聚类集成时机的确定

孟晓龙 杨 燕 王红军 肖文超
(西南交通大学信息科学与技术学院 成都 610031)

摘 要 使用集成学习技术可以提高聚类性能。在实验中发现,当各聚类成员聚类迭代到中后期时进行集成所得的结果会优于其迭代完全停止时进行集成所得的结果。利用集成网络泛化能力的偏差-方差分解理论对聚类集成过程中的上述现象进行解释,将提高集成网络间泛化能力的早期停止准则应用于聚类集成过程,并提出聚类集成时机的概念。对比实验表明,基于早期停止准则的聚类集成得到的结果较好,且更节约聚类集成的时间,为寻求聚类集成的最佳时机提供了可行性建议和方法。

关键词 聚类集成,集成时机,泛化能力,早期停止准则
中图法分类号 TP181 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2015.7.011

Occasion Determination of Clustering Ensemble

MENG Xiao-long YANG Yan WANG Hong-jun XIAO Wen-chao
(School of Information Science and Technology, Southwest Jiaotong University, Chengdu 610031, China)

Abstract Ensemble learning technique may improve the clustering performance. In the experiment, we discovered that combining the mid-to-late solutions of cluster members in different initial conditions probably get the better ensemble results than combining the end ones. We used the bias/variance trade-off of generalization ability in ensemble network to explain this phenomenon, applied the early stopping rules to the clustering ensemble and proposed the concept of clustering ensemble occasion. The experimental results show that the performance of clustering ensemble based on the early stopping rules is superior to that based on the end solutions of cluster members, while the former takes less time, thus giving some useful suggestions for seeking the best clustering ensemble occasion.

Keywords Clustering ensemble, Ensemble occasion, Generalization ability, Early stopping rules

1 引言

聚类分析作为机器学习中热门的研究领域,广泛应用于如数据挖掘、模式识别、图像分割等诸多领域[1-3]。然而,任何一种聚类算法都不能准确揭示数据纷繁复杂的簇结构。现实世界的数据具有各种结构或形状,单一聚类算法很难实现对簇结构的识别。

聚类集成利用集成学习技术,通过合并多个聚类结果,得到最终统一的聚类划分。2002年,Strehl等[4]提出聚类集成(Clustering Ensemble)的概念,Fred等[5]提出以随机初始点的K-means聚类算法作为基聚类器,以投票法为共识函数的聚类集成方法。2006年,Zhou等[6]提出选择性聚类集成概念,发现并证明选择部分聚类成员所构建的集成要优于使用全部聚类成员所构建的集成。Lu等[7]提出了基于协方差的差异性度量方法,将实验得到的一个协方差区间作为选择基聚类结果进行集成的标准。罗会兰等[8]指出选择适中差异性的聚类成员进行集成比选择最大差异性的成员更能保证集成性能。我们通过实验发现各聚类成员聚类迭代到中后期进行集成会优于其迭代完全停止(聚类过程目标函数变化为零)进行集成所得到结果。

由此,利用集成网络泛化能力的偏差-方差分解理论[9]解释聚类集成的上述现象。借鉴传统提高集成网络间泛化能力的早期停止准则[10],得到4种基于早期停止准则的聚类集成算法,并提出聚类集成时机这一概念,即研究当聚类阶段进行到何种程度时进行集成,聚类集成的整体效果最佳,为寻求恰当的聚类集成时机提供可行性建议和方法。

2 聚类集成时机的提出

2.1 聚类集成时机的实验

本文使用表1所列的UCI数据集,以随机初始点的K-means聚类算法作为基聚类器,将不同迭代时期的聚类成员结果用投票法[5]进行集成(该算法记作Voting-KM),依次得到10个等间隔迭代时期聚类成员的集成结果,按F-measure指标评价值[11]排序并打分(由大到小排序得10至1分,并求

到稿日期:2014-06-24 返修日期:2014-09-11 本文受国家自然科学基金(61170111,61134002),西南交通大学牵引动力国家重点实验室自主研究课题(2012TPL_T15)资助。

