

基于云模型的隶属概念判定算法的改进

张家精¹, 王焕宝¹, 陈金兰², 倪友聪¹

(1. 安徽建筑工业学院 数理系, 安徽 合肥 230601;

2. 安徽理工大学 资管学院, 安徽 淮南 232001)

摘要:在人工智能领域中,李德毅教授等提出的定性定量不确定性转换模型——云模型,把概念的模糊性和随机性完全集成到一起,构成定性数据和定量数据的相互映射。但是,在利用云模型进行隶属概念的判定的算法中,不论对象是否明确属于某个概念,一律进行不确定性判定,过分地强调了模糊性和随机性。这不符合人类的认识规律,而且增加了算法的时间复杂度,甚至导致隶属概念判定的明显错误。该文根据人类认识规律对上述算法进行了改进。对那些可以明确确定属于某概念的对象,不再进行不确定性判定,直接被认定为属于该概念,只对处于多个概念边缘的不能确定属于某个概念的对象进行不确定性判定,使其符合人类的认识规律,降低了算法的时间复杂度。

关键词:云模型;隶属概念;模糊性;随机性;时间复杂度;定性;定量;不确定性

中图分类号:TP18

文献标识码:A

文章编号:1673-629X(2007)10-0065-04

An Improved Algorithm for Determining Membership Conception Based on Cloud Mode

ZHANG Jia-jing¹, WANG Huan-bao¹, CHEN Jin-lan², NI You-cong¹

(1. Dept. of Mathematics and Physics, Anhui Inst. of Architecture & Industry, Hefei 230601, China;

2. Resource Exploiting and Management Eng. Dept., Anhui Univ. of Sci. & Techn., Huainan 232001, China)

Abstract: In the domain of artificial intelligence, uncertainty transformation mode between the qualitative data and the quantitative data put forward by professor Li Deyi etc., that is, cloud mode, bring the fuzziness and the randomness of conception together, realizing the reciprocal mapping between the qualitative data and the quantitative data. But, in the algorithm to determine membership conception by making use of cloud mode, all objects would be determined indefinitely, despite that they belongs to certain concept explicitly. Therefore the fuzziness and the randomness are emphasized excessively, which dose not accord with cognitive rule of the human being and increases time complexity of the algorithm, and even causes the obvious wrong determination of membership conception. This paper improves the above algorithm according to cognitive rule of the human being. Some objects that belong to certain conception explicitly are no longer determined indefinitely but are put to the conception directly. As to the objects which lie in verge of two or multi-conceptions and don't belong to certain conception certainly, they will be determined indefinitely according to the cloud mode. Such an improvement will make the algorithm more identical to cognitive rule of the human being and decrease time complexity of the algorithm.

Key words: cloud mode; membership conception; fuzziness; randomness; time complexity; qualitative; quantitative; uncertainty

0 引言

云模型是李德毅教授在传统模糊集理论和概率统计的基础上提出的一种定性定量不确定性转换模型^[1]。云模型把定性概念的模糊性和随机性完全集成到一起,构成定性和定量相互映射,作为知识表示的基

础,在不确定性人工智能的应用和研究方面受到广泛的关注。

在定性定量不确定性转换过程中,概念的划分及隶属概念的判定是两个关键的步骤。关于隶属概念的判定在有关论文和书籍中给出了算法的描述^[2~4]。但是,这些算法不管某对象隶属于某概念的隶属度是多大(甚至隶属度为1,即该属性确定属于某概念),都要拿出来和其它概念一起进行不确定性判定。这不完全符合人类对事物的不确定性认识规律,且算法复杂度,甚至导致隶属概念的错误判定。因此,这些算法需要进行改进,避免隶属概念的错误判定,使其更加符合

收稿日期:2007-01-12

基金项目:安徽2005年硕博科研启动项目(2005110126)

