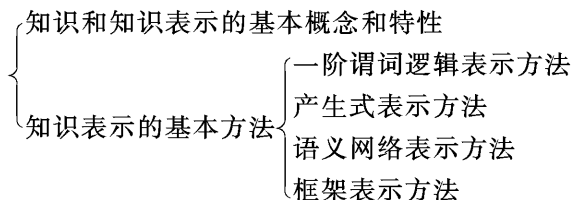


3.1 本章知识结构

本章的重要知识点可以归纳如下：



通过本章的学习读者应该了解知识的基本概念及其特点、分类方法，了解知识表示的基本概念及知识表示方法的评价原则，掌握主要的几种知识表示的方法，会用这些方法表示基本的知识。

3.2 本章主要内容

问题 3.2.1 什么是知识？知识有什么特点？

知识是人类智能的基础，人类在从事生产实践和科学实验等社会实践活动中，其智能活动是一个获取知识和运用知识的过程。人工智能是一门研究用计算机来模仿和执行人脑的某些智力功能的交叉学科，所以，人工智能问题的求解也是以知识为基础的。可以这样认为知识是经过加工的信息 (feigenbaum)，它包括事实、信念和启发式规则 (hayes-roth)。关于知识的研究称为认识论 (epistemology)，它涉及知识的本质、结构和起源。

知识具有下面一些重要特点。

(1) 相对正确性。在一定的条件及环境下，知识一般是正确的，可信任的。

(2) 不确定性。不确定主要指下面的几个方面：

- 由随机性引起的不确定性；
- 由模糊性引起的不确定性；
- 由不完全性引起的不确定性；
- 由经验性引起的不确定性。

(3) 可表示性和可利用性。

- 知识是可以表示出来的；

- 知识是可以利用的。

问题 3.2.2 人工智能中所关心的知识主要有哪些？

知识是建立在数据和信息基础之上的，一般说来人工智能所关心的知识主要包括下面几个方面的知识。

- 事实：是关于对象和物体的知识。如雪是白色的、天是蓝的等。人工智能中的知识表示应能表示各种对象、对象类型及其性质等。事实是静态的、为人们共享的、可公开获得的、公认的知识，在知识库中属底层知识。
- 规则：是有关问题中与事物的行动、动作相联系的因果关系的知识，是动态的，常以“如果……那么……”形式出现。特别是启发式规则是属专家提供的专门经验知识，这种知识无严格解释但很有用处。
- 控制：是有关问题求解步骤、技巧性的知识，告诉怎么样做一件事情。也包括当有多个动作同时被激活的时候应选择哪一个动作来执行。
- 元知识：是有关知识的知识，是知识库中的高层知识。例如，包括怎样使用规则、解释规则、校验规则、解释程序结构等知识。元(meta)的意思是“在……之上”。一个专家可以拥有几个不同领域的知识，元知识可以决定哪一个知识库是适用的。元知识也可用于决定某一领域中哪些规则最合适。元知识与控制知识是有重叠的，对于一个大程序来说，以元知识或者元规则形式体现控制知识更为方便，因为元知识存在知识库中，而控制知识常与程序结合在一起出现，从而不容易修改。

问题 3.2.3 什么是知识表示？人工智能对知识表示方法有什么要求？

知识表示就是研究用机器表示各种类型知识的可行性、有效性的一般方法，可以看作是将知识符号化并输入到计算机的过程和方法。知识表示在智能系统的建造中起到关键的作用。可以说正是以适当的方法表示了知识，才导致智能系统展示出了智能行为。在某种意义上，可以将知识表示视为数据结构及其处理机制的综合：

表示 = 数据结构 + 处理机制

其中，恰当的数据结构用于存储要解决的问题、可能的中间结果、最终解答以及与问题求解有关的世界的描述。这里称存储这些描述的数据结构为符号结构(或者为知识结构)。然而仅有符号结构是不够的，它无法表现出知识的“力量”。为此还需要给出处理机制去使用这些符号结构。因此知识表示是数据结构与处理机制的统一体，既考虑知识语言又考虑知识使用。知识表示语言用符号结构来描述获取到的领域知识，而知识的使用则是应用这些知识实现智能行为。

