若 α 为范畴为（A/B）的表达式，β 为范畴为 B 的表达式，则 αβ 为范畴为 A 的表达式。

＜2＞′若 α 为范畴为 B 的表达式，β 为范畴为（B \ A）的表达式，则 αβ 为范畴为 A 的表达式。

定义 2：一个序列是形如 T ⊢c 的表达式，其中 T 是有穷非空的范畴序列（前件），c 是一个范畴（后件）。

定义 3：一个序列是可证明的，当且仅当这个序列采用下述根岑规则证得：

---

① J. V. Benthem, *Essays in Logical Semantics*, Amsterdam: Springer, 1986, pp. 153－158.

② 缪斯肯斯在其《合并蒙塔古语义学和话语表征》（Combing Montague Semantics and Discourse Representation, 1996）中使用类型逻辑，保留 DRS 语言，并在蒙塔古语法框架内完组合 DRT 的构建，实现了自然语言语义学的两大框架的沟通，即蒙塔古语法框架和 DRT 框架。

$$\frac{}{c\vdash c}[AX]\qquad \frac{T,b\vdash a}{T\vdash a/b}[/R]\qquad \frac{T\vdash b\quad U,a,V\vdash c}{U,a/b,T,V\vdash c}[/L]$$

$$\frac{b,T\vdash a}{T\vdash b\backslash a}[\backslash R]\qquad \frac{T\vdash b\quad U,a,V\vdash c}{U,T,b\backslash a,V\vdash c}[\backslash L]$$

下面的例子表明，序列（s/（n \ s））/cn，cn，（n \ s）/n，（（s/n） \ s）/cn，cn ├s 是一个可以根据上述规则可证的序列。如果该序列的前件分别是“a”，“man”，“adores”，“a”，“woman”的范畴，那么下面的证明便是给定序列的推演，这个推演可以解释为这些词按照上述顺序组成的序列“a man adores a woman”的范畴为 s。

$$\dfrac{\dfrac{}{cn\vdash cn}\quad \dfrac{\dfrac{}{cn\vdash cn}\quad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{n\vdash n}\quad \dfrac{\dfrac{}{n\vdash n}\quad \dfrac{}{s\vdash s}}{n,n\backslash n\vdash s}[\backslash L]}{n,(n\backslash s)/n,n\vdash s}[/L]}{n,(n\backslash s)/n\vdash s/n}[/R]\quad \dfrac{}{s\vdash s}}{n,(n\backslash s)/n,(s/n)\backslash s\vdash s}[\backslash L]}{(n\backslash s)/n,(s/n)\backslash s\vdash n\backslash s}[\backslash R]\quad \dfrac{}{s\vdash s}}{s/(n\backslash s),(n\backslash s)/n,(s/n)\backslash s\vdash s}[/L]}{(s/(n\backslash s))/cn,cn,(n\backslash s)/n,(s/n)\backslash s\vdash s}[/L]}{(s/(n\backslash s))/cn,cn,(n\backslash s)/n,((s/n)\backslash s)/cn,cn\vdash s}[/L]$$

图 1　“a man adores a woman”的证明

（2）从兰贝克证明到语义匹配

证明论的成果表明，在某个证明和 λ 词项之间存在严格对应。这种对应是从柯里—霍华德对应定理给出的。范·本特姆（1986）观察到，其实在兰贝克演算和 λ 词项之间也存在上述对应，而这种对应是通过柯里—霍华德—范·本瑟姆对应定理给出的。这种对应的背后的思想是，兰贝克演算的每条规则都对应一个语义规则，并且，针对每个句法的证明，都存在一个与该证明同构的语义序列树，语义序列用 $T'\vdash\gamma$ 表示，其中 T' 是变元序列，γ 是一个 λ 词项，该词项中的自由变元恰好是 T' 中的变元。与定义 3 中的根岑规则对应的语义规则如下：

$$\frac{}{x\vdash x}[AX]\qquad \frac{T',v\vdash\alpha}{T'\vdash\lambda v.\alpha}[/R]\qquad \frac{T'\vdash\beta\quad U',u,V'\vdash\gamma}{U',w,T',V'\vdash\gamma[u:=w(\beta)]}[/L]$$

$$\frac{v,T'\vdash\alpha}{T'\vdash\lambda v.\alpha}[\backslash R]\qquad \frac{T'\vdash\beta\quad U',u,V'\vdash\gamma}{U',T',w,V'\vdash\gamma[u:=w(\beta)]}[\backslash L]$$

