



### 1. 知识与知识表示

知识与知识表示是人工智能中的一项基本技术，这项技术决定着人工智能如何进行知识学习，因此知识与知识表示是人工智能最基础的部分。

#### 1知识

- **知识是一个结论性的描述**。知识也是人类在实践中认识客观世界（包括人类自身）的成果，它包括事实、信息的描述或在教育和实践中获得的技能<sup>[1]</sup>。由于知识并非等同于真理，所以并非所有的知识都是正确的。

#### 2数据

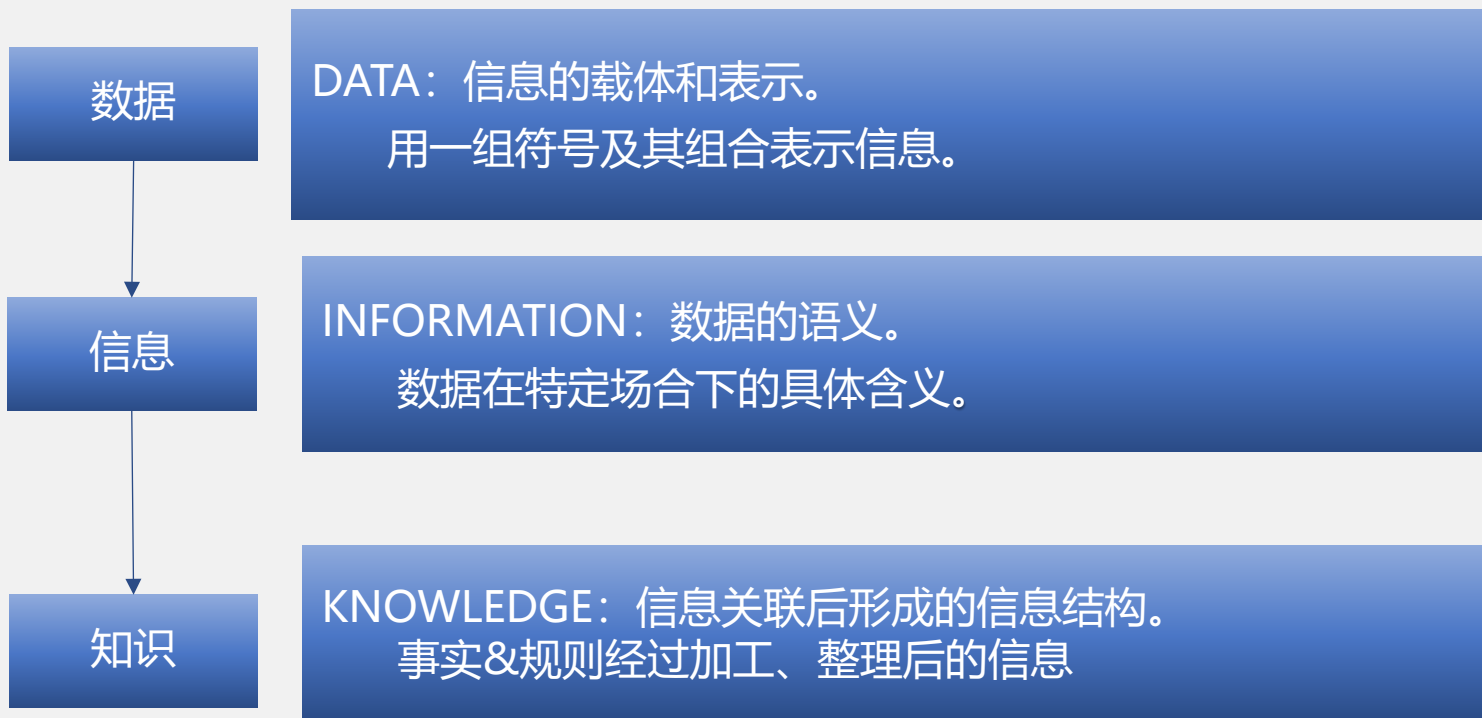
- 计算机里面所有存储的东西，都可以叫数据；**数据是信息的表现形式和载体**，可以是符号、文字、数字、语音、图像、视频等，数据就是存储在硬盘或者光盘U盘等里的0、1组合。

#### 3信息

- **信息就是数据的语义，是数据在特定场合下的具体含义**。简单的说信息就是有用的数据。因为数据不一定是有用的，或许很多数据是冗余的。



### 1. 知识与知识表示



数据、信息、知识的基本概念



### 1. 知识与知识表示

信息与数据的关系是内容与形式的关系，信息是数据所包含的内容，而数据是信息的表现形式。在数据的基础上形成信息，在信息的基础上形成知识，数据是信息的起源，信息是知识的原料，知识是信息加工提炼的结晶。



数据、信息、知识的关系



### 2. 知识表示方法

知识表示，就是用某种约定俗成的方式，对知识进行的描述。它可以是一组规则，也可以是一种计算机能接受的用于描述知识的数据结构。知识表示问题，一直就是人工智能研究最活跃的部分之一。在人工智能领域里已经发展了许多种知识表示方法，常用的有：**产生式规则、谓词逻辑、语义网络和框架**。从其表示特性来考察可归纳为两类：**说明型 (declarative) 表示**和**过程型 (procedural) 表示**。

#### 说明型表示

知识是一些**已知的客观事实**，实现知识表示时，把与事实相关的知识与利用这些知识的过程明确区分开来，并重点表示与事实相关的知识。

#### 过程型表示

知识是**客观存在的一些规律和方法**，实现知识表示时，对事实型知识和利用这些知识的方法不作区分，使二者融为一体。

好的知识表示方法应满足以下几点要求

- 1) 具有良好定义的语法和语义
- 2) 有充分的表达能力，能清晰的表达有关领域的各种知识
- 3) 便于有效的推理和检索，具有较强的问题求解能力，适合于应用问题的要求，提高推理和检索的效率
- 4) 便于知识共享和知识获取
- 5) 容易管理，易于维护知识库的完整性和一致性

## 第二章 知识表示

2.1 知识表示的内涵

2.2 谓词逻辑表示法

2.3 产生式规则表示法

2.4 语义网络表示

2.5 知识图谱表示

2.6 框架表示法

2.7 脚本表示法

习题



(1) 个体是常量：一个或者一组指定的个体。

- “老张是一个教师”：一元谓词  $Teacher(Zhang)$
- “ $5 > 3$ ”：二元谓词  $Greater(5, 3)$
- “Smith作为一个工程师为IBM工作”：  
三元谓词  $Works(Smith, IBM, engineer)$

(2) 个体是变元（变量）：没有指定的一个或者一组个体。

“ $x < 5$ ”：  $Less(x, 5)$

(3) 个体是函数：一个个体到另一个个体的映射。

- “小李的父亲是教师”：  $Teacher(father(Li))$

(4) 个体是谓词

- “Smith作为一个工程师为IBM工作”：  
二阶谓词  $Works(engineer(Smith), IBM)$



### 1. 连接词（连词）

(1)  $\neg$ :

“机器人不在2号房间” :  $\neg \text{Inroom}(\text{robot}, r2)$

“李明打篮球或踢足球” :

$\text{Plays}(\text{Liming}, \text{basketball}) \vee \text{Plays}(\text{Liming}, \text{football})$

(2)  $\vee$ : “析取” (disjunction) ——或。

(3)  $\wedge$ : “合取” (conjunction) ——与。

“我喜欢音乐和绘画” :

$\text{Like}(I, \text{music}) \wedge \text{Like}(I, \text{painting})$

(4)  $\rightarrow$ : “蕴含” (implication) 或 “条

件”

“如果刘华跑得最快，那么他取得冠军。” :

$\text{RUNS}(\text{Liuhua}, \text{faster}) \rightarrow \text{WINS}(\text{Liuhua}, \text{champion})$

(5)  $\leftrightarrow$ : “等价” (equivalence) 或 “双条件”

(bicondition) 。

$P \leftrightarrow Q$ : “P当且仅当Q”。



### 谓词公式

#### 2. 合式公式的性质

##### ◆ 合式公式的真值

表2.1 真值表

$P$	$Q$	$P \vee Q$	$P \wedge Q$	$P \Rightarrow Q$	$\sim P$
T	T	T	T	T	F
F	T	T	F	T	T
T	F	T	F	F	F
F	F	F	F	T	T

- ◆ 等价 (Equivalence) 如果两个合式公式，无论如何解释，其真值表都是相同的，那么我们就称此两合式公式是等价的。





### 谓词公式

(1) 否定之否定:  $\sim(\sim P)$ 等价于 $P$

(2)  $P \vee Q$ 等价于 $\sim P \Rightarrow Q$

(3) 狄·摩根定律:  $\sim(P \vee Q)$ 等价于 $\sim P \wedge \sim Q$

$\sim(P \wedge Q)$ 等价于 $\sim P \vee \sim Q$

(4) 分配律:  $P \wedge (Q \vee R)$ 等价于 $(P \wedge Q) \vee (P \wedge R)$

$P \vee (Q \wedge R)$ 等价于 $(P \vee Q) \wedge (P \vee R)$



(5) 交换律:  $P \wedge Q$  等价于  $Q \wedge P$

$P \vee Q$  等价于  $Q \vee P$

(6) 结合律:  $(P \wedge Q) \wedge R$  等价于  $P \wedge (Q \wedge R)$

$(P \vee Q) \vee R$  等价于  $P \vee (Q \vee R)$

(7) 逆否律:  $P \Rightarrow Q$  等价于  $\sim Q \Rightarrow \sim P$



(8)  $\sim(\exists x)P(x)$  等价于  $(x) [\sim P(x)]$

$\sim(\forall x)P(x)$  等价于  $(\exists x) [\sim P(x)]$

(9)  $(\forall x) [P(x) \wedge Q(x)]$  等价于  $(\forall x)P(x) \wedge (\forall x)Q(x)$

$(\forall x) [P(x) \vee Q(x)]$  等价于  $(\forall x)P(x) \vee (\forall x)Q(x)$

(10)  $(\forall x)P(x)$  等价于  $(\forall y)P(y)$

$(\exists x)P(x)$  等价于  $(\exists y)P(y)$



### 2. 量词 (quantifier)

(1) 全称量词 (universal quantifier) ( $\forall x$ ): “对个体域中的所有 (或任一个) 个体  $x$ ”。

“所有的机器人都是灰色的”：

$$(\forall x)[ROBOT(x) \rightarrow COLOR(x, GRAY)]$$

(2) 存在量词 (existential quantifier) ( $\exists x$ ): “在个体域中存在个体  $x$ ”。

“1号房间有个物体”：

$$(\exists x) INROOM(x, r1)$$



### 1. 命题逻辑

我们把用语言、符号或式子表达的，可以判断真假的陈述句叫做**命题**。  
其中判断为真的语句叫做真命题，判断为假的语句叫做假命题。简单的讲就是可以判断真假的语句。

