一种提高神经网络泛化能力的新方法*)

冯乃勤[1,2] 邱玉辉[1] 王 芳[1]

(西南师范大学计算机与信息科学学院 重庆 400715)[1] (河南师范大学计算机与信息技术学院 新乡 453007)[2]

摘 要 提出了一种改进神经网络泛化能力的新方法——“缩放法”。这种方法通过对输入向量的缩放处理，来缩小或模糊化训练样本和新的模式之间的差别，从而使神经网络的泛化能力得以提高。文中提出的新算法——α算法，可以找到合适的缩放因子，进而得到泛化能力更强的新网络。一些实验例证了“缩放法”和α算法的有效性，并从理论上对其进行了分析和讨论。实验和分析表明，这种方法简单可靠，对许多神经网络和模式分类问题效果明显。

关键词 神经网络，泛化，误识率，模糊理论

A New Approach to Improve the Generalization Ability of Neural Network

FENG Nai-Qin[1,2] QIU Yu-Hui[1] WANG Fang[1]

(Faculty of Computer & Information Science, South West-China Normal University, Chongqing 400715)[1]

(College of Computer & Information Technology, Henan Normal University, Xinxiang 453007)[2]

Abstract A new approach to improve the generalization ability of neural network is presented based on an angle of fuzzy theory. This approach is put into effect through shrinking or magnifying the input vector, thereby reducing the difference between training set and testing set. It is called “Shrinking-Magnifying Approach”(SMA). At the same time in this paper, a new algorithm—α-algorithm is presented in order to find out the appropriate shrinking factor α and obtain better generalization ability of neural network. Quite a few simulation experiments serve to study about SMA andα-algorithm. The results of experiments and analyses show that the new approach is not only simpler and easier, but also is applicable to many neural networks and many classification problems.

Keywords Neural network, Generalization, Misclassification rate, Fuzzy theory

1 引言

如何提高神经网络(Neural Network，NN)的泛化或推广能力一直是该领域研究者所关注的问题。为此，人们想出了一些办法，例如，提前停止[1]、正则化[2]、反馈法[3]、输入模糊化[4]、网络集成[5]等，这些方法不同程度地提高了NN的泛化能力。但总体上说，NN的泛化仍然是一个没有解决或没有完全解决的问题。由于人工神经网络的本质特征是示例学习，而通过有限的样本训练就企图使网络武艺俱全、包打天下，无异于猴子捞月。因此，有针对性地发展一些新方法，来提高NN的泛化能力，在今后很长一段时间内都是非常需要的。

在已有的改进NN泛化能力的方法中，H Ishibuchi 和 M Nii 的研究很具特色，值得重视。他们通过将输入向量模糊化来避免网络过度训练，从而改进了NN的推广能力[4]。武妍和王守觉提出了一种通过反馈提高前向网络学习性能的新算法，称为FBBP算法[3]，也可以提高NN的泛化能力。他们通过误差反传和结果反馈，在对权值调整的同时也对输入向量进行微量调整，其主导思想是通过将输入向量模糊化，从而扩大NN的适应范围，提高其泛化能力。这种思想自然质朴，切实可行，也给我们带来了新的启发。长期以来，人们一直致力于调整NN的连接权值来提高其性能，缺少新的思路。而通过输入数据模糊化方法来提高NN的泛化能力，堪称是从一个全新的角度考虑问题。就像黑夜中一盏明灯，给人以希望和方向。但我们认为，输入数据模糊化的途径并不唯一，或许另有新路。通过一些新的探索，找到了一条能有效提高神经网络泛化能力的新途径：将输入数据适当缩小或放大，借以将输入数据模糊化，从而提高NN的泛化能力。我们称这种方法为“缩放法”。α算法描述了该方法的基本过程。通过α算法可以找到合适的缩放因子，得到泛化能力更强的新网络。

2 “缩放法”和α算法

2.1 “缩放法”

“缩放法”源于生活中的简单现象。

荧屏效应：打开电脑，调整屏幕，画面可以缩小，也可以扩大。随着缩放调整，图像有时整体清晰局部模糊(缩小时)，有时又局部清晰而整体模糊(扩大时)；缩小时对象可能变成一个模糊单点，扩大时又可能溢出框外而使对象失去原貌。总之，在这两种情况下，对象分别被从局部上和整体上模糊化了。

