



中国科学院自动化研究所
INSTITUTE OF AUTOMATION
CHINESE ACADEMY OF SCIENCES

知识表示

张文生 孙正雅

中国科学院自动化研究所
中科院大学人工智能学院



讲课内容

- 一. 知识表示概述
- 二. 知识表示方法
- 三. 大数据时代知识表示



讲课内容

一. 知识表示概述

二. 知识表示方法

三. 大数据时代知识表示



知识表示的重要

- 人工智能研究中最基本的问题之一
 - 在知识处理中总要问到：“如何表示知识？”，“知识是用什么来表示的？”
 - 怎样使机器能懂，能对之进行处理，并能以一种人类能理解的方式将处理结果告诉人们
 - 在AI系统中，给出一个清晰简洁的描述是很困难的，研究报道，严格地说AI对知识表示的认真、系统的研究才刚刚开始

- 知识的定义

- **Feigenbaum**: 知识是经过消减、塑造、解释和转换的信息
- **Bernstein**: 知识是由特定领域的描述、关系和过程组成的
- **Hayes-roth**: 知识是事实、信念和启发式规则

知识库的观点:

知识是某领域中所涉及的各有关方面的一种符号表示。

- 知识的种类

- 事实性知识：采用直接表示的形式，如：凡是猴子都有尾巴
- 过程性知识：描述做某件事的过程，如：电视维修法
- 行为性知识：不直接给出事实本身，只给出在某方面的行为
如：微分方程、（事物的内涵）



- 实例性知识：只给出一些实例，知识藏在实例中
- 类比性知识：即不给出外延，也不给出内涵，只给出它与其它事物的某些相似之处，如：比喻、谜语
- 元知识：有关知识的知识，最重要的元知识是如何使用知识
的知识，如何从知识库中找到想要的知识

—

人工智能问题的求解是以知识表示为基础的，如何将已获得的有关知识以计算机内部代码形式加以合理地描述、存储、有效地利用便是表示应解决的问题

- 研究内容

- 表示观的研究

- 识论、本体论、知识工程

- 表示方法的研究

- 直接法、代替法(局部、分布, ...)

讲课内容

- 一. 知识表示概述
- 二. 知识表示方法
- 三. 大数据时代知识表示

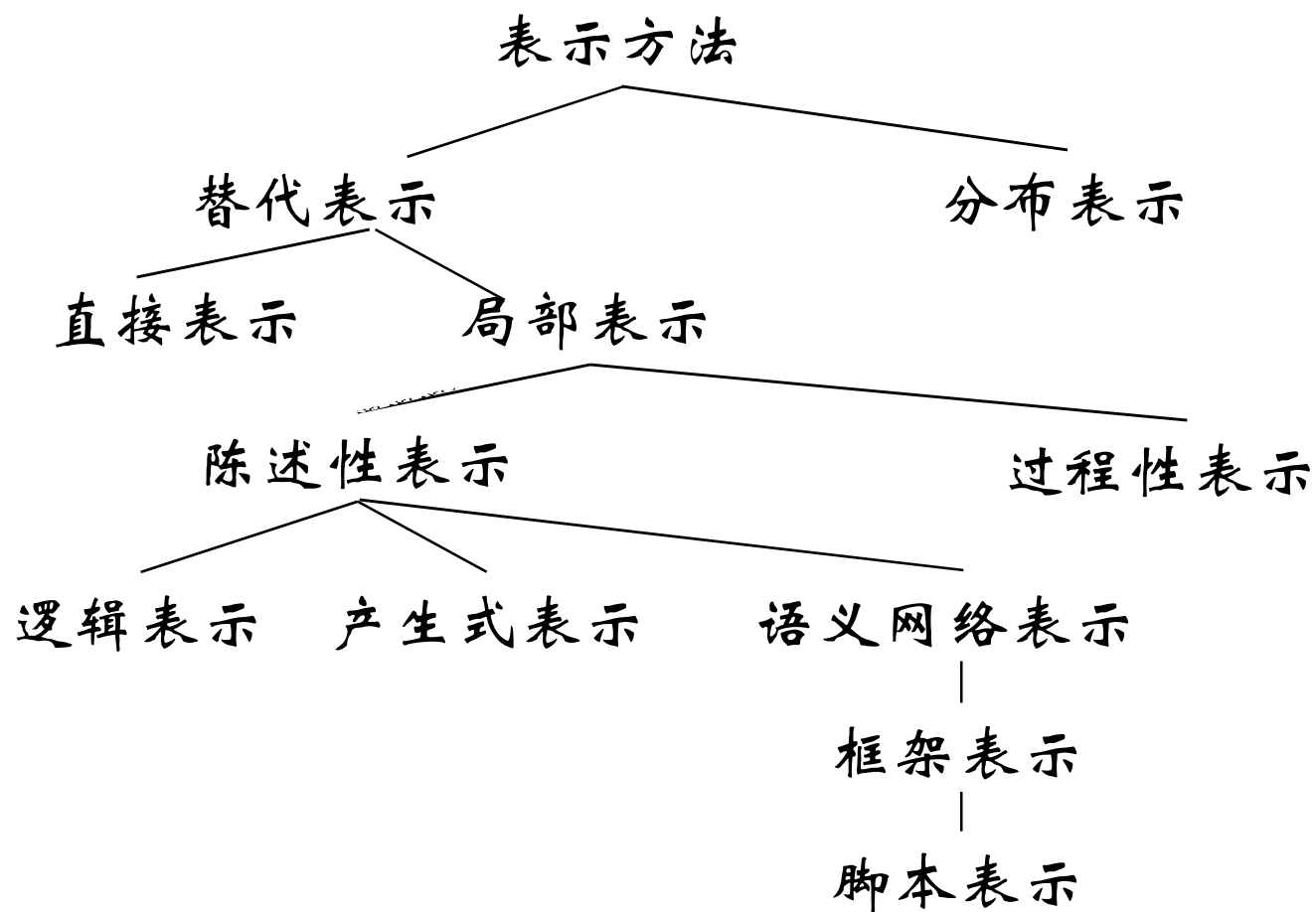


知识表示方法

- 概述
 - 直接表示
 - 逻辑表示
 - 产生式规则表示法
 - 语义网络表示法
- 框架表示法
 - 脚本方法
 - 过程表示
 - 混合型知识表示方法
 - 面向对象的表示方法



- 表示方法可以分成
 - 替代表示法
 - 局部表示类：最充分也是正统AI最经常使用的
 - 分布表示法：对局部表示法在智能行为表述尚不够充分而作的补充
 - 直接表示法：
 - 正在引起越来越多AI研究者的注意，直接表示需要借助局部或部分表示形式





表示方法

- ▶ 概述
- ▶ 直接表示
- 逻辑表示
- 产生式规则表示法
- 语义网络表示法
- 框架表示法
- 脚本方法
- 过程表示
- 混合型知识表示方法
- 面向对象的表示方法



直接表示

- 1963年由Gelernter提出的，用于基于传统欧氏几何证明的几何定理证明器
- 它的输入是对前提和目标的陈述以及图示（图示是用一系列坐标来表示的）
- 在证明过程中，证明器把图示作为启发式信息，排除在图示中不正确的子目标。从而大大地减少了搜索空间



- 长期以来直接表示没有得到长足发展，原因如下：
 - 计算机对直接表示的信息难以处理
 - 直接表示难以表示定量信息（语言设计失败）
 - 直接表示不能描述自然世界的全部信息
- 最近直接表示有所发展，现在认识到：可以用其它媒体表示的方法去补充直接表示的不足，被发展成多媒体
- 近几年AI对自主智能系统研究（完全机器做人不干预）的失望，导致对建立人机一体智能系统的尝试，这样系统所需环境的要求是直接表示兴起的原因之一



表示方法

- ▶ 概述
- ▶ 直接表示
- ▶ 逻辑表示
- 产生式规则表示法
- 语义网络表示法
- 框架表示法
- 脚本方法
- 过程表示
- 混合型知识表示方法
- 面向对象的表示方法



逻辑表示法

- 一阶谓词逻辑是谓词逻辑中最直观的一种逻辑，以谓词形式来表示动作的主题、客体，客体可以多个

如：张三与李四打网球（Zhang and Li play tennis）

可写成：**play (Zhang, Li, tennis)**，这里谓词是**play**

动词主体是**Zhang**和 **Li**，而客体是**tennis**

- 规范表达式：

P (x₁, x₂, x₃, ...)， 这里**P**是谓词，**x_i**是主体与客体。

- 谓词比命题更加细致地刻画知识：

- 表达能力强

- 如：北京是个城市， **City(x)**

把城市这个概念分割出来。把“城市”与“北京”两个概念连接在一起，而且说明“北京”是“城市”的子概念。（有层）

- 谓词可以代表变化的情况

- 如：**City(北京)**,真。 **City(煤球)**, 假

- 在不同的知识之间建立联系

—

– 在不同的知识之间建立联系

- 如: $\text{Human}(x) \rightarrow \text{Lawed}(x)$, 人人都受法律管制

$\text{Commit}(x) \rightarrow \text{Punished}(x)$, x 不一定是人也可以是动物

$\{[\text{Human}(x) \rightarrow \text{Lawed}(x)] \rightarrow [\text{commit}(x) \rightarrow \text{Punished}(x)]\}$