比如： $1 < 2$ ，为真命题。 $1 > 2$ ，为假命题。

#### 原子命题

原子命题也称**简单命题**，就是不“或”，“且”，“非”的简单判断句。如“今天是星期五”，“2是自然数”等等。

#### 复合命题

复合命题是指由**简单命题**用**联结词**联结而成的命题。

- 1) **如果**天在下雨或者洒水车工作，**那么**地是湿的。
- 2) **如果**没有电，**那么**交换机是不能工作的。

注意：虽然复合命题是由命题构造而成的，但并不是任意命题组合在一起就可构成复合命题。如果仅仅把两个命题摆在一起而没有联结词仍然只是两个命题。



### 2. 谓词逻辑

谓词，在谓词逻辑中，原子命题分解成个体词和谓词。

个体词是可以独立存在的具体事物或抽象的概念，例如，电子计算机，实数，唯物主义，油菜花等等。

谓词则是用来刻画个体词的性质的词，即刻画事和物之间的某种关系表现的词。如“苹果”是一个现实物个体词，“苹果可以吃”是一个原子命题，“可以吃”是谓词，刻画“苹果”的一个性质，即与动物或人的一个关系。

**1 常量：**常量表示事物或概念等特指对象，例如20、苹果、书、椅子等。

#### 2 变量：

变量是一个宽泛的概念，相对于常量而言的。常量是指恒定不变的量。变量就是值不是恒定不变，而是变化的量。变量常用一些符号表示，例如X、Y、Z等。

#### 3 函数

不同的变量之间往往有一定的制约关系。函数表示了两个变量之间的映射关系。比如函数这个函数表示随着的变化而变化，或者说因为的变化而变化。

如果对于任意一个 $x$ 都有唯一确定的一个 $y$ 和它对应，那么就称 $x$ 是自变量， $y$ 是 $x$ 的函数。 $x$ 的取值范围叫做这个函数的定义域，相应 $y$ 的取值范围叫做函数的值域。



### 2. 谓词逻辑

#### 4 谓词

谓词表示对象的属性和对象之间的关系，用 $P(t_1, t_2, \dots, t_n)$ 表示一个 $n$ 元谓词， $P$ 是谓词符号， $t_i (i = 1, 2, \dots, n)$ 是谓词的参量，它可以是常量或变量。

如，“姚明喜欢篮球。”这个自然语言语句可表示为“Likes (姚明, 篮球)”，Likes是谓词，姚明和篮球是参量。

#### 5 逻辑运算符

- $\wedge$  (conjunction) : 合取 (与)
- $\vee$  (disjunction) : 析取 (或)
- $\sim$  (not) : 否定 (非)
- $\rightarrow$  (implication) : 蕴含 (如果. . . , 则. . . )

#### 6 量词

- $\forall$  (universal quantifier) : 全称量词 (对于所有的)
- $\exists$  (existential quantifier) : 存在量词 (存在某个)



### 2. 谓词逻辑

## 7 分隔符

- 1) 项 (Term) : 常数、变量或函数称为项, 它可作为谓词的参数。
- 2) 原子式 (Atom) : 命题或谓词称为原子式。
- 3) 文字 (Literal) : 原子式或原子式的否定称为文字。
- 4) 范式 (Well-Formed Formula)

A. 原子公式是范式。

B. 如果A和B是范式, 则 $\sim A$ ,  $A \wedge B$ ,  $A \vee B$ ,  $A \rightarrow B$ 是范式。

C. 若A是范式, X是任何变量, 那么 $(\forall X)A$ 和 $(\exists X)A$ 是范式。

D. 有限次运用A  $\rightarrow$  C所产生的公式均为范式。

- 5) 子句 (Clause)

子句是用析取符号 $\vee$ 连接文字组成的逻辑式。

定理: 任何一个范式公式都可以等价地变换成一个子句集合, 集合中各子句之间用合取符号 $\wedge$ 连接。

若 $P_1, P_2, Q_1, Q_2, \dots, Q_n$ 为原子式, 则

$P_1 \vee P_2 \vee Q_1 \vee Q_2 \vee \dots \vee Q_n$  为子句, 根据逻辑的等价性, 上式与下式等价:

$$Q_1 \wedge Q_2 \wedge \dots \wedge Q_n \rightarrow P_1 \vee P_2$$





### 2. 谓词逻辑

## 7 分隔符

### 6) Horn子句

最多由一个正文字（非否定的原子公式）组成的子句称为Horn子句。或者说，最多包含一个结论的子句称为Horn子句。Horn子句有两种形式：

#### A. 有头Horn子句

$$P \vee \sim Q_1 \vee \sim Q_2 \vee \dots \vee \sim Q_n$$

它等价于：  $Q_1 \wedge Q_2 \wedge \dots \wedge Q_n \rightarrow P$

#### B. 无头Horn子句

•  $\sim Q_1 \vee \sim Q_2 \vee \dots \vee \sim Q_n$  它等价于：  $Q_1 \wedge Q_2 \wedge \dots \wedge Q_n \rightarrow$



### 3. 知识表示实例

谓词逻辑应用**归结原理（归结反演）**来求解问题，为证明一个语句成立，转化为归结证明其非，导出一个与已知语句（公理）矛盾的结果（为空子集），说明该语句的非不成立，则该语句成立。在归结证明过程中，要使用谓词匹配操作，即证明两个谓词表达式恒等。

谓词匹配的规则如下：

- 1) 两个谓词符相同，否则不匹配
- 2) 参量个数相等，否则不匹配
- 3) 对应参量匹配：
  - a. 两者均为常量，两者应相等
  - b. 一个为常量，另一个为变量，将常量赋给变量
  - c. 两者均为变量，作变量代换
  - d. 一变量能匹配一函数/谓词表达式，必须这个表达式不包含该变量的任一例示。



### 3. 知识表示实例

#### 1. 事实

- 关系（对象，对象，...，对象），其中，“关系”是谓词，对象是谓词的参量。  
例如，书籍信息可用书籍谓词描述，谓词参量包括：书号、书名、著者、出版社、出版日期、主题，即：  
书籍（书号，篇名，著者，出版社，出版日期，主题）。
- 主题知识可用主题谓词表示：主题（主概念，子概念）。  
其中，“主概念”可代表单个概念，“子概念”是一个词表，标志着数据库中这个主题组面，它们直接从属于主概念。所有主题谓词集中在一起，可以形成一个主题词索引。
- 用户信息可使用用户谓词表示：  
用户（用户名，主题）。其中的“主题”描述用户感兴趣的领域。



### 3. 知识表示实例

### 2. 规则

规则是一个有头Horn子句，它的一般格式为：

关系（对象，对象，...，对象）：—

关系（对象，对象，...，对象）and

.....

关系（对象，对象，...，对象）。

其中，“：—”表示if（如果），其左边为“结论”，是总目标，其右边为前提，是子目标。该规则是逆向规则。

例如，检索操作可用一个规则表示：

检索（用户名，书号）：—用户（用户名，X）and

书籍（书号，X）。

它的含义是：如果某个用户对项目X感兴趣，并且某本书籍中包含项目X，那么这个用户对该书籍感兴趣。

用户的提问可用无头Horn子句表示：

? : —E, F 或者

? E, F 它的意思是，查找X使得E、F均为真。



### 3. 知识表示实例

### 3. 知识库

事实和规则的集合就构成了知识库，现给出知识库的例子如下：

书籍 (00001, 云计算, 张三, 电子工业出版社, 2015, 云计算技术) ;

书籍 (00002, 大数据, 李四, 清华大学出版社, 2016, 云计算大数据技术) ;

书籍 (00003, 人工智能, 王万森, 北邮出版社, 2016, 云计算大数据技术) ;

用户 (张明, 云计算大数据技术) ;

用户 (张亮, 云计算技术) ;

检索 (用户名, 书号) : —用户 (用户名, X) and

书籍 (书号, X) 。

基于以上知识库就可以执行检索操作。Prolog查找机制解决用户提问，是利用定理证明技术叫做归结 (resolution) 。

?书籍 (云计算, 张三)

系统用提问谓词“书籍”与知识库中的事实进行匹配成功，回答为“true”。

?书籍 (大数据, 2017)

回答: false

?书籍 (人工智能, 出版日期)

回答: 出版日期=2016

其中，“出版日期”是变量

?书籍 (著者, 云计算大数据技术)

对于该提问，系统查找使提问为真的所有变量值，结果为：著者=“李四”和著者=“王万森”。

?检索 (张明, X)



### 3. 知识表示实例

### 3. 知识库

其检索处理如下：

1) 自上而下搜索知识库，**查找**与提问目标匹配的事实，失败。于是用目标匹配规则的结论部分，变量“用户名”的值约束为“张明”，然后执行演绎推理求得答案。变量“用户名”的值传递给规则前提的所有子目标。

2) 系统证明前提部分中第一个子目标“用户（张明，X）”，在知识库里寻找与其匹配的事实，匹配成功，变量X的值约束为“云计算大数据技术”。

3) 第二个子目标为“书籍（书号，云计算大数据技术）”。同理，搜索知识库，子目标匹配两条书籍事实，书号的值为00002与00003，张明检索到两条书籍信息。



### 4. 谓词逻辑表示的特点和问题

一阶谓词逻辑的知识表示具有如下优点：

- 1) 逻辑表示是说明型表示，和其它知识表示形式相比，它是一种接近于自然语言的形式语言，使得句子很容易被人们理解。
- 2) 谓词逻辑能很准确地表示知识。
- 3) 它拥有通用的逻辑演算方法和推理规则，并保证推理过程的完全性。
- 4) 模块性能较好，谓词逻辑表示方法可以把知识分成小单元，用模块的形式来储存。

另一方面，这种表示方法所能表示的事物**过于简单**，不能很方便地描述有关领域中的复杂结构。此外，使用这种方法的**效率低**，逻辑推理过程往往太冗长，当用于大型知识库时，可能会发生“组合爆炸”。

## 第二章 知识表示

2.1 知识表示的内涵

2.2 谓词逻辑表示法

2.3 产生式规则表示法

2.4 语义网络表示

2.5 知识图谱表示

2.6 框架表示法

2.7 脚本表示法

习题





### 1. 正向规则和逆向规则

正向规则的一般形式是：

如果 <前提> 则 <结论> (if premise then conclusion) 或

如果 <情况> 则 <行为> (if case then action)

前提可由一个或多个条件组成，可使用逻辑运算符and及or连接起来。结论中仅包含一个条件。**一条规则表示，结论要在逻辑上或行为方式上符合于前提条件。**在调用一条规则时，系统首先检查一下前部分的值是否为真。若前提为假，系统就停止处理该规则；若前提为真，则结论的值也为真，或执行结论中有关行为。

例如：如果 体温超过37.5度则提示发烧。

逆向规则的一般形式是：

<结论> 如果 (if) <前提>

在这种格式中，前提和结论的顺序被逆转。人们称它为逆向规则 (backward rule)。在Prolog语言中，if用 “:—” 代替。

例如：提示发烧，如果 体温超过37.5度。这两种格式都是常用的表示形式。



### 2. 确定和不确定规则

我们知道在知识表示中一些规则可能是真的，但规则也不全都是真的。有些规则可能是不确定的或不精确的规则。我们可以在规则中加入确定性因子来表达不确定知识，**规则的确定性因子表示了规则为真的程度。**