孟晓龙(1988-),硕士生,主要研究方向为聚类集成、数据挖掘;杨 燕(1964-),教授,博士生导师,主要研究方向为数据挖掘、计算智能、集成学习,E-mail:yyang@swjtu.edu.cn(通信作者);王红军(1977-),副研究员,硕士生导师,主要研究方向为机器学习、集成学习与数据挖掘等;肖文超(1989-),硕士生,主要研究方向为聚类集成、数据挖掘。

和记为 SCORE)，如表 2 所列。

表 1　UCI 数据集描述

Data Sets	样本数	属性	类数
Balance	625	4	3
DNA	2000	180	3
Ionosphere	351	34	2
Iris	150	4	3
Landsat	2000	36	6
Segment	2310	18	7
Vehicle	846	18	4
Vote	435	16	2
Wine	178	13	3
Zoo	101	16	7

同样，将上述方法依次得到的 UCI 数据集的 10 个等间隔迭代时期的聚类集成结果，用 F-measure、Dunn Index 和 DBI 等[11-13]指标评价排序并综合打分，从而得到各迭代时期聚类集成结果在 3 个指标下得分的累加和，如图 1 所示。其中，F-measure 通过组合信息检索中查准率与查全率的思想来评价聚类集成结果的好坏，其值越大聚类集成结果越好；Dunn Index 通过计算簇间距离和簇直径之间的比值来度量聚类集成结果的好坏，其值越大聚类集成结果越好；DBI 指标利用几何原理进行运算，分别以聚类间离散程度与位于同一类内的数据对象的紧密程度作为依据来评价聚类集成结果的好坏，其值越小聚类集成结果越好。

表 2　不同迭代时期 Voting-KM 聚类集成结果 F-measure 值得分表

Data Sets	1	2	3	4	5	6	7	8	9	10
Balance	0.4350	0.4411	0.4305	0.4402	0.4310	**0.4510**	0.4430	0.4443	0.4428	0.4422
DNA	0.3671	0.6197	0.6902	0.7189	0.7354	0.7441	0.7480	0.7509	0.7510	**0.7512**
Ionosphere	0.4709	0.5824	0.6397	0.6728	0.6859	0.6931	0.6984	**0.6995**	**0.6995**	**0.6995**
Iris	0.6573	0.7907	0.8466	0.8653	0.8570	0.8675	**0.8770**	0.8651	0.8650	0.8432
Landsat	0.5469	0.6388	0.6613	**0.6736**	0.6679	0.6653	0.6613	0.6593	0.6573	0.6581
Segment	0.5324	0.5915	0.5893	0.6281	0.5956	0.6240	0.6239	**0.6289**	0.6141	0.6209
Vehicle	0.3359	0.3518	0.3502	0.3578	0.3629	0.3643	**0.3671**	0.3654	0.3646	0.3650
Vote	0.7987	0.8576	0.8599	0.8613	0.8622	0.8635	0.8646	**0.8647**	0.8641	0.8641
Wine	0.7047	0.8247	0.8888	0.9256	0.9377	0.9436	0.9478	0.9494	**0.9503**	0.9497
Zoo	0.4285	0.4636	0.4471	0.4432	0.4532	0.4739	**0.4890**	0.4836	0.4638	0.4570
SCORE	12	29	30	54	49	73	82	82	70	69

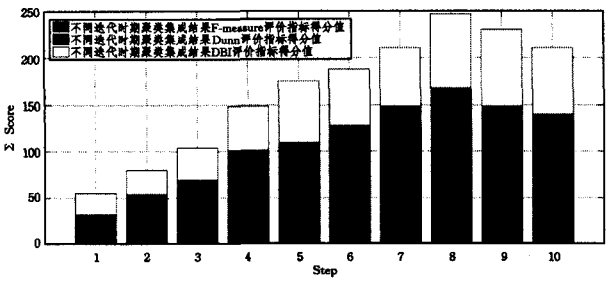

图 1　不同迭代时期 Voting-KM 聚类集成结果采用 3 种评价指标的得分累计和

由表 2 和图 1 可以看出，当聚类成员聚类迭代到中后期时，如本实验中对第 8 或第 9 个等间隔迭代的结果进行投票法集成时，聚类集成结果在各评价指标下的得分较高，这说明当各聚类成员聚类迭代到中后期进行集成所得到的结果会优于其迭代完全停止(聚类过程目标函数变化为零)时进行集成所得到的结果。

