

时态查询语言的并发 Lambek 演算及范畴语法

刘冬宁^{1,2}, 汤庸¹, 黄昌勤^{1,3}, 汤娜¹

(1. 中山大学 计算机科学系, 广东 广州 510275; 2. 中山大学 数学系, 广东 广州 510275; 3. 华南师范大学 教育信息技术学院, 广东 广州 510631)

摘要: 时态逻辑不可递归公理化的性质, 造成它的公理化系统和证明论方法不适用于时态查询语言的建模. 这使得时态逻辑无法利用公理化系统的良好性质及相关证明论方法对时态数据库的推理和查询做更为严谨和细致地刻画. 因此寻找时态逻辑的替代者, 以公理化的方式对时态查询语言做句法和语义的分析是必要的. 考虑的 2 个主要工具是作为句法分析工具的以 Lambek 演算为核心的范畴语法系统, 和作为语义分析工具的类型演算 λ -演算. 这主要是基于类型论的演算特点、SQL 语句与陈述句的相似性、Lambek 演算和 λ -演算的公理化与证明论方法, 及它们作为句法和语义分析工具之间的密切联系与对应性决定的. 据此从 Lambek 演算出发, 结合时态的处理, 构建了并发的 Lambek 演算 (L_{CTQ}) 及相应的范畴语法, 对以公理化系统为基础的时态查询语言的句法分析做相关研究, 并从证明论性质上保障了计算性资源, 使得系统更为严谨和完善.

关键词: 时态查询语言; 句法分析; 并发的 Lambek 演算; 范畴语法

中图分类号: TP301 **文献标识码:** A **文章编号:** 1673-4785 (2009) 03-0245-06

Concurrence Lambek calculus and its categorical grammar in temporal querying languages

LIU Dong-ning^{1,2}, TANG Yong¹, HUANG Chang-qin^{1,3}, TANG Na¹

(1. Department of Computer Science, Sun Yat-sen University, Guangzhou 510275, China; 2. Department of Maths, Sun Yat-sen University, Guangzhou 510275, China; 3. College of Educational Information Technology, South China Normal University, Guangzhou 510631, China)

Abstract: Axiomatic systems and proof methods for temporal logic have so far found relatively few applications in querying language modeling of temporal databases because the standard temporal logic is not recursively axiomatizable. As a result, temporal logic can not use axiomatic systems and proof methods to religiously depict inference and querying of a temporal database. Thus it is necessary to find a replacement for temporal logic which, by axiomatic methods, can become the implement of a syntactic and semantic analysis of temporal querying languages. The Lambek calculus and its categorical grammar as the former and λ -calculus as the latter have become those substitutes in this paper, because of the properties of type theory, the comparability between SQL and state sentence, the axiomatic and proof methods of Lambek calculus and λ -calculus, and their correspondence by Curry-Howard Isomorphism. So, according to the operations of temporality, we built a new axiomatic system used in syntactic analysis which is called concurrence Lambek calculus (L_{CTQ}), and use its categorical grammar to deal with the temporal querying language. Prof theory shows that for protection of computable resources, L_{CTQ} is more regorous and ideal.

Keywords: temporal querying language; syntactic analysis; concurrence Lambek calculus; categorical grammars

关于时间的表达、推理与查询一直是时态数据

库中最为关键的研究点. 目前许多的时态查询语言已经使用于时态数据库, 其理论基础也相对繁杂; 但其中最为直接关联并得到广泛应用的就是基于时态逻辑的查询语言^[1]. 这主要是由于时态逻辑能充分体现时间的特性.

然而在 1994 年, Gabby 等论证了时态逻辑是不

收稿日期: 2008-07-15.

基金项目: 国家自然科学基金资助项目 (60673135、60373081); 国家自然科学基金重点资助项目 (60736020); 广东省自然科学基金资助项目 (7003721); 广东省科技攻关资助项目 (07B010200052); 广州市科技计划资助项目 (07Z3-D3191).

通信作者: 汤庸. E-mail: issty@sysu.edu.cn.