意为如果由于某个 x 是人而受法律管制, 则这个人犯了罪
就一定要受到惩罚



- 用逻辑（谓词）表示知识实质上是把人类关于世界的认识变成一个包含个体、函数和谓词的概念化形式
- 基本步骤：
 - 给出有关世界的个体、函数和谓词
 - 构造一阶谓词公式（集）
 - 对公式（集）给出解释，使该解释是相应公式（集）的一个模型



逻辑表示

例：一个房间里，有一机器人**Robot**，一个积木块**Box**，两个桌子**A**和**B**，怎样用逻辑法描述从初始状态到目标状态的机器人操作过程？

- 先引入谓词：

Table(A)

表示**A**是桌子

EmptyHanded(Robot)

机器人**Robot**双手空空

At(Robot,A)

表示机器人**Robot**在**A**旁

Holds(Robot,Box)

机器人**Robot**拿着**Box**

On(Box,A)

积木块**Box**在**A**上

- 设定初始状态：

EmptyHanded(Robot)

On(Box,A)

Table(A)

Table(B)

- 目标状态是：

EmptyHanded(Robot)

On(Box,B)

Table(A)

Table(B)



- 机器人的每个操作的结果所引起的状态变化，可用对原状态的增添表和删除表来表示。如机器人有初始状态是把**Box**从**A**桌移到**B**桌上，然后仍回到**Alcove**凹室，这时同初始状态相比有：

增添表 **On (Box, B)**

删除表 **On (Box, A)**

- 机器人从初始状态，走近**A**桌，拿起**Box**，同初始状态相比有：

增添表 **At (Robot, A) Holds (Robot, Box)**

删除表 **At (Robot, Alcove) EmptyHanded(Robot)**

On (Box, A)



- 机器人的每一操作还需要先决条件，如机器人拿起**A**桌上的**Box**这一操作，先决条件：

On (Box, A) (**Box**在**A**上)

At(Robot, A) (机器人在**A**旁边)

EmptyHanded (robot) (机器人手空空)



- 先决条件成立与否的验证可以使用归结法，将初始状态视作已知条件，而将要验证的先决条件作为结论，便可使用归结法了。

- 归结过程如下：

1) $At(Robot, A)$

2) $EmptyHanded(Robot)$

3) $On(Box, A)$

4) $Table(A)$

5) $Table(B)$

6) $\sim On(Box, A) \vee \sim At(Robot, A) \vee \sim EmptyHanded(Robot)$ (否定)

7) $\sim At(Robot, A) \vee \sim EmptyHanded(Robot)$ 3, 6

8) $\sim EmptyHanded(Robot)$ 1, 7

9) NULL 2, 8

- 于是验证了先决条件的成立。



表示方法

- ▶ 概述
- ▶ 直接表示
- ▶ 逻辑表示
- ▶ 产生式规则表示法
- 语义网络表示法
- 框架表示法
- 脚本方法
- 过程表示
- 混合型知识表示方法
- 面向对象的表示方法



- 表示形式

事实的表示：可看成是断言一个语言变量的值或是多个语言变量间的关系的陈述句，语言变量的值或语言变量间的关系可以是一个词，不一定是数字。

例1：香蕉是黄色的。语言变量——香蕉，值——黄色的

例2：小李喜欢小莉。语言变量——小李、小莉，

关系值——喜欢

一般用三元组（对象，属性，值）或

（关系，对象1，对象2）

— 例：（Li, Age, 25），（Friend, Li, Chang）



- 产生式系统的基本特征：

- 一组规则，即产生式本身，每个规则分左边右边

- 如：天上下雨 → 地上湿

- 中国的首都 → 北京

一般左边表示情况即什么条件，发生时产生式被调用，

通常用匹配方法和式情况，匹配成功时，执行右边规

定的动作

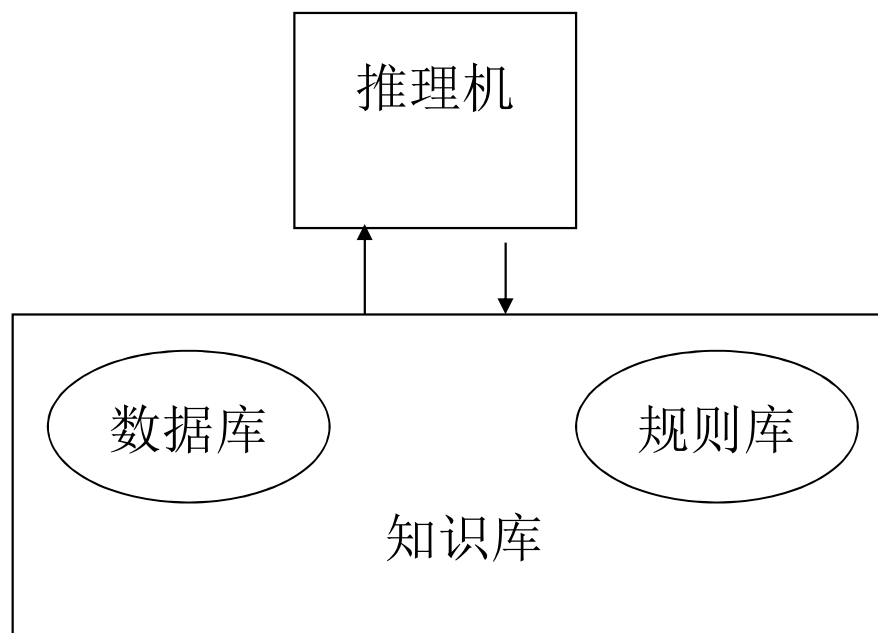
- 数据库

存放的数据是构成产生式的基本元素，又是产生式作用的对象。这里的数据是广义的常量、变量、多元组谓词、表、图像等。往往事实或断言——知识元

- 一个解释程序

从匹配成功的规则（可能不止一个）中选出一个加以执行

- 产生式系统基本结构



产生式系统结构图

- 正向推理方法：

从已知事实出发，逐步推导出最后结论。

其推理过程大致是：

- 用工作存储器中的事实与产生式规则的前提条件进行匹配。
- 按冲突消解策略从匹配的规则中选择一条规则
- 执行选中规则的动作（依次，修改工作存储器
- 用更新后的工作存储器，重复上述工作，直到得出结论或工作存储器不再发生变化为止



- 反向推理方法：

首先提出假设，然后验证这些假设的真假性，找到假设成立的所有证据或事实。

其推理过程大致是：

- 看假设是否存在于工作存储器中，若在，则假设成立，推理结束
- 找出结论与此假设匹配的规则
- 按冲突消解策略从匹配的规则实例中选择一条规则
- 将选中的规则的前提条件作为新的假设，重复上述工作，直到假设的真假性被验证或不存在激活的规则