例如：

如果 体温超过37.5

并且 伴随咳嗽

则有80%的把握：这个人感冒了。

其中**80%**表示了结论“感冒”的确定性因子。



### 3. 特殊和一般性规则

知识表示中的规则也可以根据不同的概括级别分为特殊规则和一般性规则。

规则表达的越一般，则应用越广泛。

例如以下例子中规则1和规则2为特殊规则，我们可以整理表达成一般性规则，应用的范围就广一些。

规则1:

如果 一个成人体温每上升1度  
则心跳会增加6~10次/分;

规则2:

如果 一个儿童体温每上升1度  
则儿童心跳会增加15~20次/分;

我们可将这两条特殊规则表达为一条一般性规则:

规则3:

如果 一个人体温每上升1度  
则心跳会增加6~20次/分;



### 4. 元规则 (metarules)

目前，规则表示法中存在许多使用规则的方法，但是其中有一种规则用于控制其它规则的行为，这种规则称为**元规则**。

当知识库中的规则逐渐增多时，就必须开发元规则来组织和管理知识库中规则的活动。

例如：

如果 顾客的年龄大于65岁 （规则集合1）

且规则集合1的前提中包含热爱广场舞的信息 （规则集合2）

且规则集合1的前提中包含无心脏疾病的信息 （规则集合3）

则优先应用规则集合1，后用规则集合2。

建立这种元知识是困难的，因为元规则是更一般性的规则，常存在例外。在上例中，也许存在一些老年人，他们热爱广场舞，身体可能具有心脏病。

## 第二章 知识表示

2.1 知识表示的内涵

2.2 谓词逻辑表示法

2.3 产生式规则表示法

2.4 语义网络表示

2.5 知识图谱表示

2.6 框架表示法

2.7 脚本表示法

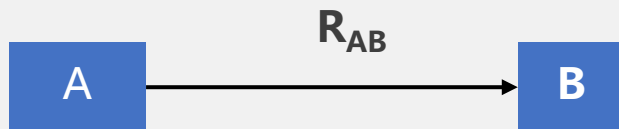
习题



### 1. 基本语义关系

语义网络通过概念及其语义关系组成的有向图来表达知识和描述语义。

一个语义网络是由一些有向图表示的三元组（节点1，弧，节点2）连接而成的。节点用来表示物理实体、概念或者状态，弧表示节点1与节点2之间的关系。三元组用图表示，称为一个基本网元。



其中，A，B分别代表两个节点； $R_{AB}$ 表示A与B之间的某种语义关系。

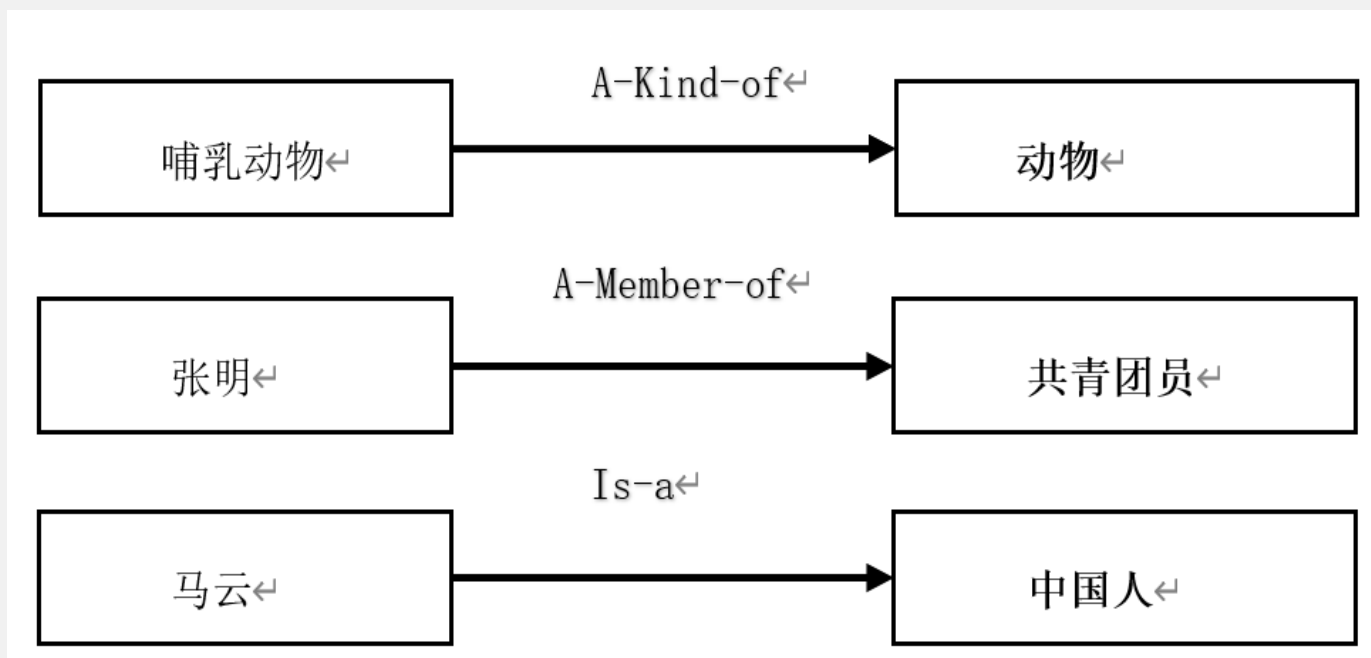


### 1. 基本语义关系

弧的定义有多种方法，依赖于所表示的知识类型。弧所表示的各种关系可以归纳为以下五类。

1) **类属关系** (A-Kind-of)：是指具有共同性质的不同事物间的分类关系、成员关系或实例关系。

例如，哺乳动物是动物的一种类型。张明是一名共青团员。马云是一名中国人

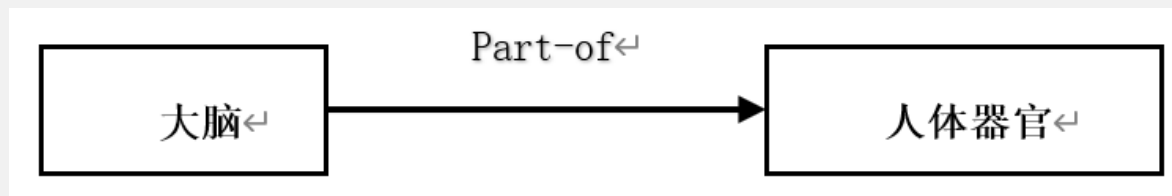




### 1. 基本语义关系

#### 2) 包含关系 (部分与整体之间的关系)

包含关系 (Part-of) 表示子集合与集合之间的关系。例如，大脑是人体的一部分，也就是大脑是人体的子集。

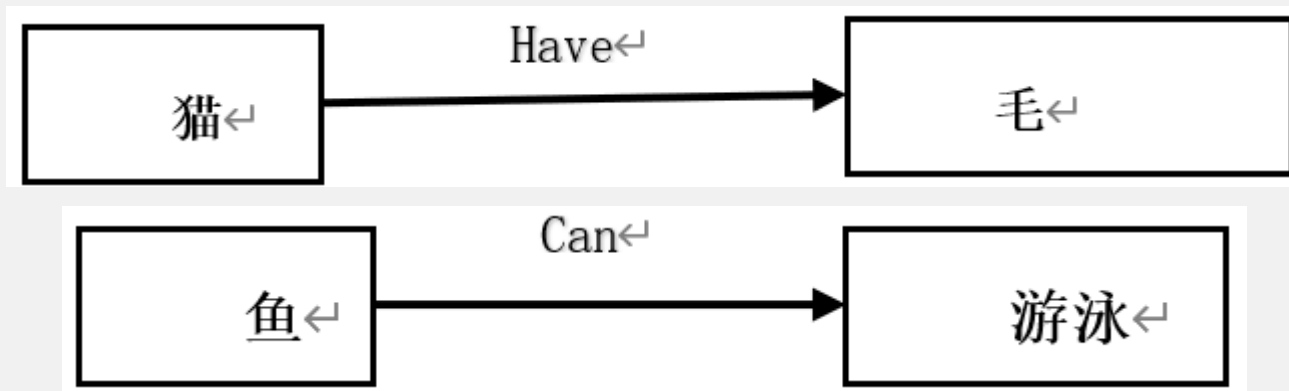


#### 3) 属性关系 (有、能.....)

属性关系是指事物和其属性之间的关系：

Have: 含义为“有”，表示一个结点具有另一个属性。

Can: 含义是“能”、“会”，表示一个事物能做另一件事情。





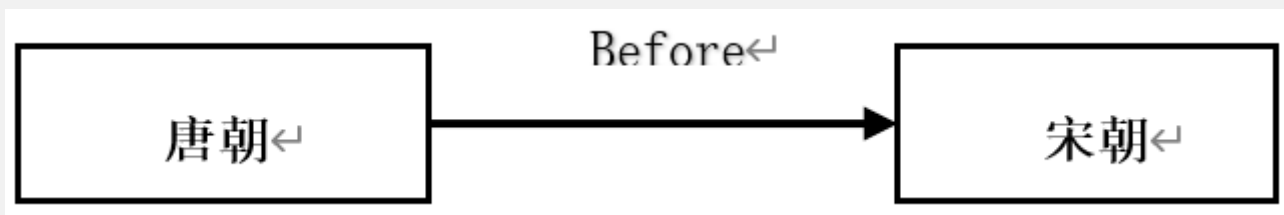


#### 1. 基本语义关系

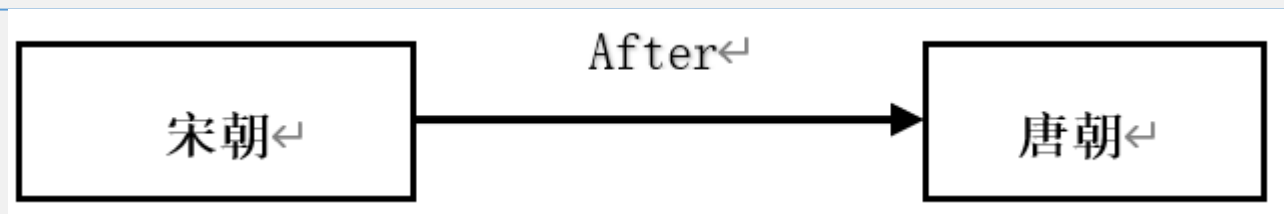
##### 4) 时序关系 (之前、之后.....)

