一种基于抽样的大规模混合数据聚类集成算法

庞天杰[1] 梁吉业[1,2]
(太原师范学院计算机系 太原 030619)[1]
(山西大学计算智能与中文信息处理教育部重点实验室 太原 030006)[2]

摘 要 混合数据聚类是聚类分析中一个重要的问题。现有的混合数据聚类算法主要是在全体样本的相似性度量的基础上进行聚类,因此对大规模数据进行聚类时,算法效率不高。基于此,设计了一种新的抽样策略,在此基础上,提出了一种基于抽样的大规模混合数据聚类集成算法。该算法对利用新的抽样策略得到的多个样本子集分别进行聚类,并将结果集成得到最终聚类结果。实验证明,与改进的K-prototypes算法相比,该算法的效率有了显著提高,同时聚类有效性指标基本相同。

关键词 聚类,大规模混合数据,聚类集成,抽样,有效性指标
中图法分类号 TP391 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.9.041

Clustering Ensemble Algorithm for Large-scale Mixed Data Based on Sampling

PANG Tian-jie[1] LIANG Ji-ye[1,2]
(Department of Computer Science, Taiyuan Normal University, Taiyuan 030619, China)[1]
(Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, Shanxi University, Taiyuan 030006, China)[2]

Abstract In clustering analysis, one of the important problems is mixed data clustering. The clustering of existing algorithms is mainly based on similarity measurement of all samples. Therefore, the efficiency of clustering for large-scale data is not high. So we designed a new sampling strategy and proposed an ensemble algorithm for large-scale mixed data based on sampling. This new algorithm clusters subsets which are obtained by the use of the new sampling strategy respectively and the final clustering results can be gotten by clustering ensemble. Experiment shows that the efficiency of algorithm is improved significantly and the clustering validity indexes are almost the same compared with the modified K-prototypes algorithm.

Keywords Clustering, Large-scale mixed data, Clustering ensembles, Sampling, Validity index

聚类分析是机器学习的一个重要研究领域,传统的聚类分析算法[1-6]主要针对单一数据类型进行研究。近年来,有学者提出了一些针对混合型数据的聚类算法。例如,Huang[7]在K-means和K-modes算法的基础上,提出了K-prototypes算法,该方法是对混合数据中的对象提出的新的相似性度量方法,采用与K-means和K-modes相同的策略对数据进行聚类;Liang等[8]针对K-prototypes算法中相似性度量方法进行了改进,提出了改进的K-prototypes算法,并在此基础上提出了聚类个数的确定方法;He等[9]提出了CEBMDC算法,该算法将数据集按照数值属性和符号属性分别聚类,然后将得到的两组聚类结果进行集成;Luo等[10]利用谱聚类和聚类集成技术对混合数据进行了聚类。上述方法都在一定程度上解决了混合数据的聚类问题。但是由于这些方法都是在对全体样本相似性度量的基础进行聚类,因此对大规模数据聚类时,上述算法所消耗的时间将大大增加。

我们知道,集成学习是机器学习领域新的重要研究方向。聚类集成是将一个数据集进行多次聚类的结果组合成统一的聚类结果的方法,目前已有学者提出了许多聚类集成算法。与传统聚类算法相比,聚类集成算法有以下优势:

(1)可以提高聚类结果的质量;

(2)可以提高聚类算法的鲁棒性;

(3)对非平衡和非线性可分等特殊分布的数据可以得到较好的聚类结果;

(4)特别对于大规模数据,可以采用并行的方法,来提高聚类算法效率。

在此基础上,本文提出了一种基于抽样的大规模混合数据集成聚类算法。在该算法中,首先设计了一种新的抽样策略,按照这种策略在全体数据集上进行多次随机抽样,在每次的抽样结果中利用改进的K-prototypes算法进行聚类,找出聚类中心,并对全体数据进行划分得到多个聚类结果,最后对聚类结果进行集成,得到最终的聚类结果。实验证明,该算法是有效的。

到稿日期:2015-10-20 返修日期:2016-03-02 本文受国家自然科学基金项目:"用户行为数据"稀疏表示的理论与方法研究(61273294),山西省回国留学人员科研资助项目:基于多粒度与变粒度的群决策方法研究(2013-101)资助。
庞天杰(1980—),男,硕士,讲师,主要研究方向为数据挖掘与机器学习,E-mail:pangtj@tynu.edu.cn;梁吉业(1962—),男,教授,博士生导师,主要研究方向为数据挖掘与机器学习。