在实践中，人们常采用类荧屏效应对事物进行放大或缩小。放大法是将事物的细节放大到便于观察，以便更精确地研究事物的局部，然而在局部精确化的同时，事物的整体性却相对淡化或模糊化了，因此属于整体性模糊化方法；缩小法是

*)基金项目：河南省自然科学资助项目(0511012500)；教育部科技司资助项目“基于禁忌搜索的模糊神经控制研究”。冯乃勤 博士研究生，副教授，主要研究方向：人工智能，模糊理论，神经网络。邱玉辉 教授，博士生导师，主要研究方向：人工智能，计算智能，神经模糊控制等。王 芳 博士研究生，讲师，主要研究方向：神经网络，优化算法。

通过缩小事物,略去细节,来概括事物,宏观看待事物的整体,然而在突出整体的同时,事物的局部却被淡化或模糊化了,因此属于局部模糊化方法。所以,放大和缩小这样看似矛盾和对立的两种方法,实际上也有着内在的统一性,可视为两种不同的模糊化方法。当两种事物从整体上或局部上被模糊化以后,它们在整体上或局部上的差别势必跟着减小,使得原来差别较大的两个事物更加接近,甚至可以归为同类。

基于上述模糊化观念,产生了"缩放法"(Shrinking-Magnifying Approach,SMA)。它是指为了缩小或模糊化训练样本和测试样本间的差别,提高神经网络的泛化能力,而预先对输入向量进行适当缩小或放大的数据预处理方法。合适的缩放因子(Shrinking Factor)α 由 α 算法(α-algorithm)确定。

2.2 α算法

Step1. 划定训练集 P_1,测试集 P_2,相应目标集 t_1,t_2。设定 α 的搜索范围$[a,b]$和搜索步长 δ; $j=0$;

Step2. for $r=a:\delta:b$

Step3. $j=j+1;\alpha[j]=r$;

Step4. $p_1=\alpha[j]\cdot P_1$; $p_2=\alpha[j]\cdot P_2$;

Step5. 用 p_1,t_1 训练或生成新网络 net;

Step6. 用 p_2 对 net 仿真测试,结果为 y_2;

Step7. 比较 y_2 和 t_2,计算误差 $E(j)=t_2-y_2$;

Step8. endfor;

Step9. 找出 $\alpha[j]=1$ 时的误差 $E_{\alpha=1}$;

Step10. if $\min(E[j])<E_{\alpha=1}$ then $\alpha_0=\alpha[j]$;

else $\alpha_0=1$; endif;

Step11. $p_0=\alpha_0\cdot P_1$;

Step12. 用 p_0 训练或生成目标网络 NET;

Step13. return。

3 实验

实验 1:概率神经网络对 Wine 分类实验

我们首选概率神经网络(Probabilistic Neural Network, PNN)对 Wine 分类问题作实验。之所以首先选择 PNN 是为了简单和便于入手,因为这种网络的规模依赖于输入向量的规模,不需要人为选定网络神经元的数目[6],同时概率神经网络也适合于模式分类问题。设定 α 算法的搜索范围为[0.0001,2],步长 δ=0.0001。这里使用的数据集来自于 UCI Repository of Machine Learning Databases 中的 Wine Recognition Database。该数据集由 Stefan Aeberhard 提供,他使用化学分析的方法确定甜酒之源(the origin of wine)。Wine 数据集有 178 个例子,每例包含 13 个酒的属性,178 例共分为 3 类,第 1 类有 59 例,第 2 类有 71 例,第 3 类有 48 例。实验中将 Wine 数据集分为两个子集,第一子集的三类例子数分别为 30、36 和 24,第二子集的三类例子数分别为 29、35 和 24,两子集分别用作生成网络和仿真测试,主要结果如表 1 所示。

从表 1 可以得出如下结果:

(1)"缩小法"能减小 PNN 对 Wine 分类的错识率(Misclassification Rate),即 SMA 具有减小或纠正 PNN 错误分类的能力,合适的缩放因子 α 大约在 0.05~0.1 左右;