时序关系是指不同事件在其发生时间方面的先后次序关系。

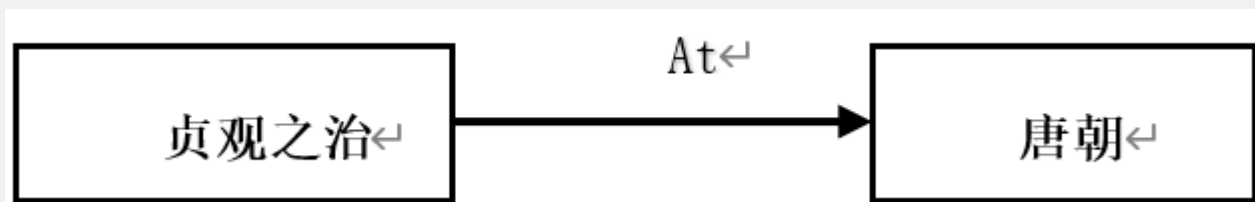
Before: 表示一个事件在另一个事件之前发生。



After: 表示一个事件在另一个事件之后发生。



At: 表示某一事件发生的时间。





### 1. 基本语义关系

#### 5) 位置关系 (前方、上方.....)

位置关系指不同事物在位置方面的关系。常用的位置关系有:

Located-on: 含义为“在上”, 表示某一物体在另一物体之上

Located-at: 含义为“在”, 表示某一物体所在的位置

Located-under: 含义为“在下”, 表示某一物体在另一物体之下

Located-inside: 含义为“在内”, 表示某一物体在另一物体之内;

Located-outside: 含义为“在外”, 表示某一物体在另一物体之外。

#### 6) 相近关系 (类似、相反.....)

相近关系 指不同事物在形状、内容等方面相似或接近。常用的相近关系有:

Similar-to: 含义为“相似”, 表示某一事物与另一事物相似;

Near-to: 含义为“接近”, 表示某一事物与另一事物接近。

例如, “猫似虎”。



#### 1. 基本语义关系

##### 7) 推论关系等 (推出.....)

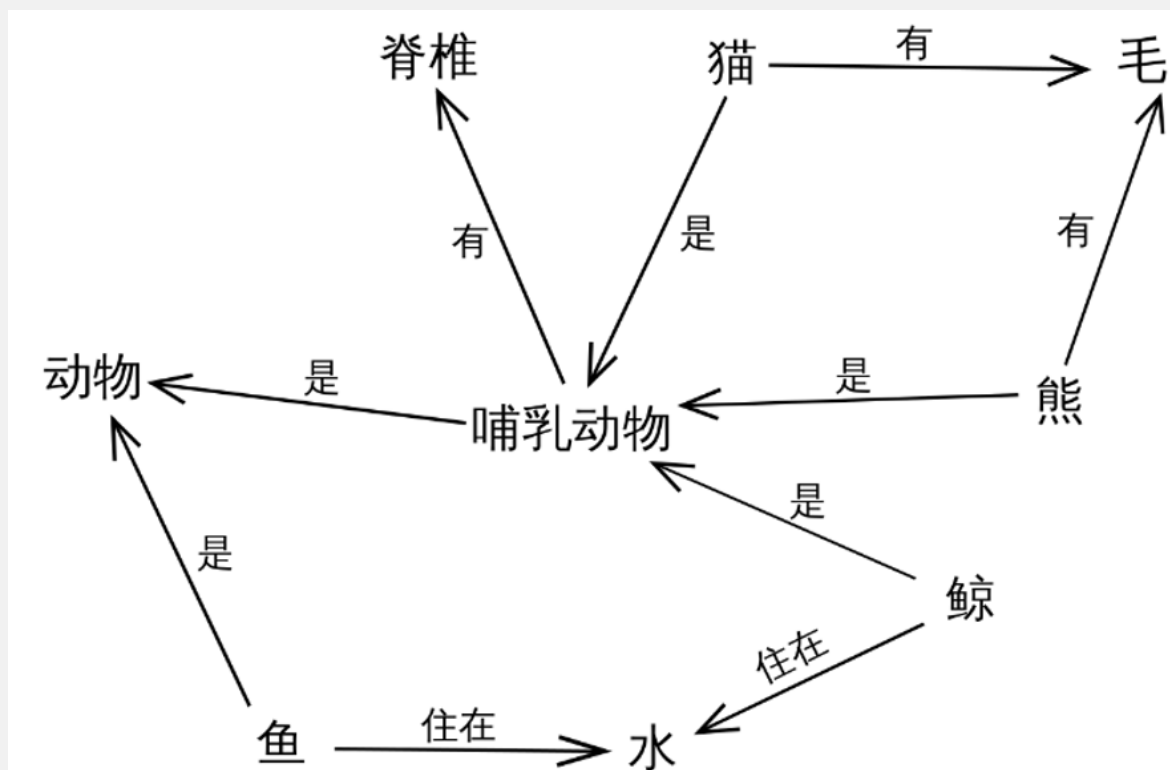
如果一个概念可有另一个概念推出, 则称它们存在推论关系。比如: 一个人体温超过37.5, 推出这个人发烧。





### 2. 语义网络的结构

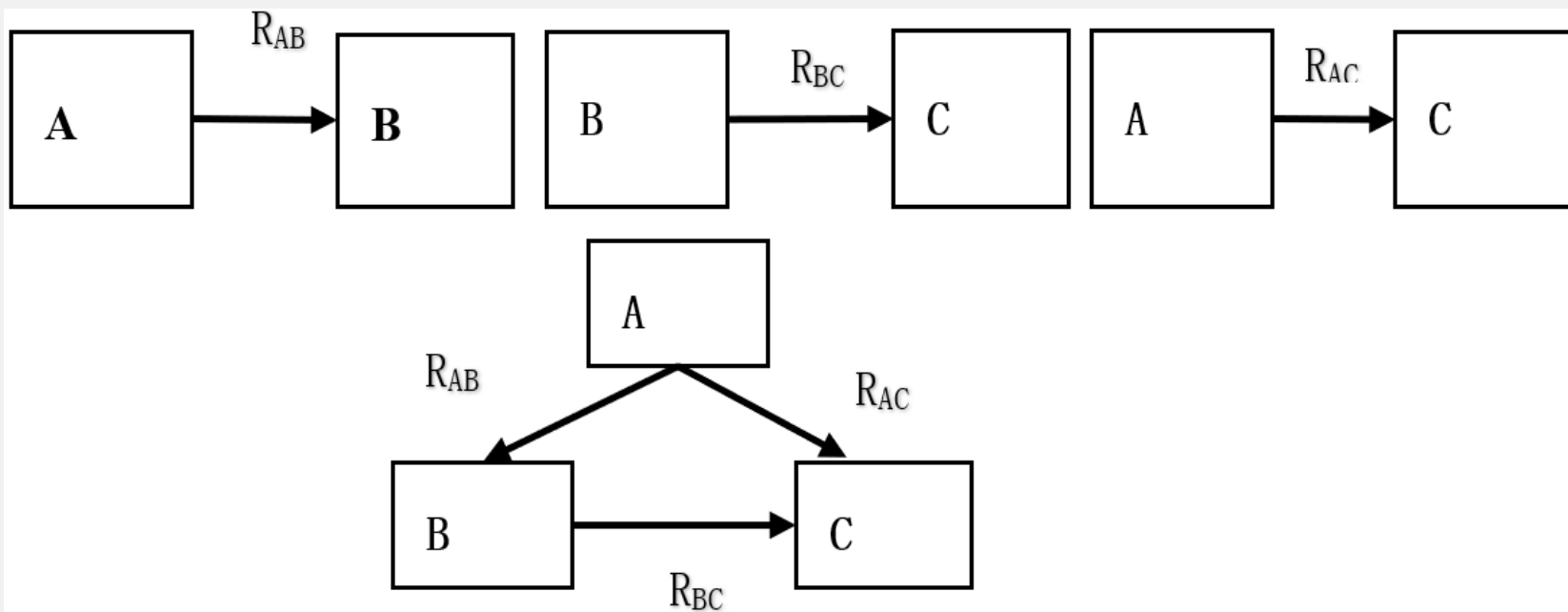
语义网络是知识的图解表示，它的逻辑结构是一种**二元关系有向图**。其用相互连接的**节点**和**边**来表示知识。节点表示对象、概念，边表示节点之间的关系。语义网络由结点和弧（有向线段）组成。结点表示事物、概念、事件等，弧表示结点之间的关系。结点和弧也可带有权值，以表示其有关的重要程度。





### 3. 知识的语义网络表示

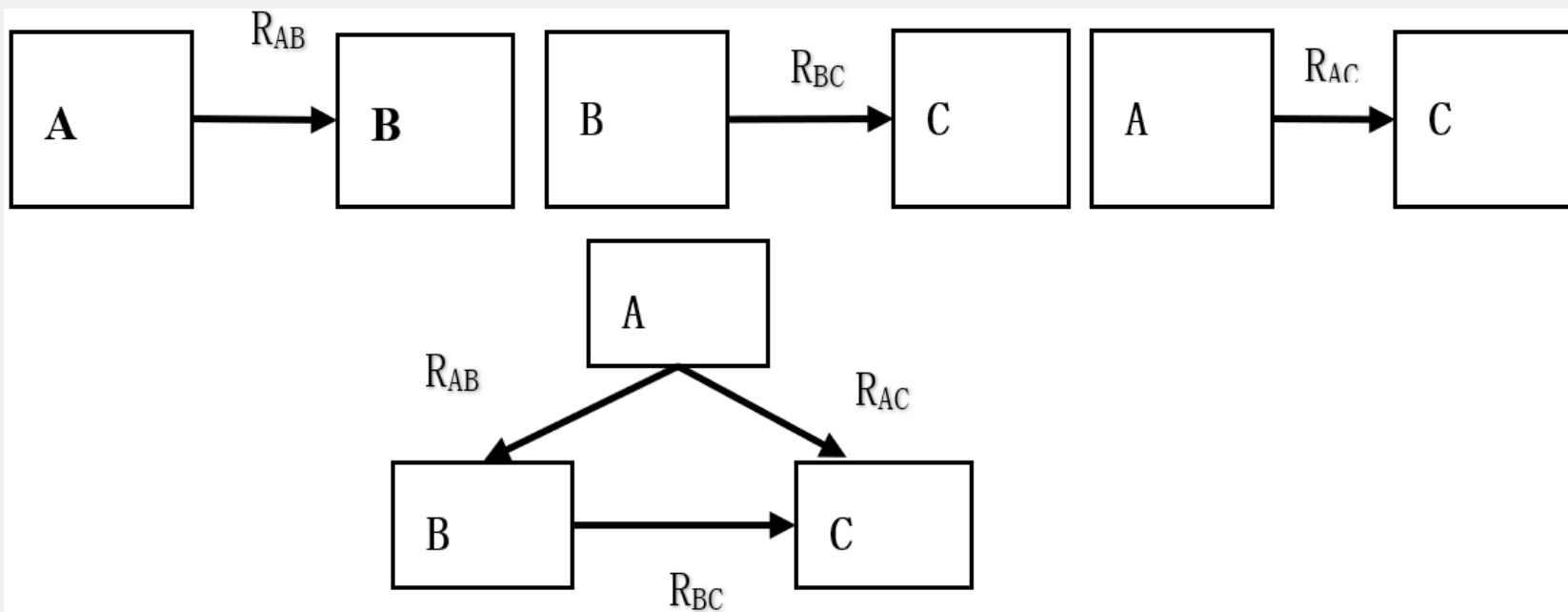
当把多个基本网元用相应语义关系关联到一起时，我们就得到了一个语义网络。  
例如我们把三个基本网元，经合并后就可以得到一个语义网络。





