

17.4 机器学习与知识发现

机器学习 (Machine Learning) 和知识发现是智能模拟研究和智能系统开发的重要组成部分。鉴于学习能力和创新能力是智能体应具有的最突出的特性,机器学习作为提高机器智能的重要手段,知识发现作为认识世界的重要功能,已受到人们越来越广泛的关注,成为智能领域研究的核心之一。本节,我们将讨论机器学习和知识发现的基本理论和实现技术,论述几种比较典型的机器学习和知识发现的方法。

17.4.1 机器学习概论

17.4.1.1 机器学习的基本概念

学习是人类智能的主要标志和获取知识的基本手段,人的成长过程本质上就是一个不断地学习,从而不断地向外界吸取经验和获取知识的过程。人工智能研究的主要目标就是使计算机能模拟人类的智能,有效、灵活地完成人所能完成的工作,不管我们采用什么方法达到这一目标,都不能忽视这样一个事实,即智能行为的基础在于具有学习能力。所以,要模拟人的智能,最基本的一点就是要模拟人的学习功能,任何一个没有学习能力的系统都很难被认为是一个真正的智能系统。

准确、完整地给出机器学习的定义比较困难,因不同的学派对机器学习有不同的理解和定义。对机器学习的主要认知有:① **机器学习就是系统性能的改善**。认为学习就是利用经验改善系统自身的性能,学习就是系统行为的改变。比如,西蒙 (H. A. Simon) 提出,“学习是使系统做一些适应性变化,从而系统在下次完成同样的或类似的任务时比前一次更有效”。② **机器学习就是知识的获取**。这是认知主义学派的学习观点,认为学习是知识的获取。一般来说,学习后可以增长知识;然而,获取的知识是否有用需要由行为效果来衡量。另外,有些学习过程不能只用知识来衡量,例如学会游泳,它应是人的一种生理性能的提高。③ **机器学习就是知识表示的改善**。比如,迈克尔斯基 (R. S. Michalski) 提出,“学习是构造或修改所经历事物的表示”。这种观点强调学得任何知识都必须以某种形式来表示和存贮,而系统性能的改善可视为是这种表示改善的目的和结果。同一内容的知识可以有不同的表示形式,并由此对系统的性能产生不同的影响;高效的知识表示形式通常会使系统具有良好的性能和行为。根据对机器学习的这种理解,我们可以认为,机器学习研究的重点应是建立学习的计算理论和方法,构造各种学习系统来获取新知识、新技能,以及用新的方法来组织已有的知识,然后把这些系统应用到各个领域。可以说,上述关于机器学习的定义都是希望从不同的侧面来阐述机器学习的本质,但都不完整。为了对机器学习有一个准确、完整的理解,关于机器学习的定义应达到从内部到外部,从动机到效果的统一。

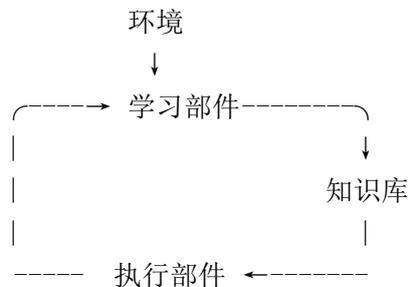


图 17.4.1 学习的简单模型

迪特里奇 (Dietterich) 以西蒙的学习定义为出发点, 提出了一个简单的机器学习模型 (图 17.4.1)。在此模型中, 环境向系统的学习部件提供某些信息, 学习部件利用这些信息修改知识库, 以增进系统执行部件的效能; 执行部件根据知识库 (内容) 完成任务, 同时把获得的信息反馈给学习部件。在具体的应用中, 环境、知识库和执行部件决定了具体的工作内容, 学习部件所需解决的问题主要由与其相关的三个部分确定。由此, 我们可以认为, 机器学习与知识获取密不可分。机器学习本质上就是一种知识获取过程。是机器根据系统 (内部或外部) 环境提供的信息, 通过自身的学习机制来自动获取知识的过程。

17.4.1.2 机器学习的发展历史

关于机器学习的研究大体上经历过以下几个发展时期: 通用学习系统的研究; 基于符号表示的概念学习系统的研究; 基于知识的各种学习系统的研究; 关于联结学习与符号学习的深入研究。

(1) **通用学习系统的研究。**通用学习系统的研究始于上世纪 50 年代, 其主流方式是构造没有或者只有很少初始知识的通用 (智能) 系统, 主要技术是构建以感知机为代表的早期神经网络 (神经元模型): 通过 (有教师指导的) 监督学习来实现神经元间连接权的自适应调整, 产生线性的模式分类和联想记忆能力。鉴于当时技术的限制, 研究主要停留在理论探索和构造专用的实验硬件系统阶段, 系统的学习则主要靠神经元在传递信号的过程中所反映的概率上的渐进变化来实现。也有人开发了应用符号逻辑来模拟神经元系统的工作, 如麦克洛奇 (McCulloch) 和皮兹 (Pitts) 用离散决策元件模拟神经元的理论。相关的工作也包括进化过程的仿真, 即通过随机演变和 “自然” 选择来创建智能系统, 如弗伦德勃

(R. M. Friedberg) 的进化过程模拟系统。这方面的研究最终形成了人工智能的一个分支——模式识别, 并创立了学习的决策论方法。

基于简单神经元模型的学习研究未取得实质性进展, 但一种简单、原始的学习方法——机械式学习 [死记式学习], 却取得了显著的成功。该方法通过记忆和评价外部环境提供的信息来达到学习的目的。采用该方法的代表性成果是塞缪尔 (A. L. Samuel) 设计的跳棋程序, 随着使用次数的增加, 该程序会积累性记忆有价值的信息, 可以很快达到大师级水平。这些成功也激励着研究者们继续进行着机器学习的探索性研究。

(2) **基于符号表示的概念学习系统的研究。**基于符号表示的概念学习系统的研究始于上世纪 60 年代中叶。当时, 人工智能的研究重点已转到符号推理系统和基于知识的方法研究。研究的目的是获得知识的符号描述及获取概念的结构假设。这一时期的工作主要有概念的获取和各种模式识别系统的应用。其中, 最有影响的开发工作当属温斯顿 (Winston) 的基于示例归纳的结构化概念学习系统。受其影响, 人们研究了从例子中学习结构化概念的各种不同方法。也有部分研究者构造了面向任务的专用系统, 这些系统旨在获取特定问题求解任务中的上下文知识, 其代表性工作有巴查纳 (B. G. Buchanan) 等的 META-DENDRAL, 可以自动生成规则来解释 DENDRAL 系统中所用的质谱数据。在这个时期, 机器学习的研究者已意识到应用知识来指导学习的重要性, 并且开始将领域知识注入学习系统。

(3) **基于知识的学习系统研究。**基于知识的学习系统研究起始于上世纪 70 年代中期。人们不再局限于构造概念学习系统和获取上下文知识, 同时也结合了问题求解中的学习、概念聚类、类比推理及机器发现的工作。一些成熟的方法开始用于辅助构造专家系统, 并不断地开发新的学习方法, 使机器学习达到一个新的时期。这一时期研究工作的特点是: ① 注重基于知识的方法。强调应用面向任务的知识和指导学习过程的约束, 认为为了获取新的知识, 系统必须事先具备大量的初始知识。② 积极开发各种各样的学习方法。除了示例学习外, 各种有关的学习策略相继出现, 如示教学习, 观察和发现学习。同时也出现了如类比学

习和基于解释的学习等方法。③ 结合生成和选择学习任务的能力。应用启发式知识于学习任务的生成和选择，包括提出收集数据的方式、选择要获取的概念与控制系统的注意力等。

(4) 关于**联结学习和符号学习的深入研究**。这是机器学习的现代时期。联结学习和符号学习的深入研究导致机器学习领域的极大繁荣。首先，神经网络的研究重新迅速崛起，并在声音识别、图象处理等诸多领域得到很大成功。使得基于神经网络的联结学习从低谷走出，发展迅猛，并向传统的基于符号的学习提出了挑战。同时，符号学习经历了三十多年的发展历程，各种方法日臻完善，出现了应用技术蓬勃发展的景象。最突出的成就有分析学习（特别是解释学习）的发展，遗传算法的成功，和加强学习方法的广泛应用。特别是近年来，随着计算机网络的发展，基于计算机网络的各种自适应学习软件系统的研制和开发，更将机器学习的研究推向新的高度，网络环境已成为人工智能和机器学习的重要试验基地。

17.4.1.3 机器学习的类型

机器学习有各种方法，为了系统地了解机器学习，有必要对其进行分类。一般可用以下四种方式来对机器学习系统进行归类：基于所用的基本学习策略；基于所获取知识的表示形式；基于学习系统的应用领域；综合考虑学习系统的知识表示、基本学习策略和历史渊源等因素。这些分类方法都可从某一侧面来界定一个学习系统的类别。但不管是属于哪一类，每个学习系统都包含着一个对应的学习策略，应用着一种特定的知识表达方式，并适用于某一特定的领域。

1. 基于所用的基本学习策略的学习分类

学习策略是指学习过程中系统所采用的推理策略。从学习所依赖的经验和学习所要获得的结果之间的关系来看，可将学习的基本策略分为归纳、类比和演绎三种。采用归纳策略时，输入的是概念的实例，学习的任务是从这些实例中概括出对于概念的描述，或改进对于概念的已有描述。作为学习系统产生的知识，这些描述可用于促进问题的解决。采用类比策略时，输入的是新问题的描述，学习的任务是寻找系统先前已解决的类似问题，并用解决该问题的经验和知识来处理新问题。采用演绎策略时，输入的新问题可用学习系统已有的知识加以解决，但知识库的相关部分不能被有效地使用，从而学习的任务就是将这些部分转换为更好的形式。

实际上，类比策略可看成是归纳和演绎策略的综合，因而最基本的学习策略只有归纳和演绎。从学习内容的角度看，采用归纳策略的学习由于是对输入进行归纳，所学习的知识显然超过原有系统知识库所能蕴涵的范围，即所学结果改变了系统的知识演绎闭包，因而这种类型的学习又可称为知识级学习；而采用演绎策略的学习尽管所学的知识能提高系统的效率，但仍能被原有系统的知识库所蕴涵，即所学的知识未能改变系统的演绎闭包，因而这种类型的学习又被称为符号级学习。

一个学习系统总是由学习和环境两部分组成。由环境（如书本或教师）提供信息，学习部分则实现信息转换，用能够理解的形式记忆下来，并从中获取有用的信息。在学习过程中，学习者（学习部分）使用的推理越少，他对教导者（环境）的依赖就越大，教导者的负担也就越重。反之，学习者使用的推理越多，教导者的负担就越轻。学习策略的一个分类标准就是根据学习者实现信息转换所需的推理多少和难易程度来分类的。在这种分类中，按学习者所需的推理多少和难易程度，依从简单到复杂，从少到多的次序，可分为以下基本类型：机械学习、示教学习、演绎学习、类比学习、基于解释的学习、归纳学习。

(1) **机械学习**(Rote learning)。学习者无需任何推理或其它的知识转换，直接吸取环境所提供的信息。这类学习系统主要考虑的是如何索引存贮的知识并加以利用。系统的学习方

法是直接通过事先编好、构造好的程序来学习，学习者不作任何工作，或者是通过直接接收既定的事实和数据进行学习，对输入信息不作任何的推理。

以塞缪尔的跳棋程序为例，由于关于跳棋棋盘状态的游戏树具有很深的层次，受时间限制，不可能作穷尽搜索，只能从指示当前棋盘状态的节点 n 作有限深度（例如 3 层）的搜索。程序设计了静态评价函数去计算从节点 n 可到达的最深层（第 3 层）节点的得分（棋势），然后应用所谓的 min-max 搜索技术去搜索从节点 n 到第 3 层棋盘状态的最佳路径，并推算出节点 n 的得分。显然，这样推出的得分比直接用评价函数计算出的得分更能反映节点 n 的棋势。所谓机械学习就是让下棋程序死记住这样推出的得分，以备重用于后继的下棋过程。例如，某个节点 m 指示新的当前棋盘状态，而节点 n 出现在从节点 m 可到达的第 3 层，则可直接取用由下棋程序记住的关于节点 n 的得分，不必再用评价函数计算。如此，不仅可加速对于最深层节点的评价，更有意义的是能起到加大搜索深度的作用：相当于从节点 m 搜索了 6 层。正是由于这种节点得分的积累性记忆和搜索深度的不断增大，使得下棋程序的水平能很快提高。

(2) **示教学习** (Learning from instruction 或 Learning by being told)。学习者从环境（教师或其它信息源）获取信息，把知识转换成内部可使用的表示形式，并将新的知识和原有知识有机地结合为一体。所以，要求学习者要有一定程度的推理能力，但教导者也要做大量的工作。教导者要以某种形式提出和组织知识，以使学习者拥有的知识可以不断地增加。这种学习方法和人类社会的学校教学方式相似，学习的任务就是建立一个系统，使它能接受教导和建议，并有效地存贮和应用学到的知识。目前，不少专家系统在建立知识库时就使用这种方法去实现知识获取。

(3) **演绎学习** (Learning by deduction)。学习者所用的推理形式为演绎推理。推理从公理出发，经过逻辑变换推导出结论。这种推理是“保真”变换和特化 (specialization) 的过程，可使学习者在推理过程中获取有用的知识。这种学习方法包含宏操作 (macro-operation) 学习、知识编辑和组块 (Chunking) 技术等。

与演绎推理相对应的是归纳推理，它也是构成人类学习的最重要方法之一，是发现科学定律和定理的有力工具。与演绎学习相对应，归纳推理是“保假”变换和泛化 (generalization) 的过程。归纳推理和演绎推理常在大多数学习方法中同时使用。

(4) **类比学习** (Learning by analogy)。利用二个不同领域（源域、目标域）中的知识的相似性，可以通过类比，从源域的知识（包括相似的特征和其它性质）推导出目标域的相应知识，从而实现学习。例如，一个从未开过货车的司机，只要他有开小车的知识就可完成开货车的任务。若把某个人比喻为很像“老黄牛”，则可通过观察老黄牛的行为，推断出这个人的性格。所以，类比学习可以使一个已有的应用系统通过学习转变为适应于新的领域的系统，来完成原先没有预先设计的相类似的功能。类比学习需要比上述三种学习方式更多的推理。它一般要求先从知识源（源域）中检索出可用的知识，再将其转换成新的形式，用到新的状况（目标域）中去。

(5) **基于解释的学习** (Explanation-based learning, EBL)。学习者根据教导者提供的目标概念、该概念的一个例子、领域理论及可操作准则，首先构造一个解释来说明为什么该例子满足目标概念，然后将解释推广为目标概念的一个满足可操作准则的充分条件。基于解释的学习已被广泛应用于知识库求精和改善系统的性能。

(6) **归纳学习** (Learning from induction)。归纳学习是由教导者或环境提供某概念的

一些实例或反例，让学习者通过归纳推理得出该概念的一般描述。这种学习的推理工作量远多于示教学习和演绎学习，因为环境并不提供一般性概念描述（如公理）。从某种程度上说，归纳学习的推理量也比类比学习大，因为没有一个类似的概念可以作为其“源概念”加以取用。归纳学习是最基本的，发展也较为成熟的学习方法，在人工智能领域中已经得到广泛的研究和应用。归纳学习根据有无教导者指导，可分为示例学习和从观察和发现中学习。

示例学习(learning from examples)是一种从具体事例中产生抽象概念的方法，因此也称为概念获取。示例学习是人类最基本的学习能力，也是机器学习研究的一个核心领域。已被成功地应用于专家系统的知识库自动构造中。示例学习由教导者提供某个概念的正例集和反例集，学习者通过归纳推理产生该概念的一般描述。它可分成二种类型：① 只有正例，无反例。正例提供了获取概念定义（描述）的依据，但却无法防止概念的过分泛化（Generalization）。在这种状况下，系统采用“保守”原则，即以最低的泛化（抽象）程度来建立概念的定义，或者使用已有的知识来限制（约束）经由归纳推理建立的概念定义。② 有正例，也有反例。这是一种最典型的示例学习：用正例来生成概念定义，又用反例来防止过分泛化。学习者通过归纳推理产生覆盖所有正例并排除任何反例的概念定义，而且要求获取的概念定义也应排斥任何未列入反例集的反例。示例学习系统研究得较多，其中有影响的成果包括亨特和哈兰德的 CLS 及其扩充，昆兰（J. R. Quinlan）的 ID3，米切尔的变形空间等。另外，基于神经网络的有导师的联接学习也可归入此类。

示例学习是有教导者指导的归纳学习，因为教导者已明确划分了正例和反例集。而**观察和发现学习**(Learning from observation & discovery)则是无教导者的学习，环境自动提供了一组观察事例，以支持学习者构造一般的概念描述或理论，所以这种学习是更一般、更复杂的学习形式。它要求有比以上的几种学习方法更多的推理。它既无正、反例集，也无任何的启示来划分正、反例。而且，每次的观察并不限于一个概念，可以同时涉及几个概念。这种学习又可分为观察学习和机器发现二种。

观察学习是指学习者将已知事例分类，并为每一类建立一般性概念描述。无导师的联接学习也可归入此类。观察学习又可依据是否采用渐近方式而分为概念形成和概念聚类。在概念聚类中，学习者对一次性给出的事例分类，产生每一类的概念描述；再用这些概念描述指导进一步分类，直到结果满意为止。在概念形成中，学习者将依次出现的事例分类，产生概念描述，并且构造一个关于类的层次结构。

机器发现(machine discovery)是指学习者从观察的事例或经验数据中归纳出规律或定理。机器发现是观察和发现学习中难度最大，但也是最具有创造性的一种学习形式。机器发现原来是指经验发现，现在又包括了知识发现。经验发现是从经验数据中发现规律和定律。知识发现是从已观察到的事例或已知数据中发现知识（一般是产生式规则）。此外，遗传算法也可并入此领域。

2. 基于所获取知识的表示形式分类

学习系统获取的知识可能有：行为的规则、对事物的描述、问题的求解策略、各种分类准则以及其它可用于任务实现的知识等。对于学习中获取的知识，主要有以下一些表示形式：① **数学表达式的参数**：学习的目标是调节一个数学表达式的参数或系数来达到一个理想的性能。② **形式文法**：在一个以特定语言表达的知识系统的学习中，学习就是通过对该语言的一系列表达式进行归纳，形成该语言的形式文法。③ **产生式规则**：产生式规则一般表示为条件-动作对，已被广泛地使用。学习系统中的学习行为主要是：生成、泛化、特化或合

成产生式规则。④ **形式逻辑表达式**：形式逻辑表达式的基本成分是由命题、谓词、变量、约束变量范围等构成的语句，及嵌入的逻辑表达式。系统中的学习行为就是获得这些表达式。⑤ **图和网络**：有的系统采用图匹配和图转换方案来有效地比较和索引知识。⑥ **框架和模式**（schema）：每个框架包含一组槽，用于描述事物（概念和个体）的各个方面。⑦ **计算机程序和其它的过程编码**：获取这种形式的知识，目的在于取得一种能实现特定过程的能力，而不是为了推断该过程的内部结构。⑧ **神经网络**：这主要用在联接学习中。学习所获取的知识，最后归纳为一个特定结构的神经网络。⑨ **多种表示形式的组合**：有时一个学习系统中获取的知识需要综合应用上述几种知识表示形式。

另外，根据表示的精细程度，也可将知识表示形式分为两大类：泛化程度高的粗粒度符号表示、泛化程度低的精粒度亚符号(sub-symbolic)表示。像决策树、形式文法、产生式规则、形式逻辑表达式、框架和模式等属于符号表示类；而代数表达式参数、图和网络、神经网络等则属亚符号表示类。

3. 基于学习系统的应用领域分类

机器学习在各个领域中得到了广泛应用，并在各应用领域中形成了自己固有的特色。目前最主要的应用领域有：专家系统、认知模拟、规划和问题求解、数据挖掘、网络信息服务、图象识别、故障诊断、自然语言理解和智能机器人等。从机器学习的执行部分所反映的任务类型上看，目前大部分的应用研究领域基本上集中于以下两个范畴：分类识别和问题求解。

分类识别任务要求系统依据已知的分类知识对输入的未知模式(该模式的描述)作分析，以确定输入模式的类属。相应的学习目标就是学习用于分类的准则。问题求解任务要求对于给定的目标状态，寻找一个将当前状态转换为目标状态的动作序列；机器学习在这一领域的研究工作大部分集中于通过学习来获取能提高问题求解效率的知识（如搜索控制知识，启发式知识等）。

4. 综合考虑学习系统的知识表示、基本学习策略和历史渊源等因素的分类

机器学习发展很快，无论是符号学习还是联接学习都派生出了许多分支和新的方法，研究领域不断扩大，使得不少机器学习方法很难用上述分类方法加以归类。综合分类方式则是在对机器学习方法进行分类时，综合考虑各种学习方法出现的历史渊源、知识表示方式、推理策略、结果评估的相似性、研究人员交流的相对集中性以及应用领域等诸因素来做分类。

综合分类方式一般将机器学习区分为以下六类：

① **经验性归纳学习**(empirical inductive learning)。经验性归纳学习采用一些数据密集的经验方法对例子进行归纳学习。其例子和学习结果一般都采用属性、谓词、关系等符号表示。它相当于基于学习策略分类中的归纳学习，但不包括联接学习、遗传算法和加强学习。

② **分析学习**(analytic learning)。分析学习方法是从小数几个实例出发，运用领域知识进行分析。其主要特征为：推理策略主要是演绎，而非归纳；使用过去的问题求解经验(实例)指导新的问题求解，或产生能更有效地运用领域知识的搜索控制规则。分析学习的目标是改善系统的性能，而不是新的概念描述。分析学习包括应用解释学习、演绎学习、多级结构组块以及宏操作学习等技术。

③ **类比学习**。它相当于基于学习策略分类中的类比学习。目前，在这一类型的学习中比较引人注目的研究是通过与过去经历的具体事例作类比来学习，称为基于范例的学习(case_based learning)，或简称范例学习。

④ **进化学习**。包括基于**遗传算法** (genetic algorithm) 的学习。遗传算法模拟生物繁殖的突变、交换和达尔文的自然选择 (在每一生态环境中适者生存)。它把问题可能的解编码为一个向量, 称为个体, 向量的每一个元素称为基因, 并利用目标函数 (相应于自然选择标准) 对群体 (个体的集合) 中的每一个个体进行评价, 根据评价值 (适应度) 对个体进行选择、交换、变异等遗传操作, 从而得到新的群体。遗传算法适用于非常复杂和困难的环境, 比如, 带有大量噪声和无关数据、事物不断更新、问题目标不能明显和精确地定义, 以及通过很长的执行过程才能确定当前行为的价值等状况。同神经网络一样, 遗传算法的研究已经发展为人工智能的一个独立分支, 成为计算智能的一部分。

⑤ **联接学习**。典型的联接模型实现为人工神经网络, 其由被称为神经元的一些简单计算单元以及单元间的加权联接组成。

⑥ **加强学习** (reinforcement learning)。加强学习的特点是通过与环境的试探性 (trial and error) 交互来确定和优化动作的选择, 以实现所谓的序列决策任务。在这种任务中, 学习机制通过选择并执行动作, 导致系统状态的变化, 并有可能得到某种强化信号, 从而实现与环境的交互。强化信号就是对系统行为的一种标量化的奖惩。系统学习的目标是寻找一个合适的动作选择策略, 即在任一给定的状态下选择哪种动作的方法, 使产生的动作序列可获得某种最优的结果 (如累计立即回报最大)。

在综合分类中, 经验归纳学习、遗传算法、联接学习和加强学习均属于归纳学习, 其中经验归纳学习采用符号表示方式, 而遗传算法、联接学习和加强学习则采用亚符号表示方式; 分析学习属于演绎学习。

归纳学习是人类智能的重要体现, 从而也成为机器学习的核心技术之一。归纳学习是从教导者或环境提供的事例中抽象出结论 (对于概念的泛化描述) 的知识获取过程。与演绎推理相反, 归纳推理依赖于具体的事例而不是一般性公理, 推理的目标是形成合理的能解释已知事实和预见新事实的一般性推论。归纳推理的理论就是研究如何运用各种推理技术, 在符号表示的空间中进行启发式搜索。常用推理技术包括泛化 (generalizing)、特化 (specializing)、转换 (transforming)、以及知识表示的修正和提炼 (correcting & refining)。根据所用的具体实例是否由教导者来分类 (即有无教导者指导), 可将归纳学习分为示例学习以及从观察和发现中学习。

17.4.1.4 机器学习的进一步发展

机器学习是智能系统研究的热点, 也是难点。一般认为, **机器学习=神经科学与认知科学+数学+计算**。其中, 神经科学与认知科学在机器学习的研究中扮演着先知者角色。数学充满着神经科学和认知科学与计算之间的沟壑。机器学习研究的一个明显趋势是: 尽管“学习机制”还是研究的动力, 然而, 更为重要的推动力来自“有效利用”信息。

(1) **一些传统领域希望借助机器学习提高智能化水平**。例如, 近年来, 模式识别研究最引人注目的发展是机器学习在这个领域扮演着日益重要的角色。在文本分析与自然语言理解方面, 人们关注的焦点也是机器学习。

(2) **应用驱动的机器学习方法层出不穷**。面对多种多样的信息和各种各样的应用, 机器学习方法也层出不穷。例如, 对稀疏数据的非线性特征抽取方法的研究; 对变化环境具有适应性的机器人的研究; 半监督下的多实例学习研究; 需求是事物排序的搜索引擎的学习方法研究; 大量数据快速过滤 (有害信息过滤) 的自学习方法研究等。这些问题大多数没有坚实的理论基础, 处于实验观察阶段。由于大部分问题不能满足机器学习所需的条件, 因此, 数据

分析方法正成为解决复杂问题的关键之一，领域知识与数据分析的引入也不可避免。

(3) **机器学习存在的一些理论问题亟待解决。**例如，统计类机器学习需要满足独立同分布条件，没有一般的原则；信息向符号的映射，还没有好的方法；机器学习还没有一劳永逸的解决方案。研究的进展依赖其它学科的进展，特别是神经科学与认知科学，以及对数学工具的寻找结果。

(4) **特殊比普适更重要。**在研究方法上，对一类特定问题的深入研究比普适性的研究更重要。例如，本体方法，对一个领域本体的深入研究比广泛地构造本体更重要。

(5) **基于自然语言的学习正引起人们的注意。**研究人类认知活动的切入层次无疑是自然语言层次。这无疑也是对自然语言处理研究的一种肯定，人类主要是通过自然语言来进行学习的，也是给与了自然语言学习以信心。但研究机器的自然语言处理并不容易。

现在，机器学习的研究还有许多挑战性问题需要解决。比如：

(1) **代价敏感问题。**例如，在癌症的医疗诊断中，“将病人误诊为健康人”与“将健康人误诊为病人”的代价是不同的；在信用卡盗用的检测中，“将盗用误认为是正常使用”与“将正常使用误认为是盗用”的代价也是不同的。传统的机器学习技术基本上只考虑同一代价，如何处理代价敏感性？需要研究。

(2) **不平衡数据问题。**以乳腺癌诊断为例，“健康人”样本远远多于“病人”样本；再以信用卡盗用检测为例，“正常使用”样本远远多于“被盗用”样本。传统的机器学习技术基本上只考虑平衡数据，如何处理数据不平衡性？也需要深入研究。

(3) **理解数据并自动解释问题。**以乳腺癌诊断为例，需要向病人解释“为什么做出这样的诊断”；以信用卡盗用检测为例，需要向公安部门解释“为什么这是正在被盗用的卡”，传统的机器学习技术基本上只考虑泛化不考虑理解，如何处理可理解性？也是需要研究的问题。问题是发展的动力，这些挑战性问题存在与解决无疑也是机器学习进一步发展的动力。

17.4.2 示例学习

示例学习是有教导者指导的归纳学习，其所用的实例由教导者预先分类。

1. 示例学习问题的基本描述

从例子中获取概念描述的学习方法是机器学习中研究的最深入的一种方法。早在上世纪70年代中期，温斯顿(Winston)就以积木世界为例，设计了著名的结构化概念学习程序。该程序从获取和分析积木块的线条画开始，通过近似匹配、概念泛化和概念特化技术，从一系列正、反示例中归纳出某类积木块的概念定义并表达为语义网络的结构化描述。可以说，示例学习的任务是基于某个概念的一系列实例，生成一个反映该概念本质的定义。这个定义应覆盖所有的正例，而不包含任何反例，同时又可用来指导对新例子的分类识别。

示例学习遵从一般的归纳推理模式，其问题可描述如下：

已知：1. 关于观察（观察到的事例）的描述 F （表示与某些对象、状况、过程等事例相关的特定知识）；2. 初始的归纳断言（可以是空的）；3. 问题域的背景知识（用于约束关于观察的描述和归纳断言的表示）；

求：归纳断言 H ，其应蕴涵关于观察的描述，并满足背景知识。

我们知道，一个断言 H 蕴涵 F （记为 $H \Rightarrow F$ ）是指 F 为 H 的逻辑结果（ H 特化为 F ）[也可记为 $F \Leftarrow H$ （ F 泛化为 H ）]。如果 $H \Rightarrow F$ 成立，并且 H 为真，则 F 必为真。因此，从 H 推出 F （演绎推理）是“保真”的。反之，从 F 推出 H （归纳推理）是非“保真”的。但是若 F 是假的，则 H 必为假，称之为“保假”。

对于任一给定的事例集合，可能生成无穷多个蕴涵这些事例的假设。因此，背景知识是必需的，以便提供约束和评判标准，使归纳推理的结果集中于一个或几个有限的最优假设。

2. 示例学习过程中搜索和获取概念描述（定义）的基本思路

在示例学习中，概括（覆盖）所有正例的概念描述，称为完全描述；而不概括任何反例的概念描述则称为一致描述。对于任何包含正、反例的例子集，都可获得既完全又一致的概念描述，称为解描述；一般情况下，解描述可以有无数个。实际上，任何学习系统（包括人）都要求解描述遵从一定的标准（或限制），包括解描述的表示语言（如句法、词汇），产生解描述的策略和选取最佳解描述的评判标准等，这些标准构成了问题域的背景知识。

示例学习系统引入了二个概念：例子空间和假设空间（又称概念空间）。所有可能的正、反例构成例子空间；可能的概念描述称为假设，它们构成假设空间。假设空间中的每一假设都对应于例子空间中的一个子集，使得该子集中的例子均是该假设的例子。

若假设空间中两个假设 D_1 、 D_2 ，其中 D_1 所对应的例子集是 D_2 所对应例子集的子集，则称 D_2 比 D_1 泛化，或称 D_1 比 D_2 特化。假设空间中各假设间可能存在泛化关系，泛化关系是反对称、可传递的，因而假设空间其实就是一个半序集（偏序集）。

当用于表示概念描述的语言确定时，假设空间也就确定了。鉴于学习过程是知识不断增长的过程，用于表示概念描述的词汇也可能不断增多，假设空间可以动态扩展。

在示例学习中，任何概念获取的过程都可以看成假设空间（概念空间）中的搜索过程，如图 17.4.2 所示。且不同的搜索方式对应于不同的学习策略：

- 逐步泛化的学习策略，对应于自底向上的搜索；
- 逐步特化的学习策略，对应于自顶向下的搜索；
- 双向学习策略，对应于双向搜索。

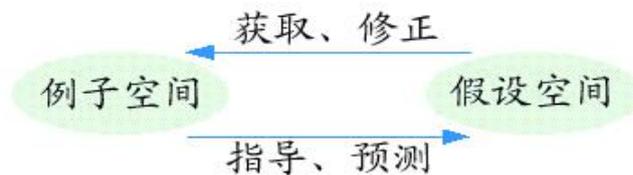


图 17.4.1 例子空间和假设空间

由于泛化/特化是反对称、可传递的关系，所以假设空间是个半序集；即上下层假设间具有泛化/特化关系，而同层假设间是无关的。假设空间中的搜索方式可划分为特化搜索和泛化搜索。前者从最泛化的假设（概念描述）出发，每次取用一个新的例子，就产生一些特化的描述，直到将初始最泛化的假设特化为解描述；后者则从最特化的假设（相应于例子空间中的一个例子）开始，每次取用一个新的例子时，就产生一些泛化的描述，直到产生出足够泛化的解描述。实际上，大多数机器学习方法都采用这二种方法或这二个方法的结合。当然，我们不排除其它的各种搜索方法，如遗传学习算法中就不使用这种半序性搜索方式。

3. 逐步泛化的学习策略

在逐步泛化的学习策略中，正例用于指导系统生成泛化的假设，而反例则用来剪裁过于泛化的假设。系统按此模式进行下去：遇见正例就泛化某些假设以保证假设的完全描述性，遇见反例则删去某些假设以保证假设的一致描述性，直至得到一个既完全又一致的解描述（假设）为止。这个解描述即可作为满足给定例子集的概念定义—亦即学习系统所获得的新知识。逐步泛化的学习策略采用宽度优先、自底向上的搜索方式。其具体实现的算法如下：

H: 当前的假设集, H 的初始值就是第一个观察的正例 P;

N: 已观察到的反例集, N 的初始值为空集 {};

观察的下一个例子记为 i,

如果 i 是正例,

则 ① 对每一个不覆盖 i 的假设 $h \in H$, 用能覆盖 i 和 h, 且泛化程度又最低的假设 (可以有多个) 代之;

② 移去 H 集中泛化程度并非最低的假设 (即过于泛化的假设);

③ 移去 H 集中能覆盖以往观察到的反例 $n \in N$ 的假设 (以保证一致性);

如果 i 是反例,

则 ① 把 i 加入到反例集中 N;

② 从集 H 中移去能覆盖 i 的假设;

重复上述过程, 直到观察尽所有的例子, 得到的 H 就是解描述的集合。

在对假设/例子对进行归纳时, 我们采用的通常是保守的原则——最低限度的泛化, 即新的假设刚好覆盖现有的假设/例子就行了。

4. 逐步特化的学习策略

与泛化策略相比, 特化策略沿相反的方向搜索, 但是实施这二种策略的基本方式是类似的, 即新例子的加入会导致新假设的增加和已存在假设的删除。但是在特化策略中, 正例和反例所起的作用同在泛化策略中的作用是相反的。逐步特化的学习策略从最泛化的假设开始。系统是依靠反例来生成一些特化假设。其具体实现的算法如下:

H: 当前假设集, H 初始化为最泛化的假设 (概念描述);

P: 已观察到的正例集, P 初始化为空集 {};

观察的下一个例子记为 i,

如果 i 是一个反例,

则 ① 对每一个覆盖 i 的假设 $h \in H$, 用可被 H 覆盖但排斥 i, 且特化程度最低的假设代之;

② 移去 H 集中那些特化程度并非最低的假设 (即过于特化的假设);

③ 移去 H 集中不覆盖已往观察到的正例 $p \in P$ 的假设;

如果 i 是一个正例,

则 ① 把 i 加入 P 中;

② 从集 H 中移去所有不覆盖 i 的假设;

重复上述过程, 直至观察尽所有例子, 得到的 H 就是解描述的集合。

显然, 以特化策略获得的解描述 (学习系统期望的概念描述) 是特化程度最低的, 而以泛化策略获得的解描述则是泛化程度最低的。如果给出充分多的例子, 那么二者的结果就可能是相同的概念描述。当然, 当训练例子不多时, 二者的结果可能会不一样。在此过程中我们同样采用了“保守”的原则: 最低限度的特化, 使得特化的假设在覆盖已有正例的同时只是刚好能排斥反例。

值得一提的是, 这二种策略都要求例子自身正确无误, 即无噪声。对于有噪声的情况, 会导致学习方法的实现难度加大。

5. 双向学习策略

双向学习策略是将上述二种策略结合起来，同时从二个方向搜索假设（概念描述）空间，以期获得仅用单一策略所不具有的优点。双向学习策略的一种常见做法是，从一个特定的假设开始，每当遇到一个新的正例时就作泛化搜索，而当遇见一个新的反例时，则作特化搜索。

米切尔以不同的方式实现双向学习策略，称为版本空间法 (Version Space)。该方法用两个假设集 S 、 G 分别表示作泛化、特化搜索的假设空间。当遇见一个新的正例时，如不能使 S 集的描述覆盖该正例，则在该集中进行泛化搜索；而当一个新的反例产生时，如被 G 集包含，则在该集中进行特化搜索。 S 和 G 实际上指示了期望获取的最终解描述的上、下界，当 S 、 G 合一时，合一的解描述就是期望得到的概念描述（定义）。

版本空间法的特点之一是系统不必保留正例和反例。因为 S 本身就蕴涵了已取用的所有正例，可用来删除 G 集中过于特化的假设；同样 G 本身就蕴涵了对所有已取用反例的排斥，可用来消除 S 集中过于泛化的假设。另一特点是系统知道何时推理任务完成，即当 S 、 G 合一时。该算法的具体实现如下：

输入第一个正例 p ；

初始化 S 为和 p 一致的最特化的假设；

初始化 G 为和 p 一致的最泛化的假设；

对以后每一个遇见的例子 i ，

如 i 是一个反例，

则 ① 保留 S 中那些不覆盖 i 的假设；

② 将 G 中覆盖 i 的假设替换成不再覆盖 i 的特化假设，并使它们每个都能蕴涵 S 中的相应假设；

③ 删去 G 中那些可被 G 中其它假设蕴涵的假设；

如 i 是一个正例，

则 ① 保留 G 中那些覆盖 i 的假设

② 将 S 中不覆盖 i 的假设替换成覆盖 i 的泛化假设，并使它们每个都能被 G 中的相应假设所蕴涵；

③ 删去 S 中那些可以蕴涵 S 中其它假设的假设。

版本空间法可以保证找到所有和观察例子相一致的概念描述（在描述语言允许的表示范围内），并且和例子提出的次序无关。另外 S 集和 G 集都以一种有效的方式表示了假设（概念描述）空间，由于它反映了例子的有关信息，所以例子用过之后就可不必要再保留。

6. 决策树学习

如果学习的任务是对一个大的例子集作分类概念的归纳定义，而这些例子又都是用一些无结构的属性值对来表示，则可以采用示例学习方法的一个变种——决策树学习，其代表性的算法是昆兰（J.R. Quinlan）提出的 ID3。

ID3 的输入是描述各种已知类别实例的列表。例子由预先定义的属性值对来表示。归纳推理产生的结果不是以往讨论的那种合取表达式，而是一棵决策树（也称判别树，并可转而表示为决策规则的一个集合），用它可正确地地区分所有给定例子的类属。

树中的每一非叶节点对应一个需测试的属性，每个分叉就是该属性可能的取值；树的叶节点则指示一个例子事物的类别。ID3 的显著优点是归纳学习花费的时间和所给任务的困难度（取决于例子个数，用来描述对象的属性数，所学习概念的复杂度即决策树的节点数等）仅成线性增长关系。当然，ID3 只能处理用属性-值对表示的例子。

在 ID3 中，每一个例子用相同的一组属性来表示，每一个属性又有自身的属性值集，如“颜色”属性可取值是{红、绿、兰}等。构造决策树的目的是为了对事物作出正确的分类。决策树形式的分类规则适用于任何的对象集 C。如 C 是空的，那么它无需分类，对应的决策树也为空；如 C 中的对象是同一类的，那么决策树就一个叶节点，即该类名；如果 C 集中的对象属于二个不同的类别，那末我们可以选取一个对象的属性，随后按其可能值把 C 划分成一些不相交的子集 C_1, C_2, \dots, C_n ，其中 C_i 是含有所选属性的第 i 个值的那些对象集。对每一个这样的子集又可以用同样的策略处理，最后的结果是一棵树。它转而可表示为决策规则的一个集合，用于区分待识别事物的类属。

下面给出的是一个关于人分类的例子（对象）集，并预先定义了指定的一组属性及其可取值：高度{高，矮}，发色{黑色，红色，金色}和眼睛{兰色，棕色}。这里，将人分为两类，分别以 a(+)、b(-) 来指示。

高度	发色	眼睛	类别
矮	黑色	兰色	b
高	黑色	兰色	b
矮	金色	兰色	a
高	金色	棕色	b
高	黑色	棕色	b
矮	金色	棕色	b
高	金色	兰色	a
高	红色	兰色	a

如我们首先选取“发色”为树的根节点，即可获得图 17.4.3 所示的决策树。

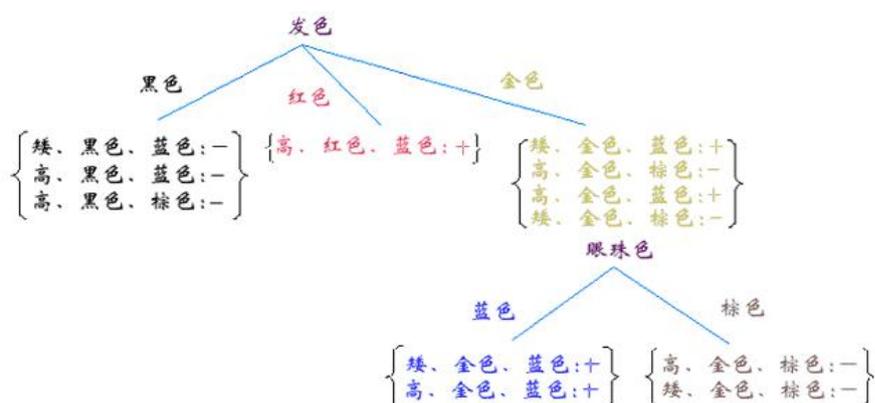


图 17.4.3 示例决策树

相应于“黑色”、“红色”和“金色”三值都有一个对象子集。随后我们再测试“金色”这一分支，又按属性“眼睛”把其所含的对象集划分成兰色和棕色二类。至此，所有叶结点相应的对象子集只含同一类的对象，我们就可以用相应的类别名（本例中的 a 和 b）来取代各子集，得到一决策树，如图 17.4.4 所示。

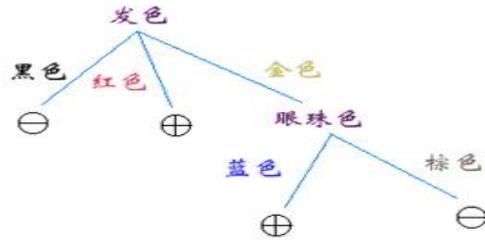


图 17.4.4 带有类别名的决策树

一个对象所属类的判别从决策树的根开始，首先依据根节点指示的属性（发色），选择与该对象属性值相应的分支；然后，以同样的方式进行下去，一直到达叶节点。有时判别一个对象的类别，由于从根到叶节点的路径较短，只要测试少量的属性。如上例中，若“发色”的值是“黑色”或“红色”，则对象就可以直接归属相应的类别，而不必再去判定其它属性的取值；若为“金色”，则再测试一次“眼睛”的值（而不必考虑“高度”）就可以进行判别。

若不存在这样的二个对象：它们在每个属性上都具有相同的值，却属于不同的类别，那么这种生成决策树的过程是可行的。产生这种归纳学习方法的关键在于如何选择一系列有用的属性来测试一个对象集，以使生成的决策树是最小的。ID3 采用了 Shannon 信息论中的方法以使分类时期望（平均）的测试次数最小。

一个决策树可看成一个信息源，即给定一个对象，可从决策树产生一个该对象所属类别的消息（比如类别“a”或“b”）。决策树的复杂程度与借助这个消息所传递的信息量密切相关。若决策树传递的不同类别消息的概率用 P_+ （对应于“a”类）和 P_- 表示（对应于“b”类），那么这个消息的期望信息量为： $-P_a \log_2 P_a - P_b \log_2 P_b$ 。

对给定的物体集 C ，我们可以把这概率近似地表示为相对频率，此时 P_a 和 P_b 分别等于 C 中类别为“a”和“b”的对象所占的比例。

从 C 集对应的决策树中得到消息的期望信息量记为 $M(C)$ ，并定义 $M(\{\})=0$ 。对于上述例子， C 集有 8 个例子，3 个为“a”，5 个为“a”，则 $M(C) = -(3/8) \log_2 (3/8) - (5/8) \log_2 (5/8) = 0.954 \text{ bits}$ 。

图 6.13 是一个局部决策树， A 为构造决策树时下一个可能选取的属性， A_i 为属性 A 的值且是互斥的。属性 A 将集合 C 划分为若干个子集合 $\{C_1, C_2, \dots, C_n\}$ 。设 $M(C_i)$ 是值为 A_i 的子集 C_i 所对应决策树的期望信息量，则确定以属性 A 作为树根的决策树的期望信息量 $B(C, A)$ 可通过权值平均而得到：

$$B(C, A) = \sum (A \text{ 值为 } A_i \text{ 的概率}) * M(C_i)$$

同样我们把 A_i 的概率用相对比例来代替，即在所有对象中具有 A_i 值所占的相对比例。

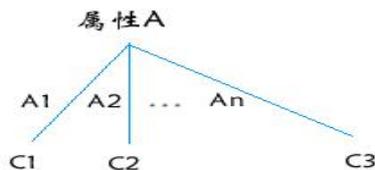


图 17.4.5 局部决策图

我们希望待选的测试属性能使决策树获得最大的信息增益即 $M(C) - B(C, A)$ 为最大值。因为 $M(C)$ 为判别一个对象的类属所要求的总的期望信息量， $B(C, A)$ 为按属性 A 构造局部决

策树后还需要的期望信息量。二者的差越大，说明测试这个属性所能传递的信息量越大，则判别的速度也就越快。

例如，图 17.4.6 中，我们选取的属性为“高度”，对高度值为“高”的分支的所需期望信息量为： $-(2/5) \log_2 (2/5) - (3/5) \log_2 (3/5) = 0.971$ bits；同样对值为“矮”的分支为： $-(1/3) \log_2 (1/3) - (2/3) \log_2 (2/3) = 0.918$ bits。则以属性“高度”作划分后进一步判别所需的期望信息量为： $B(C, \text{“高度”}) = 5/8 * 0.971 + 3/8 * 0.918 = 0.951$ ；则测试这属性传递的信息为： $M(C) - B(C, \text{“高度”}) = 0.954 - 0.951 = 0.003$ bits。

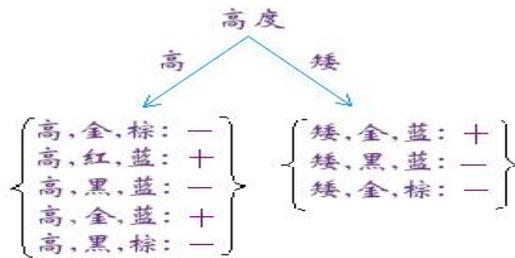


图 17.4.6 以“高度”判别的决策树

如果我们不是选择“高度”而选用属性“头发”，如图 17.4.4 所示，则有： $B(C, \text{“头发”}) = 3/8 * 0 + 1/8 * 0 + 4/8 * 1 = 0.5$ bits，则测试属性“头发”获取的信息为 $M(C) - B(C, \text{“头发”}) = 0.945 - 0.5 = 0.45$ bits。同样对属性“眼睛”测试，获得信息为 0.347bits。根据最大信息量原理，ID3 就选取“头发”为决策树的根节点属性。

ID3 通过不断的循环处理，逐步求精决策树，直到形成一棵完整的决策树，即树的叶节点所对应的对象（例子）均属于同一类。

17.4.3 基于解释的学习

基于解释的学习是分析学习的主要方式：旨在通过应用领域理论（领域知识）对单一事例所作的分析，构造满足预定目标概念并遵从可操作准则的一个解释。与基于大量训练例作归纳推理的数据密集型学习方法不同，基于解释的学习是知识密集型的。

归纳学习可视为一种基于相似的概念学习方法，其通过分析一组正例的共性和鉴别与反例的差别，找出既能概括所有正例，又能排除任何反例的概念描述。尽管归纳学习能应用问题域的背景知识去约束归纳推理过程，包括指定解描述的表示语言（如句法、词汇），产生解描述的策略和选取最佳解描述的评判标准等，但仍很受限制。这难免会使最终的解描述可解释性差，甚至得不到解描述，或得到许多解描述而使系统无从选择。同时，大量的正、反例也给归纳推理带来了严重的计算复杂性。

分析学习利用丰富的领域背景知识，只需通过分析很少的几个例子（通常是一个例子），就能将例子泛化为对目标概念的解释。分析方法主要依赖于演绎推理，以产生更有效的问题求解知识，如搜索控制知识。因而，分析学习的主要目的，是提高问题的求解效率而非获取新的概念描述。

1. 基于解释的泛化 (Explanation-Based Generalization EBG)

1986 年，米切尔 (T. Mitchell) 等人总结了通过分析观察到的例子的特性而进行学习在本质上所有共同的特征，给出了称之为**基于解释的泛化 (EBG)** 的机器学习方法。**基于解释的泛化 (EBG)** 的问题描述包含四个输入参数，即：目标概念，训练例子，领域理论和可操作准则。具体描述如下：

给定：

- 目标概念：对于所学概念的一个初始描述（其尚不满足可操作准则）；
- 训练例子：目标概念的一个正例；
- 领域理论：解释训练例子为何是目标概念正例时可用的规则和事实集合；
- 可操作准则：学到的知识（对于目标概念的解释）所需遵从的表示形式，以使这些知识能用于问题求解活动。

获取：

对于目标概念的一个特化描述，其是训练例子的泛化，且满足可操作准则。

基于解释的泛化的特点是能充分利用问题领域的知识和所学概念的有关知识。这些知识体现于领域理论中。关于目标概念的最初描述也遵从领域理论，只是过于一般化（泛化），且不满足可操作准则，即该描述不能被系统有效使用。训练例子的作用就是指导学习过程将概念的最初描述特化到一定的表示形式。而衡量这一表示形式的标准便是可操作准则，其使目标概念的特化描述能在系统中被有效地使用。

作为一个例子，米切尔等人考虑了目标概念为 Safe-to-stack(x, y) 的学习，即学习任务是识别一对物体(x, y)，使得 x 能安全地放在 y 上面。这个问题的描述如下：

给出：

- 目标概念：一对物体(x, y)，使得 Safe-to-stack(x, y)
- 训练例子：

On(obj1, obj2)
ISA(obj1, BOX)
ISA(obj2, ENDTABLE)
Color(obj1, RED)
Color(obj2, BLUE)
Volume(obj1, 1)
Density(obj1, 1)
.....

- 领域理论：

not (Fragile(p₂)) ∨ Lighter(p₁, p₂) → Safe-to-stack(p₁, p₂)
Volume(p, v) ∧ Density(p, d) → Weight(p, v*d)
Weight(p₁, w₁) ∧ Weight(p₂, w₂) ∧ less(w₁, w₂) → Lighter(p₁, p₂)
ISA(p, ENDTABLE) → Weight(p, 5)
Less (1, 5)
.....

- 可操作准则：目标概念的特化描述必须用那些描述训练例子时使用的谓词来表示（如 Volume, Color, Density）或选用领域理论中容易评价的谓词（如 Less）。

获取：目标概念的一个特化描述，其是训练例子的泛化，且满足可操作准则。

目标概念的最初描述仅指出让 x 安全地放在 y 上，但不满足可操作准则。领域理论说明若物体 p₂ 是非易碎的或物体 p₁ 的重量比 p₂ 轻，则 p₁ 可安全地放在 p₂ 上，也说明了关于物体，重量等方面的知识。训练例子是一个关于物体 obj1 可安全放在物体 obj2 上的具体实例。可操作性准则说明了目标概念的特化描述必须使用的谓词。

米切尔等人提出基于解释的泛化过程可分以下二个阶段进行，即：

(1) 解释：使用领域理论建立一个证明训练例子满足目标概念定义（初始描述）的解释结构；该结构可表示为一棵证明推理树，又称解释树，其每个分枝的叶节点上的表达式都必须满足可操作准则。

(2) 泛化：通过将解释结构中的常量变换为变量（以实现对于训练例子的泛化），获得对于目标概念的一个特化描述，并使其满足可操作准则。这可通过在解释结构中对目标概念进行回归来完成。对回归所得的表达式（相应于解释结构中的叶节点）加以合取，就建立了目标概念的一个特化描述。

在**基于解释的泛化**的第一个阶段（即解释阶段），系统的任务是确定例子的哪些特性与目标概念有关，哪些特性是无关系的，即建立关于训练例子如何满足目标概念的一个解释。所谓解释就是一个证明，其实际上是一个演绎推理过程。图 17.4.7 是证明所产生的解释结构。

如图 17.4.7 所示，一对物体(obj1, obj2)之所以满足目标概念是因为 obj1 比 obj2 轻。obj1 的重量由它的密度和体积值推出。obj2 的重量则根据指定 ENDTABLE 缺省重量的规则推出。解释结构每一分枝的终节点（叶节点）都是满足可操作准则的表达式(如 Volume(obj1, 1), Less(1, 5)等)。从解释结构中可知，与目标概念相关的特性是：Volume(obj1, 1), Density(obj1, 1) 和 ISA(obj2, ENDTABLE)。

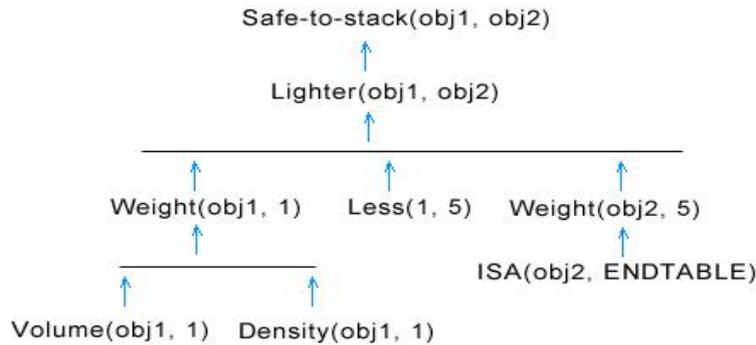


图 17.4.7 Safe-to-stack 的解释结构

第二阶段(即泛化阶段)，在图 17.4.7 给出的解释结构中对目标概念 Safe-to-stack(x, y) 进行回归(图 17.4.8)。这是一个自顶向下地遵从解释结构去逆向应用推理规则的过程，使目标概念回归到能推出它的泛化的(常量变换为变量)初始条件(相应于解释结构中的叶节点)。第一步，遵从导出图 17.4.7 顶节点的推理，目标概念 Safe-to-stack(x, y) 用规则 $Lighter(p1, p2) \rightarrow Safe-to-stack(p1, p2)$ 进行回归，得知 Lighter(x, y) 是推导 Safe-to-stack(x, y) 的一个充分条件。类似地，遵从导出 Lighter(obj1, obj2) 的推理，可从 Lighter(x, y) 进行回归，得到表达式 $Weight(x, w1) \wedge Weight(y, w2) \wedge Less(w1, w2)$ 。紧接着，Weight(x, w1) 通过规则 $Volume(p, v) \wedge Density(p, d) \rightarrow Weight(p, v*d)$ 回归为： $Volume(x, v) \wedge Density(x, d)$ 。同样，Weight(y, w2) 通过规则 $ISA(p2, ENDTABLE) \rightarrow Weight(p2, 5)$ 回归为 ISA(y, ENDTABLE)。

至此，回归过程已到达图 17.4.7 解释结构的所有叶节点。鉴于回归过程中产生的变量置换是 {x/p1, v*d/w1, y/p2, 5/w2}，Less(w1, w2) 转变为 Less(v*d, 5)。所以，可以建立目标概念 Safe-to-stack(x, y) 的一个特化描述，如下： $Volume(x, v) \wedge Density(x, d) \wedge ISA(y, ENDTABLE) \wedge Less(v*d, 5)$

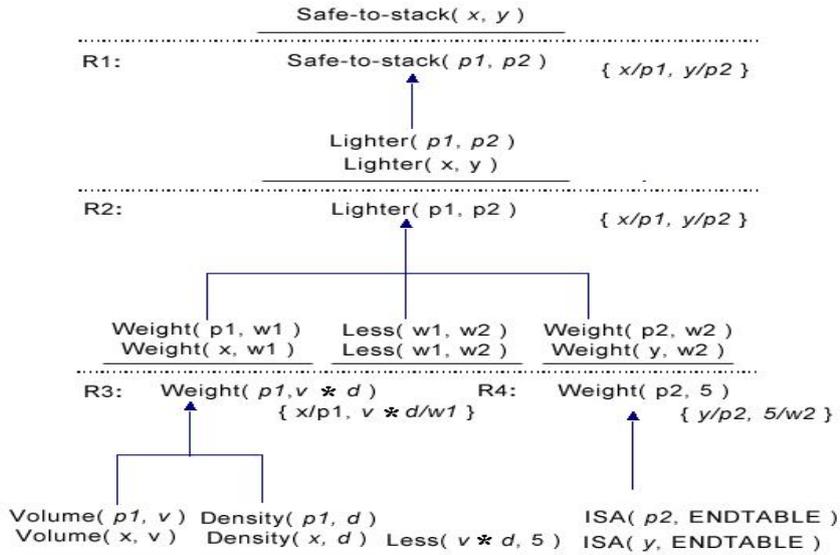


图 17.4.8 对图 17.4.7 Safe-to-stack 解释的泛化

显然，该描述满足可操作准则，且是目标概念 Safe-to-stack(x, y) 的充分解释，因为初始的目标概念是这个特化描述的推理结论。

从上面的论述中，我们可以看到 EBG 是领域理论引导的对训练例子的泛化过程。这也强调了 EBG 的一个重要特性，即学习活动特别依赖于学习程序已经知道了什么。因而 EBG 方法对它的领域理论是高度依赖的，即 EBG 方法是依赖领域理论中的知识对例子进行解释的。领域理论中知识结构的缺陷有可能导致解释失败，从而 EBG 过程失败。这些知识结构的缺陷包括规则矛盾、规则遗漏（不完全）等。这种有缺陷的领域理论又称为不完善领域理论。另一方面，EBG 又是例子指导的对目标概念的特化过程，其按可操作准则的形式重新表示目标概念，将已有的不实用的关于目标概念的最初描述转化为实用的描述形式。因而可操作准则是区分作为学习结果的概念描述与概念初始描述的关键特性，即是学习程度的重要指标。在 Safe-to-stack(x, y) 例子中的可操作准则是静态（即不随系统性能的改善而变化）和离散的（即只将表达式分为可操作和不可操作二种）。实际上，由于 EBG 学习的目的是为了 提高利用（识别）目标描述的效率，而这种效率的判别标准就是可操作准则，因而可操作准则可以随系统性能的提高而变化。

2 基于解释学习的可操作性

基于相似的学习是在概念空间中，搜索一适当概念，使之满足一致性（不覆盖任何反例）和完全性（覆盖所有正例）；而基于解释的学习则在概念描述空间中搜索概念的可操作描述。

假定所有的实例构成一个实例空间 I，概念则对应于实例空间 I 的子集（代表概念的所有实例），因而概念空间是 I 的幂集， $C=I^N$ ；概念描述是概念的谓词逻辑表示，概念描述空间则是由概念空间中每一概念的概念描述组成的集合，概念与概念描述有一对多的对应关系。同一概念可有不同的概念描述，不同的概念描述对于同一应用系统（如解释系统）有不同的使用效率。可操作准则则用于评价使用概念描述的有效性。假如我们简单地把对于概念描述的评价区分为可有效使用的（可操作的）和不可有效使用的（不可操作的），就可以把概念描述空间划分为不可操作的描述空间和可操作的描述空间，如图 17.4.9。概念 c 包含 I1、I2 等实例，D1、D2 是对同一概念 c 的不同描述，D1 为不可操作的描述，D2 为可操作描述。

在图 17.4.9 中, D1 是领域理论中关于目标概念的最初描述, 它由于过于泛化而是不可操作的; D2 则是要学习到的关于目标概念的特化描述。基于解释学习的过程可看成是在描述空间 D 中的搜索过程, D1 是搜索的起点, D2 是目标节点, 解释和泛化是搜索方式, 而可操作准则作为搜索终止的标准。

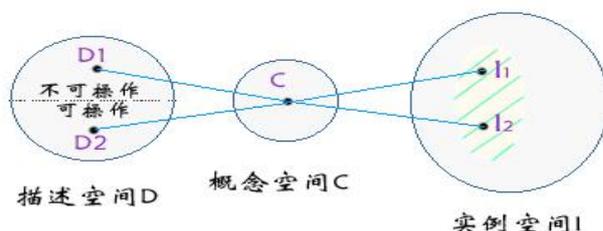


图 17.4.9 概念和概念描述间的关系

在早期的基于解释的学习中, 对可操作准则的定义仅是对实例的有效识别; 后来发展成为对系统性能的提高, 即要求描述满足以下二点: 一是能用性, 能够被学习系统用来识别所描述概念的实例。二是效用性, 当学习系统使用该描述时, 系统的执行性能应该得到改善。对实例的有效识别强调的是描述的能用性, 而效用性强调的则是描述能否有助于性能的改善, 即要求学习到的描述确实有用。

不同的学习系统可有不同的可操作准则。我们可用以下三点来度量一个可操作准则的特性: ① 可变性: 是否能随着时间的变化而变化, 即随着学习的不断深入, 执行环境的变化, 描述的可操作性能否发生变化。② 粒度: 可操作性的度量值是离散的还是连续的。典型的离散值是二值的, 即不可操作的和可操作的。③ 确定性: 对可操作性估价的可靠程度, 即能不能保证对描述作出的估价一定是正确的。

3. 不完善领域理论的完善

对于基于解释的学习, 能否从例子得出一个合理的解释很大程度上依赖于领域理论是否完善。然而, 在复杂的实际领域中, 往往难以构造出一个完善的领域理论。这就要求我们的学习系统有能力自动检测、改正不完善理论或有方法弥补领域理论的不足。

不完善的领域理论大致分为以下三种: • 不完整的理论: 由于缺少一些公理或规则, 某些演绎不能完成。• 不一致的理论: 不一致的结论可从该理论中推导出。• 不可控的理论: 演绎由于受到计算复杂性的限制而不能完成。当然, 这三种不完善性不是相互独立的, 它们之间互相有影响。如理论不完全, 则缺乏必要的信息, 可能会带来理论的不一致性; 为了使领域理论可控而加入的抽象和近似知识, 又可能导致理论的不一致和不完全性。

17.4.4 进化学习 (附: 基于遗传算法的学习系统)

这里, 进化学习即基于进化论的一类机器学习算法, 包括遗传算法(GA)、进化策略(ES)和基因编程(GP)等。遗传算法涉及二进制、整数、实值和基于顺序的交叉策略以及突变操作; 进化策略着重于参数设置和相关突变; 基因编程则关注非线性基因长度和重组突变。

一般认为, 强化学习算法(RL)和进化(学习)算法(EA)都是机器学习领域中独具特色的算法, 虽然它们都属于机器学习的范畴, 但在问题解决的方式和理念上存在明显的差异。

强化学习是目前最流行的一种机器学习方法, 其核心在于智能体与环境互动, 通过尝试和错误来学习最佳行为策略, 以最大化累积奖励。强化学习的关键在于智能体不断尝试各种行为, 并根据奖励信号调整其策略。通过与环境的交互, 智能体逐步优化其决策过程, 以达到既定的目标。强化学习中的主要组成部分包括环境、智能体、状态、动作和奖励信号。常见的强化学习算法包括 Q-learning、DeepQ-Networks(DQN)、PolicyGradient 等。

进化(学习)算法是受生物进化理论启发而设计的一类优化方法,它模拟自然选择和遗传机制来解决问题。这些算法通过对群体中个体进行变异、交叉和选择,逐步优化解决方案。这种方法在处理复杂问题时常常表现出色,因为它允许在解空间中进行全局搜索,找到最优解决方案。通过模拟进化过程,进化算法能够不断改进和调整候选解决方案,使其逐步逼近最优。

强化学习和进化(学习)算法代表了两种不同的人工智能模型训练方法,每种方法都有其优点和应用。在强化学习(RL)中,智能体通过与周围环境交互来获得决策技能,以完成任务。它涉及代理在环境中采取行动,并根据这些行动的结果以奖励或惩罚的形式接收反馈。随着时间的推移,智能体学会优化其决策过程,以最大化奖励并实现其目标。强化学习已在许多领域得到有效应用,包括自动驾驶、游戏和机器人技术。而进化算法(EA)是受自然选择过程启发的优化技术。这些算法通过模拟进化过程来工作,其中问题的潜在解决方案(表示为个体或候选解决方案)经历选择、复制和变异,以迭代地生成新的候选解决方案。EA 特别适合解决具有复杂和非线性搜索空间的优化问题,而传统的优化方法可能会在这些问题上遇到困难。

在训练 AI 模型时,强化学习和进化算法都有独特的优势,并且适用于不同的场景。强化学习在环境动态且不确定且无法预先知道最优解的场景中特别有效。例如,强化学习已成功用于训练智能体玩视频游戏,智能体必须学会驾驭复杂且多变的环境才能获得高分。而进化算法擅长解决搜索空间巨大、目标函数复杂且多模态的优化问题。例如,进化算法已用于特征选择、神经网络架构优化和超参数调整等任务,由于搜索空间的高维度,找到最佳配置具有挑战性。

在工程实践中,强化学习和进化算法的选择取决于各种因素,例如问题的性质、可用资源和所需的性能指标。在某些情况下,两种方法的组合(称为**神经进化**)可充分利用 RL 和 EA 的优势。**神经进化**涉及使用进化算法进化神经网络架构和参数,同时使用强化学习技术对其进行训练。

本节,我们仅考虑基于遗传算法的学习系统。有些内容可能已经过时,仅供参考。

附: 基于遗传算法的学习系统

遗传算法(Genetic Algorithm, 简称 GA)是一种基于进化论优胜劣汰、适者生存的物种遗传思想的搜索算法,由美国密执根大学的霍勒德(J. H. Holland)于 1970 年代初提出。遗传算法作为一种解决复杂问题的崭新的有效优化方法,在机器学习、模式识别、图像处理、软件技术等领域得到了广泛的应用。遗传算法在机器学习领域中的一个典型应用就是利用遗传算法技术作为规则发现方法应用于分类系统。

1. 霍勒德的简单遗传算法简介

霍勒德的模拟自然进化过程的遗传算法将个体的集合—群体作为处理对象,利用遗传操作—交换和突变,使群体不断“进化”,直到成为满足要求的最优解。在霍勒德的遗传算法中,采用二进制串来表示个体。考虑到物种的进化或淘汰取决于它们在自然界中的适应程度,遗传算法为每一个体计算一个适应值或评价值,以反映其好坏程度。因而,个体的适应值越高,就有更大的可能生存和再生,即它的表示特征有更大的可能出现在下一代中。遗传操作“交换”旨在通过交换两个个体的子串来实现进化;遗传操作“突变”则随机地改变串中的某一(些)位的值,以期产生新的遗传物质或再现已在进化过程中失去的遗传物质。霍勒德提出的遗传算法也称为简单遗传算法(simple genetic algorithm, SGA),是一种最基本的遗传算法,其基本算法如下。

SGA 以 0、1 组成的串表示问题域中待进化的个体(初始解)。利用遗传操作—交换和突变,SGA 从当前个体的集合—群体的各串中产生下一代群体。这一过程循环进行,直到满足了结束条件(如循环了指定次数或群体性能不再改进)。SGA 的处理过程如下:

begin

1. 选择适当表示，生成初始群体；
 2. 评估群体；
 3. While 未达到要求的目标 do
 - begin
 - 1. 选择作为下一代群体的各个体；
 - 2. 执行交换和突变操作；
 - 3. 评估群体；
 - end
- end

因此，对于一个 SGA 算法来说主要涉及以下内容：编码和初始群体生成；群体的评价；个体的选择；交换；突变；

(1) **编码和初始群体的生成**。遗传算法的工作基础是选择适当的方法表示个体和问题的解（作为进化的个体）。SGA 要求个体均以 0、1 组成的串来表示，且所有个体串都是等长的。实际上，可以任意指定有限元素组成的串来表示个体，而不影响 GA 的基本算法。

对于同一问题，可以有不同的编码表示方法。由于遗传操作直接作用于所表示的串上，因而不同的表示方法对 SGA 的效率和结果都会有影响。从原理上讲，任何取值为整数（或其它有限可枚举的值）的变量，均可用有限长度的 0、1 串来表示，而任何取值为连续实数的变量也均可用有限长度的 0、1 串来近似表示。因此，对任何一个变量，均可在一定程度上用 0、1 串来表示（编码），而当问题的解涉及多个变量时，则可用各变量对应串的拼接（形成一个长串）来表示相应解。

SGA 处理的起始点并非一个个体，而是由一组个体所组成的群体。一般可用随机方法来产生初始群体，当然最好能考虑各个体的代表性和分布概率。

(2) **群体的评价**。对群体中各个体的适应性评价常需直接利用待优化问题的目标函数。这一目标函数也可称为适应函数，个体选择（或再生）过程正是基于这一函数来评价当前群体中各个体的再生概率。

(3) **个体的选择**。选择操作是对自然界“适者生存”的模拟。评价值（目标函数值）较大的个体有较高的概率生存，即在下一代群体中再次出现。

一种常用的选择方法是按比例选择，即若个体 i 的适应值（目标函数值）是 f_i ，则个体 i 在下一代群体中复制（再生）的子代个数在群体中的比例将为： $f_i / \sum f_i$ 。其中， $\sum f_i$ 示指所有个体适应值之和。若当前群体与下一代群体的个数均维持在 n ，则每一个体 i 在下一代群体中出现的个数将是： $n * f_i / \sum f_i = f_i / f$ ，其中 $f = \sum f_i / n$ 是群体评价的平均值。 f_i / f 的值不一定是一个整数。为了确定个体在下一代中的确切个数，可将 f_i / f 的小数部分视为产生个体的概率。如，若 f_i / f 为 2.7，则个体 i 有 70% 的可能再生 $2+1=3$ 个，而有 30% 的可能只再生 2 个。

SGA 采用称为旋转盘（roulette wheel）的方法来产生各个体的再生数。方法是：每一个体均对应于旋转盘中的一个以园点为中心的扇形区域，区域角度为 $2\pi * f_i / \sum f_i$ ，因而，各个体的区域角度之和等于 2π 。然后随机产生一个 0 到 2π 的值，根据该值所对应的区域，再生一个对应个体，直到产生的个体个数达到所需的数目，从而生成下一代的原始群体。这个群体还需进一步应用交换和突变操作。

(4) **交换**。交换是遗传算法最主要的遗传操作，其工作于选择过程结束后产生的下一代群体。交换操作应用于从这一群体中随机选择的一系列个体对（串对）。

SGA 采用的是单点交换。设串长为 L ，交换操作将随机选择一个交换点（对应于从 1 到 $L-1$ 的某个位置序号），紧接着两串交换点右边的子串互换，从而产生了两个新串。当然，并非所有选中的串对都会发生交换。这些串对发生交换的概率是 P_c 。 P_c 为事先指定的 0—1 之间的值，称为交换率。

(5) **突变**。遗传算法的另一种遗传操作是突变，它一般在交换后进行。突变操作的对象是个体（即串），旨在改变串中的某些位的值，即由 0 变为 1，或由 1 变为 0。并非所有位都能发生变化，每一位发生变化的概率是 P_m 。 P_m 为事先指定的 0—1 之间的某个值，称为突变率。串中每一位的突变是独立的，即某一位是否发生突变并不影响其它位的变化。突变的作用是引进新的遗传物质或恢复已失去的遗传物质。例如，若群体的各串中每一位的值均为 0，此时无论如何交换都不能产生有 1 的位，只有通过突变。

在 SGA 算法中，一般采用的群体大小为 30 到 200，交换率为 0.5 到 1，突变率为 0.001 到 0.05。这些参数：群体大小、交换率、突变率，统称为遗传算法的控制参数，应在算法运行前事先设定。当然，已有人研究了控制参数在算法运行中的自适应调整，以提高遗传算法的灵活性。

尽管遗传算法在实际优化问题中取得了很好的效果，目前对该算法尚无一个清楚完整的理论解释。霍勒德的图式理论(schema theory)和戈尔伯格(Goldberg)的积木块假设仅在一定程度上解释了其工作原理。

