

机器学习中的概念聚类

韩建超 王红蕾 (中国人民大学信息系)

摘 要

Conceptual Clustering is an important method of machine learning, especially learning by observation. In this paper, we overview various views of conceptual clustering, mainly including extended numerical taxonomy, concept learning and concept discovery based on conceptual clustering, discussing a number of issues in conceptual clustering with several famous machine learning systems such as DISCON, CLUSTER, MITT, INDUCE, AQ15, UNIMEM, GLAUBER, IPP, and so on, analysing the differences between conceptual clustering and learning from examples, addressing two processes of conceptual clustering, concept sorting and concept description. At last, we briefly consider the levels of conceptual clustering algorithms, and summarize various search technologies used in conceptual clustering systems.

一、引言

机器学习是人工智能发展中一个十分活跃的领域。机器学习的研究就是希望计算机能够像人类那样具有从现实世界获取知识的能力。学习应是一切智能系统的重要特征之一，没有学习能力的系统都不堪称为智能系统。因此不少学者认为，如果说八十年代是专家系统的年代，那么九十年代将是机器学习的年代。

归纳学习是机器学习研究中最为困难，然而却最为诱人的一个方面。从科学哲学的观点来看，归纳能够帮助人们学习，而且也有助于对客观规律的发现，因而引起不少机

器学习研究者的高度重视。

概念聚类是机器归纳学习研究中十分重要的一种规范和技术。归纳学习实际上是一种多概念学习，是特殊到一般，个性到共性，具体到普遍的推理过程。这种概念学习与逻辑学意义上的归纳有很大区别，因此，Winston^[23]称其为“从例子中学习结构描述”，并将这种学习过程分为两个步骤：

第一步叫做分类，即根据给定例子的属性或“特性”集，正确区分哪些例子可分为一类，哪些例子不能分为一类。一般的概念学习模型只能生成一组合取的必要且充分的特性，来判断一个例子的归属。但是这样的

企图分别地探讨开放系统问题的各个方面。如，侧重于语言机制的系统，有Act1^[16]，SBA^[20]，侧重于高级接口支持的，有Act3^[14]，侧重于并行解题的，有Ether^[21]，侧重

于知识表示的，有Omega^[22,23,24]，侧重于并发面向对象的，有Act3^[14]，ABCL/1^[28]。进一步的工作还在展开之中。

(参考文献共24篇略)

分类限制太死又过于简单,因为当一个事件第一次出现时,我们并没有获取它的所有信息,因而只能让系统学会根据不完全信息进行分类,其结果往往是不可靠的,不精确的。例如,若从未见过驴,那么把它当作马就是一种合理的但不准确的理解方式。在Michalski^[6,14]等人的AQ算法中使用的相似匹配技术正反映了这种分类思想。

第二步叫做概念描述,对每一分类进行特征概括,得到一个概念的语义符号描述。描述有三种基本方法:特性匹配,抽象层次和因果关系。第一种方法中,系统抛弃一些妨碍识别新例子的特性,又从例子中找出某些未被注意的特性。第二种方法认为概括描述在于进一步以抽象层次中的限制来替换特性限制。第三种方法用于类中有着因果关系属性情形,通过对因果性质的分析而形成概括描述。

概念聚类技术既可用于归纳学习特别是概念结构学习的第一步——分类中,也可用于其第二步——概念描述。本文将简要介绍概念聚类技术的发展,和概念聚类的若干问题,讨论概念聚类算法中聚类的不同抽象层次,并分析几个有代表性的概念聚类系统中采用的实现技术。

二、概念聚类的发展

概念聚类的基本思想是把实例按一定的方式和准则进行分组,如划分不同类,不同层次等,得到的分组代表着不同的概念,从每个分组中可以诱导出一个概念的语义符号描述,这也正是概念聚类区别于普通聚类的重要方面。

概念聚类的历史并不长,Michalski 1980年首次提出概念聚类^[4],并把它作为机器学习研究的一种新方法,从此日益得到人们的关注,并迅速成为机器学习研究中非常活跃的技术。在Michalski之前,早期的研究已经触及到概念聚类问题,只是没有冠以概念聚类的名称,如Wolff设计的Mk10系统^[15],这是一个组块的层次化数据压缩系统,它基