### 3. 知识的语义网络表示

当把多个基本网元用相应语义关系关联到一起时，我们就得到了一个语义网络。  
例如我们把三个基本网元，经合并后就可以得到一个语义网络。



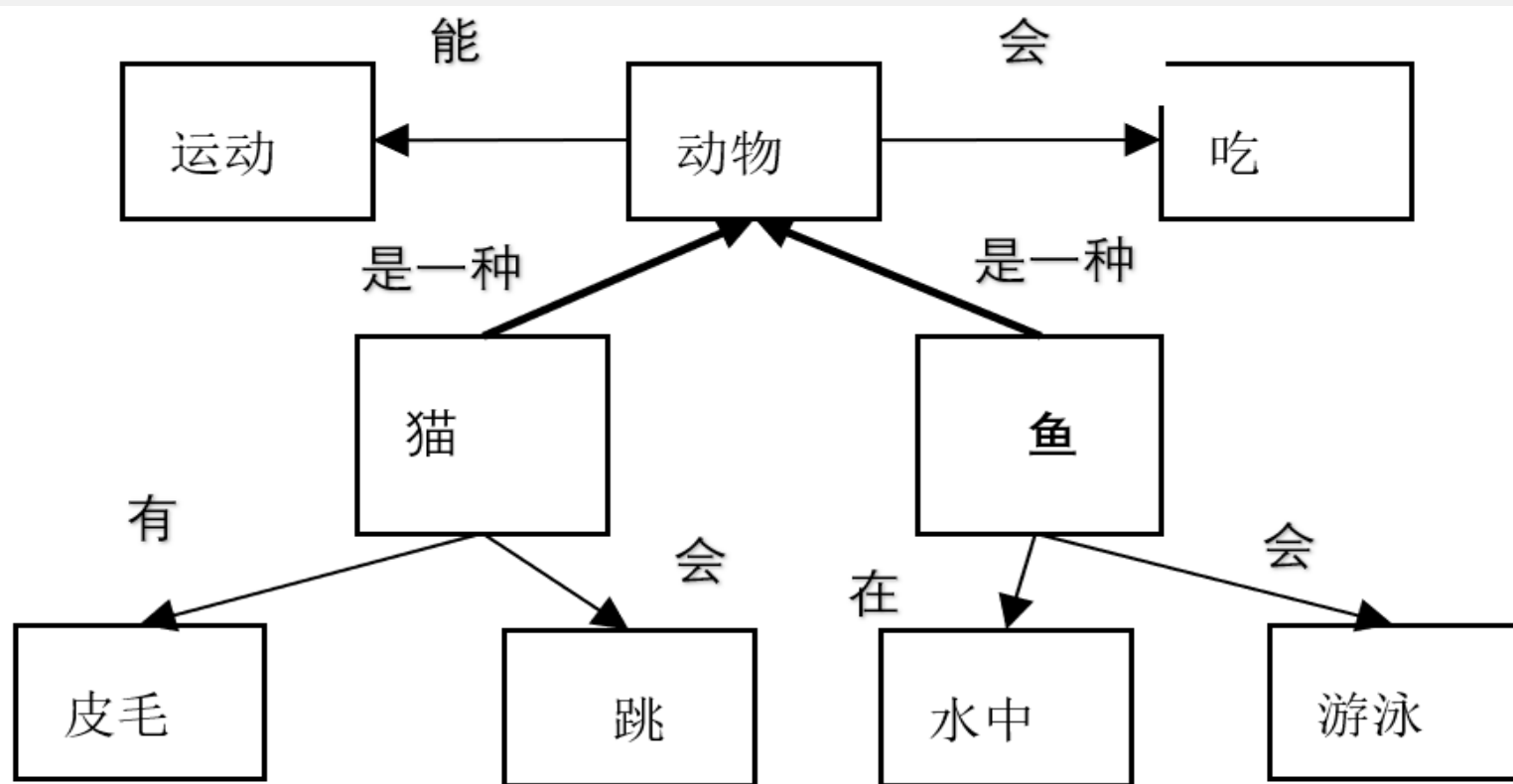


### 3. 知识的语义网络表示

#### 1) 事实或概念的表达

用节点1表示实体，用节点2表示实体的性质或属性等，用弧表示节点1和节点2之间的语义关系。

例如：动物能运动、会吃。猫是一种动物，猫有皮毛、会跳。鱼是一种动物，鱼生活在水中、会游泳。



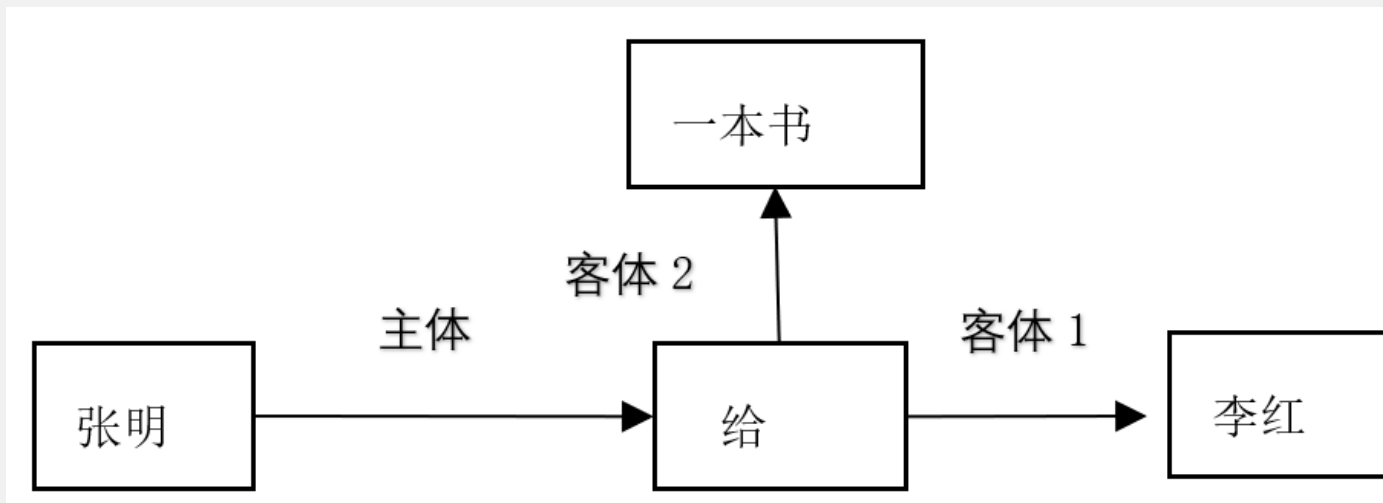


### 3. 知识的语义网络表示

#### 2) 情况和动作的表示

Simon (西蒙) 提出的知识表示方法中增加了情况节点和动作节点，用一个节点来表示情况或动作。

例如：张明给李红一本书。



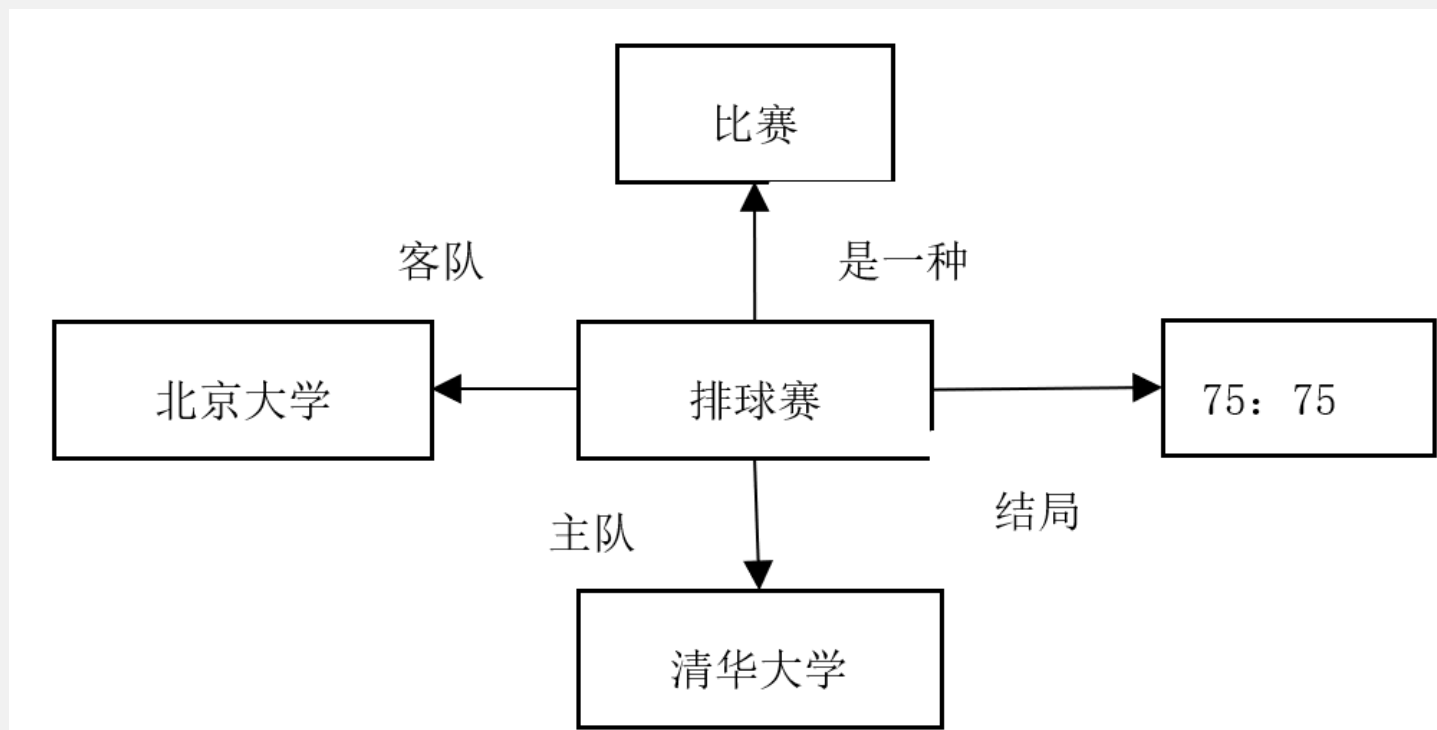




### 3. 知识的语义网络表示

#### 2) 情况和动作的表示

清华大学和北京大学两校的排球队在清华大学进行一场比赛，结局的比分是75：75。





### 4 语义网络的推理过程

这种推理过程，类似于人的思维过程。一旦知道了某种事物的身份以后，可以联想起很多关于这件事物的一般描述。例如，通常认为鲸鱼很大，鸟比较小，城堡很古老，运动员很健壮等。

语义网络中的推理过程主要有两种，一种是继承，另一种是匹配。



### 4 语义网络的推理过程

#### 1. 继承

继承就是把对事物的描述从概念节点或类节点传递到实例节点。

Pass (遗留) — the descendants get the properties from the ancestors

Add (添加) — the descendants expand the properties of the ancestors

Exclude(排除) — stop inheriting the properties

共有3种继承过程：值继承、“如果需要”继承和“缺省”继承。

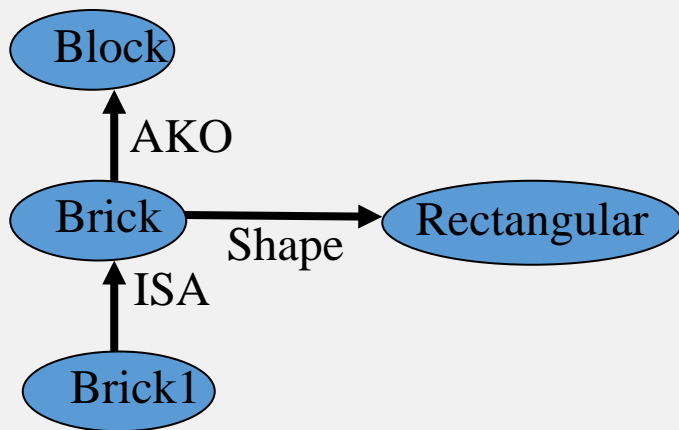


### 4 语义网络的推理过程

#### (1) 值继承：

is-a, AKO (a kind of)

What is the shape of Brick1?