2. 基于遗传算法的分类系统

遗传算法不仅可作为搜索和优化的一种方法，而且还可作为一种机器学习技术。例如，可以将基于遗传算法的机器学习应用于分类系统。霍勒德等人将分类系统视为一种认知模型，其可在环境中学习一些简单的串规则(string rules)（又称为分类器），以指导系统的行为。一个基于遗传算法的分类系统包含以下三个组成部分：执行系统；评价系统；遗传算法。分类系统的结构如图 17.4.10 所示。

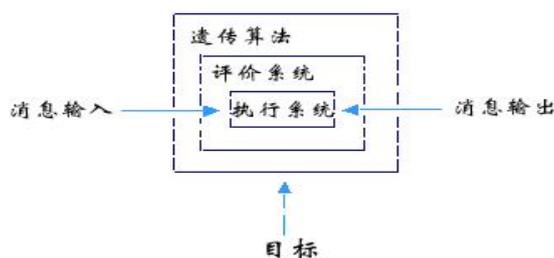


图 17.4.10 基于遗传算法的分类系统的结构

执行系统是最低层的与环境直接交互的子系统，它的作用象一个基于产生式规则的专家系统。每条规则称为一个分类器。但这种规则比较简单，其条件和动作部分都是串，起着传递消息的作用。

分类系统的学习是通过系统从环境中获得反馈信息而进行的，即通过评价分类器(规则)的正确性和效率来实现。这种评价行为由评价系统完成。其中一种有名的评价方法叫组桶式(bucket brigade)算法。

处于最高层的是遗传算法子系统。该子系统产生新的规则去替代系统中效率不高的规则。新规则的产生(发现)方法是利用遗传算法，根据规则的适应度进行选择、组合和替代。

(1) **执行系统**。执行系统实际上是一个简单的产生式系统，产生式规则形如：if <条件> then <动作>。在分类系统里，规则的条件和动作都是串（以便于遗传算法处理）。条件部分的串说明了规则所能匹配的消息集合，而动作部分则说明了规则执行时要发送的消息。

为简便起见，设串长为 k ，由表 $\{0, 1, \#\}$ 中的三种元素组成，其中 $\#$ 表示“不关心”。条件串中的 $\#$ 表示可与 0 或 1 匹配，而动作串中的 $\#$ 表示一种消息传递，即该位的值等于与条件串匹配的消息的对应位值。例如，设 $k=4$ ，有规则为：if $\#10\#$ then $010\#$ 若现有消息 1101，与规则匹配后，该规则将发送消息 0101。

为方便表示，我们可将规则（分类器）的形式改为：〈条件〉:〈动作〉。在匹配规则时，分类系统采用的是并行激活策略，即所有匹配的规则都执行其动作部分，不存在一般产生式系统中的冲突消解问题。当然，若在实际应用中，已激活规则的应用必须是互斥的，或对输出（作为规则执行的结果）的消息个数有限定，那么可依据规则（分类器）的评价值作出选用决定（见以后说明）。

执行系统的结构如图 17.4.11。执行系统基本上重复执行以下步骤：① 将输入的消息放入消息队列中；② 将消息队列中的所有消息与所有分类器的条件做比较；③ 将所有能匹配的分类器所产生的消息放入一个新的消息队列中；④ 用新的消息队列取代老的消息队列；⑤ 将消息队列中的消息加以解释并输出。其中②—④ 步可循环进行，直到没有分类器可与消息队列中的消息匹配为止。

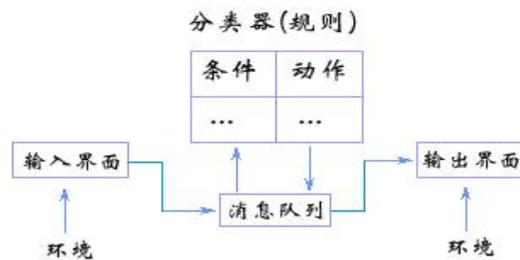


图 17.4.11 执行系统的结构

在传统的产生式系统中，当有多条规则同时能与当前事实匹配时，往往要决定选用哪条已激活规则，即冲突消解。一般的冲突解决方法是对规则的重要性作出区别，且这种区别往往是事先给定的。在分类系统中，虽然可以并行（全部）地执行已激活规则，但为了引入学习机制以不断完善分类规则，就有必要对规则（分类器）在系统运行时的表现做个评价，以支持对它们的选用。

(2) 评价系统的组桶式算法。评价系统的目的是：给每个分类器在系统运行时对执行结果的贡献作个评价（或打分）。实现评价的方法很多，其中最有名的是组桶式算法。

组桶式算法可理解为一个进行情报买卖的交易所，交易人是分类器。分类器作为一个中间商只与“上家”（发送消息使该中间商激活的分类器）和“下家”（其条件部分与该中间商所发消息匹配的分类器）作交易。因此，分类器形成了一条从消息“制造者”（系统输入）到消息“消费者”（系统输出）之间的中间商链。作为中间商，分类器的条件获得匹配时，并不马上发送消息，而是与其它几个同样获得匹配的分类器竞争（并非只有一个获胜）。为了表示分类器的竞争力，我们可给每个分类器分配一个参数值以表示它的价值，不妨称之为“力度”。力度越大的分类器越有竞争性，亦即越有价值。分类器之间的竞争采用竞价体系。每个获得匹配的分类器根据其力度 s 按某种比例出个价 B 。一旦该分类器被激活，其力度将被减去 B ，同时将所出价 B 付给提供消息的上家。当然若该分类器能进一步激活其它分类器（下家），它也同样能从中得到补偿。

当有许多分类器能与当前消息匹配时，也可并行执行这些激活的分类器。此时，分类器力度的主要作用体现在遗传算法子系统：经遗传操作产生的新的分类器将取代力度最弱的分类器。由于有力度值来衡量分类器的价值，因而也可以不采用并行激活策略。例如，只执行有最强力度的已激活分类器，或根据力度大小决定各分类器的激活“程度”，然后执行激活程度高的一个（或几个）分类器。

最后，分析一下分类器的力度在算法执行过程中的变化情况。设分类器 i 在 t 时刻的力度为 $S_i(t)$ ，当它被匹配时，出价为 $B_i(t)$ ，它从下家所得回报为 $R_i(t)$ 。除此之外，它也许

还需要支持一定数额的税 $T_i(t)$ 。因而，该分类器 i 在 $t+1$ 时刻的力度为：

$$S_i(t+1) = S_i(t) - B_i(t) - T_i(t) + R_i(t)$$

为简便起见，设出价数额和所纳税均与力度成正比，即：

$$B_i(t) = C_{bid} \cdot S_i(t) \quad (C_{bid} \text{ 为竞价系数})$$

$$T_i(t) = C_{tax} \cdot S_i(t) \quad (C_{tax} \text{ 为纳税系数})$$

因而，在 $t=n$ 时，有：

$$S_i(n) = (1-k)^n \cdot S_i(0) + \sum_{j=0}^{n-1} R_i(j) \cdot (1-k)^{n-j-1}$$

若 $0 < k \leq 1$ （一般情况可成立），且 $R_i(j)$ 为常数 R_s （即所得回报为恒定值），则有：

$$\begin{aligned} \lim_{n \rightarrow \infty} S_i(n) &= \lim_{n \rightarrow \infty} (1-k)^n \cdot S(0) + R_s \cdot \lim_{n \rightarrow \infty} \sum_{j=0}^{n-1} (1-k)^{n-j-1} \\ &= R_s/k \end{aligned}$$

即，经过足够长的时间 ($n \rightarrow \infty$) 后，分类器 i 的力度将收敛于常数 R_s/k ，即回报值的 k 分之一。

(3) 系统的遗传算法。 尽管组桶式算法提供了一种评价和选取竞争分类器的方法，但还须有一种产生新的更好的分类器的方法，以不断增强分类系统的性能。这种方法就是遗传算法。

由于分类器的条件和动作部分都是固定长度的串，应用遗传算法产生新的分类器就很自然了。设条件和动作串的长度均为 L ，我们可以将条件和动作串的拼接（代表一个分类器），即长度为 $2L$ 的串作为遗传算法的操作对象——个体，而当前分类器的集合作为当前群体，通过遗传算法的三个操作（选择、交换、突变）来产生新的分类器。但分类系统中的遗传算法与前面讨论的简单遗传算法（SGA）还略有不同，主要表现在：

① 群体的变化。在 SGA 中，由于我们是寻求问题的最优解，只要求群体能收敛于最优解，因此产生的新一代群体将完全取代上一代群体。而在分类器系统中，我们的目的是要发现有更高性能的分类器集合，因而有必要保留在上一代群体中已有较好性能（力度值较大）的分类器，亦即只要用近期发现的新分类器取代上一代群体中性能较差（力度值较小）的分类器即可。

② 个体的适应度。SGA 用面向优化的目标函数来评价个体的适应度，而分类系统则用分类器的力度去表示分类器的适应度。对于用交换或变异操作产生的新分类器，力度初始值可有多种计算方法。一种典型的方法是：如果后代从交换中产生，则将两个父代的力度值各减去 $1/3$ ，同时将被减之和作为后代的力度值；如果后代从变异中产生，则将父代的力度值减半，并将被减值作为后代力度值。

③ 突变操作。在 SGA 中，串的元素是 0 或 1，因而突变操作是在 0 和 1 之间转换，而分类器的串是由 $\{0, 1, \#\}$ 三种元素组成，因而某一元素的突变是等概率地被转换为其它两个元素，即 $\{0 \rightarrow 1 \text{ 或 } \#, 1 \rightarrow 0 \text{ 或 } \#, \# \rightarrow 0 \text{ 或 } 1\}$

在分类系统中，遗传算法仅作为系统的一部分功能，且遗传算法的运行穿插在执行系统的循环中。遗传算法的运行时机可以是确定的（即在执行系统运行 t 次后，调用一次遗传算法运算）或随机的（即执行系统平均运行 t 次后，有一次遗传算法调用）；也可以是在某些事件发生后再调用遗传算法（如没有分类器可以匹配或系统性能变差）。

17.5.5 加强学习【强化学习】

1. 加强学习的基本方法

加强学习(reinforcement learning)由于其方法的通用性,对学习的背景知识要求较少,适用于复杂、动态的环境等特点,自上世纪80年代起,就引起了人工智能及机器学习领域学者的广泛注意,并成了机器学习的一种主要方式之一。其最引人注目的成果是应用于分类系统中的组桶式算法以及一系列基于时差(Temporal difference, TD)的学习方法。

许多加强学习方法都基于一种假设,即系统与环境的交互可用一个马尔可夫决策过程来刻画:① 系统和环境可刻画为同步的有限状态自动机;② 系统和环境在离散的时间段内交互;③ 系统能感知到环境的状态,并做出反应性动作;④ 在系统执行完动作后,环境的状态会发生变化;⑤ 系统执行完动作后,会得到某种回报。

马尔可夫决策过程的本质在于:设系统在某个任意时刻 t 的状态为 s ,则事件 t (执行动作 a)发生后转变到某个下一状态 s' 的可能性仅仅依赖于状态 s 和动作 a ,而不依赖于时间和过去事件。“将来”与“现在”有关,而与“过去”无关。

所有的加强学习方法都有一个共同的特点,那就是通过与环境的试探性交互来确定和优化动作的选择,以实现所谓的序列决策任务。在这种任务中,系统通过选择并执行适当的动作,导致环境状态的变化,并有可能得到某种所谓的强化信号(称为立即回报),从而实现与环境的交互。强化信号就是对系统行为的一种标量化奖惩。系统学习的目标是寻找一个合适的动作选择策略,使基于该策略而产生的动作序列可获得某种最优的结果(如积累回报最大)。

在加强学习方法中,强化信号 r 的选取很关键。强化信号是对系统行为的奖惩,因而如何奖惩也必然与问题的性质有关。一般来说,问题的类型可分以下三种:

(1) **纯目标问题**。这类问题将达到某种状态(如八字码问题)或避免达到某种状态(如车辆避撞问题)作为目标,所以奖惩信号往往直到最终才能够确定性地给出,即达到目标状态时给出一个奖励信号,达不到时则给出一个处罚信号。通常,强化信号的设计可以是:目标状态为非零值,其他状态为0。学习的目标是使强化信号的积累最大化。例如,对八字码问题,当达到目标时强化信号为1,其他状态为0;又例如,在倒立摆控制问题中,可认为杆的倾斜带来了一个奖惩信号,强化信号选为-1,其余时候都为0。当然,这些都是典型的滞后评价问题。作为改进的措施,要重点考虑的是如何将强化信号传递到先前的行为中,而不是仅对产生最后状态的行为进行奖惩。例如,随着反复的尝试,我们对杆的直立控制会有进一步的认识。比如,一旦倾斜角度大于某个值,杆必将面临无可挽回的倾倒地局面。这些逐步建立起来的认识,事实上就变成后续学习的新的强化信号,而不是等到杆最后倾倒地。从这一角度看,一个状态/动作序列获得的反馈评价应随着知识和经验的积累慢慢建立起来。这个思想就是基于时差的学习原则:人们在学习中总是试图建立准确的预测关系,对实验结果的预测也会随着实验的增加而收敛。

(2) **受限资源问题**。这类问题不仅仅是要达到目标状态,而且要求使用的资源最少。如要求汽车在最短的时间内达到某种状态,或使用最少的燃料达到目的地。对这类问题,其相应强化信号的设计可以是:将达到目标状态时所用资源的负值作为强化信号值,其他状态为0;学习的目标是使强化信号的积累最大化。当然也可以将达到目标状态时所用的资源值作为强化信号值,其他状态为0;学习的目标则是使强化信号的积累最小化。

(3) **博弈问题**。前面讨论的问题基本上是使强化信号的积累最大化或最小化。对于博弈

问题（双人博弈），学习的目标则是使一方最大化，而另一方最小化，即寻求双策略的鞍点。

强化信号可以从环境的状态中直接获得。例如，倒立摆的角度大于一定值时就可以产生一个失败信号；当机器人与障碍物相撞，即传感器的距离信息小于给定值时都可看作是一个失败信号。另一方法，强化信号也可以从环境的状态信息间接地得到：当环境的状态值达不到预期的要求时，也可以认为产生了一个失败的强化信号。强化信号不但来自环境的状态，而且和主观的目标状态密切相关。

强化信号 r 可以是下列形式中的一种：① 二值， $r \in \{-1, 0\}$ ，这里 -1 表示失败， 0 表示成功；② 介于 $[-1, 1]$ 间的多个离散值，分段表示失败或成功的程度；③ 介于 $[-1, 1]$ 间的实数连续值，更加细致地刻画成功和失败的程度。

从获得的时间，强化信号可区分为立即回报和延时回报。前者意指学习系统执行完动作后立即从环境中获得的回报；后者则意指学习系统在以后的某个时机，将从环境中获得的回报传播给先前的决策步，作为先前决策（动作）的回报。一般来说，立即回报越多，系统的学习速度就越快。一种极端的延时回报情况是：只有在结束时（如实现了目标，或完全失败），才产生回报。对于这种情况，为了提高学习速度，往往可以在中间状态时，通过分析中间状态的情况或靠近目标的程度，产生一些估计性的回报。

在加强学习中的另一个重要概念是状态的值函数。这里，状态的值指示从该状态出发，应用某种策略达到终结状态时，所获得的累计强化信号（或称积累回报），其是立即回报与延时回报之和。状态的值反映了在指定策略下状态的价值（或称效用），而状态的值函数则在一定的策略下将状态映射为状态的值。

一般来说，加强学习的目的是为了给相应的问题寻找一个最优的动作选择策略，即状态到动作的映射，使累计强化信号最大。许多典型的加强学习方法都将上述目标转化为求最优的状态值函数，因为状态的值函数知道了，其相应的动作序列也很容易确定。从而，许多加强学习问题变成了如何设计一种算法，使其能有效地找到最优的状态值函数。

在状态空间元素较少、强化信号确定、状态转化映射确定的情况下，可应用动态规划算法求得各状态的值。但在状态空间很大、强化信号不准确、状态转化映射不确定的情况下，动态规划法就无能为力了。通过与环境的试探性交互，加强学习可以逐步精化的方式求得各状态的值。如前所述，状态的值指示积累回报，但其延时回报部分的确切值在学习过程结束之前是无法获得的。为此，可给各状态设定一个适当（或随意）的初值，并通过不断的试探（动作选取）来修正原有对各状态的估计值，最终获得最优值，从而学习到最优的策略。

目前在加强学习技术中，基于时差(TD)的方法是一类主要的算法。基于时差方法通过预测当前动作的长期影响（即预测未来回报）将奖惩信号传递到先前的动作中，象霍勒德的组桶式算法及在加强学习领域中著名的 Q-学习算法均是基于时差思想的例子。

2. Q-学习方法

沃克廷斯(Waktins)于1989提出了一类通过引入期望的延时回报来求解无完全信息的马尔可夫决策过程类问题的方法，称为Q-学习。Q-学习是一类基于时差策略的加强学习方法。

Q-学习的积累回报函数 $Q(s, a)$ 是指在状态 s 执行完动作 a 后希望获得的积累回报，它取决于当前的立即回报和期望的延时回报。所有状态-动作对的 Q 值存放在一张二维的 Q 表中，其值在每个时步中被修改一次。 Q 表作为动作选择的依据，而动作选择的策略则是选

取当前状态 s 下具有最大 Q 值的动作。该 Q 值也代表了状态 s 的效用 E ，即期望的最大积累回报。因而，有：

$E(s) = \max_a Q(s, a)$ ，状态 s 的效用-期望的最大积累回报；

$T(s, a) \rightarrow s'$ ，指示状态转换， s' 为执行动作 a (s 状态下产生最大积累回报的动作) 后的新状态。

从上述公式可知：一旦知道了各状态-动作对相应的立即回报 r (由应用域给定)，及终极状态的环境回报 (一种简单的处理方式就是成功为 1，不成功为 0)，就可以用求解动态规划的数学模型递推出所有状态-动作对的 Q 值，从而可容易地确定一个最优的关于动作选择的决策序列。但实际上这种递推方式对于大状态空间来说是不现实的。

Q -学习旨在获取优化的 Q 表。学习开始时可随机或按某种策略设置 Q 表中各元素的初始值，以后在问题求解中利用基于时差思想动态地修改 Q 值，直到满足结束条件为止。典型的结束条件是 Q 表收敛 (Q 值的变化小于某个设定值) 或学习的循环次数达到设定值。优化 Q 表的获得相当于找到了最优的状态值函数，其将每个状态映射到该状态下能产生最大积累回报的动作。进而，系统学到了最优的动作选择策略 (基于优化 Q 表的动作选择)。 Q 值的修改方法为：

$$Q(s, a) = (1 - \beta) * Q(s, a) + \beta * (r + \gamma E(s')), 0 \leq \gamma, \beta \leq 1;$$

或表示为： $Q(s, a) = Q(s, a) + \beta * (r + \gamma E(s') - Q(s, a))$

其中， r 是当前的立即回报， $\gamma E(s')$ 指示期望的延时回报。 β 代表学习率，它随学习的进度而逐步变小，直到为 0。 γ 是一个对延时回报 (下一状态 s' 的效用 E) 的折扣因子， γ 越大未来回报的比重越大。实际上， $Q(s, a)$ 的新值就是原 Q 值和新 Q 估计值 (即 $r + \gamma E(s')$) 的组合，组合的比例依赖于学习率 β 。

Q -学习的一般算法如下：

- ① 以随机或某种策略方式初始化所有的 $Q(s, a)$ ；选择初始状态 s_0 ；
- ② 循环做以下步骤，直到满足结束条件：
 - 观察当前环境状态，设为 s ；
 - 利用 Q 表选择一个动作 a ，使 a 对应的 $Q(s, a)$ 最大；
 - 执行该动作 a ；
 - 设 r 为在状态 s 执行完动作 a 后所获得的立即回报；
 - 根据上述方法更新 $Q(s, a)$ 值，同时进入下一新状态 $s' \leftarrow T(s, a)$ 。

注意，该算法往往需跨越多个学习例，而且上述 γ 、 β 都是可调的学习参数。当 β 为 1 时，原有 $Q(s, a)$ 的值对新 Q 值没有任何影响，学习率很高，但容易造成 Q 值不稳定；若 β 为 0，则 $Q(s, a)$ 保持不变，学习过程停止。另外， Q 的初值越靠近最优值，学习的过程会越短，即收敛得越快。对于 Q -学习，有一个很好的理论结果，即 Q -学习可最终收敛于最优值。

下面让我们来看一个简单的例子—— 4×4 棋盘上单一棋子的移步问题。依据棋子在棋盘中的位置 (格子)，该问题共有 16 个状态，就以棋子所在格子指示相应状态 (图图 17.4.12)。棋子的动作为在棋盘内向相邻的格子上移、下移、左移或右移；问题的目标是棋子以最少的移步从初始位置到达棋盘的左上角或右下角；问题的初试状态 (初始位置) 可任意。图 17.4.12 中的数值是为该问题各状态设置的立即回报，即若到达目标状态回报为 0，其它状态回报为 -1。因而，该问题就转化为求一个动作序列，使其积累回报最大的问题。

0	-1	-1	-1
-1	-1	-1	-1
-1	-1	-1	-1
-1	-1	-1	0

图 17.4.12 一个 16 状态的决策问题及回报值设置

若用 Q-学习方法求解上述问题，可设置一个 16*4 的矩阵作为 Q 表，将其初试化（如全部合法操作所对应的矩阵元素均设为 1，其余为某个负大数），并设计相关参数（如 γ 、 β 等）。然后，随机生成一些试探例子（即随机生成初始状态），按照前面的算法用 Q 表的值选择相应动作，且同时修改 Q 值。由于一开始 Q 值大都一样（为 1），此时的搜索策略实际上是一种随机策略。经过若干个例子的运行后，Q 表将逐步收敛，此时 $E(s)=\max_a Q(s, a)$ 基本形成图 17.4.13 所示的分布，其中 $A_0>A_1>A_2>A_3$ （ A_i 指示 $E(s)$ 值）：

A0	A1	A2	A3
A1	A2	A3	A2
A2	A3	A2	A1
A3	A2	A1	A0

图 17.4.13 Q 表收敛时， $E(s)$ 的情况

$E(s)$ 实际上反映了相应状态 s 的期望价值（即估计的积累回报）。逐步收敛后的 Q 表将表示如图 17.4.14 所示的最优决策策略：

0	←	←	↙
↑	↗	◆	↓
↑	◆	↘	↓
↙	→	→	0

图 17.4.14 最优决策策略

图中，↑表示处于该状态时最优动作为上移，↙表示最优动作为下移或左移，◆表示最优动作为四个方向移动均可。之所以某个状态下（格子中）的棋子可以有多个最优动作，是因为相应于这些动作的 Q 值十分接近（也符合问题的特征）。

3. 关于加强学习的进一步讨论

加强学习方法除了作为一种研究机器学习方法的重要理论工具外，同时也在一些实际问题中得到了应用，如博弈问题、机器人控制问题和互联网信息搜寻等。尽管加强学习方法的优点很多，但也存在一些问题，如：① 概括问题。典型的加强学习方法，如 Q-学习，都假定状态空间是有限的，且允许用状态-动作表记录其 Q 值。而许多实际问题，往往对应的状态空间很大，甚至状态是连续的；当然也有可能状态空间并不大，但动作很多。另一方面，对有些问题，不同的状态可能具有某种共性，从而对应于这些状态的最优动作是一样的，或不同的动作可能具有某种联系从而要求连续地应用于某类状态。因而，在加强学习方法中研究状态-动作对的概括表示是很有意义的。为了进行概括，传统的泛化学习方法可以得到应用，包括基于例子的归纳方法（如 ID3）、基于神经网络的学习方法等。② 动态和不确定

环境问题。加强学习通过与环境的试探性交互，获取环境状态信息和强化信号来进行学习，使得能否准确地观察到状态信息成为影响系统学习性能的关键。然而，许多实际问题的环境往往含有大量的噪声，无法准确地获得环境的状态信息，若不抑制噪声，就可能无法使加强学习算法收敛，如 Q 值摇摆不定。③ 收敛问题。当状态空间较大时，算法收敛前的实验次数可能要求极多。④ 多目标的学习问题。大多数加强学习模型仅针对单目标的问题学习相应的决策策略，难以适应多目标、多策略的学习需求。可以把多目标组合成为一个单目标，但回报可能只在问题求解结束后才有，因而将影响多目标问题的学习速度；一种解决多目标问题的方式是将问题空间分解为若干小空间，在每个小空间中用加强学习方法求解子问题，然后再进行合成。⑤ 适应性问题。许多问题面临的是动态变化的环境，其问题求解目标本身可能也会发生变化。一旦目标发生变化，已学习到的策略有可能变得无用，整个学习过程又要从头开始。为适应动态变化的环境，出现了两类加强学习的方法：基于值函数空间的方法和基于策略空间的方法。前者并不显式地表示策略，而是通过学习求得各状态（动作）的积累回报，以时差法为代表；后者则显式地表示策略，并通过搜索操作修改这些策略，以遗传算法为代表。

17.4.6 基于范例的学习

基于范例的推理（Case-Based Reasoning, CBR）是指利用过去经历的典型事例（称为范例）求解或理解当前问题。这推理形式在实现生活中非常常见。例如，有经验的建筑设计师在设计新的建筑结构时，往往会回想起以往类似的例子。在日常活动安排及其它许多方面都存在类似情况，即处理问题时不是从头开始考虑各种细节及其关系，而是依据过去典型的事例，做适当调整以处理当前问题。因而基于范例推理又被称为“即时推理”（instant reasoning），特别适合于知识缺乏或知识太复杂而经验又相对丰富、稳定的领域。

基于范例的推理是一种类比推理方式。与一般的类比推理相比，基于范例推理有以下特点：

- 作为过去经验的范例一般比较固定的表示结构，通常用框架形式表示；
- 欲求解的问题与范例中的问题同属于一个领域，且一般是同性质的，即是两类同性质问题的类比。

基于范例的推理不仅是一种有效的推理方法，也可用于建立一种很好的机器学习方法——基于范例的学习（Case Based Learning, CBL），其学习能力主要表现在：

- 通过记忆和调整老问题的解，使得新问题的求解不必从头做起，因而推理更有效率。
- 通过记忆更多的正、反范例，使得系统的推理能力更强。
- 通过对范例库中同类范例的归纳，可抽象出更一般、有用的结论。

以下，若无特别指明，将对基于范例的推理和基于范例的学习不加区分。

1. 基于范例推理的过程

与一般的推理方式相比，基于范例的推理主要是回忆一个或几个具体事例（即范例），进而通过新、旧问题的对比，提出解决新问题的方案。科洛德诺（Kolodner）将基于范例推理的一般过程细分为以下六个步骤：范例检索、提出初始解、调整/证明（adapt/justify）、评论、评价和范例存贮。以上六个步骤包容了一般类比推理过程的四个构成阶段：选择、映射、评价和巩固，即范例检索相当于类比源的选择过程；提出初始解和调整/证明过程相当于映射过程；评论和评价相当于类比的评价过程；而新范例的存贮则相当于类比推理结果的巩固。调整/证明过程针对范例推理的两种形式：解答改编、问题解释，前者强调通过改编

范例提供的解答来获得新问题的解答，后者则应用范例来分析和理解新问题。图 17.4.15 表示了基于范例推理的过程，其中评论和评价的结果有可能导致对解答改编环节的修补。下面从问题求解的角度，简述这 6 个步骤。

- 范例检索：即从范例库中检索出一个或几个与新问题最相似的范例。新问题与范例的相似性往往取决于两者主要特性的比较，也可以在适当的抽象层次上通过间接比较(如根据所属类别)来决定。这一过程涉及的另一问题是范例库的组织，即如何组织范例并提供适当索引，以便快速、准确地检索范例。

- 提出初始解：即从检索出的范例中找出与新问题相关的部分，以形成有待于进一步改编的初始解。特别是范例涉及范围比新问题范围更广时，此步比较有用。

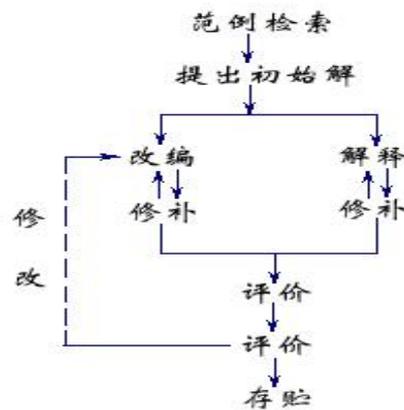


图 17.4.15 基于范例的推理过程

- 改编/解释：将初始解改编为适合于新问题的解答，或用初始解分析和理解新问题。这是基于范例作推理的关键步。这一步的基础工作是比较范例与新问题的差异，以支持解答改编和问题解释。

- 评论：在把推理结果（改编后的解答或对新问题的解释）拿去实际应用前进行评论。一种做法是将所得结果与范例库中的其它近似例子（包含正、反例）做比较，以发现是否存在问题。另一种做法是提出一种假想（模拟）环境直接测试所得结果。不理想的评论结果可能导致对解答改编环节的修补。

- 评价：通过实际应用后，从环境反馈中分析、评价推理结果。比较实际结果与期望结果，若它们间有差异，则试图解释这种差异，进而对解答改编环节做出相应的修补。

- 结果存贮，即把推理结果作为一个新范例存入范例库中，供以后使用。所需存贮的内容（范例）一般包括：问题描述、相应解等等。这一过程需考虑的首要问题是如何建立新范例的索引。同时，不合适的存贮结构和索引方式也可能在这一步加以修改。

基于范例作推理的质量依赖于以下四个因素：① 系统所具有的经验，即范例库的内容。范例库所具有的范例越多、覆盖面越广，有利于推理质量的提高。② 应用范例理解当前问题的能力，取决于能否从范例库中找到最合适的范例，以及对于范例与新问题的差异分析。③ 解答改编的灵活性，即能否有效地将范例提供的解答改编为符合新问题的解答。④ 推理结果的评价能力。高质量的范例推理应能善于从环境的反馈中评价推理结果，并依据不足之处对解答改编环节做出相应修补，使以后的推理能力更强。

传统的专家系统采用的推理方式一般是基于规则的推理（Rule-Based Reasoning，即 RBR），同基于规则的推理方法相比，基于范例的推理方法有以下优点：① 比单纯的 RBR

更接近于人类的决策过程，是一种自然的方法。专家解决问题时，总是试图回忆曾遇到过的类似问题，并借助以往的解决办法来求得新的解决方法。② 范例库比知识库容易构造。应用领域总会有些解决问题的先例，这些先例可以作为范例库的“种子”。许多领域往往已有这些先例的成文材料，稍加整理即可利用。同时，范例是相对独立的，每个范例均有其自身的结构完整性，相互间没有依赖关系。而规则库的建造有赖于知识工程师从领域专家那里收集、整理和编码规则，这是一项繁重而费时的工作。③ 范例库比规则库容易维护，更具灵活性。范例的相对独立性使得增减一个范例不会影响其它范例的存在。在规则库中，一条规则的增删可能引起规则库的一致性、完全性问题。因而，对大型规则库的维护工作十分困难。④ CBR 比 RBR 有更快的执行速度。RBR 是一种链式推理，简单的推理可能触发多条规则，而链环的检测更是费时。CBR 不同，其推理只涉及与当前问题相关的若干范例，改编、评价、修补等只围绕有限的范例进行，加上硬件发展和并行算法的实施，可使范例检索非常迅捷。这就像一“知道”答案的专家和一个需要“想一想”的专家之间的差异。遇到过类似问题的专家有足够的经验将推理快速聚焦到问题的可能解答；而缺乏这种经验的专家则被迫在大的问题空间中寻找可能解。因此，有人称 CBR 为一种即时推理。⑤ 拥有学习能力。CBR 能够自动地将新问题的解决（无论成功或失败）作为范例加入范例库，从而使系统的“经验”不断丰富，求解问题的能力逐渐增加。失败的范例能使系统避免重犯过去的错误。更进一步地，借助其它机器学习技术，可从各种范例中抽象出一般的原理和方法，使知识获取的自动化成为可能。

当然，CBR 的问题求解性能和效率依赖于范例库的覆盖范围、范例检索的合适性和解答改编的可靠性。在许多应用场合下，单纯的 CBR 方法不足以保证系统求解问题的良好性能，往往需要 RBR 技术加以补充。

一般认为，CBR 方法适合于缺乏完备和健全的理论，但又可获取丰富经验（范例）的领域；而 RBR 则适合于对领域有充分认识，能以完备和健全的形式表示领域理论的场合。对于难以获取完备和健全的理论，又不便于建立大型范例库的应用领域，或许建立基于规则和范例的混合系统是明智的。

2. 应用实例

下面将简要介绍一个基于范例作推理的专家系统实例——智能饲料配方系统 ICMIX（何钦铭）。该系统以基于范例的推理技术管理、检索和处理典型的饲料配方模型（范例），并辅以规则推理技术，运用经验规则将范例提供的配方模型改编成符合实际的要求。

所谓饲料配方问题就是根据所养对象（动物）的营养要求、各种原料的营养成分及其价格等因素，决定构成饲料所需的原料及其含量（百分比），使得饲料在满足饲养对象营养要求的情况下成本最低。

一般来说，饲料配方问题是列出相应营养和原料的约束方程，并应用数学规划方法（线性/目标规划等）求解相应方程的问题。例如，对于用线性规划求解饲料配方的问题来说，相应的约束方程有两类：①原料约束，即对相应原料所占百分比范围的约束（由上、下限所决定）；②营养成分的约束（称营养指标），即对所关心的重要营养成分所需要的含量给出一个范围（由上、下限决定）。线性规划的目标函数是饲料价格最低。

对于一个配方专家来讲，其经验（技能）主要体现在所列的约束方程是否适当，即所选参加计算的原料是否适当，所定的原料用量和营养指标约束是否适当。一旦这些因素确定，余下的工作就可用通用的线性规划求解程序来完成。

市场上的饲料配方软件从本质上讲只是配方专家的辅助工具,即为配方专家提供一种方便的编辑和管理环境,以便专家根据饲养对象及其它因素确定配方模型:所想用的原料及其各种约束,而后由系统完成规划计算。配方模型的确定需要配方人员的经验和智能。其中考虑的主要因素(问题描述)有:饲养对象(品种)、生产阶段(如几周龄)、环境温度、饲料价格、饲养对象价格等环境和经济因素。这些因素决定了饲料的日粮类型(原料组合方式)和饲养标准(营养指标),而它们两者则构成了配方模型。在特定动物(如鸡)的配方中,存在若干典型的配方模型,可用以建立范例库。在具体的饲料配方过程中,配方人员可根据问题描述的差异,依经验对这些范例提供的配方模型进行调整(改编)。因而,完全有可能利用基于范例的推理技术,根据用户给出的关于饲养对象的品种、环境和经济因素等描述,自动确定适合用户的配方模型,进而运用最优规划方法求得最优解。

ICMIX 智能饲料配方系统就是应用 CBR 技术(辅以规则推理)实现的用于蛋鸡饲料配方的智能系统。以下仅叙述与 CBR 有关的部分。

ICMIX 中的范例就是配方模型(而不是最后配方),由日粮类型子范例和饲料标准子范例所组成。ICMIX 的范例处理过程如下图 17.4.16 所示:



图 17.4.16 ICMIX 中的范例处理过程

(1) 范例检索。即根据用户给出的问题描述(涉及饲养品种、环境和经济等因素)计算出相应的日粮类型和饲养标准(根据经验规则),并以此作为索引从范例库中分别检索出日粮类型子范例、饲养标准子范例和标准问题描述(一般为缺省值)。

(2) 解答改编。比较当前问题描述与相应范例问题描述间的差异,并根据经验规则调整(改编)范例提供的配方模型。日粮类型子范例的调整主要有:原料种类的调整(根据用户不用某种原料的要求,将范例中的相应原料用其它合适原料替代)和原料用量约束(上、下限)的调整。饲料标准子范例的调整主要是对营养指标约束(上、下限)的调整。所有调整都依据领域的经验(启发式)规则。

(3) 优化规划。调整后的范例配方模型用于作优化规划。ICMIX 首先用线性规划(LP)进行计算。若计算成功地结束,所得结果就是最后配方。若求解失败,则应用若干经验规则和失败时的计算数据对原范例的配方模型进行修补,而后再次计算;若再次失败,则应用目标规划进行计算,所得结果就是配方。目标规划必定有解,但其结果不一定满足原料用量约束和价格最低的条件。线性规划失败时,修补所用的规则目前很少,因而修补效果不是很好。但在实际运行时,需要修补的机会并不多。

饲料配方领域是经验知识丰富的领域。人们在实验和实际配方中,总结了大量调整范例配方模型的经验规则,且积累了许多稳定的范例,因而特别适合于应用 CBR 方法。由于初始构造时的范例已基本能(辅以范例配方模型的调整)覆盖蛋鸡配方的各种具体需求,因而在 ICMIX 中没有设置新范例的存贮功能。

值得一提的是,对于调整后的范例配方模型,ICMIX 系统还需先从数学和营养角度做一

些合理性检查,这一过程相当于科洛德诺提议的评论步骤,而用线性规划(LP)求解该模型相当于评价过程。

17.4.7 知识发现与数据挖掘

目前,信息技术和网络技术已经成为两个最重要的科技领域,触及到人类生活的各个方面。全世界的数据库和因特网正以极快的速度增长。虽然简单的数据查询可以满足某些低层次的需求,但人们更为需要的是能从大量数据资源中挖掘出对各类决策有指导意义的知识。数据的急剧膨胀已远远超过了人们手工处理的能力,人们迫切需要高性能的自动化数据分析工具,以高速、全面、深入、有效地加工数据。知识发现(Knowledge Discovery)与数据挖掘(Data Mining)技术正是在这种背景下产生的,而其中表演重要角色的是机器学习中的发现学习技术。

17.4.7.1 定理发现[知识发现]

发现在科学发展过程中起着举足轻重的作用。例如,在早期的科学研究中,经验公式的发现就占据很大的比例。经验公式的发现可定义为:

给出:一系列相关数据项 x_1, x_2, \dots, x_n 和相应的一组观察值 $(v_{i1}, v_{i2}, \dots, v_{in})$,
 $i=1, 2, \dots, m$,

找出:一个公式 f , 使 $f(x_1, x_2, \dots, x_n) = 0$ 满足这组观察值。

一个典型的发现学习系统是由兰利(P. Langley)等设计的BACON系列。BACON系统试图再现人类在天文、物理、化学等领域发现定理和规律的过程,从而找出发现的机理。系统设计的本意也在于对人类思维的探索。系统就是以英国科学哲学家培根命名的。BACON系统可用不同的描述层次(从低到高)来表示数据,其最低层的是直接观察到的数据,最高层的描述则是较复杂的定律及解释已知数据的假说。这种逐层扩展的表示方式使系统可以把下层发现的假说作为上层观察的数据,并可递归地应用启发式规则。尽管BACON系列软件的使用方法各有差异,但都结合了类似的数据驱动型启发式来指导搜索有趣的定律。其中,BACON4是一个发现经验公式的产生式系统,用于产生不同层次的信息描述。系统采用一组启发式来寻找符号型或数值型数据的规律,即数据中的不变性和趋势性,以期形成假说及按数据规律去定义推理项。BACON4可扩展为数据驱动型通用的发现学习系统。下面,仅就BACON4中假说的形成、推理项的确定及符号型变量固有性质的提出等几方面加以介绍。

(1) **假说的形成。**标准的科学分析方法把世界划分成数据(观察事实)和假说(定律)二部分。假说是对这些数据的解释和归纳。BACON4用一个统一体实现这种二分法,在这个统一体中,信息用不同层次的描述来表示,最低层的描述信息就是数据,而最高层的称为假说,在中间层次的则是这二种概念的混合。因此就某一层描述来说,比它低一层的描述是它的数据,而比它高层的描述是它的假说。

举例来说,现考虑某些服从理想气体定律的数据。这个定律可以描述为: $PV/nT=8.32$, 式中, P 是气体的压力, n 是摩尔数, T 是温度, V 是气体的体积。假设给定BACON4系统的数据项是 $P=1, n=1, T=300, V=2496.0$ 。如果前三项是在系统控制下的变量,则可以把它们的值看成是数据项 V (因变量)取值的条件。现在假定系统又搜集了另一些数据,在这些数据组中, T, n 保持不变,则BACON4发现总有 $PV=2496.0$ 。这就形成了第二层假说,其归纳了所有具有类似条件的第一层观察描述(数据)。同时第二层的假说又可以看成是第三层假说依赖的观察描述(数据)。如果系统又可生成另一个第二层假说(通过变动 T),这二个第二层假说(即描述)结合在一起又可导致第三层描述,可归纳出当 $n=1$ 时,有 $PV/T=8.32$ 。

依此类推，当达到第四层描述时，系统就回到了理想气体定律。

为指导观察描述的逐层归纳，系统采用了经典的归纳推理启发式，如下：

IF 在 L 层上存在一组描述，且这些描述中的因变量 D 具有相同的值 V；

THEN 生成一个 L+1 层的描述，其指出 D 取值为 V，且把 L 级的那组描述中的所有公共条件作为 D 取值为 V 的条件。

这个启发式指导寻找因变量取值相同的一组描述，因变量的值既可为数值也可为符号。BACON4 系统容许待处理的数值数据中包含小的噪声，但不能处理偏离正常规律太大的数据。

(2) **推理项的确定。**在上述的理想气体定律中，被归纳的因变量(V, PV, PV/T)逐步变得复杂。V 用在第一层描述中，PV 用于第二层，而在最后的定律描述中所用的则是复杂的算术组合项 PV/nT。这种由可直接观察的变量（这些变量的值可直接测量到）组合而生成的项（因变量）称为推理项。推理项并不是通过直接观察获得的，而是由计算得来的。推理项的引入，可以明显地简化复杂规律的描述。

系统是如何得到像 PV/nT 这样有用的推理项呢？BACON4 使用称为趋势探测器的启发式搜索方法来搜索推理项空间。这种探测器旨在寻找数值-变量对间的单调上升/下降关系。下面是一个启发式规则：

IF 层的一组描述中，因变量值 a_1 随着变量 a_2 相应值的增加而增加；

THEN 注意 a_1 和 a_2 之间存在的递增关系，并计算 a_1 和 a_2 关系曲线的斜率。

在上述规则中，一旦这种递增趋势被发现，系统就计算这二个变量关系曲线的斜率。如果斜率是常数，则系统在 L+1 层建立二个新的推理项，以定义这二个变量的线性关系。

例如，下表是 BACON4 发现关于行星运动的开普略第三定律中的一个简单例子。这个定律可表示为： $D^3=kP^2$ ，式中 D 为一个物体和它所围绕运行的另一星体的距离，P 是该物体的运动周期。

星体	距离 D	周期 P	D/P	D ² /P	D ³ /P ²
A	5.67	1.769	3.203	18.153	58.13
B	8.67	3.571	2.427	21.038	51.06
C	14.00	7.155	1.957	7.395	53.61
D	24.67	16.889	1.418	36.459	53.89

BACON4 调用上述的启发式，寻到了 D 和 P 的单调趋势关系，即 P 随 D 增大而增大，但相应的斜率项不是常数，而是随 D 的增加而减少。这又导致 BACON4 定义 D²/P，此项的值也不是常数，但随 D/P 减少而增加，结果系统考虑项 D³/P²，这个值接近常数（系统给出了一个允许的误差范围如 7.5%）。BACON4 根据这结果就归纳出该定律了。

一旦一个推理项定义后，它和直接观察的变量就没有区别了。例如，理想气体定律例中，趋势探测器会首先确定如 PV 这样的推理项，并进而确定如 PV/T 那样的推理项。也可以发现这些推理项所取值之间的关系，又从中重新派生出新的推理项，导致对直接观察的变量更为复杂的描述如 PV/nT。BACON4 递归地应用相同的启发式逐步生成更复杂的高层次描述，这种推理能力使系统具备相当强大的搜索经验定律的功能。

(3) **符号型变量固有性质的提出。**虽然 BACON4 的趋势探测器对建立数值型变量间的关系是十分有用的,但是当符号型变量影响数值型变量时,它们就不灵了。例如,在一个电路中接入不同的线圈 X、Y、Z 时,电流发生了变化。这时,系统就要调用另一种启发式规则来假设符号型变量具有某种固有性质(如电导率)。这种启发式如下所示:

IF 在 L 层描述中, a_1 是一个符号型变量, a_2 是一个数值型因变量,且 a_2 值随 a_1 值的变化而变化;

THEN 提出 a_1 具有某种固有性质,其值取 a_2 的值,而且将这些值与 L 层各描述的前提条件相联系。

从上述规则中可以看出,固有性质(作为新的推理项)的值等于数值型因变量的观察值,而且每个值都对应于该变量的观察条件。任何时候,只要这些条件满足就可以重新获得相应的固有性质值。

在确定一个固有性质,并指定其值时,BACON4 又定义了一个新的变量—假想性质:

$$\text{假想性质} = \text{因变量值} / \text{固有性质值}$$

系统在第一次导致建立这个假想性质的观察中,假想性质的值为 1.0(此时因变量值等于固有性质值)。显然,这个值没有任何意义(不引进任何知识),只是一个假设的比较参考值。随着观察条件的改变,假想性质的值也会不同了,这时系统就可以发现新的推理项,以建立新的经验定律。

举例来说,在欧姆定律的发现过程中,二个变量是电池和电线(符号型变量),分别取符号值 A、B、C 和 X、Y、Z 等。电池和电线连成一个简单的电路。单个因变量是电路中观察到的电流 I,它是数值型。下表是在不同的电池和电线组合时观察到的值。

电池	电线	电流(I)	电导率(C)	电压(I/C)
A	X	3.4763	3.4763	1.0000
A	Y	4.8763	4.8763	1.0000
A	Z	3.0590	3.0590	1.0000
B	X	3.9781	3.4763	1.1444
B	Y	5.5803	4.8763	1.1444
B	Z	3.5007	3.0590	1.1444
C	X	5.5629	3.4763	1.6003
C	Y	7.8034	4.8763	1.6003
C	Z	4.8952	3.0590	1.6003

从前三行可看出,电线是影响电流的符号型变量,因而 BACON4 假设一个与电线有关的固有性质 C 存在,并将其取值为电流值。可将这个推理项解释为电线的电导率,或电阻的倒数。随后,系统计算因变量 I 和推理项 C 的比值(I/C),称为假想性质。根据定义,在前三行中其值必定为 1.0。当观察第二个三行的情况时,就有更有趣的发现。这里,电线同上面一样,但是电池变化了,BACON4 把新观察到的 I 值和上次建立的 C 值做比较,发现二者有线性关系,系统就推断电导率仅和所用的电线有关(当然若电池的电阻不能忽略,那么线性关系就不成立),并计算出假想性质 I/C 的新值为 1.1444,而不是 1.0。从最后三行(即又

换一个电池)，可以发现 I/C 的新值为 1.6003。

总结 I/C 的值和其发生的条件，如下表所示：

电池	I/C	电压 V	I/CV
A	1.0000	1.0000	1.0000
B	1.1444	1.1444	1.0000
C	1.6003	1.6003	1.0000

因为 I/C 值随电池的不同而变化，BACON4 又要假设电池的固有性质 V（其值等于 I/C），并定义一个假想性质 I/CV，在当前观察情况下其值为 1.0。

在这个例子中，当 BACON4 改变下一个自变量的值（如电池）时，因变量（电流）的值可以和上一个自变量（如电线）的固有性质（如导电率）比较，如果二组值有线性关系，BACON4 可推断出下一个自变量的变化不会影响上一个自变量的固有性质（即电导率和电池无关）。这就有可能在假想性质（电流除导电率）取新值时，产生新的经验规律，从而导致系统在一个更高的层次上假设一个新的固有性质（把电压和电池联系起来）。相反，如果上述因变量和自变量的固有性质不成线性关系，BACON4 可以推论，因变量的值仅是该固有性质取值的相关条件。

此外 BACON4 在给符合型变量的固有性质赋值时，会检查这些值是否有公共分母（公约数）。如果有公共分母，则将这些值除以公共分母，以产生整数，并将这些整数作为固有性质的值。这时的因变量值和固有性质值之比（即假想性质的值）不是 1.0 而是该公共分母。

17.4.7.2 数据挖掘

数据挖掘是从大量的、不完全的、有噪声的、模糊的、随机的数据中，提取隐含在其中的、人们事先不知道的但又是潜在有用的信息和知识的过程。它通常采用机器自动识别的方式，不需要更多的人工干预。可以说，数据挖掘就是知识发现技术在数据领域中的应用，它可在一个已知状态的数据集上，通过设置一定的学习算法，发掘出数据间隐含的一些内在规律，即获取（发现）所谓的知识。

与数据挖掘密切相关的一个概念是数据仓库 (Data Warehousing)。随着信息化建设的不断深入，数据积累越来越大，信息系统本身的构成也越来越复杂，由此就出现了一些庞大而异构的数据资源。数据仓库就是要将这些相关的数据资源集成起来，以满足决策支持等需求一种技术。数据仓库实质就是一个广义的数据库，但是它存储的数据与普通数据库中的数据不太一样，它存储的是从常规数据库抽取出的经过一定加工整理的数据。例如，对于商业应用来说，原有数据库中存储的是每一笔交易的数据，而数据仓库则要根据已往的历史记录进行提炼整理，存放的可能是某种产品某月在某地区的特定销量等记录。数据挖掘可以在数据仓库上进行操作，作为基于数据仓库的分析工具。

采用数据挖掘技术，可以为用户的决策分析提供智能的、自动化的辅助手段，在多个领域都可以有很好的应用。比如，数据挖掘能解决的典型商业问题包括：客户群体划分、背景分析、交叉销售等市场分析行为，以及客户流失性分析、客户信用打分、欺诈发现等等。目前，数据挖掘的应用市场正在逐渐形成，应用前景十分广阔。

1. 数据挖掘的应用类型

根据应用需求的不同，实现数据挖掘的模型大致有以下几类：① 分类模型。分类模型的主要功能是根据数据的属性将数据分类。在实际应用过程中，数据挖掘的工作旨在通过归纳推理去分析样本数据的分类属性，找出基于这些属性作分类的模式，以建立分类模型。可以利用挖掘出的分类模型来分析其它数据，并预测新数据将属于哪一个类（组）。分类模型应用的实例很多，例如，可以将作为样本（学习例）的银行网点分为好、一般和较差 3 种类型，并以此分析这 3 种类型银行网点的各种属性，特别是位置、盈利情况等属性，找出决定它们分类的关键属性及相互间关系，此后就可以根据这些关键属性对其它银行网点进行分析，以便决定那些银行网点属于哪一种类型。② 关联模型。关联模型旨在发现数据项间隐含的相关性，通常表示为关联规则集，并设置信任级别来度量关联规则的强度。采用关联模型的典型案例是“尿布与啤酒”的故事。在美国，一些年轻的父亲下班后经常要到超市去买婴儿尿布，通过关联规则的挖掘，超市发现了一个规律：在购买婴儿尿布的年轻父亲中，有 30%~40%的人同时要买一些啤酒。超市随后调整了货架的摆放，把尿布和啤酒放在邻近的地方，明显增加了销售额。③ 时序模型。时序模型主要用于分析数据仓库中的某类同时间相关的数据，并发现某一时间段内数据间的时序关系。时序模型可以看成是一种特定的关联模型，即在关联模型中增加了时间属性。④ 聚类模型。在面向数据分类的应用域，会遇到样本数据自身的分类无法预先确定的情况，这时可以采用聚类模型。聚类模型按照某种相近度量方法，将待分析数据分成互不相交的一些分组，称为聚类。每一个聚类中的数据相互接近，不同聚类之间的数据则相差较大。聚类分析是一种实用性很强的技术，其核心就是建立测量数据间相近程度的定量方法。在早期机器学习研究中，聚类分析也是发现学习的一个重要方法。采用聚类模型，系统可以从部分数据发现规律，从而找出对全体数据的描述。

2. 数据挖掘（曾经）采用的典型方法及工具

针对不同应用类型，数据挖掘研究也提出过多种不同的实现方式与算法，并推出了一些相应的商业化软件及工具。其典型的实现方法包括：① **神经网络**。神经网络具有极强的自学习能力，可以对大量复杂的数据进行分析，并完成对人脑或其他系统来说极为复杂的模式抽取及趋势分析。神经网络的典型应用是建立分类模型。② **决策树**。决策树是通过一系列规则对数据进行分类的过程。采用决策树，可以将数据的分类规则可视化，其输出结果也容易理解。例如，在金融领域中可以将贷款对象分为低贷款风险与高贷款风险两类。通过判别对象的一系列属性（形成一个树型结构），确定贷款申请者是属于高风险的还是低风险的。比如说，某个公务员月收入 4000 元，尽管申请“高贷款”，却被认为属于“低风险”人群。作为对照，某个炒股人月收入 10000 元，也将被认为属于“高风险”人群。决策树方法的精确度比较高，不像神经网络那样不易理解，同时系统也不需要长时间的构造过程，因此比较常用。一个典型的决策树学习算法是 ID3 算法。③ **粗糙集技术**。粗糙集理论是一种处理含糊性和不确定性的数学工具，在数据挖掘中发挥了重要的作用，主要用于挖掘关联规则。④ **联机分析处理技术**。联机分析处理(On Line Analytical Processing, OLAP)主要通过多维的方式来对数据进行分析、查询和产生报表。它不同于联机事物处理(On Line Transaction Processing, OLTP)应用。OLTP 主要用来完成用户的日常事务处理，如民航订票系统、银行储蓄系统等等，通常要进行大量的更新操作，同时对响应时间要求比较高。而 OLAP 主要是对用户的当前及历史数据进行分析，辅助决策。其典型的应用包括对银行信用卡风险的分析与预测、公司市场营销策略的制定等。OLAP 往往涉及大量的查询操作，对响应时间的要求不太严格。OLAP 现在主要是作为一种求证性的分析工具，即已有一个假设，通过 OLAP 来得

到验证。OLAP 所采用的验证方法往往基于被称为数据立方体的多维数据模型，即通过对数据立方体的切片、切块、旋转、钻取等操作来实现对数据立方体快速的多维分析。因此，用户可从不同的角度和抽象层次去观察所需的数据，以支持决策的制定。目前常见的 OLAP 主要有基于多维数据库的 MOLAP 及基于关系数据库的 ROLAP。在数据仓库应用中，OLAP 一般是作为数据仓库应用的前端工具，同时 OLAP 工具还可以同统计分析工具及其他数据挖掘工具配合使用，增强决策分析功能。⑤ **数据可视化(Data Visualization)技术**。一些数据仓库系统中包含着大量的数据，并且充斥着各种数据模型，很容易使分析人员变得不知所措。此时，数据挖掘工具和数据的可视化可以很好地相互协作工作。数据挖掘工具允许用户设定富有成效的探索起点，并按恰当的隐喻来表示数据，将数据可视化则可为数据分析人员提供很好的帮助。但它也需要复杂的高性能可视化工具。

作为知识发现的重要工具，数据挖掘技术正处在发展之中。数据挖掘要涉及到数理统计、模糊理论、神经网络和计算智能等多种技术，技术要求较高，实现难度也较大。数据挖掘技术同可视化技术、信息系统、统计分析系统相结合，可较好地发挥其功能与性能。

1. 基于神经网络的数据挖掘

数据挖掘的困难主要存在于三个方面：首先，巨量数据集的性质往往非常复杂，非线性、时序性与噪音普遍存在；其次，数据分析的目标具有多样性，而复杂目标无论在表述还是在处理上均与领域知识有关；第三，在复杂目标下，对巨量数据集的分析，目前还没有现成的且满足可计算条件的一般性理论与方法。在早期工作中，研究者们主要是将符号型机器学习方法与数据库技术相结合，但由于真实世界的数据关系相当复杂，非线性程度相当高，而且普遍存在着噪音数据，因此这些方法在很多场合都不适用。而将神经计算技术用于数据挖掘，将可望借助神经网络的非线性处理能力和容噪能力，较好地解决这一问题。从挖掘出的知识种类来看，目前数据挖掘研究主要着重于关联规则、特征规则、分类规则、聚类规则、时序规则、模式相似性、Web 浏览路径等方面。虽然利用神经计算挖掘关联规则具有较大难度，但其完全可以胜任其他种类知识的挖掘。因此，设计出基于神经网络的数据挖掘方法并将其用于真实世界问题，不仅是可行的，而且也是必要的。

一些研究者指出，将神经计算技术应用于数据挖掘主要存在两大障碍。首先，神经网络学到的知识难于理解。其次，学习时间太长，不适于大型数据集。如果这两个问题得以解决，基于神经网络的数据挖掘将具有广泛的应用前景。

针对上述问题，基于神经网络的数据挖掘主要有两方面的研究内容，即增强网络的可理解性以及提高网络学习速度。目前，前者的解决方案是从神经网络中抽取易于理解的规则，后者的解决方案则是设计快速学习算法。

对于规则抽取，根据设计思想的不同，目前的方法大致可以分成两大类，即**基于结构分析的方法**和**基于性能分析的方法**。基于结构分析的神经网络规则抽取方法把规则抽取视为一个搜索过程，其基本思想是把已训练好的神经网络结构映射成对应的规则。由于搜索过程的计算复杂度和神经网络输入分量之间呈指数级关系，当输入分量很多时，会出现组合爆炸。因此，此类算法一般采用剪枝聚类等方法来减少网络中的连接以降低计算复杂度。在相关的研究中，Gallant 等曾设计了一个神经网络专家系统，并提出了一个简单的规则抽取算法用于解释专家系统所做的推理。该算法通过抽取单个规则来解释神经网络如何为某个给定事例(case)得出结论。其基本思想就是从当前已知的信息集中选择一个能有效地产生该结论的最小信息集合，也就是说，不管其他未知输入分量的取值为多少，只要满足该最小信息

集合的取值要求就可以得出结论。由于该算法非常简单，但只适用于连接权较少的小型神经网络。Towell 和 Shavlik 为基于知识的神经网络 (Knowledge Based Artificial Neural Networks, KBANN) 设计了一种规则抽取算法，即 MoFN 算法。该算法先用标准聚类算法合并 KBANN 中权值接近的连接以创建等价类，并将每个等价类的权值设为该组连接权的平均值，然后去掉那些对结果影响不大的等价类，在不调整权值的前提下对神经网络重新进行训练，最后直接根据网络结构和权值抽取出形如式 $\text{if } (M \text{ of } N \text{ antecedents are true}) \text{ then } \dots$ 的 MoFN 规则。MoFN 规则形式不仅减少了抽取的规则数，还使得规则集比较简单易懂。另外，由于对连接进行了聚类，也使得规则搜索空间大为减少，从而较大地降低了规则抽取的时间开销。由于在普通的神经网络中，连接权大多发散地分布在权值空间中，不象在 KBANN 中那样容易聚为等价类，因此一般来说，MoFN 算法仅适用于 KBANN。为此，Craven 和 Shavlik 提出，可以用柔性权共享 (soft weight-sharing) 方法训练网络，然后用 MoFN 算法抽取规则。由于柔性权共享方法会促进连接权在训练中聚类，这样就使得 MoFN 算法的适用范围有所扩大。但是，由于 MoFN 算法对神经网络的结构有一些很强的要求，例如要求神经元激活值为二值模式、每个神经元表示唯一的概念、网络输入为离散值等，这使其适用范围始终受到很大的限制。

在相关研究方面，R. Setiono 等提出了一种从神经网络中抽取规则的三阶段算法。他们首先用权衰减 (weight-decay) 方法训练一个 BP 网络，该网络中较大的连接权反映了较重要的连接；然后对网络进行修剪，在预测精度不变的情况下删掉不重要的连接；最后通过对隐层神经元的激活值进行离散化，进而为每个输出结点抽取相应的规则。该算法中离散化隐层神经元激活值的处理别具一格，这使其摆脱了很多规则抽取方法对激活值类型的限制，可以处理非二值模式的激活值。但是，由于无法保证网络的功能在离散化处理和修剪处理前后的一致性，因此该算法抽取的规则在保真度上有一定的缺陷。之后，他们又提出了一种适用于三层前馈网络的通用型规则抽取算法。该算法不仅使用了其设计的激活值离散化技术，还使用了一种独特的隐层神经元分裂技术，即当某个隐层神经元的输入连接数较多时，将其分裂为若干个输出神经元，并通过引入新的隐层神经元来构建子网络，从而递归地进行规则抽取处理。该算法可以产生相当精确的规则，但由于要训练多个子网络，其时间开销相当大。另一方面，该算法只适用于规模较小的网络，这是因为在输入神经元较多时，待分裂的隐层神经元数以及递归分裂的次数极大。之后，他们提出了一种从三层前馈网络中抽取倾斜规则 (oblique rule) 的算法 NeuroLinear。与普通的规则相比，倾斜规则通常可以更好地表示边界与属性空间轴非垂直的判定域，从而较大地减少规则前件数。NeuroLinear 首先通过修剪网络去除冗余连接，并对隐层神经元激活值进行聚类以降低组合复杂度。然后用隐层神经元聚类后的离散激活值表示输出层神经元的输出，用输入层神经元的激活值表示隐层神经元聚类后的离散激活值，从而得到层次形式的规则。再对这些规则进行合并，从而得到直接用输入属性表示网络输出的规则。之后，他们又提出了快速规则抽取算法。所谓快速是相对于其他的基于结构的规则抽取算法而言，一般地说，为了避免组合爆炸的问题，大多数规则抽取算法都要对原神经网络进行剪枝操作，去掉一些不重要的连接，但为了保证神经网络的精度，需要对剪枝后的神经网络进行再训练，这增加了算法的开销，降低了算法的效率。为此，Setiono 等还提出了 FERNN 算法，该算法无需对神经网络进行多次训练，可以抽取 MOFN 规则或是 DNF 规则。

与基于结构分析的方法不同，基于性能分析的神经网络规则抽取方法并不对神经网络结构进行分析和搜索，而是把神经网络作为一个整体来处理，这类方法更注重的是抽取出的规则在功能上对网络的重现能力，即产生一组可以替代原网络的规则。在相关研究方面，Saito 和 Nakano 等提出了 RN 算法。该算法先从少数正例中抽取规则，然后根据未被覆盖的正例扩展规则，根据覆盖的反例缩减规则，直到规则覆盖了所有的正例，并且不覆盖任何反例为止。抽取的规则表示为析合范式形式。虽然该算法并不对网络结构进行分析和搜索，但其要搜索正、反例空间，因此该算法在示例空间较大时将面临组合爆炸问题。Craven 和 Shavlik 则将规则抽取视为一个目标概念为网络计算功能的学习任务，是“给定一个训练好的神经网络以及用于其训练的训练集，为网络产生一个简洁而精确的符号描述”，提出了一种基于学习的规则抽取算法。该算法使用了两个外部调用（Oracle），其中 EXAMPLES 的作用是为规则学习算法产生训练例，SUBSET 的作用则是判断被某个规则覆盖的示例是否都属于某个指定类。算法为每个分类产生各自的 DNF 表达式，它反复地通过 EXAMPLES 产生训练例，如果某训练例没有被当前该类的 DNF 表达式覆盖，则新规则被初始化为该训练例所有属性值的合取，然后反复尝试去掉该规则的一些前件，并且调用 SUBSET 来判断该规则是否与网络保持一致，从而使规则得以一般化。该算法不需使用特殊的网络训练方法，也不需将隐层神经元近似为阈值单元，但其计算量较大。Thrun 为前馈神经网络提出了一种基于有效区间分析（Validity Interval Analysis）的规则抽取算法。该算法的关键是为所有或部分神经元找出激活区间，即有效区间。算法通过检查有效区间集合的一致性而不断排除导致不一致的区间。Thrun 描述了两种不同的操作方式，即从特殊到一般和从一般到特殊。前者是从一个随机选择的示例开始，通过不断扩大相应的有效区间，逐渐得到一般的规则；后者则是从一个未加验证的假设集开始，通过有效区间来验证假设集中的规则。利用该算法可以抽取出精度较高的规则，但其以区间形式表示的规则前件使得规则的可理解性较差。另外，由于该算法的计算开销非常大，因此其只适用于对规则进行理论验证，难以完成实际的神经网络规则抽取任务。陈兆乾等从功能性观点出发，提出了一种基于统计的神经网络规则抽取算法 SPT。与其他算法不同的是，SPT 并不在规则抽取开始时离散化所有连续属性，而是仅在离散属性不足以缩小未知属性空间时，才选择一个聚类效果最佳的连续属性进行离散化，这样就大大降低了离散化处理中由于属性空间内在分布特性未知而造成的主观性。除此之外，SPT 采用优先级规则形式，不仅使得规则表示简洁紧凑，还免除了规则应用时所需的一致性处理；利用统计技术对抽取出的规则进行评价，使得其可以较好地覆盖示例空间。SPT 不依赖于具体的网络结构和训练算法，可以方便地应用于各种分类器型神经网络。与其他规则抽取算法相比，网络的各输入分量之间相关度较低时，由于 SPT 独特的离散化机制有助于降低无关属性交互引起的不良影响，因此，SPT 可以取得更好的效果。

基于神经网络的数据挖掘主要是希望借助神经网络的非线性处理能力和连续属性处理能力，这一点在处理回归估计问题时尤为明显。遗憾的是，目前神经网络规则抽取方面的研究几乎都是针对分类器型网络进行的，回归估计型神经网络的规则抽取几乎是一片空白。如果能在后者的研究上取得突破，将极大地促进基于神经网络的数据挖掘的发展。另外，基于神经网络的规则抽取目前主要着重于提高抽取出的规则对网络的保真度，即规则是否可以真实地再现网络的功能。然而，在面向数据挖掘的应用中，规则的可理解性往往更加重要，在一些实际领域中，需要牺牲一定的保真度以获取更好的可理解性。因此，如何在规则的可理解性与保真度之间达成折衷，将是一个有待研究的课题。

目前，基于神经网络的数据挖掘虽然已取得了不少成果，但并没有充分发挥神经计算的能力。进一步扩展其挖掘的知识类型，拓宽应用范围，将是基于神经网络的数据挖掘在将来相当长时间内的重要研究内容。

17.4.7.3 数据库与网络中的知识发现

数据挖掘仅是从数据库或数据仓库中发现知识的一个特定步骤，要形成发现知识的完整过程，数据挖掘还需要有数据收集、数据整理、知识验证等作为前序和后验步骤。另一方面，随着互联网的迅猛发展，网络上的数据已变得极其丰富，数据挖掘技术同样可以推广到网络数据的挖掘和知识发现中。

1. 基于数据库的知识发现

为了获取隐藏在大量数据之后的重要信息，如关于数据整体特征的描述和发展趋势的预测等，人们迫切希望增强对大量数据的自动分析能力。基于数据库的知识发现（Knowledge Discovery in Database, KDD）应运而生。KDD 的研究是信息技术的汇总，它融数据库技术、人工智能技术、数理统计技术和可视化技术为一体，是一个多学科相互交叉融合的一个新兴的具有广泛应用前景的研究领域。

目前，对于 KDD 的研究主要围绕着理论、技术和应用 3 个方面展开。理论方面的研究包括：数据和知识的表示；结构化多媒体数据的模型构造；不确定性的管理；知识发现的理论；数据挖掘的算法复杂性和效率分析；海量数据集的统计研究等。技术方面的研究主要包括数据挖掘的方法、数据挖掘的算法和知识发现的过程等。其中，数据挖掘的算法研究包括空间数据、文本数据和多媒体数据的挖掘算法研究，并行和分布式数据的挖掘技术等。知识发现过程的研究包括数据的预处理技术，如数据去噪、样本选取、数据缩减等。此外，技术研究还包括知识的评估、融合和解释，数据和知识的可视化等。

(1) KDD 可发现的知识模式。KDD 可在数据库中进行以下模式（知识）的抽取：① 依赖关系：若可以从一项数据预测另一项数据，则称这两项数据存在依赖关系。② 分类知识：数据子类的标识知识。子类可以由某一（一些）属性的特定取值确定，也可以由附加的领域知识来定义。③ 描述性知识：关于类别特征的概括性描述。主要包括两类知识：特征描述知识和区分性知识。特征描述知识是本类数据所共有的；区分性知识则指示本类区别于它类的特征。④ 偏差型知识：关于类比差异的描述。包括：标准类中的特例；各类边缘外的孤立点；实际观测值和系统预测值间的显著差异等。

(2) KDD 的处理过程。KDD 是一个多步骤的处理过程。其主要处理步骤如下：① 准备：熟悉应用领域的数据和背景知识。② 确定 KDD 的目标：根据要求，确定所要完成的 KDD 应发现何种类型的知识，并确定知识模式。③ 数据收集和选择：根据知识发现任务从数据库中提取相关的数据集合。④ 数据预处理：对与发现任务相关的数据集合进行再加工，检查数据的完整性和一致性，并消除数据中的噪音。⑤ 确定知识发现方法：根据发现任务的目标和拟发现知识的模式，设计或选择合适的知识发现算法。⑥ 发现知识：调用相应的数据挖掘算法发现所需的知识。⑦ 知识解释和评价：对发现的知识进行解释，并将发现的知识以一种适当的形式呈现给用户。其中包括对知识的一致性、新颖性和有效性的评价。当然，整个知识发现过程也不是简单的线性流程，步骤之间包含着循环和反复。

(3) KDD 系统的结构。一般来说，KDD 系统可视为用户和数据库之间的沟通桥梁，系统根据用户要求重新组织、汇聚海量数据，使用户能通过系统对自己感兴趣的知识有清晰的了解，或由系统应用所提取的知识解决具体的实际问题。图 17.4.17 为 KDD 系统的一个基本框架。

其中，发现控制模块用于初始化系统中的其他构件，控制发现活动的启动、状态转移和终止。数据库接口响应查询请求，执行 SQL 操作，对原始数据进行加工处理。背景知识库是与特定领域相关的背景知识库。汇聚模块通过对用户任务需求的理解，利用各种数据库操作和数据统计操作，完成初始知识模板的生成。数据模式（知识）抽取模块集成了各种数据挖掘算法，对知识发现的状态空间执行各种操作，完成知识发现任务。模式（知识）评价模块依据评价准则，对发现过程中产生的中间模式（知识）集进行筛选。知识验证模块结合各种策略，解决结果知识中的冗余和冲突，达到去伪存真的目的。

系统的信息来源由两部分组成：一方面用户提出发现任务，并直接干预发现活动，输入各种高级控制指令和各种特定领域的背景知识；另一方面数据库管理系统提供数据库中的各类原始数据。系统的知识发现的过程为：从数据库中选择的数据先经过汇聚模块的过滤形成初始知识模板，再经过发现控制模块调用各种数据挖掘算法形成中间模式集。这些中间模式经由评价模块的筛选，形成最终的知识发现结果。最后，知识验证模块对发现结果进行验证，然后送入系统知识库以支持后续的知识发现活动。

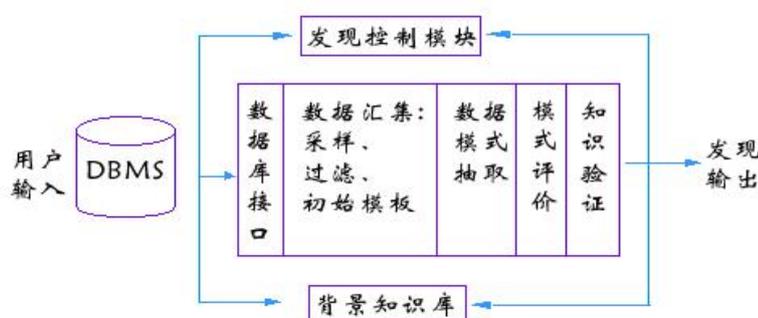


图 17.4.17 KDD 系统的基本框架

2. 面向网络信息的知识发现

面向因特网的数据挖掘可以说是一个比较前瞻性的领域，有人称之为因特网挖掘（Web Mining），已取得了一些令人感兴趣的结果，例如，有系统可用来筛选因特网上的新闻，保护用户不受无聊电子邮件的干扰和促进商业推销等。面向因特网的数据挖掘比面向单个数据仓库的数据挖掘要复杂得多，因为数据库中的数据是结构化的，而因特网上的数据则是半结构化的，这就决定了面向因特网的数据挖掘将是一个颇具挑战性的课题。所谓半结构化是相对于结构化和非结构化而言的。传统数据库中的数据结构性很强，可称之为完全结构化的数据，但同时也还存在一些诸如一本书或一幅图像等完全无结构的数据。但是因特网上存在的数据既不是完全结构化的也不是完全非结构化的，因为网页具有一定的描述层次，存在一定的结构，所以可视为半结构化的数据。