可递归公理化的,由此它的公理化系统和证明论方法不适用于时态数据库查询语言的句法分析和语义分析^[2].这使得相关形式化研究工作存在较大的遗憾,时态逻辑无法利用公理化系统的良好性质及相关证明论方法对时态数据库的推理和查询做更为严谨和细致的刻画.因此,目前用于时态数据库查询语言的时态逻辑都是非公理化的,无论其适用与否,对于计算性语言这都是一个不足.然而,即便是可递归公理化,在句法分析层面上,经典的公理化时态逻辑系统仍不适合时态查询语言.主要原因在于做类似数据库查询这类相干性较强的^[3]、类似自然语言的推理时,经典时态逻辑由于结构性问题不能进行合理的处理^[4].例如经典逻辑中包含了弱化(weakening)、收缩(contraction)和交换律(exchange)等结构规则^[5],而这些结构规则影响了演算的处理,由此不适用于 SQL 半结构化的推理和演算.在语义分析层面上,上述结构规则同样会在经典时态逻辑系统引起“蕴含怪论”等问题,也将影响语义的演算与推理.

时态逻辑之所以与时态查询密切相关,主要是因为它体现了时间的特性,在句法分析和语义分析方面都对时间的处理给予极大支持.因此,要取代时态逻辑在时态查询语言中的地位,替代者也必须能恰如其分地体现时态属性、时态关系之间的运算,并最终用于时态查询语言的句法分析和语义分析.针对此,考虑的时态逻辑作为时态查询语言理论工具的替代者为:作为句法分析工具的以 Lambek 演算为核心的范畴语法系统^[6],和作为语义分析工具的类型演算 λ -演算^[7].首先,将时态属性、时态关系运算转化为类型,通过类型演算进行处理,能恰如其分地保持时间的特点(类型).其次,作为半结构化的 SQL 语言,由于其与自然语言的陈述句型高度相似,因此将善于处理陈述句的、上下文无关的、NP-完全的 Lambek 演算^[8-11]及其范畴语法引入到时态 SQL 语言做句法分析是恰当的.而作为子结构逻辑系统之一的 Lambek 演算,在逻辑性质和证明论上都有良好的性质,能避免大量的逻辑结构性问题和“蕴含怪论”等.再次,作为可递归公理化的类型演算 λ -演算,其在可计算性上有较强的可塑性,并满足程序语义,尤其擅于语义分析^[7].而且很重要的一点是, Lambek 演算与 λ -演算之间可以通过 Curry-Howard 对应理论直接转换,使得句法分析和语义分析之间的过渡更直接、更无缝^[12-13].另外,在国际上 Lambek 演算和相关线性逻辑系统及 λ -演算的许多研究成果已融入到语义网(semantic web)^[14]、数

据流程序^[15](dataflow programming)和模糊分析信息系统^[16]等人工计算语言的句法分析和语义分析工作上,这也证明了其是有效的分析方法,并且值得借鉴.

据此,从 Lambek 演算出发,结合时态的处理,构建了并发的 Lambek 演算(L_{CTQ})及其相应的范畴语法,用来对以公理化系统为基础的时态查询语言的句法分析做相关研究.相关工作是中山大学协同软件研究开发中心时态数据库中间件 TempDB 课题的后续研究^[17],而同时也是时态查询在句法、语义公理化研究方面的第一步.由于 Lambek 演算与 λ -演算的对应性,必须先完成句法分析部分工作,才能在后续工作中更好地完成语义分析工作.

1 类型分析

在时态查询中,关键的研究对象是时态属性和时态关系运算.作为数据库表中的属性之一,时态属性主要体现了时态的特点,但又保留了普通属性的泛性;而时态关系运算,既针对时间,也保留了普通关系运算的泛性,因此在使用 Lambek 演算和类型论分析处理时态属性和时态关系运算的类型时,可做分析如下:

1) 时间属性的类型须体现时态特点,即时态性;

2) 时间属性在体现时态性时,仍需保持作为数据表字段之一的泛性,但该泛性应由时态驱动,并突出时态与非时态的区别;

3) 时态关系运算须体现时态特点,即时态性;

4) 时态关系运算在体现时态性时,仍需保持作为运算本身的泛性,在句型演算中,关系运算时态性与泛性的处理应是并发的,可以理解为专门对时态进行处理的关系运算.

根据以上分析,假设原关系运算类型为 A , t 为时间类型,则时态关系运算的类型可表示为: $t|A$ (或 $A|t$),其中,连接词“ $|$ ”表示“并发”的意义,在“ $|$ ”符号两侧的类型对于句法分析的词项而言是“并发”拥有的.