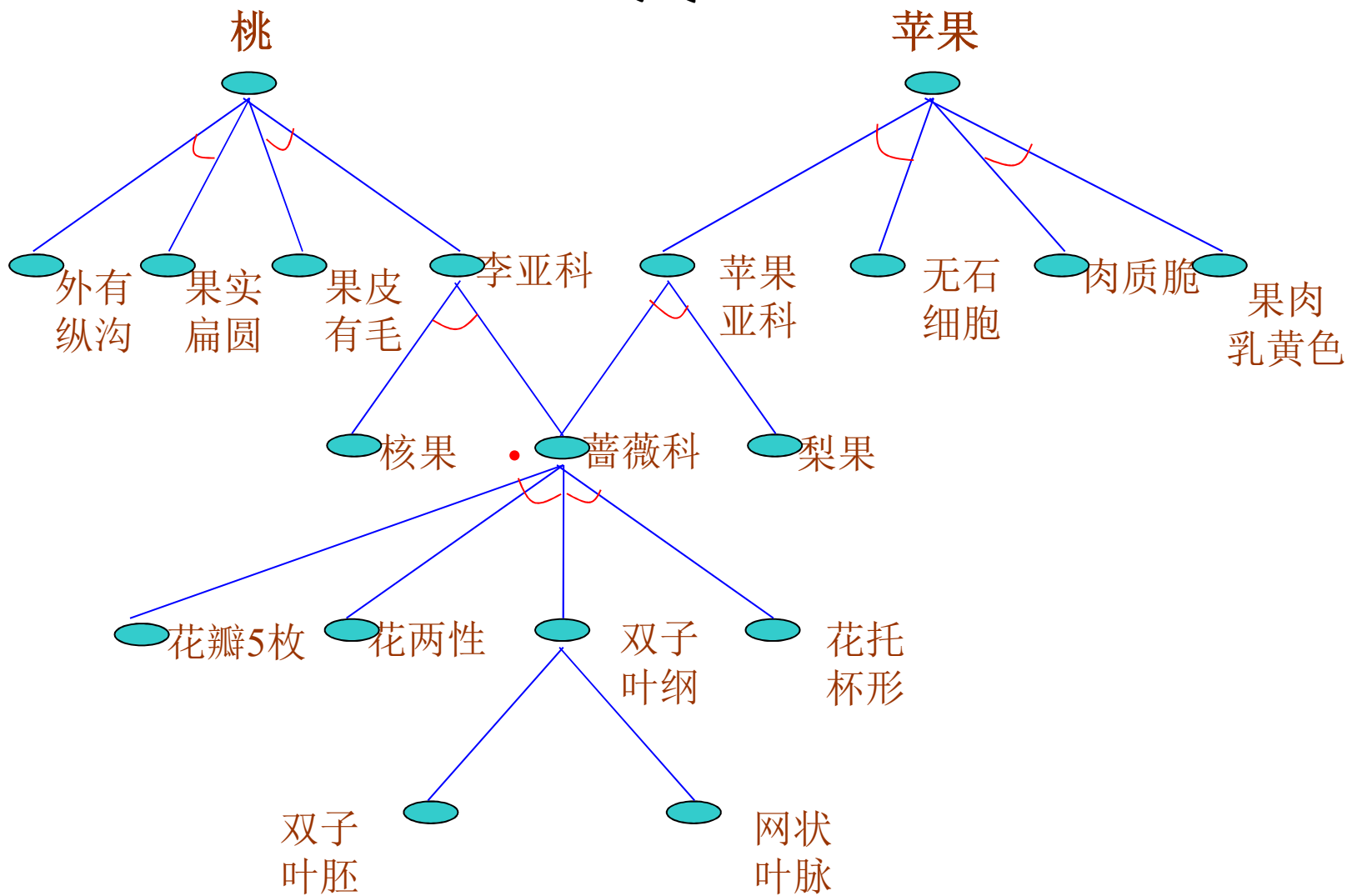
- 双向推理方法：

即自顶向下、又自底向上作双向推理，直至某个中间界面上两方向结果相符便成功结束

该方法较正向或反向推理所形成的推理网络小，从而推理效果更高



与或树





- 推理方法的选择

推理方法的选择取决于推理的目标和搜索空间的形状

- 如果目标是从一组给定事实出发，找出所有可能的结论，那么，通常使用正向推理
- 如果目标是证实或否定某一特定结论，那么，通常使用反向推理，否则，从一组初始事实出发盲目地正向推理，可能得出许多和所要证实的结论无关的结论



- 缺点：

- 知识库维护难
- 效率低，为了模块一致性
- 理解难，由于规则一致性彼此之间不能调用

- 应用实例：

- 用于化工工业测定分子结构的DENDRAL
- 用于诊断脑膜炎和血液病毒感染的MYCIN
- 估计矿藏的PROSPECTOR



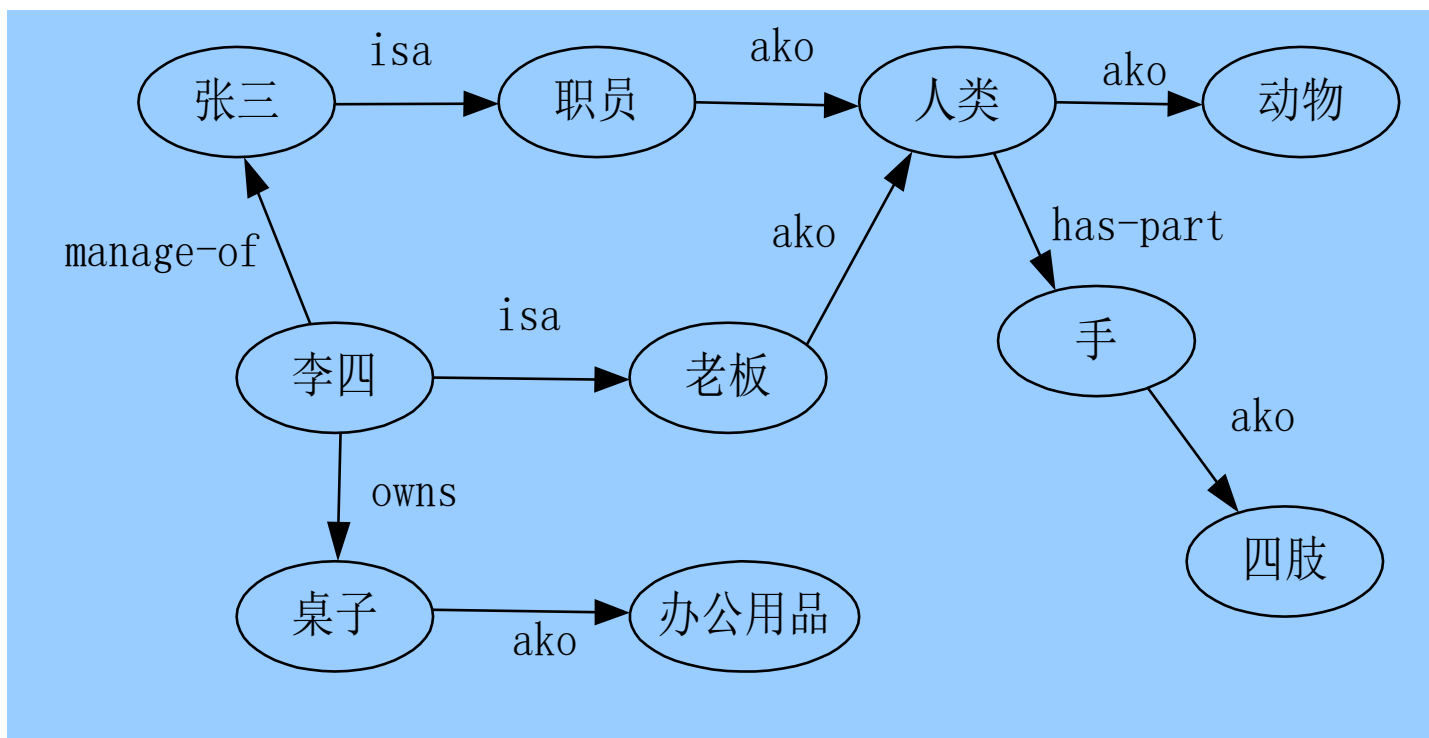
表示方法

- ▶ 概述
- ▶ 直接表示
- ▶ 逻辑表示
- ▶ 产生式规则表示法
- ▶ 语义网络表示法
- 框架表示法
- 脚本方法
- 过程表示
- 混合型知识表示方法
- 面向对象的表示方法



- 表示形式

每一个要表达的事实用一个“结点”表示，
而事实之间的关系用“弧线”表示，有向图
表示的三元组，（结点1， 弧， 结点2）连接
而成





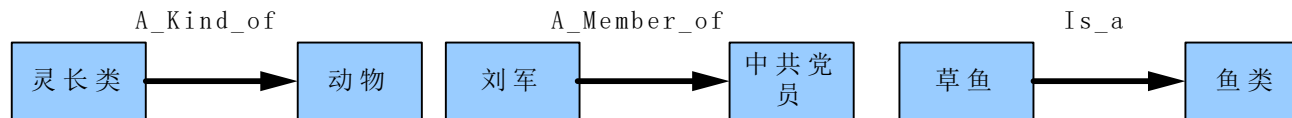
类属关系

- 类属关系指具有共同属性的不同事物间的分类关系、成员关系或实例关系
- 注：它体现的是“具体与抽象”、“个体与集体”的概念。类属关系的一个最主要特征是属性的继承性，处在具体层的结点可以继承抽象层结点的所有属性
- 常用的属性有：

A-Kind-of:表示一个事物是另一个事物的一种类型

A-Member-of:表示一个事物是另一个事物的成员

Is-a:表示一个事物是另一个事物的实例



类属关系实例

- 注：在类属关系中，具体层的结点除了具有抽象层结点的所有属性外，还可以增加一些自己的个性。

包含关系

包含关系也称为聚类关系，是指具有组织或结构特征的“部分与整体”之间的关系

注：它和类属关系的最主要的区别就是包含关系一般不具备属性的继承性

常用的包含关系的有：

Part_of:表示一个事物是另一个事物的一部分



包含关系实例

属性关系

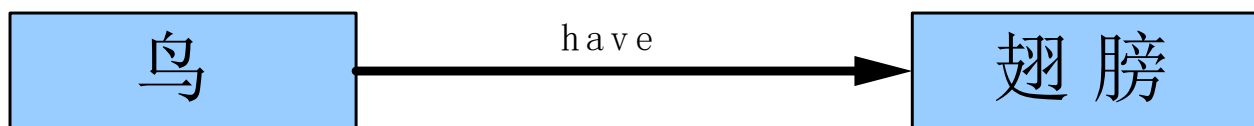
属性关系是指事物和其属性之间的关系。

常用的属性的关系有：

Have：表示一个结点具有另一个结点所描述的属性

Can：表示一个结点能做另一个结点的事情

例：鸟有翅膀



属性关系实例

位置关系

位置关系是指不同事物在位置方面的关系。

常用的位置关系：

| | |
|------------------|----------|
| Located-on: | 一物在另一物之上 |
| Located-at: | 一物在何位置 |
| Located-under: | 一物在另一物之下 |
| Located-inside: | 一物在另一物之中 |
| Located-outside: | 一物在另一物之外 |