1 相关知识

1.1 混合数据信息系统

$MDT=(U,A,V,f)$是一个混合数据信息系统，其中，$U=\{x_1,x_2,\cdots x_n\}$为样本集，$x_i=(x_{i1},x_{i2},\cdots,x_{im})$为包含 m 个属性的样本；$A=A^r\cup A^c$ 为属性集合，A^r 为数值属性集合，A^c 为符号属性集合；$V=\bigcup_{a\in A}V_a$，V_a 为属性 a 的值域；$f:U\times A\to V$，对于任意 $x\in U,a\in V$，有 $f(x,a)\in V_a$。

MDT 可以表示为 $NDT=(U,A^r,V,f)$ 和 $CDT=(U,A^c,V,f)$的组合，NDT 称为数值数据系统，CDT 称为符号数据系统。

1.2 混合数据信息系统中相似性度量

由于传统的 K-means 和 K-modes 算法只能对单一数据类型的数据集进行聚类，因此 Huang 等针对包含数值型和符号型的混合数据提出了一种新的相似性度量方法。

$$D(x,y)=D_{A^r}(x,y)+\gamma D_{A^c}(x,y) \tag{1}$$

其中

$$D_{A^r}(x,y)=\sum_{a\in A^r}(f(x,a)-f(y,a))^2$$

$$D_{A^c}(x,y)=\sum_{a\in A^c}d_a(x,y)$$

$$d_a(x,y)=\begin{cases}0, & f(x,a)=f(y,a)\\ 1, & f(x,a)\neq f(y,a)\end{cases}$$

在 K-prototypes 算法中参数 γ 的选取对最终聚类结果有较大的影响，因此 Liang 等提出了一种新的相似性度量方法：

$$D(x,y)=\frac{|A^r|}{|A|}D_{A^r}(x,y)+\frac{|A^c|}{|A|}D_{A^c}(x,y) \tag{2}$$

他们同时提出了混合数据信息系统下元素与聚类中心的相似性度量方法。假设 U 上的一个聚类结果为 $C=\{C_1,C_2,\cdots,C_k\}$，$Z=\{z_1,z_2,\cdots,z_k\}$为 k 个聚类中心，则元素 $x\in U$ 与 $z\in Z$ 的相似度为

$$D(x,z)=\frac{|A^r|}{|A|}\frac{D_{A^r}(x,z)}{\sum_{i=1}^{k}D_{A^r}(x,z_i)}+\frac{|A^c|}{|A|}\frac{D_{A^c}(x,z)}{\sum_{i=1}^{k}D_{A^c}(x,z_i)} \tag{3}$$

$z_i=(z_{i1},z_{i2},\cdots,z_{im})$为第 i 类的聚类中心，如果 $a_j\in A^r$，则

$$z_{ij}=\frac{1}{|c_l|}\sum_{x_i\in c_l}x_{ij} \tag{4}$$

如果 $a_j\in A^c$，则

$$z_{ij}=a_j^{(q)}\in V_{a_j} \tag{5}$$

其中$|\{w_{li}|x_{ij}=a_j^{(q)},w_{li}=1\}|\geqslant|\{w_{li}|x_{ij}=a_j^{(t)},w_{li}=1\}|$，$1\leqslant t\leqslant n_j,t\neq q$。$V_{a_j}=\{a_j^{(1)},a_j^{(2)},\cdots a_j^{(n_j)}\}$为 a_j 的值域，n_j 为 a_j 属性值的个数，如果 x_i 属于第 l 类，$w_{li}=1$，否则等于 0。

在式(1)、式(3)的基础上，提出了针对混合数据进行聚类的改进的 K-prototypes 算法，如表 1 所列。

表 1 改进的 K-prototypes 算法(算法 1)

1. 输入混合数据信息系统 MDT=(U,A,V,f)及聚类个数 k；
2. 随机选取 k 个聚类中心 $Z=\{z_1,z_2,\cdots,z_k\}$；
3. 利用式(3)计算 U 中每一个对象 x 与 Z 中每一个中心 z_i 的距离，并将 x 划分到与其最近的 z_i 所在的类中；
4. 根据式(4)和式(5)重新计算每一类的聚类中心；
5. 重复步骤 3 和步骤 4，直到每一类中的对象不再发生变化。

该算法较好地解决了混合数据的聚类问题，但是随着数据规模的增大，算法执行过程中迭代次数及算法所耗时间也会增加，因此该算法还有改进的空间。

2 基于抽样的大规模混合数据集成聚类算法

基于以上分析，本文提出了一种基于抽样的大规模混合数据集成聚类算法。首先针对 K-prototypes 算法在计算大规模数据时的不足，采取了抽样策略来降低算法的运算规模；通过多次抽样，在每次抽样的数据子集上利用改进的 K-prototypes 算法生成基聚类，再对基聚类结果进行集成，得到最终的聚类结果。