目前，在知识表示方面主要有两种基本的观点：一种是叙述性的观点；另一种是过程性的观点。叙述性的知识表示观点将知识的表示和知识的运用分开处理，在知识表示时不涉及如何运用知识的问题，例如，一个学生统计表存放了学生的基本信息，为了处理它，必须设计另外专门的程序。显然，由于学生统计表独立存储，使其能为多个程序应用，如

名单打印、学生查询,等等。过程性的知识表示观点将知识的表示和知识的运用结合起来,知识包含于程序之中,如关于一个倒置矩阵的程序就隐含了倒置矩阵的知识,这种知识与应用它的程序紧密地融合在一起,难以分离。在人工智能程序中,采用的比较多的是叙述性的知识表示和处理方法。陈述性知识在设计人工智能系统中处于突出的地位,关于知识表示的各种研究也主要是针对陈述性知识的。在实际应用中人工智能选择和建立合适的知识表示方法可以从下面几个方面考虑。

(1) 表示能力,要求能够正确、有效地将问题求解所需要的各类知识都表示出来。

(2) 可理解性,所表示的知识应易懂、易读。

(3) 便于知识的获取,使得智能系统能够渐进地增加知识,逐步进化。同时在吸收新知识的同时应便于消除可能引起新老知识之间的矛盾,便于维护知识的一致性。

(4) 便于搜索,表示知识的符号结构和推理机制应支持对知识库的高效搜索,使得智能系统能够迅速地感知事物之间的关系和变化;同时很快地从知识库中找到有关的知识。

(5) 便于推理,要能够从已有的知识中推出需要的答案和结论。

问题 3.2.4 用一阶谓词逻辑表示知识的时候一般步骤有哪些? 一阶谓词逻辑表示方法有什么特点?

一阶逻辑是 AI 中大多数知识表示模式的基础,它提供了必要的工具可以用来进行知识表示和推理。它能够通过计算机做精确的处理,而它的表达方式和人类自然语言又非常接近。因此,用数理逻辑作为知识表示工具自然很容易为人们所接受。用一阶谓词逻辑表示知识的基本步骤如下。

(1) 定义谓词、函数及个体,确定它们的确切含义。

(2) 根据所要表达的事物或概念,为每个谓词中的变元赋予特定的值。

(3) 根据所要表达的知识的语义,用适当的连接符号将各个谓词连接起来,形成谓词公式。

逻辑知识表示的主要特点是建立在某种形式逻辑的基础上,并利用了逻辑方法研究推理的规律,即条件和结论之间的蕴涵关系。逻辑是知识表示的基本手段,构成了人工智能研究的基础。逻辑表示方法的主要优点如下:

- 严格性 一阶谓词逻辑具有完备的逻辑推理算法,可以保证其推理过程和结果的正确性,可以比较精确地表达知识。
- 通用性 命题逻辑和谓词逻辑是通用的形式逻辑系统,具有通用的知识表示方法和推理规则,有很广泛的应用领域。
- 自然性 命题逻辑和谓词逻辑采用的是一种接近于自然语言的形式语言表达知识并进行推理的,易于被人所接受。
- 明确性 逻辑表示法对如何由简单陈述句构造复杂的陈述句有明确的规定,各个语法单元(如连接词、量词等)和合式公式定义严格。对于用逻辑方法表示的知识,可以按照一种标准的方法进行解释,因此这种知识表示方法明确、易于理解。
- 模块性 在逻辑表示法中,各条知识都是相互独立的,它们之间不直接发生关系,便于知识的模块化表示,具有易于计算机实现的推理算法。

虽然谓词逻辑在许多情况下很有用,并且从衡量知识表示性能的角度,可以说一阶逻辑也具有充分的表示能力,但是逻辑表示方法也有不足的地方:

