机器学习 人工智能 (6)

计算机科学 1993 Vol.20 No.5

27-40 TP18

机器学习技术与机器学习系统

赵沁平 魏 华 王军玲

（北京航空航天大学计算机系 北京100083）

摘 要

知识获取是知识处理过程中的"瓶颈"，这一问题的解决有赖于机器学习研究的进展。本文从两种不同的角度概述并评价了当前机器学习中的一些主要技术和典型系统，其一是着眼于机器学习技术所采用的推理方式，即演绎推理、归纳推理和类比推理；其二是强调机器学习系统所使用的知识类型和学习过程中信息流动的走向。

一、引 言

学习能力是人类智能的重要体现，使计算机系统具有某种程度的学习能力（即机器学习）的研究一直是人工智能研究领域所追求的一个目标。但是由于机器学习本身的难度及相关学科研究水平的限制，它的进展是曲折的。近十年来，随着计算机性能的大幅度提高，传统自动推理技术日趋成熟，特别是计算机广泛应用对机器学习所提出的迫切需求，使得机器学习越来越引起人们的重视和研究兴趣。人们从不同的应用和不同的角度出发，提出了一些引人瞩目的学习方法，构造了一些机器学习系统，如物理定律发现系统BACON、概念学习系统AQ、ID3、基于解释的学习、神经网学习和遗传算法等。机器学习的研究进入了一个新的渐进发展时期。

从采用的推理方式来说，现有的机器学习方法和机器学习系统大体有三类，即基于演绎、基于归纳和基于类比推理的学习。演绎推理是一种保真性推理，数理逻辑是经典的演绎推理的形式化研究，已经相当成熟，后来发展起来的许多非经典逻辑，如模态逻辑、时态逻辑、多值逻辑、非单调推理，甚至经典的数学归纳法也都属于演绎推理范畴。归纳推理是从个别事例到一般性命题的推理，类比推理是从个别事例到个别事例的推理。归纳和类比是非保真性推理，其结论是或然性的，与演绎推理研究相比，归纳和类比远未成熟。因此只要保真的演绎推理能满足学习过程的需要，人们就应当首先使用保真的推理。早期的机器学习系统一般都用单一的推理方式，现在则趋于集成多种推理技术来支持机器学习方法。图1对现有的一些主要机器学习方法进行了分类[1]。

另一方面，也可以从机器学习过程中信息流动的走向来刻画和划分机器学习方法。

下面，我们将从上述两种不同的观点对这些方法进行概述和分类。

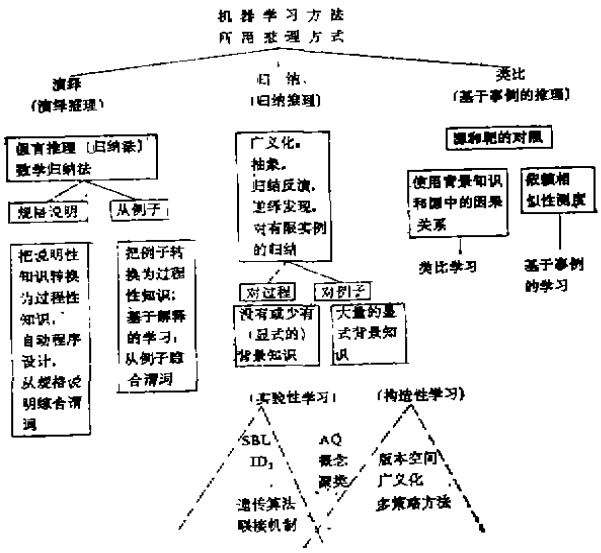

图1 使用三种推理方式的机器学习方法

二、若干基于演绎的学习方法

2.1 自动程序设计（AP）

AP的产生与机器学习没有太大的关系，

但是AP过程反映了某种面向目标的规划行为，能从用户提供的任务规格说明产生用户事前未必知道的可完成规定任务的动作序列或程序，因而我们根据它所采用的主要技术，把AP归入基于演绎的机器学习。

一个程序的规格说明包括输入向量x、输入条件C(x)、输出向量y和输入-输出关系R(x, y)。当且仅当输入向量x对于相应程序来说可接受时，输入条件C(x)为TRUE。输入-输出关系R(x,y)表示了输入与输出之间的关系链。自动程序设计就是要寻找一个程序Prog，使得：(T1) y=Prog(x)；(T2) ifC(x)then R(x,y)。例如一个商余问题的程序规格说明是：

输入向量x：(x1, x2)

输入条件C(x)：x2>0

输出向量：(y1, y2)

输入-输出关系R(x, y)：(x1=x2 * y1+y2)
&(y2<x2)

当C和R都是公式时，我们称Prog的规格说明是形式规格说明。除程序规格说明之外，还可以采用别的方式来规定程序功能。比如，在由例子综合程序时，我们可给出一个输入输出值所构成的有限集合。

当给出程序的形式规格说明时，AP就可以通过规格说明定理（specification theorem)的归纳证明获得Prog。

∀x〔C(x)⇒∃yR(x, y)〕

(规格说明定理ST)

注意，其中输入向量x是全称量化而输出向量y是存在量化。若把公式C(x)⇒∃yR(x, y)记为F(x)，则ST成为∀xF(x)。设φ是对应于ST的Skolem函数，即y=φ(x)。当输入向量属于一个良构的定义域时，我们便可以从ST的归纳证明引出φ的递归定义。这样得到的函数φ就对应于目标程序Prog。由于Prog是通过ST的构造性证明得到的，因而Prog的正确性也得到了保证。然而这种构造过程绝非易事。

2.2 由规格说明综合谓词

从形式规格说明进行谓词综合的目的是要为一个集合T的特征谓词P提供一个递归定义。设∀xA(x)为集合T上的一个公式(定理)。当下列条件成立时，称∀xA(x)部分为假。

(1) T=U∪V, U≠φ, V≠φ

(2) x∈U 当且仅当A(x)为TRUE

(3) x∈V 当且仅当A(x)为FALSE

现在要寻找一个谓词P，使得∀x〔P(x)⇒A(x)〕为TRUE。我们称部分为假的定理∀xA(x)为谓词P的形式化规格说明，而把从∀xA(x)获取P的一个可计算性过程称为从形式规格说明进行谓词综合。

谓词综合首先要由定理证明器证明定理∀xA(x)，证明失败时，证明器分析失败的原因并构造促成证明成功的条件。这些条件可能是常量条件，也可能是存在量化的引理。若是后者，则继续证明该引理，而导致综合新的递归定义的谓词。从规格说明综合谓词主要应用于由不完全的规格说明进行程序综合。

2.3 基于解释的学习(EBL)