作者简介:张家精(1977-),男,安徽霍邱人,讲师,硕士,研究方向为不确定性数据挖掘、数据库技术;王焕宝,副教授,博士,研究方向为信息安全、计算机网络安全。

人类的认识规律,降低其时间复杂度。

1 隶属概念判定算法的分析

参考文献[2]中和参考文献[5]中对隶属概念判定算法进行了如下描述。

参考文献[2]对描述隶属概念判定算法的描述如下:

算法 1^[2]:

输入:属性值 data

概念集 $A\{A_1, A_2, \dots, A_n\}$

输出:隶属概念 CONCEPT

BEGIN

$n = |A|$; // 计算概念个数

$\omega = 0$

FOR $i = 1$ to n step 1 DO

{ $u_i = X_CLOUD(A_i, data)$; // 利用 X 条件云发生器计算隶属程度

$\omega = \omega + u_i$;

}

$x = \text{random}(0, \omega)$; // 产生 $(0, \omega]$ 之间的均匀分布的随机数

$y = 0$;

FOR $i = 1$ to n step 1 DO

{ $y = y + u_i$;

IF $x \leq y$ THEN BREAK; // 根据随机数所属范围确定 data 所属概念

}

CONCEPT = A_i ;

END

下面对该算法进行分析:

虽然云模型理论强调模糊性、随机性,但是它更强调的是符合人类对客观事物认识的客观描述。该算法却片面地强调了云模型理论的模糊性、不确定性,忽略了人类在认识客观事物时的准确性和确定性,不管事物隶属于某个划分概念的隶属度是 1(确定属于某概念),还是 $0.9\dots$,一律用不确定的算法去判断隶属概念,不但不符合人类对客观事物认识的规律,而且增加了算法的时间复杂度,甚至导致隶属概念的错误判定。

首先,如图 1 所示对于 A 、 B 两个划分概念, a 和 c 两个事物的判定是确定的,即它们分别属于 A 和 B 两个概念。因为它们正处于或靠近这两个概念的概念中心。对这种隶属于某概念的隶属度很高甚至是 100% 的事物,还用判断什么,直接就可以认为它属于该划分概念。而由于 b 既可属于 A 也可属于 B ,不能明确判定属于哪个概念,因此人们在判断时就要加以思考,根据各种情况和经验去进行不确定判断。因此算法 1 中的不分什么情况一律加以“思考”、进行判断与人类对客

观事物的认识过程不符合,从而导致如图 2 所示的确定事物的不确定判定。尽管 a 事物隶属于 B 概念的隶属度很小, a 事物还是有可能被判定为隶属于 B 概念,这明显是错误的。

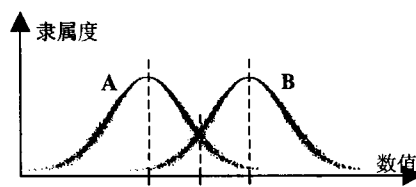


图 1 事物隶属概念图

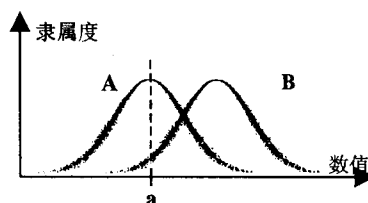


图 2 确定事物的不确定判定

其次,由于该算法采用累加触发的方式来判断事物隶属于某个概念,即先求出事物隶属于各概念的隶属值并将其累加到 ω ,然后生成 0 到 ω 之间的均匀随机数 x ,最后,再次将各概念的隶属累加到 y ,最终触发累加值 y 大于随机数 x 的那个概念即是事物隶属的概念,这种算法中,如果事物对于某概念的隶属值很高,也很大可能被划分到隶属值较小的概念中去,其原因有二:(1) 由于 x 是 0 到 ω 之间的均匀随机数,可能靠近 ω ,也很可能靠近 0,这样隶属值很小的概念将触发,即事物被判定为隶属于较低隶属度的概念。(2) 即使 x 靠近 ω ,由于累加概念的隶属度的顺序是一定的,很可能前面一个较大的隶属度累加后 y 接近 x 但并未触发 $x \leq y$,而紧接着的较小隶属度的概念却触发了 $x \leq y$,从而把事物判给较小隶属度的概念或因累加的顺序而导致不确定事物的确定判定,这是不符合人类认识规律的,甚至是错误的。