- 效率低 形式推理能够使计算机在不知道句子解释的情况下得到有效的结论,它把推理演算和知识的含义截然分开,抛弃了表达内容中所包含的语义信息,往往使推理的过程太冗长,效率低。在推理过程中可能会出现“组合爆炸”。
- 灵活性差 不便于表达启发式知识和不精确的知识。

为了能够表达更多的信息,在谓词逻辑中已经引入了全称量词和存在量词,但仍然有很多类型的语句无法表达,如“大多数同学得了 A”。在这个语句中,量词“大多数”无法用存在量词和全称量词表达。为了表达“大多数”,一个逻辑必须提供一些用于计算的谓词,如模糊逻辑。

问题 3.2.5 产生式系统的基本结构是什么?

一个产生式系统是由规则库、控制子系统以及工作区三部分组成。规则库是 IF-THEN 型规则的集合,工作区记载问题求解的初始状态和中间结果,控制子系统执行识别-行动循环,并在每一循环中选择激活(条件部分满足)的规则和执行其动作,它一般包括匹配器、冲突消解器以及规则解释器,其结构如图 3-1 所示。

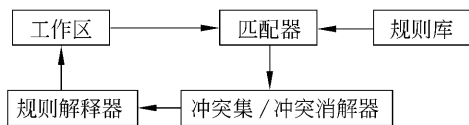


图 3-1 产生式系统的基本结构

产生式系统的规则解释过程可以用一个“识别-动作循环”(recognize-act cycle)来描述。

(1) 从规则库中寻找所有能够和工作区中已有事实相匹配的规则,并将这些规则加入到冲突集中。

(2) 若有多个规则存在,则根据冲突消解策略由冲突集中选择一条规则执行。

(3) 执行规则中的动作,根据动作向工作区中加入新的事实或删除旧的事实。

通常,根据放入工作区中的初始信息开始启动上述循环过程。如果没有规则可以激活或规则显式地说明退出或停止时,上述过程结束。

在产生式系统中,冲突消解器可以用来从多个激活的规则中选择一个规则执行。如果在一个系统中,每次只有一个规则可供选择的话,这类系统称为确定的;否则,称为非确定的。

控制子系统的性能主要和两个方面有关。

(1) 敏感性:系统可以快速地对环境的改变做出响应。

(2) 稳定性:推理过程中表现出连贯性。

问题 3.2.6 产生式系统主要分哪几种类型?

对于产生式系统的类型可以从不同的角度进行划分。按推理方向,可以分为前向、反

向和双向产生式系统。按规则库的结构特征也可以分为可交换的、可分解的和可恢复的产生式系统。

1. 可恢复的产生式系统

可恢复的产生式系统是指采用回溯控制方式的产生式系统。其求解问题的方法是：当执行某条规则后，如果发现所得到的新的工作区 WM 不可能求出问题的解，就立即撤销由该规则所产生的结果，使 WM 恢复到先前的状态，然后再另选其他规则继续求解。

2. 可交换的产生式系统

可交换产生式系统对规则使用的次序是无关的，这使得其求解过程只需要搜索其中任意一条路径，就能达到目标，而不必进行回溯。因此，这种系统的求解过程可不考虑回溯求解策略，可以节省求解问题的时间，提高效率。

3. 可分解的产生式系统

这种方法是把一个较大或较复杂的问题分解成若干较小或较简单的问题，然后通过对这些较小或较简单的问题求解，从而得到最终的解。

问题 3.2.7 产生式系统有什么特点？

产生式系统是目前在人工智能中应用最广泛的知识表示方法，主要原因在于产生式规则最适合表示各种启发式的经验性关联规则，领域专家可以无须知识工程工具就能够把自己的知识转换为 IF-THEN 规则形式。产生式系统可能是一种最简单的表示和利用领域知识的策略。这种方法有许多的优点。产生式系统的特点可以概括为：