(2)"放大法"未能减小 PNN 对 Wine 分类的错识率,随着扩大倍数的提高,错识率反而增加;

(3)设 $E_1,E_{\leqslant 1}$ 和 $E_{\geqslant 1}$ 分别表示神经网络在缩放因子 $\alpha=1,\alpha\leqslant 1$ 和 $\alpha\geqslant 1$ 时的错识率;Ra 和 Rr 分别表示 SMA 的绝对纠错率和相对纠错率,则定义,

$$Ra=(E_1-E_{\leqslant 1})\times 100\%\text{,当 }\alpha\leqslant 1 \quad (1)$$

$$Ra=(E_1-E_{\geqslant 1})\times 100\%\text{,当 }\alpha\geqslant 1 \quad (2)$$

$$Rr=(E_1-E_{\leqslant 1})/E_1\times 100\%\text{,当 }\alpha\leqslant 1 \quad (3)$$

$$Rr=(E_1-E_{\geqslant 1})/E_1\times 100\%\text{,当 }\alpha\geqslant 1 \quad (4)$$

表 1 PNN 对 Wine 分类的实验结果

Shrinking Factor	Misclassification Rate	Shrinking Factor	Misclassification Rate
0.0001	53/88	**0.07**	**25/88**
0.0003	42/88	**0.09**	**25/88**
0.0005	30/88	**0.10**	**25/88**
0.0007	30/88	0.30	26/88
0.0009	29/88	0.50	29/88
0.001	28/88	0.70	34/88
0.003	25/88	0.90	42/88
0.005	27/88	**1.00**	**44/88**
0.007	25/88	1.10	46/88
0.009	28/88	1.30	53/88
0.01	28/88	1.50	54/88
0.03	26/88	1.70	56/88
0.05	**25/88**	2.00	57/88

由表 1 及式(1)和式(3)计算得到此种情况下,SMA 的最大绝对纠错率和相对纠错率分别是,

$\max(Ra)=(E_1-E_{0.05})\times 100\%=19/88\times 100\%=22\%$

$\max(Rr)=(E_1-E_{0.05})/E_1\times 100\%=19/44\times 100\%=43\%$

可见,将"缩放法"应用于 PNN 对 Wine 分类时,其绝对纠错率和相对纠错率都是较大的,神经网络泛化能力得到如此明显的提高,令人鼓舞。

实验 1 使我们看到了"缩放法"这样一种唾手可得的方法对提高神经网络的泛化能力起到了作用。问题在于这种作用是个别的还是一般的?是偶然的还是必然的?这种作用到底有多大?要回答这些问题,必须有足够的实验。

表 2 BPNN 对 Iris 分类的实验结果

Shrinking Factor	Misclassification Rate	Shrinking Factor	Misclassification Rate
0.24	1.70/75	0.68	3.25/75
0.28	**1.30/75**	0.70	4.90/75
0.30	3.45/75	0.74	5.45/75
0.34	1.60/75	0.78	9.30/75
0.38	4.50/75	0.80	3.45/75
0.40	3.30/75	0.84	1.55/75
0.44	1.90/75	0.88	2.75/75
0.48	2.85/75	0.90	1.35/75
0.50	2.00/75	0.94	5.30/75
0.54	8.20/75	0.98	3.00/75
0.58	3.30/75	**1.00**	**10.85/75**
0.60	2.10/75	1.04	10.45/75
0.64	5.30/75	**1.08**	**3.15/75**

实验 2:BP 神经网络对 Iris 分类实验

BP 神经网络(Back Propagation Neural Network,BPNN)是目前应用最为广泛的神经网络,"缩放法"对它的泛化能力影响如何,是应该进行研究的。实验采用 4-6-3 网络结构,传递函数分别为"tansig"和"purelin",对网络的学习与训练采用尺度化共轭梯度反向传播算法,初始网络由 MATLAB 神经网络工具箱函数 newff 建立。Iris 数据库是人们广泛使用的用于模式分类的实例系统,Fisher 出版的 Iris 数据库共有 150 组数据,每组数据包含 Iris 属植物的四个属性:萼片长(Sepal Length),萼片宽(Sepal Width),花瓣长(Petal Length),花瓣宽(Petal Width),用 4 维向量表示。150 组数据分属于 3 类不同的 Iris 属植物:Iris setosa,Iris versicolor 和 Iris virginica,每类各有 50 组数据。实验中,将 Iris 数据集分为两部分 P_1 和 P_2,每部分各包含 75 个样例,其中 P_1 用作生成网络,P_2

