基于属性重要性的加权聚类融合

阳琳赟 周海京 卓 晴 王文渊
(清华大学自动化系 北京 100084)

摘 要 聚类融合是数据挖掘研究的一个热点。当前相关研究大多没有考虑进行融合的聚类成员的质量,因此较差的成员和噪声会对融合结果产生不良的影响。提出了一种对聚类成员进行加权的融合方法。该方法引入粗糙集理论中的属性重要性度量,根据聚类成员对融合的重要性赋予其权重,生成加权共生矩阵,进而产生融合结果。实验结果表明,提出的方法能较好地处理聚类成员间的质量差异,并能有效地消减噪声对融合的影响,从而得到更好的聚类融合结果。

关键词 聚类融合,共生矩阵,属性重要性度量
中图法分类号 TP18 **文献标识码** A

Weighted Cluster Ensemble Based on Significance of Attribute

YANG Lin-yun ZHOU Hai-jing ZHUO Qing WANG Wen-yuan
(Department of Automation, Tsinghua University, Beijing 100084, China)

Abstract Cluster ensemble is a hot topic in data mining research. Resent research mostly pays little attention to the qualities of cluster members. However, bad cluster members and noise may affect the ensemble result. A weighted cluster ensemble approach was proposed. This approach set weights to all cluster members according to the significance of them relative to the ensemble result. The significance of each cluster member was evaluated through information measures of significance of attribute in rough set theory. Then weighted co-association matrix was generated and the final ensemble result was obtained. The experimental results show that the proposed approach can handle well different-quality of cluster members and lessen the affect of noise effectively. Therefore, it can afford better ensemble result compared with general cluster ensemble methods.

Keywords Cluster ensemble, Co-association matrix, Measure of significance of attribute

1 引言

聚类算法是一种非监督的机器学习算法,目的是将数据集人为地划分成若干类,以揭示这些数据分布的真实情况。然而,没有任何一种算法能胜任任意形状、任意分布的数据的聚类[1]。因此,面对特定的应用问题,如何选择合适的聚类算法,是聚类分析研究中的一个重要课题。

融合方法将不同算法或者同一算法下使用不同参数得到的结果进行合并,从而得到比单一算法更为优越的结果。在分类算法和回归模型中,融合方法的使用已经比较成熟。但在聚类分析领域,聚类融合方法的研究在近几年才开始出现[2]。A. L. Fred 在文献[3]中用类似投票的方法融合聚类结果,而 A. Strehl 等在文献[4]中正式提出聚类融合的概念:将多个对一组数据进行聚类的不同结果进行融合,而不使用对象原有的特征;并提出了 CSPA, HGPA, MCLA 3 种融合算法。之后大量的研究都是基于共生矩阵的。共生矩阵体现了数据点与点之间的关联程度,它统计了两数据点在不同聚类结果中属于同一个类的频率。在此基础上,CSPA 算法[4]使用无向图割算法 METIS[5]产生融合结果,EA 算法[6]使用单连接层次聚类算法产生融合结果,WsnnG 算法[7]利用共生矩阵生成加权共享最近邻图,再使用 METIS 算法产生融合结果。

当前的聚类融合算法大多不考虑进行融合的聚类成员的聚类质量。因此,当部分聚类成员的聚类质量较差或者聚类成员有噪声干扰时,融合结果将受到影响。针对这种情况,本文提出了一种基于属性重要性的加权聚类融合(SoA-WCE: Significance of Attribute based Weighted Cluster Ensemble)方法,首先对聚类成员进行初次融合,融合算法可以选用任意基于共生矩阵的算法,然后根据融合结果由粗糙集理论中的属性重要性度量给聚类成员赋予权重,生成新的加权共生矩阵,再用与初次融合相同的融合算法产生最终的融合结果。由于对聚类成员进行加权能更好地突出对融合贡献大的成员,并能有效消减噪声对融合的影响,因此该方法能得到更好的融合结果。