从数据库的角度看，网络上各个网站的信息可看作是一个个特定的数据库，而且是一个个复杂性较高的数据库。这些数据库形成的是异构的数据源，因为每一网站跟其他网站的信息和组织形式都不一样，这就构成了一个巨大的、异构的数据库环境。如果想要利用这些数据进行数据挖掘，首先必须研究各个网站之间异构数据的集成问题。因为只有将这些网站上的数据都集成起来，提供给用户一个统一的视图或视角，才有可能从巨大的数据资源中获取所需的東西。其次，还要解决因特网上的数据查询问题。因为如果所需的数据不能有效地得到，对这些数据进行分析、处理就无从谈起。

传统的数据库都有一定的数据模型，可以根据这个模型来具体地描述特定的数据，同时也可以很好地定义相关的查询语句。而因特网上的数据很复杂，没有特定的模型可用来统一描述，且具有语义隐含和动态可变等一系列复杂特性。为解决异构数据的集成和数据查询问题，迫切需要一个模型来清晰地描述因特网上的数据。针对因特网上的数据半结构化的特点，寻找一个半结构化的数据模型就成为了解决上述问题的关键所在。此外，还需要一项技术能够自动地从现有网络数据中将这个模型抽取出来，这就是所谓的模型抽取技术。半结构化数据模型和半结构化数据模型抽取技术构成了因特网数据挖掘技术的实施前提。

因特网上的内容查询有两个方面，一是页面内容的查询，二是页面与页面之间链接的查询。传统的网页是基于 HTML (Hyper Text Markup Language, 超文本标记语言) 实现的，只提供如何在浏览器中显示信息的描述，而没有反映数据本身所包含的语义，所以要想真正做到准确、高效地查询数据，十分困难。标记语言叫 XML (eXtensive Markup Language, 可扩展标记语言) 的出现和使用已部分解决了此问题。XML 的标签 (Tag) 能指示数据的语义，并可由用户自行定义。从某种意义上说，XML 就是一种半结构化的数据模型，可用于对文档 (数据) 包含的语义作清晰的结构化描述，而且很容易和关系数据库中的属性一一对应起来，从而能够支持对于文档内容的精确查询。

面向因特网数据挖掘的一个例子是：帮助有关网站发现有价值的商业信息。随着因特网技术的发展，各类电子商务网站风起云涌。建立起一个电子商务网站并不困难，困难的是如何让电子商务网站有效益。要想有效益就必须吸引客户，增加能带来效益的客户忠诚度；而为客户提供个性化服务则是在激烈的电子商务竞争中赢得客户的重要手段。电子商务网站每天都可能有上百万次的在线交易，生成大量的日志文件和登记表，显然，对这些数据进行分析 and 挖掘，将有助于充分了解客户的喜好、购买模式，从而设计出满足不同客户群体需要的个性化服务。

在对网站进行数据挖掘时，所需要的数据来自于两个方面：客户的背景信息，主要来自于客户的登记表；浏览者的点击流，可用于考察客户的行为表现。但有的时候，客户对自己的背景信息十分珍重，不肯把这部分信息填写在登记表上，这就会给数据分析和挖掘带来不便。在这种情况下，就不得不从浏览者的表现数据中推测客户的背景信息，进而再加以利用。

就分析和建立模型的技术和算法而言，网站的数据挖掘和常规的数据挖掘差别不大，很多方法和分析思想都可以运用。所不同的是网站的数据格式有很大一部分来自于点击流，和传统的数据库格式有区别。因而对网站进行数据挖掘所做的工作主要是数据准备。

人们关心的另外一个话题是文本数据挖掘。显然，无论是数据结构还是分析处理方法，文本数据挖掘和面向数据库的数据挖掘都相差很大。文本数据挖掘并不是一件容易的事情，尤其是在分析方法方面，还有很多需要研究的专题。

17.5 智能机器人技术

智能系统理论研究各类智能系统的构成、系统中问题与知识的表示方法、系统的感知与控制、系统的基本操作原理与过程、典型的应用领域和应用方法、各种不确定性因素的处理等。在各类智能系统中，智能机器人无疑是最典型的一类。

在当前生产生活中，机器人的应用已越来越广泛，正在代替人发挥着日益重要的作用。机器人技术是综合了计算机、控制论、机构学、信息和传感技术、人工智能、仿生学等多个

学科的成果而形成的高新技术，代表着高新技术发展的前沿，是当前科技研究的热点方向。随着计算机技术、微电子技术和信息技术的快速发展，机器人技术的发展速度越来越快，智能度越来越高，应用范围也得到了极大的扩展。

智能机器人是机器人技术领域的一个重要研究分支，它是一个集环境感知、动态决策与规划、行为控制与执行等多功能于一体的综合系统，是一类能够通过传感器感知环境和自身状态，实现在未知或有障碍的环境中面向目标自主运动，完成各种不同作业功能的机器系统。对不同任务和特殊环境的适应性是智能机器人与一般自动化装备的重要区别。现在，智能机器人在外观上已远远脱离了早期机器人所具有的形状和局限，更加符合各种不同应用环境的特殊要求，其功能和智能程度已大大增强，从而为机器人技术开辟出了更加广阔的发展天地。

智能机器人研究的主要问题包括：领域知识和常识的表达与理解；机构的精细控制；情感的描述与模拟；自然感知能力[视觉、听觉、嗅觉、触觉]的实现和行为的规划等。本书主要介绍机器人的信息感知与行为规划。

17.5.1 智能机器人的信息感知与识别

17.5.1.1 智能机器人信息感知的各类传感器

感知是人与周围世界联系的窗口，它的任务是感知外界环境及人对环境刺激的感受等。同样，机器人感知也是机器人与周围世界联系的窗口，是机器人自主完成各项任务的前提。

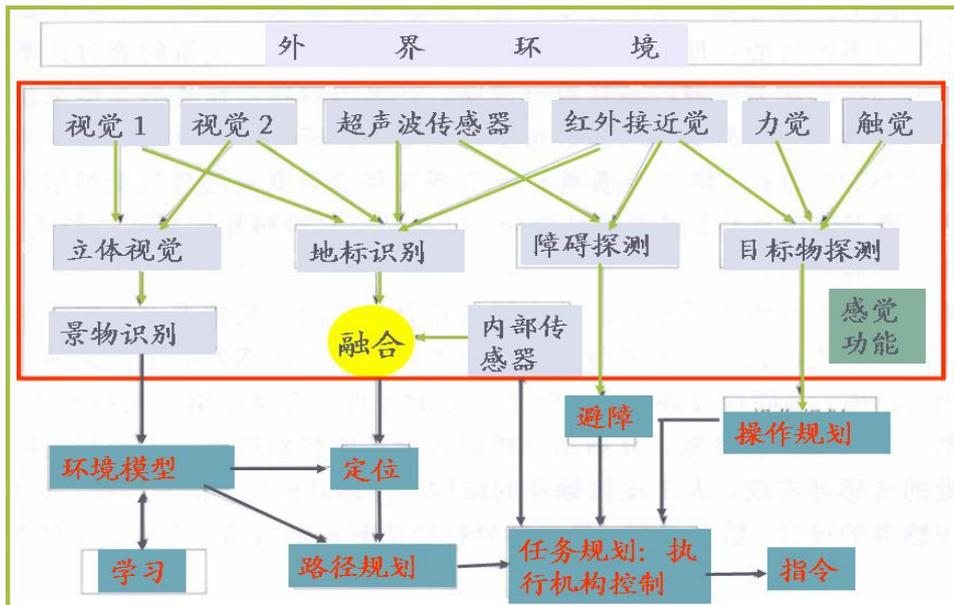


图 17.5.1 机器人内部感知与控制系统示意图

机器人要感知各种信息首先就需要有各类传感器。机器人所用的传感器有很多种，根据不同用途可分为内部测量传感器和外部测量传感器两大类。内部测量传感器用来检测机器人本身和组成部件的内部状态，包括：特定位置、角度传感器；任意位置、角度传感器；速度、角速度、加速度传感器；倾斜角、方位角传感器等。外部传感器用来检测机器人所处环境及状况，包括：视觉（测量、认识）传感器、触觉（接触、压觉、滑动觉）传感器、力觉（力、力矩）传感器、听觉传感器、接近觉（距离）传感器、以及角度（倾斜、方向、姿式）传感器等。

视觉传感器是机器人中最重要的传感器之一。机器视觉从 20 世纪 60 年代开始首先处理积木世界，后来发展到处理室外的现实世界。如今，已有了实用性的视觉系统。视觉一般

包括三个过程：图像获取、图像处理和图像理解。相对而言，图像理解技术还很落后。

作为视觉的补充，触觉能感知目标物体的表面性能和物理特性，如：柔软性、硬度、弹性、粗糙度和导热性等。一般认为，触觉包括接触觉、压觉、滑觉、力觉四种，狭义的触觉按字面上来看是指前三种感知接触的感觉。

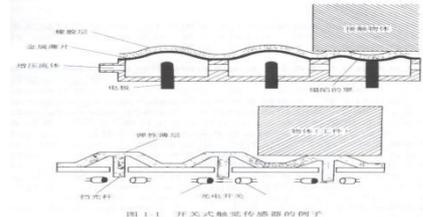
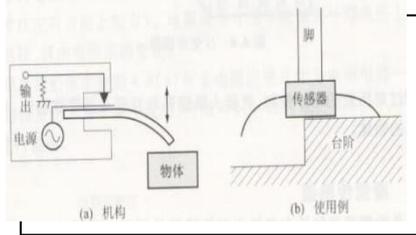


图 17.5.2 机器人力觉传感器示意图

图 17.5.3 机器人力觉传感器示意图

机器人力觉传感器有多种。通常我们将机器人的力觉传感器分为三类：装在关节驱动器上的力传感器，称为关节力传感器，用于控制中的力反馈；装在末端执行器和机器人最后一个关节之间的力传感器，称为腕力传感器；装在机器人手爪指关节（或手指上）的力传感器，称为指力传感器。力觉传感器使用的主要元件是电阻应变片。SRI（Stanford Research Institute）研制的六维腕力传感器，如图所示。它由一只直径为 75mm 的铝管铣削而成，具有八个窄长的弹性梁，每个梁的颈部只传递力，扭矩作用很小。梁的另一头贴有应变片。图中从 P_{x+} 到 Q_{y-} 代表了 8 根应变梁的变形信号的输出。

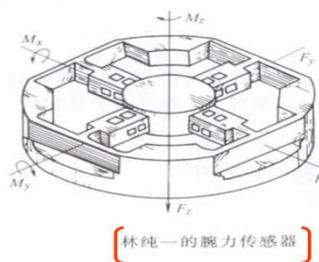
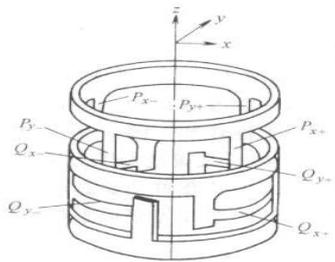


图 17.5.4 机器人腕力传感器示意图

图 17.5.5 机器人腕力传感器示意图

日本大和制衡株式会社林纯一研制的腕力传感器。它是一种整体轮辐式结构，传感器在十字梁与轮缘联结处有一个柔性环节，在四根交叉梁上共贴有 32 个应变片（图中以小方块），组成 8 路全桥输出。

机械手一般采用两种抓取方式：硬抓取和软抓取。硬抓取末端执行器利用最大的夹紧力抓取工件。软抓取（有滑觉传感器时采用）末端执行器使夹紧力保持在能稳固抓取工件的最小值，以免损伤工件。实现软抓取需要有滑觉传感器。

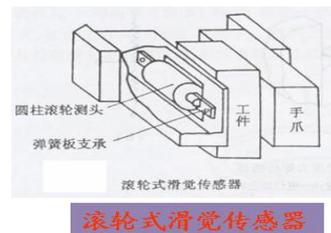
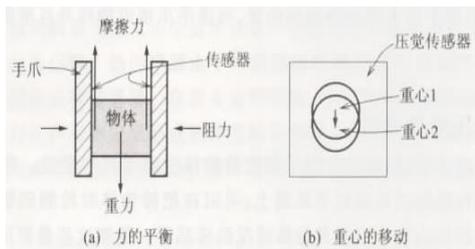


图 17.5.6 采用压觉传感器实现滑觉感知

图 17.5.7 滚轮式滑觉传感器

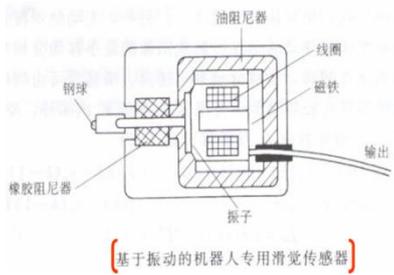


图 17.5.8 基于振动的滑觉传感器

根据振动原理制成的滑觉传感器。钢球指针与被抓物体接触。若工件滑动，则指针振动，线圈输出信号。

机器人的接近觉主要感知传感器与对象物之间的接近程度。研究它的目的是使机器人在移动或操作过程中获知目标或障碍物的接近程度，移动机器人可以实现避障，操作机器人可避免手爪对目标物由于接近速度过快造成的冲击。

智能机器人身上通常配备有多种不同的传感器，如视觉传感器、红外传感器、超声传感器、碰撞检测传感器、触觉传感器、滑觉传感器等。由于受到各传感器的检测对象、工作范围、精度等因素的影响，需要确定不同来源传感器数据的一致性，通过不同传感器信息的互相补充来获得外部完整的信息，所以多传感器信息融合可提高机器人的智能化程度。

17.5.1.2 机器人的视觉与视觉系统

视觉则是人类最重要的感觉，是人类感知外部世界的最重要的工具。机器人视觉是诸多传感信息中包含信息最丰富、最复杂和最重要的感觉，也是应用最为广泛的机器人感觉。

视觉系统是智能机器人的重要感知系统，机器人视觉系统的硬件组成如图所示，它主要由摄像机、图像采集卡和计算机组成。其中最重要的部件是 CCD（电荷耦合器件：charge coupled devices）。CCD 的基本结构是一个间隙很小的光敏电极阵列，即无数个 CCD 单元组成，也称为像素点（如 448×380 ）。它可以是一维的线阵，也可以是二维的面阵。

机器人视觉，即计算机视觉。计算机视觉技术是研究用计算机来模拟人和生物的视觉系统功能的技术，其目标是让计算机能够感知周围视觉世界，了解它的空间组成和变化规律。机器人视觉系统的工作包括图像的获取、图像的处理和分析、输出和显示，核心任务是特征提取、图像分割和图像辨识，而如何精确高效的处理视觉信息是视觉系统的关键问题。目前视觉信息处理逐步细化，包括视觉信息的压缩和滤波、环境和障碍物检测、特定环境标志的识别、三维信息感知与处理等，其中环境和障碍物检测是视觉信息处理中最重要、也是最困难的过程。机器人视觉是其智能化重要的标志之一，对机器人智能及控制都具有非常重要的意义。

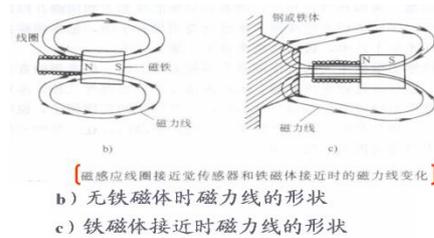


图 17.5.9 接近传感器

【磁感应线圈接近传感器和铁磁体接近时的磁力线变化】
b) 无铁磁体时磁力线的形状
c) 铁磁体接近时磁力线的形状

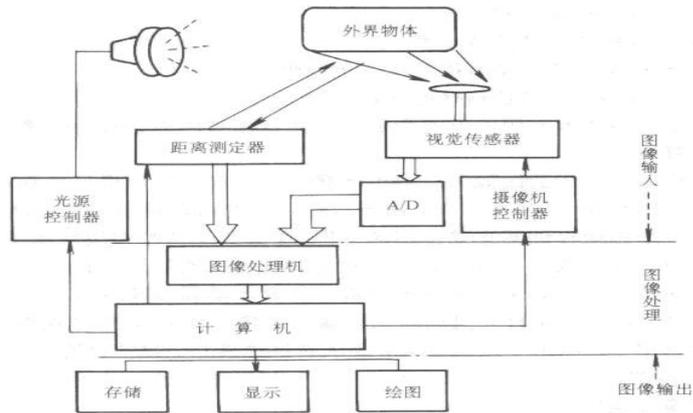


图 17.5.10 机器人视觉系统的硬件组成

智能机器人的视觉信息处理过程可以大体分为三步：首先是视觉信息的输入，选用平面视觉传感器（CCD 摄像机）可以直接获得二维平面图像信号；其次是视觉信息的预处理方法；再有就是对视觉信息的分析。由于图像处理需要的计算量大，机器人运动对实时性要求高，这些都对机器人视觉信息的处理提出了有别于一般图像处理的要求。

机器视觉理论主要有马尔的计算视觉理论。马尔（David Marr）是英国心理学家。他将心理学，人工智能和神经生理学的结果结合起来，对视觉的研究做出了重要贡献。Marr 认为，视觉是检测不变量，是一种信息处理过程，经过这个过程，可从图象中发现外部世界中有什么东西和它们在什么地方。

Marr 指出，对视觉系统的研究有三个层次：第一个层次是信息处理的计算理论层次，在这个层次研究的是对什么信息进行计算和为什么要进行这些计算，即要研究视觉计算的目的与计算策略；第二个层次是表达与算法层次，在这个层次研究的是如何进行所要求的计算，也就是要设计特定的算法；第三个层次是实现算法的机制或硬件层次，在这个层次研究的是如何具体实现某一特定算法。

视觉计算理论认为，对视觉信息的处理的有三个阶段：一是对图像的低层处理，包括图像的获取；图像的预处理（图像滤波、增强、矫正等）和抽取要素图。二是中层处理，包括对图像进行建模和空间表达；恢复其 2.5 维结构，形状与位置。根据马氏(Marr)提出的假设，视觉信息处理过程包括 3 个主要表达层次，即初始简图、二维半简图和三维简图。三是高层分析，包括对景物的识别、分析、理解和描述等。

1. 图像处理与边缘检测

图像处理包括对图像的预处理（滤波、增强、矫正）和压缩等。故图像处理要研究分辨率变化对图像的影响，灰度变化对图像的影响等。

边缘检测是视觉信息处理的基础，其主要目的就是简化信息，便于对信息作进一步的分析。边缘或许对应着图像中物体的边界，或许并没有对应着图像中物体的边界；但是，边缘具有十分令人满意的性质，它能大大地减少所要处理的信息但又保留了图像中物体的形状信息。如果我们能成功地得到图像的边缘，那么图像分析就会大大简化，图像识别就会容易得多。很多图像并没有具体的物体，对这些图像的理解取决于它们的纹理性质，而提取这些纹理性质与边缘检测也有极其密切的关系。图像原本是平面上的连续函数；边缘检测的目的之一就是将其离散化。



图 17.5.11 图像处理与边缘检测

边缘的提取一般采用计算图像辉亮的平均与导数[差分]的方法。

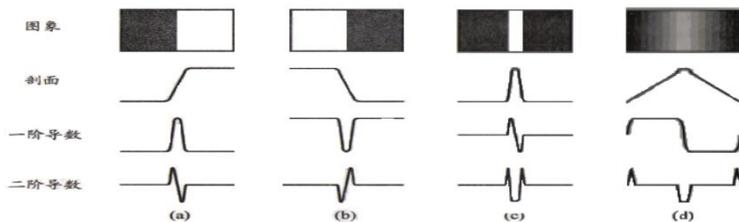


图 17.5.12 边缘和导数示例

2 图像三维重建

图像三维重建是人类视觉的主要目的，也是计算机视觉的最主要的研究方向。所谓三维重建就是指从单幅图像加景物约束、二幅、二幅以上图像恢复空间点三维坐标的过程。是从以轮廓、边缘片段和灰度斑点等表达的初始简图转化为在以观察者为中心的坐标系中，把可见表面的朝向、大致深度等表达出来的 2.5 维简图，再转化为在以物体为中心的坐标系中，用含有体积基元（即表示形状所占体积的基元）和面积基元的模块化分层次表象，描述出物体的形状和形状的空间组织形式的三维图像的过程。

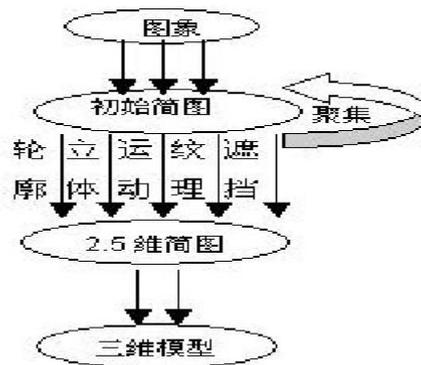


图 17.5.13 视觉系统的三个表象层次

3 景物的分析与识别

物体形状的识别方法有图匹配法(Graph matching)——当且仅当 $P(n)$ 与 $P'(n)$ 对某一给定的相似性量度相似时，形成一对配对(assignment)；若两幅图 G 和 G' 的节点具有一对一的配对，使得所有配对相互兼容，就称这两幅图是同构(isomorphic)的。若 (n, n') 为一配对，仍要求 $P(n)=P'(n')$ 。如果 G 的子图与 G' 的子图同构，称图 G 与 G' 为亚同构

(subisomorphic)的。多层匹配法(Multilevel matching) 对两种描述进行多层匹配的结果,本身就是一种有关它们相似和差异的描述。

景物分析并不是容易的事。为什么难? 难题很多, 其中之一是多尺度。

在不同的距离下观测同一图像获得的感受是不一样的, 远距离看到是图像轮廓, 近距离下看到的是更多细节。“天街小雨润如酥, 草色遥看近却无”(唐·韩愈《早春呈水部张十八员外》), “远山无皱, 远水无波, 远人无目”(宋代, 郭熙), 这就是尺度效应。如果我们从各种不同尺度下观看一棵树上的一朵花(距离不断的变大), 我们“看到”的“东西”先后将是: …, 奔腾的电子流, 分子, 细胞, 花朵, 整株植物, 一片树林, 整个森林, 地球, …。假如说我们关心的只是花朵, 那么在这个过程中, 花朵只会在某些尺度下呈现出来, 在更小的尺度或者更大的尺度下, 它们就消失了。大尺度上观察场景获得的是场景的宏观结构, 小尺度上观察所得将是其“细节纹理”。如果再以更小的尺度去观察, 纹理也是有自己的结构的。

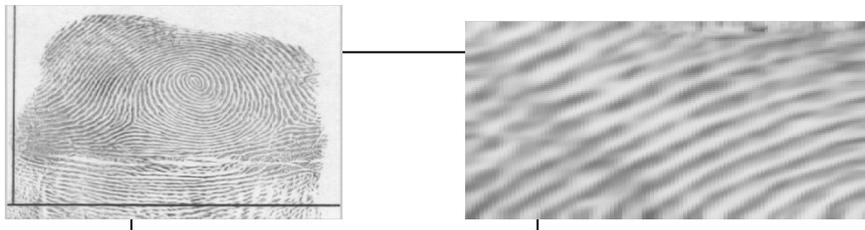


图 17.5.14 不同尺度的纹理

就视觉系统而言, 由于其对物理场景的描述是一个多尺度描述: 即每一尺度都对应着一组具有相应尺寸的感受野的神经细胞。因此, 那些在较大尺度范围内可观察到的物体结构将引起较多层神经元兴奋。

由此, 我们或可引入一个“稳定性假设”: 那些在较大尺度范围内可观察到的物体结构较之那些在较小尺度范围内可观察到的物体结构更为重要。这一稳定性假设有着直接的心理学实验支持。有人曾做过一系列心理测试, 得到的结论是: 那些在较大范围(尺度)存活的结构对视觉而言, 尤为显眼。但要定量描述物体结构的重要程度仍需对尺度参数进行重新标度, 即从众多的标度中寻求一适合生理结构的标度。显著性假设的进一步引申可得到一个基于心理物理学 Weber 定律的物体结构重要性程度的度量。

人类视觉是具有尺度效应的。我们自身的尺度决定了我们视觉系统的大小, 也就决定了我们观察世界的尺度。我们所观察到的物理世界, 其运动和规律都是和我们本身的观察或测量系统(器官)有关的。在各个尺度下看问题的结果是不同的, 所以要全面的看问题就要在多尺度中看问题。“我们望着深渺的宇宙, 看到的只是我们自己孤独的心灵以及同样孤独地注视着我们的上帝。”

17.5.1.3 机器人的听觉与听觉系统

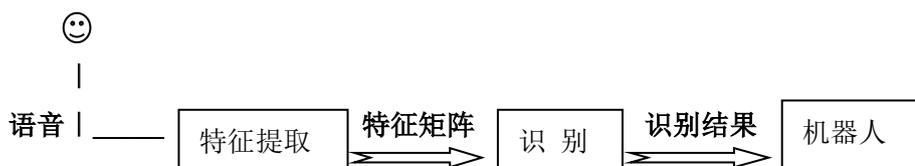


图 17.5.15 机器人听觉系统的粗略框图

机器人的听觉系统如图所示。其听觉识别可分为特定人的语音识别和非特定人的语音识

别。特定人语音识别方法是将事先指定的人的声音中的每一个字音的特征矩阵存储起来，形成一个标准模板（或叫模板），然后再进行匹配。它首先要记忆一个或几个语音特征，而且被指定人讲话的内容也必须是事先规定好的有限的几句话。特定人语音识别系统可以识别讲话的人是否是事先指定的人，讲的是哪一句话。非特定人的语音识别系统大致可以分为语言识别系统，单词识别系统，及数字音识别系统。非特定人的语音识别方法需要对一组有代表性的人的语音进行训练，找出同一词音的共性，这种训练往往是开放式的，能对系统进行不断的修正。在系统工作时，将接收到的声音信号用同样的办法求出它们的特征矩阵，再与标准模式相比较。看它与哪个模板相同或相近，从而识别该信号的含义。

17.5.2 智能机器人的导航与多传感器信息融合

17.5.2.1 智能机器人的导航

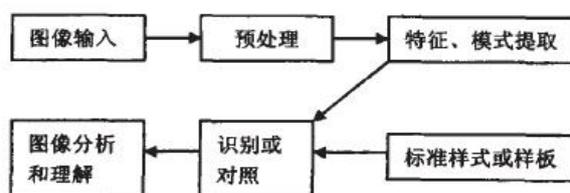


图 17.5.16 图像处理的流程

智能机器人的导航系统分为视觉导航和非视觉传感器组合导航。如果采用视觉导航，视觉导航信息处理的内容主要包括：视觉信息的压缩和滤波、路面检测和障碍物检测、环境特定标志的识别、三维信息感知与处理。视觉导航的图像处理过程如图 17.5.16 所示。

在机器人导航中，无论是局部实时避障还是全局规划，都需要精确知道机器人和障碍物的当前状态及位置，以完成导航、避障及路径规划等任务，这就是机器人的定位问题。通常采用主动式传感器系统进行机器人的定位。主动式传感器系统通过包括超声传感器、红外传感器、激光测距仪以及视频摄像机等主动式传感器感知机器人外部环境或人为设置的路标，与系统预先设定的模型进行匹配，从而得到当前机器人与环境和路标的相对位置，获得定位信息。

17.5.2.2 机器人的多传感器信息融合

信息融合是利用计算机技术对按时序取得的若干信息源的观测信息以一定准则加以自动分析、综合，为完成所需要的决策和估计任务而进行的信息处理过程，它强调三个核心方面：① 信息融合是在几个层次上完成对多源信息的处理过程，其中每一层次都表示不同级别的信息抽象；② 信息融合包括探测、互联、相关、估计以及信息组合；③ 信息融合的结果包括较低层次上的状态和身份估计，以及较高层次上的整个工作态势估计。一般来说，信息融合本质上是一个由低层到顶层对多元信息进行整合、逐层抽象信息处理的过程。

机器人多传感器信息融合技术是机器人智能化的关键技术之一。多传感器信息融合技术对促进机器人向智能化、自主化起着极其重要的作用，是协调使用多个传感器，把分布在不同位置的多个同质或异质传感器所提供的局部不完整测量及相关数据库中的相关信息加以综合，消除多传感器之间可能存在的冗余和矛盾，并加以互补，降低其不确定性，获得对物体或环境的一致性描述的过程。它通过一定的算法来合并多个信息源的信息，以产生更可靠、更准确的信息，并根据这些信息做出可靠的决策。其关键技术可对各类、各个原始信息进行校准、相关估计、误差模式识别和状态决策处理。

多传感器系统是信息融合技术的硬件基础，多源信息是信息融合的加工对象，融合算法是信息融合的核心。多传感器信息融合的一般方法如图 17.5.17 所示。

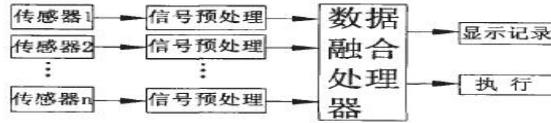


图 17.5.17 多传感器信息融合

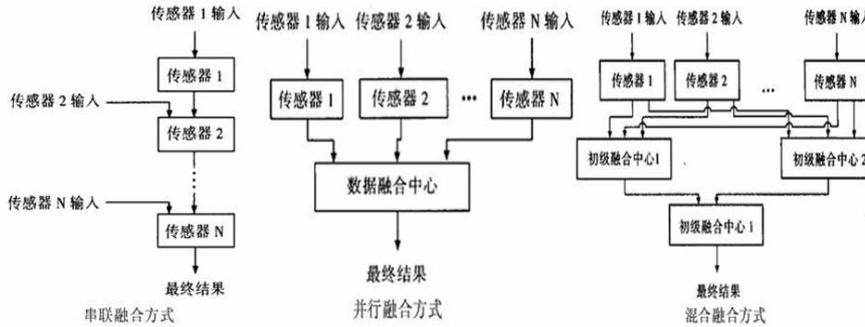


图 17.5.18 数据融合方式

多传感器信息融合技术是通过对这些传感器及其观测信息的合理支配和使用，把多个传感器在时间和空间上的冗余或互补信息依据某种准则进行组合，以获取被观测对象的一致性解释或描述。单传感器信号处理或低层次的数据处理方式只是对人脑信息处理的一种低水平模仿。多传感器数据融合系统可更大程度获取被探测目标和环境的信息量。多传感器数据融合系统有三种结构形式：串联、并联和混合融合形式。

多传感器信息融合由于其应用上的复杂性和多样性，故融合方法研究内容极其丰富。目前多传感器信息融合方法有贝叶斯估计、D-S 理论、卡尔曼滤波、模糊逻辑、神经网络、小波变换等，其中神经网络是多传感器信息融合的重要方法之一。

17.5.2.3 基于神经网络的智能机器人信息融合技术

基于神经网络的信息融合实质上是一个不确定性推理过程，充分利用外部环境的信息，实现知识的自动获取以及在此基础上进行联想推理，经过大量的学习和推理，将不确定环境的复杂关系融合为系统能够理解的信息。神经网络的研究对于多传感器信息融合提供了一种很好的方法，其非线性逼近能力在信息融合中非常引人注目，通常采用的是三层感知器模型和 BP 算法。基于神经网络的信息融合，根据系统要求和融合形式，选择网络拓扑结构，通过网络学习确定网络连接权值，即可对各传感器的输入信息进行融合。

采用神经网络进行信息融合，同时选用权重参数的迭代计算学习算法，中心、宽度的调节权重参数均通过学习来自适应调节到最佳值。训练好的神经网络参加融合的实际过程，如图 17.5.19 所示。传感器获得的信息首先经过适当的处理过程 1，作为神经网络的输入，神经网络对它进行处理并输出相关结果，处理过程 2 再将它解释为系统具体的决策行动。

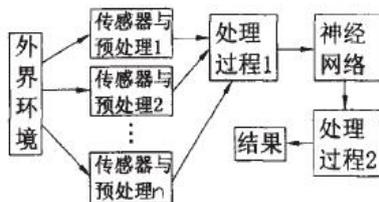


图 17.5.19 基于神经网络的信息融合过程

目前,在智能机器人多传感器信息融合中,神经网络主要用于对智能机器人目标的识别,获得智能机器人对障碍物影像的精确估计,正确地引导机器人运动。采用神经网络的多传感器信息融合方法,能够解决智能机器人的自主行走问题。为了有效地改善神经网络信息融合的效果和速度,利用阵列神经网络进行信息融合的结构模型。

基于神经网络的多传感器信息融合具有如下特点:具有统一的内部知识表示形式,通过特定的学习算法可以将神经网络获得的传感器信息进行融合,获得相应的网络参数;可将知识规则转换成数字形式,便于建立知识库;不用建立系统精确的数学模型,非常适合于非线性测试情况;具有大规模并行处理的能力,使得系统信息处理速度非常快,并且具有很强的容错性和鲁棒性。

多传感器信息融合技术是智能机器人的关键技术之一,随着传感器技术的发展以及信息融合技术水平的提高,智能机器人获取环境信息的感知能力以及系统决策能力将会得到不断的提高。基于神经网络对机器人多传感器信息进行融合,将促进机器人的智能化和自主化。

17.5.3 智能机器人的行为规划与控制

智能机器人的规划就是机器人在行动之前,决定采取何种步骤、以何种程序实现预期目标的过程。规划的基础需要机器人具有与完成预期目标相关的知识环境描述。智能机器人的决策就是通过如视觉、触觉、超声等传感器获取外部环境的实时信息,并利用历史记录的信息等环境的局部信息进行避碰、路径实时规划或已有规划的更新。通过对感知到的各种环境状态及变化,及时做出适当的判断、推理、预测、估计,给出相应的对策,使机器人能够实施一系列的动作,完成预期的目标。

规划技术的早期研究主要依据于二个已得到深入开发的技术:搜索和定理证明。可以说,最早的规划系统是**通用问题求解系统 GPS**。原则上,GPS 可用于解决任何搜索问题,并且其解决的某些问题就是规划问题。GPS 给规划的实现设置了以下假设:① 计划(作为规划的结果)是动作序列;② 执行计划的目的是到达目标状态;③ 每个动作的执行结果是完全可预言的。早期的规划技术基本上都遵循了该假设,我们将遵循了该假设的规划技术称为经典规划技术。

与GPS 同期开发的另一个规划技术是Green 方法,其将规划问题的解决归约到**定理证明**:引入了状态演算去演绎动作序列。Green 方法的优点是能以状态演算公理(表示为谓词逻辑语句)清晰地描述动作的执行对问题状态的影响(即特定的动作导致特定的状态),但定理证明寻找的推理链仅间接地关系到动作链,会导致难以解决的搜索问题,且效率极低。作为对照,GPS 规划针对过程中目标状态与初始状态的差别,寻找适当的动作去直接消除差别,规划效率较高。但GPS 也有缺点:其要求建立领域相关的程序去检查和处理状态差别,并需设计**操作符-差别表**去记载各操作符能消除的状态差别。然而,这些工作对于规划系统的设计者来讲是耗时和冗烦的。由于各自的缺点,GPS 和Green 方法都只能用于解决简单的规划问题。

斯坦福研究所于1969年开始设计了著名的机器人动作规划系统STRIPS,其用类似于**状态演算形式的动作定义**取代GPS 中的操作符-差别表,从而在综合运用GPS 和Green 方法优点的同时避免了它们的许多缺点。尽管STRIPS 能解决的规划问题比GPS 和Green 方法大得多,但其搜索控制方式过于简单,难以解决复杂的规划问题。70年代中期出现的部分排序规划技术在这方面取得了突破性进展。作为这种技术的代表,NOAH 系统和目标回归方法开

拓了基于部分计划集的搜索技术。每个部分计划（又称为计划步）对应一个其先决条件需得到满足的动作，而该先决条件又成为要通过规划达到的子目标。尽管部分排序规划技术被证明具有完备性，即能解决所有的经典规划问题，但由于大量实际规划问题并不遵从经典规划问题的假设，所以部分排序规划技术未得到广泛的应用。为消除规划理论和实际应用间存在的差距，进入八十年代中期后，规划技术研究的热点转向开拓非经典的实际规划问题。然而，经典规划技术，尤其是部分排序规划技术仍是开发规划新技术的基础。

17.5.3.1 规划问题的描述

规划是人类生产和社会活动的重要形式。规划旨在为活动实体（人、组织、机器）设计合理的行为—按时间顺序的活动序列。从知识工程的角度看，可以说自动规划是综合和构造型问题求解任务。规划问题处理的对象是动作，而约束动作的主线是时间顺序。

若用状态空间表示法来描述规划，所谓规划就是设计一个动作序列（也称为动作块），使得通过执行该动作序列，可以将系统从初始状态转变为目标状态。若以 σ —初始状态指示符， ρ —目标状态指示符， Γ —动作序列指示符集， γ —动作序列指示符， $\gamma \in \Gamma$ ， Ω —描述状态和可用动作的数据库。则 γ 作为规划的结果，它的执行将导致系统状态（或系统所处的世界状态）从初始状态（ S_i ）转变为目标状态（ S_g ）。

下面引入描述规划问题的几个基本概念：**状态、动作[操作]、数据库、框架公理和计划**。为了便于阐述，我们就以一个积木块世界中的机器人动作规划—[规划问题 1]—为例来说明这些概念。规划问题 1 假设：积木块世界中有三个积木块：A、B、C；机器人只有一个机器手，且每次只能拿起一个积木块；其初始状态 S_1 为：积木 A 叠放 C 上，B 在桌子上；要求的目标状态 S_g 是：积木依 C、B、A 的上下次序叠放在桌子上。

(1) **状态**。状态可视为在一个给定的时间点对于世界的一个快照。我们可用一个二元约束 T 来描述某个状态 S 下关于世界的特性，且特性用谓词公式加以表示。特别地，我们引入谓词 On 、 $Clear$ 和 $Table$ 分别指示一个积木块放置于另一积木块之上、积木块顶空和积木块放置于桌子上。于是规划问题 1 的初始状态 S_1 就可表示为：

$$\{ T(On(A, C), S_1), T(Clear(A), S_1), T(Table(C), S_1), T(Table(B), S_1) \}$$

作为作规划，需要给定一个起始状态，但目标状态在许多情况下并非唯一。例如，若令目标状态 S_g 描述为：

$$\{ T(On(A, B), S_g) \},$$

则有三个可能的状态作为目标状态。此外，目标状态的描述也可包含变量，例如 $T(On(A, x), S_g)$ 。

(2) **动作**。对于经典的规划问题，假设世界的变化只能是由于执行器[主体]（如机器人）执行了某些动作。规划系统常需定义一些操作符，而包含在规划的结果—动作序列中的具体动作则是这些操作符的实例。例如，积木块世界就可为机器人定义三个操作符：

$Unstack(x, y)$ —将 x 从 y 上取走，并放置到桌子上，

$Stack(x, y)$ —将桌子上的 x 放置到 y 上，

$Move(x, y, z)$ —将 x 从 y 上移动到 z 上。

则动作 $Stack(A, B)$ 就是操作符 $Stack(x, y)$ 的一个实例。这三个操作符简记为 U 、 S 、 M 。

下面引入 Do 函数表示动作的执行对于世界的影响，以 $Do(a, s)$ 表示动作 a 的执行将状态 s 转变为下一状态，并就以 $Do(a, s)$ 指示下一状态。形式地，有 $Do: \Phi \times \Psi \rightarrow \Psi$ 其中 Φ 和 Ψ 分别指示动作集和状态集。基于 Do 函数，可以建立规则去形式地描述操作符的

激活条件和其执行对于世界的影响。下面给出关于操作符 U 和 S 的规则：

$$\begin{aligned}
 U: & T(On(x, y), s) \wedge T(Clear(x), s) \Rightarrow \\
 & T(Table(x), Do(U(x, y), s)) \wedge T(Clear(y), Do(U(x, y), s)) \\
 S: & T(Table(x), s) \wedge T(Clear(x), s) \wedge T(Clear(y), s) \wedge x \neq y \\
 & \Rightarrow T(On(x, y), Do(S(x, y), s))
 \end{aligned}$$

关于 U 的操作规则意指：若在 s 状态下有 On(x, y) 和 Clear(x) 成立，则可执行 U(x, y)，并且在执行 U 后的状态（以 Do(U(x, y), s) 指示）下，有 Table(x) 和 Clear(y) 成立。

规划的结果表示为一个动作序列，通常以方括号将动作序列括起来，称为动作块。例如，对于规划问题 1，作为规划结果的动作块表示为：

$$[U(A, C), S(B, C), S(A, B)]$$

为表示动作序列（动作块）中的动作是依排列次序逐个执行的，我们定义：

$$Do(a.l, s) \Leftrightarrow Do(l, Do(a, s))$$

其中 a 为单一动作，l 为动作序列，“.” 指示 a 和 l 处于同一动作块，且 a 排列在 l 之前。

例如

$$\begin{aligned}
 Do([U(A, C), S(B, C), S(A, B)], s) & \Leftrightarrow \\
 Do([S(B, C), S(A, B)], Do(U(A, C), s)) & \Leftrightarrow \\
 Do(S(A, B), Do(S(B, C), Do(U(A, C), s))) &
 \end{aligned}$$

对应于世界状态的特性描述 P，有以下动作块公理成立：

$$T(P, Do(a.l, s)) \Leftrightarrow T(P, Do(l, Do(a, s)))$$

该公理将应用于 Green 方法中，以使规划过程中能把单一动作从动作序列中分离出来加以处理。

(3) 数据库。 规划系统使用的数据库（也称知识库）除了包括规划的初始状态、目标状态和操作符外，还应包括逻辑操作公理和关于状态约束的描述。

1) 逻辑操作公理。这些公理将二元约束 T 中的状态特性描述化简为仅是原子谓词公式，以缩减推理的复杂性。这些公理列举如下：

$$\begin{aligned}
 (\forall P) (\forall s) \{T(\neg P, s) \Leftrightarrow \neg T(P, s)\} \\
 (\forall P) (\forall Q) (\forall s) \{T(P \vee Q, s) \Leftrightarrow (T(P, s) \vee T(Q, s))\} \\
 (\forall P) (\forall Q) (\forall s) \{T(P \wedge Q, s) \Leftrightarrow (T(P, s) \wedge T(Q, s))\} \\
 (\forall P) (\forall Q) (\forall s) \{T(P \Rightarrow Q, s) \Leftrightarrow (T(P, s) \Rightarrow T(Q, s))\}
 \end{aligned}$$

2) 状态约束。状态约束表示一个问题域中所有状态下均必须为真的事实。例如，积木块世界就应该有以下状态约束：

$$\begin{aligned}
 (\forall x) (\forall s) \{T(Table(x), s) \Leftrightarrow \neg(\exists y)T(On(x, y), s)\} \\
 (\forall y) (\forall s) \{T(Clear(y), s) \Leftrightarrow \neg(\exists x)T(On(x, y), s)\} \\
 (\forall x) (\forall y) (\forall z) (\forall s) \{T(On(x, y), s) \wedge y \neq z \Leftrightarrow \neg T(On(x, z), s)\}
 \end{aligned}$$

其中，最后一个状态约束指示不允许同一积木块 x 跨放在二个积木块之上。状态约束主要用于检查规划过程中，设想要到达的中间状态是否为不合法状态，以支持规划过程中搜索路径的修剪。

(4) 框架公理。 关于操作符的规则定义指出了动作（操作符实例）执行的条件和执行后新成立的事实（状态特性描述），但这种描述并不完备，其未指出操作前后保持不变的事实（事物间的关系）。例如对规划问题 1，规划结果（动作序列）的执行过程中，事实 Clear(A)

和 Table(C) 始终不变。

框架问题就是在这样的背景下提出的,其旨在刻画一个状态描述中不由动作改变的方面,“框架”的取名来自对动画片制作的比拟。动画片往往若干个画面具有相同的背景,只需制作不同的前景。规划系统中关于操作符的规则定义相当于制作动画片的前景,而背景则用所谓的框架公理加以表示。例如,对于上述积木块世界的 Unstack 操作符,就可建立以下框架公理:

$$T(\text{Clear}(u), s) \Rightarrow T(\text{Clear}(u), \text{Do}(U(x, y), s))$$

$$T(\text{Table}(u), s) \Rightarrow T(\text{Table}(u), \text{Do}(U(x, y), s))$$

$$T(\text{On}(u, v), s) \wedge u \neq x \Rightarrow T(\text{On}(u, v), \text{Do}(U(x, y), s))$$

利用框架公理,使规划系统仅用简单的推理机制(例如归结反演),就能实现自动规划。然而,大量框架公理的引入将使规划效率大幅度下降。通常,所需框架公理的数目是状态特性描述谓词的个数与操作符个数的乘积,因此当规划问题复杂时,将导致大量框架公理的引入。鉴于实际的规划问题都比积木块世界动作规划复杂得多,必须研究新的方法,使得不求助于显式表示的框架公理,也能解决框架问题。

(5) **计划。计划就是规划的结果,表示为动作序列 γ** ,其必须满足以下充要条件:

$$(1) \quad \gamma \in \Gamma; \quad (2) \quad \Omega \models \text{Do}(\gamma, \sigma), \text{ 使得 } \rho = \text{Do}(\gamma, \sigma)$$

例如,令 σ 指示规划问题 1 中的初始状态,则通过规划应能推导出动作序列 $\gamma = [U(A, C), S(B, C), S(A, B)]$,使得 $\text{Do}(\gamma, \sigma)$ 就是期望的目标状态 ρ 。

17.5.3.2 早期的(一些)自动规划技术

最早出现的规划技术是 GPS 和 Green 方法,以后 STRIPS 综合了这两者的优点,成为很有影响力的早期机器人规划技术。

1. GPS 规划方法

GPS 针对规划的目标状态与初始状态间的差别来寻找能直接消除差别的动作。为此,需要建立领域相关的程序去检查状态差别,并设计操作符-差别表去记载各操作符能消除的差别。

在积木块世界动作规划问题中,可以通过比较目标状态和初始状态的描述(以谓词公式描述)来确定差别。例如规划问题 1 中初始状态 S_0 和目标状态 S_g 就可分别描述为:

$$S_0: \{T(\text{Clear}(A), S_0), T(\text{On}(A, C), S_0), T(\text{Table}(C), S_0),$$

$$T(\text{Table}(B), S_0), T(\text{Clear}(B), S_0)\}$$

$$S_g: \{T(\text{Clear}(A), S_g), T(\text{On}(A, B), S_g), T(\text{On}(B, C), S_g),$$

$$T(\text{Table}(C), S_g)\}$$

所以 S_g 和 S_0 之间的状态差别是: $\{\text{On}(A, B), \text{On}(B, C)\}$

表 5.1 是操作符-差别表,给出前述积木块世界机器人操作符能消除的差别。依据该表,上述差别 $\text{On}(A, B)$ 和 $\text{On}(B, C)$ 可分别通过在初始状态下执行动作 $S(A, B)$ 和 $S(B, C)$ 去消除。换言之,规划系统可以安排动作 $S(A, B)$ 和 $S(B, C)$ 去实现初始状态中缺少的,但目标状态中需要存在的世界特性

$\text{On}(A, B)$ 和 $\text{On}(B, C)$ 。为执行这二个动作,需满足它们的激活条件: A, B, C 均顶空。鉴于初始状态下 C 非顶空,所以 $\text{Clear}(C)$ 成为新的要消除的差别(也成为子目标)。类似地可从操作符-差别表查到动作 $U(x, C)$ 能消除该差别。由于 $U(x, C)$ 的激活条件 $\text{On}(A, C)$ 存在于初始状态,所以规划成功结束。经整理后,规划结果就是动作序列: $[U(A, C), S(B, C),$

S(A, B)]。

基于操作符-差别表的推理控制策略又称手段-目的分析法。拟消除的差别就是目的，而手段则是能用于消除差别的操作符。运用这种方法的主要问题在于，要消除的各个差别往往并非独立，具有一定的交互作用。例如上例中，若先消除差别 $On(A, B)$ 就会阻碍消除差别 $On(B, C)$ ，因为 $S(A, B)$ 的执行使得 $S(B, C)$ 的激活条件 B 顶空不能满足。所以，GPS 实际只能用于解决一些很简单的问题，复杂问题的解决必须依赖大量启发式知识的引入。

	U(x, y)	S(x, y)	M(x, y, z)
Clear(y)	√		
Table(x)	√		
On(x, y)		√	√
On(x, z)			√

表 5.1 操作符-差别表

2. Green 规划方法

格林 (Green) 于 1969 年设计了一个基于状态演算的规划器，称为 **Green 方法**，旨在为仿真机器人构造动作计划。该方法从一个计划存在语句 $(\exists \gamma) Goal(Do(\gamma, \sigma))$ 出发，利用归结反演定理证明方法去证明存在一个正确的计划 γ ，使得 $\rho = Do(\gamma, \sigma)$ ，并推导出 γ 包含的动作序列 (Goal 指示目标状态 ρ 的描述)。

证明过程从计划存在语句的取反开始。归结反演使用的子句集内容，除了来自初始状态和目标状态的描述外，还有相应于操作符规则和框架公理的子句。例如，相应于 $U(x, y)$ 操作规则的子句即表示为：

$$\{-T(On(x, y), s) \vee \neg T(Clear(x), s) \vee T(Table(x), Do(U(x, y), s)), \\ \neg T(On(x, y), s) \vee \neg T(Clear(x), s) \vee T(Clear(y), Do(U(x, y), s))\}$$

而相应于 $U(x, y)$ 的一条框架公理的子句则表示为：

$$\neg T(On(u, v), s) \vee \neg(u \neq x) \vee T(On(u, v), Do(U(x, y), s))$$

若将 $u \neq x$ 视为作一个附加检查，则该子句简化为：

$$\neg T(On(u, v), s) \vee T(On(u, v), Do(U(x, y), s))$$

从实现规划的角度，证明过程实际上就是一个从目标状态出发，逐步消除目标状态与初始状态差别的过程，所以实现了一种的逆向推理风格。鉴于差别的消除只有通过执行适当的动作来实现，所以应优先使用相应于操作符规则的子句进行归结；另外，预测动作序列 γ 的长度 (动作个数) 以及确定何时应使用框架公理也是十分必要的；这些均可以设计为控制证明过程的启发式知识。

从实现规划的角度，证明过程实际上就是一个从目标状态出发，逐步消除目标状态与初始状态差别的过程，所以实现了一种的逆向推理风格。鉴于差别的消除只有通过执行适当的动作来实现，所以应优先使用相应于操作符规则的子句进行归结；另外，预测动作序列 γ 的长度 (动作个数) 以及确定何时应使用框架公理也是十分必要的；这些均可以设计为控制证明过程的启发式知识。

下面，通过例子来观察 Green 方法的应用。

例 1: 规划问题的初始状态 $So: \{ T(Clear(A), So), T(On(A, B), So), T(On(B, C), So), T(Table(C), So) \}$, 目标状态 $\rho: \{ T(Table(A), \rho) \} \Leftrightarrow Goal(\rho)$ 。

则其证明过程 (其中 $Ans(a)$ 用于指示动作序列 a 的提取) :

- ① $\{ \neg Goal(Do(a, So)), Ans(a) \}; (\exists a) Goal(Do(a, So));$
- ② $\{ \neg T(Table(A), Do(a, So)), Ans(a) \};$ 与 $\neg T(Table(A), \rho) \vee Goal(\rho)$ 归结, 且有 $Do(a, So) / \rho$;
- ③ $\{ \neg T(On(A, y), So), \neg T(Clear(A), So), Ans(U(A, y)) \};$ 与相应于 $U(x, y)$ 规则的前一子句归结, 且有: $A/x, U(A, y)/a, So/s$;
- ④ $\{ \neg T(Clear(A), So), Ans(U(A, B)) \};$ 与初始状态描述 $T(On(A, B), So)$ 归结, 且有 B/y ;
- ⑤ $\{ Ans(U(A, B)) \};$ 与初始状态描述 $T(Clear(A), So)$ 归结

规划的结果是由单一动作构成的动作块 $[U(A, B)]$ 。注意, 规划过程用花括号收集归结式, 谓词公式间的逗号隐含着“析取”。

例 2: 规划问题的初始状态 So 同例 1, 目标状态: $\{ T(Table(B), \rho) \} \Leftrightarrow Goal(\rho)$

则其证明过程:

- ① $\{ \neg Goal(Do(a, So, Ans(a))) \};$
 - ② $\{ \neg T(Table(B), Do(a, So)), Ans(a) \};$
 - ③ $\{ \neg T(Table(B), Do(c, Do(b, So))), Ans([b, c]) \};$ 令 $a = b.c$ (或 $[b, c]$), 应用动作块公理;
 - ④ $\{ \neg T(On(B, y), Do(b, So)), \neg T(Clear(B), Do(b, So), Ans([b, U(B, y)])) \};$ 与相应于 $U(x, y)$ 规则的前一子句归结, 且有 $B/x, U(B, y)/c, Do(b, So)/s$;
 - ⑤ $\{ \neg T(On(B, y), Do(U(x, B), So)), \neg T(On(x, B), So), \neg T(Clear(x), So), Ans([U(x, B), U(B, y)]) \};$ 与相应于 $U(x, y)$ 规则的下一子句归结, 且有 $B/y, U(x, B)/b, So/s$;
 - ⑥ $\{ \neg T(On(B, y), So), \neg T(On(x, B), So), \neg T(Clear(x), So), \neg Ans([U(x, B), U(B, y)]) \};$ 应用相应于 $U(x, y)$ 框架公理的子句作归结, 实现状态同一 (使花括弧中的状态描述相应于同一状态, $Do(U(x, B), So)$ 指示了 So 的下一状态);
 - ⑦ $\{ \neg T(On(x, B), So), \neg T(Clear(x, So), Ans([U(x, B), U(B, C)]) \};$ 与初始状态描述 $T(On(B, C), So)$ 归结, 且有 C/y ;
 - ⑧ $\{ \neg T(Clear(A, So), Ans([U(A, B), U(B, C)]) \};$ 与初始状态描述 $T(On(A, B), So)$ 归结, 且有 A/x ;
 - ⑨ $\{ Ans([U(A, B), U(B, C)]) \};$ 与初始状态描述 $T(Clear(A), So)$ 归结
- 规划的结果是动作块: $[U(A, B), U(B, C)]$ 。

可以证明 Green 方法是健全的, 即只要规划问题有解, 就必定可以获得。然而 Green 方法的效率是很低的, 引入启发式知识, 如预测动作序列的长度、抑制框架公理的应用 (优先使用相应于操作符规则的子句)、及时实现状态同一 (框架公理的主要用处)、及时发现和修剪不可达状态等, 可以提高规划效率; 但鉴于归结反演定理证明方法本身的低效率性, Green 方法实际上无法用于解决复杂的规划问题。

3. STRIPS 规划方法

STRIPS 是早期自动规划领域中具有重要影响的一个系统。尽管其本质上是 GPS 的特化应用, 并沿用了 GPS 关于经典规划问题的假设, 但其提出的规划知识表示方法删除了显式表

示框架公理和状态指示符的需要,对于推动规划技术的研究具有十分重要的意义,其设计的目标栈规划技术和三角表计划模式也很有特色。

[1]规划知识的表示。STRIPS 采用类似于状态演算的方式描述操作符,从而删除了建立操作符一差别表的需要。不过,像 Green 方法那样定义关于操作符的规则,需要同时引入大量框架公理,对于复杂的规划问题,会导致冗繁的设计和低效的规划过程,显然不可取。STRIPS 革新了 Green 方法,提出用三个描述集(简称表)表示一个操作符,并删除了二元约束 T,不仅简化了操作符的描述,也使框架公理的隐含表示成为可能。

下面以操作符 $U(x, y)$ 为例来说明这种三表描述方法:

$$\text{Pre}(U(x, y)) = \{\text{On}(x, y), \text{Clear}(x)\}$$

$$\text{Add}(U(x, y)) = \{\text{Table}(x), \text{Clear}(y)\}$$

$$\text{Del}(U(x, y)) = \{\text{On}(x, y)\}$$

其中,前提表(Pre)描述 $U(x, y)$ 的激活条件,加表(Add)描述 $U(x, y)$ 执行后新增加的世界特性,而删表(Del)则描述 $U(x, y)$ 执行后已不再存在的世界特性,实际上,加表和删表联合起来完备地描述了世界状态的变化,未涉及的状态特性必定保持不变。换言之,只要将删表内容从执行 $U(x, y)$ 前的实际状态中消除,并增添加表的内容,就可得到执行 $U(x, y)$ 后世界状态的完备描述。从而,就不再需要用显式的框架公理去描述 $U(x, y)$ 执行前后保持不变得状态特性。实际上 Green 方法需要显式表示框架公理的要害是,关于操作符的规则未指出 $U(x, y)$ 执行后需删除的已不再成立的状态特性。

类似地,可以建立 $S(x, y)$ 的三表描述:

$$\text{Pre}(S(x, y)) = \{\text{Table}(x), \text{Clear}(x), \text{Clear}(y)\};$$

$$\text{Add}(S(x, y)) = \{\text{On}(x, y)\};$$

$$\text{Del}(S(x, y)) = \{\text{table}(x), \text{Clear}(y)\}$$

另一个简化是消除了状态指示符的显式表示。STRIPS 修改表示世界状态的数据库(相当于产生式系统的综合数据库),使其总是指示当前的世界状态。

[2]目标栈规划技术。STRIPS 采用 GPS 的手段-目的分析法解决规划问题,即把规划视为通过选用适当动作(操作符)逐渐消除目标状态和初始状态差别的过程。由于问题(世界)状态以谓词公式描述,状态差别就是那些出现于目标状态但初始状态又缺少的谓词公式。例如,对于规划问题 1 的状态差别就是:

$$\{\text{On}(A, B), \text{On}(B, C)\}$$

于是消除状态差别成为规划目标,而消除 $\text{On}(A, B)$ 和 $\text{On}(B, C)$ 则成为子目标。通常子目标并非完全独立,往往存在某些交互作用。例如,先解决子目标 $\text{On}(A, B)$,会阻碍子目标 $\text{On}(B, C)$ 的解决。

STRIPS 设计了目标栈技术来解决子目标具有交互作用的规划问题。规划开始时建立目标栈,并将规划目标压进栈中。若规划目标包含多个子目标,则用启发式知识排序子目标,将它们逐个压进栈,并使不易引起冲突的子目标出现在栈顶。然后逐个取出栈顶的子目标加以解决,即选择能解决子目标(消除相应状态差别)的操作符。若该操作符的激活条件不存在于初始状态,就把该激活条件压进栈作为新子目标。一旦-复合目标的各子目标都得到解决,该复合目标也就得到解决,并从目标栈取走。

基于目标栈解决规划问题 1 的过程可如下:首先将复合目标 $\text{On}(A, B) \wedge \text{On}(B, C)$ 压进栈,然后从中分离出二个子目标,并按启发式知识将它们压进栈。接下来,取出子目标 $\text{On}(B,$

C)加以解决：选择动作 $S(B, C)$, 并将 $S(B, C)$ 激活的前提条件之一 $Clear(C)$ (其不存在于初始状态) 作为新子目标压进栈。再将 $Clear(C)$ 取出解决：选择动作 $U(x, C)$; 当 x 束缚值取 A 时, 该动作激活的前提条件满足。

接下来, STRIPS 仿真性地执行已规划的动作 $U(A, C)$ 和 $S(B, C)$, 使仿真的积木块世界状态转变(注意, 此时真实的积木块世界并不改变), 并对世界状态作相应修改。接着, 目标栈顶的子目标已是 $On(A, B)$, 可选择动作 $S(A, B)$ 加以解决。由于此时复合目标的二个子目标都已解决, 复合目标可从栈中取出; 此时栈空, 仿真的积木块世界也到达目标状态, 规划完成, 并产生动作块 $[U(A, C), S(B, C), S(A, B)]$ 作为机器人行动计划。

目标栈规划技术的主要问题是启发式知识不能确保子目标排序正确。例如, 上例中若子目标 $On(A, B)$ 和 $On(B, C)$ 的排序搞反, 则会引起冲突。STRIPS 允许在分析冲突时, 破坏已到达的子目标(即 $On(A, B)$), 并重新将该子目标压进栈中, 等待再解决。结果产生的动作块将是: $[U(A, C), S(A, B), U(A, B), S(B, C), S(A, B)]$ 。显然这样的机器人行动计划是不能令人满意的。解决的办法有二种。一是修正产生的计划, 把冗余的动作从计划中删去, 如上述计划的第 2、3 个动作。但对于复杂的规划问题, 寻找冗余的动作并不容易, 而且在冲突的发生较多的情况下规划效率会大幅度下降。另一种办法是改革规划技术, 下一节将介绍的部分排序规划技术能有效免除冲突的发生。

[3]三角表计划模式。STRIPS 设计了一个三角形的表结构去存放规划的结果—行动计划, 包括规划的初始状态, 目标状态以及行动计划中每个动作的执行(激活)条件和结果(增加的状态特性)。通常, 三角表中第 0 列中的内容表示初始状态的特性描述, 表中最后一行中的内容表示目标状态的特性描述。行动计划包含的动作按执行的先后次序排列在 $j > 0$ 的各列顶部, 动作所在列的内容表示该动作执行后的结果。对于第 j 个动作(记在第 j 列顶部), 行号 $\geq j$ 且列号 $< j$ 的三角表部分称为第 j 个动作的核; 核表示了该动作执行前的状态特性描述, 并包括了该动作的激活条件。

只要用变量取代三角表中动作和谓词公式的常量参数, 就可建立起三角表计划模式。计划模式的用处体现在两个方面: 重用和辅助计划的执行。只要建立一个计划模式库, 就可依据当前规划问题的初始状态和目标状态, 从库中寻找匹配的计划模式, 以加快规划过程。另外, 行动计划的执行过程中, 可能会发生非期望情况, 如机器人碰掉了某个积木块, 另一机器人改变了世界状态特性等。可以通过检查计划中每个待执行动作的激活条件是否满足来确定计划的修正部位, 而不必重新规划整个行动计划。

17.5.3.3 部分排序规划技术

STRIPS 的目标栈规划技术是一种线性规划方法, 子目标一旦进入栈表, 就只能按后进先出的次序等待解决, 不能按规划过程的实际情况灵活调整排序。正如前述, 这种线性排序往往会导致子目标冲突, 使得目标栈规划技术不适合用于解决子目标有复杂交互作用的规划问题。

70 年代开发的层次规划(以 NOAH 系统为代表)和目标回归规划技术突破了线性规划的限制。前者将子目标按其派生的情况分层, 并逐层排序规划的结果; 后者则将目标栈放宽为目标集。可以从中选出任意合适的子目标加以处理。这些技术曾被称为非线性规划技术, 但就其本质而言, 无论是逐层排序规划的结果, 还是从目标集选取子目标, 都是作部分排序, 所以后来就称这些技术为 部分排序规划技术。

1. 目标回归规划方法

目标回归 (Goal Regression) 是一种极其简单但又强有力的非线性规划技术, 规划的每一步记为 $\text{Reg}(q, a)$, 意指目标状态 q 通过动作 a 的回归—回归到动作 a 执行之前的状态, 该状态就以 $\text{Reg}(q, a)$ 本身指示。换言之, 执行动作 a , 就可使上一状态 $\text{Reg}(q, a)$ 转变为下一状态 q 。动作 a 就以 STRIPS 的三表描述法表示, 则 q 通过 a 回归的准则表示为:

$$(q \cap \text{Del}(a)) = \{ \} \Rightarrow \text{Reg}(q, a) = \text{Pre}(a) \cup (q - \text{Add}(a))$$

即只有 q 与 $\text{Del}(a)$ 不相交, 才能通过 a 回归。因为从回归到的上一状态通过执行 a 转变到下一状态 q 时, $\text{Del}(a)$ 的内容应该已从世界状态的特性描述中删除, 否则就不能回归。显然回归到的上一状态的特性描述必定应包含 $\text{Pre}(a)$ 作为 a 的激活条件, 同时也包含删除 $\text{Add}(a)$ 之外的 q 特性描述。

下面观察[规划问题 2]是如何使用目标回归方法加以解决的。规划问题 2 的初始状态: $\{\text{Clear}(C), \text{On}(C, A), \text{Table}(A), \text{Clear}(B), \text{Table}(B)\}$; 目标状态: $\{\text{On}(A, B), \text{On}(B, C)\}$; 假定只允许使用操作符 U 和 S 。

目标回归过程从规划问题的目标状态开始, 对每个子目标 (描述目标状态的一个谓词公式) 作回归试探。每个回归试探包括以下步骤:

1) 确定可用于回归的动作 a , 使 $\text{Add}(a)$ 包含该子目标 (谓词公式)。原则上, 对每个子目标均能找到 1-多个动作。

2) 检查回归的必要条件 (即 $q \cap \text{Del}(a) = \{ \}$) 是否满足。

3) 建立回归到的上一状态的特性描述。

4) 检查回归到的上一状态是否合法状态。

对于初始的二个目标 $\text{On}(A, B)$ 和 $\text{On}(B, C)$, 各自找到相应的动作 $S(A, B)$ 和 $S(B, C)$ 用于回归。但通过 $S(B, C)$ 回归到的上一状态是非法状态: $\text{Clear}(B)$ 和 $\text{On}(A, B)$ 的同时存在违反了状态约束, 应修剪掉这个回归路径。从通过 $S(A, B)$ 回归到的上一状态再次寻找可用于回归的动作, 共 4 个, 但只有通过动作 $S(B, C)$ 回归到的上一状态是合法的, 其余回归路径均修剪掉。进一步通过动作 $U(x, A)$ 回归, 只要取 x 束缚值为 C , 回归到的上一状态就是初始状态。到此规划成功, 动作块 $[U(C, A), S(B, C), S(A, B)]$ 成为机器人行动计划。在规划过程中, 及时检查子目标对应的谓词公式是否已存在于初始状态的特性描述, 也很重要。若已存在, 就不必对该子目标作回归试探。本例通过 $S(A, B)$ 回归到的上一状态中, $\text{Table}(A)$ 和 $\text{Clear}(B)$ 已经是初始状态的特性描述, 所以实际上可不对它们作回归试探; 而通过 $S(A, B)$ 和 $S(B, C)$ 二次回归到达的状态中, 只有子目标 $\text{Clear}(A)$ 需要作回归试探。

目标回归在复杂的情况下会产生浓密的回归树。幸运的是, 通过回归准则的检查以及状态约束的检查, 许多回归路径会被修剪掉, 从而使规划快速收敛。显然, 目标回归是通过回归试探来决定子目标的部分排序的, 从而与基于目标栈的线性规划方法有本质的不同。

2. 层次规划系统 NOAH

NOAH 是能有效解决子目标具有交互作用的规划问题的另一种方法, 其采用称为最少承担 (Least Commitment) 的策略控制规划过程。该策略的基本思想是推迟决策到有足够的信息可用于分析的时候, 以使决策失败的风险降到最少程度。例如, 对于规划问题 1, 若我们轻率地决定解决子目标 $\text{On}(A, B)$ 优先于 $\text{On}(B, C)$, 则到后来解决子目标 $\text{On}(B, C)$ 时就会发现, 必须撤消已达的子目标 $\text{On}(A, B)$ 。这是一种浪费, 在错误动作不可撤消的问题求解任务中还会造成更严重的后果。例如, 打算油漆梯子和天花板。若轻率地决定先油漆梯子, 则会造成无法接着油漆天花板的后果, 因为必须等油漆干了。

众所周知,复杂的问题求解任务往往可分解为子问题分别加以求解,然后再综合。然而这些子问题往往只是准独立的,存在着或多或少的交互作用,这就要求必须对求解这些子问题的次序加以仔细排序。有时仅排序第一层子问题仍不够,还须对下层子问题也进行排序。例如规划问题 2,对于第一层子问题(子目标) $O_n(A, B)$ 和 $o_n(B, C)$,无论先解决那个,均不可行,必须对下层子问题 $C_{clear}(A)$ 也作排序,即先解决 $C_{clear}(A)$,再 $O_n(B, C)$,最后 $O_n(A, B)$,才能顺利完成规划。

NOAH 系统就按最少承担策略来指导规划问题的层次分解和规划结果的部分排序。每当从上层规划目标分解出子目标时,并不轻率决定子目标解决的顺序;而是先把这些子目标作为完全独立的子目标分别加以解决(建立部分计划),然后基于部分计划提供的足够信息决定动作(包含于不同部分计划中的动作)的排序。NOAH 提供二类审计程序去探查部分计划之间存在的交互作用:

- 冲突审计—审计部分计划之间的可能冲突。
- 互惠审计—审计部分计划之间的有益交互作用。

一般来说,通过对规划目标的层次分解以及对各层子目标的部分规划和汇合审计,可以灵活而有效地处理子目标间的交互作用,有利于支持复杂规划问题的解决。

17.5.3.4 非经典的自动规划技术

鉴于真实世界的复杂性,大多数实际规划问题(包括机器人行动规划)都不满足经典规划问题的假设,至使经典规划技术不能有效地用于解决它们。于是,研究者们开始了非经典规划技术的研究。非经典规划的研究,力图突破经典框架的限制,提出了基于记忆的方法、并行搜索、时间管理、仿真和纠错、局部化、计划重用等规划新技术。另一方面,为克服通用目的规划器固有的低效性,专用目的规划器的研制受到了普遍的重视。

非经典规划研究的一大亮点是面向动态变化世界的规划技术研究。研究者们对经典规划技术不满意的一个重要论点是经典技术的研究起源于、也主要面向机器人动作规划。其实,即便机器人也不满足经典规划问题的假设。现代机器人趋向工作于动态变化且不可完全预知的环境,必须不断地感知所处的环境并作出适当反应,所以将机器人行为建模为可完全预言的动作序列也是不合理的。实际上,美国麻省理工学院已在 20 世纪 80 年代中期就开发了不依赖于符号规划(基于符号推理的规划)的机器人。同期,若干面向动态变化世界的规划技术也开始问世,这些技术通过将感知(刺激)和反应直接耦合的方式,使机器人对环境的变化具有快速反应能力。不过耦合的设计主要由人完成,尚难以让计算机自动完成。

另一面向动态世界的重要问题是规划的实时性。对于需实时处理的问题(如监视传送带的工作),再好的问题处理计划,若不能及时规划出来也是没有用处的。一个称为 Anytime 的算法能在任意设定的时间间隔内提交尽可能好的行动计划,而另一个称为“思考调度”的算法则能通过量化估算,权衡超过时限获取更好行动计划的利弊。

较新的研究还有耦合经典规划的强度(制定行动计划的能力)和实时反应能力(处理动态出现的非预料事件),并导致了具有层次规划能力的机器人。这些机器人能基于下层实时动态规划技术作快速反应,又能基于上层思考型(经典)规划技术制定合理的行动计划。

专用目的规划器的研究试图通过限制适用范围,来提高规划效率和降低规划复杂度。这种规划器的极端是面向特别应用任务的规划系统,并可采用专家系统技术去解决特定的规划问题。这种规划器往往应用效率极高,但适用面也很窄。所以近几年来,具有较宽应用范围的专用目的规划器逐渐成为研究和开发的热点。

这种规划器的一个有影响的例子是运动规划，其旨在规划多维空间中的运动路径，例如给具有 6 个自由度的机器人规划运动路径。

另一个重要的应用域是调度，其可视为规划问题的特殊情况，旨在为执行一系列拟定的任务分配执行时段和资源。调度的研究起源于工业生产的需要，即使调度质量的少许提高也会显著缩减生产的成本。典型的调度问题有工厂车间的作业调度，交通运输调度和与项目管理有关的调度。调度问题往往有许多不同的表示形式，尤其在任务规模、资源消费和排序约束等方面差别很大。所以不可能设计出通用目的调度算法，必须为每类调度问题研制特别的解决方法。可以说启发式知识的应用几乎是必要的，各种人工智能技术都可应用于解决调度问题。调度和运动规划问题的共同之处是都难以解决，而且往往比经典的规划问题更难以解决。不过若不追求最优解答，只要求获得较好的解答，就可显著降低问题解决的难度。