(1) 数据驱动。系统是在数据的驱动下运行的，数据一旦发生变化，如增加数据、删除数据等，就会导致系统行为的变化。而一旦数据不再发生变化，系统的运行也就停止了。

(2) 独立性。独立性包含两方面的含义。其一是说，数据(综合数据库)、知识(规则集)和控制(控制系统)，三者之间是相互独立的，而不是混杂在一起的。其二是说，规则之间是相互独立的，不同的规则之间没有明显的连接关系，规则之间是通过数据联系在一起的。这样可以便于规则集的维护，一条规则的修改或者增删，不需要考虑它与哪些规则相关联。由于规则之间是相互独立的，因此一般来说，问题的求解与规则的排列顺序无关。

但是随着要解决的问题越来越复杂，规则库越来越大，则产生式系统问题也会越来越多。主要的问题有：

- 难以扩展。尽管规则形式上相互独立，但实际问题中往往彼此是相关的。这样当知识库不断扩大时，要保证新的规则和已有的规则没有矛盾就会越来越困难，知识库的一致性越来越难以实现。
- 规则选择效率较低。在推理过程中，每一步都要和规则库中的规则做匹配检查。如果知识库中规则数目很大，显然效率会降低。
- 控制策略不灵活。产生式系统往往采用单一的控制策略，如顺序考察规则库中的

每一条规则,这同样会降低系统的效率。

- 知识表示形式单一。产生式系统比较适合于表示非结构化的知识,对于结构化的知识可能用语义网络或框架或面向对象的表示方式更为合适。

产生式系统的效率主要决定于规则激活的顺序,因此也决定于冲突消解策略。所以冲突消解策略的选择非常重要。

问题 3.2.8 什么是语义网络? 语义网络由哪几部分组成?

语义网络是 Quillian 于 1968 年在研究人类联想记忆时提出的心理学模型,该模型认为人的记忆是由概念间的联系实现的。语义网络知识表示方法是表示命题信息的一种经典人工智能表示技术,它是通过概念及其语义关系来表达知识的一种有向网络图。

从图论的观点看,语义网络是一个“带有标示的有向图”。其中,有向图的节点表示各种事物、概念、情况、属性、动作、状态等;弧表示节点之间各种语义关系,指明它所连接的节点之间的某种语义关系。节点和弧必须带有标识,以便区分各个不同对象以及对象之间各种不同的关系。因此,一个语义网络主要包括了两个部分:事件以及事件之间的关系。

从结构上看,语义网络一般是由一些基本的语义单元构成的,这些最基本的语义单元可用三元组表示为

(节点 1,弧,节点 2)

当把多个基本单元用相应的语义联系关联在一起时,就可以得到一个语义网络。

问题 3.2.9 一般语义网络的推理过程是什么?

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,以下分别介绍这两种过程。

1. 继承

在语义网络中所谓的继承是把对事物的描述从概念节点或类节点传递到实例节点。例如,在如图 3-2 所示的语义网络中 BRICK 是概念节点,BRICK12 是一个实例节点。BRICK 节点在 SHAPE(外形)槽中填入了 RECTANGULAR(矩形),说明砖块的外形是矩形的。这个描述可以通过 ISA 链传递给实例节点 BRICK12。因此,虽然 BRICK12 节点没有 SHAPE 槽,但可以从这个语义网络推理出 BRICK12 的外形是矩形的。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。

在语义网络中,一般有三种继承过程,即值继承、“如果需要”继承和“缺省”继承。

1) 值继承

在语义网络的节点之间的语义关系中,我们可以定义 ISA(ISA)链和 AKO 链(A Kind Of),它们都可以用来实现节点之间的值继承。例如,图 3-2 中 BRICK12 的外形是矩形,就是值继承。

总之,ISA 和 AKO 链直接地表示类的成员关系以及子类和类之间的关系,提供了一

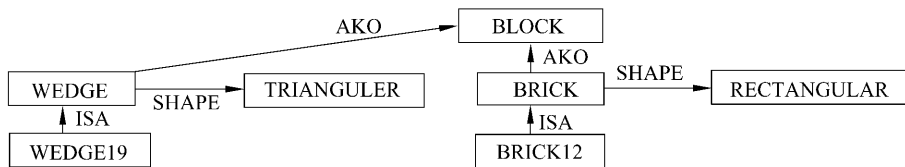


图 3-2 语义网络的值继承

种把知识从某一层传递到另一层的途径。

2) “如果需要”继承

在某些情况下,当我们不知道某个槽值时,可以利用已知信息来计算。例如,可以根据物质的体积和密度来计算积木的重量。进行上述计算的过程,称为 if-needed(如果需要)继承,进行计算的程序称为 if-needed 程序。