2.2　泛化能力的偏差-方差分解理论

利用泛化能力的偏差-方差分解理论[9]可以有效地解释聚类集成的上述现象。假设某个数据集 $\chi=\{x_1,x_2,\cdots,x_n\}$，且 $|\chi|=n$，利用某种聚类算法将其划分为 k 簇，聚类结果解空间可表示为 $\mathbb{R}=\{R_1,R_2,\cdots,R_m\}$，其中 $m=S(n,k)$，根据第二类 Stirling 数 $S(n,k)$ 的定义[13]，满足

$$S(n,k)=\frac{1}{k!}\sum_{i=0}^{k}(-1)^i\binom{k}{i}(k-i)^n \tag{1}$$

且聚类过程满足目标函数最小

$$\min J=\sum_{j=1}^{k}\sum_{i=1}^{n}\mu_{ij}\cdot D^2(x_i,p_j) \tag{2}$$

其中，μ_{ij} 表示数据集 χ 中某一数据 x_i 对于类簇中心 p_j 的隶属度，$D^2(x_i,p_j)$ 表示数据 x_i 到类簇中心 p_j 的距离。

对于数据集 χ，在某一初始条件 α 下得到的聚类划分结果可用 V^α 表示，则 V^α 等于解空间 $\mathbb{R}$ 中某个解 R_i 的概率

$$P(V^\alpha=R_i)=\omega_\alpha \text{ 且 } \sum_\alpha\omega_\alpha=1 \tag{3}$$

聚类集成结果可以近似看作各聚类成员划分结果的加权平均值，即

$$\overline{V}=\sum_\alpha\omega_\alpha V^\alpha \tag{4}$$

假设上述 2.1 节聚类集成时机实验中，某一聚类成员的输出划分为 F，则整个聚类集成网络的泛化能力可以定义为：

$$E=(F-\overline{V})^2 \tag{5}$$

对聚类集成网络的泛化能力进行偏差-方差分解，过程如下：

$$\begin{aligned}E&=(F-\overline{V})^2\\&=F^2-2F\overline{V}+(\overline{V})^2\\&=\sum_\alpha\omega_\alpha F^2-2\sum_\alpha\omega_\alpha V^\alpha F+\sum_\alpha\omega_\alpha(\overline{V})^2\\&=\sum_\alpha\omega_\alpha F^2-2\sum_\alpha\omega_\alpha V^\alpha F+\sum_\alpha\omega_\alpha V^\alpha-\sum_\alpha\omega_\alpha V^\alpha+\\&\quad 2\sum_\alpha\omega_\alpha(\overline{V})^2-\sum_\alpha\omega_\alpha(\overline{V})^2\\&=\sum_\alpha\omega_\alpha(F-V^\alpha)^2-(\sum_\alpha\omega_\alpha V^\alpha-2\sum_\alpha\omega_\alpha V^\alpha\overline{V}+\sum_\alpha\omega_\alpha(\overline{V})^2)\\&=\sum_\alpha\omega_\alpha(F-V^\alpha)^2-\sum_\alpha\omega_\alpha(V^\alpha-\overline{V})^2\end{aligned} \tag{6}$$

其中，$\overline{E}=\sum_\alpha\omega_\alpha(F-V^\alpha)^2$ 表示聚类成员个体的平均泛化误差，$\overline{A}=\sum_\alpha\omega_\alpha(V^\alpha-\overline{V})^2$ 表示聚类集成网络整体的差异度。

上述分解公式指明，若要提高聚类集成效果，不仅聚类成员个体的泛化能力应尽可能的高，而且集成网络整体的差异性也应尽可能的小。各聚类成员在其初始条件下聚类迭代到中后期时，既具有平均泛化误差大的聚类成员个体，又具有差异性适中且准确性高的聚类集成网络，故构建的聚类集成更优，这就有效地解释了 2.1 节实验中的现象。