于共同特性值对实例进行分类,然后对这些分类进行概括和抽象。Michalski以后,概念聚类学习系统频频涌现,至今方兴未艾。

1982和1983年,Lebowitz^[10,13]设计了UNIMEM和IPP系统,采用所谓的“基于概括的记忆”(generalization-based Memory)的方法,记忆实例的特征,把具有相同特征的实例归入同一个类中,实现了实例的聚类。1983年,Michalski和Stepp研制了CLUSTER/2系统^[6,7]把实例与已发现的或已知的概念进行比较,根据它们的异同进行聚类,而不是根据某一个属性值的差异进行分类。1984年,Langley与Sage开发了DISCON系统^[3],该系统通过构造最优分类树,形成层次化概念的聚类。同年Fisher研制成功了RUMMAGE系统^[10],它与DISCON的不同在于它在特征空间中采用了非耗型搜索。这两个系统都非常类似于Quinlan的ID3^[8]。此后,Stepp在CLUSTER/2的基础上又设计了CLUSTER/S^[12],用于结构化实例聚类,Langley和Simon等人在MK10算法的基础上推出了GLAUBER系统^[18],用于发现定性结构定律,如酸碱反应生成盐等。1986年,Stepp与Michalski给出了包含背景知识和分类目标的聚类算法^[12],1987年,Mogensen的CLUSTER/CA通过使用目标相关网络GDN (Goal Dependency Networks)^[17],采用面向目标的方法形成结构化对象的聚类。随着旧算法的逐步优化和新系统的不断创建,概念聚类的技术和方法也日趋成熟和完善。

三、基于数值分类的概念聚类

Fisher和Langley认为^[8],任何聚类方法,不论是概念聚类还是数值分类,都可以抽象成如下形式:

给定: 一组符号描述的对象 O ;

寻找: O 的一个划分 c_1, c_2, \dots, c_n ,使得这些类相对于某个聚类质量函数具有较高的质量(不一定最优)。

一个划分称为一个聚类,因而不同于致

学意义上的划分。 c_1, c_2, \dots, c_n 称为该聚类的类。数值分类的任务在于将符号描述的对象集划分成变量-值(variable-value)对的集合, 聚类质量是聚类中各个类的正数。设 C 是一个聚类, c_1, c_2, \dots, c_n 是 C 中的类, 记为 $C = \{c_1, c_2, \dots, c_n\}$, 则有:

$$Quality(C) = f(c_1, c_2, \dots, c_n).$$

数值分类的目标是寻找某个聚类 C^* , 使得:

$$Quality(C^*) = \max\{Quality(C)\}.$$

数值分类的局限性在于所得到的类难以用人们易于理解的概念语言较好地加以刻画。1980年, Michalski对数值分类技术进行了扩展^[4], 强调聚类中每个类的概念表达, 并认为聚类的质量不仅依赖于其中的每个类, 而且也与该类所对应的概念表达(如表达的简单性, 易理解性等)以及所表达的概念与类之间的覆盖映射(如概念与类的合适程度, 一致程度等)有关。因此聚类质量函数应为:

$$Quality(C) = f(c_1, c_2, \dots, c_n, CONCEPTS),$$

其中CONCEPTS是可用来说描述对象类的概念集。

按照Everitt^[1]的观点, 数值分类的方法主要包括如下三种:

1. 平面分类(Flat taxonomy): 即对输入对象集 O 进行划分, 形成若干个互不相交的类, 类与类之间是对等的, 不存在层次关系。形成的类的个数 k 由用户直接提供, 系统对整个对象集进行严格搜索, 构造全局最优的 k -划分。

2. 层次分类(Hierarchical taxonomy): 即按层次结构对 O 进行划分, 形成分类树, 其中树根是整个对象集, 树叶是单个对象, 中间结点表示不同层次上的类。分类树的构造可采用自顶向下的方法, 也可采用自底向上的方法。构造过程依赖于一系列“局部”决策, 从而可降低计算复杂性。在自顶向下的划分过程中, 每一步分裂与树中非祖先结点无关。

3. 组块分类(Clumping taxonomy): 类似于平面分类, 但形成的类可以相交, 即允许同一个对象属于多个不同的类。

类似地, Fisher和Langley^[3]将基于数值分类的概念聚类方法也分为三类:

1. 平面概念聚类(Flat concept clustering): 对应于平面数值分类, 参数 k 规定了概念的大小, 如CLUSTER/2中的划分模块(Partitioning Module)。

2. 层次概念聚类(Hierarchy concept clustering): 对应于层次数值分类。如CLUSTER/2中的层次建造模块(Hierarchy-building Module), 采用自顶向下的方法建造分类树, 在对每个中间结点进行分裂时, 调用划分模块, 并选择最优的划分。DISCON和RUMMAGE系统也属此类, 而MK10系统则采用自底向上的方法实现层次概念聚类。

3. 组块概念聚类(Clumping concept clustering): 对应于组块数值分类, 如UNIMEM, IPP, 和GLAUBER等, 它们所构造的分类模式等价于闭合的非循环图, 其中每个结点表示一个类, 一个对象可包含在多个类中。

四、基于概念聚类的概念学习

概念聚类也可作为从观察中学习(learning by observation)的技术, 用来学习新的概念, 这是相对于从例子中学习(learning from examples)而言的。基于概念聚类的概念学习(以下简称概念聚类)与从例子中学习都旨在形成能概括对象类的概念描述, 但从例子中学习的方法要求教师或环境指定哪个对象应归属于哪一类, 学习系统的主要任务是构造能刻画这一类的描述。而在概念聚类中, 学习系统有双重任务, 即首先将对象集划分成若干类, 然后对每一类进行概括描述。第一个任务称为聚集(aggregation)问题, 强调如何将初始对象集进行划分, 形成若干类, 每个类定义为扩展的对象枚举集, 这是概念聚类区别于从例子中学

十
三
三
号
发
号

习的重要标志。第二个任务称为刻画(characterization)问题, 强调如何确定扩展表示的对象类, 以形成概念, 这一点与从例子中学习是相同的。概念聚类过程必须强调刻画问题, 因为聚类质量依赖于可用来描述对象类的概念。

概念聚类的两个任务并不是相互独立的, 它们是概念学习过程的两个不可分割的步骤。例如, GLAUBER就是根据定义在对象集上的最常出现的关系形成类, 然后相对于其它关系刻画这些类。MK10也使用了非常相似的技术。

RUMMAGE和DISCON使用用户给定的属性表形成对象集上各种可能的划分。在RUMMAGE中, 系统根据每个属性的不同值形成多个划分, 如果有n个属性, 则形成n个划分。然后相对其它属性确定根据某个属性的不同值形成的划分的质量, 并选择其最优者。尔后对每个类, 递归重复上述过程

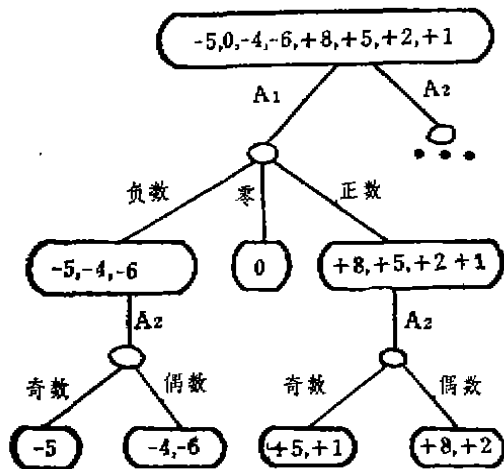


图1 RUMMAGE与DISCON中的聚类

形成层次结构, 从而解决聚类问题, 如图1所示。DISCON类似于RUMMAGE, 但DISCON并不构造每个类的严格描述。

在图1中, 对象集 $O = \{-5, 0, -4, -6, +8, +5, +2, +1\}$, 有两个属性 A_1 和 A_2 分别取值为: $A_1: \{\text{负数}, \text{零}, \text{正数}\}$; $A_2: \{\text{奇数}, \text{偶数}\}$ 。因而可形成两个划分。假定选

择依据 A_1 的划分, 则得到图1所示的分类树。经过概念描述(即刻画)可形成如下层次概念:

```

整数{
  正整数{
    正奇数;
    正偶数};
  零;
  负整数{
    负奇数;
    负偶数}}
    
```