其中，$\gamma[u:=w(\beta)]$ 是用 w（β）替换 γ 中的 u 所获得的结果。公理［AX］、［/L］和［ \ L］都引入了新变元，但是这些规则需要满足一些前提条件。第一个条件是，这三个规则中的变元必须是新引入的，即不在前面某处被引入。假定存在一个函数 TYPE，它将范畴映射到语义类型中，满足 TYPE（a/b）＝TYPE（b \ a）＝（type（b），type（a））。第二个需要满足的条件是，如果让公理 x ├x 对应兰贝克证明中的公理 c ├c，则公理 x ├x 中的变元 x 的类型必须是 TYPE（c）。此外，［/L］和［ \ L］中引入的变元 w 的类型必须是（type（b），type（a）），且 a/b 或 b \ a 必须是相应的句法证明中起作用的范畴。

有了这些语义规则，便可以构建与图 1 对应的语义树了：

$$\dfrac{\dfrac{}{P'\vdash P'}\quad \dfrac{\dfrac{}{P\vdash P}\quad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{v'\vdash v'}\quad \dfrac{v\vdash v\quad p\vdash p}{v,P''\vdash P''(v)}[\backslash L]}{v,R,v'\vdash R(v')(v)}[/L]}{v,R\vdash\lambda v'.R(v')(v)}[/R]\quad \dfrac{}{p'\vdash p'}}{v,R,Q\vdash Q(\lambda v'.R(v')(v))}[\backslash L]}{R,Q\vdash\lambda v.Q(\lambda v'.R(v')(v))}[\backslash R]\quad \dfrac{}{p''\vdash p''}}{Q',R,Q\vdash Q'(\lambda v.Q(\lambda v'.R(v')(v)))}[/L]}{D,P,R,Q\vdash D(P)(\lambda v.Q(\lambda v'.R(v')(v)))}[/L]}{D,P,R,D',P'\vdash D(P)(\lambda v.D'(P')(\lambda v'.R(v')(v)))}[/L]$$

图 2　“a man adores a woman”的语义树

这棵树根部的序列给出一个清单，这个清单表明，一旦给出“a man adores a woman”的各构成部分的语义，便可计算整个句子的语义。下面给出“a man adores a woman”各构成部分的语义。假定“a”的翻译为 λP′λP∃x（P′（x）∧P（x）），类型为（e t）（（e t）t），man 的翻译为“man”，类型为（e t），adores 的翻译为“adore”，类型为（e（e t）），woman 的翻译为“woman”，类型为（e t）。给定各个词的语义，那么就可以按照所给的清单计算出“a man adores a woman”的语义了。用 λP′λP∃x（P′（x）∧P（x））替代后件中的 D 和 D′，用 man、adore、woman 分别替 P、R、P′，则得到的 λ 词项经过化简为：∃x（man（x）∧ ∃y（woman（y）∧adores（y）（x）））。

（3）用类型逻辑定义 DRSs

虽然类型逻辑涵盖一阶与高阶逻辑，但是假如采用几个公理，只用类型逻辑的一阶部分便可定义 DRSs，或者称之为在类型逻辑框架内

模拟出 DRSs。如果将 DRT 看做是类型逻辑的一个一阶片段，那么便可以将定义出范畴语法和 DRT 之间的接口，即形成类似于范畴语法与类型逻辑那样并行推演的格局，只不过这里的语义表征层换成 DRSs，而不再是标准的类型逻辑语言。

如果能做到范畴语法与 DRT 之间的并行推演，则 DRT 的组合性问题自然得到解决。因为范畴语法本身是遵循组合性的，从自然语言到范畴语法也是严格按照组合性进行的，而范畴语法到 DRT 实现并行推演自然是遵循组合性的。这样，DRT 的组合性问题得到彻底解决。下面先在类型逻辑框架内模拟 DRT 语言。

在所使用的多体类型逻辑中，基本类型有四类：一是实体，类型为 e；二是真值 1 和 0，类型为 t；三是寄存器，类型为 π；四是状态，类型为 s。缪斯肯斯喜欢称寄存器为“鸽子窝”(pigeon-holes)，[①] 用于存储话语所指内容，可以把它看作是只存储一个物体的单元格，而不论物体多大。状态可以理解为当前所有鸽子窝中所有“住户”排列成的一个序列。动态理论中的状态类似于理论计算机科学家所使用的状态，后者所谓的状态是指某个程序在其执行的某个阶段当下所有变元值的列表。