17.5.3.5 自动规划技术的研究与未来发展

从一个完整的规划系统应具备的功能角度，理想的规划系统应由五个功能模块组成：① **操作符选择**—依据启发式知识，选择能最大限度地缩减当前问题状态与目标状态间差距的操作符。启发式知识可以是领域无关的，如 Green 方法使用的状态同一和框架公理抑制。然而最有效的启发式知识均是应用领域特有的，主要用于估算操作符的应用能使差距缩减的程度，以支持子目标排序（如 STRIPS）和动作排序（如 NOAH 的冲突审计）。② **操作应用**—应用选中的操作符计算其实例动作执行后的状态表示。例如 STRIPS 依据操作符定义的加表和删表修改状态的特性描述，而目标回归方法则计算回归到的上一状态的特性描述。③ **规划成功检查**—检查对于目标或子目标的规划是否成功。例如 STRIPS 通过检查仿真积木块世界是否满足规划问题目标状态的描述来确定规划是否成功，目标回归检查是否已回归到规划问题的初始状态来确定规划的成功，而 NOAH 则通过检查所有底层子目标相应的谓词公式是否存在于初始状态的特性描述来确定规划的结束。④ **规划路径失败检查**—及时检查出失败路径可大幅度提高规划的有效性。例如 Green 方法和目标回归方法都用状态约束去检查规划过程到达的不合法状态，以便及时回溯到其它规划路径。⑤ **解答修改**—对于复杂的规划任务，往往可以先忽略次要因素，待找到近似解答后再修正为完全正确的解答，常用的方法就是将复杂的规划问题层次地分解为准独立的子问题，忽略它们间的交互作用，分别求解（产生部分计划），然后再综合（包括修正）这些部分计划为完整的行动计划。HOAH 系统就是应用这种策略实现规划的典范。

规划实施时需要考虑的问题，一是**问题求解的要求**。规划问题本身就是一大类问题求解任务。此外，许多其它复杂的问题求解任务尽管不要求解答表示为动作序列，但也常需要对问题求解过程作规划。例如子问题有交互作用的任务，经过规划，可以让子问题按合理的次序加以解决。对于错误动作不可撤回的真实问题求解任务，例如实时控制，作适当规划尤为重要，因为一步走错就会造成灾难性的后果。尽管真实问题中的求解步骤不能撤回，但规划是在仿真世界中进行的，不受真实世界约束的限制，允许仿真动作撤回。二是**规划效应不可预测的应用域**。这些应用域的工作环境往往非静止，且对规划拟定动作的执行有动态的不可预先确定的反应。直观的解决办法是在作规划时，考虑到环境动态变化和反应的所有可能情况，但这样产生的行动计划肯定相当庞大，而实际执行时却只用到该计划中的极小部分，造成很大浪费。较好的策略是只按环境变化和反应的最可能情况，建立一个可能成功的行动计划。在执行计划时若遇到非期望的变化，就进行局部重规划，以修改原有的行动计划。三是**规划任务的分解**。鉴于真实世界的规划任务没有完全可预知的静态不变的环境，所以在行动

计划的执行过程中，为应付非预期的情况发生，修改计划不可避免。规划任务的层次分解，不仅能降低规划过程本身的复杂性，也能减少动态修改已建立行动计划（规划结果）的复杂性。假若有一个烤蛋糕的计划，其中包括将几个鸡蛋的蛋清和蛋黄分离的动作。若计划执行时不小心让一个蛋黄滑落到装蛋清的盘子中，则不应该重做烤蛋糕的规划，只需作一个补充规划：把蛋黄捞出即可。规划任务的层次分解，使得在行动计划中插入补充动作十分容易。四是**规划过程的记载**。为使上述计划修改工作容易进行，规划过程不仅应记载需加进行动计划的每个动作，将导致动作被选取的依据作为附加记载也很重要。例如，STRIPS的三角表，或其它方式的记载（如数据从属），使规划系统在行动计划的执行受挫时，能快速确定问题的根源，及时加以修补。五是**动态规划的运用**。对于规划效果不可预测的应用域，另一种策略是动态规划；即边规划边执行（摸着石头过河），一旦某动作或局部动作序列确定，就随即执行；然后，根据这些动作执行后的情况继续进行规划，直到成功。采用该策略的缺点在于一旦执行了不合适的动作，就须在撤回动作的同时，消除动作对环境的影响。作为对照，在先通过规划建立行动计划、再执行计划的常规策略中，规划过程仅在仿真世界进行，不会对真实世界产生不良影响。

自动规划是一个重要的具有实用价值的人工智能研究领域。鉴于实际规划问题的复杂性和多样化，目前尚缺乏解决它们的健全理论和方法论。一些值得进一步研究的课题包括：① 有转变能力的规划。有转变能力的规划主要体现于计划精化和规划路径回溯。前者旨在先建立抽象的部分计划，而后再逐渐精化为具体的计划；后者则通过回溯处理，将规划过程从失败的路径转变到成功的路径上。进一步的研究将是计划精化的理论框架和实现应用性更强的计划转变能力。② 对规划方法的分析。目前面向分析的方法论比较混杂。复杂度分析也主要以经验性的方式进行。为扭转这种局面，应进一步研究规划方法的分类，并确定适用的评价方式。研究适当的分析技术去对多种规划方法作横向的量化比较也是需要的。③ 基于范例和重用的规划方法研究。基于范例的规划方法和基于重用的方法已经分别得到了深入的研究。鉴于这二种方法的共同之处都是记忆已做的规划，综合这二种规划方法应该是很有意义的。④ 面向规划的学习。机器学习技术已开始应用于经典和动态（真实世界）规划系统，但仅限于简单的机器学习技术如加强（Reinforcement）学习，还不能自动获取新知识。新的研究工作应聚焦于集成更强的学习模型于复杂的规划领域，并沟通学习机制与基于范例和重用的规划器间的关系。⑤ 实时规划。规划的实时性和规划结果（计划）的质量是一种矛盾。但许多实时应用系统（如空中交通控制，核工厂监控，军事系统等）也同时要求高质量的规划结果。显然仅靠提高计算机速度是不够的，如何在限定的时间内制定尽可能好的行动决策，仍然是自动规划技术面临的重要挑战。⑥ 沟通反应型和思考型规划。同时具备快速反应的能力和预言未来行为的能力，是许多高性能系统所必需的，例如移动机器人，智能的信息筛选代理（Agent）和复杂的物理系统控制等。建立多层次规划系统（上层作思考型规划，下层作反应型规划）是正在研究中的解决办法，但不够成熟。如何实现反应型和思考型规划间的无缝连接仍是需要研究的课题。⑦ 规划的认识论模型。自动规划技术研究的成熟离不开对人类规划行为的研究，包括记忆和类比的作用，快速的未来规划，及时由更好的动作取代现行不合适的动作等。建立这方面认识论模型的研究已经开始，但需作大量的深化研究工作。

17.5.4 情感机器人与情感模拟

17.5.4.1 情感与情感系统模拟研究的概述

在过去多年的研究中，心理学家和生理学家提出了多个关于情感的理论和模型。这些理论和模型从不同的角度阐述了情感在人类个体日常行为和感官体验中所起到的关键作用。人与当前智能机器的最大区别，就在于前者具有产生情绪的能力。早在 19 世纪，情绪和情感就成为了心理科学研究的对象，然而，情感模拟在智能模拟中却一直处于被忽略的地位。直到最近三十年来，这种情况才发生了改变。不断发展的情绪理论开始进入人工智能的研究领域。越来越多的人工智能学者认识到，为了实现计算机“拟人化”的美好梦想，仅仅使其具有智能是远远不够的，还必须让计算机理解并具有类似人类的情绪，能与人进行情感的交互。我们可以预见，一个“拥有”人类情绪和情感的智能机器，将具有巨大的潜在的应用前景。

一般而言，人工智能中的情绪研究主要有两个大的目标：① 让计算机表达和辨识情绪；② 让计算机真正“具有”情绪，即智能主体能够根据特定的环境刺激产生恰当的情绪反应，使得智能主体更可信，也更具有社会性。这两大目标合而为一，统称为**情感计算**。这一概念是由 MIT 的 Picard 教授最先提出来的，其对情感计算下的定义是：“情感计算是指对与情绪有关的，由情绪引发的，或是能够影响情绪的因素的计算”。

可以看出，这个定义涵盖的内容相当广泛，它几乎包括了所有与情绪相关的计算过程。许多不同类型的研究都可以划归到这个领域，例如一些学者致力于让计算机模拟情绪影响人类解决问题或制定决策的过程，他们构造出一些能够执行情绪行为的智能主体或称为虚拟人。其核心部件是一个情绪的计算模型。它能够根据虚拟世界中的客观环境产生相应情绪，并对主体的决策和行为施加影响，从而使其活动方式更接近人类，而不是只有逻辑推理和计算能力的机器。另一些人则研究生理情绪信号（如面部或语音表达）的检测、感知或建模，以图大大提高人机交互的效率和质量。还有一些学者则关注在计算机网络媒介中人与人之间情绪的交流或分布式情绪。然而，在这些种类繁多的研究领域背后，其基础和根本则是对情绪的实质的理解和表示。

对生理情绪信号的研究已引起了不少人的兴趣，而它的核心部件则是情绪的计算模型和拟人主体的情绪行为的设计。关于虚拟主体的设计目前已经开展了大量的工作，它们主要是一些在虚拟世界中存在的软件智能体，可以按照人类的方式产生行为以及对他人行为的响应，从而实现与人或其它虚拟主体的交互。情绪模型被认为是通过与用户的情绪发生作用来实现更有效的人机交互的一个关键组成部分。

为了使计算机“具有”情绪，首先，我们必须理解人类自己的情绪是如何产生的。许多心理学方面的理论和模型都试图对这一过程做出解释，如刺激—响应理论、生理反应理论、面部表情理论、动机理论、主观评价理论等等。目前最为广泛接受的是情绪的认知评价理论。人工智能学者们对它也给予了最多的关注，许多情绪主体的构造都是基于某个评价理论的。根据认知评价理论，情绪是通过主体在产生情绪体验时对某个其主观上认为重要的事件进行评价而产生的。这种评价过程具有主观性，它取决于主体的特定目标、信念和规范等。不同的主体具有不同的内部心理结构，因此对于同一外部刺激的解释可能是不同的，最终所产生的情绪将依赖于他们各自对刺激的认知和主观评价。

情感机器人研究如今已有巨大进步，其典型的应用已包括虚拟客服、数字主播和服务人形机器人等。比如，（对话）服务机器人可从对方的声音中发现感情的微妙变化，然后通过自己表情的变化在对话时表达喜怒哀乐，还能通过对话模仿对方的性格和癖好；我国所开发的“脸部运动编码系统”可应用于人脸表情的自动识别与合成；“MPEG-4 V2 视觉标准”可

以组合多种表情以模拟混合表情；针对人的肢体运动设计了“运动和身体信息捕获设备”；开发的“语调表情构造系统”能根据语音的时间、振幅、基频和共振峰等，寻找不同情感信号特征的构造特点和分布规律；“可穿戴式智能设备”可用于增强和补偿人的感知功能。

17.5.4.2 情感与情感系统模拟已有的一些研究

情感与情感智能模拟的多年研究已提出了一些以一定的情绪心理学理论作为基础的计算机情感模拟模型和系统。在这些模型和系统中，每一个都有一些核心的思想和其独特的优秀元素，并被尝试应用于情绪模拟系统之中。

为了使计算机“自发地”产生情绪，研究者们曾使用过两种方法来模拟情绪的产生过程。一种是模拟刺激情绪发生的生理变化，这方面有代表性的工作是 Caámero 在 1997 年提出的系统。他构造了一个虚拟人类的生理系统来模拟人类的生理系统，通过人造荷尔蒙的变化导致情绪的产生。另外一类则是通过认知评价和推理过程来推理情绪的产生。目前，大部分情绪的计算模型是基于认知评价理论的，这类模型的引人之处在于它们可以较为容易的转换为计算机程序代码实现。根据不同的侧重点，这类模型又可分为两种：结构理论和过程理论。结构理论注重定义不同情绪的评价特征和模式，将与情绪相关的评价组织成一组定性的或定量的变量，称为评价维度，然后指出这些维度值如何组合来与不同的情绪相关联。过程理论则注重评价过程本身，将评价过程描述为一个特殊类型的命题推导过程。

1. 结构评价理论—已有的研究

(1) **OCC 模型**。目前最有影响力的认知情绪导出模型是由 Ortony 等人提出的 OCC 模型。它是第一个以计算机实现为目的而发展起来的模型。他们假设情绪是作为一个称为评价的认知过程的结果而产生的。评价取决于三种成分：事件、主体和对象。客观世界中的事件根据主体的目标被评价为满意的 (pleased) 或不满意的 (displeased)；主体自身或其他主体的行为根据一组标准的集合被评价为赞成的 (approved) 或不赞成的 (disapproved)；对象则根据主体的态度被评价为喜欢的 (liked) 或不喜欢的 (disliked)。由这些评价中的变量产生了一个包含 22 类情绪的层次结构。

OCC 模型为我们提供了一个情绪的分类方案，并给出了这些情绪类型之下的潜在的推理过程。它提供了一个基于规则的情绪导出机制，可以有效地通过计算机进行模拟。它没有利用心理学中普遍采用的基本情绪集合或一个明确的多维空间来描述情绪，而是使用一致的认知结构来表达情绪。基于 OCC 模型，Elliott 实现了一个称为情感推理机 (Affective Reasoner) 的系统，Reilly 也构造了一个具有可信性和社会性的情绪主体系统用于产生交互式戏剧。

(2) **Roseman 的评价理论**。Roseman 的认知评价模型是一个较为完善的理论体系。模型设定了一些认知的维度来决定一个情绪是否产生以及产生的是何种情绪。模型的修改主要是对这些认知维度的细节的修订。最初的模型包含五个维度，通过它们的相互结合来推断产生哪种情绪：① 主体是否拥有一个动机，来接近期望发生的情境状态或远离不期望发生的情境状态。② 情景是否与主体的动机状态相一致。③ 一个引起注意的事件是确定的还是不确定的。④ 主体感知这一事件是应得的还是不应得的。⑤ 事件是由谁引起的一是环境、主体自身还是其他主体。由此，Roseman 定义了一个多维度的情绪空间。一种情绪可以看作是该空间中的一个特定的点。然而这种情绪空间理论无法解释一些复杂的情绪如悲喜交集。这种情况下主体对同一情境同时做出了两种不同的评价，因此映射后的情绪空间就无法解释一个点为何同时出现在两个位置。

(3) Scherer 的模型。Scherer 定义了一个更大的系统。它不仅包含认知因素，还综合了其他心理学的成分。五个功能子系统包含在情绪过程中，通过交互作用产生情绪。系统首先通过感知、记忆、预期及对获取的信息的估计来评价刺激因素，然后通过控制神经内分泌、身体以及自治的状态来调整内部条件，据此做出规划、行为准备以及在竞争的动机之间进行取舍，然后控制运动部件将可见的行为表达出来，最后控制注意机制转移到当前状态上，并将结果反馈给其他子系统。其中第一步，即信息处理子系统就是基于认知评价的，称为刺激评价检查（SEC）。这些检查的结果又会引起其他子系统的变化。检查共可分为五种类型：① 新颖性检查：确定外部或内部刺激是否发生了改变。② 内在愉悦度检查：确定刺激是否是令人愉快的，并引发相应的趋近或回避倾向。③ 目标重要性检查：确定一个事件是帮助还是阻碍了主体目标的实现。④ 应对能力检查：决定主体认为事件可以控制的程度。⑤ 相容性检查：最终比较事件与主体的内部或外部标准的符合程度。

Scherer 的模型与 Roseman 的模型在机制上有一定的相似性，只是定义了不同的认知评价维度，通过五种类型的检查相互作用来确定最终产生何种情绪。Scherer 的模型优点在于它具有对行为进行选择的能力。行为选择通过动机互相竞争、规划和情境处理模块来实现。

根据上述理论，Scherer 最终构造并实现了一个评价理论的计算模型。这个模型的本质是一个基于知识的系统，提供了大量真实世界中的场景实例。系统的输入是一个情景描述，分为 15 个评价维度，然后与某个根据 14 个原型成分定义的情绪进行匹配。评价维度和情绪维度都是用特征向量来表示的，匹配过程是计算输入向量和目标情绪向量之间的欧拉距离。但是这种向量空间的方法有很大局限性，因为当匹配不成功时，系统给出的结果是“没有情绪发生”，这显然不符合真实情况，而且不能够将几个评价维度结合起来映射到一个单一的情绪上，无法表示复杂的情绪。

(4) Frijda 的理论。Frijda 的理论是以“关注”（concern）为中心的。一个“关注”是系统的一个倾向，希望使环境中或主体自身组织产生某些特定的状态。关注决定了系统的目标和偏好。系统在一个不确定的环境中可能产生多个关注。当一个情境产生，使得这些关注的实现受到威胁时，就会出现所谓的“行为倾向”。这些行为倾向与情绪状态密切相关，如“趋近”的行为倾向产生“希望”的情绪，而“回避”的行为倾向产生“恐惧”的情绪，等等。他还给出了一个功能性的情绪系统必须具有的九个组成部分。

Frijda 的模型定义了 15 个由行为导出的情绪，同一时刻可能存在多个关注。他的模型将重点放在了规划和行为上。这一点对于构建自治主体非常有益，可以很容易地将情绪及其对行为的影响表示出来并加以解释。

2. 过程评价理论—Reisenzein 的研究

Reisenzein 认为目前为止过多的研究重点都放在了情绪的结构理论上，而很少有人关注评价过程本身。因此，他将评价过程同结构理论区分开来。评价过程是通过被评价物体的其他信念来构造评价信念的过程。从认知角度来看，结构理论是以描述评价的说明性语义为目的的，而过程理论完善了对过程语义的描述。

Reisenzein 区分了中心和外围两种不同类型的主要评价过程。外围评价过程的组成部分有：（1）计算信念强度和期望强度的过程，作为中心评价过程的直接输入；（2）不同类型的其他评价过程，用于辅助可能性或期望强度的计算。这二者构成了对一个焦点事件的可能原因和结果的评估，并确定了当前事件与社会和道德标准所符合的程度。中心评价过程包含对期望一致性和信念一致性的检查，这是一个持续不断的并发的机制，监控新获得的信念

和期望与已有的信念和期望的相容程度。它们感知到主体内部状态，并对系统中重要的变化用信号标记。如果信号超过了一个特定阈值，就产生了干预过程，将注意集中于特定的输入上，并重新设定期望和信念的强度值，从而产生特性化的意识体验，即情绪。

总的来说，Reisenzein 从一个全新的角度给我们呈现了一个动态的、自循环的信息处理过程。它看起来更自然，也更接近人类的真实的情绪产生机制。他的评价过程可以看作是主体（agent）的内部运行机制。他还考虑了核心评价过程的结果对其他评价过程的影响，这在以前的许多模型中是从未提到过的。

3. 计算机情绪模拟系统

以上模型总结了一些典型情绪的产生原因及引发因素，并从理论角度对一个情绪系统的构成成分做出了分类。这些模型背后的运行机制是认知评价理论。下面，我们将给出一些（曾经的）具体的模拟系统，它们大部分是基于规则机制的，各自实现或部分实现了上述的那些模型。

(1) **Elliott 的 Affective Reasoner**。Elliott 的系统可以看作是 OCC 模型的一个计算机实现。这个称为 Affective Reasoner（简称 AR）的系统是一个计算机模拟器，可以在一个多主体系统中进行情绪的推理。它是基于 OCC 模型的一个扩展假设来设计的，共有 24 种不同类型的情绪，每一种情绪都由一组不同的认知导出条件通过推理得出。

Elliott 认为一种研究情绪推理过程的方法是模拟一个由主体（agent）所处的假想世界，主体能够参与情绪场景。因此 AR 是在一个多主体世界的环境中发挥作用。每个主体有一组用符号表示的评价框架，包含主体的目标、偏好、行为准则以及当前的心境。某个情境在它们的世界中产生，主体基于各自的评价框架来解释这些情境，情境中的变量与这些解释相关联，这些关联决定了产生的情绪及其强度。然后在这一特定情境中，一个确定的情绪行为就被引发了。主体可以同时具有多种情绪，甚至可以是相互冲突的。它们不但可以根据情境自己产生情绪，还可以根据其他主体的行为来推导它们的情绪。

Elliott 的系统的最大贡献就是使用了显式的评价框架，根据特定的评价变量来对事件进行特征化，从不同的角度（主体自己或其他主体）来对同一事件进行评价和解释。另外他还显式地归纳了一组能够影响情绪强度的变量。在变量的值和情绪的模拟强度之间架设了一座桥梁。但有些变量之间的差别过于细微，而且变量之间还存在着相互依赖，所以有时需要更复杂更精确的函数来描述这些过程。

(2) **Reilly 和 Bates 的 EM 系统**。Em 是作为一个更大的系统 Oz Project 中的情绪模块来发挥作用的。Oz 的目标是构建处于虚拟现实世界中的自治主体，从而实现一个交互式的戏剧系统。系统希望实现一个与其他系统相比更广泛且更浅显的主体（agent）架构。当大多数情绪计算的 AI 系统致力于某个具体的方面且试图将其概括得尽可能精细时，Reilly 和 Bates 采用了相反的方法：他们将精力集中于产生拥有更广泛能力集合的主体上，包括目标导向的反应行为、情绪状态和行为、社会知识和行为以及自然语言能力。每一个能力都都很有限，但对于构造广泛的集成化的主体来说都是必须的。

Em 是作为这个系统的一个模块嵌入其中的，用于产生更真实可信的社会行为。它根据是否喜爱的特征、行为和当前目标解释了对象、其他主体和环境中的当前状态，然后引发相应的情绪状态（如当目标实现时就会高兴，接近一个自己喜欢的对象等等）。通过引入一个累积的阈值机制来确定哪种刺激会产生一个情绪反应。

Em 的优点是虚拟主体的情绪强度可以随时间逐渐减弱或持续一段时间，这将取决于情

绪的类型和实际的情绪提取情境，与真实世界中相类似。这就赋予了每个主体一个及时更新自己情绪状态的机制，从而可以进行持续不断的交互。

另外在 Em 中，引入了动机和感知模块作为情绪产生的前提的一部分，从而使得情绪的产生机制更为简洁，且不仅仅依赖于认知结构。

(3) **Gratch 和 Marsella 的 EMA**。EMA 探讨了情绪对认知的影响，以及认知如何作用于情绪。系统使用了评价理论来描述主体是如何评价一个情境的，以及结果的情绪是如何在主体中表示出来的。因此，情绪响应不是作为感知系统的直接输入的结果，而是基于主体对当前情境的解释，它可以包含任意的认知过程甚至记忆中的前一个情境的回放。类似于 Elliott 的 AR，EMA 使用了显式的评价框架，对事件根据特定的评价变量进行特征化，然而他们的评价不仅包含当前条件，还包含可以导致当前状态的过去的事件及对将来的预期。系统主要由两个不断重复循环的步骤组成：主体对当前情境进行评价导致情绪产生，然后有选择地对这种情绪进行应对：或者改变自己的评价（集中关注于情绪：emotion-focused），或者改变自己所处的情境（集中关注于问题：problem-focused）。应对决定了主体对事件进行评价后如何行动，或维持希望的状态，或改变不希望的状态。这两种应对本质上代表了两种相反的评价方向，确定了一个情绪的原因解释，从而决定它应该被维持还是改变。

正如我们所看到的，EMA 不是直接对环境中的事件进行评价，而是将它们解释为记忆中的目标、信念、规划和意图，由一个评价框架对其进行评价。这使得 EMA 避免了大量的与某个具体领域相关的评价规则，而这在前面的一些方法中则是必须的。EMA 将一个评价看作从解释原因所具有的与具体领域无关的特征到个体的评价变量的映射。与具体领域有关的信息则限制在操作算子描述中，作为规划建立的基础。另外，EMA 中给出了事件的定义，将时间看作任何在原因解释中表示的物理行为，它可以帮助或阻止一些对主体效用不为零的状态，这在以前的系统中是没有的。

(4) **Sloman 的 CogAff**。Sloman 的情绪理论在它的 CogAff 系统中得到了实现。CogAff 是基于两个概念的结合。一是这个系统分为三个部分：输入（称为感知），中心处理，输出（称为行为）。二是每一个部分都分为三个不同层次，使得系统可以在不同级别上进行抽象：反应机制，它将外部或内部状态直接映射为行为或情绪，而不经过程序评价过程；意图机制，它对情境进行推理，做出规划，并理解行为的序列，从而通过认知过程产生情绪；元管理机制，它使得主体对不同内部状态的意识 and 评价成为可能。这一体系的不同层次概括了不同的情绪类型。层次之间的交互和竞争控制导致了更为复杂的情绪。

CogAff 的一个优点是它引入了一个先进的自我控制过程作为最高的层次，它的作用类似于心境。这种自我控制机制监控整个内部状态，并为将要发生的情绪趋向提供了一个背景和前提。

CogAff 的另一个优点是它提出了“扰动”（perturbance）的概念，将不同类型的情绪统一起来。Sloman 认为情绪通常伴随着一种称为“扰动”的状态，它可以被看作是一种失衡的状态。如果整个系统部分失去控制，即当目标实现的动机受到了拒绝、阻碍或推迟时，就会发生这种扰动。

(5) **Velásquez 的 Cathexis**。情绪不仅由简单的推理构成，还由一些低层的非认知性因素影响产生。Izard 指出了关于非认知性因素对情绪产生的影响问题，并提出人类四种类型情绪发生器的构想。MIT 的 Velásquez 据此提出了 Cathexis 情绪产生模型。Cathexis 系统由专门主体（称为 emotion proto-specialist）构成，每个专门主体代表一类情绪，对

输出行为施加影响。与 OCC 模型中每种情感都对应不同的规则所不同的是, Cathexis 模型的核心规则只有一条, 也就是它的情绪更新规则。

17.5.4.3 情感机器人研究的意义

目前, 理论界普遍认为, 研制情感机器人以及实施人工情感的主要目的在于: 使机器人具有“人情味”, 从而创造和谐友好的人机环境和制作进行情感交互的计算机。实际上, 情感对于机器人的重要作用远非如此简单。情感机器人的科学价值与现实意义意义深远。数理情感学认为: 情感的哲学本质是人脑对于事物的价值关系的主观反映, 情感的客观目的在于引导人针对不同价值的事物采取不同的选择倾向性, 从而对有限的价值资源进行合理配置, 以达到最大的价值增长率。可以认为, 情感不仅是“人情味”的主要来源, 也是思维效率性的主要来源, 还是行为自觉性、思维创造性、社交世故性、人格自尊性等人性特征的主要来源。具体而言, 赋予机器人以情感, 将会有五个方面的重要意义:

(1)**建立人性化人机界面。**机器人通过情感识别系统来识别他人的价值需要或主观愿望, 通过情感表达系统来表达自己的价值需要或主观愿望, 通过情感内部逻辑系统进行价值计算与价值判断, 通过意志内部逻辑系统形成恰当的机器行为, 从而与操作者形成亲切友好的交互。人在操作使用机器人时, 如果机器人赋予了情感, 就能够对人的面部表情、自然语言、身体姿态及对键盘和鼠标的的使用特征等进行观察, 以识别和理解人的情感, 并通过图像、文字、语音等做出智能而友好的反应, 产生生动而真实的使用环境, 帮助使用者获得高效而亲切的感觉, 形成自然而亲切的交互, 营造真正和谐的人机环境。机器人对于不同性格利益相关性、价值取向、情感特征、个人爱好和专业特长等做出不同的反应, 也有利于使用者掌握其性格脾气和功能特性, 有利于在智能玩具和游戏中构筑拟人化的风格和更加逼真的场景。系统还可以通过对不同类型的用户建模(如操作方式、表情特点、态度喜好、认知风格、知识背景等), 以识别用户的情感状态, 并以适合的方式呈现信息, 在对当前的操作作出即时反馈的同时, 还要对情感变化背后的意图形成新的预期, 并激活相应的数据库, 及时主动地提供用户需要的新信息。目前, 人们研制情感计算机的目的首先在于建立和谐而友好的人机交互环境, 使机器人能够对人的面部表情、自然语言、身体姿态以及对键盘和鼠标的的使用特征等进行观察, 以识别和理解人的情感, 并通过图像、文字、语音等并做出智能而友好的反应, 从而产生生动而真实的使用环境, 帮助使用者获得高效而亲切的感觉, 形成自然而亲切的交互, 营造真正和谐的人机环境, 以达到降低劳动强度和提高工作效率的目的。

(2)**提高思维的效率和速度。**情感的客观目的在于区分不同事物的重要性顺序, 以引导人来合理地配置价值资源。当人在进行思维时, 自觉不自觉地情感的引导下对思维过程的相关事物采取不同的选择倾向性, 使这一思维过程达到最大的价值效益。电脑或机器人一旦赋予了情感, 就可以按如下的价值特性顺序要求来生产、吸收、组织和和使用信息, 从而可以简化思维过程, 节约思维资源, 加快思维速度: 一是价值重要性顺序, 如主体将会优先发现、优先识别、优先摄取和优先使用具有较大价值量或价值率的信息; 二是时间紧迫性顺序, 如根据事物的变化时间紧迫性的顺序决定逻辑推理的过程是采取模糊、粗略而简明的方法, 还是采取精确、严密而详尽的方法; 三是利益倾向性顺序, 如对于有利于(或不利于)维护自身利益的理论观点, 主体总会优先发现支持(或否定)其成立的依据和证据; 四是利益相关性顺序, 如在对于事物的起因、现状和发展方向进行分析时, 主体总是力图把自己的作用、自身的经验、自身的利益(或命运)与该事物联系起来; 五是价值连续性顺序, 如对于过去不友好的电脑使用者通常提供不友好的使用界面; 六是价值变化的敏感性顺序, 如对于陌生事物的观察与分析以及对有关信息的处理, 采取较为谨慎的态度, 以防虚假信息出现或病毒的感染; 七是价值知识的经验性顺序, 如通过综合分析电脑使用者的相貌、体形、操作方式、操作内容、交往对象(或网站)、工作时间等, 可对他的能力、性格、职业、爱好等进行经验性判断, 从而为其提供高效、快捷的界面服务; 八是价值特性的关联性顺序, 如在对

事物的起因、现状和发展方向进行分析时，主体将会参照与该事物存在价值关联性的事物的起因、现状和发展方向。总之，有了情感（或意志）的引导与调控，主体的思维（或行为）就会具有明确的目的性和方向性，从而显著地提高思维（或行为）的效率和速度。这就是为什么人脑能够在许多方面超过电脑的重要原因。人脑相对于电脑，无论是运转速度，还是内存与硬盘容量，或者计算差错率等都远远低于，然而，人脑的优势非常明显，这就是因为人脑能够按照诸如重要性、紧迫性、倾向性、相关性、连续性、敏感性、经验性和关联性等“特性顺序参量”来生产、筛选、组织和使用信息，从而可以简化思维过程，节约思维资源，加快思维速度，减少思维差错。

（3）**赋予机器人以行为上的自觉性。**统一价值论认为，人的自觉性是指人有明确的行为目的，有坚定的信仰追求，有鲜明的原则立场，有毫不含糊的是非标准，人的行为活动自始至终都有预先设置的、明确的、稳定的目标指向，使人的随意行动具有明确而强大的约束力，使其不至于成为漫无边际的、盲目的、无规律的活动，它反映了人的价值判断标准的稳定性。智能机器人由于没有赋予情感，就没有行为上的自觉性，因此只能根据既定的程序、方法和手段，毫无选择地、毫无灵活性地完成既定的工作任务，最多也只能根据已经预计的某些特殊情况，个别地、机械地变更其程序、方法和手段，不能自主地确立和调整价值目标，不能创造性地制订和修改总体规划及行为方案，不能总结经验和吸取教训。智能机器人一旦赋予了情感，就能够以达到既定的意志目标为行为方向，以内设的价值观系统（或情感系统）、认知系统和意志系统为价值计算的依据，以实现最大价值率为行为准则，建立一系列价值计算的函数关系式或约束方程式，再根据机器人所处的自然环境和人文社会环境确定若干个边界条件，选定情感和意志的动力特性参数，就可以根据既定的意志目标确定一个相应的整体规划、行为方案和具体动作，并在实施过程中积极地、主动地、创造性地进行调整，然后对行为的最终结果进行价值评价，以便及时地修正价值观系统（或情感系统）、认知系统和意志系统，达到总结经验和吸取教训的目的，所有这些过程都将在预置的、有序的、明确的、坚定的状态下完成，这就形成了机器人在行为上的自觉性。

情感机器人能增强（主体）行为的主动性。若说智能机器人只能按照人预先编制的固定程序进行动作，则情感机器人能够以“达到既定的意志目标”为行为方向，以内设的“价值观系统（或情感系统）、认知系统和意志系统”为价值计算依据，以“实现最大价值率”为行为准则，建立一系列价值计算的函数关系式或约束方程式，再根据机器人所处的自然环境和人文社会环境确定若干个边界条件，选定情感和意志的动力特性参数，就可以主动地、创造性地调整“整体规划、行为方案和具体动作”，然后对行为的最终结果进行价值评价，以便及时地修正价值观系统（或情感系统）、认知系统和意志系统，达到总结经验和吸取教训的目的。

（4）**赋予机器人以思维上的创造性。**统一价值论认为，思维的创造性在本质上就是信息的创造性，信息是价值产生增值的根本源泉，如果没有信息的注入和信息产生，任何生产系统都最多只能进行价值的形式转换，而不会产生任何的价值增值。一个生产系统的价值率如果大于1，那么它就有信息的存在，价值率不断下降的过程，实际上就是信息不断流失的过程，或者说，就是“信息”不断转化为“常识”的过程；一个生产系统的价值率如果小于1，那么它就有负信息的存在，价值率不断上升（或回归为1）的过程，实际上就是负信息不断消失的过程，或者说，就是“负信息”不断转化为“常识”的过程。信息的原始积累来源于生物进化，虽然这一过程非常缓慢和微弱，但它形成了价值的原始基础，构成了人类生存和发展的前提条件，人类社会之所以高速发展，关键在于信息和知识的高速积累。劳动之所以被确认为是价值的唯一源泉，并不是因为抽象意义上的定义，而是因为劳动在信息的形成、传播、处理和运行过程中起着决定性作用。人类的创新过程在实际上就是信息的生产过程，在本质上就是价值的创造过程，一个人只有不断增强其创造性，才能真正地、有效地提高其劳动能力，一个社会只有不断地提高其创造性，才能为社会财富的增值提供不竭的动力。机器人赋

予情感后,就能够在其情感内部逻辑系统和意志内部逻辑系统的引导下,一方面不断变更和修改整体规划、行为方案和具体动作,另一方面也不断修改和变更认知内部逻辑系统、情感内部逻辑系统和意志内部逻辑系统,以使自己的价值率尽可能达到最大值,这就形成了创造信息、创造新事物的能力。如果某一事物在修改和变更以后的价值率得到了上升,那么,该事物中就携带了一定的新信息,就是一种创新的事物。

(5) **赋予机器人参与社会事务和开展人际交往的能力。**统一价值论认为,价值问题是人类社会普遍存在的问题,社会和个人所追求的物质目标和精神目标都是价值目标的具体体现。人与人的一切(任何形式的)社会关系(如经济的、政治的和文化的关系)在本质上都是价值关系(或利益关系),利益关系的处理是社会事务(社会管理或社会服务)和人际交往的核心内容;而情感的客观目的在于正确反映主体所拥有的价值关系,并为主体调整其价值关系提供决策依据和行为驱动力,机器人一旦赋予了情感和意志,就能够在复杂的环境条件下,了解和猜测他人的价值取向、主观意图和决策思路,就可以像人一样独立自主地、应对自如地处理它所拥有(或赋予)的价值关系。

一切形式的目标管理在本质上都是以特定价值内容为主导方向的价值管理,人类社会的一切矛盾都可归结为利益关系的矛盾,社会结构的一切变革实质上都是利益关系的调整与重构,人际交往的价值本质就是各种形式的利益交往,总之,利益关系的处理与调整是社会事务(社会管理或社会服务)和人际交往的核心内容,而这一点正是通过情感系统和意志系统对人的思维与行动进行引导和控制来实现的,因此赋予情感与意志是机器人能否参与社会事务和人际交往的关键。情感的客观目的在于正确反映主体所拥有的价值关系,并为主体调整其价值关系提供决策依据和行为驱动力。机器人一旦赋予了情感和意志,就具备了参与社会事务和开展人际交往的能力,就可以像人一样应对自如地处理其所拥有(或赋予)的各种复杂的价值关系,有效地参与复杂的社会事务和人际交往活动,就能够在复杂的环境条件下,了解和猜测主人的价值取向、主观意图和决策思路,灵活性、积极地、创造性地实施各种行为,从而使其运行过程具有更明确的目标性、更高的主动性和更强的创造性,圆满完成主人交给的各种复杂的工作任务,从而在更大的工作范围取代人。

只能简单地模拟人的某些情感表达方式(如哭、笑、愤怒等),只能对人的一些简单的情感表达方式进行机械地模式识别的机器人,还不是真正意义上的情感机器人,而只是一种表面形式上的情感模拟。机器人只有同时具备情感表达系统、情感识别系统、情感内部逻辑系统(即情感计算系统)和意志内部逻辑系统(即意志计算系统),并使之组成一个完整的、有机的整体,才能认为它具有了真正意义的类人情感机器人。

17.5.4.4 关于情感模拟与情感机器人研究的一些考虑

情绪研究已经成为近年来心理学和计算机科学的热点研究领域,它具有高度的综合性和实用性。通过将计算机科学与心理科学、认知科学相结合,研究人与人交互过程中的情绪特点,将有可能设计出具有情绪反馈的人机交互环境,真正实现和谐自然的人机交互。

然而,目前情绪研究仍面临着许多挑战,包括:情绪信息的获取、表示与建模。如情绪信息应如何存储及表示;对能够影响情绪的自然场景如何建模;人性化和智能化的人机交互如何实现,如何将大量的广泛分布的数据进行整合,以产生针对不同用户的个性化的情绪反应等等。

(1) **关于情感的理论认知。**理论的高度最终决定着生产与技术的发展极限,有人认为,目前我们尚无法研制出真正意义的情感机器人,主要是因为存在着一些理论认知上的不足。比如,关于握情感的本质,有人认为情感是“人对于客观事物是否符合人的需要而产生的态

度的体验”。也有人认为，情感的本质就是人脑对价值关系的主观反应，情感与价值的关系就是主观与客观的关系；情感所反映的客观内容，应是人的价值关系；而情感的表现形式，应包括着情感的大脑体验强度、情感的生理指标和情感的表达方式三个方面。关于情感计算，一些人只是孤立地、狭义地去理解情感计算。须知，认知、情感与意志是人类三种基本意识形式，分别对应的反映着三种基本的客观存在：事实关系、价值关系和行为关系，三者相互作用、相互渗透、相互转化，才共同完成了人脑对于客观事物的反映功能。仅仅进行狭义的情感计算，并不能解决人的心智活动的全部计算问题，还需要实施对意志的计算，并实施对知情意的交互计算。另外，不少人过低地估计了情感机器人的科学价值。认为研制情感机器人的主要目的是为了使计算机具有友好的人机界面，须知，一旦赋予了计算机以类人式的情感，机器人就可以从事人类所能从事的众多工作，包括社会管理与人际交往等。

(2) **关于情感模拟的逻辑思路。**有人认为，情感的客观目的在于识别事物的价值关系，为人的各种活动提供明确、有序和恰当的行为驱动力，以实现价值资源的最佳分配和最大增长率。情感的内部逻辑系统实际上就是价值关系的内部逻辑系统，要实现对于情感的计算，就必须首先实现对于价值的计算，因此，以此为目的的“情感计算”在本质上就是“价值计算”。而研制情感机器人，一是要创立“统一价值论”，为实现价值计算、情感计算奠定理论基础；二是创立“情感数理理论”，为建立“情感内部逻辑系统”奠定基础；三是要开发“情感软件”和“知情意交互软件”，为实现机器人的“情智融合”奠定基础。如此，所开发和创造的“情感机器人”，才可实现真正的“人机融合”。

(3) **关于海量情绪数据资源库的构建。**由于人类情绪具有的复杂性和模糊性，虽然目前已建立了很多情绪的计算模型，但它们大多偏重针对部分典型情绪进行辨识和分析，并通过假设少量的离散情绪以及小范围的情绪变化幅度来简化模型。这些工作对于情绪研究十分有益，然而在赋予计算机理解和解释人们在真实世界中面对复杂多变的外界环境时所产生的各种各样的情绪现象的能力方面它们还远远达不到要求。有人认为，这种能力必须依赖于大规模的关于真实世界中人类情绪行为的常识性知识。这种常识性知识的缺乏将大大限制智能系统与用户自然和谐地进行交流的能力。因此，建立一个系统化的、结构化的、可操作的包含了各种各样关于情绪产生的形式化常识知识的知识库将是很有必要的事。这种情绪知识将是人类知识的一个特殊组成部分，它在人机交互、情感计算、自然语言理解和心理分析中都有重要应用。目前，已有人建立了基于角色的包含了人类各种情绪常识的知识库和情绪产生场景的信息库。根据社会心理学理论，人类的心理活动与其在社会生活中所扮演的角色密不可分，这种角色常常决定了一个人的目标、期望、规范、信念等心理结构。因此，根据角色将情绪常识获取并分类存放在知识库中，并结合场景信息库建立情绪的推理机制，将能够推导特定主体在特定情景中的情绪反应，并对其做出合理的解释。这种知识库的应用前景是十分广阔的，比如，构造可以自动生成故事的智能主体或将会成为它的主要用途之一。

(4) **关于情感计算。**情感计算认为，情感的变化是具有统计规律的，围绕对情感的统计规律的描述，可以计算出情感的概率转移矩阵，从而构造出一个具体的情感模型。如此实现的情感交互将不同于以前的基于行为模式的方法，因为此种情感将表现出一种具有概率性分布的行为。若据此将情感状态划分为不连续的状态从而构成情感空间，其熵值将可表达出所构造的情感的细腻程度；若进而将情感状态在情感空间内转移，将可视为是一个马尔可夫过程。如此，通过对情感状态概率转移矩阵的构造，可以得到每个情感状态的概率分布，进而计算出情感的熵值和转移过程。情感熵为情感状态的转移提供了一种可操作的运算量度，

但是，如何构造概率转移矩阵还缺乏一个统一的标准，进行情感空间的划分和转移还未有可靠的理论基础和普适性的途径。

(5) **情感机器人的最终归宿：人机一体化？**情感在人的思维活动中占据极为重要的地位，决定和制约着人的行为活动和其它思维活动的基本框架与总体方向。赋予计算机或机器人以人类式的情感，使之具有表达、识别和理解喜怒哀乐的能力，是许多科学家的梦想。与智能模拟的发展相比，人工情感技术所取得的进展至今仍是微乎其微的。情感，始终是横跨在人脑与电脑之间一条难以逾越的“鸿沟”。有人认为，在很长时期内，情感机器人只能是科幻小说中的重要素材，很难纳入科学研究的课题之中。所有这些，都局限于两个方面：一是情感的模拟表达，二是情感表达的模式识别。事实上，真正具有类人式情感的机器人必须具备三个基本系统：情感识别系统、情感内部逻辑系统和情感表达系统。因此，能否建立“情感内部逻辑系统”是研制情感机器人的关键。但我们相信，随着心理学理论的不完善和发展，以及计算技术及智能模拟能力的不断提高，情绪研究必将越来越引起更多研究者的关注，并最终会设计出真正具有情绪的智能机器。到那时，人工情感的全面实现不仅可以使计算机具有友好的、人性化的人机界面，更重要的是能够使机器人或智能机具有更高的信息处理速度与效率，具有独立的决策能力和行为控制能力，具有创造性和开拓性的思维能力。有人更乐观的设想，也许，到那时，从纯逻辑的角度来看，人与机器人之间也许不再有“原则性”的区别了，只有机器体与肉体之间的不同；人与机器人之间很可能会实现全面的融合：一方面，机器人的一些“部件”（包括思维“部件”），可以实现“肉体化”；另一方面，人身上的一些“部件”（包括思维“部件”），可以实现“非肉体化”。更有人幻想，如此，或可实现机器人与人的相互转化：例如，一个人的肉体老化后，可以将其大脑中所有的认知、情感与意志方面的信息提取出来，输入机器人的大脑中暂时“贮存”下来，并由该机器人代为本人继续行使有关的社会职责，等本人的“克隆”体制作完成后，再把机器人的大脑中的有关信息移植到过来。

总之，未来的情感机器人与人类将会越来越接近。它们各有所长，分别从事着不同的社会活动，扮演着不同的角色。他们彼此相互协作、相互促进，共同推动着社会的发展。

17.5.4.5 情感机器人研究可能的社会影响

有人认为，机器人一旦被赋予了类人式的情感，其性质将会发生重大变化。它若因此而具有了独立的行为能力，则必然会对人类的社会生活产生深刻的影响。比如，这些影响将会体现在：

① **伦理观念的变迁。**由于情感与意志的赋予，机器人与人之间的界线将会越来越模糊。由于机器人具有了“人性”，参与了社会事务与人际交往，人应该如何对待机器人，如何处理人与机器人以及机器人之间的关系，如何评价机器人所取得的成绩，如何看待机器人的缺点和错误，将会成为需要考虑的问题。机器人作为“二等公民”，应该如何确立其“社会地位”，如何看待人与机器人以及机器人之间的“感情”问题，也将成为问题。

② **法律关系的调整。**机器人的行为效应（如违法犯罪后果、社会与经济收益等）应该由谁来承担，是研制者、制造者、控制者还是所有者？机器人如果杀了人，应该如何处理？是全部拆卸或分解，还是重新调整程序？如果机器人被人所“杀”，人应该如何承担法律和经济责任？一方面应该如何使机器人“遵纪守法”，另一方面应该如何维护机器人的“合法权益”？人与机器人、机器人与人之间的经济与法律纠纷应该如何处理？机器人是否具有选择权和被选举权，是否可以“竞选”？机器人是否有“社会地位”？我们是否只是生产了一批“机器奴隶”？

③ **经济结构的调整。**情感机器人能够参与社会事务和人际交往以后，就会在越来越多的社会管理领域、生产领域和生活服务领域取代人，机器人将会成了一支越来越庞大的“劳

动大军”，那么，机器人的机体制造厂、软件开发公司、程序调整中心、机器人医院、机器人美容店、机器人餐馆等行业将会迅速发展起来，社会的生产结构和经济结构无疑将会出现重大的调整。

④ **生活方式的变更**。随着机器人越来越多地替代了人类从事简单重复性的工作、恶劣环境和高强度的工作，人类将会主要从事自由性、自主性、创造性和复杂性较强的工作，其工作时间的随意性、工作地点的游动性、工作内容的自主性、工作报酬的随机性和工作方式的选择性将不断提高，必然会使人们的生活内容和生活方式发生着深刻的变化，包括个人消费、人际交往和家庭结构方面的变化等。此外，人将会越来越多地与机器人打交道，家庭保姆将会大规模地进入家庭，并具有了越来越强的心理功能和精神功能，那时，人们打电话时的第一句话将会是“请问，你是人吗？”或“请你的主人接电话”吗？无疑，这不是科幻，而是未来的现实。

⑤ **社会隐患的增多**。更有人担心，若机器人具有了情感以后，能够进行独立思考和自主行为，由于其信息的处理速度快，信息的贮存量大，运转的准确性高，在许多方面会具有比人类更多的优势，他们一旦“哗变”，其后果也许不堪设想，其灾难性也许不会亚于核武器的大规模引爆。这也许是“杞人忧天”，但机器人参与社会生活以后，社会矛盾将会日趋复杂化；机器人进入家庭（如家庭保姆、健康顾问、精神陪护等）以后，由于被赋予了心理功能和精神功能，家庭矛盾也将会日趋复杂化，却是不争的事实。考虑这些问题，也许为时尚早，但如何处理这些问题，却是我们最终不得不考虑的。

我们认为，想当然的“幻想”不应是科学家的主要任务，智能机器人的研究也绝对没有那些“科幻作家”们的幻想那样来的那么容易。即使具有情感的智能机器人真的可以被“很快”研制出来，“杞人忧天”式的忧虑也是没有必要的。人与机器之间，不会有“你死我活”的矛盾，即使是如真人一般的机器人也是如此；如果我们相信人类的智慧，也就应该相信人和机器人之间是可以“和谐同存”、“肝胆相照”、“荣辱与共”、“互利互惠”、“相互尊重”的，因为机器人的智慧其实就是人类的智慧。

17.6 智能模拟的多智能体系统

智能系统可分为完全智能系统和智能代理系统。即具有[完备]智能的真实主体[或系统]和具有智能的代理主体或系统。真实主体是具有主体意识[自我意识+自主意识]的完全自治体；代理主体则是被上层主体赋予一定主体意识的自动系统[自治体]。它可作为其雇主[服务对象]的“代理人”，按照雇主的意愿、规定和经验做法，为雇主处理一些授权范围内的事物和问题。所谓人工智能研究，本质上就是希望能构建一些自动化的“代理”系统，作为人类（特别是人类专家）的“代理”，可按人类的意愿、规则和经验做法，为人类处理一些授权范围内的事物或问题，从而减轻人类脑力劳动的负担，提高人类工作的效率或服务的水平。其研究主要包括：如何构建这样一些“代理”系统？如何把人类处理事物的知识和能力[本领]“传授”给他们，让他们能像人一样处理事物？或让他们处理事物[问题]时更有水平？

系统的构建问题第一是硬件问题，即适合作为人类“代理”的“自动化系统”的选择和构建问题；第二是软件[程序]问题，即如何让这些“代理”处理事物[问题]的能力能够提高的问题。因为我们的最终目标是希望这些“代理”能像人一样思考问题和处理问题，他就应该有像人一样分析问题和解决问题的能力。怎样才能做到这一点？确实是个难题。

这要搞清，人是如何分析问题和解决问题的？我们如何把这些本领传授给它们？他们应该掌握那些分析问题处理问题的思维能力？如何掌握？这需要先让“机器”能理解人类处理问题的知识和方法，而后要让“机器”能灵活运用这些知识和方法，就像人类专家一样。

高度发展的信息技术给人类带来了社会进步的巨大机遇，也提出了严峻的挑战。机遇体现在人类从来未能像今天这样充分、全面和快速地共享关于文明的信息，也从来未能像今天这样不受时间和空间的限制去交流信息和协同工作。所谓挑战就是，如何使如此庞大而复杂的信息基础发挥和提高其应用潜力。因特网、组件和分布对象等技术的发展为实现分布的协同工作提供了坚实的基础，使得设置于异构网络和异构硬、软件平台上的应用程序具有跨平台互操作能力；然而，基本的互操作能力并不能导致紧凑一致的协同工作，只有互操作的每一方均理解对方提供的服务和数据的语义，并能在面向联合目标的合作活动中协调各自的行为，才可确保协同工作的效用和性能。

智能体 (Agent) 和多智能体 (MultiAgent, MA) 技术的崛起为分布式智能系统研究的实用化和分布计算环境下软件的智能化提供了重要的技术基础。它们是具有社会和领域知识，能依据心理状态 (信念、期望、意向) 自主工作，并具有语义互操作和合作行为协调能力的软件实体，作为参与协调合作的“软构件”或智能主体，不仅可为实施紧凑一致的协同工作提供有力的支持，也为建立面向分布计算的开放性、可重构和可伸缩的新型计算环境建立了基础，更为模拟人类的社会组织智能开辟了新的途径。

17.6.1 智能体技术的研究和发展概述

在所有关于智能系统的系统模型中，智能体理论无疑是最具代表性的一个。智能体技术已是人工智能领域研究的活跃领域和热门话题。其研究背景，一是网络技术的普及应用，人们迫切要求综合集成分布于各种异构平台且语义不相容的系统，以提高其效用 [可重用性]；二是人类社会组织智能模拟的需要，人类的智慧应是众多个体智慧的集成。这使得具有行为自控和群体协作能力的智能体技术吸引了来自许多不同学科领域 (知识工程、机器人、软件工程、信息系统、人机接口等) 的研究兴趣。由此，对智能体的研究也成为集知识处理、网络通信、软件工程、社会行为认知等理论和技术的综合性研究。其中，知识的集成处理与协调运用就构成了智能体技术的基础，不仅需要基于个体领域知识的处理，也需要基于社会协作要求的协同处理和集成。

目前，智能体 (Agent 或 Intelligent Agent) 已经成为许多领域中的一个通用的概念。它既代表着一种新的研究方法的诞生，也推动着关于智能系统的理论研究向更深入的层次发展。

就智能体的定义与功能而言，它首先是一个可运行于动态环境中的具有较高自治能力的实体 (即自治体或主体，该实体可以是一个系统、一台机器，也可以是一个计算机软件程序等)，其根本目标是接受其上层主体 (即其主管主体，该主体可以是人、计算机程序、智能系统或机器等) 的委托并为之提供一定的帮助或服务，能够在目标任务的驱动下主动采取包括学习、通讯、社交等各种手段，感知其外在环境的动态变化并做出恰当地响应。此外，它与其所服务的主体或环境之间还应具有较为松散或相对独立的关系，它具有一定的智能并能通过协作来解决某些用传统方法无法解决的复杂问题。它具有交互/协作性、目标/任务驱动性、自主/可控性等。智能体系统的理论和方法，特别是多智能体的理论和技术，为分布式开放系统的分析、设计和实现提供了一个崭新的途径。尤为重要的是，智能体理论的研究在人工智能的发展史上首次直接而深入地涉及到人类智能活动的社会性。它

使智能具有了丰富而深刻的社会内涵，能表现出人类智能中来源于社会行为的复杂性和多样性，并突破了传统智能理论研究中单纯注重于个体智能而刻意回避由社会互动而产生集体智慧的局限性。同时，它从概念到方法上都有着系列质的创新，使我们切入到了一个全新的认识论视角，转换到了一个新的方法论范式，最终促使我们对智能理论的研究达到了一个更为深刻的洞察与把握。

作为一种新的计算和问题求解范式，智能体理论的主要思想来自于心理学对分布式认知的研究。分布式认知理论认为，人的认知活动产生于人们的交互过程之中，人类智能的本质是社会性的。与此相对应，单个的智能主体就像是智能社会中的一个“人”，而一个多主体的系统就是一个智能“社会”。一个智能体的智能是离不开社会中各主体间的交互活动的，而正是各个主体的交互与协作，才构成了智能化的社会。

目前，智能体技术已经成为智能领域研究的一个中心。一些人甚至认为，“智能主体既是人工智能最初的目标，也是人工智能最终的目标”。在计算机领域，一般认为，智能主体是一类运行于动态环境的、可接受另一个实体的委托并为之提供服务的、具有较高自治能力的“自主计算机系统”，是一类可模拟人类智能行为并提供相应服务的“计算机程序”。它能够通过感知器感知外界环境，能基于其对环境的理解，运用学习和推理的功能对外界环境做出解释，产生推理并做出决定，对内外部环境的要求做出反应，以达到一个或多个系统目标。简而言之，智能主体就是这样一类“计算机系统”，它能根据其服务对象[“用户”或“系统”]的需求向其提供个性化的智能服务。

智能体的描述性定义有**弱形式**和**强形式**之分。在计算机和人工智能研究领域，智能体可以看成是一个“实体”，它可通过传感器感知环境，通过效应器作用于环境。为了达到一定的目标，它有着自成规律地执行任务的工作模式。英国的Wooldridge博士和Jennings教授认为：智能体是一个具有自主性、社会能力、反应性和能动性等性质的基于硬件或（更经常的）基于软件的计算机系统，该定义允许在更宽范围的环境中设计智能体，而且增加了通信的要求。目前，人们所研究的智能体，明显具有下述特性：

自主性[自治性] (autonomy)：自主性是智能体的根本特性。智能体是一种独立的主体，其活动是自主的，在运行时不直接由人或其它东西控制，它对自己的行为和状态有控制权。在没有人或其它软件系统直接干预的情况下，它可根据状态自主地控制其行为，自主地运行和执行操作，对自身动作和内部状态有某种控制能力。对遇到的事件，在不受外界监控指导的情况下，可由自己的决策机制决定采取何种活动，在需要时能够主动与服务对象、系统资源或其它的智能体通讯。它对自身内部状态和行为有一定的控制能力，在完成任务时能根据目标、环境等要求，对自己的短期行为做出规划。它应能在无法事先建模的、动态变化的信息环境中，独立规划复杂的操作步骤，解决实际问题。在服务主体不参与的情况下，可独立发现符合服务对象需求的可资利用的资源并提供与提供服务。

社会性[协作性] (socialability)：智能体具有相互协作的能力，这是智能体系统顺利工作的关键。智能体通常是在一定的环境下持续自主运行的实体，它可通过与其他智能体合作以共同完成某项任务。这种协作可以从简单到复杂，从简一的信息服务到基于智能方式的多个智能体的协作和协商以合作完成任务。当有冲突发生时，智能体还应该具有通过协商解决这种冲突的能力。因此，各智能体之间存在着相互依赖、相互制约的关系，它们的行为既有局部效应，又有全局效应。

通信[交流]能力 (communicability)：当需要合作以共同完成某项任务时，智能体还

应具有通过某种通信语言与其他智能体进行交互或进行信息交换的能力。这种通信既要保证智能体之间的相互交流，又要不至于影响智能体之间的相互独立性，且有助于提高智能体间的内聚力。在多智能体系统中，智能体间的通信是相互协作、协商的基础。

反应[响应]能力 (reactivity)：反应能力是指智能体能对其所在的环境（可能是用户、程序、其他智能体或以上的组合）的改变及时地做出反应。智能体能感知其所处的环境，并通过行为改变环境。能感知所处的环境，并能及时响应环境的变化。

学习[进化]能力：智能体是一个开放的系统，它具有学习或进化能力。随着与服务对象之间的交互作用，作为一个独立的个体的智能体能够主动适应环境，扩充自身的知识。比如，将从服务对象处得来的服务对象的兴趣、爱好、习惯等信息直接转化为内部表示，存放在其知识库中，建立“用户模型”来指导自己的决策，使之更符合客户需求等。

智能性 (Intelligence)：毫无疑问，智能性是智能体的本质特性。智能体所有的行动都具有一个既定的目标，它能够感知环境的变化，并根据智能判断、根据以前的经验改变自己的行为，及时对这些变化做出反应。智能性的实现需要智能体具有相关的知识，并能够进行相关推理或智能计算，当其服务对象的需求没有明确给出时，它能够推测其意图、爱好或兴趣。智能体还可以做很多有高技术含量的工作，例如，理解客户用自然语言表达的需求等。

上面所述是常见智能体的一些最基本的特性。智能体还可具有其他一些特性，比如：**移动性 (Mobility)**（指在网络计算环境下，一个智能体可以被看成是代表用户驻留于网络的代理，它可以在网络上灵活机动地访问各种资源和服务，还可以就完成特定任务同其他智能体进行协商和合作，甚至把自己“迁移”到网络中的其他主机上去执行任务。它可以在任何状态下，包括在运行过程中，从一个节点移到另一个新的节点，从一台计算机移动到另一台计算机，并维持原有的运行状态）、**自发性**（或称自启动性，是指主体的行为是在感知周围环境变化的基础上自发产生的）、**预动性 (Proactive)**（指智能体能不仅仅是简单地对环境做出响应，还能主动表现出目标驱动的行为，能自行选择合适时机采取适宜的动作，以影响环境的变化等）、**诚实性 (veracity)**（不会有意传送虚假信息等）、**慈善性 (benevolence)**（总是尽力提供请求的服务）和**合理性**（总是以合理的方式行事）等。但是，在实际应用过程中，我们并没有必要构建一个包括上述所有特性的智能体或多智能体系统，我们往往需要从应用的实际需要出发，来开发具有以上部分特性的智能体系统。

从软件智能化的角度看，具有这些特性的智能体就是一类自控的、能并发执行的软件。它们封装了某些状态和知识，并能经由消息发送与其它智能体通信，可视为分布对象技术的一种自然延伸。某些人工智能的研究者则赋予智能体更强的性能，要求智能体具有拟人的特性。例如，要求智能体有心理状态，包括信念(Beliefs)、期望(Desires)和意向(Intentions)等，即所谓的BDI状态，甚至希望智能体是有情感的。但这目前还只是一种研究方向。

目前，智能体技术的研究主要集中在三个方面：智能体的行为理论、智能体的体系结构和多智能体间的协作与通信。其成果极大地促进了旨在发挥个体能力的多种类型的智能体的开发和面向松散型协同工作的多智能体系统的研究。

(1) **智能体的行为理论**。行为理论的研究旨在为设计智能体的行为制定形式理论。大多数研究者倾向于把智能体定义为有意识系统。所谓有意识，意指智能体的行为可以通过归因于信念、期望和理性的方法去预言。一般来讲，只要彻底了解一个系统的结构和操作机理，

就能准确预言其行为。然而，对于复杂的系统，彻底了解其所有的细节是困难的。将系统行为解释为有意识，可使我们能以拟人的方式描述、解释和预言复杂系统的行为。

为使智能体在协同工作过程中能相互理解和预言对方的行为，将它们定义为有意识的系统是必要的。我们可以从两个方面来表示智能体的意识（即心理状态）：信息观念和意愿观念。信息观念所反映的是一个智能体对其所处环境的感知，而意愿观念则是以某种方式指导该智能体行动的依据。这两个方面是紧密结合的，许多关于智能体行为理论的研究工作即在致力于这两者之间的关系的研究。

用经典的一阶逻辑来表示涉及意识的观念是不合适的，因为这可能会引起语法和语义问题。另外，信念是不可分割的，但作为谓词公式却可以对其参数作处理。为此，研究人员开始寻求更为合理的形式化表示方法，以解决上述语法和语义问题。存在可解决语法问题的两种基本方法：模态逻辑（Modal Logic）和元语言（Meta-Language）。前者提供非真值函数型的所谓模态操作应用于谓词公式；后者则采用元语言谓词表示意愿，并以另一语言（可以是一阶谓词逻辑）的公式作为其参数项。语义问题也有两个解决方案：可能世界语义

（Possible World Semantics）和解释的符号结构（Interpreted Symbolic Structure）。前者将智能体的信念、知识、意向等刻画为一组所谓的可能世界，并提供一致性理论去处理可能世界间的关系。后者则将智能体的信念默认为表示于特定数据结构中的符号公式。

可能世界语义面临的主要困难是所谓的逻辑万能问题，其意指智能体应是完备的推理器，无时空限制；但在真实世界应用中，总要受时空资源的限制，不可能实现完备的推理器。解释的符号结构实际上是一种“宣告性”模型（sentential model）。基于这种模型，要让一个智能体相信某个命题 Φ ，只须将 Φ 存储到表示观念的符号结构中。在许多场合，这种简单的宣告性模型是能有效地工作的。

目前，尚不存在关于智能体行为的完善理论。上述的各种形式化方式在具有各自优势的同时也存在着各种缺点。然而，这些理论研究工作已经为建立有意向的智能体逻辑模型奠定了基础，有代表性的逻辑模型包括莫尔（Moors）的能力模型、科恩和莱维斯科（Cohen and Levesque）的意向模型、罗和乔治（Rao and Georgeff）的BDI逻辑框架和伍尔德瑞奇的多智能体系统逻辑模型等。这些研究作为智能体行为理论的实用化作出了贡献。

（2）智能体的体系结构。体系结构是构造智能体的方法论，用以指明一个智能体的构成模块和模块间的交互作用，并解释输入信息和当前内部状态如何决定智能体的行动。传统的建立智能体的方式是将其视为特别类型的知识系统。这就意味着智能体的设计要遵循物理符号系统假设，从而可视智能体为一种作解释性处理的自动机，或称其是思考型（deliberative）的。这种智能体包含一个显式表示的关于世界的符号模型，并能基于模式匹配和符号操作进行逻辑推理，以决策应采取的动作。为此，提供有效的感知和表示手段是必要的，以便能及时地将真实世界及其变化转变为智能体推理所用的符号描述。尽管迄今为止，感知和表示仍然是远非已成熟解决的困难问题，但已取得的研究成果，尤其是知识表示和知识处理技术已足以支持智能体应用于许多具体的应用领域。

基于知识的思考型智能体以行为规划部件作为体系结构的核心，而心理状态，尤其是BDI状态（信念、期望和意向）则成为行为规划的依据。一个典型的智能体体系结构是布拉特曼（Bratman）设计的IRMA，它定义了四个关键的符号数据结构：计划、信念、期望和意向。IRMA的行为规划经由五个部件的联合运作来实现：推理机，用于作关于环境世界的推理；手段-目的（Means-Ends）分析器，决定选用哪个计划达到一个智能体的意向；机遇分

析器，监视环境以提供决策机遇；筛选过程，决定哪些潜在的动作流程具有与智能体的当前意向相一致的特性；思考过程，在竞争的计划和动作流程间作选择。

作为对符号主义的人工智能观念的挑战，布鲁克斯（Brooks）提出他的行为主义观念，认为智能行为可以不依赖于符号（从而知识）表示和推理，而是作为行为体与其环境交互作用的结果浮现出来。换言之，智能存在于观察者的眼睛里，而非天生的或能赋予行为体的一种特性。由此，布鲁克斯研制了基于包含（subsumption）体系结构的能快速响应环境变化的机器人。所谓包含体系结构是执行各种任务的行为的一个层次体系。这些行为竞争对机器人的控制，通常较低层次的行为面向更基本的动作，从而有更高的优先级。

遵循上述行为主义观念，导致了基于所谓“处境-动作”型规则的反应型体系结构用于建立智能体。这里的动机来自人和生物的大多数每日活动是例行公事的观察—依据处境执行习惯（常规）的动作，并且处境与动作间的对应关系极少改变。于是，这些“处境-动作”型规则可以编码成一个低级结构，以实现智能体对处境变化的快速反应。罗圣斯奇和凯尔布林（Rosenschin and Kaelbling）进一步发展了反应型体系结构，将智能体设计为所谓的处境自动机（Situated Automata）。一个说明语言用于编写“处境-动作”规则（准确地说是“感知-动作”规则），然后编译为满足陈述性说明（即这些规则）的数字电路。该数字电路能在受限时间内快速执行适当动作，但不做任何符号操作，事实上没有任何符号表达式表示于执行系统。

尽管反应型体系结构能实现智能体对处境变化的快速响应，并能避免符号操作和知识表示的复杂性和难处理性，但解决复杂问题的能力较差。于是，许多研究人员认为思考型和反应型体系结构都无法单独实现高性能智能体，提出了将两者结合起来的混合方式。

（3）**多智能体间的协作与通信**。通信机制构成支持智能体间互操作和协同工作的重要基础。建立通信标准则是设计通信机制的关键，它可促进通信接口和智能体的实现相分离。只要遵守通信标准，智能体之间就可进行互操作。

建立通信标准可以经由设计公用的通信语言来实现，主要有两种方式：过程型和说明型。前者提供脚本语言去表示需交换的过程指令—单个命令或脚本程序，后者则通过谓词逻辑语言传递信息。

过程型语言简明而有效，且能直接执行（使用语言提供的解释执行器），但存在不少缺点。首先，设计要发送的信息时可能需要某些关于接收者的信息，但这些信息不一定能获得。其次，程序是单向性的，不能适用于智能体需双向执行语句的要求。更严重的是，程序语句难以组合：当一个智能体从多个其它智能体接收需组合应用的语句时，过程型方式无法处理。

与过程型方式相对照，说明型方式将通信视为陈述语句（定义、假设、约束、规则等）的交换，具有充分的表达力（可传送广泛种类的信息，甚至包括程序）、简洁的表示形式，并能跨越不同的硬、软操作平台。这种方式的研究工作主要是由ARPA（美国国防部先进科研项目管理署）倡导的知识共享创始（Knowledge Sharing Initiative）项目来完成的。该项目的研究者定义了称为 ACL（Agent Communication Language）的说明型通信语言，并通过设计共享的本来来促进基于ACL的语义互操作。**ACL由三个部分构成：词汇表、称为KIF的内部语言和称为KQML的外部语言**。一个 ACL消息是 KQML（Knowledge Query and Manipulation Language）的表达式，其参数由 KIF（Knowledge Interchange Format）表达式（项、句子、句子组）填充，而项和句子则由定义于词汇表中的单词构成。ACL词汇表

可视为大的、可开放性扩展的字典，其中包含的每个单词都有由参与协同工作的智能体一致赞同的语义，从而构成这些智能体的共享本体——对于协同工作所涉及论域的概念化（论域中的实体、属性和关系）的清晰说明。ACL已应用于几个大的实验型多智能体系统，如PACT和SHADE，取得了可喜的成果。鉴于ACL可以作为异质（硬软件异质和本体异质）系统通信的较理想语言，其标准化工作已在进展中，一些计算机厂商也承诺要提供处理ACL的商品化产品。尽管从长远的观点，ACL的应用前景看好，但由于其不成熟性，近期内，过程型通信语言仍是作为应用的主流。

17.6.2 几类典型的智能体

随着智能体研究的深入发展，人们已提出了多种类型的智能体模型。为了对智能体技术进行深入的研究和理解，我们有必要对现已存在的智能体模型作分类考察。按照诺纳(Nwana)的观点，可从以下几个方面对各类智能体作分类分析：

- **移动性**——以能否在分布计算环境中移动，可将智能体分类为静止型和移动型。

- **思考性**——以能否基于知识表示和符号推理技术去规划和协调自身在协同工作中的行为，可将智能体分类为思考型和反应型。后者不作符号推理，其行为是对环境状态的反应。

- **行为特性**——依据智能体拥有的行为特性分类。理想的智能体的基本特性可归纳为三个：自治性、合作性和学习性。这里自治性包含预动性，而合作性则建立在社交能力和响应性基础上。真正的智能体应具有学习能力，以便在与外部环境交互作用的过程中逐步提高适应性，但目前实用的智能体通常只具备自治性、合作性。

- **扮演的角色**——按智能体扮演的角色，可以分为：提供信息、执行任务、作为接口等。

由于智能体的功能特性并不是完全独立的，因此，按这几个方面作分类分析，并不完备，也不精确。不过，我们的目的并不在于分类研究本身，只是想区分出值得深入研究一些种类。此外，这些类型中可以有二个或三个类型的特性集结于同一智能体，称为混合型智能体；当一个多智能体系统包括多种类型的智能体时，常称为异质多智能体系统。

一般认为，智能体是一类在不可预测环境中自主工作的主体。所谓“不可预测环境”，是指其设计者无法事先预测其所有可能遇到的环境状况。由于智能体直接与现实世界（包括其他智能体）发生交互作用，因此，设计者也就不可能事先预料所有可能出现的环境状况并相应地规定其恰当的处理方法，只能依靠智能体在其实际运行中的自主工作能力去“灵活地”应对各种可能发生的情况，包括既属于其功能范围而又超出其知识范围的“非预期情况”等。所以，智能体必须具有一定的自适应能力。而作为一类可在特定环境下感知环境，并能自治地运行以实现其预定目标的模型或程序，智能体可具有多种类型。目前，人们所研究的智能体主要有两类：反应型智能体(Reactive Agent)和慎思型智能体(Deliberative Agent)。而从结构上分析，智能体则包括了慎思型智能体、反应型智能体和混合型智能体等不同类型。

1. 慎思型智能体 (Deliberate Agents)

目前，理论界研究最多的智能体是“慎思型智能体”。慎思型智能体的最大特点就是智能体看作是一种意识系统(intentional system)。由于人们提出这类系统的主要目的之一就是把它作为人类个体或社会行为的智能代理，那么，这类智能体就应该或必须能模拟或表现出被代理者具有的所谓意识态度(intentional stance)，如信念、愿望、意图(包括联合意图)、目标、承诺、责任等。我们认为，把智能体作为意识系统来研究是合理的，意识态度和意识系统概念的引入将有助于我们以一种自然而又直观的方法来理解、描述、规范或预测智能系统的内部结构、运行规律和变化状态等。