3　聚类集成时机的确定

借鉴传统提高集成网络间泛化能力的思路，补充完善 Lodwich 等人[12]提出的泛化损失法，提出了梯度比法和低步

进值法，启发式地将早期停止准则应用于聚类集成过程，具体实现技术叙述如下。为了更清晰地描述这些准则，我们定义 J 为聚类算法迭代过程中的目标函数值，则 $J(t)$ 表示聚类成员在第 t 次迭代时 J 的值。

3.1 梯度比法(Gradient Ratio)

用 $J_{gre}(t)$ 表示第 t 次迭代时的梯度值，即

$$J_{gre}(t)=J(t)-J(t-1) \tag{7}$$

那么，在第 t 次迭代与第 2 次迭代的梯度比为

$$GR(t)=\frac{J_{gre}(t)}{J_{gre}(2)}=\left(\frac{J(t)-J(t-1)}{J(2)-J(1)}\right),t\neq 1 \tag{8}$$

较小的梯度比值表明聚类集成网络的泛化能力减弱，因此得到早期停止准则 1：当第 t 次迭代时的梯度比值小于某一阈值 α_1 时，则停止聚类过程。

3.2 泛化损失法(Generalization Loss)

用 $J_{opt}(t)$ 表示第 t 次迭代之前 $t-1$ 次迭代 J 的最小值，即

$$J_{opt}(t)=\min_{t'\leqslant t-1} J(t') \tag{9}$$

那么，定义在第 t 次迭代时泛化损失为

$$GL(t)=\frac{J(t)-J_{opt}(t)}{J_{opt}(t)} \tag{10}$$

较小的泛化损失表明聚类集成网络的泛化能力减弱，因此得到早期停止准则 2：一旦泛化损失小于某一阈值 α_2，则停止聚类过程。

3.3 低步进值法(Low Progress)

泛化损失法有较高的几率被修复，即 $J(t)$ 在某一时间段是抖动的，导致 $GL(t)$ 在这一时间段时而大于阈值，时而小于阈值。那么，考虑第 t 次迭代之前的一阶段 J 的变化

$$LP_k(t)=\frac{\underset{t'=t-k+1}{\overset{t}{\operatorname{avg}}} J(t')}{\underset{t'=t-k+1}{\overset{t}{\min}} J(t')}-1 \tag{11}$$

其中，$\underset{t'=t-k+1}{\overset{t}{\operatorname{avg}}} J(t')$ 表示截至第 t 次迭代之前 k 长度阶段 J 值的平均值，$\underset{t'=t-k+1}{\overset{t}{\min}} J(t')$ 表示这 k 长度阶段 J 值的最小值。

较低的低步进值表明聚类集成网络的泛化能力减弱，因此得到早期停止准则 3：一旦低步进值小于某一阈值 α_3，则停止聚类过程。通常 k 值可取 5。低步进值措施考虑了第 t 次迭代的前一阶段 J 的变化，可有效地避免目标函数 J 值抖动带来的影响。

3.4 组合法(Combination of Rules)

上述早期停止准则各有长处与不足，且计算复杂度也较低，故可以将上述任意准则组合应用。

$$ESR_{combined}=ESR_1 \vee ESR_2 \vee \cdots \vee ESR_n \tag{12}$$

组合通常表示集合中各元素的一个无序选择[13]，这就意味将有 2^n 个组合。广义来讲，其变化方式也可以是若干元素之间的交、并、补等操作，从而将会有更多的组合方式。后文实验选取的组合法为低步进法与梯度比法的交集，即要求同时满足早期停止准则 1 和 3。

如果早期停止准则太晚停止，那么聚类集成网络的泛化能力下降；如果太早停止，那么聚类成员个体的准确性就难以保证。两种情况都会导致聚类集成效果较差，故引入聚类集成时机的概念，即聚类阶段进行到何种程度进行集成，聚类集成效果最佳。