本文第 2 节介绍粗糙集理论中的属性重要性度量,第 3 节介绍本文提出的加权聚类融合方法,第 4 节用人工和真实

到稿日期:2008-05-13
阳琳赟(1980—),男,博士研究生,主要研究方向为数据挖掘、人工智能和模式识别等,E-mail: yly01@mails.tsinghua.edu.cn;**周海京** 男,博士研究生;**卓 晴** 男,副教授;**王文渊** 男,教授,博士生导师。

数据集对本文提出的方法进行测试，并与相应的单次融合算法进行比较，最后对本文的工作进行总结并展望进一步的工作。

2 粗糙集理论中的属性重要性度量

2.1 粗糙集理论的基本概念

信息系统 S 可表示为一个四元组 $\langle U,A,V,f\rangle$，其中 $U=\{x_1,x_2,\cdots,x_n\}$ 是对象的集合，称为论域，$n=|U|$；$A=\{a_1,a_2,\cdots,a_m\}$ 是对象所有属性的集合，$m=|A|$，通常有 $A=C\cup D$，C 为条件属性集，D 为决策属性集；V 为属性的值域集，$V=\bigcup V_a$，$a\in A$，V_a 为 a 属性的值域；f 为信息函数，为对象的每个属性指明一个值或值集，有 $f:U\times A\to V$，$f(x_i,a)\in V_a$。

令 R 是定义在论域 U 上的等价关系，$IND(R)=U/R$ 表示 U 在 R 关系下的等价分类，$[x]_R$ 表示 x 在 R 下的等价类。

2.2 决策表属性的信息熵表示

设条件属性集 P,Q 在 U 上导出的划分为 $X=\{X_1,X_2,\cdots,X_n\}$ 和 $Y=\{Y_1,Y_2,\cdots,Y_m\}$，则 P,Q 在 U 的子集组成的 σ 代数上的概率分布为

$$(X:p)=\begin{bmatrix}X_1 & X_2 & \cdots & X_n\\ p(X_1) & p(X_2) & \cdots & p(X_n)\end{bmatrix}$$

$$(Y:p)=\begin{bmatrix}Y_1 & Y_2 & \cdots & Y_m\\ p(Y_1) & p(Y_2) & \cdots & p(Y_m)\end{bmatrix}$$

其中 $p(X_i)=|X_i|/|U|$，$i=1,2,\cdots,n$，$p(Y_j)=|Y_j|/|U|$，$j=1,2,\cdots,m$。

属性集 P 的熵 $H(P)$ 定义为

$$H(P)=-\sum_{i=1}^{n}p(X_i)\log(p(X_i)) \tag{1}$$

属性集 Q 相对属性集 P 的条件熵定义为

$$H(Q|P)=-\sum_{i=1}^{n}p(X_i)\sum_{j=1}^{m}p(Y_j|X_i)\log(p(Y_j|X_i)) \tag{2}$$

其中 $p(Y_j|X_i)=|Y_j\cap X_i|/|X_i|$，$i=1,2,\cdots,n$，$j=1,2,\cdots,m$。

2.3 属性重要性的信息熵表示

在信息表系统 $\langle U,A,V,f\rangle$ 中，$A=C\cup D$，设 $P\subset C$，则对于任意属性 $c\in C\backslash P$ 的重要性定义为 $SGF(c,P,D)=H(D|P)-H(D|P\cup\{c\})$。若 $P=\varnothing$，则 $SGF(c,\varnothing,D)=H(D)-H(D|\{c\})$，称为属性 c 和决策 D 的互信息，记为 $I(c,D)$。

$SGF(c,P,D)$ 的值越大，说明在已知 P 的条件下，属性 c 对于决策 D 就越重要；若 $P=\varnothing$，则说明在未知任何条件的情况下，$I(c,D)$ 越大，属性 c 对于决策 D 就越重要[8]。