为了存储进行上述计算的程序,我们需要改进节点-槽-值的结构,允许槽有几种类型的值,而不只是一个类型。为此,每个槽又可以有若干个侧面,以存储这些不同类型的值。这样,if-needed 程序可以存放在 IF-NEEDED 侧面中。例如在图 3-3(a)中,一个重量计算程序存放在 BLOCK 节点的 WEIGHT 槽的 IF-NEEDED 侧面中。在需要的时候可以执行这个程序,就可以根据 BRICK12 的密度计算出其重量,并把它存入 BRICK12 的 WEIGHT 槽的侧面中,如图 3-3(b)所示。

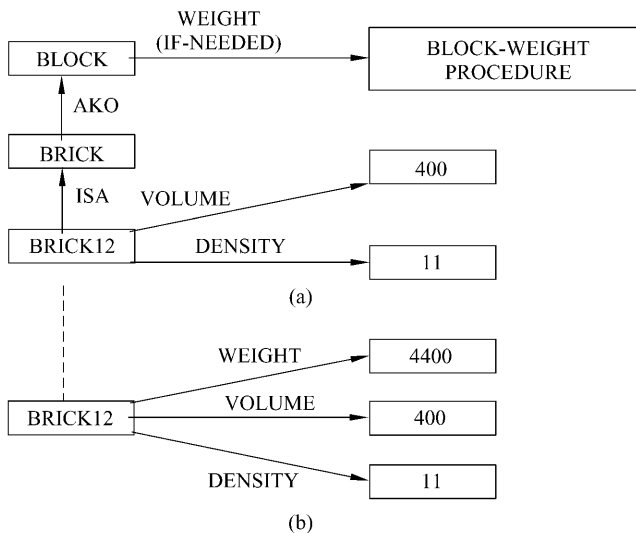


图 3-3 语义网络的“如果需要”继承

3) “缺省”继承

某些情况下,在对事物所做的假设具有一定的真实性,但又不是十分有把握时,最好对所做的假设加上“可能”这样的字眼。例如,可以认为法官可能是诚实的,但不一定是。我们把这种具有相当程度的真实性,但又不能十分肯定的值称为“缺省”值。这种类型的值被放入槽的 DEFAULT(缺省)侧面中。只要不与现有的事实相冲突,就默认这个值为该事物的值,语义网络中的这种推理称为缺省继承。

例如,在图 3-4 中,网络所表示的含义是:从整体来说,积木的颜色很可能是蓝色的,但在砖块中,颜色可能是红的。对 BLOCK 和 BRICK 节点说,在 COLOR 槽中找到的侧面都是 DEFAULT 侧面,在图 3-4 中以括弧加以标志。在图 3-4 中,如果 BRICK 节点没有指定颜色弧的缺省颜色,即 RED,则 BRICK 弧的颜色应是 BLUE。

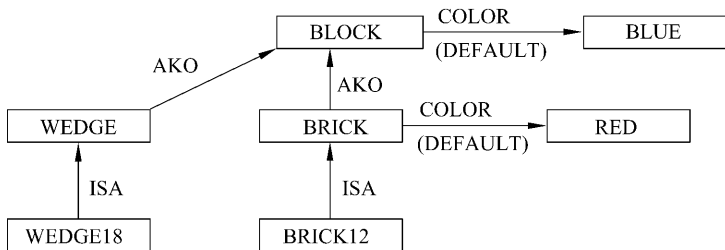


图 3-4 语义网络的“缺省”继承

2. 匹配

前面讨论的是类节点和实例节点,例如,BRICK 和 BRICK12 之间的继承值。对于更为困难一些的问题,如涉及由多个部分构成的事物,如图 3-5 中的 TOY-HOUSE 和 TOY-HOUSE77,继承过程将如何进行。这种情况下,不仅必须确定如何把值从 TOY-HOUSE 传递到 TOY-HOUSE77 的路径,而且还必须确定把值从 TOY-HOUSE 部件传递到 TOY-HOUSE77 部件的路径。