用作仿真测试。表2给出了主要实验结果。

BPNN的实验结果很有趣,值得研究:

(1)BP网络的实验结果与网络初始化有关。当用initzero函数将网络权值和域值全部初始化为0时,实验表明,"缩放法"对BP网络的泛化不起作用:对于Iris分类问题,不论缩放因子取何值,错识率始终为18/75(此时用newcf函数建立网络)。而当用Nguyen-Widrow方法(采用该方法初始化后,每个神经元的激活区域将均匀地分布在输入空间中,从而可避免神经元的浪费)初始化网络时,"缩放法"有效。具体地,对Iris分类的绝对纠错率和相对纠错率分别为:

$$\max(Ra)=(E_1-E_{0.28})\times100\%=9.55/75\times100\%=13\%$$

$$\max(Rr)=(E_1-E_{0.28})/E_1\times100\%=9.55/10.85\times100\%=88\%$$

(2)由于初始化方法的随机性,因此BP网络的工作结果也是相应变化的,表2是20次运行结果的平均值。

(3)对于BP网络,"扩大法"有效,它能够提高神经网络的泛化能力。具体地,对于Iris分类问题,"扩大法"的绝对纠错率和相对纠错率分别为:

$$\max(Ra)=(E_1-E_{1.08})\times100\%=7.7/75\times100\%=10\%$$

$$\max(Rr)=(E_1-E_{1.08})/E_1\times100\%=7.7/10.85\times100\%=71\%$$

可见,"扩大法"同样能提高BP网络的泛化能力。当然,相比之下,在低端缩小比在高端扩大更为有效。

另外,我们还做了径向基函数网络(RBFN)对Iris分类的实验和学习矢量量化(LVQ)网络对Wine分类的实验,均取得了预期的效果。其中,RBFN对Iris分类的实验结果如下:

(1)合适的缩放因子α大约在0.15~0.25左右;

(2)在实验范围内,"放大法"无效;

(3)实验所得绝对纠错率和相对纠错率分别是,

$$\max(Ra)=(E_1-E_{0.05})\times100\%=39/75\times100\%=52\%$$

$$\max(Rr)=(E_1-E_{0.05})/E_1\times100\%=39/43\times100\%=91\%$$

可见,此时"缩小法"改进RBFN泛化能力的作用是极其明显的。91%,这个数字令人惊奇。

LVQ网络对Wine分类的结果较差,实验所得:

$$\max(Ra)=(E_1-E_{2.0})\times100\%=2/88\times100\%=2.3\%$$

$$\max(Rr)=(E_1-E_{2.0})/E_1\times100\%=2/27\times100\%=7.4\%$$

SMA提高LVQ网络泛化能力的作用虽然很小,但不能说无效,在某些情况下,哪怕提高一个百分点都可能是至关重要的。

4 讨论和分析

为什么简单的"缩放法"及其α算法能够改进和提高一些神经网络的泛化能力,且有时还非常明显?分析实验结果,探究其原因,认为有以下几方面:

(1)神经网络属于软计算或非经典计算风范,有着本质上的不确定性和模糊性,而这种不确定性和模糊性就成为"缩放法"应用的基础。

(2)缩小法像一面缩小镜,它缩小或模糊化了事物之间的差别。具体来说,它缩小和模糊化了NN训练样本和测试样本之间的差别,模糊了数据的细节,从而提高了与样本相类似的数据的适应性,即提高了网络的泛化能力。例如,Iris数据集中有这样一例数据,$p_1=[x_1,x_2,x_3,x_4]=[60,22,50,15]$,应分为第3类。但是当用生成好的概率网络对其仿真测试时,它却被错分为第2类,属于被错误分类的3例中的一例。然而当用缩小法,取缩小因子$\alpha=0.04\sim0.08$,重新生成概率网络并对p_1进行测试时,这一错误消失了,纠正了1/3的错误。究其原因,可做这样的分析:任取生成PNN时所用样本集中与p_1比较接近的一个同属第3类的样本数据,例如,取$p=[x_1,x_2,x_3,x_4]=[63,28,51,15]$,站在模糊理论[7,8]的立场上,不妨把$p$和$p_1$看作是另一类模糊集,只是其隶属函数的值域为[0,100],而不是通常的[0,1]。若用海明距离公式