3 基于属性重要性的加权聚类融合

3.1 加权聚类融合

当前的聚类融合算法大多不考虑进行融合的聚类成员的聚类质量，而对所有的聚类成员同等看待。基于共生矩阵的融合算法首先根据聚类成员计算出共生矩阵 Co，其中 $Co(i,j)=N(a(x_i)=a(x_j))/H$，$H$ 为聚类成员个数，$N(a(x_i)=a(x_j))$ 为数据点 x_i 和 x_j 被划分在同一类的次数。然后在共生矩阵的基础上得到融合结果。

若考虑给聚类成员赋予权重 $\omega=\{\omega_1,\omega_2,\cdots,\omega_H\}$，则可以生成加权共生矩阵 Co_w，其中 $Co\_w(i,j)=\sum_{k=1}^{H}\omega_k I_k(a(x_i)=a(x_j))$，$\omega_k$ 为第 k 个聚类成员的权值，若该次聚类中，数据点 x_i 和 x_j 被划分在同一类，则 $I_k(a(x_i)=a(x_j))=1$，否则 $I_k(a(x_i)=a(x_j))=0$。当 $\omega_1=\omega_2=\cdots=\omega_H=1/H$ 时，加权共生矩阵还原为普通的共生矩阵。在加权共生矩阵的基础上，便可得到加权聚类融合结果。

3.2 权重的计算

我们可以用粗糙集理论中的决策表属性重要性的信息熵来衡量聚类成员的重要性，从而设置聚类成员的权重。为了衡量聚类成员对融合的贡献，首先对聚类成员进行初次融合，融合算法可以选用任意的基于共生矩阵的算法，如 CSPA 算法、EA 算法、WsnnG 算法等；然后，将聚类成员和初次融合结果构建成为一个决策表系统 $\langle U,A,V,f\rangle$，$U=\{x_1,x_2,\cdots,x_n\}$ 为数据集，$A=C\cup D$，$C=\{c_1,c_2,\cdots,c_H\}$ 为聚类成员，D 为初次融合结果，V 为属性的值域集，$V=V_C\cup V_D$，$V_C=\bigcup V_c$，$c\in C$，V_{c_k} 为 c_k 的值域，c_k 表示第 k 个聚类成员的聚类结果，V_D 为决策属性 D 的值域。f 为信息函数，有 $f:U\times A\to V$，$f(x_i,c)\in V_c$。

根据粗糙集理论，属性 c 对于决策 D 的互信息值 $I(c,D)$ 表征了属性 c 对决策 D 的重要程度。在由聚类成员和初次融合结果构造的决策表系统中，聚类成员 c_k 对于融合结果 D 的互信息值 $I(c_k,D)$ 表征了聚类成员 c_k 对于融合结果 D 的重要程度。因此，我们可以对重要程度大的成员，即与融合结果互信息值大的成员赋予较大的权重，并据此生成新的加权共生矩阵。在该加权共生矩阵的基础上，再使用与初次融合相同的融合算法产生最终的融合结果。

由于不同的融合算法有不同的特性，如 CSPA 算法产生各类数目均衡的结果；EA 算法对流形分布的数据有较好的效果；WsnnG 算法对于各类数目不平衡时有较好的效果。因此，初次融合和加权融合使用的融合算法需要保持一致。

3.3 SoA-WCE 算法流程

SoA-WCE 算法流程如下所示：

第 1 步　选用某种基于共生矩阵的融合算法，生成初次融合结果；

第 2 步　利用初次融合结果和聚类成员构建决策表系统；

第 3 步　对每个聚类成员 c_k，$k=1,2,\cdots,H$，计算它在该决策表系统中的属性重要性信息熵值 $E_k=I(c_k,D)$；

第 4 步　对每个聚类成员 c_k 设置权值：$\omega_k=E_k/\sum_{k=1}^{H}E_k$；

第 5 步　根据计算的权值生成加权共生矩阵；

第 6 步　在加权共生矩阵的基础上使用第 1 步中的融合算法，得到最终融合结果。

4 实验结果与分析

为了验证算法的有效性，我们选取两个人工数据集和 3 个真实数据集进行实验，其中“halfrings”为两个半环状分布数据集，“2D2K”为二维高斯分布数据集，如图 1 所示。