另一方面,对于时间属性,为了体现时间为主属性,假设原属性类型为 B , t 为时间类型,此时时间属性类型可表示为 $B \leftarrow t$ 或 $t \rightarrow B$.遵循类型逻辑在句法范畴的解释, $B \leftarrow t$ 可表示为函子 $(XY)_1$,其中 Y 可解释为论元 t ,全结构表示函子向右寻找论元; $t \rightarrow B$ 可表示为函子 $(XY)_2$,其中 X 可解释为论元 t ,全结构表示函子向左寻找论元^[3].由于总是时间类型 t 作为函子,因此可以将时态属性的类型看做是时态驱

动的. 下面以表 1 举一个简单的例子.

表 1 人员工作履历表

Table 1 Personnel employment Record

Works		
Valid	Name	Company
[1997, 2001]	John	DEC
[2003, forever]	John	HP
[1999, 2005]	Marry	HP
[2005, forever]	Marry	Microsoft
⋮	⋮	⋮

在表 1 中,希望查询没有失业过的人员名单,该查询语句如下所示:

```
Select    r1. Name
from      works r1, works r2
where     r1. Name = r2. Name
and       Valid (r1) meets Valid (r2)
```

这里 Valid (r1) 和 Valid (r2) 表示有效时间,这是基于时态查询结构化语言 (ATSQL2) 和 TempDB 语法习惯的^[17]. 作为时间属性, Valid (r1) 和 Valid (r2) 的类型分别为: $a \leftarrow t$ 和 $t \rightarrow a$, 其中: t 表示时间类型, a 表示属性类型, 属性类型是由时间类型驱动的. meets 是 Allen 提出的 13 种时态区间运算之一^[18], 也是 ATSQL2 规定的时态运算. 在这里它的类型为 $t \mid (a \rightarrow (s \leftarrow a)) \mid t$, 其中: t 仍然表示时间类型; $a \rightarrow (s \leftarrow a)$ 表示关系运算类型, 对于时态关系运算而言, 2 种类型应是并发处理的.

句法分析的主要目标是根据初始类型, 通过相应演算系统, 得出终止类型. 据此, 新连接词“ \mid ”的引入及词项的类型赋值变化, 要对上述例子做句法分析, 必须设计相应的 Lambek 演算系统, 作者称之为并发的 Lambek 演算系统, 即 L_{CTQ} 演算系统 (Lambek calculus with concurrence for temporal querying). 在下文中, 先讨论 L_{CTQ} 演算系统的相关性质, 并最后将其应用于上例中.

2 L_{CTQ} 演算系统

2.1 类型与自然演绎系统

L_{CTQ} 系统的类型由原子类型 p_1, p_2, \dots 等构成, 所有原子类型的集合表示为 T_p . 类型的有穷非空矢列个二元连接词 $\cdot, \rightarrow, \leftarrow$ 和 \mid .

集合写作 T_p^+ . L_{CTQ} 演算的矢列形如 $\Gamma \vdash A$, 并包含 4

该演算系统的自然演绎系统如下所示:

公理: $A \vdash A$

规则:

$$\begin{aligned}
 & \frac{X \vdash A \quad Y, A, Z \vdash B}{Y, X, Z \vdash B} (\text{Cut}); \\
 & \frac{X \vdash A \cdot B \quad Y, A, B, Z \vdash C}{Y, X, Z \vdash B} (\cdot E); \\
 & \frac{X \vdash A \quad X \vdash B}{X \vdash A \cdot B} (\cdot I); \\
 & \frac{X \vdash A \quad Y \vdash A \rightarrow B}{X, Y \vdash B} (\rightarrow E); \\
 & \frac{A, X \vdash B}{X \vdash A \rightarrow B} (\rightarrow I); \\
 & \frac{X \vdash A \leftarrow B \quad Y \vdash B}{X, Y \vdash A} (\leftarrow E); \\
 & \frac{X, A \vdash B}{X \vdash B \leftarrow A} (\leftarrow I); \\
 & \frac{X \vdash A \leftarrow B \quad Y \vdash B \mid C}{X, Y \vdash A \cdot C} (\mid E'); \\
 & \frac{X \vdash A \mid B \quad Y \vdash B \rightarrow C}{X, Y \vdash A \cdot C} (\mid E''); \\
 & \frac{X \vdash A \quad X \vdash B}{X \vdash A \mid B} (\mid I).
 \end{aligned}$$