缪斯肯斯定义了一个 V，V 是一个固定的非逻辑常项，V 的类型为（π（s e)），V（u)(i) 表示状态 i 下寄存器 u 内类型为 e 的项。运算上来说，V（u）（i）先结合类型为 π 的寄存器 u，再结合类型为 s 的 i，进而生成类型为 e 的项，即某个话语所指的值。令 i［u_1… u_n］j 是 $\forall v[(u_1 \neq v) \wedge \ldots \wedge (u_n \neq v) \rightarrow V(v)(i) = V(v)(j)]$ 的缩写，所表达的含义是，状态 i 和状态 j 的差异至多在于寄存器 $u_1 \cdots u_n$ 上的值不同，除此之外，状态 i 和 j 下的任何寄存器 v 上的值都相同。i［ ］j 是指两个状态下寄存器上的值都相同，即公式 $\forall v(V(v)(i) = V(v)(j))$ 所传达的内容。系统所采用的公理如下：

AX1　$\forall i \forall v \forall x\ \exists j\ (i[v]j \wedge V(v)(j) = x)$；

AX2　$\forall i \forall j\ (i[\ ]j \rightarrow i = j)$；

AX3　$u \neq u'$，对于任意两个不同的话语所指（类型为 π 的常项）u 和 u′。

公理 1 是说，对于任意的状态 i 任意的寄存器 v 和任意的对象 x 来说，一定存在另外一个状态 j，j 不同于 i 的地方在于给定对象 x 是给定寄存器上的值，除此之外 i 和 j 相同。这很容易让人联想起用动态观点看待指派，指派 h［x］g 是指状态 h 不同于状态 g 之处在于，在状态 h 下某个个体被指派给了 x，除此之外两个指派没有差异，其中的道理与公理 1 是类似的。公理 2 是说，如果两个状态下所有寄存器上的值都相同，那么这两个状态便是相同的。公理 3 是说，不同的话语所指对应不同的寄存器，所以更新某个话语所指的值并不改变其他话语所指的值。

下面用类型逻辑语言模拟 DRT 语言。固定类型 s 和变元 i，固定的目的是让它们不再代表别的含义。定义（u）† = V（u）（i），对于所有话语所指（类型为 π 的常项）；定义（t）†= t，对于每个类型为 e 的项 t；并采取下面的记法约定：

将类型逻辑中的 $\lambda i P(\tau)^{\dagger}$ 记为（DRT 语言中的）$P\tau$；

将类型逻辑中的 $\lambda i(R(\tau_1)^{\dagger}(\tau_2)^{\dagger})$ 记为（DRT 语言中的）$\tau_1 R \tau_2$；

将类型逻辑中的 $\lambda i((\tau_1)^{\dagger} = (\tau_2)^{\dagger})$ 记为（DRT 语言中的）τ_1 is τ_2。

其中，P 为类型为（e t）的项，R 是类型为（e（e t））的项，τ 是一个话语所指或类型为 e 的项。有了“$P\tau$”、“$\tau_1 R \tau_2$”、“τ_1 is τ_2”DRT 语言中的所谓 DRS 条件便有了，它们的类型为（s t)，意味着给定状态 s 才能确定其在该状态下的值。此外，还缺乏的是复杂条件和 DRSs：

将类型逻辑中的 $\lambda i \neg \exists j \Phi(i)(j)$ 记为（DRT 语言中的）not Φ；

将类型逻辑中的 $\lambda i \exists j(\Phi(i)(j) \vee \Psi(i)(j))$ 记为（DRT 语言中的）Φ or Ψ；

将类型逻辑中的 $\lambda i \forall j(\Phi(i)(j) \rightarrow \exists k \Psi(j)(k))$ 记为（DRT 语言中的）$\Phi \Rightarrow \Psi$；

将类型逻辑中的 $\lambda i \lambda j(i[u_1, \ldots, u_n]j \wedge \gamma_1(j) \wedge \ldots \wedge \gamma_m(j))$ 记为（DRT 语言中的）

① R. A. Muskens，Categorial Grammar and Discourse Representation Theory. *Proceedings of COLING 94*，Kyoto，1994，p. 511.