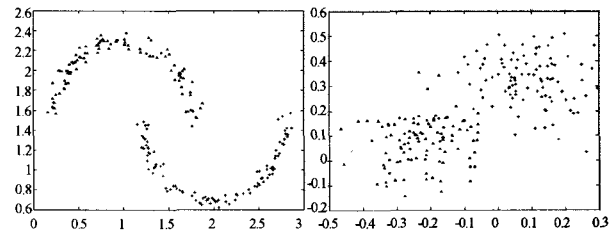

图 1　halfrings 和 2D2K 数据集

"Iris","WDBC","Wine"可从 UCI 机器学习数据库[9]中获得,所有的数据集均有参考类标识,并假设数据集的类数是已知的。数据集的详细信息如表 1 所列。

表 1　实验数据集信息

名称	类型	类数	维数	点数	各类分布
halfrings	人工	2	2	400	200/200
2D2K	人工	2	2	400	200/200
Iris	真实	3	4	150	50/50/50
WDBC	真实	2	30	569	357/212
Wine	真实	3	13	178	59/71/48

当数据有分类信息时,可认为该分类信息在一定程度上表达了数据的一些内部分布特性。如果该分类信息没有被聚类过程所利用,则可以用它来评价聚类效果。我们使用常用的 F-measure 准则[10]对结果进行评价。F-measure 值在 0 到 1 之间,越接近 1 表明融合结果质量越好。

我们使用两种策略来产生聚类成员:一是使用 k-means 算法选择不同的参数 k 和不同的初始中心点生成聚类成员;二是分别使用 k-means 算法、single-link 算法、complete-link 算法、ave-rage-link 算法、ward-link 算法等比例产生聚类成员,其中 k-means 算法选择不同的参数 k 和不同的初始中心点,其他 4 种算法选择不同的参数 k。对于两种策略产生的聚类成员,我们还在有随机噪声干扰的情况下进行了实验。

设 H 为聚类成员个数,在无噪声的情况下,对于策略一,我们首先在区间$[k_{min},k_{max}]$中随机选取参数 k,再随机选择初始中心点,用 k-means 算法得到聚类结果作为一个聚类成员,并重复 H 次以产生 H 个聚类成员;对于策略二,设 $H=5H_s$,使用 k-means 算法、single-link 算法、complete-link 算法、ave-rage-link 算法、ward-link 算法分别产生 H_s 个聚类成员。k-means 算法产生聚类成员的方式与策略一中相同。对于另外 4 种算法,我们同样先在区间$[k_{min},k_{max}]$中随机选取参数 k,然后使用相应的算法得到聚类结果作为一个聚类成员,并重复 H_s 次以产生 H_s 个聚类成员。

在增加随机噪声测试的情况下,设 p 为噪声率,H_c 为正常聚类成员,$H_n=H_c*p$。对两种策略我们均首先按无噪声的情况生成 H_c 个聚类成员,然后生成 H_n 个随机噪声成员。对于每个随机噪声成员,我们在区间$[k_{min},k_{max}]$中随机选取参数 k,对数据集中的每个数据点在标识集$\{1,2,3,\cdots,k\}$中随机选取一个标识。

我们分别选用 CSPA 算法、EA 算法、WsnnG 算法为融合算法进行实验。在无噪声的情况下,我们选取 $H=\{10,20,\cdots,100\}$,k_{min}为数据集的实际类数,$k_{max}=\{k_{min},k_{min}+2,\cdots,k_{min}+16\}$,对两种聚类成员生成策略分别进行了 90 组实验,计算 F-measure 值的平均值,并与相应的单次融合算法进行比较,实验结果如表 2(策略一)和表 3(策略二)所列。