除 $(\mid E')$ 、 $(\mid E'')$ 、 $(\mid I)$ 3 条规则外, 其余规则是经典 Lambek 演算 L 系统的全部规则, 它们组成原 L 系统^[6]. $(\mid E')$ 、 $(\mid E'')$ 、 $(\mid I)$ 3 条规则是为新添加的连接符“ \mid ”加入的消去和引入规则. 在消去规则中, $(\mid E')$ 为左消去规则, $(\mid E'')$ 为右消去规则, $(\mid I)$ 为引入规则.

定义 1 如果矢列 $\Gamma \rightarrow A$ 在 L_{CTQ} 演算中是可推导的, 则写作 $\Gamma \vdash_{L_{CTQ}} A$.

2.2 代数模型

与经典 Lambek 演算的 L 系统大致相同, L_{CTQ} 系统的模型也为剩余代数 (residuated algebra) 模型^[3]. 对于 L_{CTQ} 系统的剩余半群 (residuated semigroup) 和剩余运算 (residuated operation) 描述如下.

定义 2 L_{CTQ} 的剩余半群的结构为 $M = (M, \cdot, \rightarrow, \leftarrow, \mid, \leq)$, 使得 (M, \cdot) 是 1 个半群, \leq 是 M 上偏序关系, 且 $\rightarrow, \leftarrow, \mid$ 是 M 上的二元运算, 并对所有 $a, b, c, d \in M$ 满足以下关系:

- 1) $b \leq a \rightarrow c$ iff $a \cdot b \leq c$ iff $a \leq c \leftarrow b$;
- 2) $a \leq b \mid c$ iff $(a \leq b \ \& \ a \leq c)$;
- 3) $a \leftarrow b \cdot (c \mid d) \leq a \cdot d$ iff $(b \leq c \ \& \ c \leq b)$;
- 4) $(a \mid b) \cdot c \rightarrow d \leq a \cdot d$ iff $(b \leq c \ \& \ c \leq b)$.

L_{CTQ} 系统的可满足性和有效性定义如下:

定义 3 L_{CTQ} 系统的代数模型为一序对 (M, μ) , 使得 M 是 1 个剩余半群, μ 为 M 上的类型赋值. 如果 $\mu(A_1) \cdot \dots \cdot \mu(A_n) \leq \mu(A)$, 则序列 A_1, \dots ,

$A_n \vdash A$ 在模型中是真的. 如果 $A_1, \dots, A_n \vdash A$ 在所有 (M, μ) 都是真的, 则它在模型中是有效的.

作为公理化的句法分析系统, 必须保证系统具有可靠性和完全性, 在此对其进行证明.

定理1 (可靠性定理) L_{CTQ} 演算系统关于剩余半群模型 (M, μ) 是可靠的.

要证明可靠性定理, 只需证明在该模型中的规则 $(\vdash E')$ 、 $(\vdash E)$ 、 $(\vdash I)$ 是可靠的, 其余规则均同原 L 演算. 其证明主要根据定义2进行, 结论显然.

定理2 (完全性定理) L_{CTQ} 演算系统关于剩余半群模型 (M, μ) 是完全的.

文献[9-10]中给出了关于经典 Lambek 演算的完全性证明. 因为 L_{CTQ} 系统是在经典 Lambek 演算上添加了“ \vdash ”连接符和相关规则, 所以只需证明“ \vdash ”的完全性. 因此, 采用的是 Buszkowski 的 product-free 的证明方法^[3], 其证明方法类似于典范模型的证明方法. 现证明如下:

设定一个序列 $X_0 \vdash A_0$, 令 S 表示该序列中所有公式的子公式集合. 显然, S 是一个有穷集, S 上的语言为 S^+ 的子集. 定义赋值 μ 如下所示:

$$\mu(p) = \{X \in S^+ : X \vdash_{L_{CTQ}} p\}.$$

需要证明的是:

$$\mu(A) = \{X \in S^+ : X \vdash_{L_{CTQ}} A\}. \quad (1)$$

在此, 只需要证明 $A = B \mid C$ 的情况, 其他情况的证明见文献[3].

$$\text{证明 } \mu(B \mid C) = \{X \in S^+ : X \vdash_{L_{CTQ}} B \mid C\}.$$

1) 左 \Rightarrow 右.