在CLUSTER/2中, 系统首先随机选择N个对象作为“种子”, 每个种子看作某类的正例, 其余看作该类的反例。然后, 对每个类形成最一般的判定描述(maximally-general discriminant descriptions), 每个描述覆盖该类中的种子, 而不覆盖其它种子, 当然也可覆盖若干非种子对象。一旦形成所有对象的分类, 则废弃这些最一般判定描述, 而从每个类中导出最特殊的特征描述(maximally-specific characteristic descriptions)。

在Hanson^[11]的WITT系统中也采用了概念聚类的学习方法, 主要用于对象集的划分。整个系统主要由三部分构成: 线性目标栈, 用来存放系统目标状态下的各参数; 假设生成器, 生成分类规则, 并将所有假设规则进行排序, 和转移矩阵, 用于考察实例(对象)的凝聚程度。

WITT有三种目标状态: ①目标猎取——把实例加入现有的对象类; ②预种子(proto-seeds)猎取——系统在特征空间中搜索相似的实例, 并把相对密集的实例归并到一起, 形成预种子; ③预种子合并——把现有的对象类与预种子合并或预种子之间合并。三种状态下各有一个算法过程, WITT在各个状态下循环操作, 直至不能从现有目标达到新的目标, 然后从目标栈中取出下一种状态, 考察适合于新状态的假设。

图2所示是WITT的聚类过程。系统首先通过一个预分类过程, 形成第一批预种

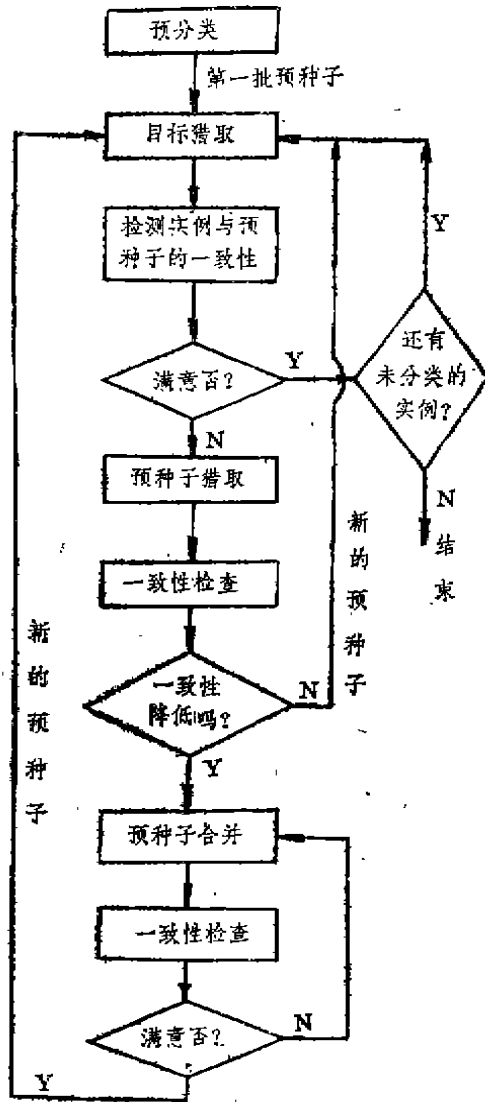


图2 WITT的算法流程

子。设 H 是某个信息量测度函数， x 和 y 是两个对象（实例），则 x 和 y 合并到一起的信息丢失测度（或称信息转移度） $t(x, y)$ 为：

$$t(x, y) = H(x) + H(y) - H(x, y).$$

如果 $t(x, y)$ 相对较小，则 x 与 y 可属同一预种子。

获取第一批预种子后，系统进入目标猎取状态，为每个预种子寻找新的可加入成员。这个成员必须满足不影响概念一致性的条件，即该成员加入后，每个预种子仍具有较高的凝聚度。设 c_i 是一个预种子， x 是一对

象，对所有 $c_j (c_j \neq c_i)$ ，计算 $t(x, c_j) / t(x, c_i)$ ，如果其值较高，则说明 x 与 c_i 的一致性程度最高，因而不将 x 加入 c_i 。如果目标猎取失败，则恢复到预种子猎取状态，获取新的预种子。如果新的预种子降低原来的一致性程度，则重新进入目标猎取状态；否则说明存在一些预种子太接近，难以区分，这时进入预种子合并状态。重复上述过程，直到所有实例（对象）均已分类为止，就形成一批较好的预种子，即对象类。