表 2　无噪声时实验结果(策略一)

数据集	CSPA		EA		WsnnG	
		加权		加权		加权
halfrings	96.9%	[redacted]	97.3%	97.3%	75.6%	[redacted]
2D2K	97.2%	97.2%	90.6%	[redacted]	95.5%	[redacted]
Iris	95.3%	[redacted]	80.4%	[redacted]	80.8%	[redacted]
WDBC	80.0%	[redacted]	74.9%	[redacted]	88.8%	[redacted]
Wine	92.8%	[redacted]	74.9%	[redacted]	81.3%	[redacted]

表 3　无噪声时实验结果(策略二)

数据集	CSPA		EA		WsnnG	
		加权		加权		加权
halfrings	100%	100%	100%	100%	78.2%	[redacted]
2D2K	96.9%	[redacted]	73.0%	73.0%	95.5%	95.5%
Iris	94.2%	[redacted]	77.7%	77.7%	76.0%	[redacted]
WDBC	83.7%	83.7%	68.6%	[redacted]	78.5%	[redacted]
Wine	91.4%	[redacted]	52.4%	[redacted]	71.1%	[redacted]

由表 2 和表 3 的实验结果可知,不论使用哪种聚类成员生成策略,在 3 种不同的融合算法下,加权融合的融合结果普遍优于单次融合的融合结果。在策略一中,对于 5 个数据集,比较融合结果的 F-measure 值,加权 CSPA 比 CSPA 平均提高了 0.14%,最高提高了 0.3%;加权 EA 比 EA 平均提高了 1.26%,最高提高了 2.9%;加权 WsnnG 比 WsnnG 平均提高了 1.02%,最高提高了 2.2%。在策略二中,加权 CSPA 比 CSPA 平均提高了 0.26%,最高提高了 1.1%;加权 EA 比 EA 平均提高了 0.08%,最高提高了 0.3%;加权 WsnnG 比 WsnnG 平均提高了 2.28%,最高提高了 5.5%。同时,我们还发现,采用策略一选取聚类成员在除 halfrings 外的 4 个数据集上的融合结果要普遍好于采用策略二选取聚类成员的融合结果,这是因为策略二使用了 5 种不同的算法生成聚类成员,各聚类成员之间的差异性太大,从而影响了融合的效果。而 halfrings 数据集是两个半环状分布的数据集,使用各种 link 聚类算法均能得到较 k-means 算法更好的聚类结果,因此在策略二中聚类成员的质量普遍优于策略一中聚类成员,从而能得到更好的融合结果。

在增加噪声的情况下,我们选取 $H_c=\{10,20,\cdots,50\}$,k_{min}为数据集的实际类数,$k_{max}=\{k_{min},k_{min}+2,\cdots,k_{min}+16\}$,$p=\{0.6,0.8,1,1.2,1.4\}$,对两种聚类成员生成策略分别进行了 225 组实验,计算 F-measure 值的平均值,并与相应的单次融合算法进行比较,实验结果如表 4(策略一)和表 5(策略二)所列。

表 4　有噪声时实验结果(策略一)

数据集	CSPA		EA		WsnnG	
		加权		加权		加权
halfrings	96.7%	[redacted]	91.3%	[redacted]	71.9%	[redacted]
2D2K	96.8%	[redacted]	85.6%	[redacted]	88.7%	[redacted]
Iris	93.3%	[redacted]	79.7%	[redacted]	74.3%	[redacted]
WDBC	79.8%	[redacted]	72.2%	[redacted]	82.6%	[redacted]
Wine	92.6%	[redacted]	72.5%	[redacted]	78.7%	[redacted]

表 5　有噪声时实验结果(策略二)