设 $X \in \mu(B \mid C)$, 所以 $X \in \mu(B) \cap \mu(C)$, 所以 $X \in \mu(B)$ 且 $X \in \mu(C)$, 据归纳假设, $X \vdash B$ 且 $X \vdash C$, 所以据 $(\vdash I)$, $X \vdash B \mid C$.

2) 右 \Rightarrow 左.

设 $X \vdash B \mid C$, 所以 $X \vdash B$ 且 $X \vdash C$, 据归纳假设, $X \in \mu(B) \cap \mu(C)$, 所以 $X \in \mu(B) \cap \mu(C)$.

据上, $\mu(B \mid C) = \{X \in S^+ : X \vdash_{L_{CTQ}} B \mid C\}$ 得证, 即 $\mu(A) = \{X \in S^+ : X \vdash_{L_{CTQ}} A\}$ 得证.

下面证明 L_{CTQ} 演算系统的完全性: 设 $X_0 \vdash A_0$, 据式(1), $X_0 \notin \mu(A_0)$, 又因为 $X_0 \vdash X_0$, 再据式(1), $X_0 \in \mu(X_0)$, 因此 $\mu(X_0) \not\subseteq \mu(A_0)$, 所以 $F \models A_0$. 故完全性定理得证.

3 L_{CTQ} 范畴语法及时态查询句法分析

3.1 L_{CTQ} 范畴语法

据上, 可给出 L_{CTQ} 范畴语法, 利用范畴语法与类型论对时态类型的演算进行处理. 首先, 用集合 B 表

示基本范畴, 例如表示时间的类型 t , 表示表属性的 a 类型, 表示终止符的 s 类型, 表示数据库表的 f 类型等. 由此, L_{CTQ} 范畴语法的定义如下:

定义4 范畴集合定义

集合 B 表示基本范畴集合. $CAT(B)$ 为一最小集, 使得

- 1) $B \subset CAT(B)$;
- 2) 如果 $A, B \in CAT(B)$, 则 $A \leftarrow B \in CAT(B)$;
- 3) 如果 $A, B \in CAT(B)$, 则 $A \rightarrow B \in CAT(B)$;
- 4) 如果 $A, B \in CAT(B)$, 则 $A \cdot B \in CAT(B)$;
- 5) 如果 $A, B \in CAT(B)$, 则 $A \mid B \in CAT(B)$.

定义5 L_{CTQ} 范畴语法的定义

给定字母表 Σ . 一个 L_{CTQ} 范畴语法 G 是1个三元组 $\langle B, LEX, S \rangle$. 其中有穷集 B 定义如上, LEX 是 $\Sigma^+ \times CAT(B)$ 上的一个有穷子关系, S 是 $CAT(B)$ 的一个有穷子集(终止符范畴).

该语法上的语言定义如下.

定义6 $G = \langle B, LEX, S \rangle$ 是字母表 Σ 上的1个 L_{CTQ} 语法. 则对 $\alpha \in L(G)$ iff 存在 $a_1, \dots, a_n \in \Sigma^+, A_1, \dots, A_n \in CAT(B)$, 且 $S \in S$ 使得

- 1) $\alpha = a_1 \cdots a_n$;
- 2) 对所有 i 使得 $1 \leq i \leq n$, $\langle a_i, A_i \rangle \in LEX$, 且 $\vdash A_1, \dots, A_n \Rightarrow S$.

3.2 词汇表与句法分析

至此, 根据 L_{CTQ} 范畴语法, 可以对第1节例子中的查询建立词汇表(LEX), 如表2所示.