构建慎思型智能体的经典方法是将其看作是一类特殊的基于知识的问题求解系统,其中,选择什么样的意识态度来刻画智能系统是构建慎思型智能体首先需要考虑的问题。不同的智能模型或系统对意识态度可有不同的认识和观点。比如,Shoham等人根据应用的特点就将意识态度分为了信息类(重点刻画智能体所具有的信息)、动机类(重点刻画智能体的动作选择)、社会类(重点刻画与其他智能体的社会、道德关系和理性行为等)和其他类(重点刻画诸如欢喜、恐惧等)。而Wooldridge等人则把单个智能体应具有的意识态度(或精神状态)分为两大类:信息态度和意愿态度。前者是指智能体所拥有的关于自己、环境及其他智能体的信息和知识,如信念和知识等;后者是指那些能导致智能体执行动作的状态,如愿望、目标、意图、承诺、责任、能力等。认为,一个理性的智能体总是基于信息态度去采取意愿态度,如根据信念去选择目标和形成意图等。

慎思型智能体强调智能体必须具有主体应有的**意识状态(intentional stance)**,如**信念、期望、意图、承诺等**。基于这种思想,Rao和Georgeff等提出了BDI(belief-desire-intention,即信念-期望-意图)模型,用信念、愿望和意图这三类意识态度来刻画智能体的结构,并最终通过规划库来研究BDI模型的抽象性质和推理过程。具体来说,一个BDI模型包含三种基本成分:① Belief,即主体拥有的知识等。由于这种知识是不完善的、可错的,按西方文化传统称为belief(信念);② Desire,接近于我们通常所说的“目标”,但更宽泛。比如可以是非强制性的,仅提供某种选择范围等;③ Intention,即行动计划或规划,它可由主体根据其Belief和Desire自动生成。目前Intention的生成机制仍基于演绎原理。

在上述慎思型智能体的体系结构中,信息数据库是一些现实世界的事实和知识,是完成智能功能的基础;规划数据库是关于执行的动作或反应规划的组合;解释器是对规划数据库中选择合适的规划放入意图结构,并对整个系统进行管理;通过用户接口和传感知外界事实,调整信息和知识库等。

从外部看,慎思型智能体是一个具有信念、意向、期望等意识属性的统一体。各智能体之间可相互通信或相互合作,而且,智能体自身还可以有诸如计算、推理或其他人们所希望具有的某一特定的智能特征或行为等。

我们可以把慎思型智能体看作是将通常应用于人类自身行为描述的一些概念如信念、意图、承诺等赋予智能体而形成的一类意识系统,也可以认为它们是我们为机器所建立的一些比较简单(与人相比)的信念、知识和愿望理论。这也就是说,我们对人类智能的研究和对智能体的研究可以相辅相成,互相促进,而在现阶段大都是将前者的一些研究成果应用于后者。由于人脑是巨系统,其内部结构异常复杂,像慎思型智能体这样的意识系统,同样也应该是一个庞大、复杂的系统。并且,这样的系统在实际研究时还可以根据功能需要把其内部结构分成很多的系统环节。与人的行为相类似,该智能体也应有各种各样的行为,如自主自利行为,特别是人在正常精神、意识状态下可以表现出的简单或复杂的行为等,并能表现出一定的复杂性。而从动力学角度来看,这些行为应是按时间展开的具有一定完整意义的状态变化过程。除了自主性与交互性的特点外,智能体的行为还应具有预期性,这样,它才能根据其意识状态、意识属性对其行为加以预测。

在现有慎思型智能体的研究(包括BDI建模以外的方法)中,基本上是沿袭人工智能中基于演绎的问题求解思想,即:依主体的Belief,从能够实现主体当前Desire的行为中选择适当的行为予以执行。这就有可能出现一个问题:当主体遇到“非预期情况”时,就不

能产生恰当的行为，从而导致问题求解失败。事实上，这一难题在以往的知识工程实践中早就出现了，即所谓知识系统的“脆弱性问题”；并且被认为是导致多数知识系统不实用的首要原因。“非预期情况”问题已受到国际学术界的普遍关注。事实上，这一问题能否解决，将在很大程度上决定多主体系统的研究能否满足实际应用的基本要求。对此问题，目前已提出两种解决思路：**基于“机会发现”的慎思式途径和基于学习、进化或行动选择理论的适应性途径**。前一途径目前处于概念化研究阶段，几乎还没有技术性工作。后一途径尽管相关的研究很多，但绝大多数不是针对慎思式主体的，或者至少在技术层面上与慎思式主体的研究是相互脱离的，这使得双方的成果目前还难以相互借鉴。另外，“适应”未必只能通过学习或进化的方式，例如，“随机应变”也是一种与学习和进化相“正交”的慎思式适应。

2. 反应型智能体(reactive Agent)

从上面的论述中我们可以看出，传统人工智能的特点和种种限制给慎思型智能体带来了许多尚未解决、甚至根本无法解决的问题，这就导致了反应型智能体的出现。反应型智能体强调交互的作用。反应型智能体的支持者们认为，智能体的智能主要取决于感知和行动，因而，智能体的模型应是一类“感知-动作”模型。他们认为，某一个领域的先验数据和知识对智能体并不是必须的，智能体可以像人类一样通过和外界的交互以及对环境的感知而逐步自我进化，而智能行为也只有在现实世界与周围环境的交互作用中才能表现出来。反应型智能体的支持者们还认为，符号人工智能对真实世界中客观事物及其行为工作模式的描述是过于简化的抽象，因而不可能是真实世界的客观反应。

反应型智能体的最有力的支持者当数R. Brooks，他提出了一种被称之为子前提结构(subsumption architecture)的构建智能体的控制机制，该结构是由用于完成任务的行为(behaviours)来构成的分层结构，这些行为通过相互竞争以获得对智能体的控制权。这种简单的结构在实践中被证明是非常高效的，它甚至解决了传统符号人工智能很难解决的问题。

反应型智能体不拥有关于环境的内部符号模型，代之以刺激-响应的方式对所处环境的状态作出反应。尽管反应型智能体的理论、体系结构和程序设计语言已得到开发，但最重要的还是其结构以及与其它同类智能体交互作用的简单性。复杂的行为模式可从它们的交互作用中浮现。马艾斯(Maes)将这类智能体的基本特点归纳为以下三点：① 浮现作用。交互作用的动态性导致智能行为浮现的复杂性，所以在建立反应型智能体时不必对它们的行为作预先说明。② 任务分解。一反应型智能体自身可视为自治操作并执行特别任务(如感知、控制、计算等)的模块的集合，模块间的通信最小化并相当低级(通过数字电路实现)。智能体中没有一个是拥有全局的模型，所以全局(总体)行为只能是某种浮现(观察者的感受)。③ 反应型智能体的操作基于接近于原始传感数据的表示，而非高级的符号表示。

更一般地，一个反应型智能体可以定义为一个三元组(S, C, A)，其中，S 是一个反应式规则集，C 是一个或一组选择机制，A 是一个或一组适应机制。相应地，现有反应式主体可以进一步划分为三种类型：① 简单反应式结构。其选择机制为“条件激活”，其适应能力来自规则本身。因此，严格地说，简单反应式结构不具备充分的适应能力和自主性。② 包容结构。其选择机制基于“局部条件激活”的“整合”，其适应能力来自规则的整合效应和系统与其环境间的交互作用。当环境特性发生变化时，往往需要进行重新调试才能保证

其适应性。因此，包容结构的适应能力并不如原来认为的那么强。③ 学习分类器系统。其选择机制基于“拍卖”策略，其适应能力立足于“盈亏”概念和遗传算法。相对而言，其适应能力在现有模型中是最强的。

3. 混合型智能体 (hybrid Agent)

从以上的分析中可以看出，反应型智能体反应较快，交互性能较好，能及时而快速地响应外来信息和环境的变化，但其智能程度较低，也缺乏足够的灵活性。慎思型智能体具有较高的智能，灵活性好，类似于传统的专家系统，但无法对环境的变化作出快速响应，而且执行效率相对较低。混合型智能体则旨在综合应用它们的技术，综合二者的优点，希望兼具较强的灵活性和快速响应性。

由于混合型智能体既要有特定的知识系统，以提高其智能性，又要有较好的交互和自我进化能力，因此，混合型智能体通常被设计成至少包括如下两部分的一个层次结构：高层是一个包含着符号世界模型的认知层，它用传统人工智能的方式处理规划和进行决策；低层是一个能快速响应和处理环境中突发事件的反应层，它不使用任何符号表示和推理系统。反应层通常被给予更高的优先级。采用分层结构时要处理的主要问题是，各层应采用什么样的控制框架以及各层之间应如何交互。混合型智能体的一个典型实例是过程推理系统 (procedural reasoning system, 简称PRS)，它是一个可在动态环境中进行推理和执行任务的BDI系统。

从当前的研究和应用现状来看，思考型智能体模型仍占据着主导地位，因为多数研究和开发者都喜欢使用自己已经较为熟悉的人工智能技术和方法；反应型智能体模型的研究和应用目前尚处于初级阶段；混合型智能体模型由于集中了上述两种智能体的优点而成为人们关注的一个研究热点。在所有有关智能体理论和结构的研究中，BDI模型以其坚实的理论基础和方便的可操作性而成为目前研究和应用领域中使用最多的结构。

17.6.3 (已有的) 几种典型的智能体描述模型

智能体是一个高度开放的智能系统，其结构如何将直接影响到系统的性能和智能。智能体研究的任务之一就是设计智能体的程序，即实现智能体从感知到动作的映射函数。这种智能体程序可在某种计算设备上运行。一般认为，智能体的体系结构和程序之间存在如下关系：智能体=体系结构+程序。具体的说：在计算机系统中，智能体相当于一个独立的功能模块，独立的计算机应用系统，它含有独立的各种功能操作程序、数据结构和相应的输入输出功能。智能体程序的核心部分叫做决策生成器或问题求解器，起到主控作用，它接收全局状态、任务和时序等信息，指挥相应的功能操作程序模块工作，并把内部的工作状态和所执行的重要结果送至全局数据库。智能体的全局数据库设有存放智能体状态、参数和重要结果的数据库，供总体协调使用。智能体的运行是一个或多个进程，并接受总体调度。特别当系统的工作状态因工作环境而经常变化以及智能体的具体任务时常变更时，更需要总体协调。

智能体比对象更能反映现实。它不但抽象出实体的特性、动作，还有感觉、心智、承诺等。一方面，由于智能体的自制性、反应性，这就意味着它有自己的控制流程；另一方面，智能体的系统特性和系统协作的智能性，也意味着智能体具有交互、协作的功能。为了对智能体进行深入研究，人们曾从多个方面来对智能体进行描述，提出多个智能体模型，下面，我们仅举出几个典型示例。

(1) 智能体的BDI模型

在所有有关智能体理论和结构的研究中，BDI模型以其坚实的理论基础和方便的可操作

性而成为目前的研究和应用中使用最多的模型。BDI模型研究的典型代表当数Rao和Georgeff, 他们提出了一系列描述智能体意识态度的BDI逻辑, 这种BDI逻辑用3个基本模态算子刻画: 信念、愿望和意图。

(2) 智能体的面向对象描述模型

从形式化的角度并结合面向对象的技术, 一个智能体 (Agent) 可用一个5元组表示:

$$\text{Agent} ::= \langle \text{ID}, \text{Mental}, \text{Rule}, \text{Action}, \text{Interface} \rangle$$

其中, ID 为Agent的唯一标识符。Mental 为Agent的心理状态类, 包含其能力、信念、承诺等精神状态; 能力体现了它对自身能力的意识, 信念表示得到的事实, 而承诺则体现了与其它Agent间的关系; Agent接收到的消息可以作为事实以信念的形式进行处理。Rule 是Agent的行动规则类, 对于知识Agent而言, 它体现了对该领域知识进行推理的控制; 而对于功能Agent而言, 它体现了与各知识Agent间协作的知识, 以及对用户请求的分解、对返回结果的综合。Action 是Agent的执行动作类, 体现了Agent的实际能力; 在知识Agent中, 它主要是负责使用推理机进行知识推理; 在功能Agent中, 它是负责对知识的综合处理。Interface类包括了Agent与用户或其它Agent之间进行交互的接口。

在上述模型中, 可以将一个智能体 (Agent) 的行为能力简单地概括为感知能力、对自身的控制能力和活动能力, 各个智能体在问题求解过程中所担任角色可以归纳为解释、分类、转换和控制等等, 所采用的问题求解方法可以包括传统的软件技术和基于知识的处理技术, 甚至是由特定的人来亲自来处理。

(3) 智能体的结构形式化模型

目前, 对智能体在各种不同环境下的行为研究也日益普遍。Dennett认为人们通常从三个方面看待系统: ① 物理立场, 基于系统的物理特性和规律; ② 设计立场, 基于系统的设计目标; ③ 意识立场, 把系统看作理性主体, 通过信念、愿望和其他意识属性来预测其行为。而现有的研究一般建立在智能体是一个意识系统的基础之上的, 同时, 也是把智能体看作一个统一整体。因此, 在已有的智能体形式化描述中, 一般把智能体 (Agent) 作为一个整体并定义为:

$\text{Agent} ::= \langle \text{Ms}, \text{IB} \rangle$, 其中, Ms 表示Agent的内在精神状态, IB表示Agent的外部交互行为;

$\text{Ms} ::= \langle \text{Cg}, \text{It}, \text{If}, \text{Cd}, \text{Pl}, \text{Ds} \rangle$, 表明Agent是具有认知、意向、推理、协调、规划和决策等精神状态的一个整体;

$$\text{IB} ::= \langle \text{Sp}, \text{Sg}, \text{S1}, \text{As}, \text{Se}, \text{Ex}, \text{Ef} \rangle,$$

其中, $\text{Sg}:\text{Sp} \rightarrow \text{Sg}$ 表示相空间到状态空间的映射, Sp表示相空间, Sg表示Agent外部状态集合;

$\text{Se}:\text{Sg} \rightarrow \text{S1}$ 是感知函数, Agent由其通过人机交互和相互通信识别外部状态;

$\text{Ex}:\text{S1} \rightarrow \text{As}$ 是执行函数, 表示Agent在当前识别状态下欲采取的行动;

$\text{Ef}:\text{As} \times \text{Sg} \rightarrow \text{Sg}$ 是效用函数, 表示当前活动的执行所产生的结果状态。

上述的模型为形式化描述智能体的行为提供了基本的方法, 因而是十分重要的, 但由于模型中没有细化智能体的内部结构, 所以并不能很好地把握智能体行为与其内部状态的变化过程联系起来, 特别是模型中的Ms和IB没有体现出关联性, 因而也就无法体现智能体的外部交互行为应与其内在意识、精神状态密切相关这一基本思想。为了把智能体的行为和它所应具有的内部结构联系起来, 文献[]提出一种智能体结构的形式化描述模型。其定义如下:

$$(a) \text{ Agent} ::= \langle \text{St}, \text{Fs}, \text{VL}(\text{Fs}), \text{Ir}, \text{Is} \rangle$$

其中, St 表示智能体内部物理构成, 它是智能体存在的物理基础, 象人一样, 它也可分成意识系统和与行为、活动有关的系统环节; F_s 是 St 依据各个系统环节功能的抽象, 表示智能体的逻辑结构, 显然, 在现阶段认清 F_s 是重点, 只有充分认清 F_s 的结构, 才有办法设计出 St ; $VL(F_s)$ 是取值函数, 它可以用来对 F_s 内部各个不同系统环节进行取值, 从而了解该环节的状态、行为等; Ir 是智能体对自我行为与活动的表述, 即自知的活动记录, 它是按时间展开的序列, 该序列中既有有意识的行为、活动表述, 又有一些自我的无意识状态与意识的转换过程等相对应; Is 是站在其他智能体的角度观察到的该智能体的行为与活动。

(b) $F_s ::= \langle EGO, Obj, Psh, FL, SSL, PCL, RSL, ARL,$

$SCL, MDS, ITL, ACTL, AS, AP, AB, RFL, \dots \rangle$

其中, EGO 是一个假想的精神中心, 是智能体自我的所在, 它是智能体行为与活动的主要发起者; Obj 表示对象, 是外部各种事物在智能体意识系统中的反映; Psh 表示心理环节, 智能体感知信息需经过心理环节才能投射到 EGO , Psh 是低层次自我的所在, 因而可部分地决定智能体的行为; FL 表示感情系统, Obj 要通过感情系统与 EGO 发生联系; SSL 是感觉环节, 用来接收、转换和输出各种感觉信息, 其中包括各种感觉器官; PCL 表示知觉环节, 在它的帮助下, 智能体可以知道某种感觉对自我的意义如何, 自我才能决定下一步该做什么; RSL 表示推理环节, 智能体在获得知识时常需借助此环节来分析所获得知识与以前知识是否矛盾、有什么关系等等; ARL 表示协商环节, 虽说智能体的自我中心是 EGO , 是统一的, 但智能体的自我会有多种不同的愿望, ARL 环节则起到在智能体的不同愿望之间相互协调以做出更合理的决策; SCL 表示意识状态转换环节, 智能体所处的思维状态是经常可以发生变化的, 在下一个时刻到来前, 智能体都面临这样的选择, 是继续当前思维状态不改变还是要转换到别的思维状态? 这种改变与不改变便是通过 SCL 环节来实现; MDS 表示情绪系统, 因为 EGO 处在情绪的环境中, 它可以有情绪, 也可以感受情绪, 并时常因情绪的变化而导致行为的改变; ITL 表示愿望环节, EGO 可以有各种强弱不同的愿望, 它们是智能体行为的主要动力; $ACTL$ 表示动作环节, 该环节主要表现为控制系统的特征, 在有了一定的动作指令时, 通过 $ACTL$ 便可产生相应的动作; AS 表示感觉信息投射区, 是各类感觉信息集中的地方, 这些信息又会经不同的渠道被送往不同的地方; AP 表示知觉信息投射区, 感觉的信息要让 EGO 注意到必须经过知觉环节把有关信息输出到 AP ; AB 表示行为信息投射区, 是集中各类行为指令的地方, 与动作有关的信息会经过 AB 交 $ACTL$ 执行, 而与思维活动有关的信息会经过 AB 投射到智能体的内部相应的行为环节上; RFL 表示映射环节, 它控制不同信息区之间信息的映射; 其他还有记忆环节、注意力环节等等。

上述模型中的 F_s 根据需要可以进一步分解, 分解越细, 得到的逻辑结构将越接近于其物理结构。由于各个环节内部复杂程度可能会有所不同, 因而也允许分解层次可以不同。

(c) 如果 $F_s ::= \langle F_{s1}, F_{s2}, \dots, F_{sn} \rangle$, 其内部各个环节 F_{si} 可以继续分解, 则:

① 允许不同环节的分解层次是不同的。

② 在 F_s 的模型中, 若 F_{si} 可以分解成 $F_{si1}, F_{si2}, F_{si3}, \dots, F_{sim}$, 则可以用分解形式取代 F_{si} 表示成:

$F_s ::= \langle F_{s1}, F_{s2}, \dots, F_{s(i-1)}, \langle F_{si1}, F_{si2}, \dots, F_{sim} \rangle, F_{s(i+1)}, \dots, F_{sn} \rangle$ 。

③ 在模型分解过程中, ②的方法可以反复使用。

由于模型分解太细, 将不利于从功能角度去认识智能体的内部结构, 所以, 一般在达到具体的功能环节, 如感觉系统达到听觉、视觉等层次即可。此时, 为了方便, 常把该功能

环节作为一个整体来考虑。

取值函数也是上述模型运用的关键之一，它灵活方便却又不易掌握。不同的系统环节，其取值的方法可能不同。考虑到观察的需要，在Fs细分的同时，取值函数要能随之深入模型内部进行取值，必要时甚至可以通过它得到某个具体状态变量的值。

由于智能体是一个复杂的系统，其内部可分成许多不同的功能环节，这些环节的有机协调与合作是智能体的一个关键。上述模型把智能体的行为与其内部属性联系起来，这对了解智能体行为与其内部结构的关系，了解智能体意识的实现过程与机制，都是有意义的。

(4) 智能体的慎思式适应模型 文献[?]认为，现有智能体研究存在的问题主要是：① 反应型智能体缺乏慎思能力。虽然某些类型的反应型智能体具有较强的适应能力，但其慎思能力却较弱，尤其是在BDI 层次上的处理能力和适应能力很弱。② 慎思型智能体缺乏适应能力。由于演绎法的限制，现有BDI模型只能借助于“例外规则”处理“例外情况”，因而它在本质上并不具有自主适应能力。因此，有必要在两种途径之间取长补短，建立一种既有慎思能力、又有适应能力的机制。Holland 等人试图在反应式结构的框架上发展慎思能力，文献[]则尝试在慎思式主体的框架上发展适应能力。认为，尽管目前慎思式主体缺乏适应能力，但慎思式结构和BDI 模型本身并不必然具有这种局限性，并不必然局限于演绎法。一个强有力的证据是，人类的某些适应能力，比如“随机应变”，往往是在慎思的基础上完成的。文献[]所给出的慎思式适应模型的基本框架是（其中=>表示“归结为”）：

自主性=>不可预测环境中的适应性=>information 维适应+ motivation 维适应
=>信念修改+目标修改

目标修改=>愿望修改+意图修改

愿望修改=>愿望保持性+ 效用权衡

意图修改=>承诺可传递性+可行性+效用权衡。

其中，基于“目标修改”的“慎思式适应”原理上包括两大部分：

(A) 非预期情况的判别，即识别在给定情况下能否通过传统方法生成可实现当前目标的行动。

(B) 目标修改，即在出现非预期情况时自动生成恰当的“替代目标”，使智能体在非预期情况下仍可采取合理的行动。又包括两个方面。

(B1) 替代目标生成，非演绎地建立原目标的“替代目标”；

(B2) 替代目标选择，经过适当的判断选择恰当的“替代目标”作为决策结果。

上述慎思式适应的主体部分是“目标修改”。显然，“目标修改”不应是盲目试错，“替代目标”也不应与原目标无关。为了保证认知合理性和计算可行性，要求“替代目标”与原“目标”之间至少具有“意向保持关系”；当所涉目标为Intention 时，还要求具有“承诺可传递性”关系。这些关系的形式化是目标修改和慎思式适应的关键所在，但无法简单地用经典逻辑直接刻画。为此，文献[]发展了一种新的形式理论Lm4c，为愿望保持性和承诺可传递性提供一种恰当的形式规范。假如上述想法是合理的，方案是可行的，通过这一研究有望获得一种将“慎思”与“适应”融为一体的问题求解机制。它是为沟通这两大研究提供的一种有益的尝试。

17.6.4 多智能体系统的理论基础

目前，对智能体理论和技术的研究大致可分为智能体理论、多智能体系统理论（multi-agent system, 简称MAS）和面向Agent的程序设计（agent oriented programming,

简称AOP)这3个相互关联的方面。智能体理论是多智能体系统理论研究的基础,我们可以将有关智能体理论的研究统一在多智能体系统理论的研究之下,这样,智能体理论就可以被看成是多智能体系统理论研究中的微观层次,而有关智能体间关系的研究则构成了多智能体系统理论研究的宏观层次。

多智能体系统是由多个智能体组成的系统。与单个的智能体相比,多智能体系统具有如下特点:每个智能体成员仅拥有不完全的信息和问题求解能力(因而其观点是有限的);不存在全局控制,数据是分散或分布的,计算过程是异步、并发或并行的。但多智能体系统的协作求解问题的能力要大大超过单个的智能体,这是导致多智能体系统产生的最直接的原因。多智能体系统理论则主要是研究如何将多个智能体构成一个完整的智能系统,其研究的重点包括多智能体系统的构成,多智能体系统中各智能体间的协商和协作,包括联盟的形成以及行为的规划等。

1. 多智能体系统的联合意图

在单一智能体的模型中,关于智能体的意识态度的表示和形式化处理问题,是我们研究的一个重要方面。对于多智能体系统,我们除了考虑上述因素外,还要考虑多个智能体意识态度之间的交互问题,这是多智能体系统理论研究的一个重要方面。

能够对环境中其他智能体的意识态度进行推理是智能体间共存、竞争或协作的要求,智能体之间的协同、协商和协作行为是在其各种精神状态的支配和控制下才产生、进行和完成的。与共享精神状态相关的理论主要涉及到相互信念、联合目标和联合意图等概念,其中以联合意图为代表。在多智能体环境下,相互信念即是所谓的公共知识,它与多智能体间的通信密切相关。

从目前的研究现状来看,对联合意图的研究主要有两种观点:一是宏观观点,即在每个智能体个体之外存在一种联合意图控制整个组织或社会的协作行为;二是微观观点,即每个成员的意识态度导致个体动作一起完成协作目标。Haddadi提出的关于联合意图的社会/组织观点和个体观点即分属于上述两种观点。Jennings提出的关于联合意图的承诺和公约(convention)也分别对应于上述两种观点。Bratman和Rao等人对联合意图的要求则只限于微观观点,他们基于单个智能体的BDI模型递归地定义了社会智能体的概念,并引入了社会规划,认为每个团队成员都对联合行为作出承诺,联合意图是实现共享联合目标的方法,成员间要相互承诺,并及时把各自动作的成功或失败通知给其他成员。

2. 多智能体协商

协商(negotiation)是多智能体系统实现协同、协作、冲突消解和矛盾处理的关键环节。有关多智能体协商的关键技术可以概括为协商协议、协商策略和协商处理这3个方面的内容。

协商协议的主要研究内容是智能体通信语言(ACL)的定义、表示、处理和语义解释。协商协议的最简单形式是如下一条协商通信消息:(〈协商原语〉,〈消息内容〉)。其中协商原语即消息类型,它的定义通常基于言语行为理论。消息内容除包含消息的发送者、接收者、消息号、发送时间等固定信息外,还要包括与协商应用的具体领域有关的信息描述。协商协议的形式化表示方法通常有3种:巴科斯范式表示、有限自动机表示和语义表示。巴科斯范式表示具有简洁、明了的特点,是最常用的表示方法。智能体通信语言中最著名的是KQML语言。

协商策略是智能体决策和选择协商协议和通信消息的策略。协商策略包括一组与协商协

议相对应的元级协商策略和策略的选择机制或函数两部分内容。协商策略基本上可以分为5类：单方让步、竞争型策略、协作型策略、破坏协商和拖延协商。后两类策略显然不利于推进协商进程，而单方让步策略只在协商陷入僵局或协商不再有意义时才起作用，所以只有竞争型和协作型策略才是有意义的。竞争型策略一般是指协商参与者坚持自己的立场，在协商过程中表现出竞争行为，使协商结果向有利于自身利益方向发展的协商对策，合同网协商模型、劳资协商、基于对策论的协商过程等都属于此类。协作型策略则是指协商各方都从系统整体利益出发，在协商过程中互相合作，它们采取的协商对策有利于寻找互相能接受的协商结果，采用协作型策略的协商过程包括部分全局规划（PGP）、FA/C等。智能体应动态、智能地选择适宜的协商策略，从而在系统运行的不同时刻表现出不同的竞争或协作行为。策略选择的通用方法是依据影响协商的多方面因素，给出适宜的策略选择函数。策略选择函数包括效用函数、比较或匹配函数、兴趣或爱好函数等。策略选择函数的设计除了要考虑影响协商的各种因素之外，还要考虑冲突综合消解以及与应用领域有关的属性等。

协商处理包括协商算法和系统分析两部分内容。协商算法用于描述智能体在协商过程中的行为，包括通信、决策、规划和知识库操作等。系统分析的任务是分析和评价智能体协商的行为和性能，回答协商过程中的问题求解质量、算法效率以及系统的公平性和死锁等问题。协商协议主要处理协商过程中智能体间的交互，协商策略主要涉及智能体内的决策和控制过程，而协商处理则侧重于对单个智能体和多智能体协商社会的整体协商行为的描述和分析。前两者描述了多智能体协商的微观方面，而后者则刻画了多智能体协商的宏观层。

有关多智能体协商的典型工作有：Sycara以劳资协商为背景对非协作类的多智能体相互作用的研究，给出了基于实例推理和多属性效用优化理论的“劝说性辩论”模型；Grosz等人在会谈理解研究中，给出了支持人机交互通信的形式化模型；Wellman等人把面向市场方法用于设计智能体间的协调过程，提出所谓基于一般平衡理论的“面向市场程序设计”机制等。

3. 多智能体规划

规划是连接心理意识状态与动作执行之间的桥梁，有关动作和规划的研究一直是智能体理论研究的活跃领域。多智能体系统中的规划与经典规划不同，属于适应性规划（reactive planning），需要反映出环境的持续变化等。

目前，对多智能体系统中规划的研究主要是从两个不同的角度进行：一是将规划看作是一种可以在世界状态间转换的抽象结构，典型地，如与或图；二是将规划看作是一类复杂的智能体心理意识状态。这两种方法都在一定程度上降低了经典规划中解空间搜索的代价，从而有效地指导了资源受限智能体的决策过程。在将智能体的规划库定义为一个与或图结构的方法中，每一条规划都包括如下4个部分：① 规划目标，这是规划的点火条件，表示该条规划能达到的目标；② 规划前提，表示该规划被执行前必须满足的环境或状态条件；③ 规划体，是规划的程序部分，由规划序列和规划子目标组成；④ 规划结果，表示执行规划后对环境或状态的更新结果。

如何在自私的多智能体动态环境中实现智能体间的灵活通信和动作执行，也是当前多智能体系统规划研究的热点。目前，多数多智能体系统规划方法存在着以下的不足之处：规划不太适合动态变化的环境；很多形式化较好的理论模型与实际问题距离太远。如何更好地解决多智能体规划中的资源冲突，如何在多智能体规划中引入质量因素等等，这些都是

智能体规划需要解决的问题。

目前，对于多智能体系统的研究主要有两个方面：一是关于智能体基础理论的研究；二是关于特定智能体的体系结构、程序设计语言及应用环境等的研究。由于多智能体系统理论的研究时间还不长，其理论和技术各方面还不完善。Wooldridge等人曾从政治、管理、概念、分析和设计、微观角度（单个智能体级）、宏观角度（社会级）、实现这7个方面论述了在使用智能体技术开发应用程序时所存在的种种误区。Jennings和Nwana也多次提到了在应用智能体技术时所存在的一些问题。总结起来，多智能体系统理论研究中其目前存在的主要问题是：① 缺乏明确的多智能体系统设计规范以及较为严格、系统、有效的理论与方法体系；② 缺乏强有力的智能体互操作和多智能体系统开发工具，尤其是缺乏软件可重用工具，难以建立已有系统模型与设计到实现间的映射关系，系统的可扩展性、开放性差；③ 缺乏社会范围内的支持。这些问题是智能体技术所面临的挑战，也是未来智能体技术发展的重要方向。

17.6.5 合作型智能体和多智能体协作系统的设计

1. 一种合作型智能体的体系结构 ARCHON

ARCHON 是英国伦敦大学研制的一个合作型智能体体系结构，作为设计合作型智能体的通用目的框架，它提供了一些基础技术来实现合作的问题求解。ARCHON 的体系结构分为两个层次：合作和控制层、领域层。领域层可以是预先存在的或特别设计的，用于执行应用领域的具体任务。合作和控制层可视为操作于领域层之上的元级控制器，以确保一个智能体的本地领域层活动与整个多智能体系统中的其它智能体协调一致。元级控制包括：决定什么任务应在本地执行；决定何时和如何与其它智能体合作；接收来自其它智能体的合作请求等。从合作和控制层的角度看，领域层所执行的各种特别任务都是由该智能体的技能（Skills）来调度的。尽管技能的实现涉及一至多个领域任务的执行，但合作和控制层中的处境分析和合作模块只将技能视为不可分割的原子单元，只有控制模块才能看到技能行为的结构化表示。

合作和控制层有三个主要的处理模块：控制、处境分析和合作。每个模块均是一个正向链产生式系统；拥有自己的推理机和工作存贮。这些模块间的通信以消息传送的方式进行，以便消除隐含的交互作用和传递抽象的语义，获取更强的可理解性、可靠性和对于存取权限的控制；并促进智能体间协作过程的清晰化表示。包含于各模块中的规则是通用的（独立于应用域）知识，适用于在大范围真实应用中控制合作活动。

控制模块作为与领域层的接口，负责管理与领域层的所有交互活动。处境分析模块制定影响合作模块和控制模块的决策；决定要展开的处理活动是在本地进行还是派遣给其它智能体；决定应该响应哪些合作请求，如何执行请求的服务（任务），应执行什么动作去应答新到达的信息，等等。该模块发送请求到其它两个模块，并接收来自它们的反馈。典型的发向合作模块的请求有“获取信息 X”、“请某个智能体执行任务 Y”等；典型的发向控制模块的请求有“悬挂任务 T1”、“启动任务 T1”等。合作模块负责管理本地智能体的社交活动，以满足处境分析模块提出的请求和为处境分析模块的决策提供依据。需支持的主要社交目标有三个：建立新的社交（如寻找一个能提供所需信息的智能体），追踪进展中的合作活动，应答其它智能体发起的合作建议。

其它部件提供支持功能。如，信息存贮部件是一个支持处境分析的参考信息库，其记载领域层任务产生的结果以及本地智能体关注的协同工作进展状况和中间结果。熟人模型和自身模型是对一个多智能体系统中其它智能体和本地智能体的抽象描述，构成决定智能体个体

行为的应用域特有知识。熟人模型描述熟人（其它智能体）的技能、兴趣、当前状态和工作负载等，形成多智能体协同工作的上下文刻画，是协调合作活动所必需的基础；自身模型描述本地智能体自身拥有的技能，以及这些技能的结构化行为。通信管理部件负责发送消息到其它智能体，并接收来自其它智能体的消息。

处境分析模块构成合作型智能体的核心思考部件，它分析协同工作的现状，并制定和修改本地智能体参与协同工作的联合规划；合作模块通过与其它智能体交换 BDI 状态信息，支持协同工作的规划；而控制模块则执行制定的规划，包括通过合作模块请求其它智能体提供规划中拟定的服务和信息，并监控本地技能的运作（即领域层任务的执行）。控制模块能取用存放于自身模型中的关于本地技能如何运作的结构化知识。这些知识表示为预先定义的由多个分支构成的或图，每个分支由需顺序执行（自上而下）的若干计划构成，指示技能执行的一个可选方式。

ARCHON 的分层和模块化体系结构十分适应于计算机主流应用环境，尤其是将领域层从合作和控制层分离出来，使得各个用户的“遗产”软件系统能以较小的修改代价归并进多智能体系统。而面向控制模块知识和技能运作知识的描述语言，则使得控制模块能够屏蔽领域层系统各构成部件的异质性，从而允许合作和控制层能操作于同质的表示环境。

2. 多智能体协作系统的设计的一般原则

在介绍如何基于 ARCHON 体系结构来设计一个多智能体协作系统之前，我们首先阐述设计多智能体协作系统的一般原则。

多智能体系统的设计通常建立于对应用领域的需求分析和将其转变为多智能体问题分析的基础上。因此，设计的焦点也集中于确定智能体粒度（软件尺度）、鉴别每个智能体角色、设计用户界面、定义智能体技能和智能体间传递的消息。只有这些设计完成之后，才能例示智能体的个体结构。

（1）确定智能体的粒度 为减少重构“遗产”软件的工作量，智能体的粒度应与这些软件相同，只有在应用域需求增长或重构以及原软件粒度本来就设置不合理的情况下才需改变。例如在某一应用领域，若直接映射构成原系统的两个专家系统为两个智能体，可能不利于开发多智能体系统协同工作的潜力；但分解原系统为六个智能体后，就可使多智能体并发工作和异步交互作用的优势得到充分发挥，则将其分解为六个智能体。限制智能体尺度的其它重要因素还有计算资源的可用性、问题的自然分布和结构、重构“遗产”软件的难易性等。

（2）确定智能体的角色 在确定粒度后，应为每个智能体分配功能角色并确定整个多智能体系统的功能。如果“遗产”软件已满足了所有的必要功能，则让各相应的智能体保持原有的角色，仅需扩充适当合作功能。然而在“遗产”软件不满足需求的情况下，确定各智能体的角色要花费较多的时间和精力。首先整个应用系统的职能应加以描述，然后划分出可鉴别的子问题，再映射为技能并分配到适当的智能体。

（3）设计智能体的用户界面 接下来需要为每个智能体设计用户接口。基本的决策问题是：是将所有智能体隐藏在单一用户界面后，从而使多智能体系统的功能透明（即内部交互作用不可见）；还是让每个智能体有自己的用户界面，从而使多智能体系统内的交互作用可见。一般来讲，前者适合于采用集中控制方式的应用，后者则更有利于地理上分布或采用分布控制方式的应用。

（4）定义智能体的技能 智能体功能角色取决于其拥有的技能。在智能体领域层作全新设计的情况下，设计者有较高的自由度决策各智能体技能的定义；但对于“遗产”软件，技

能的定义要受软件原有结构的约束。技能配置的总的原则是在填充各智能体拟定角色的同时，使技能执行的任务粒度尽可能大。换言之，任务粒度既应大到足以确保技能的执行效率，又要小到能简要地填充智能体角色。此外，还必须定义各技能的必要和可选输入、以及应产生的中间结果和最终结果等。

(5) 定义消息 基于智能体技能的配置，可以定义智能体间需传递消息的类型，包括内容格式和期望传送的数据量，由 ARCHON 支持的通用消息类型包括请求数据、请求执行技能、主动提供信息、应答请求、报告当前状态、通知执行的失败和通知紧急状态的出现等。此外，应设计确保只发送有用信息的通信机制，以提高通信效率。例如，一个可行的机制是要求消息的发送方只有在确信接收方需要的情况下才主动提供信息，否则必须应接收方请求才发送。

3 基于 ARCHON 的多智能体协作系统的设计

下面，将通过分析一个具体“输电管理”任务对于分布式协同工作的需求，来示例如何构建一个合作型智能体和一个多智能体协作系统。

(1) 基于任务需求分析的多智能体协作系统的总体设计

输电系统管理旨在确保安全供电到所有用户。这涉及到发电厂的维护和修理、为实现稳定操作所需的网络重配置和恢复因故障而停止的电力供应。气候（气温变化、刮风、雷击等）、需求变化（随时间、周日和季节变化）、事故（人为损坏、设备故障、灾害等）和按计划执行的网络维修或扩展工作，都会使电力供应偏离稳定状态。这就要求网络控制中心及时决策应采取的措施。高压电网必须由输电网控制中心统一管理，这意味着任何由现场工程师执行的动作都必须得到控制中心的授权；而低压网的维修动作则可以让现场工程师按预先制定的规范自行操作，只须事后汇报发现的重要情况。

监控输电网是一件极端困难的任務，涉及到分析来自分布于输电网的种种设备产生的大量测量、指示和警报信息。信息往往是不完全的，而采取的应付措施则需及时做出并确保安全性。所以电力公司早已设计了一些应用软件去辅助输电网管理，包括电网切换调度辅助系统、高压诊断专家系统、低压诊断专家系统和气象监视专家系统。这些系统经适当变换后成为在输电管理多智能体系统中参与协同工作的智能体，依次命名为 SSPA, HVDA, LVDA 和 WWA。为支持协同工作和提高输电管理质量，还从 HVDA 分离出两个智能体：TA（作遥感信息处理）和 IA（作为输电网数据的信息查询前端），并建立了三个新智能体：AVA（以规范的方式给输电网控制中心工程师提供网络管理信息），SCA（电网切换检验）和 NSA（故障发生时的电网安全检查）。

(2) 任务中的合作要求

在输电管理中多个智能体协同工作的主要利益之一，在于自动整合不同智能体产生的结果，以确保信息能及时和一致地为相关的智能体共享。例如，为一个例行维护操作而制定的电网切换规划拟定下一天执行，若恰好在此期间维护区域内出现故障，则此规划必须重新制定。通过智能体间的互操作，类似于这种分布于电网各处的信息会受到自动的审核。

在信息丢失或不可获取的情况下，智能体间的互操作可显著提高监控能力的强度。例如，在发生故障时，断路器自动切断故障线路以保护整个电网。TA, HVDA（高压诊断智能体）和 LVDA（低压诊断智能体）都能推测这些做切断操作的断路器的位置。TA 接收到关于某断路器切断操作的遥感信息时，通常能准确定位该断路器。然而存在着三种例外情况：有的断路器未配置遥感，遥感信号丢失（从而电网控制中心未收到），遥感不连续工作。在这些例外情况下，HVDA 和 LVDA 仍然能推测作切断操作的断路器位置。例如，HVDA 可以通过 IA 获取

数据库中存放的电网模型，然后通过仿真确定故障处断路器位置；LVDA 则可依据用户打电话报告的断电情况推测故障处断路器位置。在必要的情况下，TA, HVDA 和 LVDA 可以联合工作，以作出准确判断。

LVDA 也能通过与 WWA 的合作改进诊断能力。当输电网故障出现于雷电气候时，LVDA 可请求 WWA 确定是否有电力设备处于雷击发生处。WWA 能提供雷击处的地理位置，但无电子设备配置的知识。所以若雷击并非出现在故障发生时，WWA 可立即应答 LVDA 的请求。否则，WWA 必须请求 IA 查询雷击处配置的电力设备，再将该电力设备及其位置信息发送给 LVDA。

(3) 智能体合作和控制层的设计

基于对合作输电管理具体任务的理解和多智能体系统的设计原则和方法，接下来的任务就是例示各智能体之 ARCHON 体系结构的合作和控制层，再将其与领域层联接而建立起完整的智能体。将这些智能体组合起来即可形成一个面向具体任务[输电管理]的多智能体系统。下面仅以 LVDA 的故障诊断任务为例，介绍其合作和控制层的详细设计，以及与 TA 和 IA 的交互作用的实现。

[1] 定义智能体 LVDA 的技能

LVDA 的功能角色通过定义其应用的技能来操作化。LVDA 扮演的角色是诊断低压网中出现的故障，并需要由 TA, IA, WWA, HVDA 分别提供的相关信息以及用户打电话传送来的信息。其中，由 TA 传送来的遥感信息可能会延迟或丢失，从而导致最终诊断比一般情况下产生得晚一些，但必须在规定的最后期限前产生。基于这些分析，以下技能被分配给 LVDA：

- DEAL-WITH-TELEMETRY：接收发自 TA 的关于断路器切断线路的遥感信息，建立相应区域的电网模型（基于由 IA 提供的数据库信息），并精化诊断。
- DEAL-WITH-TELEPHONE-CALLS：接收用户电话传送的信息，建立相应区域的电网模型，并精化诊断。
- REPORT-DIAGNOSIS：若电网故障发生后的一个足够长时间内未接收到期望的信息（即遥感或电话信息），则请求 WWA 提供是否在故障区域发生雷击损坏的信息，并产生最终故障报告。
- ACCEPT-HV-FAULT：接收来自 HVDA 提供的关于高压故障（其可能影响低压网）的信息，并用以精化诊断。
- DEAL-WITH-SUBSTATION-FAILURE：接收关于遥感系统失效的报告，并通过采用弥补性措施来精化诊断。

鉴于 ARCHON 的合作和控制层的三个处理模块（控制、处境分析和合作）是通用的（设置了独立于应用领域的推理知识），LVDA 的例示主要是建立自身模型和熟人模型，以获取决定 LVDA 个体行为的应用域特有知识。

[2] 建立智能体自身模型

自身模型定义 LVDA 本地的问题求解活动，包括定义各技能的结构化行为知识和将技能加入自身模型。如，LVDA 的技能 DEAN-WITH-TELEMETRY 的行为的结构化只有一个分支，分支包括顺序执行的两个计划：GET-NETWORK-DATA 和 DIAGNOSE-FAULT，前者建立故障相关区域的电网模型，后者则基于遥感信息和电网模型作诊断。技能的各分支可按从左到右的排序决定其选用优先级，而分支中的计划则按自上而下的顺序执行。

计划 GET-NETWORK-DATA 自身也定义为一个或图，其包含的节点称为 MU（监控单元）。该计划定义如下：

```

PLAN GET-NETWORK-DATA
    :MU          MU-NEED-NETWORK
    :PRECONDITIONS  NIL
    :CHILDREN    (MU-GET-NETWORK :END)
    :MU          MU-GET-NETWORK
    :PRECONDITIONS  ((:output-match NETWORK-DESCRIPTOR))
    :CHILDREN    (:END)

```

作为根节点 (MU), MU-NEED-NETWORK 无条件执行, 其调用领域层任务 TASK-NEED-NETWORK, 并传送参数 TELEMETRY 给它。该 MU 定义如下:

```

(define-MU-type
  :name      MU-NEED-NETWORK
  :is-task   TASK-NEED-NETWORK
  :parametrs (TELEMETRY)
  :outputs   (NETWORK-DESCRIPTOR))

```

该任务的执行返回一个网络描述项:

```

(:message-type NETWORK-DESCRIPTOR)
  :content  ((SUBSTATION SUB1) (VOLTAGE 11000))

```

它描述了 LVDA 需考察电网的哪个区域; 或者返回 FALSE (若该区域电网模型已经下载到 LVDA)。

一旦该 MU 执行完成, 就取用其 CHILDREN 槽中指明的第一个 MU: MU-GET-NETWORK。若这个 MU 的先决条件满足 (即网络描述项已由其父节点产生), 则执行该 MU; 否则该计划的执行成功结束 (因为 CHILDREN 槽中的第二个元素是关键字:END。MU-GET-NETWORK 的定义如下:

```

(define-MU-type
  :name      MU-GET-NETWORK
  :is-task   TASK-CREATE-NETWORK-MODEL
  :parameters ((NETWORK-DATA :para NETWORK-DESCRIPTOR))
  :output    NIL)

```

在调用领域层任务 TASK-CREATE-NETWORK-MODEL 之前, 必须确保所有调用参数都可用。若参数 NET-WORK-DATA 尚不可用 (即故障区域电网模型还未下载), 则控制模块发消息到处境分析模块, 请求提供该参数 (即区域电网模型), 并传递 NETWORK-DESCRIPTOR 用于指明故障区域。处境分析模块决定请求其它某个智能体提供参数, 并经合作模块确定 IA 有该技能后, 将下面的请求:

```

(INFO-REQUEST NETWORK-DATA
  :parameters ((SUBSTATION SUB1)
               (VOLTAGE 11000)))

```

发送给 IA。IA 将最终提供故障区域电网模型给 LVDA, 并使控制模块能启动 MU-GET-NETWORK 中指向的领域层任务 TASK-CREATE-MODEL。由于该 MU 无子节点 (其 CHILDREN 槽内容为:END), 整个计划 GET-NETWORK-DATA 的执行成功结束。

该计划的成功执行为下一计划 DIAGNOSE-FAULT 的执行奠定基础。下一计划只包含单一 MU, 定义如下:

PLAN DIAGNOSE-FAULT

```
:MU                MU-DIAGNOSE-FAULT
:PRECONDITION      NIL
:CHILDREN          (:END)
(define-MU-type
  :name             MU-DIAGNOSE-FAULT
  :is-task          TASK-DIAGNOSE-FAULT
  :parameters       (TELEMETRY)
  :output           (LVDA-DIAGNOSIS))
```

MU-DIAGNOSE-FAULT 无条件执行，导致领域层任务 TASK-DIAGNOSE-FAULT 被调用。调用前必须使参数 TELEMETRY 可用，若 TELEMETRY 尚不可用，则 LVDA 请求 TA 提供之。该任务执行时，首先核对由 TA 提供的 TELEMETRY 是否相关于当前假设的故障区域。若不相关，则建立新的故障假设，并设置定时器。定时器指示当故障出现时，接收到所有相关数据所需等待的时间。这个时间应足够长，因为来自用户电话报告的故障情况不会很快接收到。若 TELEMETRY 相关于当前假设的故障区域，则可用以精化该假设。期间，LVDA 的其它技能（如 DEAL-WITH-TELEPHONE-CALLS 和 ACCEPT-HV-FAULT）也可启用，以进一步精化诊断。

一旦 MU-DIAGNOSE-FAULT 执行完成，诊断结果 LVDA-DIAGNOSIS 产生并输出。由于 DIAGNOSE-FAULT 计划不包含其它 MUs，而技能 DEAL-WITH-TELEMETRY 的这个分支又无进一步计划存在，所以该技能的执行完成。

每个技能在其结构化定义完成后，应加进 SA 自身模型，以便合作和控制层的三个处理模块能用其于推理。以上定义的技能 DEAL-WITH-TELEMETRY 可用下面的表示方式加进 LVDA 自身模型：

```
Name:              DEAL-WITH-TELEMETRY
Trigger:           TELEMETRY-TRIGGER-FOR-LVDA
Inputs:            ((TELEMETRY: mandatory))
Result:            (LVDA-DIAGNOSIS)
Plan-Name:         PLAN-GET-NETWORK-DATA
Children:          (DIAGNOSE-FAULT)
```

在许多工业应用中，智能体间拟交换的数据往往有复杂的结构。这种结构必须定义于自身模型，包括说明数据的构成部件及其类型，并让 MA 系统中的所有智能体均理解。例如，上述数据 TELEMETRY 是类 TELEMETRY-MESSAGE-TYPE 的实例，该数据类可定义如下：

```
Type-Create:       TELEMETRY-MESSAGE-TYPE
Component:         ((TEXT: STRING) (TIME: INTEGER)
                   (TEXT-TIME: STRING) (SUBSTATION: STRING)
                   (PLANT: STRING) (OPERATION: STRING)
                   (SOURCE: STRING) (VOLTAGE: INTEGER))
```

[3]合作问题求解活动

为展开合作问题求解活动，智能体需要二个方面的知识：指导智能体间互操作和合作协调的通用知识，应用领域的专门知识。前者表示为供合作和控制层中处境分析和合作模块使用的产生式规则，后者表示为熟人模型和自身模型中的 BDI 状态。鉴于我们的目的在于介绍如

何用 ARCHON 体系结构来设计合作型智能体，而非合作协调的方法论，所以这里不给出关于这两个方面知识的系统化描述，仅就 LVDA 参与合作问题求解活动涉及的知识加以讨论。为用遥感信息进行故障诊断，TA 和 LVDA 的合作模块应提供以下功能。

① TA 主动提供遥感信息给其它智能体。为压缩不必要的通信，需了解想要该信息的智能体的期望。这可以记载于 TA 的熟人模型中。例如，关于 LVDA 对该信息的期望表示如下：

```
Name:          TELEMETRY-INTEREST-FOR-LVDA
Data_Name:     TELEMETRY
Condition:     ( ( ! ! ( VOLTAGE ?X ) ! ! ) ( <= ?X 11000 ) )
```

其中，符号!!和?X是模式匹配指示，前者匹配一个任意长度的模式，后者则束缚到模式中对应符号。应用该期望去产生合作行为的相应通用规则可表示如下：

```
(rule voltage-info
  (if (and (INFO-AVAILABLE ?I)
           (ACQUAINTANCE-INTERESTED-IN ?I ?ACQ) ) )
  (then (SEND ?I TO ?ACQ AS UNREQUESTED-INFORMATION) ) )
```

这里，LVDA 只对低压电网感兴趣，所以其期望表示中设置了电压小于或等于 11kV 的条件。当 TA 主动发送遥感信息时，就检查其熟人模型，确定该信息是否满足某熟人的条件。例如，该信息中描述的电网电压值（参见前述 TELEMETRY-MESSAGE-TYPE 的定义）是否满足 LVDA 期望表示中的条件。只有条件满足时，才向 LVDA 发送该信息。

② 为建立故障区域电网模型，LVDA 需知道哪一个智能体能提供关于网络的结构化信息。如上所述，IA 可提供这些信息（NETWORK-DATA），所以 LVDA 的熟人模型中应加入对 IA 有该技能的信念：

```
Name:          GET-NETWORK-FOR-SUBSTATION
Input:         SUBSTATION
Results:       NETWORK-DATA
```

应用该信念去产生合作行为的相应通用规则可表示如下：

```
(rule generate-service-request
  (if (and (INFO-NEEDED ?I)
           (CANNOT-BE-PRODUCED-LOCALLY ?I)
           (ACQUAINTANCE-CAN-PRODUCE ?I ?ACQ) ) )
  (then (SEND INFO-REQUEST TO ?ACQ TO PRODUCE ?I) ) )
```

③ 当 LVDA 接收到关于故障的遥感信息时，就触发技能 DEAL-WITH-TELEMETRY 的执行。该触发器（即触发条件）定义于 LVDA 的自身模型中，其要求接收到的 TELEMETRY 中关于区域电网的电压小于或等于 11kV（这个条件与设置于 TA 熟人模型中的 LVDA 期望表示相同）：

```
Name:          TELEMETRY-TRIGGER-FOR-LVDA
Data-Name:     TELEMETRY
Condition:     ( ( ( ! ! VOLTAGE ?X ) ! ! ) ( <= ?X 11000 ) )
```

相应的通用规则表示如下：

```
(rule start-skill
  (if (and (RECEIVE-INFO ?I)
           (INFO-IS-SKILL-TRIGGER ?I ?SKILL) ) )
```

(TRIGGER-SATISFIED ?SKILL)))

(then (EXECUTE-SKILL ?SKILL)))

(4) 智能体合作和控制层与领域层的联接

ARCHON 体系结构的合作和控制层实现对领域层任务的元级控制，控制知识表示为技能行为的结构化描述。如此，对领域层任务的调用实际上仅是在元级控制下的异步函数调用。

不过，对于“遗产”软件的调用并没有想象的那样简单，因为这些软件的控制结构仅从孤岛型应用的角度设计，往往不能有效地支持协同工作。例如，在一些应用中，原有的诊断专家系统将诊断过程的基本步组织于循环中，外部对控制的影响只能通过插入控制信息于该专家系统的事务表来间接实现。这显然不利于多智能体系统对协同工作的协调控制。把诊断过程的基本步定义为技能，并将顶层控制循环迁移到合作和控制层，可以促进智能体间合作的协调控制，并使某些基本步提供的计算和信息服务可以为多个智能体共享。又如，与 LVDA 相应的原低压诊断专家系统具有自己的电网模型，由于改造为多智能体系统后，该电网模型已由另一智能体（即 IA）提供，故将诊断过程的顶层控制迁移到合作和控制层，并把基本步改造为 LVDA 拥有的技能（即前述的五个技能）。

(5) 多智能体系统的建立 只要设计了所有参与协同工作的智能体，建立多智能体系统是水到渠成的事。那就是配置智能体到分布计算环境的适当站点，指定网络地址，并启动整个系统。当然，还要做某些检查和测试工作。首先，应校核各智能体的熟人模型和自身模型，以确保 BDI 状态表示的一致性。例如，TA 熟人模型中关于 LVDA 对于 TELEMETRY 的期望表示和 LVDA 自身模型中 TELEMETRY 触发器的表示应一致。测试工作在仿真环境中进行，不仅应测试 MA 系统的拟定功能能否实现，通信效率、一致性和可靠性的评价也是重要的。

17.6.6 多智能体间协作的实现（方法）

在真实世界应用环境中，设计一个多智能体系统所面临的巨大挑战是如何协调多个智能体的行为以达到协同工作的紧凑一致性。合作的协调要求每个智能体都确保在合适的时间做正确的事情，从而① 避免不必要的、冗余的动作；② 满足最后期限；③ 预先提供以后需要的有促进作用或有帮助的信息；④ 平衡 SA 间的负载。然而，在很多情况下，实现合作的协调是困难的，这在复杂、动态、环境状态不可预测以及资源受限（时间、费用、计算服务共享）的情况下尤为突出。可以说，环境的动态性和每个智能体对环境观察的不完全性所引起的不确定性是合作难以协调的主要根源。为解决此难题，给每个智能体建立关于合作问题求解（广义地，协同工作）的清晰模型是必要的。模型不仅要描述合作过程按规划正常进展时智能体应如何举止，也须说明意外事件发生时应如何处理。为建立描述协同工作的清晰模型，詹宁斯提出的基于联合意向（Joint Intention）的联合负责（Joint Responsibility）模型，为此提供了有效的表示手段，可显著地促进合作的协调。该模型旨在制定实现协调合作的各种要求和准则，并给出了应用该模型去开发实用系统的途径。下面，我就以该模型为例，讨论如何实现合作的协调。首先介绍联合负责模型及其依赖的基本概念，然后以示例阐述如何实现基于联合负责模型的多智能体协作管理，包括联合行动的建立以及联合行动过程中合作的协调。

17.6.6.1 联合负责模型简介

为实现协调的合作，目前流行的控制方式可分为两类：集中式和分布式。前者将合作和协调的控制权赋予多智能体系统的某个智能体（或固定层次结构中的各较上层智能体），使其具有对协同工作的完全观察，并能全面控制和协调智能体间的合作过程。然而在复杂的应

用域，尤其是要求紧耦合协作的情况下（即使在松散型协同工作的多智能体系统中，也有需紧耦合协作的时刻），集中式控制会引起控制和通信瓶颈。而且集中式控制在承担控制职责的智能体发生故障时会引发灾难性的后果。分布式控制可以避免集中式控制的弊病，但其面临的主要困难是不能有效地实现协调一致的整体行为，尤其是在环境动态变化、智能体感知和动作易犯错误的场合。

随着分布计算环境的普及应用和协作任务的日趋复杂，分布式控制逐渐成为主流方式，如何建立这种控制方式的合作问题求解模型已成为智能体技术和分布式人工智能研究的共同热点。传统的合作问题求解模型存在两个主要的问题：合作行为的隐含表示和合作协调的隐含表示。詹宁斯所提出的联合负责模型，直接依赖于二个重要概念：对于联合目标和公共处方的联合承诺。下面，我们将首先引入个体意向（Intention）和联合意向的概念，然后阐述联合目标和公共处方，并由此定义联合负责。

1. 个体意向和联合意向

个体意向是在一定的心理状态下承诺要付诸的行动，布拉特蒙（Bratman）认为，由智能体意向控制的个体行为有三个重要方面：① 当一个智能体决定追求一目标时，它就承诺导致一个特别的结果。② 意向用于组织将来的动作，即一个智能体必须依据其承诺追求的目标所在的上下文制定子序列决策—在一些特别的时刻执行必要的动作。③ 意向导致“手段-目的”分析，因为智能体总是先承诺追求其目标，然后再规划如何实现目标的所有低级细节。

联合意向可定义为在一定的共享心理状态下对于执行某个动作集的联合承诺。目前已出现若干关于联合意向的形式模型，詹宁斯在评价和综合这些模型的基础上，提出以联合意向来定义合作行为的要点：① 智能体必须有联合目标；② 智能体赞同通过合作达到联合目标；③ 智能体赞同为到达联合目标而制定的“公共处方”（Recipe）；④ 在联合行动的上下文中由不同智能体执行的动作是相互依赖的。⑤ 智能体必须有公约用于监视它们承诺的可行性。

显然，实现合作行为的关键在于需要有智能体群组期望达到的公共（联合）目标，并且所有智能体均赞同以合作的方式达到该目标。实际上，公共目标成为组合个体行动为内聚整体的粘合剂。此外，约定以合作的方式达到公共目标，使得拟定目标的合作能清晰地区别于恰好有相同目标的独立智能体的交互作用。这种区分是重要的，因为将导致不同的合作行为。公共目标和合作意向的确定使得智能体联合承诺达到它们的联合目标，导致合作的协调；而恰好有相同目标的独立智能体在资源短缺的情况下会导致难以协调的竞争。

应注意的是，仅建立联合目标并不足以确保协调地合作。例如，一个国家的政府和独立于政府的国家银行都将降低通货膨胀作为共同目标，并愿意合作达此目标。然而除非双方协商制定达此目标的一个公共处方，不可能存在有效的合作活动。公共处方是对通过联合行动来达到共同目标的一个规划，提供了上下文去指导个体制定其在联合行动中的具体活动。

2. 智能体的联合目标

联合负责模型借用科恩和莱维斯科的联合意向理论去表示对联合目标的联合承诺。承诺建立在联合目标（或称为持久联合目标）的基础上，而联合目标又依赖于所谓的完成目标这一概念。完成目标用于定义在联合目标是产生 p 而动机为 q 的情况下，参与协同工作的个体 a 具有的心理状态，并可区分为两个状态：① 正规完成目标— a 尚不相信 p 为真（所以其成为追求的目标），并将使 p 最终变为真作为目标。② 弱完成目标—正规完成目标是其一种

情况，其它三种情况是：当 a 确信 p 已为真、决不可能为真或不相关（即 q 为假）时，a 将让协同工作的所有伙伴都相信这个事实作为目标。

为此，当且仅当所有智能体都有正规完成目标时，才能说协作组成员拥有动机为 q 且想要产生 p 的联合目标。在协作组联合承诺产生 p 时，每个成员都将产生 p 作为正规完成目标。但随着时间的推移，某个或某些成员会不再将产生 p 作为正规完成目标，转而将其作为弱完成目标。原因在于它们在联合行动过程中发现该目标已满足、不可能满足或不相关，并转而致力于让这个事实为其它成员相信。鉴于产生 p 已不再作为某些成员的正规完成目标，产生 p 的联合目标和对于该联合目标的联合承诺也就不再有效，进而导致联合行动的不一致。

然而，由于相应的弱完成目标仍存在于这些发现问题的成员，其它成员最终都会被通知到联合目标和联合承诺的失效。这意味着每个智能体的行为都依赖对其它智能体所作的承诺，首先是对正规完成目标的承诺，然后是对弱完成目标的承诺（即在联合行动出现问题的情况下，将问题通知到其它智能体）。具有弱完成目标对于联合行动的协调一致十分重要，因为不正常现象和意外事件总是只能为单个或少数成员所发现，及时让其它成员了解真相有利于快速调整它们的行动，避免徒劳甚至有害的动作，进而确保联合行动的一致性。

归纳起来，在一组智能体致力于追求某联合目标时，为确保联合行动的一致性，需要协作组中每个智能体承诺将该目标作为自己的正规完成目标，并通过处理异常情况的公约去监控目标的实现。一旦某智能体发现承诺无法遵守（因确信目标已满足、不再可能满足或追求该目标的动机已不成立），它必须遵从公约采用适当的行动，使其它 SA 都知道真相。

3. 对于公共处方的联合承诺

对于公共处方的联合承诺是通过所谓的联合处方承诺来说明的，联合处方承诺又建立在个体处方承诺的基础上。个体处方承诺说明协作组的每个成员应遵守承诺，执行其赞同承担的由公共处方拟定的动作，除非以下情况出现：① 一个处方步（拟定动作）的期望结果已经获得。例如，结果已经由另一个动作序列产生，或由另一个智能体提供。② 实施该处方不再可能获得想要的结果。例如，环境的变化导致处方中拟定的动作系列不再可能完成。③ 某个由处方拟定的特定动作无法执行。例如，为该动作提供处理信息的传感器出现故障。④ 某个由处方拟定的特定动作无法在预定的时间执行。例如，承诺执行该动作的智能体因处理某个意外的但优先级更高的事件而不得不推迟执行该动作。

协作组的成员可以通过以下方式发现公共处方的执行出现异常情况：① 其自身卷入错误行为，例如未能在预定的时间执行拟定的动作。② 根据观察到的伙伴成员的行为，推测出某个异常情况已发生。③ 接收到伙伴成员发送来的关于异常情况的消息。由此，一旦某智能体发现公共处方的执行出现异常情况，就可以分析自身的处境，并决定是否应放弃对该处方承诺的责任。

个体处方承诺定义每个智能体应如何按照其在公共处方中承诺的责任行动。当一智能体发现公共处方的执行出现异常情况时，它不应该简单地放弃对该处方的承诺，因为伙伴成员可能尚未发现这些异常情况，所以需要建立联合处方承诺，用以说明每当一智能体放弃其承诺时，必须设法将此意向通知给协作组的所有其它成员的责任，这样一来，整个协作组可以对处方进行重新分析，以决定放弃或加以修改。由于无用的活动因此而尽早中止，资源的浪费可望最小化。

4. 定义联合负责

联合负责模型通过综合应用上述关于联合目标和公共处方的概念和基本原则来清晰地定义协同工作。它要求协作组的所有成员 (a_1, a_2, \dots, a_n) 都有一个与追求联合目标 σ (如产生 p) 相关的完成目标, 都依据联合处方承诺的原则执行面向 σ 的公共处方 Σ , 并在展开联合行动时都相互了解每个成员在做什么。下面给出这个定义协同工作的联合负责模型。

联合负责 ($a_1, a_2, \dots, a_n, \sigma, \Sigma$):

对于每个 $a_i \in \{a_1, a_2, \dots, a_n\}$,

当 a_i 将联合目标 σ 视为正规完成目标, 且 a_i 对面向 σ 的公共处方 Σ 维持其个体:

处方承诺时, 并发进行以下活动:

遵守对联合目标和公共处方的承诺,

监控对 σ 所作承诺的合理性,

监控对 Σ 所作承诺的合理性;

暂停执行与 Σ 相关的本地动作;

根据违反承诺的原因作以下处理:

若原因是 a_i 不再将 σ 视为正规完成目标,

则 放弃对 σ 的承诺, 并撤消 Σ 中拟定由 a_i 执行的动作, 否则,

若原因是 a_i 不再对面向 σ 的 Σ 维持其个体处方承诺,

则 若 可以修改有问题的动作,

则 选择可行的修改,

否则 设计新的公共处方并寻找必要的辅助;

通知协作组其它成员关于承诺的违反、违反的理由和可能的修改动作。

在联合负责模型中, 区分对联合目标和公共处方这两种不同类型的联合承诺是必要的, 因需要不同的处理。在取消对联合目标承诺的情况下, 联合行动终结: 若目标已满足, 协同工作成功完成; 若追求目标的动机消失或目标不可能满足, 则没有必要继续联合行动。然而, 在取消对公共处方承诺的情况下, 可以存在继续联合行动的需要。例如, 若问题出在联合行动中由不同智能体执行的动作定时不合理, 则重新定时; 若公共处方无法修复, 就重新设计它。可见, 取消对这两种类型的联合承诺, 会引起很不相同的后果。

17.6.6.2 基于联合负责模型开发的原型系统 Grate* 【过时内容 仅供参考】

联合负责模型提出了设计智能体协同工作的一个高层体系结构, 它以联合意向作为核心概念去指导智能体间联合行动的协调和控制。然而, 该模型只给出对多智能体系统设计的高级约束, 留下充分的余地供应用开发者去细化设计适合需要的体系结构。例如, 如何表示意向和承诺, 用什么机制来进行智能体间协商, 如何建立联合目标和发展公共处方等, 都可由多智能体系统开发者自由设定。

詹宁斯应用这个联合负责模型开发了原型系统 Grate* (Generic rules and agent model testbed environment) 作为建立多智能体系统的通用框架。Grate* 以 ARCHON 作为建立个体智能体的体系结构, 并应用联合负责模型来改进[输电管理]系统的性能。

Grate* 以产生式规则形式表示联合负责的操作语义。规则划分为两组, 分别用于分析协作组处境和管理社交活动(控制合作时的互操作)。按照联合负责模型的定义和联合意向的特性, 需设计处境分析规则去决定面向某联合目标的合作何时合适, 发展和获得对于公共处方的赞同, 确保遵守已作的承诺, 确保新的和已有的承诺相一致, 监控问题求解状态并依据公约决定在必要时放弃承诺, 决定在承诺放弃的情况下应采取的动作。管理社交活动的规

则用于确保在成员放弃承诺时让协作组其余成员都得到通知,并知道放弃承诺的理由和建议的修补动作(若有的话)。如同描述智能体技能那样,应定义相应的数据结构去表示目标、处方、意向和联合意向。

Grate*应用于输电管理时,对原先的多智能体系统设计作了修改(包括智能体的命名和技能配置)。这里不准备作全面描述,仅从基于联合负责模型实现合作协调的角度,集中注意力于三个智能体和它们将协同解决的故障探测和诊断问题。三个智能体中的两个—警报分析智能体(AAA)和断电区域鉴别器(BAI)实际地执行诊断(但精度不同并基于不同的信息);第三个智能体—控制系统接口(CSI),探测电网障碍的发生,并监视电网状态的发展。

17.6.6.3 智能体联合行动的建立

在协同工作能进展前,应首先建立协作组成员间赞同一致的联合行动。对联合行动的需求起源于某个智能体对于自身处境状态的分析。当这个智能体试图实现某个期望的目标时,它必须决定达此目标的全部活动是否能在本地完成还是需其它智能体的辅助。分析的结果有三种情况:需要联合行动、完全在本地解决或需要其它智能体辅助,但不必有联合负责意义上的联合行动。

若该智能体决定需要联合行动,它就是联合行动的组织者,并承担以下工作:(1)鉴别愿参与协作的成员;(2)让协作组成员都认可需建立一公共处方,并赞同使用联合负责模型;(3)实际地设计需要的公共处方。

为提高工作效率,Grate*将建立联合行动的所有需求描述于拟发送的单一消息中,并采用一双相协议去征求协作组潜在成员的赞同。协议如下:

第一相:

组织者探测到需建立联合行动去达到目标G,并决定处方R是达此目标的最好手段;

组织者与能对R做贡献的所有熟人接触,以了解它们是否愿意参加这个基于联合负责模型的联合行动;

令 Ω 为愿意参加协作的熟人集合。

第二相:

对于R中所有拟定的动作,

选择熟人A($A \in \Omega$)去执行动作 θ ($\theta \in R$);

(遵循准则:最小化协作组成员个数)

计算 θ 拟定被执行时刻 t_θ ;

(基于 θ 在R中的时间安排和预计的通信延迟)

发送建议(θ, t_θ)给A;

A参照已存在的承诺集(C)评价该建议:

若(θ, t_θ)不与C冲突,则建立A对(θ, t_θ)的承诺 C_θ ;

否则,

若冲突且承诺 θ 的优先级大于C的优先级,则建立 C_θ 并重安排

C;否则,

若冲突且承诺 θ 的优先级小于C的优先级,则寻找某个空闲时刻

($t_\theta + \Delta t_\theta$),注释 C_θ ;

返回接受信息或修改后的时刻给组织者;

组织者决定是否采纳修改后的时刻，

若采纳，则依据 t_0 修改 R 中其它动作的执行时刻。

协议的第一相确定哪些熟人愿意参加协作并遵从联合负责原则，类似于合同网协议的任务发布相。第二相则由组织者指派联合行动中每个拟定动作应由谁去执行、在什么时刻执行。在复杂的动态变化环境下，通过双相协议来协商联合行动是必要的，因为组织者不可能获取制定联合动作所需的全部信息，也不了解潜在的协作组成员当前的期望和已作的承诺。分两相进行协商可以降低协商过程的复杂度。

设想上述的输电管理应用中，AAA 接收到电网故障的警报，并决定需要一个联合行动来诊断故障。指导这个联合行动的联合意向已在系统设计阶段制定，并置于 AAA 的自身模型中。该联合意向定义如下：

```
Name: (DIAGNOSE-FAULT)
Motivation: INFO-AVAILABLE DISTURBANCE-DETECTION-MESSAGE)
Recipe:
  ( (PAR (START-OFF (IDENTIFY-BOA ?BLACK-OUT-AREA)
    (START-OFF (HYPOTHESIS-GENERATION ?BLOCK-ALARMS
      ?INITIAL-HYPOTHESES) )
    (START-OFF (MONITOR-DISTURBANCE) ) )
  (START-WHEN
    (INFO-AVAILABLE ?INITIAL-HYPOTHESES)
    (DETAILED-DIAGNOSIS ?INITIAL-HYPOTHESES?BLACK-OUT-
AREA ?VALIDATED-FAULT-HYPOTHESES) ) )
Start Time: -
Maximum End Time: -
Duration: -
Priority: 20
Status: ESTABLISHING-GROUP
Outcome: (VALIDATED-FAULT-HYPOTHESES)
Participants: ( (SELF ORGANISER AGREED-OBJECTIVE)
  (CSI TEAM-MEMBER AGREEING-OBJECTIVE)
  (BAI TEAM-MEMBER AGREEING-OBJECTIVE) )
Bindings: NIL
Proposed Contribution
  ( (SELF ( (HYPOTHESIS-GENERATION YES)
    (DETAILED-DIAGNOSIS YES) ) )
  (BAI (IDENTIFY-BOA ?) )
  (CSI (MONITOR-DISTURBANCE ?) ) )
```