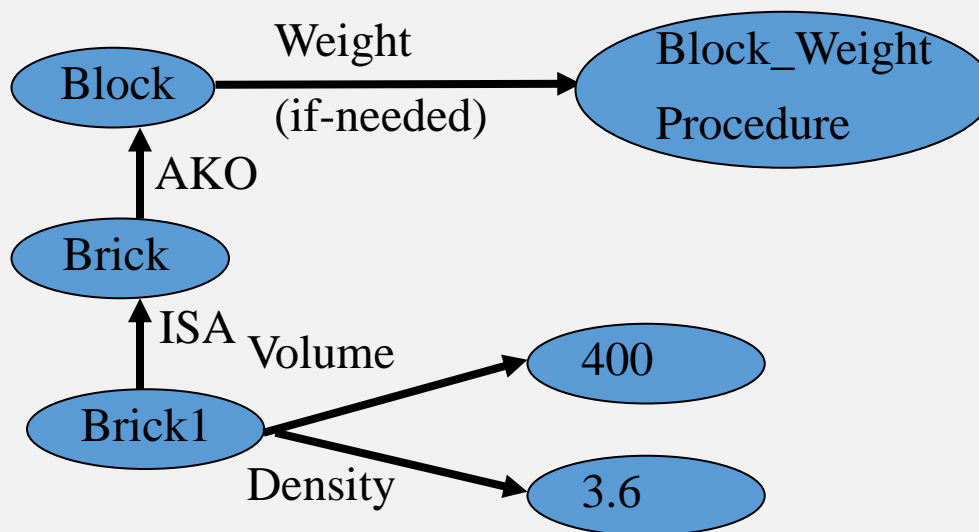




### 4 语义网络的推理过程

#### (2) “如果—需要” 继承：

if-needed (it can not inherit from the ancestors, we get it from other programs )



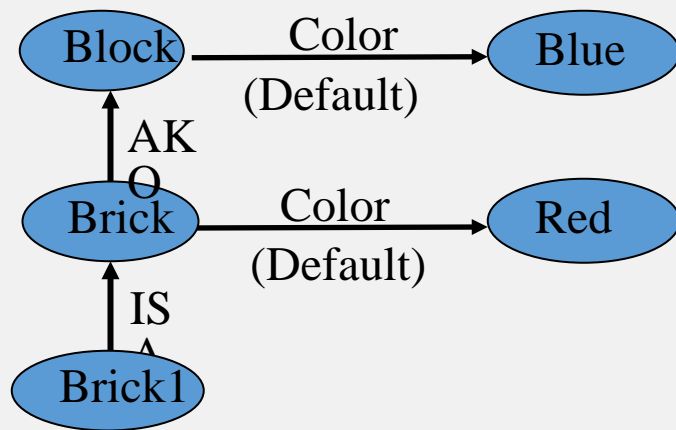


### 4 语义网络的推理过程

#### (3) “缺省”继承：

Default—it is mostly the truth

Example: the birds can fly

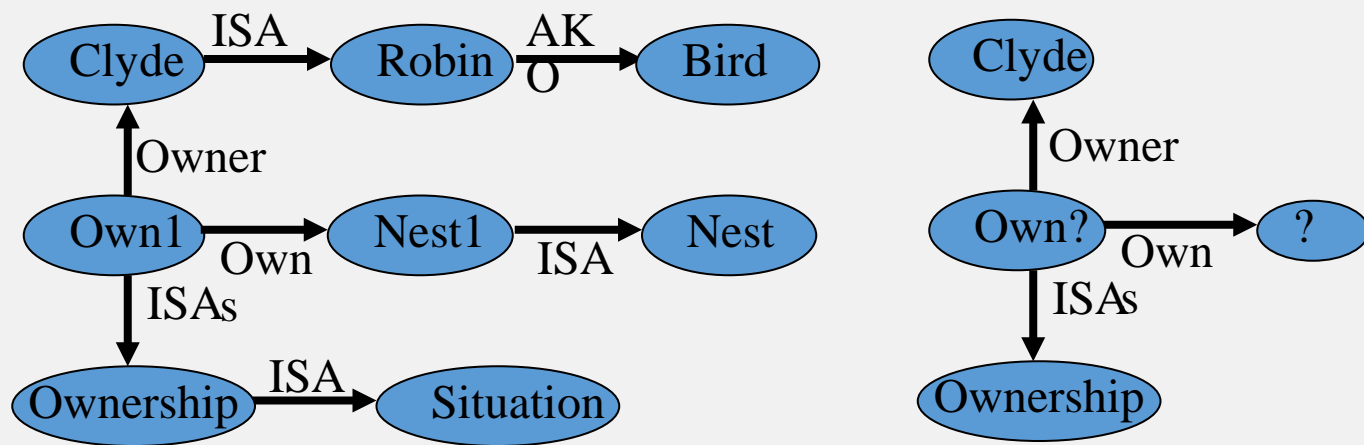




### 4 语义网络的推理过程

#### 2. 匹配 (Matching)

当涉及由几个部分组成的事物时，必须考虑值的传递问题。



What does Clyde own?



### 4. 语义网络的推理过程

**语义推理**指的是依据词项之间的语义关系而进行的推理。

例如，从“张明是上海人”可推出“有人是上海人”和“张明是中国人”等。

这种推理不同于命题逻辑或谓词逻辑中的形式推理，这种推理所依据的是词项“张明”与“人”、“上海人”与“中国人”之间具体的语义关系。

语义网络的推理过程主要有两种：

#### 继承

把对事物的描述从抽象节点**传递**到具体节点，通常是沿着IS-a、A-Kind-of等继承弧进行。通过继承可以得到所需节点的一些属性值。

#### 匹配

在知识库的语义网络中寻找与待求解问题相符的语义网络模式。

- 1) 根据待求解问题的要求构造一个网络片段，该网络片段中有些节点和弧的标识是空的，成为询问处，它反映的是待求解的问题。
- 2) 根据该语义片段到知识库中寻找所需要的信息。
- 3) 当待求解问题的网络片段与知识库中的某语义网络片段相匹配时，则与询问处相匹配的事实就是该问题的解。





### 5. 语义网络表示法的特点

语义网络的**主要优点**是可以用来表示复杂的知识结构，它**侧重于表示语义关系知识，还体现了联想思维过程**，给我们提供了很自然的知识表示构架。

人们基于联想记忆模型，可执行语义搜索，把相关事实从其直接相连的结点中推导出来，而不必遍历整个庞大的知识库，从而避免了组合爆炸。人们可以利用等级关系建立分类层次结构实现继承推理；也可利用继承特性，实现信息共享，将结点的公共性质存放于较高层结点中，可被子孙结点继承。因此语义网络很适合表示专业领域知识，如叙词表。

语义网络的**主要缺点**是因为目前的网络还**缺乏标准的术语和约定**，语义解释取决于操作网络的程序，所以造成网络结构复杂，建立和维护知识库较困难。

因此网络搜索、调控的执行效率需要制定强有力的原则。在语义网络表示中，由于没有形式语义，没有统一的结构模型，人们根据不同的需求可以构成不同类型的语义网络（如重视联想的、重视推理的、表示词语的等）。人们在应用过程中，一部分网络用说明型方法表示知识，从演绎推理的角度来研究，发展成为另一类实用的知识表示方法，如框架表示法。

## 第二章 知识表示

2.1 知识表示的内涵

2.2 谓词逻辑表示法

2.3 产生式规则表示法

2.4 语义网络表示

2.5 知识图谱表示

2.6 框架表示法

2.7 脚本表示法

习题



### 1. 知识图谱的定义

辛格博士在介绍知识图谱时提到的：“The world is not made of strings , but is made of things.”

**知识图谱旨在描述真实世界中存在的各种实体或概念。**其中，每个实体或概念用一个全局唯一确定的ID来标识，称为它们的标识符(identifier)。每个属性-值对(attribute-value pair, 又称AVP)用来刻画实体的内在特性，而关系用来连接两个实体，刻画它们之间的关联。知识图谱亦可看作是一张巨大的图，图中的**节点**表示实体或概念，而图中的**边**则由属性或关系构成。

①实体：实体指的是具有可区别性且独立存在的某种事物。

世间万物即为实体，如某一种动物、某一个人、某一种植物、某一种商品。

②语义类（概念）：主要指集合、类别、对象类型、事物的种类，如同人物、地理等。

③内容：内容通常作为实体和语义类的名字、描述、解释等，可以由文本、图像、音视频等来表达。

④属性(值)：指一个实体指向它的属性值。

不同的属性类型对应不同类型属性的边。属性值主要指对象指定属性的值。

⑤关系：关系形式化为一个函数，它把K个点映射到一个布尔值。在知识图谱上，关系则是一个把K个图节点(实体、语义类、属性值)映射到布尔值的函数。



#### 1. 知识图谱的定义

三元组的基本形式主要包括(实体1—关系—实体2)和(实体—属性—属性值)等。每个实体(概念的外延)可用一个全局唯一确定的ID来标识，每个属性—属性值对AVP可用来刻画实体的内在特性，而关系可用来连接两个实体，刻画它们之间的关联。