$$d(p,p_1)=\frac{1}{n}\times\sum_{i=1}^{n}|\mu_{p(xi)}-\mu_{p1(xi)}| \tag{5}$$

计算它们之间的语义距离,则在正常情况下的语义距离和采用缩小法($\alpha=0.05$)时的语义距离分别为:

$$d(p,p_1)=\frac{1}{4}\times\sum_{i=1}^{4}|\mu_{p(xi)}-\mu_{p1(xi)}|=\frac{1}{4}(3+6+1+0)=2.5$$

$$d(\alpha.p,\alpha.p_1)=\frac{\alpha}{4}\times\sum_{i=1}^{4}|\mu_{p(xi)}-\mu_{p1(xi)}|=0.125$$

显然,缩放法的模糊化作用使样本数据和测试数据间的语义距离小了许多倍,从而能将p_1正确归类。

(3)正则化方法可以提高神经网络的泛化能力,在此,也可用以解释SMA的作用:

在训练样本集大小一定的情况下,网络的推广能力与网络的规模直接相关。如果神经网络的规模远远小于训练样本集的大小,则发生过度训练的机会就很小。而当网络规模较大,需调整的权值参数较多时,对同样的训练样本集,要达到较高的性能指标,调整起来就比较困难,训练的次数可能会大大增加,从而造成过度训练的可能性就很大。但是对于特定的问题,确定合适的网络规模往往是一件十分困难的事情。正则化方法[2]独辟蹊径,通过修正神经网络的训练性能函数来提高其泛化能力。一般情况下,神经网络的训练性能函数采用均方误差函数,即

$$mse=\frac{1}{N}\sum_{i=1}^{N}(e_i)^2=\frac{1}{N}\sum_{i=1}^{N}(t_i-a_i)^2 \tag{6}$$

正则化方法将网络的性能函数改进为如下形式:

$$msereg=\gamma.mse+(1-\gamma)msw \tag{7}$$

其中,γ为比例系数,msw为所有网络权值平方和的平均值,即

$$msw=\frac{1}{n}\sum_{j=1}^{n}W_j^2 \tag{8}$$

通过采用新的指标函数,可以在保证网络性能函数尽可能小的情况下使网络具有较小的权值,即使得网络的有效权值尽可能小,这实际上相当于自动缩小了网络规模。

"缩放法"有利于减小网络的权值,这一点在概率网络表现最为明显(对RBFN亦然)。PNN是可用来把模式分类的三层网络,其标准形式是不进行训练的,仅将训练向量变为第一层的权值向量[6]。因此,当SMA法将训练向量缩小时,网络的权值便跟着缩小,相当于自动缩小了网络的规模。由式(7)可见,有利于性能函数的减小,即有利于误差的减小和推广能力的提高。当然,事物总有限度,也不可能无限缩小,而是有一个合适的缩放因子。

(4)一般来说,"扩大"和"缩小"是矛盾和对立的两种方法,但在一定条件下,二者又是统一的。对于PNN和RBFN,"缩小法"能有效提高它们的泛化能力,而"扩大法"无效;但在LVQ网络和BPNN的特定条件下,两种方法同时有效,即

"缩放法"在高端和低端同时有效。作者的理解是,"扩大法"在此也是一种模糊化方法,它扩大和精确化事物的细节,却模糊化了事物的整体,即缩小和模糊化了训练样本和新的模式之间的整体差别,同样促进了网络泛化能力的提高。

"缩小法"适用于 PNN 和 RBFN,"扩大法" 适用于 LVQ 网络和 BPNN。不同的方法适用于不同的对象,不同的对象需要用不同的思想方法去认识它们。"荧屏效应"和下面给出的"模糊模缩"算子,可进一步揭示"缩放法"和"扩大法"的作用机制。