其中，Motivation 槽指出执行联合行动的理由是发现了电网障碍。故障诊断的处方则由一系列需执行的动作构成，并指出这些动作执行的排序。该处方说明，IDENTIFY-BOA（鉴别断电区域）、HYPOTHESIS-GENERATION（产生故障诊断的初步假设）、MONITOR-DISTURBANCE（监控电网障碍的发展）这三个动作应并发执行，然后只要获得初步的假设列表，就启动

DETAILED-DIAGNOSIS（细化诊断）动作。处方只说明需做什么，而非由谁去做和应在何时执行，后者在协商协议的第二相决定。

Priority 槽指明意向（包括本地意向和联合意向）的重要程度，以支持在多个意向竞争时选择优先级高的执行。优先级由静态的固有优先级指数和动态优先级指数两个部分叠加而成。动态优先级取决于意向是否涉及联合行动，以获得必要的灵活性。这里，意向的固有优先级为 10，动态优先级也为 10，叠加为 20。

Status 槽指出联合行动的建立状态，其有三个可能值：establishing-group，developing-solution 或 executing-joint-action。Outcome 槽给出意向执行期望的结果，Participants 槽指示协作组的构成和每个成员的卷入状态。这个为诊断故障所需的联合行动应包括三个成员：组织者（AAA 自身）和其它两个参与者（BAI 和 CSI）。既然 AAA 是组织者，它必定赞成参与联合行动，而 BAI 和 CSI 则需协商后才能确定。Bindings 槽专用于协商协议的第二相。

Proposed Contribution 槽用于记载由组织者判定能为联合行动作贡献的 SA，并说明它们是否已赞成作出贡献。由于 HYPOTHESIS-GENERATION 和 DETAILED-DIAGNOSIS 动作由 AAA 自行执行，当然必定承诺执行之；另两个动作分别请求 BAI 和 CSI 执行，必须征求它们的承诺。

一旦鉴别了联合行动的需要，建立协作组的过程就可开始（Status 槽的值为 establishing-group），以确定哪些熟人愿意参与协作。第①步由组织者的处境分析模块通知其合作模块：需要社交活动。然后由合作模块根据组织者熟人模型中提供的关于熟人技能的知识，鉴别能参与联合行动的所有潜在熟人。由于在工业应用环境下智能体的配置相对稳定，技能知识可认为是足够精确和全面的；然而在更开放性的环境下（例如 SA 或许频繁地加入和离开 MA 系统），则应采用更为动态的方式。这个阶段的策略是鉴别尽可能多的有能力参与联合行动的熟人，以便组织者在协商协议的第二相有足够的选择余地。一旦鉴别完成，合作模块就向这些熟人发送邀请它们参与联合行动的消息（第②步）。该消息包括了记载于正在被例示的这个联合意向中的许多信息，如意向名、动机、处方、优先级、期望的结果等。

每个潜在的参与者通过其合作模块接收到该消息，并核对是否与自己相关。若不相关，则产生一个否定回答（第③'步）。否则，将消息传送给处境分析模块（第③步），由其分析该智能体是否有足够资源承担联合行动中指派的动作。这涉及到分析是否与已承诺的其它活动冲突和能否成功地执行该动作。无论分析的结果是接受或否定，都将回答传送合作模块（第④步）。最后合作模块把回答反馈给组织者（第⑤步）。若潜在的参与者愿意参与，则其建立一个相应的联合意向表示于其自身模型中。

通过返回应答消息给 AAA，BAI 和 CSI 都表示愿意参与关于诊断故障的联合行动。这两个智能体也赞同建立面向联合行动的公共处方，并遵从联合负责模型去维持合作。但这些准则并不经由通信清晰地在智能体间交流，因此这些知识隐含地表示于处境分析和合作模块中。

在组织者接收到所有潜在参与者的回答消息后，建立联合行动的第二相开始。这时组织者将该联合意向的 Status 槽的值改为 developing-solution，并修改 Participants 槽去反映实际情况（槽值为愿意参与联合行动的 SA 列表）。组织者按一定的原则决策参与者的选择。在输电管理应用中，采用使协作组成员最少化的原则，以降低智能体自身故障的可能性和压缩通信费用（不同的应用域可能会采用不同的原则，例如，在要求平衡负载的智能体系统中，或许采用最多化原则）。组织者更新 Proposed Condition 槽的内容去指示选择的结

果:

Proposed Condition

```
( (SELF ( (HYPOTHESIS-GENERATION SELECTED)
          (DETAILED-DIAGNOSIS SELECTED) ) )
  (BAI (IDENTIFY-BOA SELECTED) )
  (CSI (MONITOR-DISTURBANCE SELECTED) ) )
```

然后, 组织者为处方中每个动作拟定执行的时刻, 并填充联合行动的持续时间、起始和结束时间。动作和拟定执行的时刻组合为将发送给选定的参与者的建议, 记载于 Bindings 槽中。例如, AAA 联合意向的 Bindings 槽内容如下:

Bindings:

```
( (BAI IDENTIFY-BOA 19)
  (SELF HYPOTHESIS-GENERATION 19)
  (CSI MONITOR-DISTURBANCE 19)
  (SELF DETAILED-DIAGNOSIS 36) )
```

这些建议发送给选定的参与者后, 参与者作评价并决定是否接受。若未发现冲突, 则其采用 AAA 的联合意向, 并建立相应的个体意向表示于自身模型中。下面是 BAI 建立的鉴别断电区域的个体意向:

```
Name: (ACHIVE (IDENTIFY-BOA) )
Motivation: (SATISFY-JOINT-ACTION (DIAGNOSE-FAULT) )
Recipe: (IDENTIFY-BOA)
Start Time: 19           Maximum End Time: 34
Duration: 15            Priority: 5
Status: PENDING         Outcome: (BLACK-OUT-AREA)
```

然后, 选定的参与者返回应答消息给组织者。若拟定的动作执行时刻不可接受, 参与者建议一个可行的别的时刻, 作为试探性的暂时承诺, 反馈给组织者。组织者可能会认同这个修改后的时刻, 并将其 Bindings 槽内容作适当调整。若修改不可接受, 则组织者将在 Participants 槽中寻找别的智能体去执行该动作。

关于动作执行的时间, 协商可能会反复进行, 直到全部成功为止。这时, 就产生赞同一致的公共处方, 并由组织者将这个结果通知所有愿意参与联合行动的智能体。联合行动已成为可操作, 一旦联合行动启动操作, 组织者就将 Status 槽内容改为 `executing-joint-action`, 并更新 Participants 槽, 以指示协作组成员都已处于执行该联合行动的过程中。

17.6.6.4 联合行动过程中合作的协调

一旦联合行动开始执行, 协作组成员就必须监控执行过程。如前所述, 联合负责模型已经说明智能体应在怎样的处境条件下重新考虑其承诺, 以及这种条件出现时智能体应如何以有益于自身和伙伴的方式采取行动。Grate* 提供了结构和组织框架去支持开发者设计必要的通用监控规则, 以实现遵从联合负责模型的合作协调。

联合负责模型提供一个方法, 用于识别对联合行动的承诺需重新审查的处境和在此处境中应采取的弥补动作。一个智能体的处境分析模块不断监控智能体系统中各种事件的发生。事件可分两大类: 本地领域层系统产生的事件 (如目标完成、信息需求), 和智能体系统中其它智能体引起的社交事件 (如请求提供信息。返回需求的信息)。在大多数情况下, 这些

事件不会影响联合行动。然而在某些情况下，事件会损害联合行动，并迫使该 SA 重新考虑相关的承诺。

在传统的智能体系统中，不存在可用于审查事件是否对联合行动产生损害的有效手段。然而，应用联合负责模型，相关的事件能被清晰地指明。这些事件产生迫使智能体取消对联合目标承诺的条件（即目标已满足、目标不再可能满足、追求目标的动机已不存在）和迫使智能体取消对公共处方承诺的条件（即拟执行动作的结果已经存在、处方被违反、处方失效），可以作为重新审查智能体处境的理由。

“匹配”过程的任务就是鉴别出哪些事件会损害联合行动（从而成为重新审查智能体处境的理由），并产生因它们而迫使智能体取消承诺的条件。关于各种事件的定义超出联合意向模型的范围，必须由智能体系统开发者自行设计。Grate* 将损害联合行动的事件编码为 ARCHON 处境分析模块中产生式规则的前提部分，而规则的结论部分就是联合负责模型定义的导致智能体取消承诺（对联合目标的承诺和对公共处方的承诺）的条件。下面给出四条这样的匹配规则，它们分别说明了什么样的事件会导致联合目标已经满足、联合目标的动机不再成立、公共处方被违反和公共处方变成无效。

匹配规则 R1:

若 任务 t 已结束执行，且产生了联合目标 G 想要的结果；
则 G 已满足。

匹配规则 R2: 若 接收到信息 i，且 i 使联合目标 G 的触发条件不能成立；则 G 的动机不再成立。

匹配规则 R3: 若 动作 t1 延迟，t1 是某联合行动处方 REC 拟定的动作，且 t1 必须与 REC 拟定的另一动作 t2 同步执行；则 REC 被违反。

匹配规则 R4: 若 公共处方 REC 的执行结束，REC 期望的结果未产生，且存在替代 REC 的处方；则 REC 失效。

若上述规则中的任意一条激活，则表示处境分析模块已鉴别出执行某联合行动的意向遇到困难。R1 和 R2 的激活意味着该 SA 不再维持面向联合行动的正规完成目标；而 R3 和 R4 则意指个体处方承诺已出现问题。依据 Grate* 的方法，下一步就是选择适当的弥补动作。选择弥补动作也定义于处境分析模块，表示为产生式规则。规则前提即为激活的匹配规则提供的结果（迫使智能体取消承诺的条件），而规则结论则列举可能的修补动作。例如，当一智能体取消对公共处方的承诺时，它应暂停执行其关联的本地动作并着手对处方作修改；而若该智能体认为联合目标的动机不再成立，则其应取消所有关联的本地动作。下面给出四条这样的选择规则。

选择规则 R1: 若 联合目标已满足；则 取消所有关联的本地活动，通知合作模块：联合行动已成功结束。

选择规则 R2: 若 联合目标的动机不再成立；则 取消所有关联的本地活动，通知合作模块：联合行动的动机不再成立。

选择规则 R3: 若 公共处方 REC 被违反，且 REC 可成功地重新设置；则 暂停与 REC 关联的本地活动，重新设置 REC 中拟定动作的时间安排，通知合作模块：REC 已被违反，建议新的时间安排。

选择规则 R4: 若 公共处方 REC1 失效，且替代性处方 REC2 存在；则 取消与 REC1 关联的本地活动，通知合作模块：REC1 失效，建议 REC2 作为替代性处方。

根据联合负责模型，尽管这些规则的激活仅引起本地行动，但由于这些行动均关系到个体承诺的取消或修改，必须通知其它 SA 也因此而作处境审查和调整。这就要求合作模块依据处境分析模块提供的信息，发送适当的消息给协作组其它成员。下面是四条这样的通知规则。

通知规则 R1：若 联合行动已成功结束；则 通知协作组其它成员关于此成功，决定是否将此消息发送给协作组以外的 MA 系统成员。

通知规则 R2：若 联合目标 G 的动机不再成立；则 通知协作组其它成员：G 需要被取消。

通知规则 R3：若 公共处方已被违反，且新的时间安排建议存在；则 通知协作组其它成员关于此违反，建议将此新的时间安排作为克服问题的手段。

通知规则 R4：若 公共处方 REC1 失效，且替代性处方 REC2 存在；则 通知协作组其它成员：REC1 失效，提供 REC2 作为替代性处方。

可以说，绝大部分现存的分布式人工智能系统都不提供关于合作协调本身的清晰表示。协调合作的基本原理隐含于协商协议和给个体智能体的合作行为特别设计的监控机制中，无法像联合负责模型那样获得因提供清晰的组织框架而带来的利益。传统知识系统采用的产生式规则可用于建立环境条件（如处方的执行成功结束或未如期完成）和相应动作（如通知协作组成员关于处方的执行成功结束或未如期完成）的直接关联（经验性关联），以支持合作协调的实现；但由于缺乏清晰的合作协调理论支持，这些浅层合作协调知识往往不全面，在不确定环境下易于陷入困境，表现出传统知识系统常有的脆弱性和不可靠性。

联合负责模型使这些条件一动作型的经验性关联知识具有严密而清晰的理论作为基础，而 Grate* 则提供了清晰的组织框架去支持这些知识的获取和应用，就像本节阐述的那样。这意味着智能体能够将它们的推理活动建立在原理性合作模型基础上，从而可更灵活和更有力地适用于各种复杂的应用域。

17.6.7 智能体间的通信与交流

信息和知识共享以及合作协调的需求，使得经由消息传递来实现通信成为智能体协同工作的重要基础。尽管先进的网络设备、网络操作系统和网络通信协议（如 TCP/IP）给智能体互操作提供了必要的手段，但智能体协同工作仍面临着一个主要障碍—缺乏语义互操作能力。智能体设计者观察应用领域的思想方法的不同和需解决的问题特征的不同，使参与协同工作的各个智能体往往是基于不同的本体，缺乏理解通信内容的公共基础，从而在互操作时易于产生语义失配问题。例如，两个智能体对于同一事物采用不同的术语表示，同一几何问题在两个智能体中分别以直角坐标和极坐标描述，等等。建立共享本体，并将遵从共享本体的通信内容翻译为与各智能体特有的本体相容的形式，是促进语义互操作的有效途径。此外，目前的网络通信协议太低级了，必须提供高级的互操作协议和提高通信效率的手段，才能有效地支持智能体协同工作的大规模应用。

美国的 SHADE (SHARED Dependency Engineering) 项目在综合和发展已有成果 KIF、Ontolingua 和 KQML 的基础上，曾提出了在知识级实现信息和知识共享的智能体通信机制。其中，KIF、Ontolingua 和 KQML 技术的综合形成能支持语义互操作的智能体通信语言 ACL；而引入称为“促进器”的智能体，则可促进智能体间的通信和协调。SHADE 通信机制的研究背景是支持工程设计环境下智能体的协同工作，并能推广应用于一般环境下的多智能体系统。

SHADE 通信机制已应用于工程设计领域开发的若干应用系统，如 PACT、MACE、Cosmos 和 VT 等。

目前已存在大量的工程设计辅助工具。当应用多智能体系统技术去集成这些工具，以支持并行工程和协同工作时，如何实现这些异质工具间的互操作，尤其是语义互操作，成为必须解决的紧迫问题。SHADE 项目从三个方面研制了支持智能体语义互操作的通信机制：信息和知识的共享，交互协议和通信促进。

1. 信息和知识的共享

SHADE 应用 Ontolingua 作为开发共享本体的工具，KIF 作为形式语言去定义本体和表示遵循本体的领域知识。由 Ontolingua 提供的翻译机制，则使 KIF 格式的信息和知识自动转变为各智能体内部采用的表示形式，以支持异质智能体间的信息和知识共享。

(1) 关于共享本体的概念

为实现信息和知识共享，必须定义协作成员都认同的一个词汇表，以使协作成员都理解交换的信息和信息间的从属关系。这个词汇表以及词语合成方法（即应用词汇去描述事物的方法）即构成协作成员共享的本体——对共享的概念化理解的清晰说明。需要共享本体是工程设计领域存在多种表述方法的直接结果。由于存在许多关于设计的不同观点（功能、性能、制造），每个都采用不相同的语言。然而，各种观点往往重叠，只要将重叠部分表示为共享本体，就能使协作成员正确理解所交换信息的语义。

目前，本体已成为人工智能研究的一部分。本体这一术语借鉴于哲学。在哲学中，本体原本是对“存在”的系统化阐述。而从人工智能的角度，本体被视为设计智能系统时建立的世界观，清晰表示的本体则等价于对应用领域概念化的说明。可以说，任何软件系统（包括智能系统）都基于某种本体，只不过传统软件系统（包括智能系统）的本体往往是隐含的，即仅存在于设计者头脑中或者说隐含于程序中。

从知识表示的角度，清晰表示的本体都有一个论域，即拟定表示和处理的对象集。对象（概念和个体）以及它们之间可描述的关系构成表示应用领域知识的词汇表。如此，我们可通过定义由基本术语构成的词汇表以及用基本术语去合成词汇外延（即领域知识）的法则来建立关于智能系统的本体。Ontolingua 已经提供了形式化定义本体的方法。以这种方法定义的本体都有一个词汇表，其中每个术语都与论域中一个对象的名字关联，具有一个供人阅读的解释术语意义的文字描述，以及供智能体处理的公理化形式定义。

以形式化语言来表示共享本体可提供充分的表示能力，使应用不同表述方法开展设计活动的智能体能在参与协同工作时沟通彼此的观点。这并非指共享本体必须包容由这些智能体采用的所有表述方法。协作成员共同感兴趣的观点往往只占全部的较小部分，大部分观点仅用于指导各智能体自身的行为，与其它智能体无关。不过本体会进化的，一旦基本词汇定义完成，更多的共享观点会随协同工作的深入开展而加进共享本体。这就要求表示语言能支持不同程度的知识共享，并允许用其表示的共享本体能逐步扩展。

人、组织、软件系统都必须通信。但由于不同的背景上下文和需求，对于同样的主题，存在很不相同的观点和设想，概念、结构和方法失配，但又有许多部分重叠。缺乏共享的理解导致人、组织、软件系统间沟通的困难。不同的建模方法、风范、语言和软件工具不仅严重阻碍了软件系统间的互操作，也导致软件设计中的许多重复工作。清晰地表述软件系统的本体，可以促进理解的沟通，进而促进软件的重用和共享、互操作以及可靠性。

我们说一个智能体遵从某共享本体，是指其可观察到的动作与共享本体的定义（约定）

相一致。这种关于本体约定的思想建立在知识级观点上，因为本体约定独立于各智能体内部的符号级表示和推理机制。智能体拥有的知识反映在它们向其它智能体发送的消息中，至于这些知识在智能体中的表示、存贮和推理，并不是其它智能体关心的事情。

实用上，一个共享本体定义了智能体在相互交换询问和断言时使用的公共词汇，遵从这些本体约定可使通信以紧凑一致的方式进行。应注意，共享词汇与共享知识库有本质区别。共享词汇的智能体不能直接取用其它智能体拥有的内部知识，只能请求其它智能体提供所需知识。但正因为如此，我们就避免了建立和维护大型知识库的一切烦恼，而把经由通信共享知识视为共享一个虚拟的公用知识库。

(2) 知识交换格式 KIF

KIF 作为异质软件（不同人在不同时间用不同语言编写的程序）间交换知识时采用的形式语言，是一阶谓词演算的一个扩展文本。尽管 KIF 可以作为与用户交互的表示语言，但提倡异质软件以各自原有的方式与用户交互；KIF 也可以作为智能软件内部的知识表示语言，但提倡这些软件应用自己原有的方式表示知识，而仅在相互交换知识时，才把知识转变为 KIF 形式。由于 KIF 追求逻辑表示的充分性，所以并不是能高效支持推理的知识表示语言；但不依赖于特别解释器的可理解性，使其十分适合于作为通信内容的表示语言。KIF 的基本特点如下：① 有陈述性语义。KIF 中表达式的语义不需依赖于操作表达式的解释器就可理解，这不同于应用特别解释器的其它表示语言（如 Prolog）。② 表示的充分性。KIF 允许表示谓词演算中任意的句子，不像大多数表示语言那样对句子的表示作某些限制（如 Prolog 限制句子只能表示为 Horn 子句）。③ 支持元知识表示。即可表示关于知识表示的知识。KIF 允许显式表示所有的知识表示决策，也允许引入新的知识表示结构而不改变语言本身。

KIF 的基本结构类似于 LISP 语言的表达式，即一个表达式可以是单词或表达式的有穷序列（括在园括号中）。单词分为三大类：变量、常量和操作符。表达式分为四种类型：术语、句子、规则和定义。应用这些基本结构，KIF 不仅可以表示一阶谓词演算中的任意表达式，还可定义类似于 LISP 语言的可执行程序。

KIF 有它自己的本体—按照对象、函数和关系等术语对世界作概念化说明。所有假设（可能）存在于世界的对象构成论域。对象可以是抽象的概念或特定的个体，可以是基本单元或合成体。尽管 KIF 的用户可以定义各不相同的论域，但基本对象应包含于每个论域，如 KIF 的关键词、数、列表和集合等。当然，用户可以加任何必要的非基本对象到论域中。函数和关系则描述论域中对象之间的关联和作用，并且都定义为列表的集合。实际上，函数仅是关系的特殊情况，函数将其参数（对象的有穷序列）关联到单一对象（值），而关系则无此限制。

鉴于 KIF 有充分的逻辑表达能力，Ontolingua 选其作为共享本体的基础语言。用于编写本体中的公理化定义。例如，可以定义公理“All writers are misunderstood by some reader”如下：

```
(forall ?W
  (=) (writer ?W)
      (exists (?R ?D)
        (and (reader ?R)
              (document ?D)
              (read ?R ?D))
```

(not (understands ?R ?D))))))

这个定义是自明的，只要学过谓词逻辑，任何人不难理解这个公理化定义。

(3) 共享本体开发工具 Ontolingua

由于不同的智能体往往有不同的本体，并以内部的表示语言描述，尤其是在智能体建立于“遗产”软件基础上的情况下，这就需要开发参与协同工作的智能体共享的本体，并提供机制将共享本体翻译为与这些智能体的本地本体相容的并以本地表示语言描述的形式。

Ontolingua 在 KIF 语言的基础上提供了开发共享本体和对其作翻译的机制，其将类 (classes)、关系 (relations)、函数 (functions)、实例 (instances) 和公理 (axioms) 作为定义共享本体的基本元素，并能把这些元素翻译为几个常用的知识表示系统可接收的形式。

Ontolingua 能分析任何 KIF 语言并识别许多关于表示的习惯用法，从而可确保在其支持下开发的共享本体能翻译为智能体采用的内部知识表示形式。然而，由于现行的大多数知识表示系统都强加了语法和语义限制于一阶谓词逻辑，并不是所有 KIF 语句都能翻译为目标表示形式。当出现共享本体的某个描述语句无法翻译为目标表示形式时，Ontolingua 会通知用户，但不中止翻译。

Ontolingua 识别的知识表示习惯用法定义于一个称为 Frame Ontology 的本体中。该本体以陈述形式说明面向对象的基本表示元素，如类、实例和槽约束等。这个框架本体和 KIF 自身的本体、工程数学本体、配置本体以及工作分配本体一起构成了支持分布计算环境下工程领域内协同工作的共享本体库。

尽管教科书和其它技术文献中已存在一致的表示法去描述数学表达式，但这些表示法并未清晰地表示设计者想要表达的许多语义，这使数学表达式隐含了让其它人理解时所需的许多细节。例如，表达“ $f = kx + c$ ”就未清晰地指出：哪些符号是变量，哪些是常量；这些符号指示什么物理量（如力、长度等）；它们是实数、矢量，还是高阶张量；这些量是静态值、时间的函数，还是时间和空间的函数；这些量使用什么测量单元等。读者必须具有背景知识和了解此表达式的上下文，才能正确理解此公式。

工程数学本体为理解数学表达式建立了概念化的基础，提供了独立于上下文和读者背景知识的概念定义，使表达式中的每个符号均有清晰的语义。工程数学本体是应用 Ontolingua 开发的本体范例，其建立在抽象代数和测量理论的基础上，而上述框架本体则提供了关于对象和关系的表示理论。

工程数学本体实际上是若干本体构成的层次体系，物理量本体成为其核心，在这个层次体系中，下层本体均可视为上层本体的外延。例如，标量以及标准单位和维是从物理量本体外延出来的两个本体，标量本体又外延出矢量和一元函数本体。层次体系有利于工程数学本体的渐进开发，并允许实用时仅承诺遵从整个本体的一个子集。例如，可以只遵从标量本体而不包括矢量本体。

设计物理量本体的目的在于定义量、测量单元和物理维之间的关系。目前已定义了物理量的分类，对于这些类的代数操作，以及这些类之间的其它函数和关系。关于单位和维的理论支持不同单位之间的换算，而抽象代数定义的操作则适合于写约束方程。

只要将工程数学本体作为参与协同工作的智能体共享的本体，就能确保智能体相互理解交互的有关数量及数学表达式的信息。例如，下面是一智能体发送给另一智能体的遵从工程数学本体的信息：

(physical-dimension length)
(unit-of-measure inch)
(= (q. dimension inch) length)
(quality (diameter shaft-a))
(= (diameter shaft-a) (* 3.6 inch))
(= (q. magnitude (diameter shaft-a) feet) 0.3)

鉴于关系 physical-dimension、unit-of-measure 和 quality 以及函数 dimension、magnitude 和 diameter 都已定义于工程数学本体，所以接收智能体能无二意性地理解：长度是物理维，inch（英寸）是测量单位，且其维是长度（即长度的测量），杆 α (shaft-a) 的直径是一个量且等于 3.6 英寸（既然量度单位是 inch，则该量的维是长度），杆 α 的直径以 feet 为单位量度时长 0.3。

2 智能体交互协议

智能体的互操作在许多方面都超越了数据库管理中简单的询问-应答范式。除了作简单的询问和断言外，智能体必须能进行持久的询问，广告它们的能力，说明所发消息的结构（单一解答、解答流、所有解答等），说明所发消息的路由等。为此，智能体需要交互协议。

显然，若不致力于理解和标准化智能体交互协议，必将出现大量互不相容的特别通信语言。为此，SHADE 采用 KQML 作为消息定义语言（称为通信的外部语言）去包装 KIF 形式的通信内容（KIF 称为通信的内部语言），并定义处理各种类型消息的交互协议。KQML 与 KIF 联合构成规范化 ACL，允许遵循共享本体编码的信息和知识能在智能体间共享和交换。ACL 正在成为智能体的准通信标准。

想象一下由智能体协同工作支持的分布式设计环境。若一个工程设计任务由一组设计人员分担，每个设计人员都有一个智能体作为智能助手。可以认为，各智能体拥有的知识（无论是显式或隐式表示）联合起来就构成一个虚拟知识库，而基于 ACL 的消息传送则使虚拟知识库中的知识得以共享。这种知识共享属于知识级，因为包含于消息的知识具有独立于各智能体表示和推理机制的语义。例如，一个智能体可发送消息去广告其可提供信息服务的潜力或能力；另一个智能体或许发送消息去表明对什么样的信息感兴趣，从而导致这两种消息的匹配。显然，广告能力和表明兴趣促进了信息和知识的共享。

为支持知识级信息和知识的传送，SHADE 项目应用 KQML 作为智能体间的交互协议。KQML 是一个点对点消息传送机制，而基于共享本体和 KIF 语言表示的通信内容则包装于 KQML 定义的各种类型消息中。严格地讲，KQML 注重于消息传送级，并不在意通信内容以什么方式表示。但 KQML 设计的目的是提供包装通信内容的外部语言，其与基于共享本体和 KIF 的内部语言（表示通信内容）一起构成支持智能体间知识级通信的规范化语言 ACL。

KQML 工作的语义模型是提供一个简单的统一上下文，使各“智能体”能彼此观察对方的能力和行为。各“智能体”不必都是知识系统，它们可以是简单的数据库应用系统或应用特别数据结构的程序。但只要能以 ACL 来表示和处理交换的信息，就可以把这一“智能体”视为知识系统，即每个“智能体”均管理一个虚拟知识库。在这个意义上，一个智能体能于知识级通过 ACL 去查询、增加和删除其它智能体虚拟知识库中的信息和知识，而不管信息和知识实际上以什么形式表示和处理。

KQML 与现存的其它通信协议的本质区别在于，它不仅仅负责传递消息本身，而且能通过定义丰富的消息类型及它们的语义去提出接收者应如何处理消息内容和如何应答的期望，

从而促进智能体间通信的协调。消息的类型以称为 performative 的关键字指示，所以每种类型的消息也称为 performative 消息。例如 tell 是一种 performative（类型），则相应的消息称为 tell 消息。

KQML 消息有如下格式：

```
(⟨performative⟩{:⟨参数关键字⟩⟨单词⟩|⟨表达式⟩})* )
```

常用的参数关键字有：

content--消息拟传递的内容信息，其格式取决于所用的表示语言。

language--关于内容信息的表示语言。

ontology--内容信息遵循的个体。

reply-with--指示消息发送者期望的应答标记。

in-reply-to--在应答消息中指示被期望的应答标记。

sender--消息发送者。

receiver--消息接收者。

to--在内容信息的转发中指示最终接收者。

from--在内容信息的转发中指示信息源（原始发送者）。

消息格式中的填充内容必须遵从格式中指定的表示语言。KQML 语言的消息类型几经扩充，以适应于日趋复杂的通信需求。目前 KQML 已可支持 40 种以上的消息类型，除一般通信中常用的 ask、tell、reply 外，还特别增加了信息流（stream-one、stream-all）、网络（forward、register、pipe）、兴趣和能力陈述（advertise、subscribe、monitor）以及代理（broker、recruit）等新的类型。这些消息类型促进了信息流的结构化，使得智能体不必直接向最终目的地发送每一个消息，从而提高了通信的有效性。

作为示范例，下面观察两个智能体（A 和 B）间的通信。A 首先广告（advertise）其意欲提供关于发动机[motor1]的信息：

```
(advertise:language KQML :ontology k10
  :content (subscribe: language KQML :ontology k10
    :content (stream-about:language KIF
      :ontology motors
      :content motor1) ) )
```

作为对该广告의应答，B 发送以下消息给 A：

```
(subscribe:reply-with s1 :language KQML :ontology k10
  :content (stream-about :language KIF
    :ontology motors
    :content motor1) )
```

然后，A 随时间的推移发出一系列消息给 B，包括一次对首发消息内容的取消要求（untell）：

```
(tell :language KIF :ontology motors :in-reply-to s1
  :content (= (val (torque motor1) (sim-time 5))
    (scalar 12 kgf) ) )
(tell :language KIF :ontology structures
  :in-trply-to s1
  :content (fastens frame 12 motor1) )
```

```
(untell :language KIF :ontology motors :in-reply-to s1
      :content (= (val (torque motor1) (sim-time 5))
                (scalar 12 kgf)))
(tell :language KIF ontology motors :in-reply-to s1
     :content (= (val (torque motor1) (sim-time 5))
                (scalar 13 kgf)))
```

SHADE 已提供了应用编程接口，使智能体能在 LISP 和 C++ 编程环境下交换 KQML 消息。SHADE 也提供了智能体间经由 E-mail 交换 KQML 消息的工具集 ServiceMail。

3. 通信促进服务

在分布计算环境下，当智能体的数量增加时，势必导致通信的复杂度增加。为降低通信复杂度和减轻通信负载，各种通信促进服务纷纷出现，如路由服务、演绎数据库服务和翻译服务。演绎数据库服务可以通过设计信息代理来提供，翻译服务则可通过基于共享本体的“翻译器”或其它因特定需要而设计的翻译智能体来提供。类似地，路由服务由被称为“促进器”的智能体来提供。

促进器有路由促进器 (Routing Facilitator) 和匹配器 (Matchmaker) 两种。路由促进器通过一个智能体在因特网上公布的兴趣说明来为消息决定路由。当关于智能体的因特网很大并且是动态地变化时，智能体间能力和需求的相互预先了解将变得不可能也不实用。基于内容的路由可以为这种环境下、相互间不知道对方是否存在的智能体建立通信和共享信息。为实现基于内容的路由，这个智能体必须能清晰地说明它们的兴趣。简单情况下，这种说明可以表示为匹配模式，以供合一 (unification) 检查，更有用的是表示为随意复杂的一阶逻辑语句，以便遵循某种共享的本体评价兴趣能否满足。当相关的信息由某些其它的智能体发布时，路由促进器就能根据合一检查或内容匹配作判定，将满足需要的信息转发给需求者。基于 KQML 和 KIF 的 ACL 以及共享本体的制定，为路由促进器的工作提供了支持。路由促进器的核心部件就是作合一检查和匹配评价的推理机。

尽管基于内容的信息路由提供了重要的通信促进服务，但存在着明显的不足。那就是若所有智能体都自愿提供所有信息，将会导致无人需求的消息堆积到路由促进器，而且大量的消息转发也会降低通信效率。克服此不足的办法是，智能体不要主动提供信息（除非其预言肯定被需要），转而仅说明提供信息的能力。路由促进器将这种能力与信息消费者的需求兴趣作比较，并把匹配的消费者介绍给信息生产者，然后由生产者直接发送信息到消费者。提供这种服务的路由促进器称为匹配器。

匹配器的工作建立在信息生产者广告生产能力的基础上。广告也包括一种匹配模式，作为对可提供信息的轮廓性刻画，并能说明信息的提供方式。例如下面的两个广告指出了提供信息的两种不同方式。

```
广告 (advertise
      :content (ask-about
                :content (subcomponent ?x ?y)))
```

指出其能提供有关产品“?x”的部件“?y”的所有信息。

```
而广告 (advertise
        :content (subscribe
```

```
:content (stream-all
          :content (subcomponent ?x ?y) ) ) )
```

则指出其能将有关“?x”的部件“?y”信息的变化持续地提供给需求者（即只要“?y”的信息有变化，就主动通知需求者）。

这种广告方式产生的重要意义在于，向信息消费者展示了能提供某方面全部知识（和信息）或知识变化的一个虚拟知识库的存在，从而可推动知识级信息和知识的共享。广告能力使信息生产者避免主动发送可能无消费者接收的信息，并使匹配器能为智能体间的通信提供灵活有效的交互渠道。KQML 提供了三种类型的消息来支持匹配器的这种工作：broker（代办）、recruit（征求）和 recommend（推荐）。这些请求获取信息的信息类型使匹配器在匹配的信息生产者和消费者之间扮演不同的中介作用：

（1）当接收到 broker 请求时，匹配器作为一个完全的信息代办，将[SA1]的信息需求转发给[SA2]（[SA2]广告的能力匹配该信息需求），然后[SA2]将生产的信息经匹配器转发给 SA1。

（2）当接收到 recruit 请求时，匹配器仅将[SA1]的信息需求转告[SA2]，然后由[SA2]把生产的信息直接发给[SA1]。

（3）当接收到 recommend 请求时，匹配器仅简单地把有能力提供所需信息的[SA2]推荐给[SA1]，然后由[SA1]直接向[SA2]请求提供所需的信息。

SHADE 在应用 KQML 提供上述路由功能的同时，还增加了支持持久信息请求的功能。在这种情况下，匹配器不是一次性对请求的信息作匹配检查，而是能在以后的任何时候把新发现的广告有此能力的智能体介绍给信息请求者。另一个增加的功能是元级推理，即匹配器在接收到信息请求消息时，首先基于元规则审查请求是否遵从共享的本体，只有在审查通过的情况下才对该请求作匹配处理。

尽管 SHADE 的匹配器支持形式的逻辑语言和结构化的文字表示，但不支持自由的文字表示。支持自然语言的表示需要基于全文检索的匹配器。此外，基于内容的路由和匹配器的技术还远非成熟，所以在全面推广应用之前，尚需深入开展大量研究工作。

17.6.8 智能体理论研究的现实与未来

17.6.8.1 智能体理论研究的意义

作为对有头脑、有思维的人的一种模拟，智能体本质上是一类能提供智能化服务的自治系统。它能在自主的情况下，独立完成一定的任务。从模型的特征来看，智能体在完成任何一项工作时都是自治的，这是一个独立主体才具有的特征。而作为对人的模拟，智能体必须包括“大脑”和“身体”。在这里，所谓的“大脑”就是这个系统的意识区和无意识的结构或程序。意识区主要处理模糊问题。它包括记忆和编程功能。无意识程序主要处理精确问题。包括过滤信息、多任务功能和储存记忆。而所谓的“身体”则包括感知和行为执行机构等。

作为一种智能系统模型，智能体系统理论有多方面的创新。其在方法上的创新主要表现在：

① 由于智能体所具备的驻留性与自制性，它可以“存活”于一定的环境中，具有一定的“生命力”，并能摆脱传统方法的制约，在没有人类指令及其他智能体的干预下自主、持续运作。

② 它模拟了人类的信念、期望、意图等心智状态和规划、学习等心智活动品质，具有

传统人工智能无法具备的人类智能中至关重要的能动性。它不再像以往那样，总是被动地接受已经预设好的算法与指令的驱使，而是能够面向陌生的状态与不确定性，积极主动地进行认知与行动。

③ 通过构造混合智能体等方法，智能体理论将传统模式中各执一词的符号主义与行为主义等方法结合起来，能够发挥出仅靠各行其是的传统单一方法所不能产生的综合集成能力。

④ 传统人工智能在认识与分析问题时，总是忽视或回避现实中的人类、事物及环境之间的互动作用及其复杂性，并常常对这种复杂性加以一种十分消极和封闭的抽象与还原，将其削足适履地硬性纳入到一个或几个普适模式中。这种方法总是竭力在静态、呆板以及过度的简约性中“雕琢”智能。显然，它无法反映一个复杂的、异构的、开放的、无序的，特别是由无序演化出有序的真实智能世界。与此不同的是，智能体方法以其移动性、适应性、交互性及协作性直面现实中的动态性及复杂性，将主观意向的认知、表达及转换与客观世界的变化紧密结合在一起，认识到主客观世界的互动过程中各因素之间深层次的联系、作用及变化才是产生与发展智能的本质。并力求在对这些互动作用的动态性及复杂性的认识 and 把握中获得智能并增长智能，是一种复杂性中的认知与思维。

⑤ 智能体方法从整体出发，在对问题层次化、模块化分割解析的基础上，对子问题进行从局域到全域、从同层次到跨层次并行不断的分解、重构与整合，形成一种既有分析又有综合的多元整合求解机制，从而突破了旧有模式中只注重分析法的局限性。

⑥ 每个智能体都具有各自不同的知识背景(知识库)、心智品质、能力及个性，甚至具备诸如亲善性、真诚性等人类特征，而且它的这些能力与特征都不是被设定为一成不变的，它可以随着情况的变化而不断地进行能动的自我更新。它不是在一个一步到位的、静态的模式中模拟智能，而是在一系列过程性的、动态连续的进程中发展智能。它是用能动的多样性超越被动的一般性，并在多样性的能动中，通过不断学习、逐步深化认识的方式突破传统方法所执着的一般性。因而，智能体方法更切近于客观世界的真实情况及人类解决问题的实际方法，是智能理论研究上认识论与方法论上的进步。

⑦ 就其结构而言，智能体系统没有一个僵化、固定的边界，能允许其成员相对独立且自由地参与系统的联合，即它的结构可以根据不同的任务进行开放的组合，并且系统能够对此实时地自感知。它采用的是较为灵活松散而非传统方法中过于苛刻呆板的组织机制，不仅系统的外在环境在不断地变化而且其自身的组份也在实时动态的变化着。此外，在系统内部及外部之间始终存在着大量的信息和资源的非对称、非均衡的多渠道交换流动。就其功能而言，系统可将整合过的子问题分配给各组能力不同的智能体模块，通过其交互作用来解决，并最终达到其结构与功能的和谐与有序。它能把握住种种开放条件，不断地主动去获得新的知识与信息，进而不断地突破自己的界限（也正是传统人工智能的界限），并适时能动地调整其意图和策略以消解系统变化中的复杂性。在不断的自组织、自适应中提升智能并挑战新的开放性与不确定性。因此，智能体方法具有很强的系统论意义。

⑧ 人类社会不仅是一个开放的社会更是一个互动的社会，许多问题正是甚至只能靠社会互动所产生的整体智能来解决。因而当这类问题出现时，即便是再高明的个体也只会力不从心。传统模式却始终只是守望着个体智能的修修补补，一味地追求着个体能力的卓越，而排斥考虑人类智能中不可忽视的社会性因素，拒斥从社会互动的角度来提升智能。同时竭力追逐着个体智能高度的一致性与完备性，而十分缺乏人类智能中至关重要的容错能力。

更缺乏从自己和他人的错误中学习，从来自多方面信息的反馈与重估中提高，从彼此的商讨、辩论、借鉴特别是从知识的共享与才能的集成中获得集体智慧的能力。因此，不管这些单个个体的能力如何出色，因为其缺乏必要的社会交流与联系，而始终无法凝聚成一个协同的整体。这样，当系统运行时，若某个或某群成员由于某种原因在能力上无法胜任或某个行动方案出错，就可能导致整个任务的全面失败，一损而俱损。反之，当同样的情况发生时，多智能群体可以重新规划自己的意图，寻找另外的伙伴及方案或以更为积极的沟通与交流通过整体协作来解决这些问题。即，首先从微观到宏观上对任务按照各自不同的能力分配与协调（社会分工）、然后通过彼此之间不断地进行交互协商、协同行动（社会合作），并通过多种协调与协同方法的集成来“互通有无”进而“取长补短”，突破个体智能的片面性及局部性，从而发挥出集思广益的社会智能，将问题解决，并由此来“容忍”与克服行动中所不可避免的错误。这样，智能体社会通过对个体智能的集成和对社会智能的追求，通过合理的保持个体智能与社会智能的两极张力，在开放中学习、交流，在竞争中合作，在使用中进化，因而能逐步深化出更为全面的整体智能状态。因此，这种方法不仅超越了传统方法论的单一模式，突破了单个智能体解决问题时的诸多盲点，而且体现了开放社会中，个体通过社会合作而从其知识与能力的“残缺”中“涌现”出集成智能的新思路。那么，这种更接近于人类本身以社会智能解决问题的“返朴归真”方法，不仅确立人工智能研究中从个体智能到社会智能的范式转换，更会促使我们建构一个包括计算机、智能体、机器人与人类本身共同组成的复杂巨系统形式的广义智能体社会，这也将是人工智能研究的一条充满希望的道路。

在我们看到智能体研究的深远意义的同时，也要看到多智能体系统实现的艰难性。举例来说：

① 如果智能体仅以效用原则从严格自利的理性假设出发通过社会博弈来实现其整体效用，那么，根据同一假设，个体在互动中又必然会首先以优化它自己的效用来采取策略。这样，就可能会产生某些问题。首先，在传统方法中，出于个体智能的特征，其决策原则可以是单一的、完备的。但在一个开放的智能体社会中，它的决策原则更应根据多元准则而不仅仅是一个单一的效用原则来进行。在现实的博弈过程中，智能体如果按照完全理性来进行博弈，则往往达不到社会效益的最优。即便这种整体最优能够实现，其博弈次数也将是十分惊人的，而无法在一个有限与有效的次数内完成。同时，博弈者的数量也会受到相当的限制，无法达到智能体系统所预设的规模；而且，这种博弈始终要求是同一群博弈者，而不是多智能体系统中数量在不断变化的开放的博弈群体；同时，它还会出现许多个均衡，而无法实时地提供惟一的最优解。尤其需要注意的是，智能体系统的博弈规则和策略变量是内生的、历史的、演进的，而不是外生的、预设的，静态的（抑制智能体的策略能动性），在现有的理论框架中是不清晰的，当智能体的委托-代理问题大量出现，从而造成很难准确地界定合作中的冲突性和博弈类型等问题时，就会使得博弈论方法的“刚性框架”和智能体理论灵活的“柔性要求”一再发生矛盾，很难兼顾。

② 现实世界中的协商大多是在动态的、非对称、非完全信息下进行的，因而对任何个体来说，其知识都是一种局域性知识，而社会智能的实现却总是有赖于社会互动中知识与信息的共享及能力的协调，那么，由何种（合作性）博弈才能在诸多不对称条件下实现由个体局域知识到社会整体的知识共享（即共同知识，在这里，我们对共同知识仅作狭义上的直观理解）一进而能够克服一直困扰着诸多研究者的“共同知识”难题并最终获得最优

解？需要研究。此外，社会中的个体理性大多是以综合其自身微观效用和社会宏观效用的复杂的“折衷”理性来进行决策，而实际博弈中的个体理性要比目前的理论描述复杂微妙得多，且完全无法由目前狭隘的“理性”所能够包容和处理，那么，从自利理性出发又达不到共同知识的群体博弈又怎样才能达到整个社会效益的完美均衡而不至于陷入到类似于囚徒困境的结局中呢？依照古典经济学观点，在没有全局协调者的情况下，即使每个主体都严格按照自利的理性行动，也有可能出现如亚当斯密所谓的“看不见的手”的驱使，达到个人理性与集体理性的并行不悖。而按照当代集体选择理论，个体理性并非是实现集体理性的充分条件。因此，在描述这种大规模频繁合作的社会行为时，除了其巨大的社会交易成本的制约外，现成的博弈论特别是其理性基础有着十分明显的不足。这样，除非我们对理性进行更为深入的探究并重新开发出更为有力的新理论，否则，将给智能体理论带来某些相当难以克服的问题。当我们把逻辑理性和经济理性联系起来看时，使用有符号推理的逻辑理性无法使决策效用最优化，而使用数值计算的博弈理性又忽视了理性的逻辑推理，那么如何较好地协调这两种机制，即如何协调好逻辑推理的必然性与决策中的不确定性，显然又是一个十分艰难的问题。

③ 有人想从社会学的角度通过建立全面而严格的智能体社会规范来避免种种破坏性的冲突或二难困境，但通常个体遵守或故意违反规范不仅有其个体因素更有其内在的社会原因。而且，过于严格和精致的社会规范又会反过来制约到智能体系统的灵活性和适应性，甚至形成一个个新的始料未及的窘境。另外，现实社会中存在的诸多规范（如法律等）都是强制性的而并非单纯靠协商或承诺达成的，因此，智能体社会仅靠协商与合作就可能无法维持其良性运作。同时，在一个开放社会中，各种社会规范总是随着新的形势发展及其结构的变化而不断转化的。因此，在开放的社会特征下，当我们赋予智能体以诸如友善性、诚实性等特性时，就会发现，我们只能在一个封闭的世界中才能使所有的智能体都是友善和诚实的，在一个开放的社会中很难保证所有的智能体都能这样。因为既然是一个开放的社会就不可能排除自私的，彼此利益和目标冲突，甚至于怀有恶意的主体的存在。因而智能体的友善性和诚实性只能针对其所具体服务的主体，这样，它就无法完全开放于任何其他人的智能体或人类主体，那么，这种对于社会合作极为重要的诚实性与友善性其意义及可行性就更为可疑了。因此，在智能体目前的规范约束下，智能体几乎就不可能做到既诚实又友善地去完成代理任务。

④ 与人类的个体交往相比，智能体为了实现其大量的、不间断的社会协同，很多时候智能体之间及其与人类主体的通讯交往会特别的频繁，其行为的非线性更为明显，很可能具有某种比人类社会自然的演化机制更为复杂并夹杂着更为剧烈突变的深化机制。如何从社会进化的角度研究这种新的深化机制，也是我们需加以高度重视的。我们注意到，在现实的人类社会中，即便人人都遵守规范，也不能保证没有冲突发生；因为社会本身就是一个交织着各种复杂欲望和矛盾，且不断动态变化和发展的社会。某一时刻所制定的规范可能已不再适用于发生了改变的具体情况。这样，人类越是严格地按自己理想中的纯粹理性来建构一个人工智能体社会，就越会遭遇到连自己都已无法完全把握的复杂多变的现实环境，从而不断地产生新的二难困境。

总之，智能体理论的研究是一项极富有挑战性的工作。对它的研究是最终取得人工智能研究的巨大成功，还是人类只是把自身不能实现的理想和梦想隐隐约约地寄托在了一个纯净的理性“智能体乌托邦”上？这些都不是我们马上就能够回答的问题。对智能体理论及

人工智能（实际上是对我们人类本身）的透彻认识和研究，我们还有一段非常漫长而艰巨的路要走。

17.6.8.2 智能体系统技术—现实与未来的几点应用

无论是作为一种软件开发范式、信息处理模式还是智能模拟理论，多智能体理论都有着广泛的应用。目前，多智能体系统技术的应用更多地是集中于软件开发和智能和信息处理方面。其现实的和未来的应用可表现在诸多领域，一些比较典型的应用如下：

(1) **网络信息服务** 网络信息服务是便利广大用户利用网络信息的一种考虑。对于信息内容已经相当丰富的现代社会来说，进一步提高网络信息服务的质量，改变目前信息服务中存在的“信息过载”和“资源迷向”的状况，是信息服务业所面临的最紧迫的任务。智能体技术正好可以适应这方面的需要。具体地说，用于信息服务的智能体技术至少可以完成以下功能：① 导航，即告诉用户所需要的资源在哪里；② 解惑，即根据网上资源回答用户关于特定主题的问题；③ 过滤，即按照用户指定的条件，从流向用户的大量信息中筛选符合条件的信息，并以不同级别(全文、详细摘要、简单摘要、标题)呈现给用户；④ 整理，即为用户把已经下载的资源进行分门别类的组织；⑤ 发现，即从大量的公共原始数据中筛选和提炼有价值的信息，向有关用户发布。这些都是使信息服务走向个性化主动服务不可缺少的功能。目前，在此方面已经有了一些能够使用的系统，但智能化的程度还不够，亟待尽快发展更先进的人工智能技术予以解决。

首先是智能化的界面。未来的网络和人-机界面将是智能的，可自动地调节以适合于用户的使用模式。这种智能的界面能帮助人们发现其感兴趣的东西和做想要做的事情，人们只需以他们感觉到自然的方式提出要求，智能体作为智能人机界面或用户的个人助手即可完成他们想得到的服务。为了使作为智能界面的智能体具有足够的智能，以下几个特点应该得到满足：① 集成性和表达性—集成性指不应迫使用户去记忆特定服务的细节以及为有效应用服务所应提供的信息，而是反过来让人机接口以可理解和一致的显示风格去适合用户的应用习惯和知识背景。表达性则应使用户在形成随意的问题和请求时感觉到十分便利，不受严格菜单的限制，也不必学习人工的询问语言，并可自由地选择表示问题和请求的方式。用户只需以熟悉和灵活的方式与他们的智能代理通过自然语言、线框图或其它方便的形式会话即可。尽管基于视窗的图形人机界面已普及应用，但离人与人之间的自然交互方式还相差太远。所以提高接口的感知能力是重要的，它可使人机间能通过文字、图形、动画、声音以及这些方式的协调组合进行交互。近年来计算机感知能力的研究已有了长足的进展，但技术上的问题仍然存在，尤其是脆弱性和不可靠性尚未解决。表达性是与对已获取数据的自动解释能力紧密相关的，这就要求智能体不仅能获取、处理和产生数据，而且能自动作归纳总结，并以最适合于表达数据含意的方式与用户交互，使表达的方式与用户的知识背景、理解能力和兴趣相适应。② 面向目标和合作性—用户只需以其认为的自然方式说明要计算机提供的信息和服务目标，至于如何和何时达到目标则由智能体自行处理，并在不受用户监护的情况下执行。为避免因用户在说明目标时的不完全和不正确性而引起的问题，用于智能界面的智能体应与用户合作建立可为系统接受的询问。为此，智能体必须能理解大范围内用户对于各种可能目标的说明，与信息源的拥有者协商许可和代价事宜，并最终组合来自不同信息源的结果。智能规划技术为迎合这一挑战提供了坚实的基础，它可使系统能应用关于用户目标的逻辑编码和描述可用信息源、数据库、设施、协议及软件命令的纲要库，去建立、解释和执行能完成期望目标的计划。智能规划器不依赖于事先决定的控制流，而是自动和动态地产生和执行完成用户目标的计划。先进的规划技术支持人和计算机的联合规划，以完成无论是人或计算机

均不能单独达到的目标。作为一个附加的利益，自动规划可使人从繁琐的细节化规划任务中解脱出来。③ 定制和适应性——作为用户个人助手的接口型智能体应按任务的需求定制，并能通过接受用户的直接询问和从与用户交互的经历中学习来适应不同的用户。许多用户都喜欢定制人机界面的感观风格，但仍需接口型智能体来决定人机交互的细节和处理可能的冲突。智能体不仅应适应用户的经历、兴趣以及它们的变化，还必须适应环境的进化和变迁。机器学习技术对于建立这种自适应性是关键。学习程序能不令人注意地观察用户与计算机的交互行为，然后作适当泛化以确定界面应如何剪裁以适合用户。作为这种学习程序的应用，智能调度可以学到用户喜爱的会谈时间和地点，通信辅助器能从用户的行为学会如何优化电子邮件消息。进一步的应用将是让接口型智能体能自动提供用户感兴趣的信息，掌握用户的需求，代替用户进行事务处理和向用户提出某些建议等。

其次是信息和管理支持服务。考虑到网络空前的规模、范围和资源的异质性，以及为实现对资源（信息和服务）的充分存取而需要的互操作性，网络系统需要一套高级的信息服务支持系统。这些服务至少应包括：系统管理、数据和知识管理、集成和翻译、知识发现。① 随着计算机和网络系统的资源量越来越大，系统管理的整个趋势是正在朝着“傻瓜式”的方向发展。很多原来由系统管理员干的事情，现在已由计算机代管了。在系统自身复杂性不断提高的情况下，人驾驭系统资源的能力反而在不断降低，代管系统可以有效提高系统的管理性能。在网络管理方面，随着 Client/Server 计算成为主流，可完成固定功能的智能体的存在和应用已经成为事实。分布在不同设备上“监视”设备状况，并随时向主机报告数据的智能体，已经是当今网管系统中很平常的一部分了。但是，这些从事部分系统/网络管理的智能体目前还不能称为真正的智能体，它们的智能化程度仍然有待提高。② 数据和知识的管理可为“寻找与用户任务或目标相关的信息，为可提供的信息寻找感兴趣的用户”提供服务。基于智能体技术的导航工具可以提供定制的漫游指导，以满足用户个人的特别兴趣和主动搜索最佳路由。由此，作为信息消费者的用户就能在异质、分布数据的巨大海洋中快速定位所需的信息。这里的挑战性技术是要提供有效且效率高的搜索方法，以便仅在适当的地方作漫游导航和询问处理；能检索各种异质的资源，并解释和汇总结果。然而，异质性和可伸缩性使定位和信息传送面临困难的挑战。首先，分布于网络的信息是以种种不同的形式存贮，从视频图象、声频、各种语言的文字，到数据库关系和数学方程。由于存在多种分类方式，索引信息是困难的。多种人工智能技术，包括推理和分类方法、自然语言理解、视觉分析算法、似然推理等可用于支持各种异质信息的索引和检查。可伸缩性指对于各种不同的信息和服务需求，响应时间仅是搜索量的线性函数。中央控制型索引和检索机制往往会导致指数级增长的响应时间，建立多个相互竞争又能在必要时相互合作的信息型智能体可提供更快的响应时间、更好的对于网络环境进化的适应性。提供被检索内容的语义信息将可进一步提高信息型智能体的效用，知识表示和本体开发技术可用于促进这种语义信息的抽取和表示。③ 集成和翻译服务。数据的异质性使得即使在内容上类似的信息也往往会以很不相同的方式表示和操作。尽管人们在数据格式和互操作的标准化方面付诸了许多努力（如产品数据表达和交换标准 STEP 及其描述语言 EXPRESS、分布对象互操作规范 CORBA 和 COM 等），但其抽象级别太低，远远不能满足网络用户的需求。应代之以基于标准的数据翻译。知识表示、本体开发、机器翻译、协调和合作等人工智能技术可以提供有力的支持。这些信息往往是以不相同的形式和操作方式提供的。④ 知识发现。快速增长的数据和信息已经建立起一个需求机遇——从数据库和维持它们的一致性过程抽取知识。网络所包容的信息海洋使得基于手工方式的知识

抽取几乎无法实现，而必须依赖自动化工具去扫描数据库，核对一致性，产生总结，支持逻辑推理和诱导，促进浏览、问题回答、解释和辨证，并发现数据之间隐含的联接。一致性检验的问题使得知识发现变得十分困难，因为一致性检验的时间复杂度是数据库容量的指数级，而且当分布的信息源（数据库）处于不断更新的状态时，实际上也无法获得一致性。知识发现服务能追踪新信息源的建立和已有信息源的更新，并经由索引横穿相关的论题去发现隐含的关系和产生总结。启发式知识能用于帮助集中注意力于最有可能的不一致问题，并建议解决方式。现今的知识发现应用已在多个领域开发，知识表示、似然推理、本体、机器学习和机器感知的技术研究将推动知识发现应用的开拓。

(2) 复杂系统软件的开发和支持环境 现行的软件开发工具和语言使软件系统的结构冗余且易于出错，难以满足开发大型和高性能的应用系统的需求。因此，应为软件系统开发提供方法论的支持，并由此建立强大的工具集和开发环境。智能体技术至少可在以下方面提供支持：① 快速原型法。其组合面向说明和精化设计的服务与已开发软件和世界知识的模块库，能以较低的代价快速建立最终软件产品的可执行模型。设计者只需制定一个试探性的、可能是部分的需求说明，就可建立原型，然后再反复测试和修改，直到获得一个令人满意的需求说明（可作为最终软件产品的需求说明）。由于需求说明要不断精化，必须提供形式语言（如逻辑型语言）去描述需求，以便修改，并应具有自动生成文档、提供联机帮助信息的能力。知识表示和本体技术可以用于定义软件设计的高级概念和支持模块化、层次结构和多继承的实现。定理证明和逻辑推理算法则可用于探测需求说明中可能出现的二意性、不完全性和不一致性，以支持对需求说明的测试和有效性验证。最后，可以构造具有设计指导、决策解释和设计追踪等功能的智能体去帮助设计者开展工作，以减轻他们的负担。② 智能化的项目辅助管理。用以辅助软件系统开发项目中的合作开发、分布的决策制定和资源可用性检验。在非期望条件出现时，辅助项目负责人作重规划。高性能应用系统往往是复杂的大规模软件开发项目。尽管已有许多项目管理系统可以应用，但仍难以适应越来越庞大的系统开发要求，需要性能更高级的智能化项目管理。首先，为支持软件项目开发中的群组合作，应给每个参与者提供信息共享、虚拟配置和任务协调设施，使按时间、地点和知识背景分布的人能协同工作。建立这样的合作支持环境需要开发能用于建模过程和规划的工具，以协调项目和管理 workflows 的约束。促进软件开发人员间的通信也十分重要，可以提供多种智能体去代办和筛选开发人员间的信息传送，并快速查询各种数据库。此外，建模合作工作、联合规划和关系协调十分重要，电子会议和各种翻译工具则支持开发人员间的有效协商。③ 提供分布的仿真和综合环境，用以支持教育、训练、产品的仿真原型等应用。为评价原型软件系统的功能和性能，必须能近似地仿真软件系统将来工作的环境。而且仿真在复杂系统（如航空交通控制、电话交换网、电力网、分布的传感系统、工厂资源配置、远程导航等）管理、训练（如教育、飞行训练、虚拟手术等）、规划和优化、远程诊断、感知、合作交互等领域也十分有用。这样的仿真环境应能综合真实和虚拟的对象，组合视觉和计算的训练法。构造这种综合环境、仿真人和生物的真实感三维运动以及描述关于世界的大量知识，是系统仿真面临的三个困难问题，也是智能技术的机遇。

(3) 虚拟现实与远程展示 在许多情况下，**虚拟现实**能给用户以能直接操作三维物理现实的感觉。例如，先进的 CAD 系统可以给小汽车设计者观察和操作尚处于设计蓝图阶段新产品的虚拟环境；设计者能围着虚拟车绕行，走进车中，并沿着试车道驾驶它。虚拟环境对教育、训练和娱乐业的作用更为明显。远程教育是促进教育机会平等的重要手段。在网络环境下，可以调动多种教学手段，包括讲解、演示、练习、实验和考试等。其中，练习和实验环节是智能体可以大有作为的地方。智能体可以作为虚拟的教师、虚拟的学习伙伴、虚拟的实验室设备、虚拟的图书馆管理员等出现在远程教育系统中，增加教学内容的趣味性和人性化色

彩,改善教学效果。例如,在一个虚拟东京的虚拟日本商店中,可以让学生有机会练习实地的日语会话,而不必去日本旅行。远程展示则允许人们操作危险的环境并安全地执行诸如海底勘探或核反应器维修等任务。目前,虚拟环境仅能提供有限程度的现实近似,大规模的应用还需要硬件和软件技术的进一步提高。智能技术对虚拟现实能提供有力的支持,尤其是智能体技术、口语理解、图象处理。鉴于构造虚拟环境需要大量领域世界的知识,进一步提高知识表示和本体开发的研究对虚拟现实也十分必要。

(4) **协同工作与虚拟组织构建** 基于 INTERNET 或 INTRANET 的网络办公正在改变着人们的工作方式,企业和组织机构的运作方式。它可把一切具有信息属性的、可以用符号等价物替代实物的工作流程,统统在网络计算环境中予以实现。在这样的背景下,企业和机构的相当一部分管理职能,完全可以由网络上自主工作的一组智能体来协同实现;涉及到设计、写作、数据操纵、软件开发等许多符号性的工作岗位之间的协同工作,也同样可以靠智能体来协调并代理完成一些辅助性的工作。

宏观上有序的世界文明的快速发展也导致了世界在微观上的复杂、无序和不可预言,这迫使人们努力寻求动态的联盟与合作,以提高自身解决所面临问题的能力和快速应变能力。虚拟组织可将时空上分布但能力和资源上互补的常规组织或个人组合在一起而不必配置公有资源,以便应特别使命的需求而快捷地建立并在使命完成时自动解体,这一模式正在发展成为实现的动态联盟和合作的主流方式。虚拟组织、虚拟企业和虚拟团队已被作为 21 世纪新一代组织的核心。无疑,强大的信息基础是虚拟组织即时(应变)建立和高效运转的必要保证。由于要求信息基础具有高度的开放性、可重构性和可伸缩性,传统的信息技术和软件工程方法难以支持。基于智能体的体系结构不仅能改变传统的被动型软件模块为由智能体启动的主动型模块,还能无缝地封装传统的信息系统、软件和信息源,因而可以作为特化通用信息基础到支持虚拟组织高效运转的有效途径。可以说,正是由于智能体协作技术的快速发展,才推动了虚拟组织从理想走向现实,并必将渗透到人类社会的各个层面和角落。开发虚拟组织的关键在于提供强大的信息基础,运用智能体技术,以及支持智能体协同工作的通信技术(包括共享本体、通信语言和通信促进器),可以在现有信息基础上建立起支持虚拟组织运转的信息基础。

17.6.8.3 未来的智能体社会

因特网正在成为现代社会信息服务的基础设施,也正在成为人们交流和工作的场所。未来,当各种分布式计算和智能体系统广泛运行于其上之时,网络无疑将会发展成为一个广义的“智能体社会”。在这一广义的智能体社会中,有各种各样的人,有各种各样的组织,有各种各样的信息源,有各种各样的交流与协作,也有各种各样的服务等。如此,人与人的交互与协作,人与“因特网”的交互与协作,都可在基于“智能体社会”的网络社会上来进行。而各种各样的智能体,将在其中起着重要的作用。它们的服务和协同工作,将辅助或代表人们(特定用户)去完成各种任务,实现人们(各类用户)之间的便利地交互与合作。

人们在期望广泛应用的“智能体”所带来便利的同时,也不应忽视它们可能会产生的问题甚至灾难。有人认为,鉴于大量的智能体将“纵横”于网络之上,若对它们的行为不加约束,任其在网上随意行动,就很有可能会使高度动态的网络本来就有的混沌状态雪上加霜。所以,在未来的智能体社会中,为各智能体建立一定的行为准则,将是十分必要的。而制定智能体行为准则,首先需要全面考察智能体用户、智能体设计者和智能体所存取信息资源的提供者们的需求、行为和动机。智能体服务对象的需求和动机应是很明确的,即在于可获得

快速、准确和优质的信息和服务；智能体的设计者们所注重的，则在于通过提供智能体以补足和延伸网络信息和服务系统对信息和事务处理的能力；而信息源提供者的兴趣，则在于传递他们希望发出的信息。这些需求和动机的不同也使得我们可以把所有的智能体划分为两大类：**供应智能体**和**需求智能体**。前者提供后者所需求的信息和服务，后者则请求前者提供信息和服务。为此，在制定智能体行为准则时，应分别考虑这两类智能体的特点而对其行为作一定限制。

有人提出，供应智能体所需遵循的行为准则应包括：

- 可鉴别性—供应智能体的活动应易于辨认，并可由管理人员追踪。
- 开放性—由供应智能体提供的信息和服务应能为大范围需求智能体所取用。
- 节制性—供应智能体提供信息和服务的频率和步伐应与服务器和网络通信能力相适应。
- 遵从性—供应智能体须遵从管理人员施加的管理约束。
- 有效性—供应智能体提供的服务应保持精确和及时更新。
- 可信性—供应智能体所提供的信息和服务应是可信的，或者应是有可信级别的。

显然，这些行为准则相互制约，需达到某种平衡。而缺乏有效服务的供应智能体是不会被智能体社会所“容忍”的。

与之相对应地，需求智能体也应遵从一定的行为准则，这些行为准则应包括：

- 可鉴别性—需求智能体的活动应易于辨认，并可被追踪。
- 节制性—需求智能体对信息和服务的请求频率和步伐应与服务器和网络通信能力相适应。
- 合适性—需求智能体应只向合适的服务器提出适当的问题。
- 警觉性—需求智能体不应接受会产生不可预测后果的服务和请求。

此外，对于作为“用户代理”去自行处理信息和事务的智能体，还应遵从以下行为准则：

- 维护主人的隐私权。
- 遵从主人的授权，不会因越权行事而损害主人利益。
- 对于主人和伙伴智能体诚实。

等等。

为了维护智能体社会的安全性和可靠性，有人还建议，所有智能体均不应破坏性地改变网络信息环境，而应在其工作结束时尽量维持环境不变，并限制对公共资源的浪费。制定健全的智能体社会行为准则对未来智能体社会的建设无疑是十分重要的，它的完善将促使更多的智能体应用于分布式的网络服务环境。

17.7 人-机自然交互与自然语言理解

对自然语言理解研究的动力来自于多个方面：一是人际交往。由于多种语言的现实存在，在全球化的交往进程中，需要语言的翻译，特别是机器的翻译。二是人-机交互。人-机具有天然的互补能力，但无论是机器代替人类工作或是人类操作机器，无论是机器学习还是人性化界面设计，都需要机器能“懂”人的话。另外，自然语言是表示知识最为直接的方法。因此，自然语言理解的研究也可为专家系统的知识获取提供了新的途径。

自然语言理解作为人-机交互和语言信息处理技术的一个高层次的研究方向，一直是人工智能所关注的核心课题之一。也是现今计算机科学技术领域最活跃的研究方向之一。显然，如果计算机能够真正理解自然语言，人-机间的信息交流能够以人们所熟悉的自然语言来进

行，无疑将大大方便人们的生产和生活，它既具有无法估量的现实价值，也将是人工智能研究理论上的一项重大突破。

随着计算机硬件的快速发展，（智能）计算机越来越广泛地进入到我们的日常生活，计算机与自然语言相结合的领域也越来越广阔。继机器翻译之后，信息检索、文本分类、篇章理解、自动文摘、自动校对、词典自动编纂、文字自动识别等等领域都在不同程度上要求计算机具备自动分析、理解和自动生成自然语言的能力。自然语言理解研究可以使得计算机在一定程度上能理解人类的自然语言，从而帮助人们完成人-机交互、机器翻译、信息提取、信息检索、文本分类等各项工作，对提高社会生产率，便利人们生活，都有巨大的价值和意义。

智能的核心是思维；语言是思维的表达；因此，语言也是智能的表达。理解（认知），是智能的内核；因此自然语言理解是智能系统研究的一个重要方向。自然语言理解本质上是人工智能、信息科学、认知科学和语言学的交叉科学。自然语言理解系统本质上就是一类智能系统。

本节，我们将在介绍人-机自然交互的基础上，进一步讨论计算机理解自然语言所需采用的方法和技术，然后介绍建立在语言理解基础上的机器翻译技术。自然语言可区分为书面语和口语。书面语理解包括词法、句法和语义分析；口语理解需外加语音分析。考虑到书面语理解比口语理解容易，且又是口语理解的基础，本节只论述书面语理解。并以句法和语义分析为重点。

17.7.1 人-机自然交互研究概述

人-机交互是一门研究人类所使用的交互式计算系统的设计、实施、评估及相关主要现象的学科。一个简单的人-机交互简化模型如图 17.7.1 可所示。

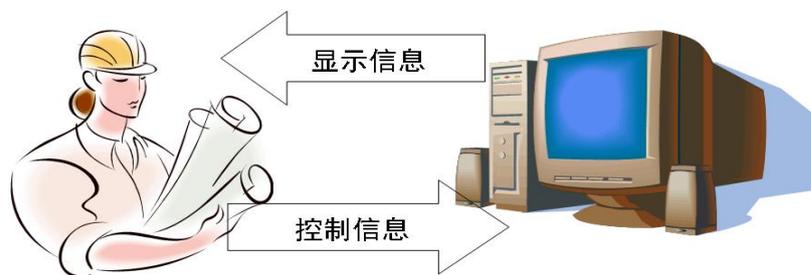


图 17.7.1 人机交互的简化模型

人-机交互研究的目标，是使人在利用计算机完成任务时，能达到：安全性(Safely)、有效性(effectively)、高效率(Efficiently)、愉悦感(Enjoyably)和高可用性(Usability)等要求。其中，高可用性(Usability)包括易于学会如何使用；可高速执行任务；低的用户错误率和高的用户的主观满意度；对用户的持久吸引力等。人-机交互研究的**历史**，基本上就是计算机发展的历史！对人-机交互研究的理解，可通过对计算机应用范式变迁的认识来完成。

计算机视频显示部件的采用曾是人-机交互的一个里程碑式的发展，计算机从此可以进行可视化操作和信息处理。道格拉斯·恩格尔巴特(Douglas Engelbart)发明的鼠标，Windows的图形用户界面，使人-机交互实现了新的范式：直接操作(Direct Manipulation, DM)。其特点是：对象可见；渐进动作和快速反馈；使用动作代替语言。Ted Nelson是Hypertext(超文本)、hypermedia(超媒体)等概念的(合作)提出者。Hypertext、WWW等将信息看

作互联的节点，而不是线性流。Mark Weiser 提出普适计算是 21 世纪的计算模式，无处不在的计算描述了具有丰富计算资源和通信能力和环境之间关系的场景，在需要的任何时间和任何地点都可以提供信息和服务，这个环境和人们已逐渐地融合在了一起。“最好的仆人是看不见的仆人”。技术的发展除了使计算机“Invisible”，还提出了“宁静技术（calm technology）”的概念，可充分利用人体丰富多彩的感知和动作器官以及人们与日常物理世界打交道时所形成的自然交互技能来获得计算机所提供的服务。

人-机交互研究的历史，过去一直是人适应计算机的历史！人-机交互研究的未来应是计算机适应人的未来！未来我们会如何使用计算机？下一代自然的人-机界面会是什么？需要大家的想象力，也需要人-机交互研究的努力！

人们对于现存的人-机交互方式并不满足，因为人与人的交互不是通过人手操作来实现的，而是通过听觉与视觉以语言、表情、手势和体势等来进行的。人们期望新一代的人-机交互能像人与人交互那样自然、准确和快捷。于是人-机交互的研究便进到了多模态的阶段，称为人-机自然交互（Human-Computer Nature Interaction, HCNI 或 Human-Machine Nature Interaction, HMNI）。人-机自然交互与过去的人-机交互的方式有本质上的差别，这是一种人-机交互的革命，是摆脱人手操作，赋予机器以智能，通过听觉和视觉通道来实现的新一代交互方式。

人-机自然交互的一个基本特征是“只动嘴，不动手”，赋予计算机听觉与视觉智能，使计算机能认识交互的对象，理解以语言形式表达的说话内容，理解交互对象的情感，可按人的要求去做，或回答人提出的问题，达到交互的目的。