$[u_1 \ldots u_n \mid \gamma_1, \ldots, \gamma_m]$；

将类型逻辑中的 $\lambda i \lambda j \exists k(\Phi(i)(k) \wedge \Psi(k)(j))$ 记为（DRT 语言中的）$\Phi;\Psi$。

其中，Φ 和 Ψ 是类型为 $(s(st))$ 的任意项；DRSs 的类型也是 $(s(st))$，说 Φ 和 Ψ 是任意项，从上面类型逻辑与 DRT 语言的对应来看，Φ 和 Ψ 代表任意 DRSs；γ 代表任意一个 DRS 条件，是类型为 (st) 的项；$[u_1 \ldots u_n \mid \gamma_1, \ldots, \gamma_m]$ 是 DRSs 的线性记法；最后一个条款则不是标准 DRT 语言中的合式表达式，它是为了组合地处理自然语言语义而给出的，借助于“;”算子，便能针对自然语言的短语、句子、文本给出组合的翻译，并且，就 DRSs 自身构造来说，经“;”联结的 DRSs 序列仍是一个 DRS，这便彻底遵循了组合性。① 该算子取自于命令式程序语言（与当今的函数型程序语言相对）。

这样，标准 DRT 语言中的条件和 DRSs 都有了，可以说，DRT 语言作为类型逻辑的一阶片段便刻画完毕。归功于来自计算机科学的状态、寄存器等术语，该系统所使用的类型逻辑能刻画、对接它的一阶片段：DRT 语言。类型逻辑是严格按照组合性进行运算的，类型逻辑的一阶片段当然也是遵循组合性的。

令 $\vec{u}$ 和 $\vec{u}'$ 为话语所指的序列，$\vec{\gamma}$ 与 $\vec{\gamma}'$ 是条件序列，则下面的合并引理成立：

合并引理：如果 $\vec{u}'$ 不出现在任意中 $\vec{\gamma}'$，则

$\models AX\ [\vec{u} \mid \vec{\gamma}];\ [\vec{u}' \mid \vec{\gamma}'] = [\vec{u}\vec{u}' \mid \vec{\gamma}\vec{\gamma}']$

注意，这里的合并引理之所以成立，是因为合并引理的前提条件和之前的公理 3 保证，不同寄存器上的话语所指均不同，这样不会出现范·艾杰克和汉斯·坎普所提及的“Px；x；Qx”、“x；Px；x；Qx”情形，自然也就不用重命名策略；也不会出现因为顾及顺序而在合并的时候去掉第二个 DRS 的话语所指集。

类型逻辑中模拟出的 DRT 语言的语义与格伦内捷克和斯托克霍夫（1991）年的工作中给出的动态谓词逻辑的语义有共同点，也存在差异，这里不做深入探讨。

（4）兰贝克证明与 DRSs 构建并行推演

前面的工作表明，DRT 语言可以作为类型逻辑的一个片段，现在可以将兰贝克证明与 DRSs 联结起来，正如兰贝克证明与类型逻辑存在句法—语义接口一样。所需要做的工作是定义一个函数 TYPE 并为某个英语片段给出一个特定的词库，剩下的工作在为证明指派意义过程中得到处理。TYPE 在前面也提到过，它是从范畴到类型的函数，其定义如下：

（1）TYPE（txt）＝ TYPE（s）＝ $(s(st))$；

（2）TYPE（n）＝ π；

（3）TYPE（cn）＝ $(\pi(s(st)))$；

（4）TYPE（a / b）＝ TYPE（b \ a）＝（TYPE（b），TYPE（a）），需要满足的条件如前所述。

注意，条款 1 中出现的两种“s”是不同的，前面正常的“s”是范畴，对应句子，而后面斜体的“s”是状态的类型。

现在引入一个缩写，类型 $\alpha_1(\ldots(\alpha_n(s(st))\ldots)$ 缩写为 $[\alpha_1 \ldots \alpha_n]$。原本 DRSs 的类型是 $(s(st))$，那么根据缩写，DRSs 的类型为 $[\ ]$；原本通名的类型为 $(\pi(s(st)))$，现在通名类型为 $[\pi]$，其余的以此类推。