模糊模缩算子 对于向量 p、模 m 和缩放因子 α,定义 p 的 α 模糊模缩运算 $F \bmod S$ 为:

$$F \bmod S(\alpha p)=\begin{cases} f(\alpha p), & \alpha\leqslant 1 \\ g((\alpha-m)p), & \alpha>1 \end{cases} \tag{9}$$

其中,$m\in I$(整数集),f 和 g 是实函数,二者可以相同,例如,有时可取 $f(\alpha p)=\alpha p, g((\alpha-m)p)=(\alpha-m)p$,等等,但一般来说,二者是不同的。

可以认为,在 BPNN 和 LVQ 网络的内部,存在着函数 f 和 g,它们都是隐函数而不是显函数。在对 BPNN 和 LVQ 网络运用 SMA 时,网络内部固有的模糊模缩运算使得在 α 取值的某些区间和某些点上,训练样本和测试样本之间的语义距离减小了,从而减小了误识率,提高了网络的泛化能力。

结束语 本文从模糊理论角度提出了一种改进 NN 泛化能力的新方法和新算法,它们是通过缩小或放大样本数据,从而缩小或模糊化训练样本和测试样本之间的差别来实现的。大量实验例证了其有效性,并对其原理进行了初步的分析和讨论。实验和分析表明,SMA 方法和 α 算法简单可靠,效果明显,具有一定的理论基础,可用以改进和提高许多神经网络的泛化能力。当然,NN 是个大家族,尽管我们做了大量实验,且有很多实验结果在文中无法一一给出,但毕竟不能穷尽。因此,SMA 方法及 α 算法应用价值和理论价值的进一步提升还依赖于更多的研究和实践。

参考文献

1 Sarle W S. Stopped training and other remedies for overfitting. In: Proc. of the 27th Symposium on the Interface, 1995

2 Hinton G E. Connectionist learning procedures. Artificial Intelligence, 1989, 40: 185~234

3 武妍,王守觉. 一种通过反馈提高神经网络学习性能的新算法. 计算机研究与发展, 2004, 41(9): 488~492

4 Ishibuchi H, Nii M. Fuzzification of input vector for improving the generalization ability of neural networks. In: The Int'l Joint Conf. on Neural Networks, Anchorage, Alaska, 1998

5 Hansen L K, Salamon P. Neural Network Ensembles. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1990, 12(10): 993~1001

6 Specht D F. Probabilistic neural networks. Neural Networks, 1990, 3(1): 109~118

7 Jang J S R, Sun C T, Mizutani E. Neuro-Fuzzy and Soft Computing. Upper Saddle River, NJ: Prentice-Hall Inc, Simon & Schuster/A Viacom Company, 1997

8 冯乃勤. 模糊概念的模糊度研究. 模式识别与人工智能, 2002, 15(3): 290~294

(上接第 192 页)

$\max(1-f_{\overline{A}_R}(x), 1-f_{\overline{B}_R}(x))$

$=\max\{\max\limits_{y\in U}\{\min(R(x,y),1-f_A(y))\},\max\limits_{y\in U}\{\min(R(x,y),1-f_B(y))\}\}$

$=\max\limits_{y\in U}\{\max(\min(R(x,y),1-f_A(y)),\min(R(x,y),1-f_B(y)))\}$

$=\max\limits_{y\in U}\{\min(1-R(x,y),\max(1-f_A(y),1-f_B(y)))\}$

$=1-f_{\overline{A\cup B}_R}(x)$

即 $\overline{A\cup B}_R=\overline{A}_R\cup\overline{B}_R$。

3) $t_{\underline{\Delta}_R\cup\underline{B}_R}(x)=\max(t_{\underline{\Delta}_R}(x),t_{\underline{B}_R}(x))$

$=\max\{\min\limits_{y\in U}\{\max(1-R(x,y),t_A(y))\},\min\limits_{y\in U}\{\max(1-R(x,y),t_B(y))\}\}$

$\leqslant\min\limits_{y\in U}\{\max(\max(1-R(x,y),t_A(y)),\max(1-R(x,y),t_B(y)))\}$