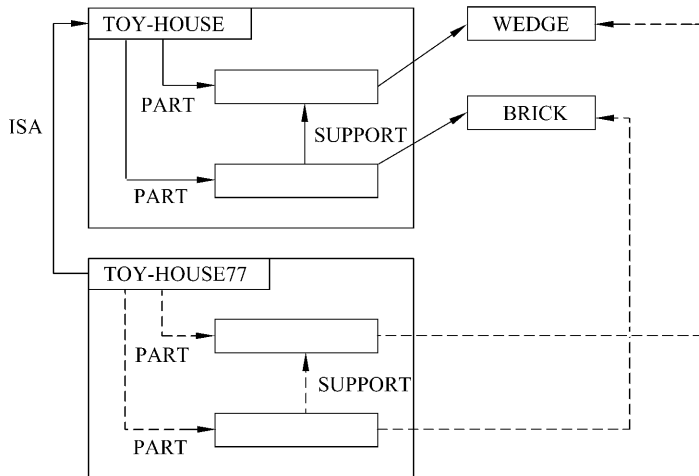


图 3-5 虚节点和虚链

例如,很明显,由于 TOY-HOUSE77 是 TOY-HOUSE 的一个实例,所以它必须有两个部件,一个是砖块,另一个是楔块(wedge)。另外作为 TOY-HOUSE 的一个部件的砖块必须支撑楔块。在图 3-5 中,TOY-HOUSE77 部件以及它们之间的链,都用虚线画的节点的箭头来表示。因为这些知识是通过继承而间接知道的,并不是通过实际的节点和链直接知道的。因此,虚线所表示的节点和箭头表示的链是虚节点和虚链。

对于图 3-6 中的 STRUCTURE35。已知这个结构有两个部件,一个砖块 BRICK12 和一个楔块 WEDGE18。一旦在 STRUCTURE35 和 TOY-HOUSE 之间放上 ISA 链,可知 BRICK12 必须支撑 WEDGE18。在图 3-5 中用虚线箭头表示 BRICK12 和 WEDGE18 之间的 SUPPORT 虚链。因为在语义网络中很容易做部件匹配,所以虚线箭头的位置和方向很容易确定。WEDGE18 肯定和作为 TOY-HOUSE 的一个部件的楔块相匹配,而 BRICK12 肯定和砖块相匹配。

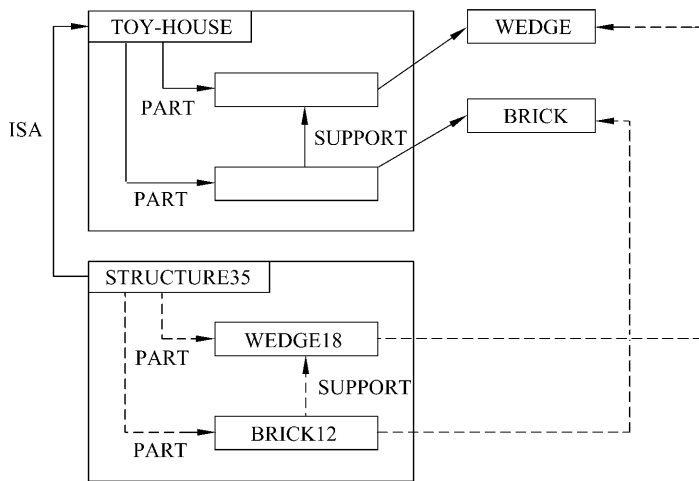


图 3-6 语义网络中部件匹配

问题 3.2.10 语义网络表示方法有什么特点?