4 实验结果及分析

采用 4 种早期停止准则：梯度比法、泛化损失法、低步进值法和组合法，以随机初始类中心的 K-means 聚类算法为基聚类算法，以投票法作为共识函数的聚类集成算法分别简称为 ESKMV-GR、ESKMV-GL、ESKMV-LP 和 ESKMV-CR；同样，以 FCM 聚类算法为基聚类算法的聚类集成算法分别简称为 ESFCMV-GR、ESFCMV-GL、ESFCMV-LP 和 ESFCMV-CR。为了实验方便，统一将以 K-means 聚类算法作为基聚类算法的早期停止聚类集成算法阈值 α 的大小定为 $1/N$，而以 FCM 聚类算法作为基聚类算法的 α 则定为 $1/2N$，N 为各数据集的样本数目。表 3—表 5 是上述不同聚类集成算法在 10 个 UCI 数据集上的 F-measure、Dunn Index 和 DBI 评价指标累计得分表。无论是以 K-means 还是 FCM 为基聚类算法，4 种早期停止准则得到的聚类集成算法累计得分与传统投票法聚类集成算法(Voting-KM 和 Voting-FCM)的得分相近，在某些评价指标时得分甚至更高，这就说明本文提出的 4 种早期停止准则(梯度比法、泛化损失法、低步进值法和组合法)是有效的。

但以 FCM 聚类算法作为基聚类算法时，泛化损失法的聚类集成算法表现不太理想，而相同阈值下的低步进值法聚类集成算法表现较好，其原因是泛化损失法易受到聚类目标函数 $J(t)$ 抖动的影响，较早地停止聚类阶段，影响聚类集成结果；而基于低步进值法的聚类集成算法考虑了第 t 次迭代之前一阶段 J 的变化，可有效地避免目标函数 J 值抖动带来的影响，从而得到了满意的聚类集成结果。

表 3 不同聚类集成算法在 10 个 UCI 数据集上的 F-measure 值

Data Sets	ESKMV				Voting	ESFCMV				Voting
	-GR	-GL	-LP	-CR	-KM	-GR	-GL	-LP	-CR	-FCM
Balance	**0.4501**	0.4496	0.4479	0.4377	0.4410	**0.4547**	0.4365	0.4512	0.4389	0.4427
DNA	0.7292	0.7361	0.7279	0.7321	**0.7470**	**0.5796**	0.5528	0.5246	0.5317	0.5175
Ionosphere	0.6992	0.6993	0.6992	**0.6995**	0.6994	**0.6995**	0.6946	0.6991	0.6991	0.6991
Iris	0.8458	**0.8662**	0.8546	0.8613	0.8599	0.8915	**0.8940**	0.8861	0.8879	0.8913
Landsat	0.6622	0.6603	**0.6639**	0.6603	0.6581	**0.6939**	0.6638	0.6924	0.6898	0.6864
Segment	0.6062	0.6143	**0.6241**	0.6059	0.6122	0.6757	0.6214	**0.6764**	0.6761	0.6748
Vehicle	0.3616	0.3667	0.3657	**0.3688**	0.3653	0.3743	0.3716	0.3745	**0.3753**	0.3667
Vote	0.8643	**0.8647**	0.8630	0.8645	0.8642	**0.8596**	0.8583	0.8590	0.8591	0.8595
Wine	0.9482	**0.9511**	0.9485	0.9502	0.9497	0.9278	0.6773	**0.9500**	0.9481	**0.9500**
Zoo	0.4689	**0.4959**	0.4929	0.4755	0.4611	0.3437	0.2003	0.5168	0.4375	**0.6530**
SCORE	22	41	28	33	26	39	18	32	31	30