换句话说，**知识图谱是由一条条知识组成，每条知识表示为一个主语—谓语—宾语（SPO）三元组。**



**主语主题**可以是国际化资源标识符internationalized resource identifiers (IRI) 或空白节点 (blank node) 。

**谓语属性**通常是国际化资源标识符。

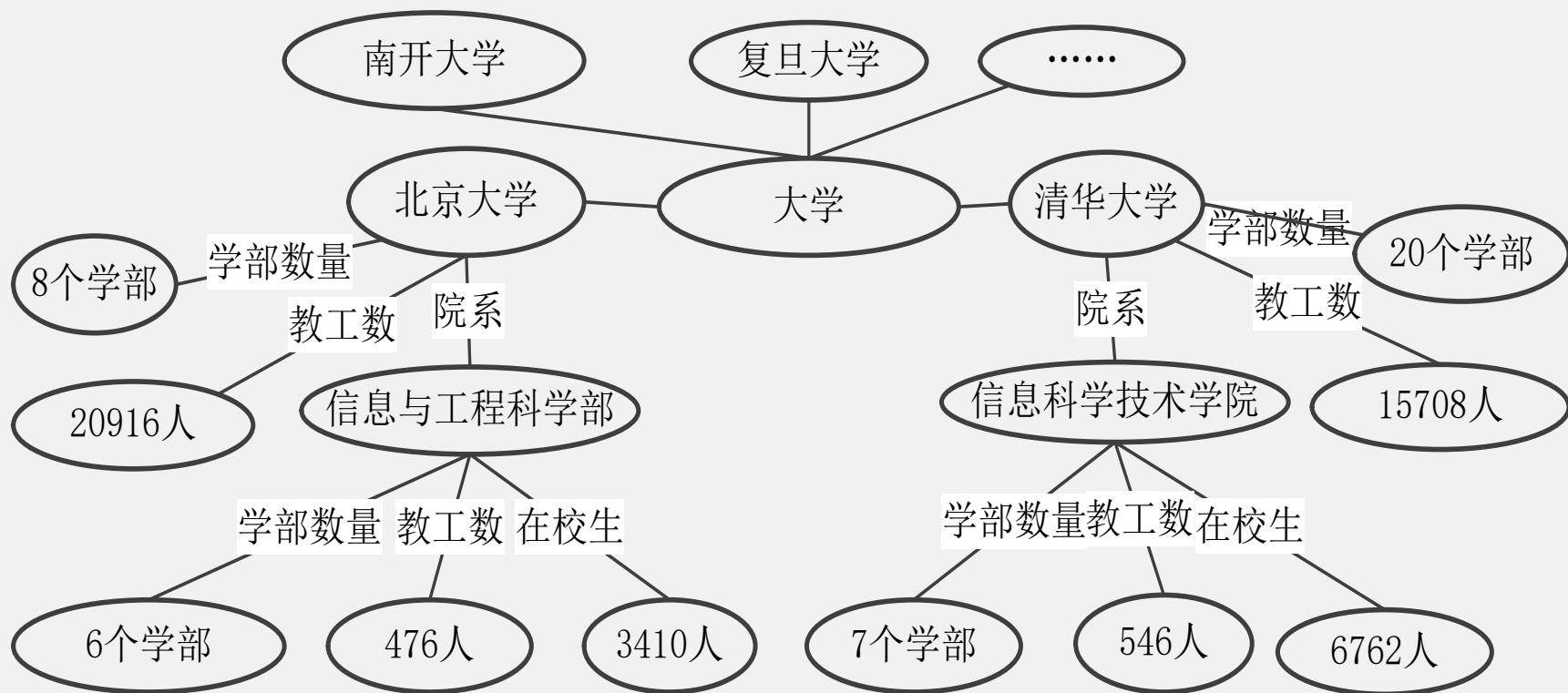
**宾语客体**是国际化资源标识符、空白节点或常量。



### 1. 知识图谱的定义

清华大学是一个实体，北京大学是一个实体，北京大学—院系—信息与工程科学部是一个（**实体—关系—实体**）的三元组样例。

北京大学是一个实体，教工数是一种属性，20916是属性值。北京大学—教工数—20916构成一个（**实体—属性—属性值**）的三元组样例。



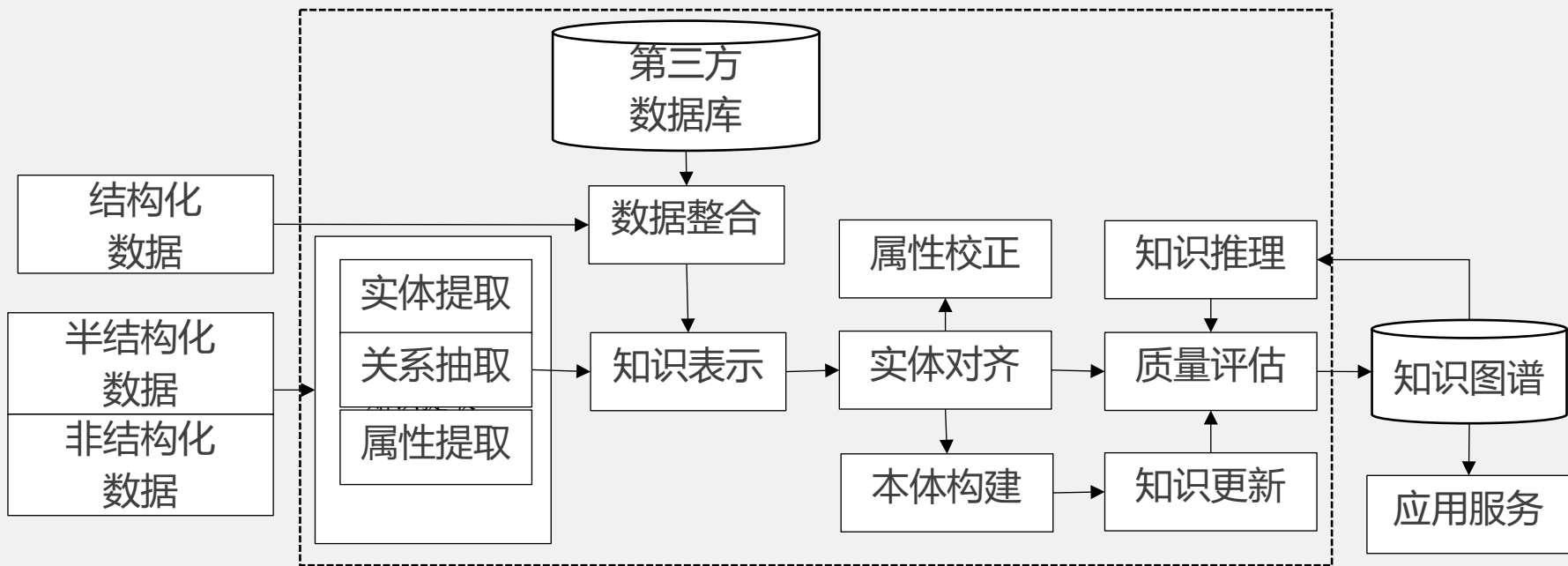


### 2. 知识图谱的架构

#### 1) 知识图谱的逻辑结构

知识图谱在逻辑上可分为模式层与数据层两个层次，数据层主要是由一系列的事实组成，而知识将以事实为单位进行存储。如果用(实体1, 关系, 实体2)、(实体、属性, 属性值)这样的三元组来表达事实，可选择图数据库作为存储介质，模式层构建在数据层之上，是知识图谱的核心，通常采用本体库来管理知识图谱的模式层。本体是结构化知识库的概念模板，通过本体库而形成的知识库不仅层次结构较强，并且冗余程度较小。

#### 2) 知识图谱的体系架构





### 3. 知识图谱构建的关键技术

#### 1) 知识提取

知识抽取主要是面向开放的链接数据，通常典型的输入是自然语言文本或者多媒体内容文档（图像或者视频）等。然后通过自动化或者半自动化的技术抽取出的知识单元，知识单元主要包括实体(概念的外延)、关系以及属性3个知识要素，并以此为基础，形成一系列高质量的事实表达，为上层模式层的构建奠定基础。

实体抽取：基于百科或垂直站点提取、基于规则与词典的实体提取方法、基于统计机器学习的实体抽取方法、面向开放域的实体抽取方法。

2) 语义类抽取：并列相似度计算、上下位关系提取、语义类生成。

#### 3) 属性和属性值抽取

属性提取的任务是为每个本体语义类构造属性列表（如城市的属性包括面积、人口、所在国家、地理位置等），而属性值提取则为一个语义类的实体附加属性值。

#### 4) 关系抽取

关系抽取的目标是解决实体语义链接的问题。关系的基本信息包括参数类型、满足此关系的元组模式等。



### 4. 知识图谱的表示方法

以深度学习为代表的表示学习技术取得了重要的进展，可以将实体的语义信息表示为稠密低维实值向量，进而在低维空间中高效计算实体、关系及其之间的复杂语义关联，对知识库的构建、推理、融合以及应用均具有重要的意义。

#### 1) 代表模型

距离模型

- 距离模型提出了知识库中实体以及关系的结构化表示方法SE，其基本思想是：首先将实体用向量进行表示，然后通过关系矩阵将实体投影到与实体关系对的向量空间中，最后通过计算投影向量之间的距离来判断实体间已存在的关系的置信度。

单层神经网络模型

- 针对上述提到的距离模型中的缺陷，提出了采用单层神经网络的非线性模型SLM，模型为知识库中每个三元组  $(h, r, t)$  定义了一个评价函数。单层神经网络模型的非线性操作虽然能够进一步刻画实体在关系下的语义相关性，但在计算开销上却大大增加。

双线性模型

- 双线性模型又叫隐变量模型LFM，主要是通过基于实体间关系的双线性变换来刻画实体在关系下的语义相关性。模型不仅形式简单、易于计算，而且还能够有效刻画实体间的协同性。





#### 4. 知识图谱的表示方法

神经张量  
模型

- 在不同的维度下，将实体联系起来，表示实体间复杂的语义联系。神经张量模型在构建实体的向量表示时，是将该实体中的所有单词的向量取平均值，这样一方面可以重复使用单词向量构建实体，另一方面将有利于增强低维向量的稠密程度以及实体与关系的语义计算。

矩阵分解  
模型

- 通过矩阵分解的方式可得到低维的向量表示，故不少研究者提出可采用该方式进行知识表示学习，知识库中的三元组集合被表示为一个三阶张量，如果该三元组存在，张量中对应位置的元素被置1，否则置为0。

翻译模型

- 将知识库中实体之间的关系看成是从实体间的某种平移，并用向量表示。该模型的参数较少，计算的复杂度显著降低。与此同时，翻译模型在大规模稀疏知识库上也同样具有较好的性能和可扩展性。



### 4. 知识图谱的表示方法

#### 2) 复杂关系模型

知识库中的实体关系类型也可分为1-to-1、1-to-N、N-to-1、N-to-N 4种类型，而复杂关系主要指的是1-to-N、N-to-1、N-to-N的3种关系类型。

TransH模型

- TransH模型尝试通过不同的形式表示不同关系中的实体结构，对于同一个实体而言，它在不同的关系下也扮演着不同的角色。TransH使不同的实体在不同的关系下拥有了不同的表示形式，但由于实体向量被投影到了关系的语义空间中，故它们具有相同的维度

TransR模型

- 由于实体、关系是不同的对象，不同的关系所关注的实体的属性也不尽相同，将它们映射到同一个语义空间，在一定程度上就限制了模型的表达能力。模型首先将知识库中的每个三元组(h, r, t)的头实体与尾实体向关系空间中投影，然后希望满足给定的约束关系，最后计算损失函数。



### 4. 知识图谱的表示方法

#### TransD模型

- 之前的TransR模型使它们被同一个投影矩阵进行映射，在一定程度上就限制了模型的表达能力。除此之外，将实体映射到关系空间体现的是从实体到关系的语义联系，而TransR模型中提出的投影矩阵仅考虑了不同的关系类型，而忽视了实体与关系之间的交互。因此，TransD模型分别定义了头实体与尾实体在关系空间上的投影矩阵。

#### TransG模型

- TransG模型认为一种关系可能会对应多种语义，而每一种语义都可以用一个高斯分布表示。TransG模型考虑到了关系 $r$ 的不同语义，使用高斯混合模型来描述知识库中每个三元组 $(h, r, t)$ 头实体与尾实体之间的关系，具有较高的实体区分。