最后,由于在现实世界里进行概念划分时,处于两种或多种划分概念的事物应该是处于划分概念的边缘。一般情况而言,这些事物毕竟属于少数,而为了这些少数事物的不确定性的判定,把整个论域中的事物都一一拿来来进行如此复杂的判断(从算法 1 中产生 $(0, \omega]$ 之间的均匀分布的随机数的一行开始往下到算法结束),其中还有一个 For 循环,其算法的时间复杂度 $T(n) = O(n \times k)$,其中 k 为划分概念的个数, n 为事物的个数。显然增加了时间的复杂度。

参考文献[5]中认为隶属概念的判定有两种方法:随机判定法和极大判定法。其中,随机判定法是根据属性值对概念集中所有概念的隶属程度比例关系,

在隶属程度大的前几名中随机选择隶属概念。如果属性值对某一概念的隶属程度大,则随机选择此概念的概率也就大,属性值隶属于它的可能性就大于其它概念。而极大判定法则根据属性值对概念集中所有概念的隶属程度的大小,选择最大隶属程度的概念作为隶属概念^[5]。

极大判定法是随机判定法的一个特例。随机判定法算法的描述如下:

算法 2:

输入:属性值 x

基于云模型的概念集 $C\{C_1(Ex_1, En_1, He_1), C_2(Ex_2, En_2, He_2), \dots, C_m(Ex_m, En_m, He_m)\}$;

候选概念的个数 k 。

输出:隶属概念 CONCEPTS

算法步骤:

BEGIN

CONCEPTS = \emptyset ;

FOR $i = 1$ TO m DO

{ //利用 X -条件云发生器计算 x 对每个概念的隶属程度

$\mu_i = X_CLOUD(C_i, x)$;

CONCEPTS = CONCEPTS $\cup \{(C_i, \mu_i)\}$

}

//依据隶属度大小,选择前 k 个概念

CONCEPTS = MAX_K(CONCEPTS, k);

END

此算法对算法 1 中累加方式进行了改进,采用的是根据属性值对概念集中所有概念的隶属程度比例关系,在隶属度较大的几个概念中随机选择概念,属性值对某一概念的隶属程度大,则随机选择此概念的概率也就大,属性值隶属于它的可能性就大于其它概念,避免了事物被较大可能判为较小隶属度的概念的缺点。但此算法没有改变算法 1 中的第一种情况描述的不论隶属度大小一律通过判定算法进行判定的缺点。

2 隶属概念判定算法的改进

根据上述分析算法 2 优于算法 1,但还存在不足,所以这里采用算法 2,并对其进行改进。改进后的算法描述如下。

算法 3:

输入:属性值 x

基于云模型的概念集

$C\{C_1(Ex_1, En_1, He_1), C_2(Ex_2, En_2, He_2), \dots, C_m(Ex_m, En_m, He_m)\}$;

候选概念的个数 k 。

最大隶属度与次大隶属度之间的差值的阈值 δ

输出:隶属概念 CONCEPTS

算法步骤:

BEGIN

CONCEPTS = \emptyset ;

MAX = 0; SECOND = 0; //存放隶属度最大和次大概念的序号
FOR $i = 1$ TO m DO

{ $\mu_i = X_CLOUD(C_i, x)$; //利用 X -条件云发生器计算 x 对每个概念的隶属程度

IF($\mu_i > \text{MAX}$) //判断所得隶属度是否大于前面所有隶属度的最大值

{ //是,则将 SECOND 赋值为 MAX,再将 μ_i 标记为最大

SECOND = MAX;