表 4　不同聚类集成算法在 10 个 UCI 数据集上的 Dunn 值

Data Sets	ESKMV				Voting	ESFCMV				Voting
	-GR	-GL	-LP	-CR	-KM	-GR	-GL	-LP	-CR	-FCM
Balance	1.0668	1.0638	1.0513	**1.0698**	1.0589	1.0412	1.0493	**1.0512**	1.0327	1.0460
DNA	0.2961	0.2963	0.2966	0.2974	**0.2984**	0.2549	0.2425	0.2565	**0.2618**	0.2578
Ionosphere	1.0162	1.0162	**1.0163**	1.0162	**1.0163**	1.0169	**1.0206**	1.0161	1.0161	1.0161
Iris	2.4090	2.4115	2.4068	**2.4280**	2.3834	2.2330	2.2279	2.2845	2.2787	**2.3162**
Landsat	1.0907	**1.0917**	1.0895	1.0899	1.0859	1.0853	0.9711	1.0865	**1.0890**	1.0844
Segment	**0.5539**	0.4259	0.4825	0.5018	0.5255	0.5132	0.4311	0.5147	**0.5156**	0.5084
Vehicle	1.1224	1.1153	**1.1427**	1.1345	1.1273	1.0300	1.0085	1.0391	1.0262	**1.0935**
Vote	1.3375	1.3386	1.3383	1.3392	**1.3404**	**1.3377**	1.3373	1.3377	1.3377	1.3376
Wine	0.6919	0.6925	0.6767	**0.6946**	0.6882	0.6768	0.5866	**0.6815**	0.6786	**0.6815**
Zoo	0.3295	0.3330	0.3228	0.3237	**0.3462**	0.2066	0.1664	0.3388	0.2441	**0.4272**
SCORE	27	29	23	38	33	26	17	37	34	36

表 5　不同聚类集成算法在 10 个 UCI 数据集上的 DBI 值

Data Sets	ESKMV				Voting	ESFCMV				Voting
	-GR	-GL	-LP	-CR	-KM	-GR	-GL	-LP	-CR	-FCM
Balance	1.7984	**1.7932**	1.8091	1.7938	1.8009	1.8242	**1.8141**	1.8217	1.8475	1.8258
DNA	6.4654	6.5268	6.4827	6.4916	**6.4564**	7.4711	7.7923	7.3922	**7.3116**	7.3641
Ionosphere	1.5158	1.5158	**1.5156**	1.5157	1.5157	1.5148	**1.5077**	1.5159	1.5159	1.5159
Iris	0.5558	**0.5625**	0.5689	0.5698	0.5729	0.5924	0.6044	**0.5868**	0.5946	0.5883
Landsat	1.0236	1.0113	**0.9979**	1.0237	1.0079	**1.0059**	1.3686	1.0131	1.0336	1.0280
Segment	1.4353	1.5363	1.4221	1.4767	**1.4078**	1.3856	1.4291	1.4026	1.3777	**1.3597**
Vehicle	1.3582	1.3504	1.3166	1.3426	**1.3030**	**1.0059**	1.4100	1.3746	1.3575	1.3824
Vote	1.4863	**1.4862**	**1.4864**	1.4864	1.4866	**1.4864**	1.4876	1.4867	1.4864	1.4868
Wine	**1.1037**	1.1145	1.1229	1.1165	1.1239	**1.1314**	1.1458	1.1319	1.1277	1.1265
Zoo	2.0384	**1.9044**	2.1557	2.0284	2.0183	3.0658	5.5235	**2.0451**	1.9811	2.1675
SCORE	30	32	30	26	32	34	18	31	34	33

表 6　不同聚类集成算法聚类阶段迭代次数对比表

Data Sets	ESKMV				Voting	ESFCMV				Voting
	-GR	-GL	-LP	-CR	-KM	-GR	-GL	-LP	-CR	-FCM
Balance	8.92	7.22	10.343	9.4	13.991	3.655	3.375	4.033	4.048	6.049
DNA	8.4583	8.215	11.495	10.201	23.01	3.219	3	5.01	5.1467	5.155
Ionosphere	5.95	5.6767	6.2217	6.2233	6.2683	6.616	3.550	10.143	9.762	16.639
Iris	5.01	5.055	6.4233	6.19	7.2046	9.005	8.7133	11.423	10.838	16.148
Landsat	18.005	19.605	21.975	21.293	29.051	9.005	19.520	27.201	25.851	62.313
Segment	11.96	11.45	15.505	14.716	18.383	7.6233	23.895	31.361	30.461	57.180
Vehicle	10.616	10.02	13.746	11.876	20.188	17.030	12.625	18.788	17.651	47.010
Vote	4.5483	4.5667	6.3883	6.3367	6.935	6.376	3.180	10.456	9.890	18.836
Wine	5.97	6.125	6.4233	6.6117	6.6638	8.758	3.513	12.883	12.193	19.798
Zoo	4.4	4.3317	4.5617	4.5667	4.6333	9.2583	3.1217	13.293	11.42	67.272

