

17.3.3

① λ 表示法: 一种从 FOL 公式 进行抽象的方法。

$\lambda x. P(x)$
variable 变量

e.g. 1 假设有一个常量 A

$\lambda x. P(x)(A)$
 $P(A)$

λ 化简
 λ -reducing

e.g. 2 将一个 λ 表达式

作为另一个 λ 表达式的一部分

柯里化
currying

$\lambda x. \lambda y. \text{Near}(x, y)$

假设如

$\lambda x. \lambda y. \text{Near}(x, y)(\text{Bacaro})$

$\lambda y. \text{Near}(\text{Bacaro}, y)$

↓ 继续化简得:

$\lambda y. \text{Near}(\text{Bacaro}, y)(\text{Centro})$

$\text{Near}(\text{Bacaro}, \text{Centro})$

$\text{Near}(\text{Centro}, \text{Bacaro})$

项 项

↓
它们所包含的域元素 \in "Near" 所指示的元组中。
被包含

17.3.5

② 取 λ 推理 = FOL 提供的推理方法。

α
 $\alpha \Rightarrow \beta$
 β

如果 α 为真, 则 β 为真

e.g. 3 Vegetarian Restaurant (Leaf)

$\forall x. \text{Vegetaurant}(x) \Rightarrow \text{Serves}(x, \text{VegetarianFood})$
 $\text{Serves}(\text{Leaf}, \text{VegetarianFood})$

- 取式推理的应用 (chapter 21)

-1. 正向链

-2. 反向链

17.4

Serves (Leaf, VegetarianFare)

1. FOL中, 谓词应有固定的元数 (arity),
固定数目的论元.

e.g. 事件变量 (event variable)

如 "e" 作为句柄 (handle)

$\exists e \text{ Eating}(e, \text{Speaker}, \text{TurkeySandwich}, \text{Lunch}, \text{Desk})$

↓ Eating 事件发生在 Tuesday.

$\exists e (\dots) \wedge \text{Time}(e, \text{Tuesday})$

↓ 这种表示事件的方式 = Davidsonian 事件表示.

* ↓ 利用额外的关系、捕获、所有论元来、消除歧义.

$\exists e \wedge \text{Location}(e, \text{Desk}) \wedge \text{Time}(e, \text{Tuesday})$

2. 时态逻辑的表示方法:

二元谓词 Precedes, 如: 论元1表示的时间

在

论元2 之前

如: $\text{Precedes}(\text{Now}, e)$

$\text{Precedes}(e, \text{Now})$

之前 之后 → 表明在 Now 之前发生.

→ Eating (参照点)

体 (aspect) =

1. 静态体 (stative)

2. 行动体 (Activity)

3. 完成体 (Accomplishment)

4. 达成体 (Achievement)

会达到某特定目标, 故, 偶而可以归为一类.

17.5

* 除 FOL 方法外, 还有:

TBox = C ⊆ D (声明)

ABox

1. 语义网络 (semantic networks)

2. 框架方法 (frames | slot-filler) (槽填充)