从原理上讲，EBL是把一条通用规则Rg1转换成较为特殊的规则Rp。由于Rp隐含了对Rg1应用于特定实例的某种分析，因此Rp更为有效。这样，我们可以把EBL看作是一种获取应用Rg1的策略知识的途径。这种策略知识直接并入Rp。由于Rg1和Rp采用同一逻辑表示，因此，起始的通用规则集合{Rg}中没有元规则。已有不少关于EBL方面的工作，典型的系统有Mitchell的LEX-Ⅱ和Minton的PRODIGY[2]。

在一般的EBL中，{Rg}和{Rp}都属于一阶逻辑，而PRODIGY除了有一阶逻辑的{Rp}之外，还包含一些有关问题求解程序的元知识，因此它也包含一些二阶知识{Rg2}。由于综合了二类知识，所以PRODIGY既能生成规则{Rg}，又能生成二阶规则{Rg2}(元知识)。

2.4 由例子综合谓词

对例子进行综合而生成谓词的方法主要依赖于一种称为内部构造(intraconstruction)的演绎过程。内部构造的关键所在是归结的反演(inversion of resolution)。

2.4.1 归结的反演 我们用Res(A, B)表示Horn子句A和B的归结结果，A、B分别为Ha, −Ba, Hb, −Bb。设La是子句A头部即Ha的句节(literal)，Lb是子句B的体即Bb中的一个句节。如果通过置换δ，La可与Lb匹配，即δLa=δLb，那么子句A和B就可以进行归结。我们用Bb′代表从Bb中删除Lb后获得的句节集合。这样归结结果为：

$$Res(A, B)=\delta(Hb, -Ba, Bb') \quad (1)$$

归结的反演是归结的逆过程：给定A和Res(A, B)，求B。

2.4.2 内部构造 内部构造是一个通过三个子句的归结反演而产生新谓词的过程：给定两个子句B1和B2(B1、B2具有相同的头部)，求解三个子句C1、C2(C1、C2具有相同的头部)和A，使得B1=Res(C1, A)，B2=Res(C2, A)，且集合{B1, B2}与集合{C1、C2、A}等价。下面讨论内部构造的算法。

当执行归结Res(C1, A)和Res(C2, A)时，子句C1和C2的头部分别与子句A的体中的同一句节匹配并相互抵消。而在其逆过程内部构造中却是要生成新的谓词，并出现在C1和C2的头部。两次使用公式(1)，可得：

$$B1=Res(C1, A)$$
$$=\delta1(Ha, -Bc1 \& Ba')$$
$$B2=Res(C2, A)$$
$$=\delta2(Ha, -Bc2 \& Ba')$$

从上式可以看到子句A的组成部分Ha, −Ba′同时在B1和B2中出现，因此，内部构造应从B1和B2的推广以及相应的变量替换开始。该推广构成了A的一部分。然后，把新的谓词加到该推广以及B1、B2的子句体中。在内部构造过程中，要保持变量的链接关系。

例如，有如下两个子句：B1=uncle(X, Y): −brother(X, Z)&father(Z, Y)和B2=uncle(X, Y): −brother(X,Z) &mother(Z, Y)。首先选择这两个子句的一个推广，假定我们选：G=uncle(X, Y): −brother(X, Z)。一般来说总会有多种可供选择的推广，例如uncle(X, Y)、uncle(X, Y): −brother(U, V)等。因此，必须由用户对此做出决断，这一步是归纳过程。

下一步处理推广的"剩余"部分，并把这些剩余部分看作头部为"newpred"的子句的体。本例中B1和B2的剩余部分分别是"father(Z, Y)"和"mother(Z,Y)"，故新子句为：C1=newpred(Z, Y): −father(Z,Y)和C2=newpred(Z, Y): −mother(Z, Y)。至此，便给定了"newpred"，然后，我们把newpred加到G的子句体中，得：

A=uncle(X, Y): −brother(X, Z) &
newpred(Z, Y)。

这个例子比较简单，无需进行变量替换。对于较复杂的情况，需注意变量替换的传播，使得两个等式B1=Res(C1, A)和B2=Res(C2, A)成立。从而保证{B1, B2}与{C1, C2, A}等价。

关于内部构造的实现方法，参见[3]。

2.5 弱理论(Weak-Theroy)中的基于解释的学习

由于客观世界的复杂性、多样性以及人类认识世界的无限性，使得许多理论在当前或在某一阶段都是不完全、不一致和难以处理的，我们称之为"弱理论"。有一些工作正是寻找从这种弱理论中提取解释的方法。比如PROTOS系统、DISCIPLE系统和Schank的XPs理论就分别是以上述三种不同弱理论为研究对象的学习系统。

2.6 可能近似正确(PAC)学习

使归纳过程更为可靠的最常用的方法是把潜在错误的统计量限制在给定的概率范围之内。Valiant提出的PAC学习采用了这一方法，他希望用学习结果可能发生错误的近

似值来刻画一类可在多项式时间内学会的学习问题的特征。

为了具体说明PAC概念，举一个赌场新看门员的例子。该看门员要识别18岁以下的未成年人。他需要跟有经验的看门员学习这一技能。老板希望他不必花费很多时间便能胜任工作，因此这是一个多项式时间问题。另外，新看门员的判断应有一定的置信度，设置信因子为δ。若老板希望他以99.9%的正确率来识别未成年人，那么，δ=0.001。同时老板也允许他的判断有一定误差ε。如果10,000个成人中至多允许判断错一位，那么ε=0.0001。这就是说，老板要保证发生"误差大于ε"的错误的概率低于δ。下面给出更为形式化的描述。

设U是对象的可数空间，本例中U为赌场中赌博者构成的集合。设L1、L2，……为U的一组可数子集。我们的任务是在访问样本集合（新看门员在培训期间遇到的人构成的集合）期间，需识别一个子集Lun(不到18岁的未成年人构成的集合)。设d是两个子集Li和Lj之间的距离测度。当且仅当Lun与Ln之间距离大于ε的概率低于δ时，我们称Ln的识别过程是一个正确的PAC识别，即：

$$P[d\ (Lun,\ Ln)>\varepsilon]<\delta$$

现在给出公式中d的定义。设D(也许未知)是U上的一个概率分布，DIFSYM (S, T)是集合S和T的对称差，即DIFSYM(S,T)=(S−T)∪(T−S)。$P_D(x)$代表在U中找到元素x的概率(相对于分布D)。因此，

$$d(S,\ T)=\sum_{x\in DIFSYM\ (S,\ T)} P_D(x)$$

距离d实质上表示了一个随机调用能够获取DIFSYM(S, T)中元素的概率。要精确地定义一个PAC识别必须选择这种距离。在本例中，U的两个子集间的距离是它们所含的成人数目。因此，Lun与Ln之间的距离恰为Ln中的成人数。读者可看到这个距离非常粗糙，任何年龄段的成人与一个未成年人混淆都会得到同样的结论。