数据集	CSPA		EA		WsnnG	
		加权		加权		加权
halfrings	99.9%	99.9%	95.0%	[redacted]	73.2%	[redacted]
2D2K	96.9%	[redacted]	70.8%	[redacted]	79.8%	[redacted]
Iris	90.6%	[redacted]	76.3%	[redacted]	66.5%	[redacted]
WDBC	82.6%	82.6%	68.5%	68.5%	68.6%	[redacted]
Wine	90.7%	[redacted]	50.5%	[redacted]	70.7%	[redacted]

由表 4 和表 5 的实验结果可知,在增加噪声的情况下,融合结果的 F-measure 值相比无噪声的情况普遍下降。但不论使用哪种聚类成员生成策略,在 3 种不同的融合算法下,加权融合的融合结果仍普遍优于单次融合的融合结果。在策略一中,对于 5 个数据集,比较融合结果的 F-measure 值,加权 CS-

(下转第 249 页)

and Communications Review, 2002, 6(4): 65-67
[3] Sharmin M, Ahmed S, Ahamed S I. MARKS (Middleware Adaptability for Resource Discovery, Knowledge Usability and Self-healing) for mobile devices of pervasive computing environments// Proceedings of the Third International Conference on Information Technology: New Generations (ITNG 2006). Las Vegas, Nevada, USA, April 2006: 306-313
[4] Lewellyn-Jones D, Merabti M, Shi Q, et al. A security framework for executables in a ubiquitous computing environment// Proceedings of Globecom 2004. Dallas, USA, November 2004: 2158-2163
[5] Hill R, Al-Muhtadi J, Campbell R, et al. A middleware architecture for securing ubiquitous computing cyber infrastructures. IEEE Distributed Systems Online, 2004, 5(9): 1-14
[6] Kalasapur S, Kumar M, Shirazi B. Evaluating service oriented architecture (SOA) in pervasive computing// Proceedings of the Fourth Annual IEEE International Conference on Pervasive Computing and Communications (PerCom 2006). Pisa, Italy, March 2006: 276-285
[7] Shirazi B, Kumar M, Sung B Y. QoS middleware support for pervasive computing applications// Proceedings of the 37th Annual Hawaii international Conference on System Sciences (HICSS' 04). Big Island, Hawaii, USA, January 2004: 294-303
[8] Liu R, Wang Y, Yang H, et al. An evolutionary system development approach in a pervasive computing environment// Proceedings of the 2004 International Conference on Cyberworlds (CW'04). Tokyo, Japan, November 2004: 194-199
[9] 戴汝为. 社会智能科学[M]. 上海: 上海交通大学出版社, 2007: 22-25
[10] Sharmin M, Ahmed S, Ahamed S I. SAFE-RD (Secure, Adaptive, Fault Tolerant, and Efficient Resource Discovery) in pervasive computing environments// Proceedings of the IEEE International Conference on Information Technology (ITCC 2005). Las Vegas, NV, USA, April 2005: 271-276
[11] Ahamed S I, Zulkernine M, Anamanamuri S. A dependable device discovery approach for pervasive computing middleware// Proceedings of the International Conference on Availability, Reliability and Security (AreS' 06). Vienna, Austria: IEEE CS Press, April 2006: 66-73
[12] Sharmin M, Ahmed S, Ahamed S I. An adaptive lightweight trust reliant secure resource discovery for pervasive computing environments// Proceedings of the Fourth Annual IEEE International Conference on Pervasive Computing and Communications (PerCom 2006). Pisa, Italy, March 2006: 258-263
[13] Ahmed S, Sharmin M, Ahamed S I. Knowledge usability and its characteristics for pervasive computing // Proceedings of the 2005 International Conference on Pervasive Systems and Computing (PSC-05) in Conjunction with the 2005 International Multi-conference in Computer Science and Engineering. Las Vegas, NV, USA: CSREA Press, June 2005: 206-209
[14] Ahmed S, Sharmin M, Ahamed S I. ETS (Efficient, Transparent, and Secured) self-healing service for pervasive computing applications. International Journal of Network Security, 2007, 4(3): 271-281
[15] Xu W, Xin Y, Lu G. A lightweight, fault-tolerant, load balancing service discovery and invocation algorithm for pervasive computing environment// Proceedings of the 3rd International Conference on Innovative Computing Information and Control (ICICIC2008). Dalian, China, June 2008
[16] 徐文拴, 辛运帏, 卢桂章, 等. 普适计算环境下信任管理模型的研究[J]. 计算机科学, 2009, 36(2): 103-106, 113