表 1（词库）：

表达式	范畴	类型	翻译
a_n	$(s/(n\backslash s))/cn$ $((s/n)\backslash s)/cn$	$[[\pi][\pi]]$	$\lambda P\lambda P'([u_n \mid\ ];P'(u_n);P(u_n))$
no_n	$(s/(n\backslash s))/cn$ $((s/n)\backslash s)/cn$	$[[\pi][\pi]]$	$\lambda P\lambda P'[\ \mid \mathbf{not}([u_n \mid\ ];P'(u_n);P(u_n))]$
every_n	$(s/(n\backslash s))/cn$ $((s/n)\backslash s)/cn$	$[[\pi][\pi]]$	$\lambda P\lambda P'[\ \mid ([u_n \mid\ ];P'(u_n)) \Rightarrow P(u_n)]$
Mary_n	$s/(n\backslash s)$ $(s/n)\backslash s$	$[[\pi]]$	$\lambda P([u_n \mid u_n\ \mathbf{is}\ mary]\ ;\ P(u_n))$
he_n	$s/(n\backslash s)$	$[[\pi]]$	$\lambda P(P(u_n))$
him_n	$(s/n)\backslash s$	$[[\pi]]$	$\lambda P(P(u_n))$
who	$(cn\backslash cn)/(n\backslash s)$	$[[\pi][\pi]\pi]$	$\lambda P'\lambda P\lambda v(P(v)\ ;\ P'(v))$
man	cn	$[\pi]$	$\lambda v[\ \mid man\ v]$
stinks	$n\backslash s$	$[\pi]$	$\lambda v[\ \mid stinks\ v]$
adores	$(n\backslash s)/n$	$[\pi\pi]$	$\lambda v'\lambda v[\ \mid v\ adores\ v']$
if	$(s/s)/s$	$[[\ ][\ ]]$	$\lambda pq[\ \mid p \Rightarrow q]$
.	$s\backslash(txt/s)$ $txt\backslash(txt/s)$	$[[\ ][\ ]]$	$\lambda pq(p\ ;\ q)$
and	$s\backslash(s/s)$	$[[\ ][\ ]]$	$\lambda pq(p\ ;\ q)$
or	$s\backslash(s/s)$	$[[\ ][\ ]]$	$\lambda pq[\ \mid p\ \mathbf{or}\ q]$

注：该微型词库来源于缪斯肯斯的系统。

① 经“;”合并的 DRSs 会遇到诸如顺序问题、变元名字重复等问题，参见：J. V. Eijck and H. Kamp，2011，Discourse Representation in Context. In：J. V. Benthem and A. T. Meulen eds.，*Handbook of Logic and Language*，Second Edition. Amsterdam：Elsevier，2011，pp. 181－252。

上面表 1 给出一个很小的英语片段的词库，词库中表达式给予加标处理。注意，为表达式指派的范畴并不唯一，这能从第二栏可以看出。第三栏的类型对应第二栏的范畴，通过函数 TYPE 保证。第四栏给出该类型的翻译。其中，P 是类型为［π］变元，p 和 q 是类型为［ ］的变元，v 是一个类型为 π 的变元。

给定句子“a man adores a woman”，其加标形式如下：

（1）A^1 man adores a^2 woman.

一般上标表示前件而下标表示回指词。由上面的语义树可知，（1）的语义清单为（2）：

（2）D，P，R，D′，P′ ⊢D（P）（λv. D′（P′）（λv′. R（v′）（v）））

用 a^1 的翻译 λP′λP（［u_1｜］；P′（u_1）；P（u_1））替代（2）后件中的 D，用 man 的翻译 λv［｜man v］去替代（2）后件中的 P，再经过几次 λ_转化，便会得到（3）：

（3）［u_1｜］；［｜man u_1］；D′（P′）（λv′. R（v′）（u_1））

将合并引理应用到（3），得到（4）：

（4）［u_1｜man u_1］；D′（P′）（λv′. R（v′）（u_1））

再将“adores”、“a^2”、“woman”的翻译以此方式代入处理，最终得到（5）：

（5）［u_1 u_2｜man u_1，woman u_2，u_1 adores u_2］

如果（1）A^1 man adores a^2 woman. 这个句子扩展成句子序列（6）：

（6）A^1 man adores a^2 woman. She_2 abhors him_1.

根据表 1，“.”的翻译为 λpq（p；q），用它来结合前后两个子句的翻译便得到整个文本的翻译。“She_2 abhors him_1”的翻译为：

（7）［｜ u_2 abhors u_1］

将 λpq（p；q）施用于（5）和（7），得到（8）：

（8）［u_1 u_2｜man u_1，woman u_2，u_1 adores u_2，u_2 abhors u_1］

针对兰贝克证明，现已配上 DRSs。通过一个非常小的片段，这里完成了两件事情，一件是实现了 DRT 的组合性，另外一件事是表明了 DRT 可以看成是类型逻辑的一阶片段。假如出现“$Every_1$ man adores a_2 woman”这样的句子，由于存在辖域歧义现象，在兰贝克证明中出现两个句法证明树，相应地，也会出现两个语义树，自然会存在两个 DRSs 与之匹配。