MAX = μ_i ;

}

CONCEPTS = CONCEPTS $\cup \{(C_i, \mu_i)\}$

}

IF($(\mu_{\text{MAX}} - \mu_{\text{SECOND}}) > \delta$

//判断最大隶属度与次大隶属度之间的差值是否超过阈值 δ

{ //超过阈值 δ 可直接将 μ_{MAX} 定为隶属概念

CONCEPTS = C_i ;

}

ELSE

{ //依据隶属度大小,在前 k 个概念中随机选择一个作为隶属概念

CONCEPTS = MAX_K(CONCEPTS, k);

}

END

该算法的改进思路为:一般情况下,人们判断某事物属于哪种概念时,并非所有的事物都是模糊的亦此亦彼的属于两个或多个概念,相反大部分事物可以明确的属于某个概念,因此,对于那些明确属于某概念的事物不再进行不确定性判断,而直接将其判定为属于某概念。只对边缘事物进行不确定性判断,如图 3 所示落在 (a, b) (c, d) 区间,即虚线中间的部分的事物。

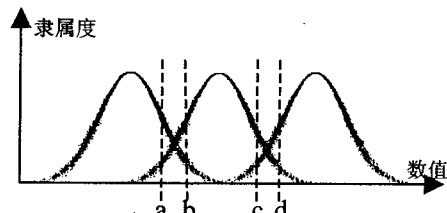


图 3 边缘事物的模糊判断(I)

该算法的关键是如何判断某事物是否明确属于某个概念。这里通过判断某事物属于各概念的隶属度的最大值和次大值之间的差值是否超过阈值 δ 。根据正态分布的“ 3σ 规则”, δ 值可通过算法根据各概念(云模型)的三个数字特征和它们之间的距离 S 或交合度 $S - 3(En_i + En_j)$ 自动生成(其中, S 为两云模型之间的距离, En 为云模型的熵,如图 4 所示),或根据经验人

工给出。

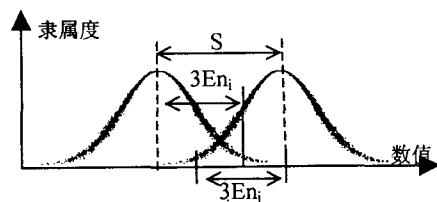


图 4 边缘事物的模糊判断(II)

3 改进后的算法与原算法的比较

改进后的算法 3 与算法 1、2 相比有以下优点:

(1) 算法 3 比算法 1 和算法 2 更符合人类认识事物的规律。算法 3 只对那些少数处于各概念边缘且和邻近概念交合的地方的事物进行不确定性判断,而不是对每个事物进行不确定性判断。

(2) 算法 3 比算法 1 和算法 2 对事物属于哪个概念的判断更准确。虽然,云模型强调随机性和模糊性,但并不是没有规律的没有标准的把一个 100% 或 90% 多的属于 B 的事物判定为属于 20% 或者 10% 属于 A 或 C 的概念,而算法 3 把那些相对比较明确的属于某概念直接判给它应该属于的概念,这样就可以避免确定属于 B 概念的事物被判给隶属度较小的 A 或 C 概念。这里通过算法 1、算法 2 和算法 3 对 $x(20)$ 这个对象属于概念集 $\{\text{cloude1}(10.0, 5, 0.2), \text{cloude2}(20.0, 5, 0.2), \text{cloude3}(28.0, 5, 0.2), \text{cloude4}(35.0, 5, 0.2)\}$ 中哪个概念分别进行了 100 次判定,其中四云概念图及三判定算法的结果如图 5、图 6 所示。