$=\min\limits_{y\in U}\{\max(1-R(x,y),\max(t_A(y),t_B(y))\}=t_{\underline{A\cup B}_R}(x)$

$1-f_{\underline{\Delta}_R\cup\underline{B}_R}(x)=\max(1-f_{\underline{\Delta}_R}(x),1-f_{\underline{B}_R}(x))$

$=\max(\min\limits_{y\in U}\{\max(1-R(x,y),1-f_A(y))\},\min\limits_{y\in U}\{\max(1-R(x,y),1-f_B(y))\})$

$\leqslant\min\limits_{y\in U}\{\max(\max(1-R(x,y),1-f_A(y)),\max(1-R(x,y),1-f_B(y)))\}$

$=\min\limits_{y\in U}\{\max(1-R(x,y),\max(1-f_A(y),1-f_B(y)))\}$

$=\min\limits_{y\in U}\{\max(1-R(x,y),1-f_{A\cup B}(y))\}=1-f_{\underline{A\cup B}_R}(x)$

即 $\underline{A}_R\cup\underline{B}_R\subseteq\underline{A\cup B}_R$。

$t_{\overline{A\cap B}_R}(x)=\max\limits_{y\in U}\{\min(R(x,y),t_{A\cap B}(y))\}$

$=\max\limits_{y\in U}\{\min(R(x,y),\min(t_A(x)),t_B(x))\}$

$=\max\limits_{y\in U}\{\min(\min(R(x,y),t_A(x))),\min(R(x,y),t_B(x)))\}$

$\leqslant\min(\max\limits_{y\in U}\{\min(R(x,y),t_A(x))\},\max\limits_{y\in U}\{\min(R(x,y),t_B(x))\})$

$=\min(t_{\overline{A}_R}(x),t_{\overline{B}_R}(x))=t_{\overline{A}_R\cap\overline{B}_R}(x)$

$1-f_{\overline{A\cap B}_R}(x)=\max\limits_{y\in U}(\min(R(x,y),1-f_{A\cap B}(y))\}$

$=\max\limits_{y\in U}\{\min(R(x,y),\min(1-f_A(y)),1-f_B(y)))\}$

$=\max\limits_{y\in U}\{\min(\min(R(x,y),1-f_A(y)),\min(R(x,y),1-f_B(y)))\}$

$\leqslant\min(\max\limits_{y\in U}\{\min(R(x,y),1-f_A(y))\},\max\limits_{y\in U}\{\min(R(x,y),1-f_B(y))\})$

$\leqslant\min(1-f_{A_R}(x),1-f_{B_R}(x))=1-f_{\overline{A}_R\cap\overline{B}_R}(x)$

即 $\overline{A\cap B}_R\subseteq\overline{A}_R\cap\overline{B}_R$。

结论 本文研究了一般关系下 Vague 集合的近似问题,建立了一般关系下粗糙 Vague 近似的框架。首先分析了经典的粗集理论、模糊集理论、Vague 集理论三者之间的关系,提出了一般关系下粗糙 Vague 集的概念,并用构造性方法定义了粗糙 Vague 近似算子,讨论了粗糙 Vague 的性质。粗糙 Vague 集的研究对于处理一定形式的数据或知识信息具有工具性使用价值,进一步开展粗糙集与 Vague 集融合的研究具有现实的意义。

参考文献

1 王珏,苗夺谦,周育健. 关于 Rough Set 理论与应用的综述. 模式识别与人工智能, 1998, 11: 34~40

2 Yao Y Y. A Comparative study of fuzzy sets and rough sets. Journal of Information Sciences, 1998, 109: 227~242

3 米据生,张文修,徐宗本. 粗糙模糊集的构造与公理化方法. 计算机学报, 2004, 27(3): 197~202

4 闫德勤,迟忠先. 粗糙集与 Vague 集. 计算机科学, 2004, 31(8): 133~135

5 江莉,等. Vague 决策表的知识获取. 计算机科学, 2004, 31(8): 111~112

6 李凡,等. 基于 Vague 集的加权多目标模糊决策方法. 计算机科学, 2001, 28(7): 60~62

7 符海东,卢正鼎. 基于 Vague 集的模糊决策方法. 小型微型计算机系统, 2004, 25(9): 1684~1686