关于 DRSs 的真值条件，缪斯肯斯将其与一阶逻辑建立对应，组合地给出关于 DRSs 和 DRSs 条件的真值条件。这样，DRT 的组合性问题便都得到解决了。

三、改进方案简述及类型逻辑语法思想总结

首先，简述以上经典方案的不足以及哪些方面可继续发展。

上述方案的语义层可以经过改进，变得更强大进而能处理更多的语言现象。

类型逻辑是严格按照组合性进行运算的，类型逻辑的一阶片段当然也是遵循组合性的。但是由于标准 DRT 不是理想的语言，故可采用它的改进版本作为语义表征层，比如与标准 DRT 等价的简约版本。这里的简约版本来源于奴提雅（Noortje，2013），其中所使用的 PDRT（投射 DRT）可以翻译为相应的一阶逻辑与标准 DRT。鉴于投射 DRT 在处理预设等方面具有优势，故将上述类型逻辑中定义出的标准 DRT 翻译为投射 DRT。翻译工作在奴提雅（2013）中已完成，故可直接用投射 DRT 替换标准 DRT。

投射 DRT 的基本思想是在原地（*in situ*）表征所有被投射的内容，即在引入投射的地方进行表征，用指针表明在何处投射内容得到解释。在投射 DRT 中，区分事实断定的内容和投射的内容。投射部分和断定部分的差别在于它们指向不同的语境，断定部分在它被引进的 PDRS 那里获得相应指针，而投射部分的指针则指向某个可及的 PDRS。这些区分在投射 DRT 框架内是通过用数字、字母标记来实现的。

对第二部分的句法、语义并行推演进行丰富，将语义层配上投射 DRT，于是可以得到句法层（范畴语法，这里用的是兰贝克演算给出

自然语言的句法）和语义层（投射DRT）之间的并行推演。

然而，如果处理动词省略等现象，则需要改进奴提雅（2013）方案，即进一步丰富语义层。限于篇幅，其间需要谈及回指与省略的关系、处理回指的范畴语法竖线算子“|”、DRS合并算子“+”和“*”等等，这里不做赘述，成果演示也略去。[①]

其次，为更好理解本文类型逻辑思想在审视DRT的组合性方面的重要意义，现扼要总结本文思想。

一般来说，类型逻辑涵盖一阶与高阶逻辑，这正是其优势所在，在处理自然语义方面有十分重要的作用。缪斯肯斯（1994）的工作将类型逻辑与DRT框架“建立桥梁”，这项“工程”对于联结蒙塔古语义和话语表现理论（DRT）两大语义框架意义非凡，可以说是开创性的工作。在类型逻辑框架内模拟出DRSs即将DRT看作类型逻辑的一个片段，这样便可以使兰贝克范畴语法和DRT之间存在句法—语义接口，即形成类似于范畴语法与类型逻辑那样并行推演的局面，只不过这里的语义表征换成DRSs，而不再是标准的类型逻辑语言。

当然，兰贝克证明可以进行改进，以更好地表征自然语言句法结构。也就是说，将兰贝克证明与DRT理论进行联结，形成“句法—语义”并行推演之后，无论是句法层还是语义层都可以进一步丰富或扩充，增强表达能力，进而处理更多的语言现象。

本文作者：西南政法大学行政法学院讲师
责任编辑：周勤勤

① 参见石运宝：“基于组合性的DRT研究”，中国社会科学院研究生院博士学位论文，2015年。

Exploring Compositional Problems of DRT from the View of Type Logical Grammar

Shi Yunbao

Abstract: Characterizing the meaning of natural language can either be done by translating a natural language into the logical language first and then giving its model, such as Montague's PTQ (1973) and DRT which is focused on in this article, or be done by giving its model directly. Since the translation in the former approach lies between the natural language and its model, it is called "the middle layer", which is the semantical part at the syntax-semantics interface. The so-called compositional problems are concerning the syntax-semantics correspondence from natural language to this middle layer and from the middle layer to its model. As to these compositional problems, the existing solutions are about designing the syntax structure of natural language and the middle layer, and on dealing with the semantical interpretation of the middle layer. These solutions focus on elaborating the syntax-semantics correspondence. This paper introduces a classical approach, namely, combining categorical grammar and DRT, and puts forward suggestions for its further development.

Keywords: standard DRT; compositionality; Type Logical Grammar