通常采用PAC学习方法来确定一个问题所需要的例子数。例如，要从含有n条已知规则的集合{L1，……，Ln}中，学习一条与m个例子相符合的规则，那么当例子数目$m=\frac{1}{\varepsilon}\ln(n/\delta)$时，我们就可以说这一学习算法是一种PAC学习算法。这是因为与m个例子相符合的规则出现"误差大于ε"的错误的概率低于δ。设Ln是一条规则，且d(Lun, Ln)>ε，那么任意实例与Ln符合的概率低于(1−ε)；因此m个随机实例皆与Ln相符合的概率低于$(1-\varepsilon)^m$，它由$e^{-\varepsilon m}=\delta/n$所定界。显然，至多有N−1条规则有这种结果，因此其中一条规则与m个实例相符合的概率上限为δ。由定义可知，该算法是一种PAC学习算法。

假设现在只有两条备用规则。若要求置信度大于99%，错误的误差低于0.0001，那么学习该问题所需要的实例数目m=10,000*ln(200)≈53,000。由于实际上大多数人只需要一个例子就可完成该学习，所以人们可能会怀疑PAC与现实之间的关系。然而，最近Grewier和Elkan计算了建立一个表达式所需的实例数目，并发现它的界限与PAC学习的界限数量级相同。实际上要求如此之多的例子是容易理解的，因为表达式所引入的领域知识已经隐含了大量的实例。

上述研究表明提供显式的知识可以取代大量有关例子，从而精化学习任务。把例子与已有知识相结合，这是近十年来机器学习研究的趋势之一。

三、ML常用的几种归纳过程

Price将归纳定义为"从事实建立理论的过程"。Kodratoff则认为，归纳法把一个或几个基本归纳过程与许多必要的演绎步骤相结合。从这种意义上讲，归纳是一种多策略机制。

用于ML的最著名的基本归纳过程是推广(generalization)和逆绎(abduction)。除此之外，本节还将介绍其它五种常用的归纳推

理，即因果判定(causality determination)、特征属性(attribution of property)、时空包含(spatial or temporal inclusion)、聚类(clustering)和组块(chunking)，以及它们和数值归纳的关系。

3.1 逆绎推理

逆绎推理是演绎推理，典型的是假言推理的逆过程。一般的假言推理是由A和A⇒B，推出B，而逆绎推理则由B和A⇒B归纳出A。例如，已知∀x[Drunk(x)⇒Stumbling(x)]，根据演绎推理，可从Drunk(bob)推出Stumbling(bob)。反过来，当看到有人摔倒时，我们也可以推断此人可能喝醉了。

由于可从多条定理演绎出同一事实，因此，反过来从一个事实就会逆绎出多个可能的结论。例如，∀x[Wounded(x)⇒Stumbling(x)]，∀x[Sunstuck(x)⇒Stumbling(x)]等都成立。

3.2 推广

推广是蕴含式为∀x[F(x)]⇒F(a)的假言推理的逆过程。假言推理从∀x[F(x)]推出F(a)，而推广则把F(a)推广为∀x[F(x)]。

推广过程中所依据的知识一般来说是按照概括性分类法则进行组织的，其中父辈比子辈的概括层次高。因此，推广也是一种相当复杂的知识表示机制。在这种意义上，推广是依一定条件由子辈上行到父辈的过程。在归纳上行到父辈之前，应当观察一定数量的子辈，以保证推广的合理性。正因为推广体现了概念或关系之间的一种"IS-A"从属关系，因此推广在知识的组织中有着重要的地位[4]。

3.3 因果判定

当若干现象同时出现时，探究它们之间可能存在的因果关系，即进行因果判定，是发现自然规律，认识客观世界的重要途径。因果判定在归纳逻辑中研究较多，著名的"穆勒五法"，即求同法、求异法、求同求异合用法、共变法和剩余法是常用的因果判定方法。若联合使用其中若干种，将会提高因果判定的可靠性。但是穆勒五法对于因果判定问题来说既是不完备的，又是冗余的。这也说明了客观世界的复杂多样性。

除归纳以外，也可以使用演绎方法。这首先要求在同时出现的现象中假定起因现象，然后进行演绎证明看能否得到结果现象。

因果判定在机器学习中应当有重要作用，但是目前这方面的研究还不多。有一些关于概率因果关系方面的工作[5]。

3.4 特征属性

与因果判定不同，当若干现象同时发生时，我们也可假定其中某一现象(观察结果)是概念，而其它现象(观察结果)则是它的特征。概念与特征的选择是归纳过程，即将合取式转换成蕴含式。例如，看到名为July的天鹅时，可将观察结果写成swan(July)&white(July)。这个表达式没有反映概念与特征的关系。我们假设swan(July)是概念，它有特征white(July)。这样便可将上述合取式转换成蕴含式：swan(July)⇒white(July)。

由于该过程形象地描述了概念与特征的关系，因此也非常有利于构造知识。通常一个概念属于某个概念类，比如天鹅便是鸟这一概念的一个子辈。注意，天鹅这一概念并不是更为广义的鸟这一概念的特征。因此一般性与概念-特征关系之间不会产生混淆。

3.5 时空包含

空间(时间)包含是指在空间上(时间上)被包含的区域内发生的所有事件，也在包含该区域的区域内发生。例如，∀x[happens-in-NewYork(x)⇒happens-in-USA(x)]表明在纽约发生的任一事件，也在美国发生。时空包含反映了一种"PART-OF"从属关系，这种关系可把某种结构引入知识。通常可根据所采用的具体类型，通过演绎或归纳来建立时空包含。

3.6 聚类和组块

聚类是把满足一定条件或准则的若干知

识元，如领域对象、概念、特征或关系等归入一类。在数据分析中，聚类技术是归纳式数值技术。而当聚类是针对知识元所具有的符号特征时，则称为符号聚类。在符号聚类的情况下，如果所有知识元共用部分相同的信息，则聚类是归纳的。

组块是把问题求解过程中常用的操作序列看成一个宏操作单元。组块本质上是一种演绎过程，但如果积累了许多组块，并研究过一些问题求解过程，则会发现它具有一定的统计特性。SOAR系统是引入组块机制的典型系统。

值得一提的是，由于聚类基于概念对实例进行划分，组块则在连续的动作（或连续的概念）之间建立联系，所以这两种推理均把结构引入了知识。

3.7 归纳概率

众所周知，归纳结果的正确性是或然性的，保证归纳结论的可靠性是归纳逻辑的研究任务。与演绎逻辑不同，归纳的前提与结论之间的关系是用归纳强度或归纳概率来刻画的。本节介绍两种归纳概率的定义。