表2 词汇表

Tabel 2 Lexicon(LEX)

词 项	类 型	简要描述
Select	$(n \rightarrow s) \leftarrow a$	动作类型
r1. Name	a	属性类型
r2. Name	a	属性类型
from	$(s \rightarrow s) \leftarrow f$	介词类型
Works	f 或 $f \leftarrow f$	表示“表”的时候类型为 f , 修饰表的时候类型为“ $f \leftarrow f$ ”
r1, r2	f	表类型
Where, and	$(s \rightarrow s) \leftarrow s$	连词类型
Valid (r1)	$a \leftarrow t$	时态属性类型
Valid (r2)	$t \rightarrow a$	时态属性类型
Meets	$(a \rightarrow (s \leftarrow a)) \mid t$	时态运算类型
,	$f \rightarrow (f \leftarrow f)$	连词类型
=	$a \rightarrow (s \leftarrow a)$	运算类型
System	n	系统公设类型

在这里, 相关词汇类型的设置是根据范畴语法对自然语言的处理相应转换至半结构 SQL 语言进

行处理的,额外添加的类型主要体现在时态类型、时态关系运算以及系统公设上.由于添加了系统公设,第1节例子的查询语句的转换如下.

```
System
Select    r1. Name
from      works r1, works r2
where     r1. Name = r2. Name
and       Valid (r1) meets Valid (r2)
```

各部分类型演算步骤分列如下:

1) “System”: 系统公设, 类型为 n ;

2) “Select r1. Name” 类型演算为: $((n \rightarrow s) \leftarrow a) \cdot a \vdash n \rightarrow s$;

3) “from works r1, works r2” 类型演算为: $((s \rightarrow s) \leftarrow f) \cdot ((\leftarrow f) \cdot f) \cdot (f \rightarrow (f \leftarrow f)) \cdot ((f \leftarrow f) \cdot f) \vdash s \rightarrow s$;

4) “where r1. Name = r2. Name” 类型演算为: $((s \rightarrow s) \leftarrow s) \cdot a \cdot (a \rightarrow (s \leftarrow a)) \cdot a \vdash s \rightarrow s$;

5) “and Valid (r1) meets Valid (r2)” 类型演算为: $((s \rightarrow s) \leftarrow s) \cdot (a \leftarrow t) \cdot (t \mid (a \rightarrow (s \leftarrow a))) \vdash s \rightarrow s$;

6) 全句最终类型演算为: $n \cdot (n \rightarrow s) \cdot (s \rightarrow s) \cdot (s \rightarrow s) \cdot (s \rightarrow s) \vdash s$.

据 L_{CTQ} 演算, 上述句型的类型演算最终能推出终止符类型 s , 使得句法分析演算停止并得到正确结果. 而对于不符合句法的查询, 则不能得到终止符类型 s , 这是由可靠性和完全性保证的.

4 证明论性质

文中根据时态查询语言需要表现时态性的特点, 修改了 Lambek 演算, 在其基础上添加了并发连接词“ \mid ”和相关规则, 新的系统称为 L_{CTQ} 演算. 事实上, 如果在系统中添加了收缩律 (contraction), 如下所示:

$$\frac{X, A, A, Y \vdash B}{X, A, Y \vdash B} \text{ (contraction)},$$

则并发连接词“ \mid ”将与经典的连接词“ \wedge ”等价. 但这不是所希望得到的, 因为收缩律的缺失, 使得每一个演算的资源都至多只能使用 1 次, 保证了句法分析的词项不能重复使用. 同时, 系统由于没有弱化规则 (weakening), 如下所示:

$$\frac{X, Y \vdash B}{X, A, Y \vdash B} \text{ (weakening)},$$

使得每一个演算的资源都至少要使用 1 次. 这样的

一类既没有收缩规则, 又没有弱化规则的逻辑系统称为资源敏感逻辑 (resource-sensitive)^[3], 它能使得参与演算的资源都“用到且仅用到一次”, 满足了计算性资源的特点, 由此在计算语言学中得到广泛应用. 因此, 使得在此基础上建立的 L_{CTQ} 演算和 L_{CTQ} 范畴语法能更好地用于时态查询语言的句法分析. 从证明论性质上, 由于计算性资源的保障, 也使得系统更为严谨和完善.

5 结束语

由于时态类型区别于语言学中诸如动词、名词、副词等语言结构类型, 针对时态查询, 文章提出了并发的 Lambek 演算, 证明了其可靠性和完全性, 并通过实例进行了阐述与演算. 文中提出的并发的 Lambek 演算 (L_{CTQ}), 从证明论角度上, 保障了计算性资源, 避免了蕴含怪论等问题, 使得系统更为严谨和完善. 在下一步工作中, 将主要研究句法分析和语义分析的衔接问题以及语义分析, 其基本思路是通过 Curry-Howard 对应理论和类型演算 λ -演算将句法演算过渡到语义演算, 对相关语义进行分析, 并建立原型系统或直接在 TempDB 上进行验证.