五、基于概念聚类的概念发现

1987年，Stepp提出了概念聚类的另外两种观点^[18]：概念分类与概念发现，前者即所谓的扩展型数值分类，目的在于把对象集划分成不同的类，它关心的是如何把对象分类，形成不需要保持其描述类，也就是说，类的形成与类的描述无关。后者把实例对象作为形成概念的催化剂，系统必须不断地检查和维护类的描述，每一类的成员都受到用以导致这一描述结果的概念的限制。在数值分类中，大都基于统计的方法，测度两个对象之间的距离，因而分类质量是实例对的函数： $FS(e_1, e_2)$ 。概念分类与数值分类的不同主要在于所形成的类必须具有能导出概念解释的特征。概念分类的方法主要是基于属性的方法，通过属性及其值之间的相互关系形成分类，如WITT就使用类内、类间的凝聚性测度类的质量，而COBWEB^[19, 20, 21]则使用类的效用 (utility) 确定如何对实例进行划分。因此，概念分类的质量不仅依赖于实例之间的距离，而且还依赖于分类环境，表示为 $FA(e_1, e_2, environment)$ 。

由于概念分类中不关心分类中每一步的语义描述，因而往往只能得到一些零乱的概念特征属性。如何选取这些特性形成概念描述是概念分类的难题。对此，Michalski提出了多概念的思想^[5, 7]，他认为可以把这些特征属性表示成析取形式，然后从中选择一部分来形成某一概念描述，这样随着选取的不同可以得到多个概念。但用这种方法得到的

概念不一定令人满意，因为从析取式中截取一段不一定是概念的本质属性，这样形成的概念描述就可能与相应的类有较大偏差。

概念发现的中心任务在于新概念的描述，聚类的关键问题是如何提高描述一般化的质量。对类的描述必须符合定义的各种约束条件，同时需要满足概念的定性测试，象 CLUSTER/S, INDUCE/2^[12], AQ15^[14] 等系统利用实例类中样本属性-值对来激发、约束概念描述，同时通过对描述质量的测度和评价反过来引导实例的划分。这样，概念聚类的质量不仅依赖于实例间的关系、环境，而且还依赖于用以描述的概念，表示为：

$FC(e_1, e_2, environment, concepts)$

而概念的运用则与系统的背景知识有关。因此在概念发现中必须有背景知识，它提供各种约束条件、启发式规则、评价函数等，并直接影响概念的有效性。可以说没有背景知识，概念发现就退化为概念分类。

概念发现不同于概念分类的另一方面是，系统形成的类可具有层次结构，并且用来描述每个类的概念可含有实例对象中没有发现过的新属性。一般地，用于概念发现的概念聚类过程可描述为^[12]：

给定： 一组观察对象；

一组用来刻画对象的属性；

背景知识库；

寻找： 对象类及其层次结构，以及每个类或子类的描述，并满足最优聚类准则。

其中背景知识包括分类目标，分类评价准则；演绎和归纳推理规则；问题约束；属性的类型和值域说明；属性间的层次及其它关系说明；新属性的形成结构等。系统的目标、子目标以及各相关属性之间的关系构成目标相关网络GDN，通过GDN上的推理，可诱导出新的属性，如INDUCE/2, CLUSTER/S. Stepp和Michalski进一步给出了用于概念发现的聚类算法RD和CA，并详细讨

论了聚类质量测度函数LEF。

基于概念聚类的概念发现过程不是一个单纯的归纳推理过程，也包含依据背景知识的演绎推理来确定实例的本质属性以及实例类的概念描述，如CLUSTER/CA系统。一个系统如果具有足够的背景知识操作和使用背景知识的演绎机制，就能使实例的属性之间实现较大范围的适当转换，有助于概念形成。例如，背景知识中的启发式规则可以使系统从一个（或一些）属性的值预测另一个（或一些）属性的值；又如在背景知识中提供属性的定义，就能使系统以易于理解的方式描述多个概念，形成新的属性。