相近关系

相近关系是指不同事物在形状、内容等方面相似和接近。

常用的相近关系：

Similar-to:

相似

Near-to:

接近



时间关系

是指不同事件在其发生时间方面的先后关系。

常用的时间关系有：

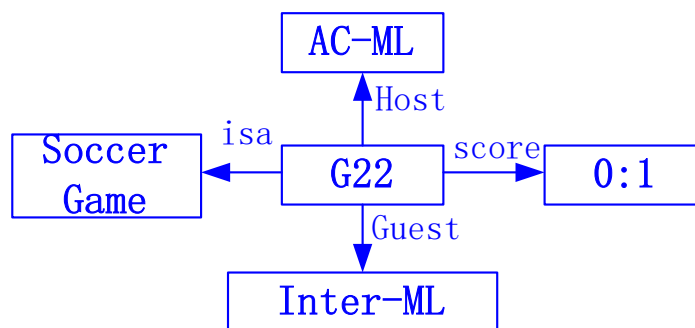
Before：表示一个事件在一个事件之前发生

After：表示一个事件在一个事件之后发生。

例如：香港回归之后，澳门也会回归了。

多元逻辑关系

例如AC米兰队和国际米兰队在一场足球比赛中的成绩为0:1，逻辑表示法为 SCORE(AC-MILAN, INTER-MILAN, 0:1), 可以通过加入附加结点的办法将其改成语义网络表示法，其根本方法是将多元关系表示成二元关系的组合或合取。本例通过加入附加结点G22。

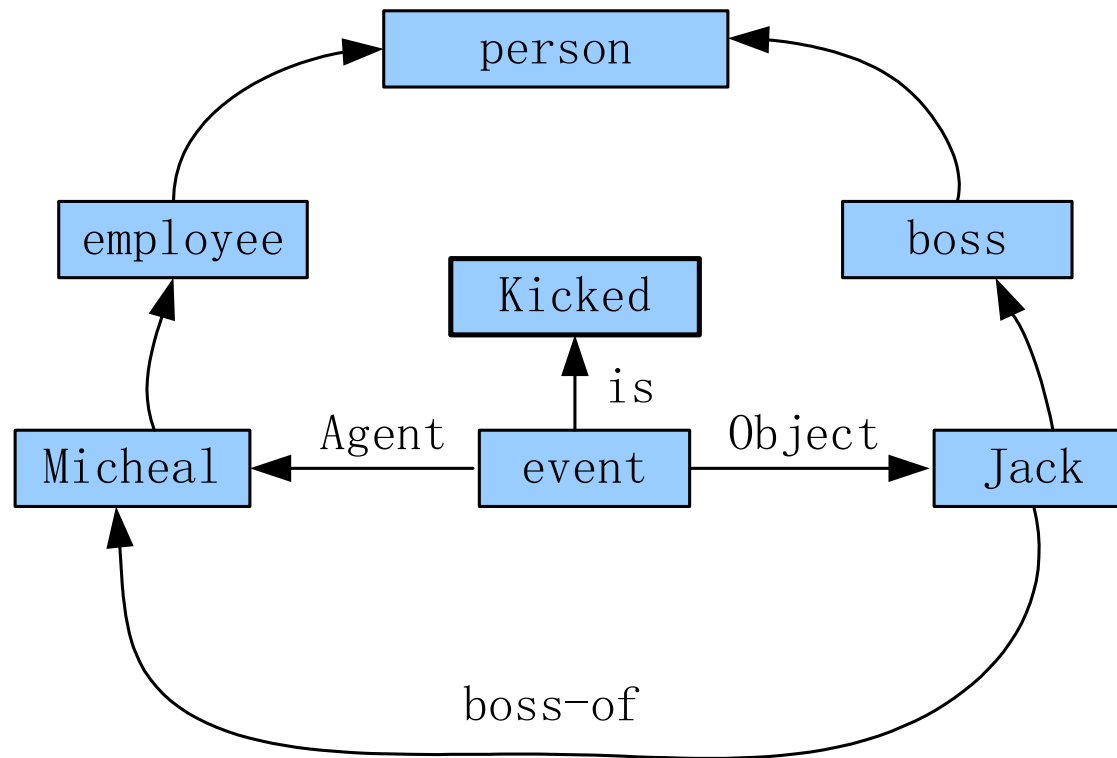


多元逻辑关系语义网络实例

- 可以看出，原来的多元关系都变成了G22结点属性。

Micheal is an employee and Jack is his boss. Someday Micheal kicked his boss.

语义描述



推理方法

- 网络匹配：结构上的匹配，包括结点和弧的匹配
- 继承推理：利用如：成员联系、特征联系、相互作用联系、集合联系、合成联系、因果联系、活动方式联系、活动目标联系、蕴含联系等具有继承性质的语义联系建立一些并不一定显示存在于网络知识库中的网络结构
- 语义网络上的推理：网络上的搜索过程，正向、逆向、双向



- 继承的一般规则:

- IF $X (AKO) Y$ and $Y (AKO) Z$ then $X (AKO) Z$
- IF $X (ISA) Y$ and $Y (AKO) Z$ then $X (ISA) Z$
- IF $X (AKO) Y$ and $Y (属性) Z$ then $X (属性) Z$
- IF $X (ISA) Y$ and $Y (属性) Z$ then $X (属性) Z$
- IF $X (属性) Y$ and $Y (AKO) Z$ then $X (属性) Z$
- IF $X (属性) Y$ and $Y (ISA) Z$ then $X (属性) Z$

- 推理特点

- 不十分明了，有继承规则
- 可以用关系如：成员联系、特征联系、相互作用联系、集合联系、合成联系、因果联系、活动方式联系、活动目标联系、蕴含联系等
- 还可以将语义网络引入逻辑含义，表示 \wedge ， \vee ， \sim 关系。用归结推理法



表示方法

- ▶ 概述
- ▶ 直接表示
- ▶ 逻辑表示
- ▶ 产生式规则表示法
- ▶ 语义网络表示法
- ▶ 框架表示法
- 脚本方法
- 过程表示
- 混合型知识表示方法
- 面向对象的表示方法



- 定义

- 框架是由若干个结点和关系（统称为槽）构成的网络，是语义网络的一般化形式的一种结构，同语义网络没有本质的区别。如将语音网络结点间弧上的标注也放到槽内就成了框架表示形式

- 表示形式：

- 由框架名、槽名、侧面、值组成

- 推理方法：

- 没有固定的推理机理。但和语义网络一样遵循匹配和继承的原理

• 性质

- 对事物进行描述，对其中某些细节做进一步描述，可将其扩充为另外一些框架。如：汽车载货或人
- 可以通过它对一些从感官中没有直接得到的信息进行预测，对于人来说这种功能是很强的。如：一想到桌子就可以想到它腿的形状与位置
- 可以在它基础上进行判断推理
- 可通过它来认识某一类事物
- 可以通过一系列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)



【 框架 名 】

| | | | | |
|-------|------------|--------------|--------------|----------------|
| 槽 名 A | 侧面 名 A_1 | 值 A_{11} ， | 值 A_{12} ， | 值 A_{13} ... |
| | 侧面 名 A_2 | 值 A_{21} ， | 值 A_{22} ， | 值 A_{23} ... |
| 槽 名 B | 侧面 名 B_1 | 值 B_{11} ， | 值 B_{12} ， | 值 B_{13} ... |
| | 侧面 名 B_2 | 值 B_{21} ， | 值 B_{22} ， | 值 B_{23} ... |
| 槽 名 C | 侧面 名 C_1 | 值 C_{11} ， | 值 C_{12} ， | 值 C_{13} ... |
| | 侧面 名 C_2 | 值 C_{21} ， | 值 C_{22} ， | 值 C_{23} ... |

约束条件：

约束条件 1
约束条件 2
约束条件 3



简单框架的例子:

Micheal

Gender: man

Profession: singer

Height: 185cm

Weight: 79kg

Age: 27

例如：要确定一个人的性别，已匹配的知识库中的框架为

【槽名

Gender **NIL**

If needed **ASK**

If added **CHECK】**

启动过程如下：

- 1) 如果没有默认值，**if needed**条件满足
- 2) 启动**ASK**，向用户查询并等待输入
- 3) 若有输入（**if added**），执行**CHECK**，检查输入的合法性
若有默认值而无输入，则不执行**CHECK**