为了比较 4 种早期停止准则聚类集成算法在时间上的优势，聚类阶段停止时的平均迭代次数值如表 6 所列。相对传统投票法的聚类集成算法，基于早期停止准则的聚类集成算法可以较早地结束聚类阶段，进入集成阶段，时间上有明显的优势。但可以发现以 FCM 算法为基聚类算法时，聚类集成算法会过早地结束聚类阶段，进入集成阶段，导致部分数据如 Breast、Zoo 等表现不佳，原因在于早期停止准则中阈值 α 的设定不能对所有数据集有效。总的来说，将早期停止准则应用于聚类集成过程，可为寻求恰当的聚类集成时机提供可行性建议和方法。

结束语　本文提出聚类集成时机的概念，将早期停止准则应用于聚类集成过程，以寻求聚类集成的最佳时机。

在进行本文工作时，相关研究工作还非常少，鉴于聚类集成与神经网络结构方面的相似，可以给聚类集成的相关研究提供更多的启发。在实验时，早期停止准则公式中阈值 α 的设定仍然不能对所有数据集的聚类集成结果有效，如何寻找合适的 α 取值方法会是下一步工作的重点。

参 考 文 献

[1] Han Jia-wei, Kamber M, Pei J. Data Mining Concepts and Techniques [M]. Beijing: China Machine Press, 2012

[2] 郭鹏飞，刘万军，林琳，等. 结合随机游走和 FCM 的脑图像分割方法[J]. 计算机科学，2014，41(7)：322-325
Guo Peng-fei, Liu Wang-jun, Lin Lin, et al. Brain Image Segmentation Method Based on FCM and Random Walk[J]. Computer Science, 2014, 41(7): 322-325

[3] 吕明磊，刘冬梅，曾智勇. 一种改进的 K-means 聚类算法的图像检索方法[J]. 计算机科学，2013，40(8)：285-288
Lv Ming-lei, Liu Dong-mei, Zeng Zhi-yong. Novel Image Retrieval Method of Improved K-means Clustering Algorithm[J]. Computer Science, 2013, 40(8): 285-288

[4] Strehl A, Ghosh J. Cluster ensembles-A knowledge reuse framework for combining partitionings[J]. Journal of Machine Learning Research, 2002, 3: 583-617

（下转第 84 页）

architectures under performance constraints[C]// Proceedings of the 2003 Asia and South Pacific Design Automation Conference. USA: ACM, 2003: 233-239

[9] Hu J, Marculescu R. Energy-and performance-aware mapping for regular NoC architectures[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2005, 24 (4): 551-562

[10] Hamedani P K, Hessabi S, Sarbazi-Azad H, et al. Exploration of temperature constraints for thermal aware mapping of 3D Networks on Chip[C]// Proceedings of the 20th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP). USA: IEEE, 2012: 499-506

[11] Murali S, De Micheli G. Bandwidth-constrained mapping of cores onto NoC architectures[C]// Proceedings of the conference on Design, automation and test in Europe-Volume 2. USA: IEEE, 2004: 20896

[12] Zhao K, Bian J. Instruction-level hardware/software partition through DFG exploration[C]// Proceedings of the 15th International Conference on Computer Supported Cooperative Work in Design (CSCWD). USA: IEEE, 2011: 55-60

[13] Bouchhima A, Gerin P, Pétrot F. Automatic instrumentation of embedded software for high level hardware/software co-simulation[C]// Proceedings of the Design Automation Conference Asia and South Pacific (ASP-DAC). USA: IEEE, 2009: 546-551

[14] Wang D, Li S, Dou Y. Collaborative hardware/software partition of coarse-grained reconfigurable system using evolutionary ant colony optimization[C]// Proceedings of the Design Automation Conference Asia and South Pacific (ASP-DAC). USA: IEEE, 2008: 679-684