第一种是Carnap的定义，他把归纳逻辑看做是一种与证据支持测度有关的理论，也就是说要衡量一个事件对理论的支持程度。具体定义如下：

设E={e}是实验事件的集合，{e′}是{e}的子集。如果对于{e′}，A(x)为TRUE（或FALSE），使得{e′}对于{e}有统计意义，则我们可导出A(x)对于{e}为TRUE(或FALSE)。显然，这相当于定义了所谓的"统计意义"。简单的作法是通过比较{e′}和{e}的大小并求{e}中{e′}的"空间分布"来定义"统计意义"。

这种类型的归纳概率并不要求{x′}在{x}内具有特殊的分布模式，而只需建立它们的实例之间的距离分布。因此，可以说它是在一种非常广的意义上构造知识。

第二种定义是通过概率因果关系获得的，称事件B与事件A有概率因果关系，当且仅当：1）B比A出现得早；2）当B出现时，出现A的条件概率比出现A的非条件概率大。

概率因果关系链也是构造知识的一种基本途径。

3.8 结论

归纳方法在原本互相独立的知识元之间建立了一个蕴涵式网络。这种网络必须是一致的，即不能由指向A的蕴涵式链导出¬A。经典的生物分类法便是这种网络的一个典型实例。

蕴涵式的来源不同，其语义也各不相同。例如，由因果判定方法得到的蕴涵式传递因果关系，不进行特征继承，而由推广归纳方法获得的蕴涵式不传递因果关系，只进行特征继承。也就是说，知识可由一个蕴涵式网络表示，根据链接的具体来源，这些蕴涵式由诸如IS-A、CAUSES、HAS-PROPERTY和PART-OF等这类语义来标识。

四、若干归纳学习的例子

本章介绍九种归纳方法，前七种源自ML领域，后两种是独立发展起来的。除结构匹配之外，其它归纳方法本质上都基于0阶表示。为此，我们首先讨论0阶逻辑和一阶逻辑。

一阶逻辑可以有全称量化和存在量化的变量，并且能很好地表达特征之间的关系，而0阶逻辑在这方面则比较欠缺。为说明这一点，我们举一个简单的例子。由多条线段构成的图，其中至少有两条线段互相垂直并且有共同的端点。用0阶逻辑可以表达如下：

(angle segment1 segment2=90°)&
(end1 segment1=end1 segment2)

假如图中互相垂直的两条线段是segment3和segment4，则上述0阶表达式不能适用，必须增加知识表达，否则知识系统不能识别后者。

如果采用一阶逻辑，相应的特征表达式可写成(angle(x1, x2)=90°)&(end(x1)=end(x2))。给出该表达式之后，识别系统首先察看所有线段对，选出互相垂直的线段

对，然后从中选择具有共同端点的线段对。这种识别过程体现了一阶逻辑的另一个特点，即需要遍历所有满足给定关系的实例组合。这种组合过多，会带来难以接受的计算复杂性。这时，就必须选择启发式函数，以减少复杂性。

在进行学习构造识别函数时，也需要考察实例组合。例如，我们想学习关于父亲的知识。给定三人John、Peter和Tom。假设John和Peter有成绩优秀的孩子，John和Tom有爱好体育的孩子。若从John和Peter开始推广，就会失去John的孩子也爱好体育这一知识。因此，一阶逻辑的推广技术必须对所选例子进行优化。

总之，与0阶表达式相比，一阶逻辑的主要特点是：(1) 可以表达事物特性之间的关系。(2) 要生成识别函数，需考虑可能的谓词组合。

4.1 ID3方法

ID3 是 *Quinlan*提出的一个算法。给定一个特征集合、一个例子集合以及例子所属的概念集合，而给出对所有例子的概念分类。该算法考察与每个特征相关的信息，从中选择信息量最大的一个，并遍历特征集合，建立一个识别例子的判定树。*ID3*避免引入特征的线性组合，以保证其结果的易理解性或易读性。

ID3是典型的乏知识系统。最近推出的ID3改进版本，所做的工作都侧重于使它的结果更易理解。

4.2 Star算法

Star算法是Michalski研制的AQ系统的核心算法。AQ系统是一个可为专家系统生成规则集合的系统。Star算法的目标是构造一个能识别所有正例(完整性)，排除所有反例（一致性）的函数R。为构造这样一个函数，需要首先建立一个最为通用的识别函数R_i，识别例子e_i(正例集合POS的当前成员)，排除例子$\{e_j\}$(反例集合NEG)。例子可用特征值语言表示为：

$$Eq: (X_1=V_1)\ \&\cdots\cdots\&\ (X_k=V_k)$$

其中，X_i是第i个特征，V_i是第i个特征的值。显然，识别函数可用同一语言写出。当识别函数的特征值和例子E的特征值相同时，这个识别函数就可识别例子E。R_i的构造分两步完成。第一步建立函数G_{ik}，G_{ik}是识别e_i，排除e_k的最通用的函数，其中$e_k\in\{e_j\}$。开始G_{ik}为空。如果在e_i和e_k中特性X_i不同，则令V_k等于e_k中X_i的值，并且把$\vee(X_i\neq X_k)$加到G_{ik}中，这里需要对识别函数进行仔细的选择。

第二步，对$\{e_j\}$中的每一个e_k构造G_{ik}，并设$R_i=\&_{ek}G_{ik}$，$e_k\in\{e_j\}$。如果R_i恰巧可以识别POS中的所有正例，则$R=R_i$。否则，再选一个e_j，构造R_j，并用$R_i\vee R_j$代替R_i。依此类推，直至$R_1\vee\cdots\vee R_n$完成为止。

用上述方法构造的识别函数往往相当复杂，为此，Michalski还给出了几种限定函数复杂程度的准则。

4.3 结构匹配(SM)

如果两个公式除常量和实例化的变量之外相同的话，就称它们在结构上互相匹配。形式定义如下：

设E_1和E_2是两个公式，E_1结构匹配E_2，当且仅当存在一个公式C和两个代换δ_1、δ_2，使得：(1) $\delta_1(C)=E_1$，$\delta_2(C)=E_2$；(2) 代换δ_1和δ_2中不出现的公式或函数。

一般来说，SM是困难的，不可判定的。但是在许多情况下，我们可以利用过去SM的经验来启发当前SM的过程和目标，而且即使SM不成功，对以后的SM来说，这种尝试本身也是有意义的。

4.4 知识精化

知识精化是发现知识库中的错误，使知识库中的知识逐步得到修正、增长和增强的机器学习技术。典型的具有一定程度知识精化能力的系统有CLINT、DISCIPLE和MOBAL。这三个系统都产生一些运行实例，由用户做出接受还是不接受的选择，系统根据用户的反应对知识进行精化。DISCIPLE对