参考文献:

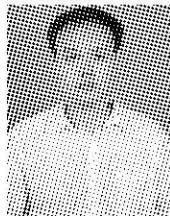
- [1] CHOMICKI J, TOMAN D. Temporal logic information systems[M]//CHOMICKI J, SAAKE G. Logics for databases and information systems. Norwell, USA: Kluwer Academic Publishers, 1998: 31-70.
- [2] GABBAY D M, HODKINSON I, REYNOLDS M. Temporal logic: mathematical foundations and computational aspects [M]. New York, USA: Oxford University Press, 1994.
- [3] BUSZKOWSKI W. Categorical grammars and substructural logics[C]//Proceedings of the 5th International Conference on Logic and Cognition. Guangzhou, China, 2006.
- [4] RESTALL G. An introduction to substructural logics[M]. New York, USA: Routledge, 2000.
- [5] JÄGER G. Anaphora and type logical grammar[M]. Dordrecht, the Netherlands: Springer, 2005.
- [6] LAMBEK J. The mathematics of sentence structure[J]. The American Mathematical Monthly, 1958, 65: 154-170.
- [7] BARENDEGT H P. The lambda calculus: its syntax and semantics; volume 103 of studies in logic and the foundations of mathematics[M]. Amsterdam, the Netherlands: Elsevier, 1984.
- [8] PENTUS M. Lambek grammars are context-free[C]//Proceedings of the 8th Annual IEEE Symposium on Logic in

- Computer Science. Montreal, Canada, 1993: 429-433.
- [9] PENTUS M. Lambek calculus is L-complete: Report LP93-14[R]. Amsterdam: Institute for Logic, Language and Computation, University of Amsterdam, 1993.
- [10] PENTUS M. Models for the Lambek calculus[J]. Annals of Pure and Applied Logic, 1995, 75(1/2): 179-213.
- [11] PENTUS M. Lambek calculus is NP-complete: CUNY PhD Program in Computer Science Technical Report TR-2003005[R]. New York: The Graduate Center, The City University of New York, 2003.
- [12] CURRY H B. Some logic aspects of grammatical structure [C] // Proceedings of the Twelfth Symposium in Applied Mathematics. New York, USA: American Mathematical Society, 1961: 56-68.
- [13] HOWARD W. The formulae-as-types notion of construction [M] // SELDIN JR, HINDLEY JP. To H B Curry: essays on combinatory logic, Lambda calculus and formalism. New York: Academic Press, 1980: 479-490.
- [14] RAO JH, KUNGAS P, MATSKIN M. Logic-based Web services composition: from service description to process model [C] // Proceedings of the IEEE International Conference on Web Services. Washington DC, USA: IEEE Computer Society, 2004: 446-453.
- [15] UUSTALU T, VENE V. The essence of dataflow programming[J]. Lecture Notes in Computer Science, 2006, 4164: 135-167.
- [16] HAJEK P. Metamathematics of fuzzy logic [M]. Dordrecht, the Netherlands: Kluwer Academic Publishers, 1999.

[17] Research Team on Temporal Data Processing Component. TempDB[EB/OL]. [2008-07-11]. <http://www.cosoft.sysu.edu.cn/TempDB/index.htm>, 2008.

[18] ALLEN J F. Maintaining knowledge about temporal intervals[J]. Communications of the ACM, 1983, 26(11): 832-843.

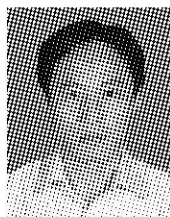
作者简介:



刘冬宁,男,1979年生,助理研究员,博士后,主要研究方向为人工智能逻辑、时态数据库。曾获罗克韦尔奖(Rockwell),发表学术论文20余篇。



汤庸,男,1964年生,教授,博士生导师,博士。主要研究方向为时态信息处理。获宝钢教育奖、丁颖科技奖;开发国内第一个时态数据库中间件软件Temp-DB。发表的学术论文被SCI收录11篇,EI收录57篇。



黄昌勤,男,1972年生,副教授,硕士生导师,博士。主要研究方向为服务(网络)计算、语义Web、Web智能等。发表学术论文20余篇,出版专著1部。