在AQ15系统中采用了覆盖截尾技术和相似性匹配构造类的概念描述。其中概念用属性值的析取式（称为选择子（selector））的合取形式（称为复合（complex））来表示。每个复合用一对参数（ t, u ）来测度，其中 t 表示该复合所可能覆盖的实例总数，而 u 表示只被该复合覆盖的实例数。复合按 t 的值降序排列，参数 t 可看作是作为概念描述的复合的典型性度量。 u 值最高的复合，则认为它描述了该概念最典型的一些实例；若 u 值很低，则认为该复合覆盖了概念中的例外情况。

AQ15采用相似性匹配技术形成概念描述，通过测试实例与概念识别规则条件部分的相似程度，选择所有候选概念中相似度最高者。为了避免形成的概念描述过于复杂，系统使用覆盖截尾技术简化概念表示。一个覆盖（cover）表示为一组复合的析取式，这些复合按 t 值降序排列，从中截掉末尾 t 值最小的复合，这样概念描述就被截短了。覆盖截尾与相似性匹配是相辅相成的。图3所示说明了覆盖截尾技术的应用，设 $CPX_i, i=1, 2, 3, 4$ ，均为复合， $CPX_1 \vee CPX_2 \vee CPX_3 \vee CPX_4$ 构成一个覆盖。图中数对为 (t, u) ， a, b, c 为截取点。在 c 点，截去 t 值最小的 CPX_4 ，在 b 点截去 t 值次低的 CPX_3 ，当在 a 点截去 CPX_2 后，只剩下一个最佳复合

CPX₁, 它不仅 t 值最高, 而且 u , t 值也很接近, 描述了概念的典型特征。

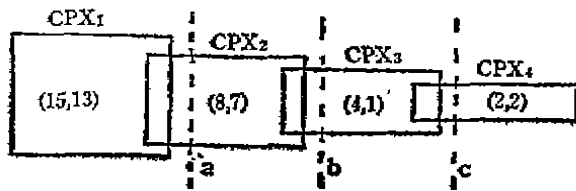


图3 AQ15的覆盖截尾技术

六、概念聚类算法的分级

目前在机器学习研究中使用的概念聚类算法多种多样, 每种算法适合于不同的领域和用途。Stepp^[13]按照实例类的质量测度方法, 以及聚类的两个方面——分类与描述和背景知识的运用等准则将这些算法分为七个级别, 级别越高, 形成的类的描述(概念)越抽象, 概括性越好, 算法的启发式搜索越深入, 执行效果也越好。

0级: 基于统计的定性测度 (statistic-based quality measure), 无概念解释。它包括传统的数值分类, 有一个统一的标准(如相似矩阵)处理每个属性。输出结果仅是各个实例类, 需要用户或其它系统对结果进行解释。

1级: 基于统计的定性测度, 有概念解释。它在进行数值分类后, 紧接着有一个实例学习系统对实例类进行概念描述, 如Michalski的AQ系列算法, Quilan的ID3系列算法等。

2级: 基于属性的定性测度 (attribute-based quality measure), 无概念解释。这类系统采用格式塔 (Gestalt) 信息论模式对属性和实例集进行测度, 形成最优分类, 但不关心每类所代表的概念, 如WITT及Rendell的基于效用的聚类系统PLS^[10]。

3级: 基于属性的定性测度, 有独立于聚类形成的概念解释(描述)。这类系统的分类方法类似于2级系统, 但在分类形成后, 另有一个独立的过程对实例类进行概括抽象, 构造其概念描述, 如COBWEB, UNIM-

EM, IPP, GLAUBER系统等。

4级: 基于属性的定性测度, 有统一的分类与描述过程, 但没有背景知识, 并且不进行演绎推理, 系统只用到最一般的内部聚类目标和启发式, 如DISCON, RUMMAGE, CLUSTER/2等。

5级: 基于概念的定性测度 (concept-based quality measure), 这类系统类似于4级系统的算法, 但有简单的背景知识, 能进行必要的演绎推理, 导出一些新的属性, 启发式和聚类目标, 如INDUCE/2, CLUSTER/S, AQ15等。

6级: 基于概念的定性测度, 有丰富的背景知识。这类系统具有5级系统的功能, 但5级系统通常属于“平面聚类”, 而6级系统能够处理结构化对象, 因而属于“层次聚类”, 如CLUSTER/CA等。

七、概念聚类中的搜索技术

基于概念聚类的学习策略与从例子中学习的主要区别在于前者包含概念形成(实例分类)和概念描述两个过程。目前的大多数概念聚类算法在这两个过程中都采用了不同的搜索技术^[3]。