- 框架之间的关系

框架也分为类框架和实例框架。通过引入类-超类（AKO）及实例-类（ISA）关系来表示框架之间的包含关系和属于关系，框架理论将知识看成相互关系的成块组织

- 推理方法：

- 匹配：和语义网络一样遵循匹配原理
- 槽计算：继承（属性值、属性、限制），附加过程，即附加在数据结构上，启动时计算槽值

框架名：〈学校〉

类 属：〈教育机构〉

类 型：

范围：（大学，中学，小学）

位 置：（省（直辖市），市）

面 积：单位（平方米）

教工人数：

学生人数：

框架名：〈大学〉

类 属：〈学校〉

类 型：

范围：（综合性大学，专科性大学）

专 业：默认值：综合

学 院 数：

教 学 楼：

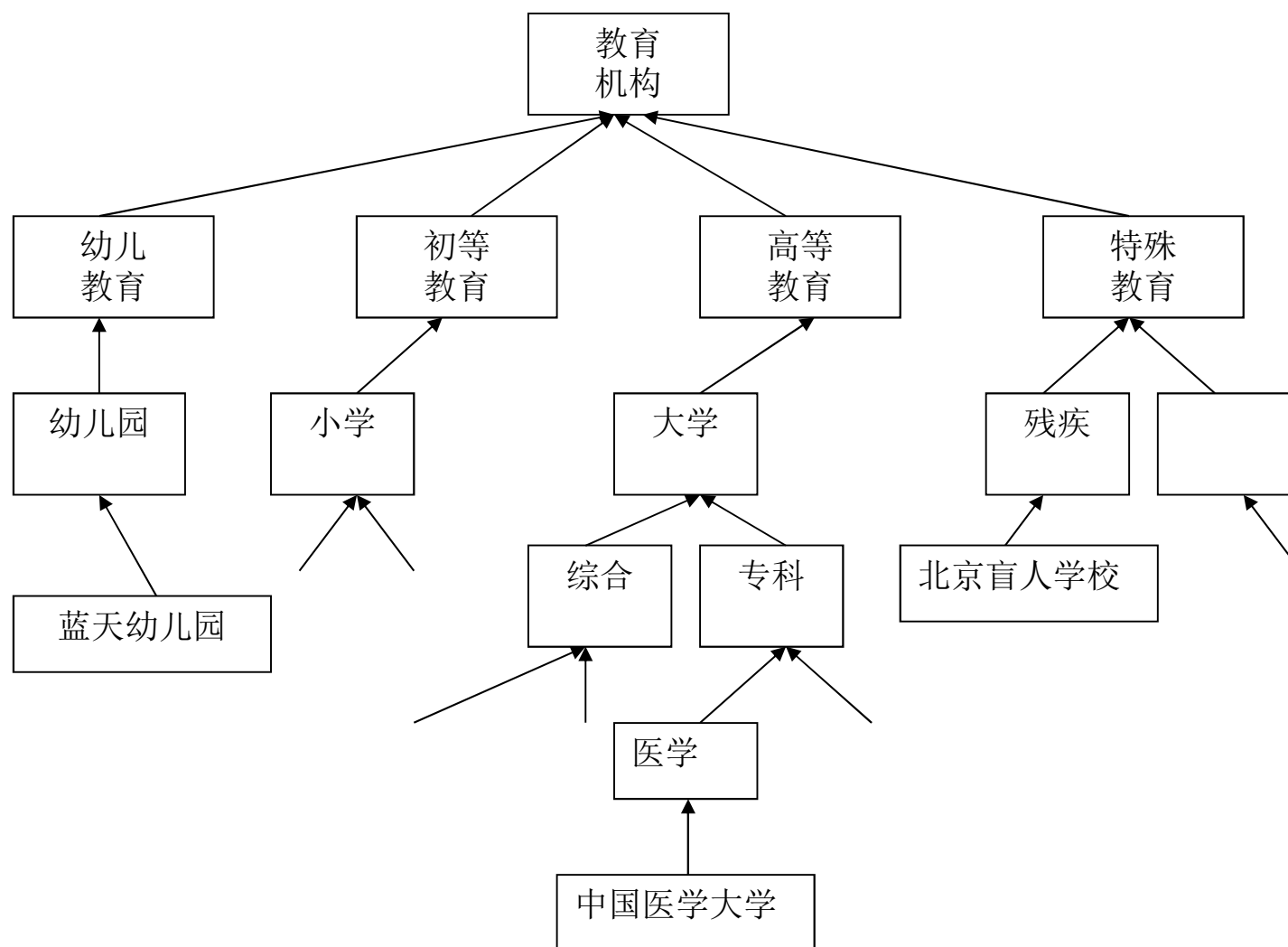
教工人数：

职工人数：

学生人数：

位 置：（省（直辖市），市）

面 积：单位（平方米）



框架系统结构



表示方法

- ▶ 概述
- ▶ 直接表示
- ▶ 逻辑表示
- ▶ 产生式规则表示法
- ▶ 语义网络表示法
- ▶ 框架表示法
- ▶ 脚本方法
- 过程表示
- 混合型知识表示方法
- 面向对象的表示方法



脚本表示法

- 脚本方式是采用一个专用的框架，用来表示特定领域的知识
- 脚本通过一些元语作为槽名来表代要表示的对象的基本行为
- 有些象电影剧本



开场条件

1. 病人有病。
2. 病人的病需要找医生诊治。
3. 病人有钱。
4. 病人能够去医院。

角色

病人、医生、护士。

道具

医院、挂号室、椅子、
桌子、药方、药房、
钱、药。

场景

场景1 进入医院

- (1) 人走进医院
- (2) 病人挂号
- (3) 病人在椅子上坐下等待看

病

场景2 看病

- (1) 病人进入医生的办公室
- (2) 病人向医生所说病状
- (3) 医生向病人解释病情
- (4) 医生给病人开药方

场景3 交费

- (1) 病人到交费处
- (2) 病人递交药方
- (3) 病人交钱
- (4) 病人取回药方及收据

场景4 取药

- (1) 病人到药房
- (2) 病人递交药方
- (3) 病人取药

场景5 离开

- (1) 病人离开医院

结果

1. 病人看病了，明白了自己的病是怎么回事。
2. 病人花了钱，买了药。
3. 医生付出了劳动。
4. 医院的药品少了。



表示方法

- ▶ 概述
- ▶ 直接表示
- ▶ 逻辑表示
- ▶ 产生式规则表示法
- ▶ 语义网络表示法
- ▶ 框架表示法
- ▶ 脚本方法
- ▶ 过程表示
- 混合型知识表示方法
- 面向对象的表示方法



过程表示法

- 前面的几种知识表示方法均是知识和事实的一种静止的表示方法，称这类知识表示方式为**陈述式表达**。强调的是事物所涉及的对象是什么，是对事物有关知识的静态描述，是知识的一种**显式、说明性**知识表达形式
- 说明性表示知识给出事物本身的属性及事物之间的相互关系。对问题的解答就隐含在这些知识之中。而**过程性知识**则给出解决一个问题的具体过程

- 说明性知识和过程性知识

- 说明性知识比较简要、清晰、可靠、便于修改，效率低
- 过程性知识比较直截了当，效率高，由于详细地给出了解决过程，使这种知识表示显得复杂、不直观、容易出错、不便于修改
- 说明性表示和过程性表示实际上没有绝对的分界线
- 任何说明性知识如果要被实际使用，必须有一个相应的过程去解释执行它。对于一个以使用说明性表示为主的系统来说，这种过程往往是隐含在系统之中，而不是面向用户

- 过程式表示定义：

- 过程式表示就是将有关某一问题领域的知识连同如何使用这些知识的方法均隐式地表达为一个求解过程
- 它所给出的是事物的一些客观规律，表达的是如何求解问题，知识的描述形式就是程序。所有信息均隐含在程序中——效率高、没有固定形式
- 如何描述知识完全取决于具体的问题

✦ 实际上的系统都是陈述与过程观点的结合，陈述之中多少包含了过程方法。



表示方法

- ▶ 概述
- ▶ 直接表示
- ▶ 逻辑表示
- ▶ 产生式规则表示法
- ▶ 语义网络表示法
- ▶ 框架表示法
- ▶ 脚本方法
- ▶ 过程表示
- ▶ 混合型知识表示方法
- 面向对象的表示方法



混合型知识表示法

- 知识表示虽各有特点，适用的领域也不同
 - 谓词逻辑方法只适用于确定性、陈述性、静态性知识，而对动态的、变化性、模糊性知识则很难表示。
 - 产生式规则方法推理方法太单一，如果前提条件太多，或规则条数太多，则推理的速度将慢得惊人。
 - 语义网络方法表达的知识面比较窄。
 - 框架方法表示的知识横向关系不太明确。（纵向从属继承关系很明确）
- 对于复杂的、深层次的知识，就很难用一种知识表示来解决问题



- 根据~~需要~~表示的知识的特征来决定用二、三种方式联合表示
 - 逻辑与框架：框架里的槽值可以对应与谓词项
 - 语义网络与框架：结点对应~~与~~框架，结点的参数就是框架的槽值
 - 产生式与框架：框架的槽值对应于一条产生式规则
 - 逻辑、产生式和过程式：产生式两端以谓词形式出现“活动”是个过程
 - 与神经网络结合

• 框架与产生式

- 在产生式系统中，随着产生式规则数量的增加，系统设计者难以理解规则之间的相互作用。原因是每条规则的自含性使得知识表示的粒度过于细致。因此，需要对规则的适当划分，将其组织易于管理的功能模块。框架系统具有组织成块知识的良好特性。两者的有机结合，有利于系统的开发、调试和管理
- 框架的表示机制可以用作产生式语言和推理机制设计的一个重要构件
- 框架可以直接用于表示规则（每个规则作为一个框架，一组规则组成一类）



表示方法

- ▶ 概述
- ▶ 直接表示
- ▶ 逻辑表示
- ▶ 产生式规则表示法
- ▶ 语义网络表示法
- ▶ 框架表示法
- ▶ 脚本方法
- ▶ 过程表示
- ▶ 混合型知识表示方法
- ▶ 面向对象的表示方法



面向对象的知识表示法

- 面向对象表示法中的对象指物体，消息指物体间的联系，通过发送消息使对象间相互作用来求得所需的结果
- 任何事物都是对象，对象按照“类”、“子类”进行分类
- 特点：
 - 有属性继承、特征描述结构化等优点

Corpus-Based Knowledge Representation

- Key Advantage:
 - Avoid the laborious(艰苦的; 费劲的) process of building a (often brittle 易碎的, 脆弱的) knowledge base.
- “We emphasize the corpus-based representation is not a replacement for traditional knowledge representation. There are many tasks in which very finely tuned reasoning is required, and such reasoning can only be done with a very well designed knowledge base (e.g., medical diagnosis, monitoring spacecraft, and making sense of tax law).”