系统运行的成功与失败做出某种解释，并与用户交互，由用户对知识进行修正，MOBAL和CLINT则对知识库进行一致性检查，当遇到不一致时，与用户交互。

知识精化是知识库和机器学习领域中的一个非常重要而且相当困难的课题。

4.5 归纳式程序设计(ILP)

ILP是机器学习领域中的一个新的研究方向，它所关心的问题是，在编写归纳程序时所采用的归纳策略和步骤是否合理。目前ILP有两种主要方法，即语义方法和语法方法。语义方法的基础是反演归结原理和反演推导；语法方法则以经典的推广和一些算符为主体。

我们早在1988年就提出了归纳式程序设计的概念[7]，当时国际上还未明确出现这一思想。我们认为归纳系统并不能完全"代替"人，而是"辅助"人完成一定的归纳学习任务。正如在一般程序设计过程中人的创造性活动反映在算法的设计上那样，归纳式程序设计过程也反映了人的创造性活动。我们在[7]中给出了归纳式程序的基本内容和典型模式：

Given：(a)观察语句(事实)集，F；(b)假定的初始断言；(c)归纳规则和归纳策略；(d)背景知识。

Find：归纳断言H，H应蕴涵F。

定义上述内容的过程称为归纳式程序设计过程；用于书写归纳式程序的语言称为归纳式程序设计语言；能执行归纳程序，完成特定归纳学习任务的系统称为归纳学习系统。我们的上述思想至今仍具有独到之处，当然其中还有一些困难需要研究与突破。

4.6 科学发现

几个早期的机器发现程序模拟了多种科学发现任务，并显示了有趣的性能。但是，与人类的能力相比，这些程序的适用范围还显得十分狭窄。从实验数据中发现科学定律是科学发现的主要途径，一般来说，要经过如下四个步骤：

- 根据某些相似点聚集数据；
- 构想能够与所给数据相符合的定理形式；
- 找出数据所隐含的规律，并给出描述该规律的确切的定理形式；
- 通过测试定理与数据的相符程度来评价定理。

近几年来，人们通过集成这四个步骤，试图提出较完整的科学发现计算模型，相继开发出一些系统，如可以确定一条定理应用条件的ARC系统，实验设计系统KEKADA和具有类比推理和理论修正能力的STAHLP。Sleeman的非形式化定量模型(IQM)则集成了上述几种系统所实现的所有功能。

4.7 概念聚类

概念聚类要求在聚类与其内涵描述之间建立关系。传统的数据分析也可以计算数据的聚类并得到每个聚类的概括性(Generalization)描述。概念聚类的特点在于，它采用与刻画聚类的概念相关的质量标准来获得聚类结果，即相关实例的集合。这需要一种比数据分析所用的距离更为复杂的实用测度。下面我们介绍两种概念聚类系统，COBWEB和KBG[8]。

COBWEB是一种采用0阶逻辑并以增长方式完成概念聚类的系统。它使用的测度称为范畴效用，该测度相对于概念的通用程度来评估概念的实用性。一个有用的概念实质上是在通用概念与特殊概念之间进行折中。COBWEB从实际的例子中计算实用测度值。系统允许在现存的类中添加新的例子，而当创建一个新类比添加例子更有效时，就建立一个新类，也可合并两个类，或对类进行划分。

随着新例子的加入，COBWEB根据实用测度逐步创建新的概念，修正或删除已有的概念。概念由属性值概率来刻画，在这个意义上，COBWEB中的概念是概率的。学习完成以后，系统便为所用的例子建立起一棵分类树，树中的每个结点就是由系统发现的一个概念。

KBG是一种采用一阶逻辑并以非递增方式实现概念聚类的系统。我们知道，一阶逻辑有两个主要特征，一是可以表示特征之间的关系，二是利用谓词间可能进行的组合生成识别函数。在KBG中，为避免发生组合爆炸，使事件出现的总数不致过高，所以采用的实用测度不是与事件出现的统计量，而是与相似性测度相关。这种测度用于刻画n元关系存在的可能程度。当计算出所有对象间的相似性测度之后，根据用户给定的阈值和算出的测度对对象进行聚类。最后，选出一类对象的最具代表性的实例，对其进行推广。

4.8 遗传算法(GA)

我们介绍GA的几个常用原则[9]。

4.8.1 表达式的选择 以一个具体实例为例。假定一个机器人在设有障碍物的环境中移动。环境由八个方向标志，障碍物的位置由01串表示。如果沿第i(1<i<8)个方向行进可遇到障碍物，则01串的第i位置1，否则置0。比如，机器人沿方向2、5和6可看到障碍物，那么障碍物的位置就表示为：(01001100)。如果机器人只能沿4个方向移动，那么我们可用4位二进制串代表机器人马达的工作情况。如果第i个马达是活动的，则将二进制串的第i位置1。所以，(1000)表示只有马达1是活动的。我们还可用*表示二进制串中的既可取1，又可取0的值，其作用相当于变量。这样，机器人移动的指令就可由一组形如(********)→(****)的规则来表示。规则左部代表机器人所处的环境，右部是与环境相关的机器人工作马达的状态。

4.8.2 规则的补偿与选择 一条规则的"强度"由它在所给环境中的实用性来定义，这是由GA的设计者给出的一个函数，它在很大程度上依赖于设计者的领域知识。如果一条规则的强度定义得不好，就会导致遗传算法难以收敛。

设F代表机器人与最近的障碍物P之间的平均距离，F_i^+是函数F应用规则i之后得到的值，F_i^-是函数F在应用规则i之前的值。V_i是应用规则i之后机器人的行进速度(若机器人未向障碍物靠近，则V_i为0，否则等于机器人的行进速度)。下面对规则i的应用定义一个补偿函数$r_i(n)$：$r_i=F_i^+-F_i^--V_i$。设$r_i(n)$代表第n次应用规则i时，该规则的补偿函数。一条规则的强度与该规则的应用次数相关，我们把它定义为所有补偿的平均值：

$$STRENGTH_i(n)=\frac{1}{n}\sum_{k=1}^{n} r_k(n)$$

4.8.3 规则系统的修改 从K条规则中选出M条最强的规则，再从这M条规则中生成P条新规则。经过这一循环，系统将含有M条最强的规则及其生成的规则。现有三种主要的用于生成新规则的遗传算法：

(1) 复制。通过复制一条已有的规则，以增加该规则的统计份量。

(2) 变换。对一条规则进行变换。例如，考察规则(01001100)→(1000)，可知沿方向5和方向6不管是否存在障碍物，都可移动。

(3) 交接互换。如果两个二进制串(010*01001*110)和(0010*10*****10)在第6位之后进行交接互换，则生成两个新的二进制串：(010*010*****10)和(0010*1001*110)。