1. 分类空间的搜索

对分类空间的搜索旨在形成合适的实例分类, 这种搜索技术主要分为两类:

①**平面空间搜索。**即分类过程一步完成, 这种搜索技术要求用户指定用于描述实例的属性 and 属性值, 通过对属性-值对的选择形成实例集的不同划分, 然后对每种划分进行评价, 选择其中最优者, 如GLAUBER, MK10, UNIMEM, IPP, CLUSTER/2等。

②**层次空间的搜索。**即分类过程迭代完成, 形成实例集的分类树。层次空间的搜索技术可从搜索方向和搜索控制两个方面来分析。

- 搜索方向, 主要包括自顶向下法、自底向上法、和中间开始法 (Middle-out)。

- 搜索控制, 主要包括耗尽型搜索和非耗尽型搜索两种。

2. 描述空间的搜索

描述空间又称特征空间, 对这种空间的搜索旨在形成每个实例类的概念描述。对描述空间的搜索技术, 可从如下四个方面加以分析。

① **搜索模式**。对实例类的描述, 一般都是先形成一个初始描述, 然后根据类中的元素和类间的关系逐步优化。依据初始描述的不同, 可将搜索技术分为三种:

- 从非常特殊的初始描述开始, 逐步求精, 形成更一般的描述覆盖整个实例类。这种模式称为概括学习。

- 从非常一般的描述开始, 可能覆盖有反例, 逐步寻找更特殊的描述, 恰好覆盖类中各实例。这种模式称为判定学习。

- 从一随机描述开始, 通过一般化和特殊化操作, 双向搜索, 期望获得满意的描述。这种模式类似于 Mitchell 的版本空间 (version space) 策略^[8]。

② **搜索策略**。根据搜索方向可分为深度优先搜索和广度优先搜索。根据搜索控制方式可分为耗尽型搜索和非耗尽型搜索。非耗尽型搜索, 如 CLUSTER/2 中的束搜索 (beam

-search), 要求有某种评价函数对生成的各种描述进行排序, 因此评价函数不同, 搜索结果亦不同。

③ **驱动方式**。根据搜索的驱动方式可分为数据驱动和模型驱动两种技术。在数据驱动方式中, 数据 (即实例) 引导对描述空间的搜索; 而在模型驱动方式中, 使用某种模型生成相应类的初始描述, 仅在对描述进行评价时使用数据, 如 UNIMEM, GLAUBER 等。

④ **知识增长**。根据系统中的观察实例与已学习过的概念是否同时参与概念描述的生成过程, 可将描述空间的搜索技术分为非渐增式和渐增式两种。非渐增式搜索属于“就事论事”, 只考虑实例集, 不关心环境变化, 主要用于模拟科学数据分析, 如 DISCON, RUMMAGE, GLAUBER, MK10 等。渐增式搜索主要根据与环境的不断交互, 随时吸收新的实例和概念来形成新的概念, 如 UNIMEM, IPP, CLUSTER/CA, AQ15 等。一个概念聚类系统能否实现知识增长, 不仅与描述过程有关, 也与分类过程有关。

(参考文献共23篇略)



第二次全国计算机逻辑学术研讨会圆满成功

由计算机学会直属理论计算机科学学组委托华东化工学院计算机系主办、苏州大学工学院协办的第二次全国计算机逻辑学术研讨会于1991年5月在苏州大学召开, 参加会议的有70余名代表。国内知名学者莫绍揆、陈火旺、孙永强、罗铸楷分别作了专题报告, 会上交流了论文50余篇, 并且与1991年在上海市举行的第五次全国“双B”代数学术会议联合出版了《“双B”代数和计算机逻辑论文集》。会议还就计算机与逻辑的关系作了专门讨论。为促进计算机科学与逻辑的联系与结合, 会议决定成立计算机逻辑学组, 属理论计算机科学学组领导。经过广泛征求意见与讨论, 会议选举南京大学徐洁磐教授为学组组长, 华东化工学院沈百英教授、北京航空航天大学赵沁平副教授为副组长。会议初步决定下次学术讨论会将于1993年在太原市召开。

第二次全国计算机逻辑学术研讨会会务组供稿