讲课内容

一. 知识表示概述

二. 知识表示方法

三. 大数据时代知识表示

知识图谱

- 知识图谱

- 也称科学知识图谱，它通过将应用数学、图形学、信息可视化技术、信息科学等学科的理论与方法与计量学引文分析、共现分析等方法结合
- 并利用可视化的图谱形象地展示学科的核心结构、发展历史、前沿领域以及整体知识架构达到多学科融合目的的现代理论。
- 为学科研究提供切实的、有价值的参考。

- Google知识图谱

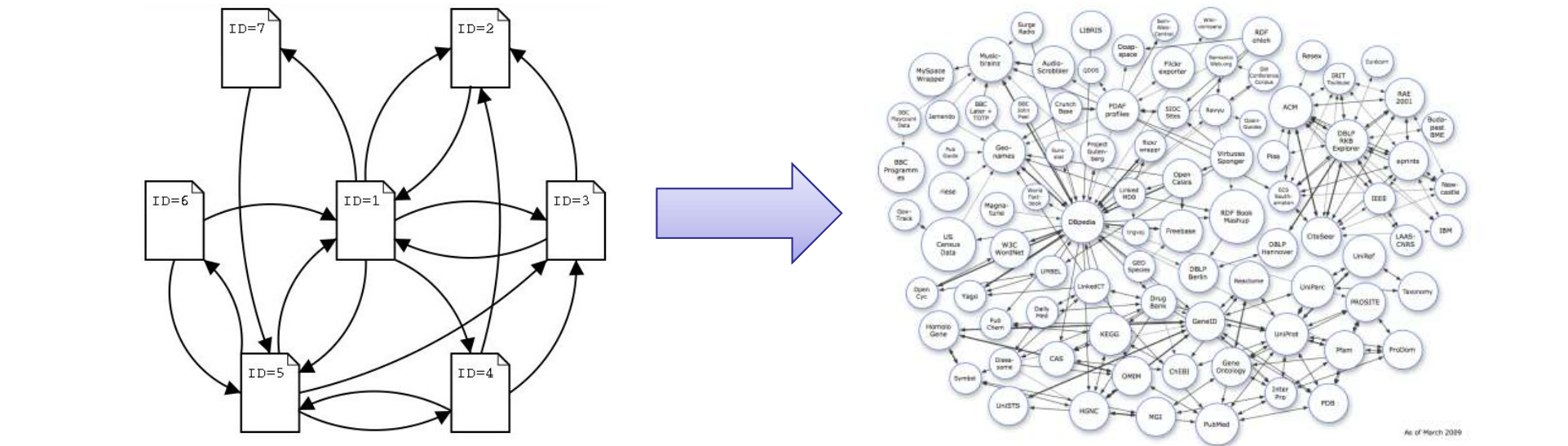
- 实体及其之间的关系图；
- 规模：5亿个对象，35亿个事实和关系；



- 知识图谱的本质：知识库、语义网络；
 - 知识库：知识的数据库；
 - 例如：**Freebase**是一个知识库；
- 知识图谱的形式：**RDF**、图数据库；
 - **RDF**提供了资源的通用描述方式；
 - 图数据库是指以图作为数据结构存储和查询数据的数据库；
 - 例如：**GraphDB**、**AllegroGraph**等；

Semantic Web

- **Tim Berners-Lee 1998年提出语义网的概念**
 - 通过给全球信息网上的文档（如：标准通用标记语言下的一个应用**HTML**）添加能够被计算器所理解的语义“元数据”（**Metadata**），从而使整个互联网成为一个通用的信息交换媒介