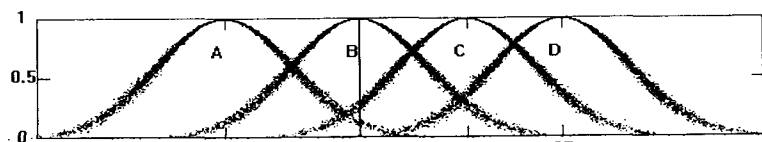


图 5 四云概念图

从四云图(图 5)中可以看出 $x(20)$ 的对象正处于 B 概念的中心,隶属度为 100%,但是,从以上的三判定结果(图 6)可以看出,算法 1 和算法 2 有可能将对 B 概念隶属度为 100% 的对象 $x(20)$ 判定为属于 A 或 C,这明显是错误的。而算法 3 则不会出现这种错误。

(3) 算法 3 比算法 1 和算法 2 降低了算法的时间复杂度。毕竟在划分概念时,概念之间交合的部分只

占少数。因此,同时属于两个或多个概念的事物的数量较少。本算法把大多数明确属于某概念的事物直接判给该概念。这样通过简单的“明确属于某概念的判断”,省去了对大量事物的复杂的不确定性判定,降低了算法的时间复杂度。

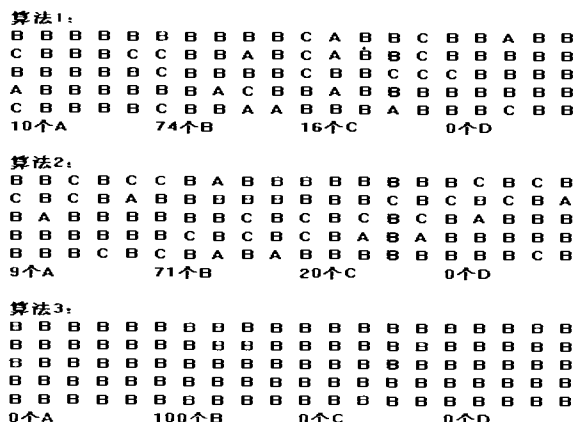


图 6 三判定算法对 $x(20)$ 的 100 次判定结果

4 结束语

分析了基于云模型的隶属概念判定的两种算法存在的不符合人类对客观事物的认识规律、错误判定和时间复杂度高等不足,并给出了改进算法,从而避免了“确定事物的不确定的判定”和“不确定事物的确定判定”的错误,增加了算法的准确性和科学性,使算法更加符合人类对客观事物的认识规律,并且降低了算法的时间复杂度。

参考文献:

- [1] 邱凯昌. 空间数据发掘与知识发现[M]. 武汉: 武汉大学出版社, 2001: 1-121.
- [2] 蒋 嵘, 李德毅, 范建华. 数值型数据的泛概念树的自动生成方法[J]. 计算机学报, 2000, 23(5): 471-476.
- [3] 李兴生, 李德毅. 一种基于云模型的决策表连续性离散化方法[J]. 模式识别与人工智能, 2003, 16(1): 33-38.
- [4] 张家精, 姜其岩, 陈金兰. 基于云理论和 Rough 集理论的数据发掘和知识发现[J]. 贵州大学学报: 自然科学版, 2004, 21(增刊 2): 14-17.
- [5] 李德毅, 杜 鹄. 不确定性人工智能[M]. 北京: 国防工业出版社, 2005.

(上接第 11 页)

参考文献:

- [1] 刘 瑞, 吴跃进, 王宗跃. Visual C++ 项目开发实用案例[M]. 北京: 科学出版社, 2006.
- [2] 求是科技. Visual C++ 6.0 程序设计与开发技术大全[M]. 北京: 人民邮电出版社, 2004.

- [3] 郎 锐. Visual C++ 数据库开发基础及实例解析[M]. 北京: 机械工业出版社, 2005.
- [4] 启明工作室. Visual C++ + SQL Server 数据库应用实例完全解析[M]. 北京: 人民邮电出版社, 2006.
- [5] 扬富国. Visual C++ 程序开发案例解析[M]. 北京: 清华大学出版社, 北京交通大学出版社, 2006.