人-机交互要识别交互对象，或称说话人识别，即知道在与谁对话，这可以由视觉和听觉信息通道获取的知识中进行识别和判断；要理解对话的内容，以便执行对话人发出的指令，或回答对话人提出的问题，这就要“懂”的对话人所说的语言。由于语言本身是双模态的，除听话听声外，理解对方的情感状态对于理解对方的对话的内容、思想和意图也是非常重要的。人与人交流时，总是注视对方，看对方的眼神，多人会话还要区分说话人，因此，在人-机自然交互时，输入与输出应是多模态的，至少是听觉与视觉双模态的；在交互方式上，应包括人-机的对话、情感识别和理解，有时甚至需要进行定位与跟踪。因此，在交互界面上，应采用智能化的界面，具有“看、听、读、说、写、触、嗅”等感觉和知觉能力，以完成人与计算机的多模态交互；它应具有本体感觉和自我意识，具有学习、记忆乃至情感等认知能力，具备说话、绘画等表达能力，以实现拟人的会话功能；它是开放性、协调性、分布性的，同时也是人性化服务的；在交互环境上，还应能营造 2 维或 3 维的真实的虚拟环境和虚拟真人，完成虚拟环境的要求，使人-机的交互环境好像人与人之间的交互环境一样。

比尔·盖茨曾经预言，电脑毫无表情的时代即将结束，21 世纪将是情感电脑大行其道的时代。当有那么一天，计算机达到了这一步，那将是什么样的情景呢？我们期待着各种人性化的驯服的智能工具的出现，我们让它做什么，它就会高兴地去做什么。智能化的人-机交互无疑将会发挥重要作用。

17.7.2 自然语言理解研究的概况与困难

让机器理解人类语言，是人类的愿望，也是现实的需要。计算机技术和人工智能技术的快速发展，曾导致和推动了自然语言理解的研究和开发。机器翻译是自然语言理解最早的研究领域。20 世纪中叶，由苏、美等国开展的俄-英和英-俄机器翻译研究工作，开启了自然语言理解的早期研究。由于缺乏先进的理论，再加上当时计算机计算能力低下，机器翻译研

究未取得实质性进展。1966年，美国科学院曾发表过一份报告，认为，机器翻译在较长时期内不会取得成功。

计算机技术和人工智能技术的发展，一直推动着自然语言理解的研究和开发不断取得进步。与机器翻译的不成功相对照，20世纪60年代中期出现的一些基于关键词匹配技术的初级自然语言理解系统取得了成功，并投入了实验性应用。这些初级系统中最有名的一个是美国麻省理工学院设计的ELIZA系统，它能模拟心理治疗医生去同患者谈话，并几乎达到以假乱真的程度。ELIZA获得成功的关键在于其应用了事先存放于系统中的大量包含某些关键词的语句模式，而且使用的匹配样本灵活多样，以至于能依赖患者的回答不断产生随机应变的相关问话，使对话显示出惊人的真实性。然而，关键词匹配技术并不作真正意义上的语法分析，仅将当前输入的语句同系统中包含的语句模式逐个进行匹配检查；匹配的成功只取决于语句模式中包含的关键词及其排列次序，非关键词不能影响系统理解。所以，严格地讲，基于关键词匹配的理解系统并非真正的自然语言理解系统，其既不懂语法，又不懂语义，充其量只是一种近似匹配系统。近似匹配的最大弱点是容易导致错误的语句分析。例如，可能会把“我朋友的姐姐喜欢我”理解为“姐姐喜欢我”，而出现严重错误。

进入20世纪70年代后，自然语言理解的研究在句法-语义分析技术方面取得重要进展，若干有影响的自然语言理解系统脱颖而出，如伍兹（Woods）的LUNAR系统，威诺甘德（Winograd）的SHRDLU系统，亨德瑞克斯（Hendrix）的LADDER系统等。这些系统都是受限的人机对话系统。由于把句法、语义、推理、上下文和背景知识灵活地结合于一体，在语言分析的深度和难度方面都比60年代的初级自然语言理解系统有了长足的进步。进入80年代之后，实用化和工程化的努力导致一批商品化的自然语言人-机接口和机器翻译系统出现于国际市场。著名的人机接口系统有美国人工智能公司（AIC）研制的Intellect和费雷公司研制的Themis。在自然语言理解研究取得重要成功的基础上，机器翻译研究工作重新兴起，并逐步走向实用。有较高实用化水平的机译系统包括美国的META和LOGOS，日本的PIVOT和HICAT；我国中软公司开发的译星（TRANSTAR）等。

尽管相关研究取得了长足进展，但自然语言理解的困难，使得人工系统至今仍无法与人的理解能力相比。导致理解困难的有多个方面，其中最主要的因素有：目标表示的复杂性；映射类型的多样性；源表达中各元素间交互程度的差异性。

(1) 目标表示的复杂性。不同的应用领域对自然语言理解系统的目标表示提出不同的要求，进而导致目标表示的复杂程度有很大差异。在简单的情况下，只须抓住语句中出现的预定类型的单词作所谓的关键词匹配检查，就可实现肤浅的理解。例如，在火车订票询问系统中，一个受限的自然语言接口或许会接收到这样的询问语句：“有从上海到北京的特快车票吗？”通过与预先提供的匹配样本：“有从〈起点〉到〈终点〉的〈车种〉车票吗？”作匹配比较，可以建立目标表示：（搜索 起点=上海，终点=北京，车种=特快），该目标表示可进一步转变为数据库查询语句，就可检索出满足顾客询问的车票供应情况，显示给顾客。但上述基于匹配样本作关键词匹配属肤浅的理解，其实并非真正的自然语言理解。要让人工系统像人那样理解接收到的语句需要复杂得多的目标表示。例如，要理解以下语句：“小王告诉小丽：他不想与她一起去看电影；小丽觉得很伤心。”并回答诸如“为何小丽很伤心？”那样的问题，就必须要建立复杂得多的目标表达，而且，还必须知道相关的背景知识，如小王和小丽各自的情况和他们以前的关系等，才能使人工系统产生深入的理解，以便贴切理解小丽感到伤心的原因。显然，建立复杂的目标表示远比建立简单的目标表示困难得多，因为需要从接收的语句中获取更多的信息，并且需要与相关的背景知识建立相关联系。

(2) **映射类型的多样性。**在自然语言理解过程中，源语言表示转变为目标表示存在四种方式：一对一映射—这是最简单的映射；多对一映射—即多个不同的源语句可转变为同一目标表示；一对多映射—即单一源语句可转变为多个目标表示；多对多映射—是一对多映射和多对一映射的综合。若将自然语言语句转化为语义网络表示，一对一映射意指语句与语义网络表示间有严格的对应。考虑到自然语言词汇和结构的丰富性，几乎不存在这种简单的映射关系，普遍存在的是多对一映射和一对多映射。例如，“小王给小丽一本书；小丽得到小王给的一本书；一本书是小王给小丽的”；表达的是同样的语义，所以应映射为同一语义网络表示。这种多对一的映射要求自然语言理解系统了解用源语言表示同一语义的所有方法，但通常不要求过多的其它相关知识。而一对多映射则反映了自然语言固有的歧义性。例如，“他说他不打了”就有多种可能的语义，并映射到不同的语义网络：这里的歧义性集中体现在语句中第二个“他”的所指是其本人还是其他什么人，“打”又意指什么。显然，仅靠语法知识是无法消解这种歧义性的，还要获取关于说话场合的相关背景知识。多对多映射是多对一映射和一对多映射的联合，更需要综合应用语言学知识和说话背景知识，才能获得较高水平的理解。

(3) **源表达中各元素间交互程度的差异性。**自然语言是上下文相关的语言，体现在句子成分间的语义相关性，即在语句中处于同样位置的词组，会因语义相关性而导致语法结构的较大差异。

显然，导致自然语言理解困难的上述因素是不可避免的，只有设法克服它们，才能设计出高性能的理解系统。自然语言是人类特有的用于交流的手段，世界上大部分民族都有自己独特的语言。理解自然语言是一件困难的任务，不仅要依赖语言学方面的知识，还需要与话题相关的背景知识。因此，必须较好地应用和组合这两种知识，才能建立有效的自然语言理解程序。因此，自然语言理解的研究要涉及到历史学、社会学、语言学、逻辑学、心理学、计算机科学和数学等相关学科发展，特别是与人工智能的研究息息相关。

17.7.3 自然语言理解（曾提出的）实现模型

自然语言理解研究的目标是研制能理解自然语言的计算机系统。从微观上讲，语言理解是指从自然语言到机器(计算机系统)之间的一种映射。从宏观上看，语言理解是指机器能够执行人类所期望的某些语言功能。这些功能包括：回答有关提问；提取材料摘要；不同词语叙述；不同语言翻译等。由于自然语言是音义结合的词汇和语法体系，是实现思维活动的物质形式。它首先是由特定的语句组成，而每个语句又由具有特定的“音、[形]、义”的词汇组成；词汇组成语句和语言时，又要遵循一定的语法与语义规则，才可构成有意义的和可理解的句子，句子按一定的形式再构成篇章等。因此，自然语言的理解过程也应是分层次的，要包括：语音分析、词法分析、句法分析和语义分析等。

语法是语言的组织规律。语法规则制约着如何把词素构成词，把词构成词组和句子。语言正是在这种严密的制约关系中构成的。用词素构成词的规则叫构词规则。词素是构成词的最小的有意义的单位。语法中的另一部分就是句法。句法也可分成两部分：词组构造法和造句法。词组构造法是词搭配成词组的规则。造句法则是用词或词组造句的规则。整个语言的构造系统可如图 17.7.2 所示。

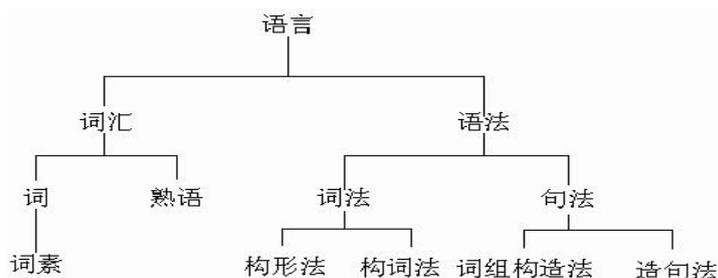


图 17.7.2 语言的构成系统

关于自然语言的理解，尽管一直是智能研究十分活跃的领域，但至今尚无精确的定义。原因在于理解并非一个绝对的概念，其取决于应用目标，如回答问题、执行命令、人-机对话、机器翻译等。从设计自然语言理解程序的实用主义角度，可以按照目标实现的正确性来衡量理解程序的性能。这就涉及到目标表示的问题。所以，从这个意义上讲，语言的翻译是把语言的一种表示形式转变为另一种表示形式的过程，而自然语言的理解，则至少应包括下列几个方面的内容：

- ① 能够理解句子的正确词序规则和概念，又能理解不符合规则的句子；
- ② 知道词的确切含义、形式、词类及构词法；
- ③ 了解词的语义分类以及词的多义性和歧义性；了解词的指定和不定特性及其所有属性特征；
- ④ 相关问题领域的结构知识和时间概念；
- ⑤ 语言的语气信息和韵律表现；
- ⑥ 有关语言表达形式的文学知识；
- ⑦ 论域的背景知识。

作为自然语言理解技术的重要应用，机器翻译，包括书面语翻译和口语翻译，是其重要内容。机器翻译旨在让计算机自动将源语言表示的语句转变为目标语言表示的语句，具有双重语言理解和语言自动生成功能。

目前，关于自然语言理解的研究自底而上可以分为四个基本层面，它们分别是：词典构造、语法分析、语义分析和篇章分析。下面，我们将分别论述其相关研究。

17.7.4 自然语言理解—词典的构造

语言词典是自然语言理解的基础平台，其中包含了计算机对自然语言进行分析处理所需的语法、语义等知识。研究语言词典的构造可为语法、语义分析等工作提供更好的平台。

自然语言理解中的知识平台—**语言词典**又被称为“电子词典”、“机器词典”、“计算词典”等。一部具有尽可能多的相关语法信息、语义信息、语用信息的语言词典是自然语言理解系统平台的基础和关键所在。机器词典是机器翻译的基础，没有好的机器词典，机器翻译就等于无米之炊，根本无法进行的。

长期以来，词典一直被看作是“储存某一语言之特定知识的知识库”。传统词典是放在书房或图书馆的书架上，供人们不时查阅的参考书。在互联网时代，这些词典已经演变成了各种在线词典—可被计算机访问的词典数据库。这些在线词典是新旧形式的一种简单而自然的结合，为人们查找词条提供了非常快捷的方法。而我们所要研究的用于自然语言理解的知识平台—“语言词典”（计算词典）与这种一般意义上的在线单语或双语词典之间又有很大

区别。二者间的不同是由查阅对象的不同造成的。在线词典用于提供给人类进行查阅语义和译文等以理解生词，而计算词典提供给计算机查阅，不仅需要包括词的语义知识，还需要包含词的语法知识，甚或语用知识，更甚或世界知识。而且计算机阅读的机制与人类不同，因此，在词典构成上，计算词典也大大不同于一般的在线词典。

一般在线词典不是按照词义信息来组织词语的，一些意义上有关联的词语被随意分布在了词典各处，且它们之间缺少纽带相连接。一般在线词典的组织结构使得它不适合直接用于自然语言处理。从这个意义上来说，自然语言处理所需要的词典是那种以人类心理词典为原型构造，并且适合计算机分析与“理解”的词汇知识库。因此，怎样为语言词汇的知识表示提供一个合适的模式，是自然语言理解的研究中特别关注的一个课题。

目前国内外在语言词典方面的研究成果主要有 Princeton 大学的 WordNet、Berkeley 大学的 FrameNet、巴黎大学 Gross 的 INTEX，Microsoft 的 MindNet 等。

(1) **WordNet** 美国普林斯顿大学认知科学实验室开发的 WordNet 被称为“或许是迄今为止对自然语言处理系统最为重要和使用最为广泛的词汇资源”，已有很多个版本。它由 Miller 等一批心理词汇学家和语言学家开发，是一个描写英语词汇层语义关系的词典数据库。它是在当前基于人类词汇记忆的心理语言学理论推动下产生的，是传统词典信息与现代计算机技术以及心理语言学的研究成果有效结合的产物。Miller 等人最初的想法只是打算提供一个能与传统的在线词典紧密结合的辅助工具，用来按概念而不仅仅按字母顺序查词典。然而，随着工作的进展，Wordnet 的组织结构以及它所追求的目标都闪现出其特有的火花，因为它实践了基于心理语言学家研究成果的设想。

WordNet 目前有近 95600 个不同的词型(51500 个简单词和 44100 个搭配词)，这些词组成约 70100 个词义(或者说同义词集合)。Wordnet 与其他标准词典最显著的不同在于 Wordnet 将词汇分成五个分类：名词、动词、形容词、副词和虚词。实际上，Wordnet 仅包含名词、动词、形容词和副词。基于虚词通常是作为语言句法成分的一部分，WordNet 忽略了英语中较小的虚词集。系统中的名词、动词和形容词都聚类为代表某一基本词汇概念的同义词集合，并在这些同义词集合之间建立起各种语义关系，如同义关系(Synonymy)、反义关系(Antonymy)、上下语义关系(Hyponymy)、部分-整体关系(Part of)等等。

在 WordNet 中，名词是按主题的等级层次组织，动词是按各种搭配关系来组织，形容词和副词以 n 维超空间组织。加强文法分类的代价是增加了大量的冗余，但优点是其语义组织之间的根本差异清晰可见并可被系统地开发。WordNet 最具特色之处是试图根据词义而不是词形来组织词汇信息，从这方面讲，WordNet 更像是一部语义词典。WordNet 已被广泛应用于各种研究和应用中。北京大学计算语言学研究所开发的中文概念辞书(CCD)，这是在 WordNet 框架下研制的现代汉语概念词典。在欧洲，以 WordNet 为原型采用半自动的处理路线开发了 EuroWordNet，以类似的格式来表现各种欧洲语言的词汇层语义关系。

(2) **FrameNet** 美国 Berkeley 大学 Fillmore 教授主持的 FrameNet 工程是一个基于框架语义理论的在线英语词汇资源库。其框架语义从格语法发展而来，并对各种“格”进行了细分和具体化。

FrameNet 主要描写动词的语义，它从 WordNet 上选取了 2000 个动词，从中得到 75 个语义框架。例如动词“Categorize”的框架被定义为：某人(Cognizer)将某个物件(Item)视为某个类(Category)。同原先的格语法相比，将行为施事具体化为认知者 Cognizer，将行为受事具体化为事物 Item，并根据特定体动词的性质增加了一个作为分类结果的语义角色

仅靠智能技术尚不足以建立达到人类同等水平的自然语言理解系统；正如专家系统的成功不能脱离主流计算环境那样，自然语言理解的研究必须与快速发展的计算技术相结合，使自然语言理解系统能建立在海量的语言学知识和应用背景知识的基础上，以期突破目前理解系统的局限性。这方面研究的一个典型设想就是将自然语言理解系统建立在大型语料库基础上。尽管建立大型语料库和让计算机在作句法和语义分析时自动地从大型语料库中获取相关知识都是十分艰巨的任务，但许多研究者对其开发前景充满信心。人们相信，随着计算机技术的日新月异和研究者们的不懈努力，攻克自然语言理解难关的日子终究会来到。

17.7.5.2 句法分析的形式文法

在自然语言理解系统中，句法分析由专门设计的分析器进行，分析器依赖自然语言遵从的文法（语法）分析输入的句子，并把每个合法句转变为一棵分析树。

自然语言的文法属于所谓的短语结构文法。短语结构文法（G）可形式地表示为一个4元组： $G = (V, \Sigma, P, S)$ ，其中， V 为有穷非空集，称为总词汇表； Σ 是 V 的非空子集，称为终结词汇表；相应地， $V - \Sigma$ 是非终结词汇表，以 N 指示； P 是文法规则的有穷集，每条文法规则均是形如 $\beta \rightarrow \alpha$ 的产生式；其中， $\alpha \in N$ ， $\beta \in V$ ，指示来自总词汇表 V 的符号，可以出现零至任意多次； $S \in N$ ，称为起始符。

能被以这种方式定义的某种文法接收的（识别的）所有合法语句构成遵从该文法的语言。显然， Σ 确定了该语言中合法句可用的符号（单词）， N 指示文法描述词汇，而 P 则用于约束句子中符号的合法排序。

理论上讲，可以应用上述 $\beta \rightarrow \alpha$ 形式的文法规则去描述任何语言，包括自然语言和人工语言（如程序设计语言）。这种规则的定义几乎不受限制，即规则的两边可以出现任意多个符号，且规则可以是递归的。然而，规则表示形式的不受限制会增加句法分析的复杂性；另一方面，对规则的表示形式作限制可以简化句法分析，但同时却也限制了语言的描述功能。

乔姆斯基（Chomsky）依据文法规则表示形式的受限程度，将文法（和语言）划分为四个等级（表 17.7.1），称为乔姆斯基体系。

表 17.7.1 乔姆斯基体系

类型	语种	规则限制
0	递归可枚举语言	无
1	上下文有关语言	左部至少包括一个非终结符，右部符号的个数不少于左部
2	上下文无关语言	左部必须是单一非终结符
3	正则语言	左部必须是单一非终结符，右部必须是单一终结符或单一终结符后跟单一非终结符

处于乔姆斯基体系底部的是正则语言，其采用的3型文法限定文法规则只能是： $A \rightarrow tB$ 或 $A \rightarrow t$ ，其中 $A, B \in N$ ， $t \in \Sigma$ 。文法规则表示形式的高度受限，使得正则语言可以用有限状态自动机作高效的句法分析。有限状态自动机有若干状态，其中必有一个为起始状态，并至少有一个结束状态；自动机的输入会导致状态变化，并在到达目标状态时停机。面向正则语言句法分析的有限状态自动机就以文法规则左部的非终结符指示当前状态，文法规则右部的终结符作为输入，终结符后的非终结符就是自动机将到达的下一状态；若终结符后无非终结符，则自动机在当前状态下停机。若输入结束且此时自动机处于结束状态，则输入就作为一个合法语句而接受；否则输入的是非法语句。

尽管正则文法简单并易于分析，但文法规则的表示太受限制，使其无法用于描述哪怕是人工制定的语言。正则文法主要用于西文的词法分析阶段，如切分单词和识别非法字符。

上下文无关的语言及其文法（2型文法）已广泛应用于定义程序设计语言。这种文法的规则限定其右部只能是单一的非终结符，即非终结符通过文法规则（一种重写规则）的扩展性重写是相互独立的，不受其它符号（单词）的影响，所以称为上下文无关。这种语言的句法分析可以采用基于有限状态自动机的转变网络技术和产生式系统技术，分析的结果则可表示为句法分析树。

然而，自然语言是上下文有关的语言，上下文有关的语言需要用1型文法加以定义，文法规则允许其左部有多个符号（至少包括一个非终结符），以指示上下文相关性；但要求规则右部符号的个数不少于左部，以确保语言的递归性（即语句合法的可判性）。考虑到上下文无关语言的句法分析远比上下文有关语言有效，人们希望在增强上下文无关语言的句法分析基础上，实现自然语言的自动理解。基于这种设想的自然语言句法分析技术就是ATN（增强的转变网络）技术。

乔姆斯基体系的最上层是递归可枚举语言，其采用的0型文法对文法规则的表示形式不作任何限制，从而能使定义的语言提供充分的描述功能。但0型文法不保证语言的递归性，即不能确保语句合法的可判性，所以很少用于定义自然语言。

17.7.5.3 句法分析的策略

为提高句法分析的有效性，除设计适合于句法分析器使用的形式文法外，还应选择作句法分析的策略，包括确定句法分析方式和句法分析过程中的搜索方法。

1. 确定句法分析方式

依据文法的形式定义，作句法分析的方式可区分为两大类：自顶向下——从文法的起始符开始，应用文法规则自顶向下地进行推导，使句法分析树逐步向下扩展，直到相应于被分析句子中每个单词的终结符作为叶子节点加进分析树为止。自底向上——从待分析句子开始，逆向应用文法规则自底向上地进行推导，使句法分析树从指示终结符（相应于句子中单词）的叶子节点逐步向上汇集，直到相应于整个句子的起始符推导出为止。这二种方式的选择主要取决于推理过程中的分枝因素，如同产生式系统推理方向的选择那样。另一个应考虑的主要问题是启发式知识的可用性，即用启发式知识去尽早修剪掉不可能成功的推理路径。有时，这二种方式的组合应用效果更好。例如，可以采用伴有自顶向下筛选的自底向上分析方法，其以自底向上的分析为主；同时又借助于通过自顶向下的预处理产生的一些筛选表，在逆向建立分析树的过程中及时排除一些不可能进一步组合进高层结构的成份。

2. 句法分析中的搜索方法

句法分析是从部分解答向完全解答扩展的过程。这种搜索合法解答的过程有二个极端：搜索所有可能路径或只选择最有可能的一条路径。

下面观察一个待分析英语句子的已输入部分：“Have the students who missed the exam”。它存在二种可能的分析路径去理解这个部分输入的句子：

- 将 Have 视为应用于祈使句的主动词，并预期后续输入为“take it today.”；
- 将 Have 视为应用于疑问句的助动词，并预期后续输入为“taken it today?”。

这里可有四种分析该句子的搜索方法：

(1) 搜索所有路径法——相当于使用宽度优先搜索法对逐个输入的单词可能充当的所有角色都进行分析，并建立分析的中间结果。由于许多不成功的分析路径都被不加鉴别地搜索，会产生大量无用的分析结果，导致效率低下。

(2) 带回溯的最优路径法——相当于在使用深度优先搜索法搜索解答路径的同时，应用启发式知识对推理分支进行排序。上例中，输入的的第一个单词 Have 有二种可能的分析路径，但由于 Have 作为助动词的机会更多（启发式知识），可优先搜索相应于助动词解释的分析路径。然而，搜索失败时的回溯将大幅度降低搜索的效率，应尽量设法（通过增加和精化启

发式知识)减少回溯,或应用从属导向的回溯去减少回溯处理的工作量。

(3)带修补的最优路径法——也是每次只搜索一条分析路径;不同之处在于当发现搜索失败时并不作回溯,只是改变句子成份对应的文法结构。例如,在对于上例句子的分析中,开始时可将 Have 视为助动词,Students 作为主语;但若后续的输入中出现 take 而发现搜索失败时,并不回溯,而是仅作适当的修补工作:Have 改变为主动词,students 改变为宾语,并将缺省的主语设置为 you。通常这种方法比前二种更为有效,其缺点在于要求清晰地给出文法规则之间的交互性,以便指示句子成份文法角色的转变。这样一来,句法分析器就会变得很特别,不再能简单地只依据文法来设计。

(4)等待观察法——实际上就是运用最少承担原则于句法分析,即在分析器从左到右分析输入语句时,对于任何句子成份,只要无足够信息用于决策,就不加判定,等到后续的输入提供了足够信息时再判定。上例子句中,既然在后续的输入接收前不能确定 Have 的文法角色,就只记载 Have 为动词。接下来的带从句的名词词组也只记载其为名词词组,暂不判定其是主语还是宾语。若后续的输入是“take it today.”,则在分析出 take 为动词后,才判定 Have 为主动词,该名词词组为宾语,而 take 为动词不定式。一个称为 PARSIFAL 的句法分析器就采用了这种方法,其设计了一个缓存去存放暂时不能决定句法角色的句子成份,如上述 have。此法很有效,可在很大程度上避免回溯;其缺点在于分析复杂的句子时必须暂存许多角色不能确定的句子成份,从而降低了分析的有效性。

纵观上述四种方法,后二种由于减少或避免了回溯,效率较高;尽管不能保证找到最优的解答,但已满足许多行之有效的分析系统的需求。但需要确保找到最优的解答时,搜索所有路径的方法或许是最好的方法,尽管其低效且代价昂贵。

17.7.5.4 增强的转移网络

1. 句法分析的转移网络

句法模式匹配就是采用句法模式来对语言的句子进行匹配从而进行句法分析。但是自然语言是非常多样化的,因而需要有许多模式。这些模式可用状态转移图来表示,这种用状态转移图来表示的表达方式称之为转移网络(TN, transition network),如图 17.7.1。其中, $\{q_0, q_1, \dots, q_T\}$ 是状态, q_0 是初态, q_T 是终态。弧上给出了状态转移的条件以及转移的方向。该网络可用于分析句子也可用于生成句子。

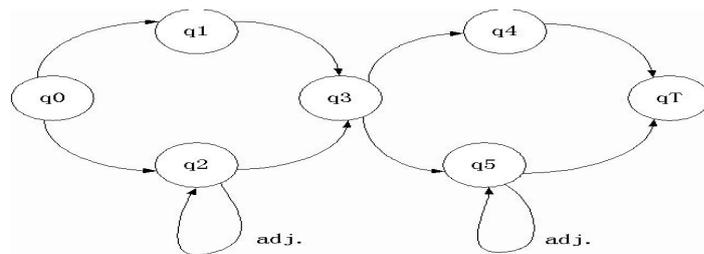


图17.7.1 转移网络示例

若用TN来识别句子 The little orange ducks swallow flies, 则识别过程如图17.7.2所示,

示,

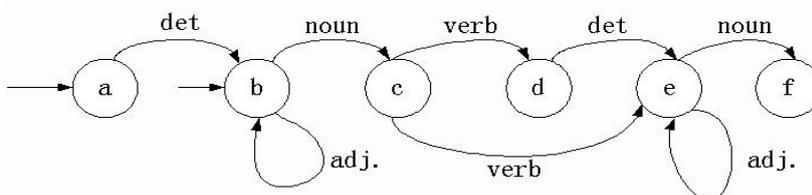


图 17.7.2 用 TN 识别句子过程示例

网络识别结果如图17.7.3所示。

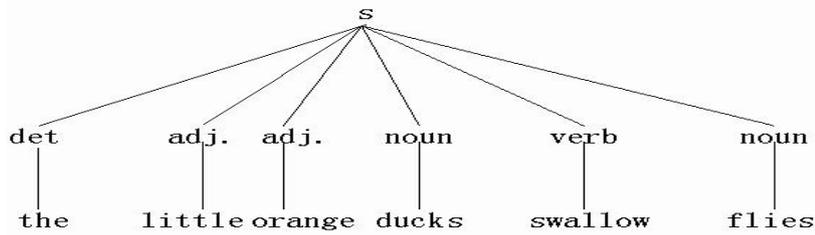


图 17.7.3 用 TN 识别句子结果示例

2. 增强的转移网络—ATN

增强的转移网络 (ATN, Augmented Transition Networks) 是一种得到广泛应用的句法分析器, 由伍兹 (W. Woods) 于 1970 年提出, 并成功地应用于他设计的 LUNAR 系统。ATN 采用增强的上下文无关文法, 即用上下文无关文法描述句子语法结构, 并同时提供有效的方式将各种理解语句所需的知识加进分析器, 以增强分析功能。每个 ATN 类似于一个有限状态自动机, 但通过在指示状态转移的弧上附加各种标签, 可大幅度增强分析功能, 从而使应用 ATN 的句法分析器具有分析上下文有关语言的能力。

ATN 在以下三方面对 RTN 作了扩展和增强:

(1) 添置了一组寄存器 (registers), 用来存储分析过程中得到的中间结果 (如局部句法树) 和有关信息 (如名词短语的人称和数。某些成分的语义特征等);

(2) 每条弧上除了用句法范畴 (如词类和短语标记) 来标注以外, 可以附加任意的测试 (tests), 只有当弧上的这种测试成功之后才能通过这条弧;

(3) 每条弧上还可以附加某些动作 (actions), 当通过一条弧时, 相应的动作便被依次执行, 这些动作主要用来设置或修改寄存器的内容。

设置哪些寄存器完全取决于句法分析的需要, 并没有硬性的规定。例如有关句型的信息: 陈述句 (DCL), 疑问句 (Q), 祈使句 (IMP), 可以存放在名为 TYPE 的寄存器中; 动词信息及其局部结构可存放在名为 V 的寄存器中; 当然也可设置象主语 (SUBJ)、谓语 (PRED)、宾语 (OBJ) 一类的寄存器来存储各种句子成分的信息以及它们的局部结构。所有这些寄存器都可以看作是程序设计中变量, 它们从属于被设置的那个 ATN 子网络。对于 ATN 的后继弧来说, 这些寄存器的内容可以被访问, 并且根据附加在后继弧上的动作可以被复制、修改或组合。

下面是用 BNF 定义的 ATN 形式体系:

```

<transition-network> ::= (<arc-set><arc-set>*)
<arc-set> ::= (<state><arc>*)
<arc> ::= (CAT<category><test><action>*(TO<next-state>))
          |(TST<label><test><action>*(TO<next-state>))
          |(PUSH<state><test><pre-action>*<action>*(TO<next-state>))
          |(POP<form><test><pre-action>*)
          |(JUMP<next-state><test><action>*)
<action> ::= (SETR<register><form>)
            |(SENDR<register><form>)
            |(LIFTR<register><form>)
            |(ADDL<register><form>)
            |(ADDR<register><form>)
  
```

```

<form>::=(GETR<register>)
          |(GETF<feature><word>)
          |(BUILDQ<template><form>*)
          |LEX
          |*
          |<isp-function>
<test>::=(x-AGREE<form1><form2>)
          |(x-START)|...

```

在 BNF 的定义中，尖括号及其内部的英文小写字母串表示非终结符；定义式右侧的圆括号和英文大写字母串都是终结符。"*"号有两种用法：当它出现在非终结符后面，表示这个非终结符可以出现零次或有限多次；当它单独出现时，表示输入串中的当前词，在 JUMP 或 POP 弧上，它同 LEX 的值一样，其词形同输入串一致；在一条 CAT 弧上，它取该词的词根形式。因此，如果 LEX="stopped"，*="stop"。在一条 PUSH 弧上，对于测试<test>和准备动作<pre-action>来说，*等于当前词，但在该弧后继的动作<action>中，它又代表被这条 PUSH 弧启动的那个子网络(即下一层)的回送值。

定义中设置了五种基本的弧型，它们分别是：判类弧 CAT，测试弧 TST，下推运算弧 PUSH，下推返回弧 POP 和跳弧 JUMP。CAT 弧要求当前词必须具有该弧第二个元素 <category>所指定的句法范畴，并满足第三个元素<test>所规定的条件，否则不能通过；当一条 CAT 弧被通过时，输入串中相应的当前词被"消耗"，即导致输入指针前移到下一个词。JUMP 弧不同于 CAT 弧，它不消耗输入串中的当前词，因此伴随着 JUMP 弧的转移可以在不对输入串进行任何处理的情况下发生。

图 17.7.4 则给出关于句子(S)的 ATN 文法定义。

```

(S/ (PUSH NP/T
    (SETR SUBJ *)
    (SETR TXPE (QUOTE DCL))
    (TO Q1))
  (CAT AUX T
    (SETR AUX *)
    (SETR TYPE (QUOTE Q))
    (TO Q2)))
(Q1 (CAT V T
    SETR AUX NIL)
  (SETR V *)
  (TO Q4))
(CAT V T
  (SETR AUX *)
  (TO Q3)))
(Q2 (PUSH NP/T
    (SETR SUBJ *)
    (TO Q3)))
(Q3 (CAT V T
    (SETR V *)
    (TO Q4))
  (SETR VP (BUILDQ (VP (V +) * ) V))
  (PUSH NP/T
    (SETR VP (BUILDQ (VP (V +) * ) V))

```

```

(TO Q5)))
(Q5 (POP (BUILDQ (S + + + +) TYPE SUBJ AUX VP) T)
(PUSH PP/T
(SETR VP (APPEND (GETR VP) (LIST *)))
(TO Q5)))

```

图 17.7.4 关于句子 (S) 的 ATN 文法定义

ATN 状态转移弧上的标签可以是以下四种形式或这些形式的组合:

① 特别的非终结符, 如 NP(名词词组)、aux(助动词)等。

② 下推到下层 ATN, 使分析控制转变到下层 ATN。例如, 图 17.7.4 中 ATN 文法定义的第一行通过“PUSH NP/”指示句法分析器将分析控制转变到关于名词词组 (NP) 的 ATN。下推的过程可以递归进行, 例如在关于 NP 的 ATN 中, 可以下推到关于 PP 的 ATN (以状态 q7 上的转变弧 PP 指示)。假如一个句子的句首部分是:

The boy with the dog with a long tail.

则 ATN 的递归调用将是:

S→NP→PP→NP→PP→NP

③ 测试程序, 用于对现行的输入或输入句子成份作检测。例如“CATAUX”就指示对当前分析的单词作词性是否为助动词的检测。

④ 结构程序, 指示形成部分或完整的句法分析结构, 例如“(BUILDQ (S + + + (VP +)))”。

此外, 分析器还可使用若干内部寄存器去保存分析得出的中间结果, 如 TYPE、SUBJ 等。保存中间结果时使用赋值语句, 如“(SETR TYPE (QUOTE DCL))”将被分析句子的句型 (TYPE) 设置为陈述句 (DCL), 而“(SETR SUBJ *)”则将主语 (SUBJ) 设置为当前输入流 (句子的已输入部分)。

3. ATN 的分析功能与不足

除提供强大的弧标签描述功能外, ATN 还支持以下分析功能: ① ATN 互联, 句法分析系统可以包括许多小的 ATN, 这些 ATN 通过 NP、PP 等弧标签互联成大系统。② 寄存器内容可交换, 以支持分析结果的适当变换。例如在分析被动句“Mary was hit by John.”时, SUBJ←Mary, OBJ of“by”←John; 分析系统可以交换这二个寄存器的内容, 使得 SUBJ←John, OBJ←Mary, 并由此将被动句转变为主动句, 使分析结果与“John hit Mary”相同。③ 在状态转变弧上设置任意检测。例如, 可将主语的单复数记载于一寄存器, 用于核查动词单复数的一致性; 这种检测可置于符号“T”的位置上。其他检测也可以这种方式加进 ATN, 包括语义方面的检查。④ 作为句法分析的通用机制。应切记 ATN 表示方法本身并不包含任何特定语言的文法, 只是作为定义和使用文法的一种手段。前述句法分析过程的四种搜索策略中, 后三种均能以 ATN 表示方法加以实现。上面给出的句法分析例采用的是带回溯的最优路径法。若让系统在发现分析错误时修正或转换各寄存器的内容, 则可以实现带修补的最优路径法。也可以设计一些系统寄存器作为缓存区, 记载尚未确定文法角色的句子成份, 以便在最终确定文法角色时把缓存区的内容送去适当的寄存器, 从而实现等待观察法。当然, 如何组合多种方法和策略进 ATN 是一个技巧问题, 并要视源语言的特点和应用领域的特点而定。

尽管 ATN 在种种自然语言理解系统中显得很有用处, 但也存在着一些不足: ① 在大量回溯存在的情况下, 应用 ATN 的代价昂贵, 并使句法分析效率低下。鉴于自然语言存在高度的二义性, 回溯往往不可避免。② 难以运用语义知识。尽管可以在状态转变弧上附加语义检测, 但目前尚无有效的方式去应用启发式知识。③ 缺乏部分匹配能力。基于 ATN 的句法分析属严格的匹配技术, 其识别的句子必须是转变网络的一条路径, 且句子中的所有单词均需识别。然而, 人的自然语言理解常容许部分匹配, 特别是口语, 即使文法不通, 也听得懂。尽管 ATN 存在这些不足, 但其仍不失为句法分析的有效手段, 并已广泛用于自然语言理解系统。

17.7.6 自然语言理解—语义分析

句法分析的通过不等于已经理解了接收的语句，必须进行语义解释，把分析得到的句法成份与应用领域中的目标表示相关联，才能产生正确唯一的理解。简单的做法就是依次使用分离的句法分析器和语义解释器。此法的困难在于，不少场合下，不结合语义是无法确定句法结构的。ATN 允许把语义信息加进句法分析，但并不充分支持语义解释。为有效地实现语义分析，并能与句法分析紧密结合，人们已经研究了多种作语义分析的方法，本节拟介绍其中最具有影响的二种：语义文法和格文法。

1. 语义文法

语义文法将文法知识和语义知识组合起来，以统一的方式定义为文法规则集。语义文法是上下文无关的，形态上与面向自然语言的常见文法相同，只是不采用 NP、VP、PP 等表示句法成份的非终结符，而是使用能表示语义类型的符号，如图 17.7.5 中的 SHIP、LOC 等，从而可定义包含语义信息的文法规则。亨德瑞克斯 (Hendrix) 于 1978 年建立的 LADDER 系统可以作为应用语义文法的典型例，其用于作为美国海军的一个大型分布式数据库的自然语言接口。

语义文法的主要特点是在终结符与非终结符之间建立固定的约束关系。例如，图 17.7.5 中的关于海军舰只的一条规则：

S → what is SHIP-PROPERTY of SHIP?

就在其右部指定 “What is” 必须与 SHIP-PROPERTY 联合构成疑问句，这种单词间的约束关系显然表示了语义信息。用语义文法分析语句的方法与普通的句法分析文法类同，特别是同样可用 ATN 对语句作语义文法分析。

S → what is SHIP-PROPERTY of SHIP?
SHIP-PROPERTY → the SHIP-PROP | SHIP-PROP
SHIP-PROP → speed | length | draft | beam | type
SHIP → SHIP-NAME | the fastest SHIP2 | the biggest SHIP2 | SHIP2
SHIP-NAME → Kennedy | Kitty Hawk | Constellation | ...
SHIP2 → COUNTRYS SHIP3 | SHIP3
SHIP3 → SHIPTYPE LOC | SHIPTYPE
SHIPTYPE → carrier | submarine | rowboat
COUNTRYS → American | French | British | Russian | ...
LOC → in the Mediterranean | in the Pacific | ...

图 17.7.5 LADDER 语义文法规则的子集

语义文法的优点归纳如下：

① 语句的理解一次完成，由于文法和语义信息已融合于文法规则，不需分二阶段作句法和语义分析，一次成功。

② 避免歧义，语义信息的加入能帮助确定句法成份的角色，从而可避免纯句法分析面临的歧义性。一个典型的例子如下：What is the closest ship to the Biddle with a doctor abroad? 这里的 with 短语修饰 ship 或 Biddle，有歧义性。但在语义文法中可以指示介词词组只能用于修饰 SHIP，就可消除这一歧义性。当然，这样一来，分析器只能理解受限的自然语言；不过对于特定的应用领域，适当的限制不会影响使用。

③ 不影响语义的句法问题可以忽略。例如疑问句 What is the length of Submarine? 照样可为图 17.7.5 所示的语义文法接受，尽管按英语语法，“What” 改为 “How much” 更好。

语义文法的使用也存在一些不足：

① 需要大量文法规则，因为要消除歧义性，就需用多条语义规则取代单一的会引起歧义性的句法分析规则。例如，要让介词词组修饰多个名词，就须为每个名词定义一条语义规

则。众所周知，自然语言中的歧义性不胜枚举，试图解决歧义性势必要求定义大量的语义规则。所以，用语义文法去理解自然语言十分低效。

② 即使对于受限的自然语言理解系统，也需要大量的语法规则，从而导致语义文法的使用代价昂贵。

2. 格文法

把对语句的句法分析和语义分析组合起来的另一途径是设计格文法。格文法的特点是许以动词为中心构造分析结果，尽管语法规则只描述句法，但分析结果产生的结构却相应于语义关系，而非严格的句法关系。例如英语句子：“Mary hit Bill.”的格文法分析结果可表示为：(Hit (Agent Mary) (Dative Bill))，这种表示结构称为格表示。在格表示中，一个语句包含的名词词组和介词词组均以它们与句子中动词的关系来表示，称为格(case)。上例中的 Agent 和 Dative 均是格，而像“(Agent Mary)”这样的基本表示则称为格结构。自然语言文法本身也有格概念，例如英语中的“I”为主格，“me”为宾格，但并不像格文法那样具有健全的格定义，以至于可完全用格来分析句子。为了作区分，称自然语言文法中的格为文法格(或浅格)，而格文法中定义的格为语义格(或深格)。语义格的定义使语句的分析结果一格表示包含了大量语义信息，从而成为有效的语义分析方法。

应用格文法作语句分析的优点归纳如下：

① 语义相同但具有不同句法分析树的语句具有同样的格表示，例如被动句“Bill was hit by Mary”与上述主动句“Mary hit Bill.”有不同的句法分析树，但格表示完全相同，以指示这二个句子的语义相同，并实现多对一的源-目映射。

② 有助于删除句法分析的歧义性。例如语句“The pie baked for three hours.”和“Mother baked for three hours.”有完全相同的句法分析树结构，但语义解释却截然不同，因为 Mother 为逻辑上的主语，而 pie 则是逻辑上的宾语。这种歧义性在格文法中可以完全避免，因为分析结果产生了截然不同的格表示：

(baked (Agent Mother) (baked (Object the pie),
(Time-period three hours)) (Time-period three hours))

③ 易于使用。格表示易于以语义网络表示法描述，从而多个句子的格表示相互关联而形成大的语义网络，以便开发句子间的关系，理解多句构成的上下文，并用于回答问题。究竟要定义哪些格才合适，目前并无一致的意见，但可以列举一些常见的格：

- Agent--动作主体，
- Instrument--动作实施的工具或手段，
- Dative--动作给与的对象，
- Factitive--事件中主体使役的对象，
- Locative--事件发生的地点，
- Source--物体移动的起点，
- Goal--终点，目的地，
- Beneficiary--事件中的受益者，
- Time--事件发生的地点，
- Object--动作的施加对象。

④ 以期望驱动的推理方式实现语句分析。这实际上是一种目标驱动的逆向推理，并需做好以下三个方面的处理工作：

• 建立动词格框架，用以指出每一个动词在使用中可能涉及的语义格。例如格框架 open [-0 (I) (A)] 指示动词 open 必须关联到一个对象 (Object)，常可关联到一个工具 (Instrument) 和一个动作主体 (Agent)，尽管可以缺省。显然，依据此框架和其它有关信息，可预言包含动词 open 之语句的格结构。例如，应用 open 的格框架，易于确定以下句子的格结构：

The door opened. 1个格：0

John opened the door. 2个格：A, 0

The wind opened the door. 2个格：I, 0

John opened the door with a chisel. 3个格：A, 0, I

- 定义映射规则，用于建立句子成份和格结构间的映射。例如英语中句子的主语可以通过以下规则来选择：若 A 出现，则它就是主语；否则，若 I 出现，则它作为主语；否则主语就是 O。

该规则指示了将动作主体 (A)、工具 (I) 和动作的施加对象 (O) 作为句子主语的优先选择次序。

- 设计一个句法分析器，用于作常规的句法分析（基于文法规则），以识别句法成份，如 NP、VP、PP 等。

期望驱动的语句分析过程如下：每当接收到一个句子时，先用该分析器作句法分析；一旦动词确定，就可按动词的格框架预言可能存在的格结构（例如 open 预言格 O、I、A 的存在），再使用映射规则证实格结构的存在。句法分析器可以就是一个 ATN 分析程序；由于 ATN 分析程序能自由地构造分析结果（通过定义结果的结构模式），可以较好地支持格文法的应用。例如，可以把英语句子的 ATN 分析结果描述为表示句子格结构的语义网络，然后再基于该网络回答用户提出的问题。

17.7.7 自然语言理解—篇章理解与机器翻译

篇章分析处于自然语言理解的较高层面，它要在语句分析的基础上研究各句子间语义的关系，从而在一个更高的层面上来理解短文或整篇文章的语义。理解了短文或整篇文章的语义，计算机就能更好地对自然语言进行处理，帮助人们完成文本分类、自动文摘和机器翻译等。

机器翻译旨在让计算机自动将源语言表示的语句转变为目标语言表示的语句，主要用于书面语翻译和口语翻译。因书面语翻译只要附加语音处理，就可实现口语翻译，故人们重点考虑的是书面语翻译。尽管早期的机器翻译研究并不成功，但随着自然语言理解研究的发展，机器翻译的研究也获得了发展，并逐步走向实用。本节将首先介绍机器翻译的基本要点，接下来讨论机器翻译必须要解决的多句理解问题，最后阐述目标语言的生成。

17.7.7.1 机器翻译的基本要点

早期的机器翻译认为，借助于源语言到目标语言的词典和这两种语言文法结构的简单转换模型，就可将源语言表示的语句直接映射到目标语言表示的语句。正是由于这种朴素的观点，忽视了自然语言理解的复杂性，才导致了早期机器翻译研究工作的失败。毫无疑问，机器翻译必须建立在对语言理解的基础上，若缺乏了自然语言理解的能力，往往会使机器翻译系统面临以下一些困境：

- 词法歧义—源语言中的一个单词往往有多个语义，且每个语义在目标语言中对应于不同的单词。显然，只有在系统充分理解了该单词在源语言中的确切语义后，才有可能准确选择目标语言中的单词。

- 文法歧义—源语言中一个语句可能会有多个不同的语义，且不同的语义在目标语言中对应于不同的语法结构。显然，这些语句在翻译时，难有一一对应的关系。

- 重复引用—多个语句重复引用同一对象时的处理难题。这涉及到代名词等的翻译问题。源语言和目标语言的代名词用法往往差别较大，且无固定的对应关系；系统必须确切理解代名词在特定语言中的指代对象，才能正确选择目标语言中的代名词。

- 成语翻译—成语与特定的文化密切相关，系统在翻译时必须可识别源语言中的成语，才可以避免将其错误“直译”。“直译成语”往往会导致翻译失败，造成“笑话”。例如，有个笑话说，有人曾将汉语“他俩青梅竹马。”（意指男女青年在儿童时期一起玩耍，且亲密无间），直译为了英语句子：“They are green plum and bamboo horse.（他们是绿色的梅子和竹子做的玩具马）”了。这种“驴唇不对马嘴”的翻译，错误显然不在字面理解，而在文化的底蕴。

所以，为了实现机器翻译，一个翻译系统必须包括二个部分：源语言理解程序和目标语言生成程序。前者对源语言表示的语句应能作理解分析，并将能准确反映源语句语义的无歧义分析结果转变为翻译系统的内部表示形式；后者可将这种“内部表示形式”转变为目标语

言中的合法语句，且符合目标语言的文化传统。可以说，所有成功的机器翻译系统，都遵从了这种工作模式。

17.7.7.2 多句理解

机器翻译的对象主要是文章选段和谈话纪录，这就要求自然语言理解程序不仅能理解单一语句，也需对一组语句构成的短文作综合分析和理解。而分析和理解短文的关键之一，是在特定背景下可分解和理解这些语句间的相互关系。这些关系数量很大，并各自在特定的上下文中起重要作用。其中，一些最常见的关系有：

- 同一对象—例如，短文“小王新买了一本书，小李也喜欢它。”中，“一本书”和代名词“它”意指同一对象。此种现象称为重复引用。
- 对象的部分—例如，短文“李伟翻开刚买的新书，发现缺少目录页。”中，目录页应视为新书的一部分。
- 行动的部分—例如，短文“小陈去上海了，他乘坐的是早班车”中，乘坐早班车是去上海这一行动的一部分。
- 行动涉及的对象—例如，短文“小陈应邀参加会议，但刚出门，就发现自行车泄气了”中，自行车是参加会议这一行动涉及的对象之一。
- 因果关联—例如，短文“雨下得很大，马路低洼处积满了水，汽车过不去。”中，三个语句依次构成因果关系。
- 规划序列—例如，短文“为了出国求学，小丽决定辞去工作，全力投入外语能力的培训。”中，辞去工作和培训外语能力都是为出国求学所规划的行动步骤。

为识别语句的这些关系，显然需要相关领域的大量知识。因而，如何在知识库中高效地表示和组织这些知识，对于自然语言理解程序乃至机器翻译系统将是至关重要的。下面，我们仅从三个方面来讨论这些知识的表示和组织。

1. 焦点空间与聚焦理解

在领域知识数量较大的情况下（常常是海量），将自然语言理解程序的注意力聚焦到领域世界知识库的某一相关部分无疑是十分重要的。如此，才可快速取得有用的知识去将不同语句中涉及的对象和行动正确地联系在一起，解决多句理解时可能出现的歧义性问题。

作为例子，我们考察下面的文章选段：“接下来，固定水泵到工作台，所用的螺栓放在小塑料包中。”显然，理解这个语句组的关键在于识别小塑料包中的螺栓就是用于固定水泵的。为此，我们需要某种聚焦理解程序注意力的知识表示方式，使得一提到水泵的固定，就可直接关联到固定水泵需用的螺栓。

语义网络无疑是表示简单对象和事件的一种好的方式。有人认为，只要在语义网络表示方式的基础上加以扩充，就可支持理解的聚焦。这种扩充主要体现在焦点空间的建立和这些空间的相互关联方面。这里，焦点空间是需集中在一起加以关注的一组节点和关系弧，这些空间的相互关联则为处于不同焦点空间中的对象建立直接的语义关联。焦点空间是将语义网络划分为可以重迭的多个区域，形成所谓的分区语义网络。焦点空间可按层次相互关联。每当一焦点空间成为自然语言理解程序聚焦的对象（即成为当前关注的焦点）时，较上层焦点空间中的元素均变为可观察，从而使下层焦点空间中的元素能直接与上层焦点空间中的元素建立直接的语义关联。

2. 目标结构的应用

对上下文多个语句的理解，往往会涉及行动的目标和行动计划。例如“王维想吃冰激凌，他向行人打听就近的冷饮店在哪里。”显然，该短文的第一句涉及一个行动目标：想吃冰激凌，而第二句则为达此目标的行动计划。复杂情况下，计划通常包括若干个行动步骤，且每个步骤相应于一个子目标。

在人类的生产和生活中，许多目标常会重复出现。为加速理解过程，可以给这些目标预先定制概念化结构，简称目标结构。每个目标结构都设计为由若干顺序固定的子目标构成的

行动计划，其中的子目标又可由定制的所谓计划盒来实现。一旦理解程序识别到某个目标结构可用于当前的语句组理解，就可用该目标结构指导分析过程，并推测语句组中未直接提及的子目标。

作为例子，这里给出一个使用某个对象 x 的目标结构：

$$\text{USE}(x) = \text{D-KNOW}(\text{LOC}(x)) + \text{D-PROX}(x) + \text{D-CONTROL}(x) + \text{I-PREP}(x) + \text{DO}$$

该目标结构定制了这样的一类行动计划，为使用某对象 x 于某个目标，必须依次实现下列子目标：知道 x 的存放地点，接近 x ，取得对 x 的控制（取用）权，做好使用 x 的准备，使用 x 。前三个子目标（带前缀 D）较常用，可通过预先制定的计划盒来指导它们的实现。例如，为实现子目标 D-KNOW，可以使用相应的规划盒 ASK，其包含这样一些信息：拟执行的动作，能控制的前提条件、不能控制的后验条件（若不满足，就撤销动作计划），期望的动作结果等。

通过识别目标结构 $\text{USE}(x)$ 的可用性，一个翻译系统理解程序就可以用 USE （冰激凌）去指导对于前述短文的分析，以便加深对于该短文综合语义的理解。如此，理解程序可以回答诸如“王维为何要打听冷饮店的位置？”那样的问题了，因为 USE （冰激凌）的第一个子目标就是寻找冷饮店的位置。

3. 应用纲要和脚本

将目标结构提供通用目的计划的思想作进一步地发展，我们也可以用纲要和脚本的形式去详细描述一些情节进展过程的公共模式，再用于指导上下文理解和回答有关问题。这里，纲要用以指定一类事件涉及的各个方面，而脚本则描述情节发展过程的细节。

脚本的知识表示形式类似于框架，但更强调事件序列的描述。例如旅行和饭店用餐的通用模式，均可用脚本表示。此时，典型的上下文（语句组）理解过程如下：首先分析上下文中的各个语句，并转为内部表示形式；然后取出匹配脚的脚本，用来自上下文的信息填充脚本的未确定部分（人物、工具、特别的动作和子过程等）；在此基础上，就可回答有关问题。由于脚本能包括比上述规划盒更多、更细节和更特别的信息，从而可显示出很强的自然语言理解能力。

17.7.7.3 目标语言的生成

目标语言的生成可认为刚好与语言理解任务相反，其目标是将语言理解程序产生的关于源语言语义的内部表示形式，转变为遵从目标语言文法和习惯阐述的语句。换言之，语言生成是语言理解的逆过程。语言生成的整个过程可以划分为三个阶段：① 建立一种结构，用以表示应传送给目标语言译文的信息，这相当于决定译文将描述什么。② 应用关于目标语言译文（或对话）结构的规则去确定和构造合适的语句序列。③ 应用词法信息和句法规则生成实际的目标语言译文。

第一阶段的任务最难实现，但却与语言学最少相关。在问答系统的情况下，应传送给目标语言语句的信息取决于需回答的问题，即需回答的问题限制了信息（来自源语句的内部表示）的传送。然而在文章翻译的情况下，语言生成程序注重于文章情节的表示，不存在类似于问答系统那样的限制。为此，建立能表示相关事件序列的描述结构是至关重要的。早期的情节生成研究几乎不应用领域世界的知识，导致产生的译文高度呆板化。以后的研究工作则转而注重基于人们目标和规划的模型，去建立拟传送到目标语言译文的关于文章情节的信息结构，研究的结果表明，这些模型恰好就是为理解人们的行为和语言所需要的。

第二阶段和第三阶段的任务是在拟传送到目标语言译文的信息已确定的情况下，使这些信息的传送能有效地进行。乍一看，语言生成似乎是比语言理解容易得多的任务，因为语言理解程序必须理解源语言的所有词法、句法和语义结构，而语言生成程序只需使用目标语言词法、句法和语义结构的一个子集。但关于目标语言知识的贫乏必将导致译文的简化和呆板，

难以生成文采丰富的好文章。所以，为产生好的译文，有关语言理解面临的所有问题，也同样应在语言生成时加以解决。

例如，在多句理解中需要解决的重复引用问题也影响到语言生成。考虑下面的译文：“小李看到商店橱窗中的一辆自行车。小李想要买那辆自行车。”即使这么短的译文，由于未使用代词，也会显得很不自在。只要用代词取代已提到过的事物，就会产生自然得多的译文：“小李看到商店橱窗中的一辆自行车。他想要买它。”

但代词的应用又是有限制的：不能使译文产生歧义性。例如，某译文原先为：“小李看到小苏取下一个玩具熊放在小车旁。小李也想买这个玩具熊。”仿照上例的代词取代法，会产生：“小李看到小苏取下一个玩具熊放在小车旁。他也想买它。”可这里的“它”就有歧义性，因为“它”既可以指示熊玩具，也可指示小车。为消除歧义性，译文中的后一句应翻译为“他也想买这个玩具”

人们已经研究了多种处理方法去解决构造目标语言译文时的重复引用和其它一些问题。其中之一是应用所谓的分合技术，该技术面向语言生成过程第二阶段的任务：先将要送到目标语言译文的信息分解为许多小的单元（通常对应于子句），然后再按合适的顺序重新组合，使它们适合于构造可读的多段落译文。

语言生成过程的最后阶段是目标语言译文的实际生成。该阶段遵从第一阶段确定的需传送到译文的信息结构和第二阶段确定的语句顺序，构造合乎于目标语言文法的译文语句。语言生成的一种有效方法就是逆向应用句法分析技术。例如，可用增强的转移网络 ATN 去支持语言生成，只需简单地将关于信息传送和语句顺序的知识结构作为条件置于状态转变弧上，并用结构程序去合成目标语言的语句和译文。

机器翻译必须以自然语言理解为基础，这应该是翻译工作的起点，可惜，我们至今连这一点都还无法做到，更别说要满足翻译“信、达、雅”的要求了。语言问题不仅是一个思维的表达问题，也涉及文化、情感及美学问题。机器翻译实现的困难，也从一个方面证明了人类智能的“可贵”。在语言的创造、使用和发展方面，人类智能确实“了不起”。

17.7.8 自然语言理解未来的发展

目前，自然语言理解的研究已经取得了令人瞩目的成绩，取得了许多重要进展和重要成果；在某些方面，有的研究成果已达到了实用化的程度。然而我们也应看到，在自然语言理解的领域中虽然有众多的理论和方法提出，但研究并未取得根本性的突破【**近期，ChatGPT的成功应是一个大的突破**】。有学者甚至认为：整个自然语言理解所获得成功的例子，不过是解决了一个极为微小的“积木世界”的问题。其基本理论、基础资源、基本工具、基本方法和性能水平离实际应用的要求还有相当大的差距。自然语言理解的研究之所以进展缓慢，其原因应归结于理论基础和技术实现两大方面。尤其在理论基础方面，我们至今还未能揭示出人类语言的本质和人们习得、运用、理解人类自然语言的机制。所以，我们需要从更深的角度来对自然语言理解的本质进行探讨，特别是从认知机制来对自然语言理解的理论进行探讨，才能揭示自然语言理解的本质，掌握自然语言理解的本体和方法论，从而推动自然语言理解研究的发展。

若从学习的角度思考，人类对自然语言的理解有三个层次：第一个层次是不假思索的直接反应。学前儿童从家庭和社会的语言习得大都是通过这种方式。很多计算机程序本质上也是在模拟这个层次的自然语言理解。一个最早和最典型的系统是20世纪60年代Weizenbaum的ELIZA。目前一个应用较为广泛的系统是ALICEBOT。第二个层次是在语言的语法和语义

分析基础上的理解。一个人从小学到中学的语文课程，学习的就是这个层次的语言理解和处理能力。在这个层次的一个较早的典型的计算机程序是20世纪70年代Winograd的“SHRDLU”。第三个层次是社会环境下的人与人的语言交流。这种语言交流依赖于双方对话人的个性、文化教育背景和话语环境等因素。并不是所有的成年人都能够成功地驾驭这种交流，所以，属于这个层次的计算机系统目前并不多见。

自然语言的“机器认知”目前主要采用两种基本方法：**一是统计（归纳）方法**。基于大数定律，方法论上是“从众”，不要求真正达到理解。这无疑是初级认知方法。优点是容易操作，尤其是有了计算技术和网络技术的支持。缺点是难免有意外。**二是规则（演绎）方法**。基于理解和规则推理，是处理新问题和复杂问题不可避免的高级认知方法。优点是可信，如果规则和前提均可信，结论就可信；缺点是规则难以建立，难以完备。这两种方法性质上应是互补的，谁也不可能单独包打天下。现在的问题是如何实现统计与规则方法的有效结合。

自然语言理解（机器认知）的困难主要表现在：一方面是自然语言表达的多样性、灵活性、复杂性和动态性，另一方面是机器“理解”能力的机械性、被动性与局限性。即使是人对自然语言的理解也需要经过长时间的训练和学习，需要在大量“记忆+模仿”实践的基础上才能逐渐生长出“自主理解”的能力。让机器理解人类语言，机器理应像人一样，既要有“理解能力”，也需要“长时间”的学习。

下面，我们仅介绍一些对自然语言理解有意义的考虑。

1. 关于语言符号、语义和表象的“理解”

认知心理学认为，语言是一种特殊的社会现象，是由基本词和语法规则构成的复杂的符号系统。语言的基本结构材料是词。词是一种符号，是声音和意义之间的一种匹配，标志着客观环境中的一定事物，是能够听到的特定声音的组合和能够理解的客观事物的形态和意义的综合。也就是说，语言是符号和符号所指代的意义即语义的结合体。相似的观点存在于与语言学密切相关的符号学中。符号学认为，语言符号是“能指”和“所指”的结合体。“能指”即语音和文字，“所指”即语义，代表一定客观事物的概念。所以，我们认为，语言是一种符号，这种符号只是一种形式，在这种符号形式的背后，隐藏了它所指称的语义概念，那才是我们必须关心和理解的内容。

对于语义，认知心理学具有这样的理解，认为语义代码是长时记忆对刺激信息进行编码的主要形式，而且语义代码是分类聚集的。认知心理学认为，长时记忆中存在着两个相互独立的编码系统：语义编码系统和表象编码系统。语义编码系统主要是以语义代码对短时记忆输入的刺激信息进行编码；而表象则是感知过的事物在人脑中留下的短暂形象。表象与实际事物或景物、图像不同，它缺乏直接刺激信息的细节内容。表象编码系统通过表象代码对刺激信息进行加工处理，并把信息存储起来。

在人的学习与记忆中，这两个编码系统都是非常重要的。其中一个表象系统专门处理非语言的客体和事件的知觉信息，另一个语义系统则专门处理语言信息。因此，有人认为，表象的抽象是语义，语义的具体化则是表象，语义是对表象的归纳，体现了隐伏于表象深层次的概念。人们理解语义通常需要将其转化为表象然后进行理解。

另外一个事实也说明了表象和语义之间的关系。认知心理学通过研究认为，在人的长时记忆中存在两个或两个以上的相对独立的部分。其中两个部分是情景记忆和语义记忆。情景记忆是个体接收和存储的关于个人在特定时间内所经历的情景或事件，以及这些事件发生的时间和与相关空间相互联系的信息。在情景记忆中，包含着个体与某个特定时间和地点相

联系的个人经验。语义记忆与情景记忆相对，它涉及人们关于这个世界的知识，包括语言的、百科全书式的知识，但不包括象情景记忆的具有个人性质的东西。语义记忆一般是按照客观事物的类别或属性、总括等抽象规则对刺激信息进行组织，即由事物的类别和属性的等级方式组织起来的事实构成。塔尔文认为语义记忆是一个心理库，是一个人拥有的有关词和其他语言符号、它们的意义和指代物、它们之间的联系，以及有关规则、公式和操纵这些符号、概念和关系的有组织的知识。因此，语义记忆与情景记忆不同，它是对语词的意义、语法规则、物理定律、数学公式以及各种科学概念等抽象事物的记忆。语义记忆包含了世界所存在的事物的意义，存储着个体运用语言或知识时所需要的和所获得的信息。塔尔文把这些关于世界上的有关事物意义的信息称为语义信息。也就是说，情景记忆是个体的特殊的在其个人经历中对世界知识的理解，可以看作表象的一部分；而语义记忆则是经表象提炼出来的关于世界知识的一般概念体系。

所有这一切都说明，对语言的理解要经过语言符号-语义-表象的转换过程。语言事实上是这样一种符号，它描述了客观事物和它们之间的联系，人们利用它来进行自身内省和相互交流。这种符号之所以存在，能引起听闻者的共鸣，是因为使用者和听闻者具有共同的客观事物体验，符号的“能指”唤起了它“所指”的客观体验在人脑中的记忆，因此语言使用者实际上是通过语言这种符号将客观体验传递给了听闻者，从而引起听闻者对这种共同体验的“再体验”，于是语言才被“理解”了。机器能产生这种体验吗？这显然是“自然语言理解”研究需要回答的一个重要问题。

2. 关于概念的差异与语言符号的任意性问题

符号学认为，语言符号具有任意性。如同样是“好”，汉语中是“好”、“不错”、“行”，英语中是“good”，法语中是“bon”，德语中是“gut”。

我们也可以看到，概念的归纳并不具有一致性。各个国家或民族由于生存的自然人文环境不同，导致思维各异，使用的语言符号不同，甚或语言符号所指的概念的内涵和外延亦不相同。一个最简单、最典型的例子是关于楼层的说法，汉语中建筑物在地面上的第一层称为一楼，而在英美国家，一楼(the first floor)指的是建筑物在地面上的第二层。另一个例子是对颜色的表达，即使使用同一种语言的人在谈论颜色时，对某种颜色的范围都可能不完全一致，而使用不同语言表示某种颜色时，颜色的范围可能更不一致。人类在理解客观世界时都存在这样那样的差异和不相融合，更何况人类和机器之间的对话。

在人类的自然语言理解中存在着的这种语词符号与思维概念并不一致的现象在逻辑学里称为语词与概念之间的不对应关系，是语词所表达的概念与思维中的概念处于一种核心确定而边缘模糊的关系的表现。它是自然语言的不确定性因素之一，也是自然语言理解中存在歧义的原因之一。有研究者通过一些实例说明，语言概念体系具有某种随意性。而这种“随意性”在机器的“自然语言理解”中显然是很难捕捉的，也是很难模拟的。

语言，是心智发展程度最高的生物—人类的标志之一，能使用自然语言也即意味着具有了人类的智能。因此，自然语言理解的问题必然就涉及到了机器智能的问题，若能理解自然语言，也就说明机器已经达到了相当的智能化的程度，因此，自然语言理解问题，说到底就是一个“机器智能”问题。

3. 基于全信息的自然语言理解

语言是信息的载体；信息是语言的内核。理解自然语言的本质是理解它所载荷的信息，因此，用信息论的观点来理解语言，应是自然语言理解的一个重要方面。但是，现有的信息

论并不能解决自然语言理解的问题。钟义信认为，人的认知能力包括：观察力，理解力，目的性；观察事物的外部形式，理解事物的内在含义，判断与自身的利害关系。在给定语境（包括主体、客体、文化、环境和常识）下，语言理解要包含该语境下的语法、语义、语用三要素。只有当语境足够深广，语法、语义、语用分析才可能有唯一解。因此，自然语言的理解应基于“全信息”。

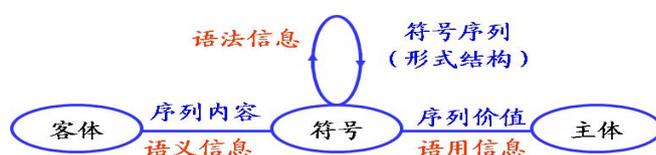


图 17.8.6 语言的全信息结构示意图

我们从幼儿识字的过程中或许对此会有所理解。比如，我们要教幼儿认识“树”这个字。要先教他怎么写（符号的笔划结构）；再告诉他“树”的真实样子（事物的形态结构），接着还要告诉他“树”的基本特征及树与草、与花的区别（语义）；再告诉他“树”的用途（语用）。明确了它的用途或害处（语用），人们才会去关注它。此后，见到“树”这个字或真实的树（语法），幼儿才会在脑海中联想到它的语义和语用；才算初步理解了这个字。这应是一个由“语法”调用“全信息”的过程。机器的自然语言理解最起码应能模拟这种认知过程。

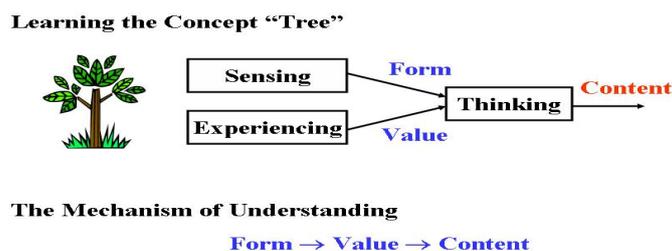


图 17.7.7 学习“树”的概念时“全面理解”的示意图

钟义信认为，基于全信息的自然语言理解存在三个相依的层次：① 了解它的语法信息——浅层理解；② 了解它的语义信息——中层理解；③ 了解它的语用信息——深层理解。从语法分析走向语义分析，是自然语言处理的重要进步；但对自然语言理解而言，仍然没有到位。目前，信息理解的研究基本上突破了纯粹语法分析的层次，正在走向语法与语义分析相结合的新阶段，但是，还少有语用要素的考虑。因此，研究还亟待深入。语法、语义、语用“三位一体”的分析，应是自然语言理解方法的最基本的要求。

人类对自然语言的理解（认知）过程既有层次性，又有灵活性。一个重要的方法[语言认知的一个基本技巧]是“快速捕获骨架”（剪枝技巧）。这包括：① 对浅显而熟悉的部分一扫而过（其实就是剪枝），只对重要而又困难的部分才做必要的分析。② “必要的分析”主要是“骨架分析”，如，篇章的骨架，段落的骨架，语句的骨架等。③ 对于人来说，其实“一扫而过”也已做了“语法-语义-语用”分析，只是因为熟悉和容易，做的极快（由下意识处理），以致自己都没有意识到。于是，一个基于全信息的自然语言理解的基本模型可如图 17.7.8 所示。

图示“基本模型”的特色是：① 目的性：明确地面向“理解”需求。② 包容性：包容了规则方法和统计方法两种规范。在规则模式下，序列是语句；在统计模式下，序列是特征集合。而且还可以包容其它基于语言行为特征的方法。③ 灵活性：整体框架是语言学风格，

包含语法分析、语义分析和语用分析；实际分析深度可依问题需求而有所不同。模块实现可以是规则的、统计的、规则统计交互互补的。④ 开放性：模型能够适应各种“理解”应用；能够随着信息表示与处理方法的进步而不断改进自己的性能。

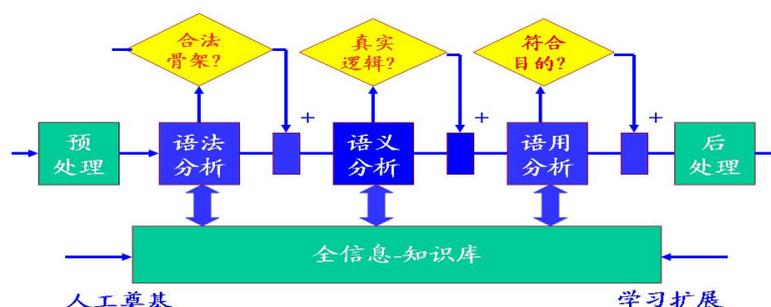


图 17.7.8 一个基于全信息的自然语言理解模型

4. 关于词计算与计算语言系统

词计算由 Zadeh 首先提出，它是语言变量概念的直接扩展。词计算提出的一个现实原因是：人类已习惯于用自然语言描述和分析事物，特别是涉及社会、政治、经济和管理中的复杂过程，我们无法排除自然语言的信息价值和作用。而另一个本质性的原因在于：感性信息或知识通常只能用自然语言来描述，由于人类分辨细节和存储信息的认知能力的内在限制，感性信息在本质上是不精确的。由于现实的原因，我们无法避开感性信息，人类最终会认识到，利用自然语言的词计算，不是可有可无的可选项，而是必需的。利用自然语言知识和信息，建立以词计算为基础的语言描述、分析、控制和综合系统，应是未来系统研究，特别是复杂系统研究的一个重要而有意义的方向。

与主要是处理基于测度的信息（Measurement-based Information）的数计算不同；词计算主要处理基于感知的信息（Perception-based Information）。进行词计算和建立计算语言系统的实质是：让机器认识语言，对语义信息可有效利用，实现人机交互。