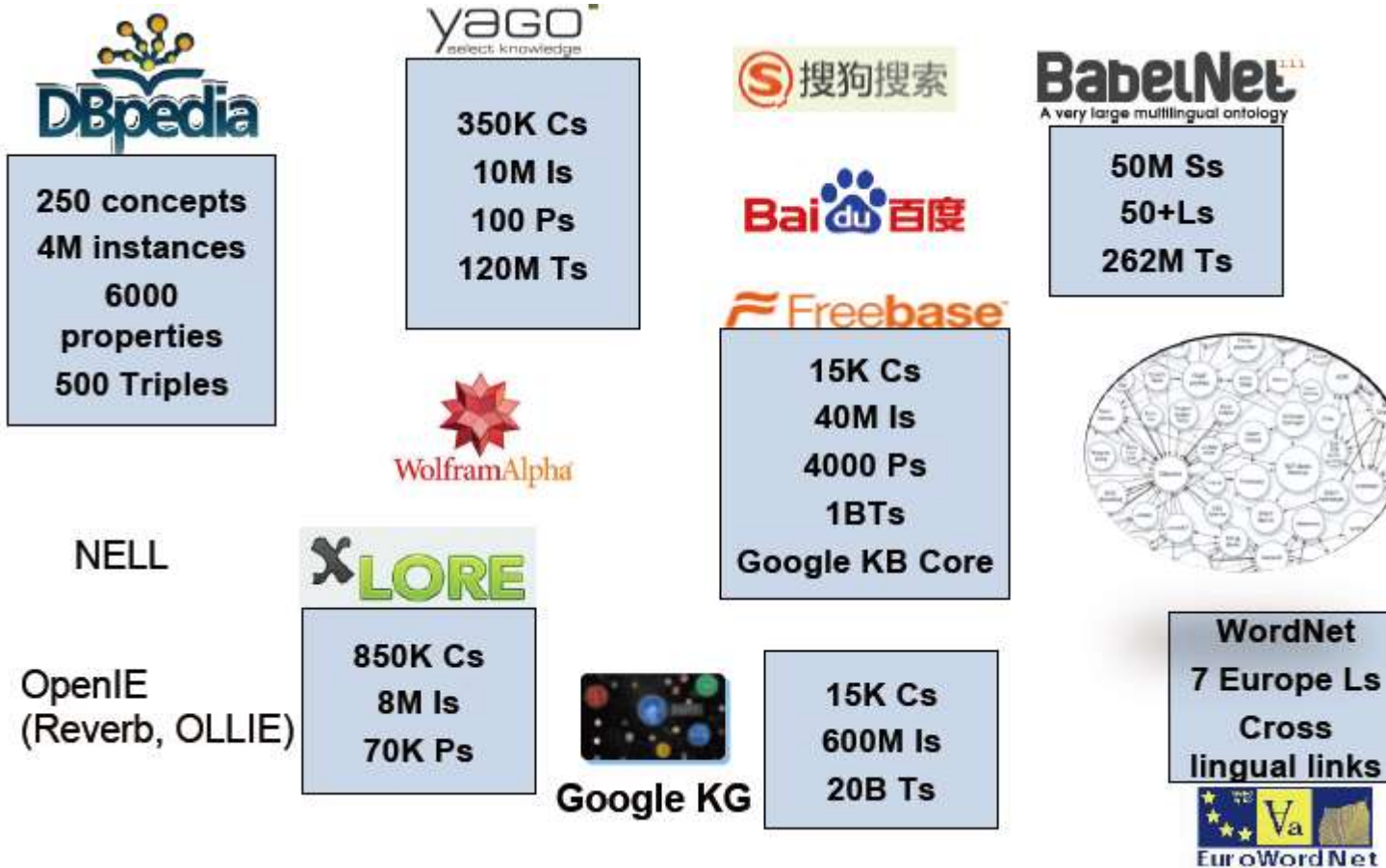


Page/Document web

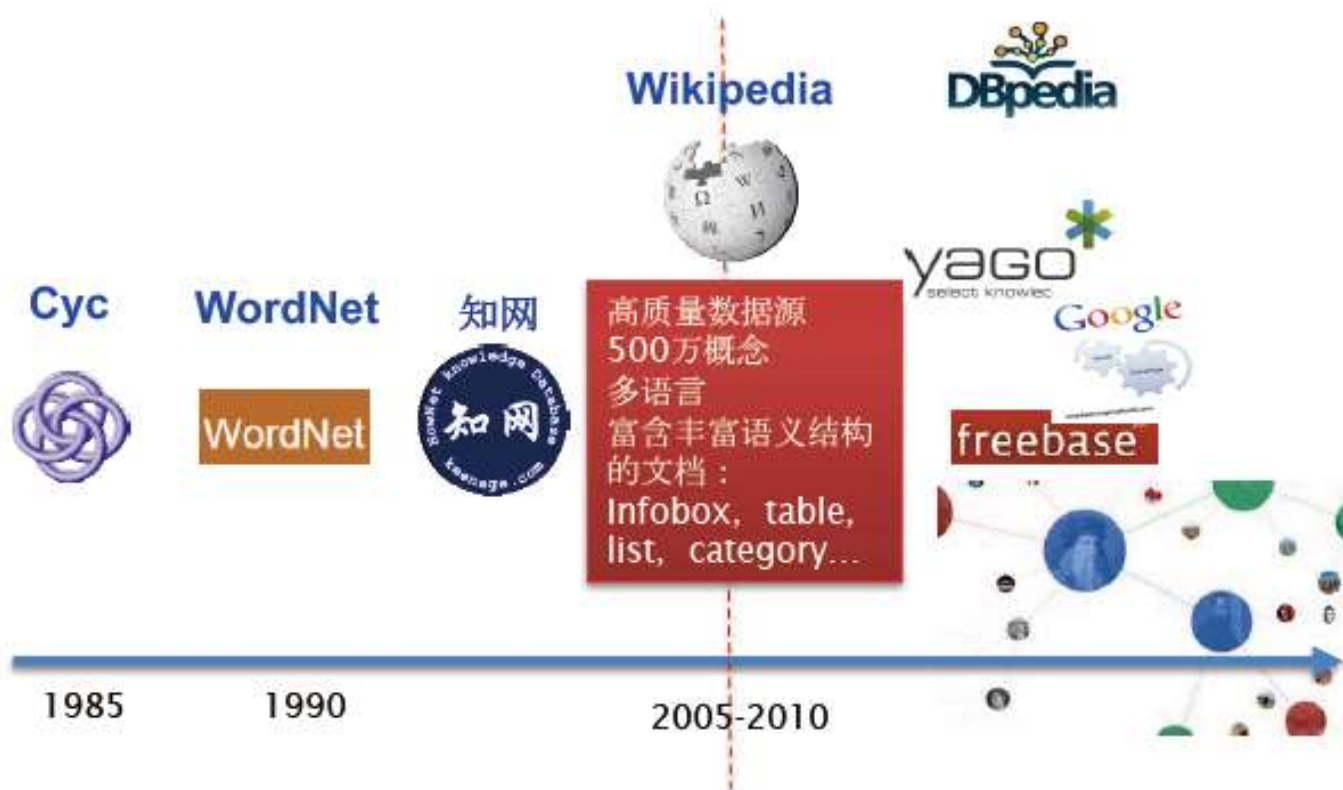
Data web



典型的知识图谱



知识图谱发展





知识图谱的应用

• 问答

WolframAlpha computational knowledge engine

Enter what you want to calculate or know about:

how big is China

Assuming "how big" is international data | Use as referring to socioeconomic data or referring to species or referring to administrative divisions instead
Assuming total area | Use population instead

Input interpretation:
China total area

Result:
 $9.597 \times 10^6 \text{ km}^2$ (square kilometers) (world rank: 4th)

Unit conversions:
 $9.597 \times 10^{12} \text{ m}^2$ (square meters)
3.705 million mi^2 (square miles)
 $1.033 \times 10^{14} \text{ ft}^2$ (square feet)

Comparisons as area:
 $\approx 0.96 \times$ total area of Canada ($9.98467 \times 10^6 \text{ km}^2$)
 $\approx 0.996 \times$ total area of the United States ($9.63142 \times 10^6 \text{ km}^2$)
 \approx largest extent of the Roman Empire ($> 9 \text{ Mm}^2$)

姚明个子有多少

网页 新闻 贴吧 知道 音乐 图片 视频 地图 文库 更多»

百度为您找到相关结果约3,030,000个 搜索工具

姚明身高:
226cm

姚明, 1980年生于上海市徐汇区, 祖籍吴江震泽。中国篮球运动员。1998年4月, 他入选王非执教的国家队, 开始篮球生涯。2002年, 他以状元秀身份被NBA的休斯敦火箭队选中。20... [详情>>](#)

来自百度百科 | 报错

• 精准搜索

奥巴马

网页 新闻 贴吧 知道 音乐 图片 视频 地图 文库 更多»

百度为您找到相关结果约86,500,000个 搜索工具

奥巴马 百度百科

姓名: 贝拉克·侯赛因·奥巴马
生日: 1961年8月4日 职业: 政治家、律师、总统
简介: 贝拉克·侯赛因·奥巴马 (Barack Hussein Obama), 1961年8月4日出生, 美国民主党籍政治家, 第44任美国总统, 为美...
[人物经历](#) [从政生涯](#) [主要作品](#) [家庭生活](#) [人物评价](#) [更多>>](#)
[查看“奥巴马”全部4个含义>>](#)
[baike.baidu.com/](#)

奥巴马的最新相关信息

奥巴马卸任后当NBA球队老板? 白宫发言人: 在讨论

ESPN消息 美国总统奥巴马即将卸任, 而他未来要做的事情似乎已经确定好了, 那就是当一支NBA球队的老板。 奥巴马卸任后当NBA球队老板? 白宫发言人: 在讨论...
网易体育 49分钟前

奥巴马离成NBA球队老板 总统助选才拉杜兰... 腾讯体育 1小时前

奥巴马的算盘 英国留驻利于美国外交 穆报财经 2小时前

奥巴马卸任后仍是特别威胁 求宝对俄制裁... 网易军事 18小时前

作为最亲密的盟友 奥巴马对英国“脱欧”... 新浪财经 18小时前

奥巴马 百度图片

来源: baidu.com 图典 全图727 17.57% 图14

历年时代周刊年度风云人物 展开»

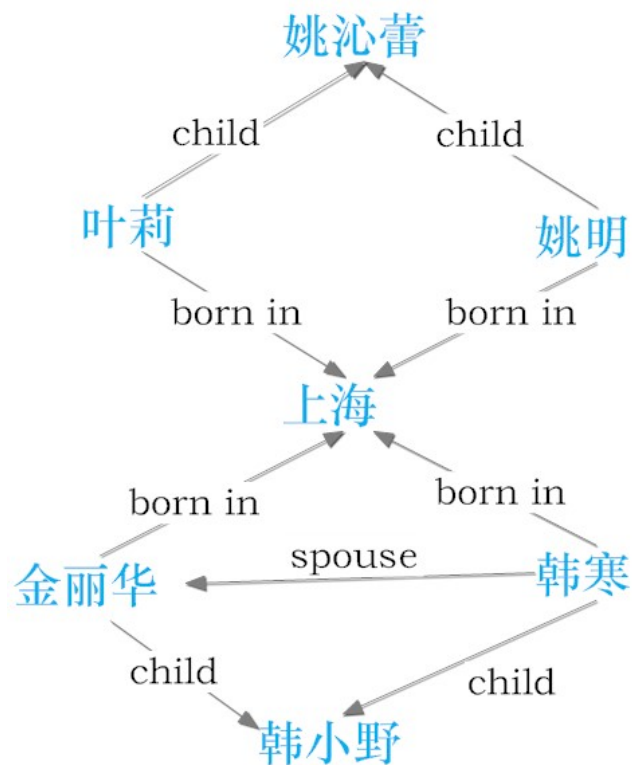
| | | | |
|---------------------|---------------------|-----------------------|-------------------------|
| 普京 俄罗斯联邦总统 | 克林顿 曾任美国总统 | 小布什 美国第43任总统 | 比尔·盖茨 微软公司创始人前首富 |
| 奥巴马 美国第44任总统 | 罗斯福 美国第32任总统 | 斯大林 苏联共产党中央总书记 | 里根 演员出身的美国总统 |

美国民主党员 展开»

| | | | |
|----------------------|-------------------|-----------------------------|---------------------|
| 希拉里 美国第67任国务卿 | 拜登 美国现任副总统 | 卡萝琳·肯尼迪 被美国人称为“全国宝贝” | 克里 美国第68任国务卿 |
|----------------------|-------------------|-----------------------------|---------------------|

基于符号的表示的知识图谱

- 知识图谱本质上是一种语义网络
 - 结点代表实体 (**entity**) 或者概念 (**concept**) ;
 - 边代表实体/概念之间的各种语义关系/属性。
 - 关系事实= (**head, relation, tail**)
 - head**: 头部实体
 - relation**: 关系/属性
 - tail**: 尾部实体





推理机制

- 相等变换：如 $P \wedge \text{True} \vdash \text{True}$
- 假言推理（三段论）：如果 $P \vdash Q$ 且 P 则 Q
- Universal Elimination: $\forall x \text{ Loves}(x, \text{FOPC}) \vdash \text{Loves}(\text{Ray}, \text{FOPC})$
- Existential Elimination: $\exists x (\text{Owns}(\text{Mary}, x) \wedge \text{Cat}(x)) \vdash \text{Owns}(\text{Mary}, \text{MarysCat}) \wedge \text{Cat}(\text{MarysCat})$
- Existential Introduction: $\text{Loves}(\text{Ray}, \text{FOPC}) \vdash \exists x \text{ Loves}(x, \text{FOPC})$
-