(上接第 245 页)

PA 比 CSPA 平均提高了 0.36%，最高提高了 1.0%；加权 EA 比 EA 平均提高了 2.0%，最高提高了 2.9%；加权 WsnnG 比 WsnnG 平均提高了 2.8%，最高提高了 4.3%。在策略二中，加权 CSPA 比 CSPA 平均提高了 0.44%，最高提高了 1.1%；加权 EA 比 EA 平均提高了 1.24%，最高提高了 2.5%；加权 WsnnG 比 WsnnG 平均提高了 4.56%，最高提高了 5.9%。与无噪声的情况相比，在有噪声的情况下，加权融合比单次融合提高的幅度更大。

结束语 聚类融合算法对多个聚类结果进行融合，从而得到比单一算法更为优越的聚类结果。然而，质量差的聚类成员和噪声的存在会对融合结果产生不良的影响。本文提出了一种基于属性重要性的加权聚类融合(SoA-WCE)方法，由粗糙集理论中的属性重要性度量来衡量聚类成员对融合的重要性，并据此对其赋予权重，生成加权共生矩阵，进而得到融合结果。实验结果表明，本文提出的方法能较好地处理聚类成员间的质量差异，并能有效地消减噪声对融合的影响，从而得到更好的聚类融合结果。本文中选用的融合算法均是基于共生矩阵的，但只要能合理定义聚类成员的加权方式，本文提出的 SoA-WCE 方法能扩展到其他融合算法。软聚类的加权融合方法是进一步研究的目标。

参 考 文 献

[1] Jain A K, Flynn P J. Data Clustering, A Review. ACM Computing Surveys, 1999, 31(3): 264-323
[2] 阳琳贤, 王文渊. 聚类融合方法综述[J]. 计算机应用研究, 2005, 12: 14-16
[3] Fred A L. Finding Consistent Clusters in Data Partitions// Proceedings of the Second International Workshop on Multiple Classifier Systems, 2001. Volume 2096 of Lecture Notes in Computer Science. Springer, 2001: 309-318
[4] Strehl A, Ghosh J. Cluster ensembles-a knowledge reuse framework for combining multiple partitions. Journal of Machine Learning Research, 2003, 3(3): 583-617
[5] Karypis G, Kumar V. A fast and high quality multilevel scheme for partitioning irregular graphs. SIAM Journal on Scientific Computing, 1998, 20(1): 359-392
[6] Fred A L, Jain A K. Data clustering using evidence accumulation // Proceedings of the 16th International Conference on Pattern Recognition (ICPR 2002). volume 4, 2002: 276-280
[7] Ayad H, Kamel M. Finding natural clusters using multi-clusterer combiner based on shared nearest neighbors// Proceedings of the 4th International Workshop on Multiple Classifier Systems (MCS'03), 2003. Volume 2709 of Lecture Notes in Computer Science. Springer, 2003: 166-175
[8] 王国胤. Rough 集理论与知识获取[M]. 西安: 西安交通大学出版社, 2001
[9] Merz C, Murphy P. UCI repository of machine learning databases. http://archive.ics.uci.edu/ml/
[10] Larson B, Aone C. Fast and effective text mining using linear-time document clustering// Conference on Knowledge Discovery in Data, Proceeding of the fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 1999: 16-22