一般来说，GA具有两个比较明显的特性，一是多个个体可采用并行方式进行搜索；二是为了检查远离收敛点的区域，可设一些离平均值较远的个体。因此，GA是一种新的搜索技术。

4.9 神经网络(NN)

NN是独立于ML自成一体发展起来的，而且能够进行某种程度的学习，因而对传统ML领域形成了一定的冲击。一些ML的研究工作者对NN方法和ML进行了比较[10]。结论大致如下：(1) NN能在无结构的数据中较好地进行学习；(2) 缺乏背景知识时，NN能较好地学习；(3)NN的学习速度一般来说相当慢；(4) 正确学习之前，NN需要大量

的准备工作；(5) NN不能提供易解释的规则。其中第2点和第5点十分重要，因为它们涉及到与用户的交互关系。

NN使用了隐蔽层，它们处于描述概念的符号属性与概念本身之间。通常，隐蔽单元的各节点没有特定的语义。一般来说，可采用两种方法"符号化"这些隐蔽单元，一是赋给它含义，即用语义上有意义的隐蔽单元来建造NN；一是从隐蔽单元中提取规则。

4.9.1 建造没有隐蔽层的NN 尽管NN在符号属性和最终概念之间还包含几个层次，但这些层次的结构和每个节点的含义都与背景知识密切相关。这样的NN系统采用了层次知识结构的建树机制。下面我们举例说明这种NN的建造过程。假设背景知识可由下述五条规则表示，并且要建造的NN的目标是识别概念A。

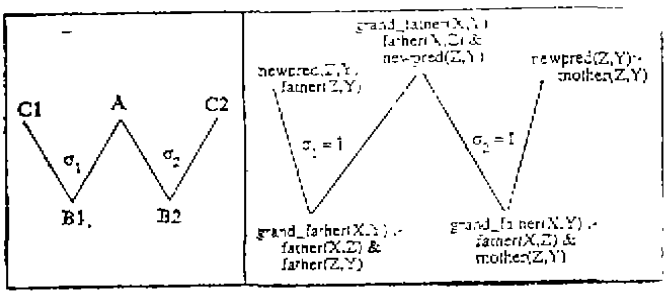

图2 内部构造的过程

A:—B, E; B:—C, D, I; C:—F,
D:—G, H; E:—I, J。

如果对背景知识提问?A，就会得到A的一棵证明树，如图(2) a所示。此后，把它们当做相应神经网络的起点。例如，若该树的整体结构保持不变，则隐蔽单元的个数等于该树的深度减去2。保留现有的连线。然后在每一层内建立所有必要的连线，并把相应的系数置0。这样，不会产生新的单元，在两个互不相邻的层次间也不会存在连线。图(2)b给出了这样一个神经网络，其中粗线代表从证明树中引入的连线，细线则代表系数为0新引入的连线。最后，在该结构上进行反向传播，根据具体情况，原有连线的系数可能减少，新连线的系数可能增加。在最终构成的网络中，结点间的所有连线都带有解释。

4.9.2 从带有隐蔽层的NN中抽取规则 如果没有隐蔽单元，最终的判定直接与它所依赖的元素相连，这样抽取规则的任务是比较容易完成的。但当存在隐蔽单元时，需要估计每个输入对最终输出的影响，需遍历所有路径，这可能会发生组合爆炸。由于各个层次都相互影响，因此产生的规则数目很大。一种解决办法是引入启发式函数来选择重要的隐蔽单元，并削减影响被选单元的选样数目。

五、通过类比和基于事例推理的学习

类比推理(AR)是由于认识到新情况(称靶)和已知情况(称基)在某些方面相似，从而推出它们在其它相关方面也相似的推理形式。下面是类比推理的一般模式，其中T2是类比结论。

B1, T1 (已知其中B1和靶中T1)
B1≈T1 (B1相似于T1)
B1≫B2 (在基中B1相关于B2)
B2 (已知基中B2)

T2 (其中T2≈B2) (推出靶有相似于B2的T2)
或B1, T1, B2, B1≈T1, B1≫B2
~T2(其中T2≈B2, ~表示类比推出)

如果将以上模式中≈改为=(相同)，≫改为⇒(演绎中的蕴含)，即变成演绎推理模式。这一变化说明：(1)类比推理是似然推理。因为≈比=弱，≫比=弱。(2) 为提高类比结论的可靠性，希望：·B1与T1的相似度越高越好，这将使B1≈T1更接近于B1=T1；·B1与T1包括的方面越多越好，B1与B2的相关性越强越好，这都将使B1≫B2更接近于B1⇒B2。

认知研究表明人们在类似情况间转换因果关系，即≫具体化为因果关系⇒。这也符合"相似原因引起相似结果"和"相似结果可能有相似原因"的类比直觉。

虽然早在1945年Polya就探讨了类比推理在数学问题求解中的重要作用，但是由于其发展依赖于相关领域的进展，如长期记忆中知识的组织和访问方式等，所以早期AI研究很少讨论机器类比(类比推理)。八十年代以来对类比推理开展了较为广泛和深入的研究，这段期间的工作可分为三类：类比问题求解，类比学习和类比推理的理论研究。

类比推理的理论研究 理论研究的目的在于捕捉类比推理的一般规律和方法。Gentner在多年工作的基础上提出了结构影射理论(简称SMT)。SMT的关键是系统性原理——"人们更喜欢由高阶关系连接的关系系统而不是孤立的关系"。SMT有两点贡献：a. 体现了以命题为中心的匹配——根据命题确定对象对应；b. 被转换的命题应是与已对应命题有连接的命题，而不是孤立的命题。Holyoak和Thagard提出了一种约束满足理论，他们采用联结机制实现了类比影射器ACME和类比检索器ARCS。

类比问题求解 Carbonell基于手段-目的分析法提出了一种类比问题求解的计算模型。在该模型中，将问题表示为初始状态、目标状态、路径约束、解(操作序列)。求解过程首先寻找与新问题的初始状态、目标状态和路径约束差别小的已求解问题作为相似问题，然后对相似问题的解应用一组变换操作，从而得到新问题的解。由于这种方法直接变换解，故不适用于问题间只有相同(似)求解策略但解的形式差别大的情形。在进一步工作中，他提出了推导类比。推导类比强调问题表示应包括求解过程中做出的决策序列，从而在出现上述现象时可以在新问题中引入类似求解策略。

类比问题求解方面的工作还有很多。总的来说，这些工作有如下特点：a. 用模式表示描述问题；b. 较多注意力放在怎样变换旧问题的解以便得到新问题的解；c. 要求相似问题有相似目标。