1. 语义网络表示方法的主要优点

1) 结构性好

语义网络是一种结构化的知识表示方法,它能够把事物的属性和事物之间的各种语义关系显示地表示出来。通过语义关系的描述和定义,可以实现概念和属性的继承、变异、补充等,实现信息的共享。

2) 联想性

语义网络本身是作为人类联想记忆模型提出的,所强调的就是事物之间的语义联系。这样可以方便实现人类的联想记忆模式。语义网络之间的语义关系可以实现信息的自检索能力,对一般的搜索问题不用对整个知识库进行检索,可以避免搜索时的组合爆炸问题,提高检索的效率。

3) 自然性

语义网络实际上是一个带有标示的有向图,可直观地把事物的属性及事物间的语义联系表示出来,便于理解。自然语言和语义网络之间的转换也比较容易。

2. 主要缺点

尽管语义网络在知识表示方面很有用,但它也有很多局限性。

1) 非严格性

前面提到过语义网络缺乏连接的命名标准,没有公认的形式表示体系,因此一个语义网络的含义完全依赖于处理程序对它的解释,通过语义网络所实现的推理不能保证其正确性,因此说语义网络是不严格的。这使得人们难于理解语义网络的设计意图以及它是否是以一种一致的方式来设计的。

2) 复杂性

另外一个问题就是节点的命名问题。如一个节点被定义为“chair”,它是表示:“一张椅子”、“椅子总类”、“椅子概念”、“会议主席”还是其他什么意思?这些都为语义网络的处理带来了很大复杂性。由于语义网络表示的知识是确定的,即知识是能够定义的,因此在实际当中,必须严格定义连接和节点的名字。

3) 组合爆炸问题

语义网络通过语义之间的联想记忆可以对一般的检索问题很快地给出答案,但是对于有些检索可能会产生组合爆炸,尤其是在查询的回答是否定的时候。因为对于一个产生否定结果的查询,可能要在语义网络中搜索很多甚至所有的连接。

4) 不充分性

由于语义网络不能像逻辑方法那样定义知识,所以语义网络在逻辑上是不充分的。逻辑表示方法可以指定一种椅子、一些椅子、所有椅子以及没有椅子,等等。另一个问题是语义网络的启发性不足,因为语义网络是一种浅知识结构,这种浅知识结构的产生是因为语义网络的所有知识都包含在连接和节点之中。这样就没有办法可以把启发性的信息嵌入到语义网络当中使之有效地搜索语义网络。启发性方法是一种经验,它可以帮助找出解决问题的方法,但并不保证算法可解。启发性方法在人工智能中很有用,因为典型的人工智能问题十分困难,没有一种可行的算法,或者求解算法十分低效。在语义网络中唯一有用的控制策略是继承,但并非所有的问题都可以描述为便于属性继承的结构。

问题 3.2.11 什么是框架理论? 什么是框架?

框架理论认为人们在现实世界中对各种事物的认识都是以一种结构化的模式存储在记忆中的,这种结构化的模式在人们的大脑中形成了一种固定的框架。框架表示了事件或事物的原型结构,当人们遇到一个新的事物时,就从记忆中找到一个合适的框架匹配,并根据实际情况对其细节加以修改、补充。例如,当一个人走进一间教室之前,他就能根据自己脑海中对教室的认识,对教室的整体结构有一个认识,如教室有桌子、椅子、黑板、门、窗、墙等,尽管他可能对这个教室的细节(如教室的大小、门窗的个数、黑板大小、桌椅的数量等)不一定很清楚,但是对教室的基本结构是可以预见到的。他之所以能够做到这一点,是由于他通过以往的认识活动已经在脑海中建立了教室的结构模型,即教室的框架,该框架不仅指出了相应事物的名称(教室),而且还指出了事物各个方面的属性(如有多少椅子和桌子、四面墙的特性等),通过对该框架的查找,很容易得到教室的各项有关特征。

像这样根据以往的经验去认识新事物的方法是人们经常采用的。人们根据过去的经验和认识形成对某一事物的概念模型,这种概念模型是通过一种通用的数据结构形式存