- 符号表示知识方法及实现

- **Logic**逻辑表示法:

- 用于描述谓词之间关系的公理;

- **Semantic Net**语义网络表示法

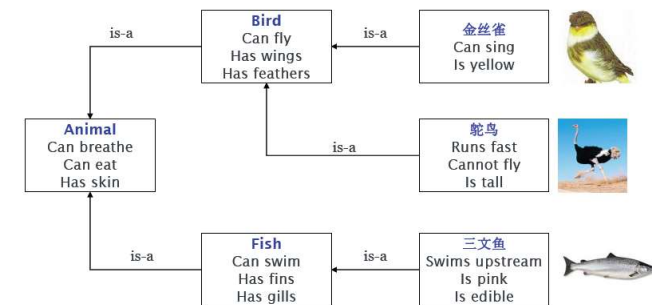
- **Frame**框架表示法

- **Frame**表示与该框架相关的对象集合;

- **Script**脚本表示法

- 是一个事件序列, 包含了一组紧密相关的动作及改变状态的框架;

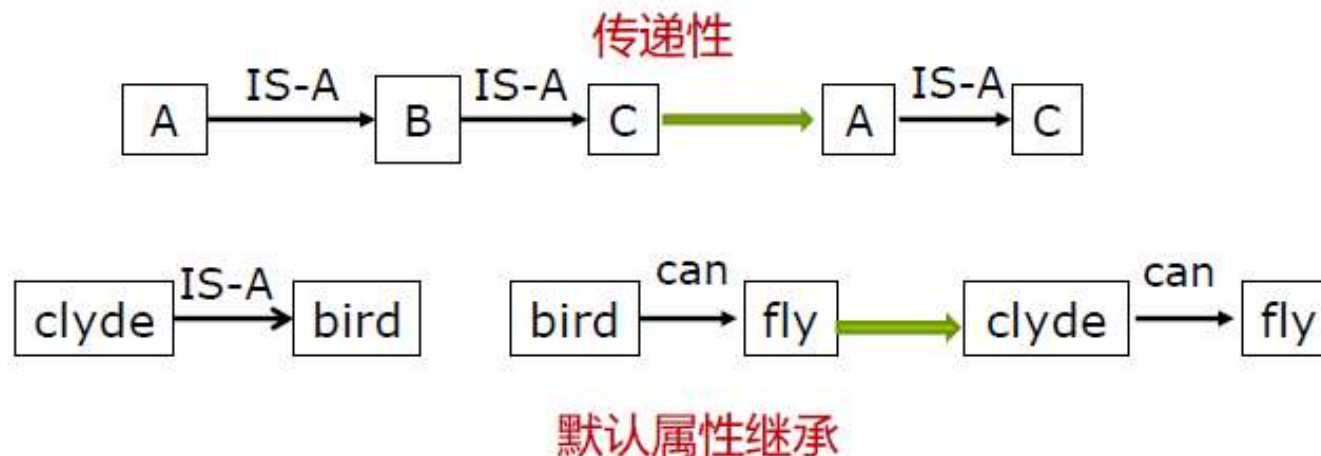
$$\forall x,y \text{ Bachelor}(x) \Leftrightarrow \text{Male}(x) \wedge \text{Adult}(x) \wedge \neg \exists y \text{ Married}(x,y)$$



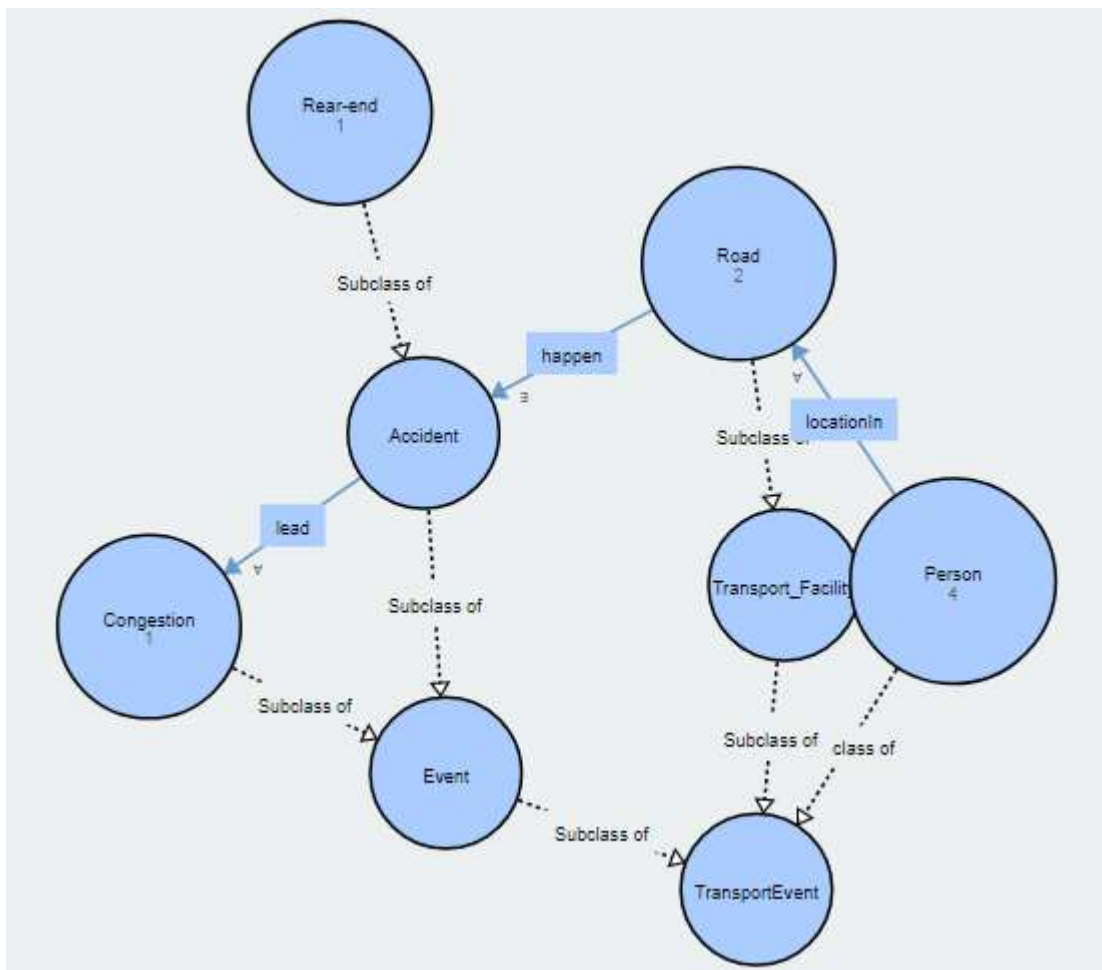
框架例子

语义网中的推理

- 语义网中通常使用的语义关系
 - **IS-A**(概念之间的继承关系)
 - **PART-OF**(整体与部分关系)
 - 领域特定的语义关系



知识图谱和推理实例



以交通事件
图谱为例

$\text{BeijingRoad} \sqsubseteq \exists \text{happen.Rear-end}$
北京路发生追尾。

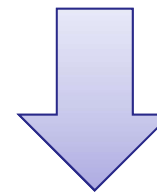
$\text{Rear-end} \sqsubseteq \text{Accident}$

追尾被定义为一种交通事故。

$\exists \text{happen.Accident} \sqsubseteq \text{Congestion}$
发生事故会导致拥堵。

推理规则：

如果用户所在的道路发
生交通事故，则用户被
堵在路上。



| User | Location | Event |
|------|--------------|-------|
| 王华 | ShanghaiRoad | ? |
| 褚歆辰 | BeijingRoad | ? |
| 李由莉 | ShanghaiRoad | ? |
| 吴朗 | BeijingRoad | ? |

哪些用户被堵
在路上？

?

• AllegroGraph

- 一种图数据库；
- 存储(主语, 谓语, 宾语)三元组
- 支持SPARQL和 Prolog；
- 在内建的推理机上进行推理；

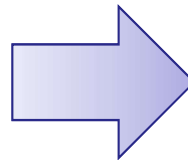
Prolog语句：

(select (?w ?z)

(q- ?w !dom:locationIn ?x)

(q- ?x !dom:happen ?y)

(q- ?y !dom:lead ?z))



| User | Event |
|------|------------|
| 王华 | null |
| 褚歆辰 | Congestion |
| 李由莉 | null |
| 吴朗 | Congestion |

AllegroGraph WebView 6.2.1

Utilities | Admin | User super

Catalogs

system

Repositories

None.

Create new repository

Name:

[Restore from a backup](#)

Start session

Session specification: (☒ autocommit, ☐ load initfile)



中国科学院自动化研究所
INSTITUTE OF AUTOMATION
CHINESE ACADEMY OF SCIENCES

非常感谢同学们的光临
欢迎交流提问