类比学习 Winston的类比学习系统用对象分类树来判断对象相似，采用强调对象的可扩充关系表示(类似语义网)描述情况。检索过程根据对象相似判断情况相似，匹配过程在对象对应集空间中选择有最多命题对应支持的对象对应集作为最好影射。并在最好影射下，将相似情况中与已对应命题有因果联系的命题引入到新情况，从而学习到关于新情况的知识。进一步推广这两个情况中相似因果链，得到一般规则。Burstern构作了CARL，它能模拟学生学习BASIC语言赋值语句的行为。CARL用因果模式表示情况，并基于具体——推广关系将情况库组织成层次结构。根据教师直接给出的类比对(新例子及相似的旧例子)，容易检索到适合新例子的因果模式。当把检索到的因果模式中的因果关系引入新例子后，就学习到了解释新例子的一般知识。通过解释与新例子同类的其它例子，可检验所学到的一般知识是否正确。

关于类比学习的研究目前存在如下局限：a. 联想能力弱。由于许多系统只根据语义识别相似情况，故难以排除表面相似情况。b. 未深入讨论如何在新情况中创建对象和谓词。c. 缺乏保证所学新知识具有较高可靠性的手段。

在总结有关工作的基础上，我们从1989年开始进行了类比推理理论和实现的研究，目前已得到一套基本完整的理论，并构作了实验性类比推理系统BHARS。我们将类比推理分为联想、求精、匹配和转换四个过程，提出了基于突出特征的类比联想和基于相关性的类比求精，建立了以命题影射为基本单位的匹配计算模型。在转换原理中，根据影射包括的命题对应数和是否符合靶目标要求来选择最佳影射，在保证影射的语义特性不变的前题下创建新对象和新谓词，并基于"相似原因引起相似结果"和"相似结果可能有相似原因"确定类比结论[11]。

基于事例的推理(CBR) 利用类比推理技术，采用考虑了浅层和深层结构知识的相

似性函数，遇到新情况、新问题时，将过去的相似情况或求解相似问题的经验（过去的经验包括失败的类似问题）进行适当的整理、修改，再重新用于新情况或新问题。

六、机器学习系统中的信息流

一个ML系统可以由学习期间系统中信息流动的方式和走向来描述。这种观点把学习系统看作是通过下述迭代过程积累经验，并从中提取信息的活动系统：ML系统在n个阶段与它的环境相互作用，这种相互作用由评价函数来刻画。评价函数对转换函数集合进行标志，指出在阶段n哪一个转换函数用于ML系统，并使ML系统转换到阶段n+1。改变评价函数或改变转换函数集合，就可以得到不同的ML系统。

6.1 不同类型的知识

从知识在ML系统中所起的作用来说，有四种类型的知识，即背景知识(Background Knowledge，记为BK)、策略知识(Strategic Knowledge，记为SK)、例子(Examples，或Case-Based Knowledge，记为CBK)及因果知识(Causal Knowledge，记为CK)。在ML过程中这些知识之间发生信息流动。

BK与某一给定的领域相关。该领域包含给定的对象集(比如在car领域中，一些指定范围内的car)，并且可用一组特征来描述(例如car的color、shape……等)。特征值给出了特征对象所处的状态（例如，对一个特定的my-car，有make=Peugeot，color=beige)。特征刻画了对象之间的关系，这些关系是在该领域中成立的一些定理(例如∀x[Peugeot(x)⇒yellow(x)]，从而表征了所研究的领域。

CBK描述了实例化的情况或事例，具体说明用户感兴趣的概念或行为。

CK对于一个特定的用户来说可看成解释性知识，它解释概念或行为间的关系，或者指明了因果符号的含义。CK是BK的子集。CBK和CK在很大程度上依赖于用户。

SK是说明如何使用BK优化已知准则的知识。一般来说，SK包括算子或特征的排序，以告知系统当给定条件满足时，应选择哪一个操作。大多数ML系统都具有一定程度的自动获取隐含的SK的功能。

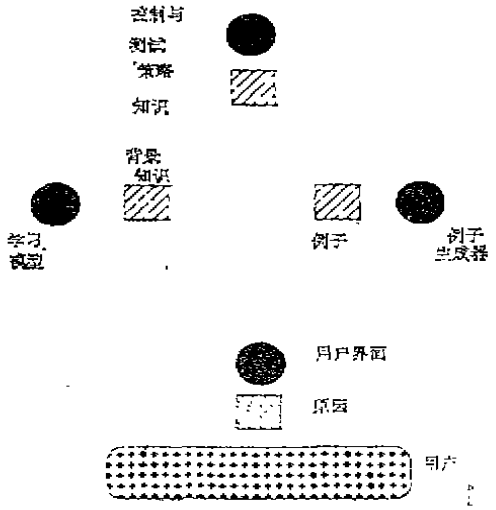

图3 四种类型的知识及其生成器

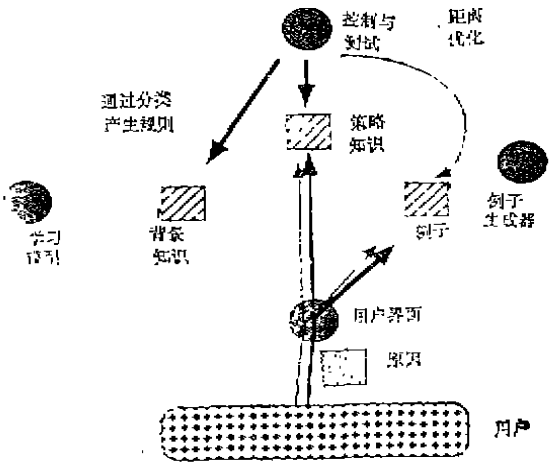

图4 数据分析中的信息流

我们假定四种类型的知识分别对应四种知识生成器，如图3所示。“学习算法”产生BK；“控制和测试算法”产生SK；“例子生成器”生成CBK；领域专家给出CK。在这些知识生成器和知识集合之间，信息可以多种方式流动。ML系统可以用这四类知识之间的信息流以及四类知识的产生过程来刻画和分类。

6.2 几种具体的ML系统中的信息流

这一节我们具体介绍几种ML系统，包括数据分析、ID3、NN、遗传算法、EBL和概念聚类中的信息流。

数据分析本质上由两种功能构成(图4)。其一是根据度量测度完成聚类操作，其二是完成分类操作，生成规则并且根据度量距离进行计算。

用户提供例子和SK。部分SK是隐式的，包含在例子的表示中。另一部分是显式的，为对象和对象集合之间的距离测度。该系统可以生成具有SK形式的聚类和可放在BK中的规则。

ID3也给出了一种简单的信息流（图5)。由用户提供例子。系统嵌入SK—"首先选择压缩信息最多的特征"。ID3通过对要使用的特征进行排序来生成一棵决策树，以识别给定的概念。这棵决策树可以转换成规则，但在使用这些规则时，需注意规则的使用顺序是由决策树强制的。因此我们用虚线连接SK和KB。