### 5. 知识图谱在搜索中的典型应用

#### 1) 查询理解

**搜索引擎**借助知识图谱来识别查询中涉及到的实体(概念)及其属性等, 并根据实体的重要性展现相应的知识卡片。搜索引擎并非展现实体的全部属性, 而是根据当前输入的查询自动选择最相关的属性及属性值来显示。此外, 搜索引擎仅当知识卡片所涉及的知识正确性很高(通常超过95%, 甚至达到99%)时, 才会展现。当要展现的实体被选中之后, 利用相关实体挖掘来推荐其他用户可能感兴趣的实体供进一步浏览。

#### 2) 问题回答

除了展现与查询相关的知识卡片, 知识图谱对于搜索所带来的另一个革新是: **直接返回答案**, 而不仅仅是排序的文档列表。要实现自动问答系统, 搜索引擎不仅要理解查询中涉及到的实体及其属性, 更需要理解查询所对应的语义信息。搜索引擎通过高效的图搜索, 在知识图谱中查找连接这些实体及属性的子图并转换为相应的图查询。这些翻译过的图查询被进一步提交给图数据库进行回答返回相应的答案。

## 第二章 知识表示

2.1 知识表示的内涵

2.2 谓词逻辑表示法

2.3 产生式规则表示法

2.4 语义网络表示

2.5 知识图谱表示

2.6 框架表示法

2.7 脚本表示法

习题



### 1. 框架的构成

#### 1) 名字

框架具有**唯一的名字**，它提供一个标志，可为任何常量。

#### 2) 描述

描述是框架的主体，由**任意有限数目的槽**组成。这些槽是数据和过程的组合模块，用于描述对象的性质（属性）或连接不同的其它框架。每个槽包含槽的名字和槽的值。一个框架中的每个槽具有唯一的名字，它局限于框架。因而不同的框架可以包含相同的槽名。每个槽有一个值侧面（存放属性值），它可具有一个或多个值，也可以是默认值。默认值是在缺乏更具体的知识时被假定的一个值。有些情况，根据对象的类型可知它必须具有某种特征，但不知道该特征的具体值，又不能设默认值。

#### 3) 约束

约束是**每个槽可包含一组有关约束条件**，如约束槽值的类型、数量等。这些约束可用若干侧面表示。一种侧面表示槽值的最少和最多个数；一种侧面描述槽值的类型和取值范围；例如一个人的年龄必须是整型数字。另一种侧面是附加过程：如果加入过程（if—added）、如果删除过程（if—deleted）、如果需要过程（if—needed），它们描述对象的行为特征，用于控制槽值的存储和检索。



### 1. 框架的构成

#### 4) 关系

关系表达框架对象之间的知识关联，包括：等级关系、语义相似关系、语义相关关系等**静态关联**，还有框架之间的互操作等**动态关联**。每个框架可以有一个或多个父辈结点，通过父—子链表达等级关系。框架中槽的值也可以是连接其它框架的链值。因此，框架可以通过槽的值相互关联，还可以使用规则相互动态连接。当一个系统中的各个不同框架共享同一个槽时，这个槽可以把从不同角度收集来的信息相互协调起来。

一个框架的基本结构由框架名、关系、槽、槽值及槽的约束条件与附加过程所组成。框架的一般描述形式如下：

《框架名》

《关系》

《槽名1》 《值1》 《约束1》 《过程1》

《槽名2》 《值2》 《约束2》 《过程2》

.....

《槽名n》 《值n》 《约束n》 《过程n》。



### 1. 框架的构成

#### 4) 关系

例如：一个描述“大学教师”的框架：

框架名——大学教师

类属——职业：<教师>

槽名——学位：（学士，硕士，博士）

缺省：硕士

槽名——专业：<学科专业>

槽名——职称：（助教，讲师，副教授，教授）

缺省：讲师

槽名——外语：

侧面名——语种：（英，法，日，俄）

默认值：英

侧面名——水平：（优，良，中，差）

默认值：良





### 2. 框架的推理

#### 1) 面向检索的继承推理

这是一种以框架间层次关系的性质继承及利用缺省值为主的推理策略。它的意思是低层框架可以继承较高层框架的性质。当检索某槽的值时，而该槽为空（缺省值），可从该框架的父辈框架或其祖先框架中继承有关槽值、限制条件或附加过程。

#### 2) 面向过程的推理

框架表示法能把描述型知识与过程型知识的表示组合到同一数据结构中。因此，可利用槽中的附加过程（或子程序）实现控制。这个程序体放在另外的地方，以供多个框架共同使用。

3) 面向规则的推理这是在综合运用框架方法和产生式规则表示法的机制中使用的推理方式。框架与规则的连接有两种方式：将规则连入框架和将框架连入规则。

将规则连入框架，也就是在框架中包含规则，即用附加过程调用规则集合，来控制信息的存储、检索和推理。但事实上，应用框架中的附加过程执行所有的推理，将起副作用。这种缠结结构产生的后果是，不仅理解和维护是困难的，且效率也低。

将框架连入规则，这种方式将规则中的前提和结论表示为框架。在推理中，应用规则控制推理，而用框架组织智能数据库来维护推理所需要的知识。

组合规则和框架方法，可以建立一种知识表示与推理相结合的综合系统。区别哪种知识在框架中描述、哪种知识在规则中描述及规则与框架的连接方法是关键问题。



### 3. 框架表示法的特点

框架表示法的数据结构和问题求解过程与人类的思维和问题求解过程相似。框架结构表达能力强，层次结构丰富，提供了有效的组织知识的手段，只要对其中某些细节作进一步描述，就可以将其扩充为另外一些框架。框架表示法可以利用过去获得的知识对未来的情况进行预测，而实际上这种预测非常接近人的知识规律，因此可以通过框架来认识某一类事物，也可以通过一些实例来修正框架对某些事物的不完整描述（填充空的框架，修改默认值）。

但是框架表示法与语义网络表示法存在着相似的问题，没有明确的推理机制保证问题求解的可行性和推理过程的严密性；由于许多实际情况与原型存在较大的差异，可能适应性不够强；框架表示法系统中各个子框架的数据结构如果不一致会影响整个系统的清晰性，造成推理的困难。

总之，框架表示法基于认知科学理论来考察问题，应用人类的框架式知识模式去组织人类对真实事物的基本理解。因此，框架方法能捕获领域专家的思考方法，应用专家使用的专门技术充分表达对象和对象之间的关系，并实现概念和术语的简明定义。框架表示法是表示专业领域知识的理想方法。

## 第二章 知识表示

2.1 知识表示的内涵

2.2 谓词逻辑表示法

2.3 产生式规则表示法

2.4 语义网络表示

2.5 知识图谱表示

2.6 框架表示法

2.7 脚本表示法

习题



### 1. 脚本的结构

脚本是框架的一种特殊形式，它用一组槽描述某些事件发生的序列，就像一出剧中每个场次出现的顺序一样，故将这种表示方法称为脚本。不同的是，脚本所表达的不是一种完全通用的结构，它的各个槽和侧面已有固定的意义。

在一个脚本中应包含以下内容：

- 1) **开场条件**：描述事件发生的前提条件。
- 2) **角色**：所描述事件中可能出现的人物的槽。
- 3) **道具**：是一些用来描述事件中可能出现的有关物体的槽。
- 4) **场景**：描述发生事件的真实顺序。一个事件可有多个场景组成，而每个场景又可以是其他的脚本。
- 5) **结局**：在脚本中描述事件出现后所产生的结果，事件发生以后必须满足的条件。



### 1. 脚本的结构

例如：以一家“餐厅”脚本为例来说明各个脚本各个部分的组成。

1) **进入条件**：  
顾客饿了，需要进餐。  
顾客有足够的钱。

2) **角色**：顾客、服务员、厨师、老板。

3) **道具**：食品、桌子、菜单、钱。



### 1. 脚本的结构

#### 4) 场景分别如下:

场景1: 进入餐厅

顾客走进餐厅

顾客注视桌子

确定往哪儿走

朝确定的桌子走

在桌子旁坐下

场景2: 点菜

服务员给顾客菜单

顾客点菜

顾客把菜单还给服务员

顾客等待服务员送菜

场景3: 等待

服务员告诉厨师顾客所点的菜

厨师做菜

顾客等待

场景4: 上菜进餐

厨师把食品交给服务员

服务员走向顾客

服务员把食品交给顾客

顾客吃食品

场景5: 离开

服务拿来账单

顾客付费给服务员

顾客离开餐厅

#### 5) 结局:

顾客吃了饭, 不饿了

顾客花了钱

老板赚了钱

餐厅的食品少了

服务员的业绩增加一单



### 2. 脚本的推理

通过“餐厅”脚本的描述，我们可以看出脚本描述事件其实是一个**因果链**，**链头**是一组开场条件，只有当这些初始条件满足时，该脚本中的事件才能开始；**链尾**是一组结果，只有当这一组结果满足时，该脚本中的事件才能结束，以后的事件或事件序列才能发生。

在这个因果链中，一个事件和其他前后事件之间相互联系，前面的事件可使当前事件产生，当前事件又可以使后面的事件产生。



### 3. 脚本的特点

由于脚本是一种框架结构，所以在具体应用前，我们需要根据环境对脚本中槽值进行赋值，这个过程成为**脚本预先准备**。在一个脚本中，场次描述了一个特定的环境下将要发生的一系列、因果关系的事件，因此它能帮助预见未被直接观察到的事实，也可以对一组观测事实进行解释。

由于脚本结构的特殊性，脚本与框架理论相比脚本要**呆板得多**，**知识表示范围也很窄**。但是对于一些特定领域，尤其是表达预先构思好的特定知识，脚本表示非常有效。



## 第二章 知识表示

2.1 知识表示的内涵

2.2 谓词逻辑表示法

2.3 产生式规则表示法

2.4 语义网络表示

2.5 知识图谱表示

2.6 框架表示法

2.7 脚本表示法

习题

# 习题：

---

1. 怎么理解数据、信息和知识三者之间关系？
2. 对下列命题分别写出他们的语义网络。
  - A、刘教授在2017年4月6日给人工智能学院的学生讲《人工智能》课。
  - B、红队与黄队进行足球比赛，最后以1：0的比分结束。
  - C、植物都有叶和根。草和树都是植物。水草是草，且生长在水中。果树是树，且会结果。苹果树是果树的一种，会结苹果。
3. 按“雇佣框架”、“雇主框架”、“雇员框架”的形式写出一个框架系统的描述。
4. 以一场“学术报告”为例来进行脚本的知识表示描述。

# 习题：

---

A、刘教授在2017年4月6日给人工智能学院的学生讲《人工智能》课。

语义网络结构：

1.节点：

1. 刘教授
2. 2017年4月6日
3. 人工智能学院
4. 学生
5. 《人工智能》课

2.关系：

1. 刘教授 → **讲课** → 《人工智能》课
2. 讲课 → **时间** → 2017年4月6日
3. 讲课 → **地点** → 人工智能学院
4. 讲课 → **对象** → 学生