[15] Srinivasan V, Govindarajan S, Vemuri R. Fine-grained and coarse-grained behavioral partitioning with effective utilization of memory and design space exploration for multi-FPGA architectures[J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2001, 9(1): 140-158

[16] Wolf W. Object-oriented cosynthesis of distributed embedded systems[J]. ACM Transactions on Design Automation of Electronic Systems (TODAES), 1996, 1(3): 301-314

[17] Henkel J, Ernst R. An approach to automated hardware/software partitioning using a flexible granularity that is driven by high-level estimation techniques[J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2001, 9(2): 273-289

[18] Chai S, Li Y, Wang J, et al. A List Simulated Annealing Algorithm for Task Scheduling on Network-on-Chip[J]. Journal of Computers, 2014, 9(1): 176-182

[19] Liu W, Gu Z, Xu J, et al. Satisfiability modulo graph theory for task mapping and scheduling on multiprocessor systems[J]. IEEE Transactions on Parallel and Distributed Systems, 2011, 22(8): 1382-1389

[20] Tahaee S A, Jahangir A H, Habibi-Masouleh H. Improving the performance of heuristic searches with judicious initial point selection[C]// Proceedings of the Fifth IEEE International Symposium on Embedded Computing. USA: IEEE, 2008: 14-19

[21] Hu J, Marculescu R. Energy-aware communication and task scheduling for network-on-chip architectures under real-time constraints[C]// Proceedings of the Design, Automation and Test in Europe Conference and Exhibition. USA: IEEE, 2004: 234-239

[22] Vahid F. Partitioning sequential programs for CAD using a three-step approach[J]. ACM Transactions on Design Automation of Electronic Systems (TODAES), 2002, 7(3): 413-429

[23] Wiangtong T, Cheung P Y K, Luk W. Comparing three heuristic search methods for functional partitioning in hardware-software codesign[J]. Design Automation for Embedded Systems, 2002, 6 (4): 425-449

(上接第 51 页)

[5] Fred A, Jain A K. Data clustering using evidence accumulation [C]// Proceedings of the 17th International Conference on Pattern Recognition. 2002: 276-280

[6] Zhou Z H, Tang W. Cluster ensemble [J]. Knowledge-Based Systems, 2006, 19(1): 77-83

[7] Lu X Y, Yang Y, Wang H J. Selective clustering ensemble based on covariance[C]// Proceedings of the 11th International Workshop on Multiple Classifier Systems, 2013. LNCS, 2013, 7872: 179-189

[8] 罗会兰,孔繁胜,李一啸. 聚类集成中的差异性度量研究[J]. 计算机学报, 2007, 30(8): 1315-1324
Luo Hui-lan, Kong Fan-sheng, Li Yi-xiao. An Analysis of Diversity Measures in Clustering Ensembles[J]. Chinese Journal of Computers, 2007, 30(8): 1315-1324

[9] Krogh A, Vedelsby J. Neural network ensembles, cross validation, and active learning[C]// Proceedings of NIPS94-Neural Information Processing Systems: Natural and Synthetic, 1994. Advances in Neural Information Processing Systems 7, MIT Press, 1995: 231-238

[10] Lodwich A, Rangoni Y, Breuel T. Evaluation of robustness and performance of early stopping rules with multilayer perceptrons [C]// Proceedings of the 2009 International Joint Conference on Neural Networks. 2009: 1877-1884

[11] 杨燕,靳蕃,Kamelk M. 聚类有效性评价综述[J]. 计算机应用研究, 2008, 25(6): 1632-1638
Yang Yan, Jin Fan, Kamel M. Survey of clustering evaluation [J]. Application Research of Computers, 2008, 25(6): 1632-1638

[12] Pakira M K, Bandyopadhyay S, Maulik U. Validity index for crisp and fuzzy clusters [J]. Pattern Recognition, 2004, 37: 487-501

[13] Brualdi. Introductory Combinatorics, Fifth Edition[M]. Beijing: Prentice Hall, 2012