NN中的信息流见图6。一个NN包含大量的BK和隐含在网络初始结构中的SK。它显式地使用反向传播来选择有意义的信息。

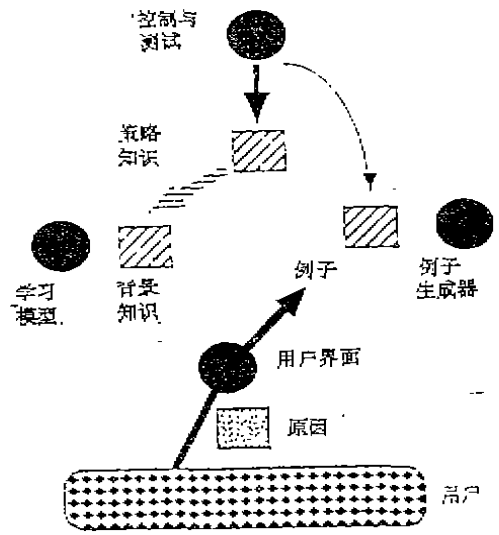

图5 ID3中的信息流

从原理上讲，由用户提供例子，NN返回关于结点的相对重要性的SK。

遗传算法体现了与前三者不同的信息流。遗传算法需由用户提供训练经验（例子)。例子也提供了某些隐式的BK。而评价函数实质上是一种SK，它生成允许分类程序根据其效率进行排序的SK。在现有两种主要的遗传算法（Michigan算法和Pitt算法）中，信息流有相当大的差别。Pitt算法中，每个个体由多条不同规则组成，因此，它表达了一种寻找问题求解方法的策略。用户可以提供一组策略，由系统从中选择一些最合适的策略。在Michigan方法中，每一个体是表示一条孤立规则的分类程序，因而是一段知识。用户提供作为BK的初始规则集合，系统通过学习产生新的BK（图7)。

对于EBL来说，需要用户提供显式的知识作为BK。同时也需要例子以及一些由操作规范约束的SK，操作规范规定了表示一条规则所需要的操作谓词。EBL生成的规则属于BK，而且它包含嵌入在例子中的SK。因此，如图8所示，EBL学习描述性的和策略性的知识。

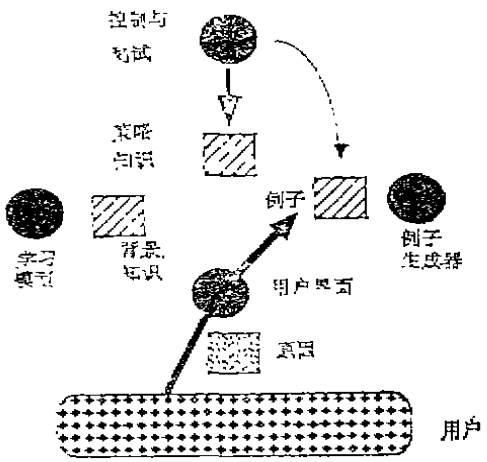

图6 NN中的信息流

概念聚类融合了EBL的符号特性和数据分析的数值特性，所以它的信息流最为复

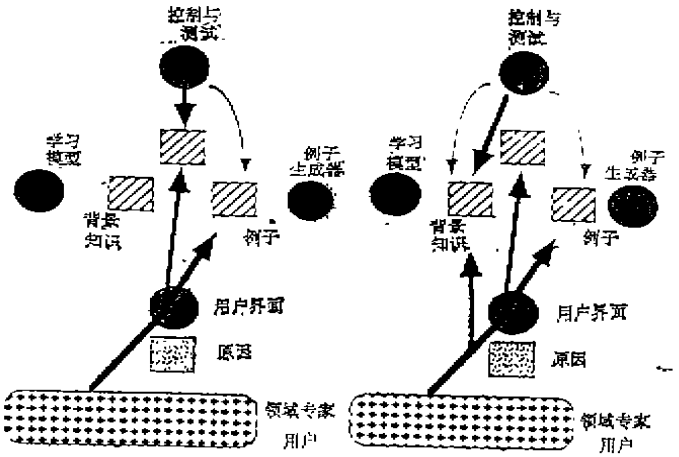

图7 遗传算法中的信息流。左图是Pitt方法；右图是Michigan方法

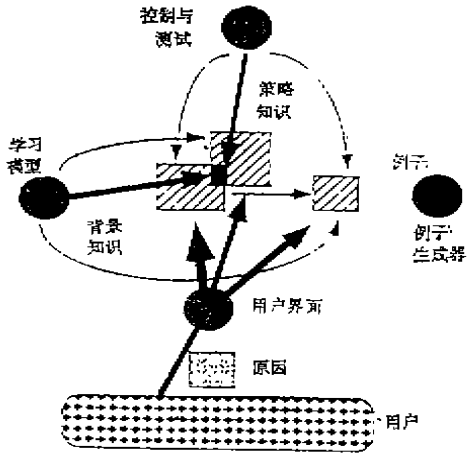

图8 EBL中的信息流

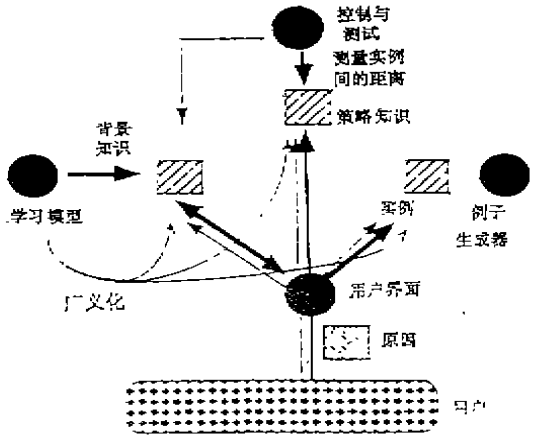

图9 KBG中的信息流

杂。我们以系统KBG为例，该系统采用一阶逻辑作为表达语言，要求用户输入例子和BK。最后，KBG产生一些有序的规则，一条规则就是一个聚类中例子的最小概括。注意图8中连接用户和BK的双箭头，这个双箭头表示由KBG产生的知识转移给用户，用户可再转换到SK。

七、结束语

本文概述了典型的ML技术和一些有代表性的ML系统。ML是一个活跃的，不断出现新概念、新思想和新方法的研究领域，同时也是一个困难的、充满争议和大量未解决问题的研究领域。这正体现了ML的重要性和生命力。总的来说，当前ML的特点是理论研究不深，应用强度不够。估计今后几年ML的研究会集中在如下几方面：a．人类学习机制与新的ML方法；b．ML的计算理论；c．多种学习方法在同一系统中的集成，ML技术与其它人工智能技术和计算机技术的结合；d．面向特定领域的应用研究。任重而道远，我们期待着ML研究的进展与突破。（参考文献见封四）