利用词计算建立计算语言系统的一些基本原则包括：不知道原则（The don't know rationale）--不可能有足够的精度去确定所有变量或参数的值，进而用数值来描述所有语言。不需要原则（The don't need rationale）--不需要用准确的数值去描述所有语言，语言值就足够了。不想要原则（The don't want rationale）--太细节的知识反而无效。不确定原则（The can't define rationale）--无法精确定义所有概念和因果关系等。而依据这些原则建立的计算语言系统，必定是模糊系统。

一个计算语言系统首先可用于复杂系统的分析与控制。例如，在对一个复杂问题制定控制策略或规则的过程中，可用语言或词来表达问题，描述问题的情形，提出语言目标函数，制定语言控制策略，确定并实施评价程序等。而其研究的核心内容是：如何实现基于词和数的混合计算；如何将语言系统由静态描述转为动态描述和推理；如何把一个非完整的开环系统转为一个完整的闭环系统等。

有人认为，为了有效地利用信息，我们必须在硬件和软件之后，建立“知件”体系。而计算语言系统可能会发展成为有效开发、综合利用各专门领域“知件”的一种解析且可计算的通用工具。可以设想，在将来的人机交互中，特定领域的文字/数字(W/N)和数字/文字(N/W)转换器，将同 A/D 与 D/A 转换器一样重要和广泛，不同的是它们是软件而不是硬件，其性能将由其语法和语言处理的智能水平所决定。

可以认为，语言计算的基础是一从表述概念的词语到语言空间的映射和转换，从语言系统到计算空间的映射和转换。在理论上，所有自然语言系统中的概念和方法，都可以根据这一转换而直接推广到计算语言系统中。这是计算智能研究的理论基础，也是希望其可应用于实践的一个基本设想。可惜的是，除了一些简单的专用系统，即专家系统，我们目前还很难实现这种转换。计算语言系统也许是自然语言理解一个不错的辅助工具，但将一切都归结为“计算”，目前还很难“完成”。

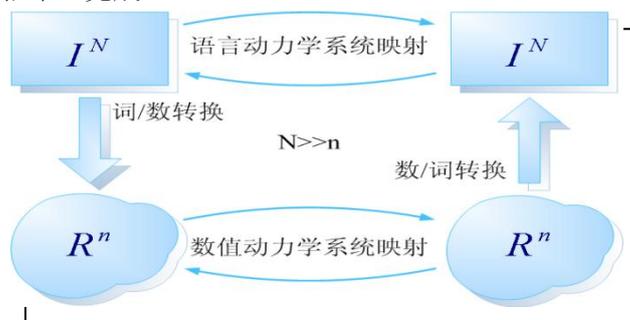


图 17.8.9 语言和数值系统的映射表示及相互关系示意图

17.8 对智能模拟和工程化研究进一步发展的展望

曾经有人问比尔·盖茨，假如让他回到创业的起点，重新办一家公司，他会选择什么？盖茨说：我会把我的精力放在人工智能上，因为那里还有很多未知的领域。从广义上讲，智能可理解为分析问题和解决问题的能力，人工指任何人工形成的，可以达到解决问题目标的方法。所以，我们可以认为，人工智能技术并不只是简单的意味着对人类分析问题和解决问题方法的模拟，而是指可以用来改善或替代人类分析问题和解决问题能力的任何合理的机器方法。

17.8.1 智能模拟和工程化的研究与发展

从工程科学的视角来看，科学技术研究的主要任务在于启迪人们：怎样利用各种资源创制先进的生产工具，扩展人类认识世界和改造世界的能力，改善和优化人类生存和发展的环境，改善和优化人的生活。从工程科学的视角看，整个科学技术进步的历史，就是一部沿着“利用外部资源、创制生产工具、扩展人的能力”方向不断向深度和广度进军的历史：古代人类凭借初等的材料科学技术，把物质资源加工成为可方便利用的材料，创造了人类社会第一代社会生产工具一只利用材料而没有动力和智能支持的人力工具，如锄头、镰刀等，便利了人的体质功能的发挥，推动了农业社会生产力的发展，创建了农业时代的文明。近代人类借助于近代材料科学技术和能量科学技术把物质和能量两种资源分别转换成为了可方便利用的材料和动力，创制了第二代社会生产工具—需要材料和动力而没有智能支持的动力工具，如机床、机车等，扩展和替代了人的体力功能，推动了工业社会生产力的发展，创建了工业时代的文明。进入现代社会以来，人类正在学习利用现代材料科学技术、能量科学技术和信息科学技术把物质、能量和信息三种资源转变成为可方便利用的材料、动力和知识，创制第三代社会生产工具—需要材料、动力和智能共同支持的智能化工具，如机器人和专家系统等，扩展和替代人的智力功能，推动着信息和智能社会生产力的发展，创建着信息时代的文明。由此可以理解，“创制智能化工具”已经成为我们时代科学技术的核心任务。

现代科学技术应当特别关注创制智能化工具的科学理论和技术，以便利用当今最现代化的工具—智能化工具—来实现工业、农业、科学技术、国防、文化教育以及其它各行各业的

现代化。只有在国民经济各个部门和社会活动各个领域广泛而充分地采用智能化的工具，才能有效地实现工业、农业、科学技术、国防、文化教育以及其它各行各业的现代化。

自从计算机诞生及人工智能理论形成以来，随着信息化、网络化和智能化日新月异的飞速发展，世界正飞速进入交互时代。人类生活的各个层面都已深刻地感受到它的影响：它不仅改变了人类的生活方式——因特系统这个全方位开放的“无主管、无国界、无警察”的巨大网络已经把整个的世界收拢在了方寸之内，地球也由于网络的联接而使地理位置变得“无关紧要”而紧缩成为了“地球村”——不管你在何处，网上成员的交流 and 沟通就如同与邻居打一个招呼那么随意平常；而且也改变了人类的思维方式——上亿人的智慧和知识几乎是同步交流，使得人类的文明正以“几何速度”在飞跃发展，而诸如“常青藤”这样的名牌大学的课程、由世界上天才人物所提出的“奇思妙想”、或诸多领军人物所做的激动人心的讲演等，也都可以在第一时间让亿万人同时分享。这正如沃尔特·里斯顿所说的那样：“芯片正在改变你我的工作方式和整个世界的运作方式”，而比尔·盖茨也认为：“信息高速公路将彻底改变我们的文化，就像活字印刷术改变了中世纪一样”。

经过半个多世纪的努力，如今，智能模拟和工程化的研究已有了蓬勃的发展。专家系统、智能决策、智能机器人、自然语言理解，以及机器学习、机器发现、机器证明等方面的成就均显示了智能技术发展的巨大威力。比如，自动推理及机器定理证明。证明定理原本是人类特殊的智能行为，不仅需要根据假设进行逻辑演绎，而且需要某些直觉技巧。机器定理证明就是把人证明定理的过程通过一套符号体系加以形式化，变成一系列能在计算机上自动实现的符号演算过程，也就是把具有智能特点的推理演绎过程机械化，用机器替代人脑去证明定理。如今，机器定理证明已经“像模像样”。它的成功说明，用机器来实现对人类脑力劳动的模拟是可行的。从本质上讲，智能机是人类运用其智能为自己大脑运行效率的增强和思维活动质量的提高所创造的一种工具或机器，它的性质就如同人类为了减轻自己的体力劳动强度而创造出来的挖掘机，为了扩大自己的活动范围和速度而创造出来的飞行器一样，它们同样都是人类智能的产物。从某种意义上说，在都是人类智能的造物这一点上，计算机和飞机在本质上是没有任何区别的，它们同样都是人类某一器官的功能的扩展和延伸。飞机是人腿功能的扩展和延伸，而计算机则是人脑功能的扩展和延伸。虽然人脑具有驾驭人腿的地位，电脑具有驾驶飞机的本领，但是，人脑在创造飞机时所花费的智力劳动与智力活动方式和在创造计算机时的情况可以说是别无二致。所不同的仅是，以蒸汽机为代表的工业革命解放的是人类的四肢，而以计算机为代表的信息革命解放的将是人类的大脑。大脑的解放在很大程度上无疑就是脑力劳动的机械化、自动化，人工智能无疑抓住了这一信息革命(实为智能革命)的关键。在为实现整个人类脑力劳动的机械化、自动化的进程中找到了一个突破口，进而也为社会生产力的进步奠定了坚实的技术基础。

如今，智能技术的应用已十分广泛，如智能控制、智能交通、智能家居、智能医疗等等。由于人的智能体现在感知、思维、行为三个层次上，因此，各种智能技术的应用也主要是要研究和解决这三个层次上的问题。在机器感知与识别方面，智能技术主要是希望计算机像人一样能感觉到气味、图像、颜色、触觉等，甚至希望计算机能感知人类所不能感知的领域。由于科学技术广泛的进步，机器感知的应用已经远远超过人类生物本身，特别是网络技术的发展，已大大扩展和延伸了人类感知的空间。在机器思维方面，智能技术主要是希望利用机器进行问题解决，具体讲就是进行计算机自动识别、计算机自动理解、计算机自动学习、计算机自动推理、计算机自动诊断、计算机自动编程、计算机自动作曲、计算机辅助绘画等等。

由于信息技术的飞速发展,人类所能感知到的信息量和计算机的运算能力均呈现出了爆炸式增长的趋势,为此,人们已研究了大量信息加工、分析的方法和规则,编制了各种软件,交由计算机处理,并提供分析结果,供决策使用。这也可算是机器思维的一部分。而更高级的感性思维和理解性推理,机器思维目前还不能做到,如能实现,将会带来人类思维的一场革命。在机器行为控制方面。计算机已可模拟、延伸和扩展人的智能行为,如语言、动作、智能监测、智能控制等。目前,机器的理性行为模拟在质量和范围方面已经超过人类,但是在感性行为方面,机器行为模拟还不尽人意。总之,智能技术的应用,已使机器在感知和行为层次,远远超过人类本身,而在思维这个智能最为关键的层次上,机器目前还仅仅在低水平的边缘徘徊,充其量只是起到辅助作用而已。不过,即使如此,计算机和智能技术的发展对人们生活的影响也是显而易见的。如今,我们已经有了能够精细观察的“机器眼”,能精细操作的“机械手”,能部分代替人的行为功能的“机器人”,以及能模拟专家在特定领域工作的电脑程序——“专家系统”。这种“专家系统”已可用于诸如选择治癌方案、预测地质发现以及在股票市场协助投资者进行投资等某些高智能性活动。计算机已经开始代替、延伸和增强人类的部分智能。

人工智能的诞生,表明人类已经认识到,单纯的数值计算并不能完全解决智能模拟问题,数据库、知识库、逻辑推理和启发式搜索等,都是必不可少的。智能模拟和工程化的研究推动着计算机信息处理技术的发展,也对社会生活和人类心理产生着重要影响。比如,它改善着人类知识能力,改善着人类的思维能力,改变着人们的思维方式与观念,改善着人类的文化生活;但也引发着人们对劳动就业的忧虑和技术失控的担忧。尽管阿西莫夫曾提出了“机器人三守则”,试图解除人们的忧虑,但智能工程化的发展对人类意识和生活的影响,仍将是深远的和不可预测的。

我们认为,人们对智能工程化的担心尽管不无道理,但在可以预见的将来肯定是多余的。因为智能模拟和工程化的研究和开发并不是一帆风顺的,在取得成功的同时也正面临着种种困难和挫折,每前进一步都困难重重。纵观智能模拟和工程化的研究和发展史,智能模拟和工程化的研究曾经经受过多次危机的考验,如上世纪六十年代纯弱法研究的危机,上世纪八十年代缺乏实用价值的危机等,都曾使其研究陷入迷茫。纯弱法研究基于这样的信念:强大的计算能力加上通用问题求解方法,可以解决一切困难问题。然而,这种脱离实际应用背景的信念无疑是缺乏现实的生命力的,于是,人们才试图利用专家系统和知识工程来解脱这一危机。知识工程的出现显示了专门知识的重要性,但也暴露了经典数理逻辑的局限性,更诱发了人们对符号主义理论的质疑。专家系统的研究并没有人们原来所期待的那样美好,人们在投入了大量的资金和人力之后,其结果却是令人失望的。我们认为,导致这一状况的原因,既有外来因素也有内部原因。外因是一部分人对智能技术作了浮夸的和不切实际的宣传,过高的期望必然使投资者对实际所取得的成果感到失望。内因则是智能技术还很不成熟。许多人认为只要让计算机获得领域专家的经验知识,就能解决常规计算程序不能解决的所有问题;其实不然。的确,基于领域专家的经验知识,可以解决许多不能以常规计算技术解决的问题,但仅凭经验知识作推理仍存在着一些严重的问题,比如:① 脆弱性和不可靠性。领域专家的经验知识往往不完善,当要解决的问题落入经验性知识可处理的范围时,系统具有专家级处理能力;但稍一超出范围,系统处理能力就急剧下降到零。这种现象被称为系统脆弱性或“知识窄台阶”问题,其使系统处理问题的范围狭小,并导致问题不能确保解决而出现的不可靠性。② 与计算机主流技术脱节。这也是智能系统难以实用化的另一个重要原因。除一

些适合于符号推理技术解决的困难问题外,大多数实际应用问题(如管理决策、工程技术等)的解决也依赖于数值计算和数据库技术,智能技术应是加强而不是取代常规软件。孤立使用的符号推理技术不利于推广应用。③ 知识获取的困难。这种困难也是许多专家系统开发者始料不及的。事实上,经验知识并不是以陈述语句的形式存在于领域专家的头脑。而从隐性知识中抽取、概括和归纳出经验知识是一种创造性劳动,其难度是很大的。

17.8.2 智能模拟和工程化研究曾经的一些热点

为推进智能模拟和工程化技术的发展,目前,人们正在进行的一些研究包括:

(1) **对问题求解方法的深入研究**。例如:① 深、浅层知识的综合应用—这里,深层知识指应用领域的专业基础知识,而浅层知识则是领域专家拥有的经验性启发式关联知识。后者知识量少,推理跨度大,可快速推进问题的解决,但具有脆弱性问题;前者知识量大,推理跨度小,推理效率低,但能解决或至少部分解决超出经验适用范围的问题。综合应用深、浅层知识可以起到相互补足的效果,增强问题求解的鲁棒性。② 问题求解知识结构化组织—其基本考虑是,不只是简单地罗列知识于知识库,而是按照知识在推理过程中所承担的角色加以结构化组织,以提高问题求解的效率。

(2) **对自动知识获取的深入研究**。按自动化的程度,知识获取可分为三种方式:人工方式—以知识工程师作为中介,从领域专家处获取知识,人工转变为适合于推理机使用的形式,存放于专家系统的知识库;自动方式—以知识自动获取系统取代知识工程师,提供基于问题求解模型的可视化交互式知识获取界面,指导领域专家自行抽取和输入知识进专家系统;机器自动学习—由机器学习程序自动从信息源(书本、数据库、知识库等)获取知识并输入进专家系统。目前,知识自动获取和机器学习已是研究的主要方向。由于学习是一切智能系统不可缺少的,在众多机器学习机制和复杂性科学的基础上,深入探索学习的本质和一般规律,建立机器学习的实用化理论,无疑也是智能技术研究的一个重要组成部分。

(3) **与计算机主流技术的综合集成研究**。智能技术的应用目前必须建立在主流的计算机硬、软件技术之上,而不能成为自成体系的孤岛型技术。与传统的计算机信息技术(数值计算、数据库应用等)的结合,除了可促进常规软件的智能化外,这种综合集成还往往会产生意想不到的惊人效果。综合集成各种新技术,如知识工程、多媒体、分布对象、网络计算、构件化软件工程、神经网络、模糊推理等,可以显著提高智能系统问题求解的能力。

(4) **对信息、知识和智能的转换机制研究**。信息世界可分为广义的信息、知识和智能三层,在不同的领域,智能的机制和原理不同,但都是在实现某种形式的信息/知识/智能转换。深入研究信息、知识和智能的关系及转化规律,建立广义的信息,知识和智能转换理论,将是实现智能工程化的重要前提。

(5) **对智能科学的逻辑基础的研究**。逻辑学派认为,逻辑是各种智能过程需要共同遵守的语法规则;各种智能过程都不过是逻辑的语义解释;逻辑无处不在,形态万千。目前,研究逻辑一般规律的泛逻辑学已经出现,未来,我们所需要的,将是进一步建立和完善可包括各种逻辑,特别是可处理矛盾、不确定性和演化的逻辑系统。

(6) **对群体智能中的自协调理论的研究**。单一领域或单一智能系统的智能总是有限的,而综合集成无疑是提高系统智能水平的最有效途径,协调则是发挥集成效果的关键所在。智能技术未来的发展也许将会从群体智能和生态平衡中吸取营养,利用复杂系统和网络拓扑的研究成果,建立自协调理论。

(7) **对生物与机器的智能交互与集成的研究**。生物有许多自然的智能功能,它们非常灵

巧和有效，获取它的成本很低；机器智能可以完全按照人的意图设计，可控制好。有人正在研究如何把两者结合起来。未来，我们或许需要在机器智能和自然智能的交互和集成方面加强研究，以建立某种生物-机器复合智能系统。这可能是解决复杂问题的有效方法之一，也可能是智能技术发展的一个突破口。

(8) **关于视觉机理与机器视觉的研究**。视觉为人提供了约 85% 的环境信息，研究视觉信息的处理机理对理解智能机制十分重要；形象思维是深入理解抽象思维和创新思维的入口，尽管它们处在不同的思维层次，未来，研究或许会发现它们的共同本质和规律。研究视觉机理不仅是研究智能机器人的需要，也是智能工程化的需要，因为智能首先需要感知功能。

(9) **自然语言级的智能系统研究**。符号演算从宏观入手研究智能的机制太过“粗糙”；神经网络从微观入手研究智能机制又过于“细微”；自然语言是思维的载体，它可反映人脑智能机制中绝大部分的运行机制，从自然语言理解层次研究智能机制，可提供比较容易切入和突破的中观手段，未来将有大的发展。

(10) **智能技术产业化模块的研究**。有人认为，随着智能技术和智能产品的广泛应用，智能产业将成为社会的主导产业。特别需要引起注意的是，“知识件”或许会从“软件”中分离出来，计算机系统将分为硬件、软件和“知件”。智能信息处理器将从一般信息处理器中分离出来，成为机器中的智能部件。于是，关于“知件”的研究已引起了不少人的兴趣。

17.8.3 智能模拟和工程化研究—未来的一些发展趋势

纵观智能模拟和工程化研究的发展历程，我们可以看出，智能模拟和工程化的研究，明显呈现出如下的一些发展趋势：

1. 从普适理论到“个性”设计的发展

当我们回顾智能模拟和工程化的研究的发展历史时，可以发现，这个历史可分成三个阶段：

一是**智能模拟的开创阶段**。近代对智能模拟的认真思考应是 60 多年前 A. Turing, N. Wiener 及 C. Shannon 等为代表的开创性工作。他们从不同角度研究了思维的机制，并被后人分别发展成为计算机科学、控制理论及信息理论。而 Turing 给出的著名的行为主义的实验，则成为机器具有智能的判别方法之一。但由于历史与技术的局限，特别是随着物理符号机制的兴起，Wiener 的控制论及 Shannon 的信息论在近代人工智能的研究中并没有得到应有的重视。

二是**智能模拟的普适理论的研究阶段**。在这个阶段，研究是以探索智能及智能模拟的普适理论为主要目标的，这些理论均如此漂亮，以致人们对人工智能充满了希望。但不幸的是，这一阶段研究的指导思想现在有些已被证明是错误的，其中，最典型的例子是：① 追求用机器完全取代人，甚至相信 10 年内可实现“电子秘书”。但是，当人工智能的研究者们提出研制“电子秘书”仅需要 10 年时间时，人类登上月球还仅仅是凡尔纳小说中的场景；现在人类早已登上了月球，而“电子秘书”的研制成功却还遥遥无期。② 假设在理想世界所得到的研究结果可以向实际世界推广。③ 忽视了控制论及信息论中的一些思想，忽视了“模拟量”在人工智能中的作用。④ “还原论”的方法统治了对智能模拟的研究，由此忽略了对智能中“个性”特征的探索。

三是**智能模拟和工程化的“个性”设计阶段**。在智能模拟普适理论的研究的同时，也有一部分研究者强调事物的“个性”在智能研究中的重要性，他们不相信仅有普适的理论和方法就可完成智能模拟的任务；相反，他们认为对“个性”的研究才是推动智能模拟和工程化

研究进展的主要途径。因此，他们更关心如何有效地为“个性”进行设计。

综上所述，我们可以看出，智能模拟和工程化的研究和发展，可以认为是从研究思维的本质开始，在技术条件的约束下不断地批判与总结的过程。而这个过程又分成**探索普适理论及发展“个性”设计**两个阶段，当人们认识到“个性”在思维中的作用时，智能模拟和工程化的研究者对研究进行了各种各样的总结及分析，并开始取优舍劣，为形成综合集成的研究做各种努力。

2. 从还原论到进化论的发展

在智能模拟和工程化研究的进程中，从研制自主系统(Autonomous system)到对人类思维的探索，均是以“还原论”作为其研究的主要方法论的。但现实告诉我们，物理学研究所推崇的“还原论”方法对自然系统的研究是有效的，却不能作为整个智能研究的方法论。尽管这并不意味着智能模拟和工程化的研究不需要对它的各个局部进行还原式的研究，如果没有对人各个局部(包括将各局部的综合集成)的深入探求，其结果将会与智能模拟和工程化研究开始时片面追求统一的基础原理的方法论一样，会导致其走进“死胡同”；但是，单纯的“还原论”方法确实不能完全解决智能的模拟问题。事实上，自 Feigenbaum 提出知识工程的概念之后，在一个系统中，因问题需要而同时使用多种模型已不是什么稀奇的事。智能模拟和工程化的研究者已为研制大型智能系统提出了大量的方案，但它们在方法论上却没有重大区别，因而没有出现重大突破。于是，一些研究者已经开始考虑离开传统研究的“还原论”之路，他们已经在考虑为智能模拟和工程化研究开凿一条基于“进化论”的新途径。比如，Minsky 认为，“我们应该从生物学而不是物理学受到启示，……”进而，他提出了“从上到下”与“自下而上”两种策略的综合集成方法。Simon 则提出，“在计算机科学中，我们永远不会有牛顿运动三定律”。也就是说，他们均不相信存在着象物理学那样简洁的普适理论，而相反认为，智能模拟和工程化的理论一定是丰富多彩的、各司其职的且互相协作的多种理论的集合。它们将包括还原论、整体论和进化论的集成等。

3. 从单纯的知识“表诀”到重视“信念”与“感悟”

知识与智能在常识意义上是“知识越多，系统越有智能”。但在人工智能的研究中往往又可以听到现在的系统是“知识越多，系统越慢(笨)”的说法。除了搜索的原因之外，还有什么原因？按照柏拉图的说法：所有知识都必须可用任何人都能用的清晰定义来描述。如果某人不能用这种清晰的指令来表述他的技能—如果他“知其然而不知其所以然”，那就不是知识，而不过是“信念”。我们可以把这段话理解为：“信念”是定性的、非精确的，它属于“非理性范畴”；而知识则是可定量的，可用“清晰的指令来表述”的，它属于“理性范畴”。但在智能系统中，真正决定其成功与否，真正使其表现出某种智能行为的，恰恰是这种“信念”。但物理符号方法要求这种非确定性及例外“必须能以清晰的指令来表示”，这也许是知识越多，系统越慢的另一个更本质的原因。

为了解决这个矛盾，重新认识系统的智能行为是必要的。Minsky 认为：“我们的心智(mind)包含了能使我们解决认为困难问题的过程，而智能(intelligence)则是我们仍不了解的那些过程的统称”。其言外之意是：心智包含在“知其然”中，而智能却表现在处理“知其然而不知其所以然”的过程中。“信念”没有独立的结构，它的结构甚至它的存在，均依赖于与其有关的知识与个人感悟。当知识与感受消失时，这些“信念”也随之失去其存在的意义。因此孤立地谈“信念”的描述是没有意义的。“信念”是对实际世界的各种感悟与描述(包括陈述及过程)综合集成的结果。由此可以认为：智能行为表现在“信念”之中，而“信念”

是个体知识与个人感悟综合集成的产物。知识固然在表现智能行为上是十分重要的，但对“信念”的研究则涉及到人工智能研究的成败。如果说知识的描述与 Minsky 的“从上到下”策略有关，则“信念”的描述将取决于与“自下而上”策略的研究结果。因此 Minsky 的“自下而上”是一类更复杂的问题。智能存在着精确处理与不精确处理两重特性。智能行为恰恰是不精确处理与精确处理两者综合集成的结果。

4. 从联结机制和物理符号机制到功能的系统综合

尽管联结机制与物理符号机制的基本原则在上个世纪六十年代就已提出，如今已是智能模拟和工程化的主要基础理论和方法，但它们仍无法处理所有智能问题。基于结构主义的联结机制认为，智能的寓所在大脑皮层，它可以由大量非线性神经元互联而成的并行信息处理的神经网络来模拟，它在模式识别、故障诊断、联想记忆、组合优化问题求解等许多方面已获得了成功。但问题是，生物神经元和神经网络的结构和功能都很复杂，完全的技术实现几乎遥不可及；而简化的神经元和简化的神经网络可以实现，但其智能水平也被简化掉了。基于功能主义的物理符号机制认为，智能系统可以理解为某种符号逻辑运算系统，它可以以计算机为平台编制“聪明程序”来实现，其典型的成就就是建筑在知识工程基础上的专家系统，已在许多领域（医疗、咨询、培训、预测等）成功的解决一些应用问题。但问题是：求解问题的“聪明程序”需要知识，而知识需要表达和获取，专家知识目前很难“充分”挖掘，特别是隐性知识，而显式表达也不容易。还有生态主义理论，希望从生态进化的观点考察智能，并认为实现智能系统的最直接的方法是仿造系统的“模式-动作”关系：识别输入模式，产生相关动作；无需知识表达与推理。但除了一些简单的示例，目前还没有哪一个复杂系统是完全基于行为主义开发的。于是，结构主义面临“结构复杂性”的困难；功能主义面临“知识获取”的瓶颈；生态主义面临“智能局限性”的难题。这种情形表明，现有理论还不到位，需要革新。方法之一就是综合集成。

讨论各种理论的综合集成是基于对它们优缺点的分析。物理符号机制的优点是可以做推理，缺点则是只能做“准精确”的符号推理。隶属于它的任何非确定性理论，均是以事先对局部评价可精确化为条件的。例如，贝叶斯决策需事先确定事件的先验概率。尽管这个机制可以完成非常复杂、只有受过长期严格训练的人才能完成的任务，但却在动物依赖本能也能完成的行为上束手无策，即对“感悟”的描述及使用是无能为力的。这个缺点是非常本质的，因为它事实上与搜索中的“组合爆炸”及不确定（常识）知识的获取等关键问题直接有关。于是，人们试图利用联接机制来解决这个问题。联接机制的优点是它可以采用极简单的结构，表示非确定性世界中的实体。但它的弱点是：就表示而言，其结构不够灵活；过程不可定量描述；学习时将可能遇到计算复杂性问题；在显示其智能行为的方式中，可模拟人类所采用的“举例说明”作为对“只可意会”推理的解释，但它离可以真正实用却还有很长的一段距离。作为一种尝试，我们或许可以利用物理符号机制与联接机制的互补特性。比如，我们可以将基于心理学激励-反应模型的逻辑表示加以扩展使其包括联接机制的联想记忆，由此建立一个包括联接机制和联想计算的逻辑体系；也可以设想将隐含存贮于联接网络中的某些知识在一定条件下抽象成一些显现的规则（即理性的知识）。智能系统不同实现方法的综合集成，其中心思想是可将不精确处理与精确处理结合起来，并采用恰当的方法去处理不同的问题。我们所提出的“模糊神经网络”和“因素神经网络”模型，就是属于此类的一种探索性尝试。

5. 从静态知识到动态知识，从封闭系统到开放系统，从单纯自省到重视反馈

A. Turing 和 N. Wiener 关于智能行为的设想，原本是从不同角度，利用不同的方法对实

现这种智能的途径的探讨，他们的思想现已成为计算机科学和控制理论的基本观点。当我们将他们的思想进行仔细分析之后就会发现，计算机科学的基本操作之一是对离散信息的搜索，而控制理论则使用了基于连续量的反馈操作。由于计算机科学的蓬勃发展以及连续量在计算机上表示的困难，使人工智能基本上没有认真考虑控制论的思想对描述智能行为的意义。智能行为来源于反馈还是来源于搜索？还是来源于反馈与搜索的综合集成？我们从中似乎看到了对机器模拟智能行为的两个极端认知：一个是 Simon 的心理实验所发现的**搜索**；另一个为 Brooks “没有推理” 的智能所采用的**反馈**。对搜索的批评已为许多研究者所讨论，其关键问题是对不精确事物和过程描述的困难及组合爆炸。而对 Brooks 工作的批评也像对他的研究赞扬一样引人注目：Kirsh 不相信 Brooks 的工作可以进化到理性的思维，他的批评文章直接以“今天的虫，明天的人？”为题，没有人会相信简单的行为模拟可以推广到去研究认知问题；Nilsson 针对 Brooks “不需要推理” 的说法引用了 W. Anden 的诗句：“不会推理的将在行动时消亡，而不会行动的则因此而消亡”；不过，在对感知描述使用“显现方式”无效时，谈基于逻辑的推理确实是无米之炊；Mitchell 则认为 Brooks 的观点更适合建造“恒温器”，而不是建造智能系统；但知识获取的过程与工人学习修理收音机的过程确有相似之处，Brooks 利用反馈将系统放入实际环境及将人考虑为被综合集成的对象来获得知识应该说有一定道理，它的重要性主要表现在可以因此而回避搜索所遇到的一些问题。但鉴于 Brooks 工作对环境表示的局限性，也使得我们不得不考虑这个**基于反馈控制的操作**是否可以与**搜索操作**综合集成呢？是否能在机器智能的两个极端方法的中间找到一个合理的可以取长补短的、新的机器智能模型呢？回答这些问题是非常复杂的，但将各种方法甚至“人”也综合集成到系统之中，将是一种不可避免的选择和途径。

6. 从人-机独立到人-机集成

一般地说，计算机是一个能既快又好地精确处理(定量处理)信息的系统，但它处理不精确信息(定性处理)的能力却是有限的。而人处理精确信息的能力是有限的，但人对非精确信息的处理则是非常高明的。人和计算机可在体系层次上表现出智能行为的两个方面。因此，将人-机进行“综合集成”，至少在目前，应是智能系统提高智能水平的一个值得考虑的最佳方案。

17.8.4 智能模拟和工程化研究未来的发展

随着信息技术和网络技术的快速发展，人类社会正在走向信息时代并导致一场新的工业革命。如果说上一次工业革命是通过机械解放人的体力劳动，则这一次新的工业革命将通过信息化，特别是管理和决策的自动化，解放人的脑力劳动。人将从繁重的脑力劳动中解脱出来，从事需用更高智能的工作和提高自己的文化修养生活。毫无疑问，智能的工程化技术将在这场工业革命中起到举足轻重的作用。

智能工程化的目标，首先是要在机器上模拟人类智能。要在对智能进行统一解释的基础上，建立一个较为完善的思维模拟模型，以便能对智能工程系统的开发进行有效的指导；其次是要“拓展”人类的智能。或者说，其近期目标是开发各类“智能机”，用以代替人类从事某些脑力劳动，即让现有的计算机更聪明更有用；而其远期目标，则是在探究人类智能和机器智能的基本原理的基础上，研究用自动机全面替代人类的日常思维过程。

由于人工智能是人类智能和大脑功能的一种延伸，因此，人工智能的作用和功能也就主要体现在它扩大了人脑的运行效率和活动质量上，而不是体现在智能水平的提高方面。不管计算机在计算能力上有多强，在运行速度上有多快，都不能改变其仿人类智能的形态或超出

人类大脑的思维模式去工作，甚至在仿照人类大脑的运行方式上也毫无改善。一般地讲，计算机能发挥身手的地方大多是那些要求严格遵循定义和逻辑规则去做的工作。如此，我们就会深刻地感受到，“智能机”只是一种可替代人类从事某些智能性工作的机器人和工具，是人类智能的一种延伸，人工智能和人类智能的性质是有着根本性的差别的。

然而，我们也认为，智能工程化研究的重大意义决不只是这一点—替代人类的部分脑力劳动，把人类从繁杂的脑力劳动中解放出来；智能工程化研究的更大意义在于对人类智能的深刻理解和积极影响，尤其是对未来人类智能进化方式的改变，亦即，智能工程化研究的发展或将促使人类智能的进化方式发生革命性的改变。

一种观点认为，从广义进化论的角度讲，人类进化与发展的基本形态是：生物进化(自然社会)—文明进化(文明社会)—智能进化(智能社会)—精神进化(精神社会)。这种提法或许值得商榷，但其所表达的某种主导思想却是我们所赞同的。

人类进化与发展的第一个进化形态是隶属于**自然进化的生物进化形态**。处于人类文明之前的人类进化，主要是作为生物的人的生物性进化，其意识性和社会性进化则是生物进化的附属产物或人类进化的非主要特征。根据人类学的研究，人起源于大约 500 万年至 700 万年以前一种已经能够两足直立行走的古猿，在经过了约 200 万年的原始自然进化过程后，最终才完成了人的起源与生物学进化。在这一进化过程中，自然选择在人的起源及其生物学进化中发挥着最重要的作用。

人类进化与发展的第二个进化形态是**基于认知和文化发展的文明进化形态**。从广义进化论的角度看，在经历了漫长的生物自然进化阶段之后，人类便进入了另一个更高层次上的进化阶段，也即文明(文化)的进化阶段。这时，人的生物学意义上的进化相对稳定和缓慢，其显著标志是文化、知识和意识的迅速积累和人类文明的迅速发展，科学技术的发展是其中最突出的一点。文明进化是人类为了能更好地适应自然社会，成为这个世界上最强大、最优秀的物种而不可或缺的手段，也是生物进化的一种继续。这个阶段，也许应该从人类有了语言和文字开始，包括着初级社会(原始社会)、农业社会、工业社会和初级的信息社会，但它们只有进化程度(深度、广度)的不同，而没有本质上的区别，即都是文明的进化而非生物的进化。然而，就是这几千年的文明、文化进化，构成了人类下一步新的进化的基础。这里所说的人类下一步的“进化”，指的就是智能的进化。

我们赞同人类进化与发展的第三个进化形态应是**智能进化形态**的提法，认为它应是以改变和拓展人类自然智能为主的一种进化，也是人类得以进入智能社会的一个重要标志。当然，智能进化也并非智能社会所独有的，在原始社会和文明社会中同样伴随着人的智能的进化。但智能社会中的智能进化是不同于原始社会和文明社会中的智能进化。原始社会里人的智能进化主要是一种纯自然的生物进化，表现为手脚的分工、脑重量的增加、脑结构的完善等。到了文明社会，人的那种纯自然的生物型智能进化基本完成，取而代之的是基于人类原态基质的文明进化、文化进化，主要表现在各种科学技术和文化的形成与发展，以及新型的生产力和生产工具的不断发展，农业文明尤其是工业文明是人类文明进化的杰出代表。这时，人的智能是处在一种较为缓慢的准自然进化过程中，这时的智能进化是在受到各种人为因素(如文化)的影响下的生物进化，是一种“半文化半生物型”的进化。还应该指出的是，这时，各种人为因素的影响，对于进化主体来说是不自觉的、被动的，甚至是无意识的。只是到了智能社会，人类智能进化的历程才翻开了新的一页，智能技术将是智能社会中促进人类智能进化的最重要的技术手段，它将影响并将促进着人类智能进化的进程。

在智能社会，人类智能的进化将是一种全新意义上的进化方式。一方面，信息的指数式增长、社会竞争的日益激烈、进化压力的不断加大，使人们对智能进化产生了前所未有的要求；由于人的生物进化速率相对缓慢和稳定，迫使人们必然会采取一种技术性的手段来帮助自己加速智能进化的进程。而另一方面，智能技术的发展，又为人们实现这一愿望提供了现实的基础；于是，基于智能技术的人对自己智能的拓展和改造将是不可避免的。

在智能社会，人为因素对人的智能进化的影响将是自觉的、主动的，人们将迫不及待地利用一切可以利用的技术手段(生物的、医学的、工具的等等)来提高自己的智能，从而加速了智能的进化程度。这时的智能进化已经不是自然生物意义上的进化，而是一种文化技术型的进化，也即是一种可以按照人的意志进行设计和控制的进化。这种文化技术型进化的结果之一便是可产生一种“人-机”混合型的“全新人类”，比如，人脑与“电脑”可以充分地双向自由联接，实现“人-机合一”。当然，那时的“电脑”也许已不是现代意义上的电脑，它将是一种全生物型的“电脑”，并且是可植入人体内的“电脑”。这也就是说，那时的网络终端可能就是大脑本身。我们相信，脑科学、神经科学、纳米科学以及各种全新的信息科学(如 DNA 计算机、量子计算机等)的有机结合，将会创造出一种全新的人类、全新的社会。

基于智能技术的智能进化不仅拓展着人类的智能，也将改变着人的生物进化本身。比如，有人认为，人类应通过基因技术修改和完善自身的 DNA，从而确保对飞速发展的“电脑”的智力优势，以遏止可能出现的智能机器人“统治世界”的局面出现。我们认为，鉴于“电脑”的发展速度是如此迅速，而人类智能的生物发展却越来越“不适应”，人类改变自身 DNA 的结构，也许会未来的一种选择。这将是人类进化在更高层次上的展开，是人类智能进化对纯自然进化或纯生物进化方式的突破。它预示着人类进化尤其是智能进化将以一种新的形式出现，从生物进化时代到文明进化时代，再真正发展到智能进化时代。

而在智能进化时代之后，也许会出现一个更高层次的“**精神进化**”时代，但这是一个对于我们还完全陌生的世界，在此，我们还无法对其做出具体的描述。不过，我们赞同在智能进化之后应有一个精神进化的形态的提法。这将是人类进化与发展的第四个进化形态，也将是其终极的进化。当物质不再是人类最迫切的要求，当生物的肉体已不再是人类的“桎梏”的时候，人类所追求的应是一种“生命的永恒”，即精神的永恒。这将是一种心灵和精神的进化。人类精神的进化会将其“心灵”“寄托”于何处，我们还不得而知，但我们相信，人终将会摆脱一切束缚，实现完全的解放，包括自己的“心灵”。这不会是另一种形式的“乌托邦”，而是人类生物、技术、文明和精神进化的必然归宿，是人类智能高度发展后的一种自我“升华”。也许那时的“人类”已不再是我们当前意义上的人类，但它最起码是人类文明的传承。

当今的智能革命无疑是对人类脑力劳动的解放，并且这种解放是以对大脑部分功能的替代为基本方式的，尽管这是一次进步，甚至是意义重大的进步，但它要付出的代价，却决不可能是人类智能的退化，或将是人类大脑被逐步的替代。智能革命所开发的高度智能化的智能系统，只会更进一步提高人类的智能。而发展一种智能控制技术—将人的大脑同“电脑”直接相连，让具有高度智能的人与高智能的“机器”共同协作，让机器智能对人类智能的发展做出贡献，应是人类智能发展的一个积极而必然的选择，它会对人类智能的发展起积极的促进作用。

当今信息技术的发展速度确实让人难以想象，到了明天，我们会不会觉得，今天的“电脑”看起来就像早年的机械式计算器一样迟钝和落后？目前，计算机已能以人类远远不能企

及的速度和准确性，完成原本是属于人类思维领域的大量任务。但是，如果有朝一日计算机在我们自以为优越的重要品质——智能上也超过了人类，那时，人类还会有自己的优势吗？因此，我们断定，人类首先会用智能技术来提升自己，人类的智能也一定会随着智能技术的发展而发展。

我们相信，人工智能会逐步向人类智能逼近。人类智能与机器智能的区别将越来越多地来自于人类智能的自我设计，而机器智能则能使人类智能得到越来越明显的提高。无疑，人工智能的目标是用机器尽可能地模拟人的活动，并且或许会最终改善并超出人的能力。它在机器推理、专家系统、心理学和精神等方面的成果已向世人展示了它的重要意义。尽管人工智能在目前还无法接近人脑甚至绝大多数其他动物的头脑，但是，这并不妨碍它对人类智能的模拟。许多人都相信，当一个系统的复杂性达到某一“临界”状态时，就会呈现或涌现出某种新的品质。那么，这一天离我们还会有多远呢？

如今，智能模拟和工程化已越来越成为一个热门的研究课题。乐观者有理由相信人类社会在数字化、信息化之后将会进入智能化时代。从政治上讲，“落后就会被动挨打”；从经济上讲，“科技是经济发展的第一生产力”。因此，各个国家必然会对智能技术加紧研究。人类迄今的发展进步都是我们大脑的功劳，而制造具有高度智能的“电脑”，也许是我们人类大脑重要的使命，但我们不相信它会是“最终的使命”。

但是，目前，对人工智能抱盲目的乐观态度也是不可取的。由于智能的模拟和工程化是一项复杂的社会工程，其发展受制于众多学科（如脑科学、心理学、信息科学、智能科学等）和相关技术的发展。因此，智能的模拟和工程化的发展历程必定是一个漫长而曲折的历程。

很多科学家锲而不舍地想使计算机能够思考。但是，计算机技术的高速发展并未导致电脑在智能化思维方面有突破性的进展，原因有多个方面。一方面，广义地讲，人工智能是对人类思维的模拟。由于人类目前还没有充分了解大脑的本质属性，因此，制造能够真正像人一样思维的机器也就非常困难。于是，探讨智能的本质和运行机制将是人工智能研究所面临的最迫切的任务，也是人类认识自身的“最后方式”。它不仅将引起技术方面的革命，而且会对我们的精神世界提出挑战。人们对智能有不同的解释是无可厚非的，但是搞人工智能就必须有对智能的准确理解。另一方面，还由于人工智能技术并非是单纯的技术问题。过去，我们所犯的最愚蠢错误之一，就是试图把智能过程数字化或者试图为其建立精确的数学模型。智能本质上是一类可解决“感性问题”的能力。所谓感性就是个别的、特殊的、随机的、模糊的、感官的、情绪化的、个人意志的。而现在电脑的信息处理本质上是纯理性的。采用纯理性的方式去解决感性的问题是很难成功的。因此，我们也许需要使电脑感性化。电脑感性化是指电脑具有像人一样的大脑功能结构，其智能过程是自主的和感性的。对于电脑而言，其智能化就是必须具有优化、扩展和改变主体已有程序和创建新程序的功能，即具有我们过去常说的主观能动性。智能可以利用一般经验或理论解决特殊问题，也可以归纳总结个别的经验使之上升到普遍的理论。目前的普通“电脑”尚不具有积累经验即学习的能力。虽然一些电脑程序中有自动记忆人机交互过程的能力，也只不过是使用者来延续该电脑程序的编制而已，并非出自于电脑的智能。因为这个过程不能突破该程序的功能和目标。不过我们可以把电脑程序的这种功能看作是初级的智能。可以认为，电脑在工作过程中依赖人—机交互的程度直接反映了它的智能化水平。

智能工程化的研究和发展需要理论和实践。理论方面要求有新的思维和行为模型；实践方面要求有构造与实现智能系统的理论和技术。目前，智能工程化的理论和技术还很不完善，

它需要更新的理论框架，以克服目前宏观与微观隔离、全局与局部割裂、理论和实际脱节的局面；需要更好的技术集成，以集成各种智能技术和其它信息处理技术，以综合集成认知科学、心理学、社会学、语言学、系统学和哲学的成果等；需要更成熟的应用方法，以研究出通用而有效的智能系统开发方法，更高级的通用语言，更有效的专用语言和开发环境等。对于未来的研究，我们认为，以下研究将是十分必要的。

(1) 深入开展智能科学的研究，搞清智能的本质，建立完善的智能科学理论。

模拟智能首先要了解智能，深入开展智能科学的研究，搞清智能的本质，建立完善的智能科学理论，是智能模拟和工程化的前提和理论基础。

深入开展智能科学的研究，首先是要深入开展思维科学的研究。人类之所以具有智能，是因为人类会思考问题，具有思维能力，所以，智能的基础研究课题应该是思维问题。思维的最早出现并不依赖于语言，早在动物阶段就已经出现了低级的动物思维，其本质应是初级的形象思维。这就是说，形象思维是思维的初始形态，是抽象思维的基础，形象思维先于抽象思维。形象思维是感性的、非语言性的；所以，形式系统不能模拟人的所有思维。要想使计算机具有人的智能，必须使计算机具有形象思维能力。思维科学应该研究思维活动的更深入更具体的规律，提出新的概念、新的方法和新的机制，比现有信息处理系统和图灵机能更广泛、更深刻地描述思维的功能，并把这些理论与特定的机器模型联系起来，以期最终得以工程实现。

深入开展智能科学的研究，更需要深入研究人类的认知过程。比如，人类认知的概念框架作为人类意向性产生的前提，它究竟是由哪些因素构成的？因为人类对于外部世界的认识绝大部分限于我们已有知识的范围之内，专家系统和知识工程就是在对人类概念框架的认识的基础上所形成的模拟。

深入开展智能科学的研究，还有必要研究人类的动机、情绪的产生机制，研究它们与人的生理、心理的关系，与周围环境因素的关系；有必要研究人类认识的逻辑化、自然化与日常化的关系，研究人类在交往过程中是如何运用知识的，特别是要研究人类日常认识的语境因素以及这些语境之间的不自相矛盾性等。有必要研究人的思想与行为之间的关系，研究思想在人的行为中究竟起着什么样的作用，人又可以从行为中得到些什么。有必要研究人类认识的联想以及控制系统的产生，以使人工智能能模拟人的联想与控制机制。有必要研究人类知识是如何解决问题、如何成长与丰富起来的进化问题，因为一个智能人只有在能够解决问题时，才能显现出他的能力来，也只有不断地丰富与发展自己的知识才能提高自己的智能，使自己的智能得以进化。

总之，为了实现人工智能，人工智能研究应从以定性研究为主的实验性学科向以定量研究为主的理论性学科转变；从点上的孤立研究为主向面上的综合研究为主转变；从主要研究确定性问题向主要研究演化类问题转变；从以线性演绎为主向以非线性归纳为主转变；从机械的因果决定论向辩证的系统演化论转变。现在，人工智能学科正处于发生深刻质变的转型期，我们面临着难得的创新机遇！

(2) 多学科联手与多机制融合

人工智能是一门涉及哲学、心理学、生理学、认知科学、智能科学、语言学、自动控制、计算机科学等诸多学科的边缘科学，仅仅依靠计算机学科的孤军奋战是不可能实现真正的人工智能的。广义的人工智能也并不局限于计算机，计算机只是实现人工智能的途径之一。事实上，人工智能的每一步发展都与其它学科有关。比如形式化方法来自数理逻辑学，联结主

义来自于神经生理学。如果我们把眼光放得更开些,是不是可以得到一些更好的办法呢?比如,认识论能否对人工智能做出贡献,关键也在于哲学家们必须从富于哲理的安乐椅上站起来,进入实验室,并与神经科学、脑科学、认知心理学、人工智能的专家们一起,共同进行协作研究。如此,认识论的研究与人工智能的研究才可互相依赖并同时成长。

智能科学和技术的形成需要多学科的交叉融和;智能应用系统的实现也需要学科内外多种实现机制的大综合。结构主义、功能主义和行为主义都只是从某一个侧面观察和模拟智能,难以完全触及它的本质;而智能系统的完美实现则需要真正揭示智能生成机制的本质。为了避免复杂问题环境下的“盲人摸象”,必须重视各种技术的综合研究和运用。

(3) 开展新的智能信息处理系统平台的研究

目前,可实现智能化信息处理的系统平台是电脑,目前电脑的工作和记忆都是通过高低电位和逻辑门实现的。电脑的最基本构成是处理器、内存和总线结构,它们只能对电路的开关做出反应和发生作用。这种结构可以看作是电脑的思维活动平台,它也限定了电脑的“思维”方式。毫无疑问,结构如此简单的电脑是绝对不可能实现人脑的思维方式的,它所能支撑的软件-即“思想”也只能是有限的。且不说这种电脑不存在意识,没有心理平衡问题,无法建立主体价值观,不能自动对所有的感受进行过滤以便处理有用和必要的事情。人脑也绝不会是如此单纯地只是处理0和1的装置,它的知识存储和运用也不仅是经验块堆的建立、组合和联系。如果要让电脑实现人脑的功能,它必须在结构和工作机理上彻底翻新。

人们所考虑的电脑的发展方向,一是使用生物芯片,二是使用量子器件。它们的工作原理与目前的电脑有本质不同。生物芯片的原理是在分子水平上的,这与生物学水平是一致的。而量子器件的原理则是在更微观的原子、光子或更深层次的水平上的。我们假设人脑的工作机理可以在生物学水平上得到解释或在量子水平上得到解释,那么,在这种水平上制造的计算机就有可能达到人脑的水平。

根据推测,未来光子计算机的运算速度可能比今天的超级计算机快1千到1万倍。而一台具有5千个左右量子位的量子计算机可以在大约30秒内解决传统超级计算机需要100亿年才能解决的素数问题。相对而言,生物计算机研究更加现实,据说,1克DNA所能存储的信息量可与1万亿张CD光盘相当。如果未来上述技术能够成熟运用,那将对人工智能的发展起到决定性的作用。可以期望,一旦DNA计算机全面实现,那么,真正的“人-机合一”也就会开始实施。到那时,人们最不需要的可能就是现在的电脑,因为大脑本身就是一台自然的DNA计算机,人们真正需要的只是一个接口。DNA计算机蕴涵的理念不仅可以使计算的方式产生进化,而且可以使人类的大脑、思维产生进化。

(4) 关于人-机结合和人-机融合的研究

智能程序集中的是人的经验的总结,其本质是理性的。所谓理性就是理论的、有序的、精确的、数字化的、结论性的、有规律的、普遍性的、公共的、合乎逻辑的。简单类比就如同是一本操作手册,人们只要照它去做就是了。显然,人的智能不是被用来解决这类理性化的问题的。因为当一切都规定好了、程序化了,就根本不需要智能了。由此我们得出一个结论:无论电脑的功能有多么强大,只要它只能按给定的程序来工作,它就不能算作具有真正意义上的智能。

人有智能,因为人是感性和理性的矛盾统一体。未来,电脑可以让我们的社会数字化,但是,就目前技术水平而言,我们却很难让电脑感性化。电脑距离我们的人脑,似乎还有着遥远的距离,中间似乎还隔绝着许多不可逾越的鸿沟。用电脑完全替代人脑,目前,确实是一个难题。

不过,人们研究人工智能的根本意义,并不是要制造一个智能机器来取代人类思维,人

们研究人工智能的目的，主要是为了更好地为人类的生产、生活服务，正如机器延伸了人的体力，人工智能主要是为了延伸人的智力。因此，最大限度地发挥计算机的潜力，让计算机减轻人的脑力劳动，才是人工智能研究目前最主要的目的。人与计算机各有优势，因此，人工智能发展的最佳模式应是人-机结合。例如对于 NP 问题，如果结合人类智能和计算机的搜索能力，特别是借助人脑在分析与待解问题相关的特定信息中所得到的启发式策略来确定搜索的方向（这种启发式策略主要来自于人类社会的经验和法则以及大自然的运行规律），将会获得较为满意的结果。计算机在节省令人厌倦的脑力劳动方面肯定是有帮助的，它可替代我们进行繁杂的计算。还有，当我们必须在限定的时间完成某项复杂的计算任务时，计算的速度也是关键的。人-机结合可以最大限度地发挥人和计算机的优势。

近年来，一些研究者把视线从“认知模拟”转向“人-机合一”的研究。既然人脑中进行的思维难以用电脑来复制，那就干脆设法将其直接引入电脑中，作为一种控制环节好了。这一思想的转变又一次鼓舞了人工智能研究的专家们，似乎又看到了新的曙光，但要真正实现，却远非容易。有人认为，用机器来模拟人类智能，这个总的指导方针被称为“认知模拟”，如今的“人-机合一”，只不过是模拟不成便索性在机器中引入认知，但是，目前，其研究者所引入的，只不过是猴脑对肢体的控制、反射，并非认知；人脑能控制肢体运动，这仅仅是人类作为动物的本能，决非作为智慧动物的人类的智能。将来如何发展，是否真的能将人类的认知引入？即使真的引入后又将如何？现在就下结论还为时尚早。我们认为，其实“模拟”也罢，“引入”也罢，万变不离其宗，始终是针对人类的智能拓展在作千方百计的努力。

人类智能的根本特征是能从已知去得出新知，可从事认知和创新。人类个体的知识来源途径有三：亲身感知、他人告知、逻辑推知。人脑的工作分为无意识和有意识两大部分。无意识就是不知不觉的大脑活动，而意识就是经过记忆和思维的过程。在无意识中包括了本能，它是先天具有的，而另一部分无意识则是后天形成的。人脑的一个显著特性就是“不愿”有意识地做重复的事情。它会自发地把重复性的工作和经验转交给无意识支配，以腾出精力处理新鲜的事情。如果这种无意识支配的工作继续重复下去，人脑会自发地把这种支配工作转化为本能，甚至为了适应这个过程，它还会变更身体的生理状况并将其遗传给后代。这可以部分解释人体结构和大脑的进化过程。“人-机合一”有可能会使这一进程加快，因此，它可从另一个方面提高人类的智能。

总之，我们认为，人工智能的发展将有助于解决现实层面的各类问题，从而促进人类智能的进化，以及人类智能进化方式的进化。我们既不同意这样一种观点：机器智能永远无法达到人类的智能；也不同意另一种观点：机器最终将“征服”整个人类世界。我们认为，更有可能的结果是：人-机合一、人-机共生。机器既不会永远落后于人类，也不会最终消灭人类。通过科学技术的进化，人类高度发达的意识与机器永不磨损的躯壳将趋于完美的结合，最终使人类变得不朽，我们每一个人都会成为“超人”。

在这个日新月异，充满挑战，充满机会的年代，个人或群体若想取得成功和生存下去，就不仅要依靠他们追求的目标和遵循的价值，也同样要依靠他们掌握的工具和拥有的能力。自古以来，人们不断探索和发明，寻求解放他们繁重劳动的工具，试图用各种机器替代人的部分脑力劳动。随着社会的发展，知识的飞速增长。人类迫切希望用机器替代人脑的某些功能，以提高人类征服自然的能力和效率。人工智能就是用人工的方法在机器（计算机）上实现的智能，或者说人类智能在机器上的模拟或人们使机器具有类似于人的智能。

展望未来，一方面我们对智能的工程化充满信心，因为社会已经或正在为发展人工智能

创造有利条件：社会信息化已经进入智能化的新阶段；脑科学、认知科学、复杂性科学和非传统逻辑学正在迅速发展，可提供坚实的理论基础；应用系统的复杂性和不确定性日益增加，对机器智能和智能机器的需求日益强烈；对智能本质和机制的认识已日益深入全面。另一方面，我们又需要对智能的工程化有冷静的思维。正如何华灿所言，历史上曾反复出现过对人工智能发展的过头预言和对新兴学派的顽固排斥，这都是源于对智能的本质和特点缺乏深刻认识。我们常犯的错误是把智能想象得太简单、太单纯、太固定、太绝对，以偏概全。智能科学和宇宙科学一样复杂，我们目前所知道的仍然很少，需要各方面不懈地探索。智能的复杂性、艰巨性和长期性，以及多层次、多因素、非线性和演化特性等，注定了急于求成是绝对不行的。若我们能放下日常使用的小视野、短时空、细粒度的思维方式；切换到长视野、长时空、粗粒度的思考中来，这或许才能够帮助我们掌握人工智能学科发展的脉络，才可从智能研究中不断地发现新的规律和新的方法，不断地推进智能的工程化进程。

17.8.5 未来的智能机器人—是人类的助手还是人类的主人？

关于未来的智能机器人，一直是一个可引起人们无限遐想的对象。无论是科普或科幻，总是引起人们的不尽的猜想。

我们知道，人是有关价值观的，人的价值观是在实现自身欲望和需求的过程中以及在接受教育、体验生活和社会实践中形成的。这种价值观完全是无意识的，它的作用是产生实现生理和心理平衡的驱动力，以支配人的判断和选择，即态度和行为指向等。而机器没有生命，也就没有作为主体的欲望、需求、情感和意志，没有自发的心理平衡和驱动力。有人认为，如果要赋予机器以智能，就需要给它的智能以导向。对此，我们同意。但是，作为人类智能的延伸，未来智能机器的智能朝向应该与人类的价值观在逻辑上是一致的。如此，这种智能化的“机器人”就需要我们在赋予它智能的同时，也为它建立一种与人类同质的价值观和自发动态平衡的机制，以使其主体化。这种智能要靠外部驱动力激发，并按人类的智能模式发展，其对外交互的方式也主要是针对人类的，而且人类自认可以控制它的价值取向。

那么，会否出现另外一种将智能赋予机器而使其自发产生价值观的本能，而这种本能几乎是非人文取向的，由其形成的价值观完全是天然的，也许是人类无法理解的情况呢？对此，我们还不得而知。但这种危险无疑是存在的。这种智能的危险在于，它会脱离人的支配，甚至会出现与人类基本价值观念不符或某些反人类的指向。有人认为，我们人类的价值体系是以人类为中心和建立在人类的利益高于一切的基础上的，而智能机器人也许会把维持大自然的某种平衡看得更为重要。应该说，这种担心不无道理。因此，尽管现在智能机器人还只是一片浮在空中的云，但是，一些对智能机器的担心和相关反驳却早已滚滚而来。英国雷丁大学的凯文·沃里克确信，到2050年，机器将征服人类。研制人造脑的俄罗斯科学家维塔利·瓦利采夫自称，他新研制的电脑已超越了以前的大脑模型。但如果受到虐待的话，这个全新的大脑可能会变成科幻小说中所描写的能毁灭其创造者的怪物。他说：“这个机器要像新生儿那样接受训练。对我们来说，最要紧的是要把它变成朋友，而不是罪犯或敌人。”日本高级通信技术国际研究所的雨果·德加里斯承认，他创造的东西可能会把他“像苍蝇一样拍死”的想法一直搅得他心绪不宁。当然，也有一些研究人员认为，这样的存在物拥有高超的智慧，不会不尊重各种各样的生命形式。伦敦帝国理工医药学院的伊戈尔·亚历山大说，机器会怀有恶意的想法是建立在这样一种错误的假设基础之上的，即智能机器行为方式将与人类非常接近，“具有人类的弱点和所有的特征”。但是，没有性别又知道自己是机器并且基本上可以永远存在的生物，不会去竞争领土和配偶—而这正是人类残暴和虐待低级生命形式的两个

主要原因。如果超级智能机器认为人类不适宜与它们共同生存在地球上，也许它们只会在自己身上建造一个助推器，然后飞向太空。一些机器总是要飞向太空的，去寻找新的知识，因为太空旅行对它们来说只是小事一桩。

我们认为，不管人类的科学技术如何发展，不管人类创造出的智能体如何高级，人类也决不会允许把自己当作某种外在意志所支配的筹码。人工制造的高级智能体，是为人服务的，是让它听从人的指挥的。一切智能机器的属性仍然是工具，它的用途范围将仅仅是帮助人类制定、探索或发现规则以及利用这些规则，替代人并为人服务。人的行为分为社会行为和个人行为。智能在人的社会行为中的作用主要是制定社会规则、探索和发现自然规则以及选择和利用这些规则。而智能在人的个体行为中主要是通过个人情感和意志起作用以处理新鲜感受。对于智能在人类社会行为中的作用，我们可以形象地把人类智能比作是指挥官、管理者、医生或法官，而把未来的智能机器比作服从命令的办事员。未来的智能机器能给我们带来什么，会有如人一样的七情六欲吗？会帮助我们解决生活中的所有问题吗？能够让我们把世间的纷扰看得更清楚吗？能够让我们不再彷徨无助吗？我们还不知道，不过，有一点可以肯定，不论未来的智能机器将是什么样子，未来的社会必定是高度智能化的人类与高度智能化的人类社会，未来社会仍需要发展，仍需要人类的智慧和努力，成功仍然要靠人的智慧和努力。

参考文献

- 1701 高济 《人工智能》课件
- 1702 徐从富 《人工智能引论》课件
- 1702 人工智能与智能进化 郝宁湘, 郭贵春 科学技术与辩证法 2005年6月第22卷, 第3期
- 1703 徐宝祥, 叶培华 知识表示的方法研究 情报科学 2007(5)
- 1704 史忠植、余志华: 认知科学和计算机, 科普出版社, 1990.
- 1705 史忠植, 高级人工智能, 科学出版社, 1998
- 1706 史忠植, 智能科学, 清华大学出版社, 1998
- 1707 王永庆. 人工智能原理与方法. 西安交通大学出版社, 1998.
- 1708 陆汝钤. 人工智能. 科学出版社, 2000.
- 1709 蔡自兴等. 人工智能及其应用. 清华大学出版社, 1996
- 1710 冯天瑾, 智能学简史, 科学出版社, 2007.
- 1711 李衍达 关于智能产生机制的思考-智能与信息 自然科学进展 2007年1月第7卷第1期
- 1712 李强, 杨晓京, 魏岚 基于神经网络信息融合的智能机器人 机电工程技术 2006年第35卷第6期
- 1713 智能科学网站: [//www.intsci.ac.cn/](http://www.intsci.ac.cn/)
- 1714 Sowa, J. F. Knowledge Representation: Logical, Philosophical, and Computational Foundations. China Machine Press, 2003.
- 1715 Russell, S., Norvig, P. Artificial Intelligence: A Modern Approach. Pearson Education, 人民邮电出版社, 2002.
- 1716 Thomas Dean, James Allen, Yiannis Aloimonos 著, 顾国昌等译, 人工智能—理论与实践(Artificial Intelligence: Theory and Practice), 电子工业出版社, 2004
- 1217 James Allen 著, 刘群等译, 自然语言理解(Natural Language Understanding),

电子工业出版社，2005。

1718 陆汝钤主编，世纪之交的知识工程与知识科学，清华大学出版社，2001

1719 张仰森，人工智能原理与应用，高等教育出版社，2004

1720 Simon Haykin McMaster 神经网络与机器学习(英文版第3版) 机械工业出版社 • 2009.03

1721 Zadeh “Fuzzy Logic = Computing with Words” IEEE Transactions on Fuzzy Systems, Vol. 4, No. 4, pp.103-111, 1996.

1722 Amari, S., Differential Geometrical Methods in Statistics, Springer Lecture Notes in Statistic, 28, Springer, 1985

1723 Baddeley, A D. Working memory. Science, 1992, 255: 556-559.

1724 Bredeche, N., Zhongzhi Shi, Zucker, J. Perceptual learning and abstraction, Proceedings of ICCI' 03, 2003

1725 Crick, F. and Christof, Koch, Consciousness and Neuroscience, Cerebral Cortex 8: 97-107, 1998

1726 Haken, H., Principles of Brain Function, a Synergetic Approach to Brain Activity, Behavior, and Cognition. Berlin: Springer 1995

1727 Jibu M., Yasue K., Quantum Brain Dynamics and Consciousness: An Introduction, Benjamins, John Publishing Company, 1995

1728 Mayeux, R., Kandel, E.R. Disorders of language: the aphasias. In: Kandel, E.R., Schwarz, J.H., Jessell T.M. (eds). Principles of Neural Science, 3rd ed., Elsevier, 1991, pp. 840-851

1729 Popper, Karl R. & Eccles, John C., The Self and Its Brain, Springer International, 1977

1730 Rapaport W.J., Cognitive Science, Grove's Dictionaries, New York, pp. 227-233, 2000

1731 Reber, A.S. Implicit learning of artificial grammars. Journal of Verbal Learning and Verbal Behaviour, 6, 855-863, 1967

1732 Richard, F.T. and J.J. Kim. Memory system in the brain and localization of a memory. Proc. Natl. Acad. Sci. USA, Vol.93, pp. 13438-13444, Nov. 1996

1733 Posner, M.I., Attention: the mechanism of consciousness. Proc. National Acad of Sciences, U.S.A., 91(16):7398-7402, 1994

1734 Treisman, A., & Gormican, S., 1988. Feature analysis in early vision: Evidence from search asymmetries. Psychological Review, 95, 15-48

1735 Winston, P.H., Why I am optimistic, Invited Speaker, AAAI-99, 1999.

1736 玛格丽特·博登 (刘西瑞 王汉琦译) 人工智能哲学(The Philosophy of Artificial Intelligence) 上海译文出版社 2001年11月

1738 陈兆乾 周志华 陈世福 神经计算研究现状及发展趋势

<http://wenku.baidu.com/view/74225bf67c1cfad6195fa7ce.html>

1739 高济、朱淼良、何钦铭 人工智能基础 高等教育出版社 2002-08-01

1740 蔡自兴 人工智能基础 高等教育出版社 2005-05-08

- 1741 Baldi P and Brunak S. *Bioinformatics: The Machine Learning Approach*. Cambridge, MA: MIT Press, 1998
- 1742 Berthold M and Hand D J. *Intelligent Data Analysis: An Introduction*. Springer-Verlag, 1999
- 1743 Bezdek J C. On the Relationship between Neural Networks, Pattern Recognition and Intelligence. *The Int J of Approximate Reasoning*, 1992, 6(2): 85-107
- 1744 Bezdek J C. What is Computational Intelligence? In Zurada J M et al (eds): *Computational Intelligence Imitating Life*, 1-12, New York: IEEE Press, 1994
- 1745 Cai Z and Tang S X. Controllability and robustness of T-fuzzy system under directional disturbance. *Fuzzy Sets and Systems*, 2000, 11(2): 279-285
- 1746 Cai Z. Disciplinary frame and general features for intelligence science. *Proc. 1st China-Korea Joint Workshop on Intelligent Systems*, Seoul, Korea, 2002: 1-4
- 1747 Cai Zixing. *A Knowledge Based Flexible Assembly Planner*. IFIP Transaction, B-1. North Holland, 1992: 365-371
- 1748 Cai Zixing. *An Expert System for Robotic Transfer Planning*, *Computer Science and Technology*, 1988, 3(2): 153-160
- 1749 Cios K, Pedrycz W, and Swiniarski R. *Data Mining Methods for Knowledge Discovery*. Boston: Kluwer Academic Publishers, 1998
- 1750 DeJone K A. *Genetic Algorithms: A 25 Year Perspective*, In: *Computational Intelligence Imitating Life*. Eds. J M Zurada et al, New York: IEEE Press, 1994
- 1751 Durkin J. *Expert Systems: Design and Development*. New York: Macmillan Publishing Company, 1994
- 1752 Fogel D B, *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence*, Second Edition. IEEE Press, 2000
- 1753 Fogel L J. *Intelligence Through Simulated Evolution: Forty Years of Evolutionary Programming*. A Wiley-Interscience Publication, 1999
- 1754 Fu K S. *Learning Control Systems and Intelligent Control Systems: A Intersection of Artificial Intelligence and Automatic Control*. *IEEE Trans*, 1971, AC-16(1): 70-72
- 1755 Gen M, Cheng R. *Genetic Algorithms and Engineering Optimization*. A Wiley-Interscience Publication, 2000
- 1756 Genesereth M R, Nilsson N J. *Logic Foundation of Artificial Intelligence*. Los Altos, CA, USA: Morgan Kaufman, 1987
- 1757 Gevarter W B. *Artificial Intelligence Applications: Expert Systems. Computer Vision and Natural Language Processing*, NOYES Publications, 1984
- 1758 Goldberg D E. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Readings, MA: Addison-Wesley, 1989
- 1759 Grossberg S. *Neural Networks and Neural Intelligence*. Cambridge, Mass: MIT Press, 1988
- 1760 Hajek P, et al. *Uncertain Information Processing Expert Systems*. Boca Raton, Florida: CRC Press, 1992
- 1761 Han J and Kamber M. *Data Mining: Concepts and Techniques*. Los Altos, CA, USA: Morgan Kaufmann Publishers, 2001
- 1762 Hayes Roth F, Waterman D, Lenat D eds. *Building Expert Systems*. New York: Addison Wesley, 1983

- 1763 Hertz J, Krogh A, and Palmer R G. Introduction to the Theory of Neural Computation. Reading, MA: Addison-Wesley, 1991
- 1764 Hopfield J J. Artificial Neural Networks. IEEE Circuit and Devices Magazine, 1988, 12
- 1765 Kelly R V Jr. Practical Knowledge Engineering. Digital Press, 1991
- 1766 Khanna T. Foundation of Neural Networks. New York: Addison Wesley. 1990
- 1767 Kosko B. Neural Networks and Fuzzy Systems. A Dynamical Systems Approach to Machine Intelligence. New York:: Prentice Hall, 1992
- 1768 Maes P. Designing Autonomous Agents. Cambridge, MA: MIT Press, 1990
- 1769 Marks R. Intelligence: Computational versus Artificial. IEEE Trans. Neural Networks, 1993, 4(5): 737-739
- 1770 Minsky M. Society of Mind. New York: Simon & Schuster, 1986
- 1771 Morell R et al eds. Minds, Brains, and Computers: Perspectives in Cognitive Science and Artificial Intelligence. Ablex Publishing Corporation, 1992
- 1772 Murthy S K. Automatic construction of decision trees from data: A multi-disciplinary survey. Data Mining and Knowledge Discovery, 1998, (2): 345-389
- 1773 Negoita C V. Expert Systems and Fuzzy Systems. Benjammin/Cummings Publishing Company Inc, 1985
- 1774 Newell A, Shaw J C, Simon H A. A Variety of Intelligent Learning in General Problem Solver. Self Organizing Systems. New York: Pergamon Press, 1960
- 1775 Nilsson N J. Artificial Intelligence: A New Synthesis. Morgan Kaufmann, 1998
- 1776 Winston P H. Artificial Intelligence (Third Edition). Addison Wesley, 1992
- 1777 Zadeh L A. A new direction in AI: toward a computational theory of perceptions. AI Magazine, Spring 2001: 73-84
- 1778 Zadeh L A. Making Computers Think Like People. IEEE Spectrum, August, 1984
- 1779 丁永生, 邵世煌, 任立红. DNA 计算与软计算. 北京: 科学出版社, 2002
- 1780 何华灿, 王华等. 泛逻辑学原理. 北京: 科学出版社, 2001
- 1781 何新贵. 知识处理与专家系统. 北京: 国防工业出版社, 1990
- 1782 焦李成. 神经网络系统理论. 西安: 西安电子科技大学出版社, 1990
- 1783 廉师友. 人工智能技术导论, 第二版. 西安电子科技大学出版社, 2002
- 1784 林尧瑞, 郭木河. 人类智慧与人工智能. 北京: 清华大学出版社, 2001
- 1785 林尧瑞, 张钊, 石纯一. 专家系统原理与实践. 北京: 清华大学出版社, 1988
- 1786 陆汝铃. 人工智能. 北京: 科学出版社, 2000
- 1787 陆汝铃主编. 世纪之交的知识工程与知识科学. 北京: 清华大学出版社, 2001
- 1788 史忠植. 高级人工智能. 北京: 科学出版社, 1998
- 1789 史忠植. 知识发现. 北京: 清华大学出版社, 2002
- 1790 史忠植. 智能主体及其应用. 北京: 科学出版社, 2000
- 1791 涂序彦. 人工智能及其应用, 北京: 电子工业出版社, 1988
- 1792 阎平凡, 张长水. 人工神经网络与模拟进化计算. 北京: 清华大学出版社, 2000
- 1793 杨炳儒主编. 知识工程与知识发现. 北京: 冶金工业出版社, 2000
- 1794 尹朝庆, 尹皓. 人工智能与专家系统. 北京: 中国水利水电出版社, 2002
- 1795 张钊, 张铃. 问题求解理论及应用. 北京: 清华大学出版社, 1990

1796 张文修, 梁怡. 遗传算法的数学基础. 西安交通大学出版社, 2000