2.1 抽样

聚类集成的过程分为两部分[12]：首先对数据集 U 采用不同聚类方法或者同一聚类方法(选取不同参数)，或者对数据集中的不同子集进行聚类等策略生成基聚类；然后使用一致性函数对基聚类成员进行集成，生成最终的聚类结果。聚类集成流程如图 1 所示。

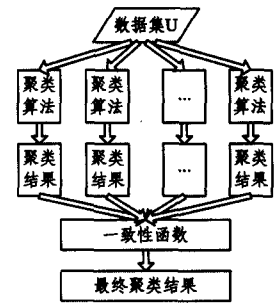

图 1 聚类集成算法流程

为了解决改进的 K-prototypes 聚类算法对大规模数据聚类时计算量过大的问题，本文在生成基聚类时采用了抽样方法。由于基聚类要有一定的差异度才能全面反映数据集的整体结构，同时为了避免由于部分基聚类结果较差而对集成结果造成不良影响的情况，各基聚类的结果之间要保持一定的关联。因此，在对数据集 U 进行抽样时，要保证多次抽取的样本子集之间有一定比例的相同样本。

基于以上分析，提出了一种新的抽样策略。首先，预先确定抽取样本子集的比例 r、各样本子集之间相同样本所占样本子集的比例 s 以及抽样的次数 h；然后从样本集 U 中随机抽取 $n_0(n_0=|U|\times r\times s)$个样本作为各样本子集的共同样本集 U_0；最后，在 $U-U_0$ 中不重复地抽取 h 个规模为$(\frac{1}{s}-1)n_0$ 的样本集，并分别与 U_0 构成样本子集 $U_1,U_2,\cdots,U_h$。使用以上策略可以得到既有一定的差异度又有一定比例的相同样本的样本子集。抽样算法如表 2 所列。

表 2 抽样算法(算法 2)

1. 输入样本集 U、抽样比例 r、样本子集中相同样本比例 s 及抽样次数 h；
2. 从数据集 U 中抽取 $n_0(n_0=|U|\times r\times s)$个样本，构成集合 U_0；
3. 令 $U'=U-U_0$；
4. for i=1 to h

   从 U'中抽取$(\frac{1}{s}-1)n_0$ 个样本，与 U_0 合并成为样本子集 U_i；

   $U'=U'-U_i$；
5. 得到样本子集 $U_1,U_2,\cdots,U_h$。

采用抽样算法得到样本子集 $U_1,U_2,\cdots,U_h$ 后，在每一个 U_i 上使用改进的 K-prototypes 算法进行聚类，同时得到聚类中心$\{z_1^i,z_2^i,\cdots,z_k^i\}(1\leqslant i\leqslant h)$，并利用聚类中心对全体数据集进行划分，得到基聚类结果。在得到基聚类结果后，利用一致性函数对结果进行集成。

2.2 一致性函数

针对大规模数据，为了提高集成效率，本文采用了投票的方法。投票法的基本思想是根据基聚类对数据集的多次划分，统计对象被划分到每一类的投票次数，将对象划分到投票次数最多的那一类中。

但在投票法中存在类标签对应的问题，即对于不同划分结果中的两个类，其类标签虽然不同，但逻辑上这两类是等价的。为了解决这个问题，Zhou 等[11]提出了一种类排列算法。该算法的基本思想是通过计算不同聚类结果中类与类之间重复对象的个数来度量类间相似度，相似度最大的两个类认为是等价的，等价的两个类选用相同的类标签。具体算法如表 3 所列。

表 3 类排列算法(算法 3)

1. 输入基聚类$\{C_1^a, C_2^a, \cdots, C_k^a\}, \{C_1^b, C_2^b, \cdots, C_k^b\}$；
2. 初始化矩阵 $M=(m_{ij})_{k\times k}, m_{ij}=0$；
3. for i=1 to k
   for j=1 to k
   $m_{ij}=\frac{|C_i^a \cap C_j^b|}{|C_i^a \cup C_j^b|}$；
4. for i=1 to k
   $(\mu,\upsilon)=\arg(\max(m_{ij}))$；
   将 C_υ^b 的类标签与 C_μ^a 的类标签统一；
   去掉 M 的第 μ 行、第 υ 列，生成新的 M；
5. 结束。

2.3 基于抽样的大规模混合数据集成聚类算法

基于以上分析，本文提出一种基于抽样的大规模混合数据集成聚类算法，具体算法如表 4 所列。

表 4 基于抽样的大规模混合数据集成聚类算法(算法 4)

1. 输入混合数据信息系统 MDT=(U,A,V,f)及聚类个数 k；
2. 调用算法 2 即抽样算法，得到 $U_1, U_2, \cdots, U_h$；
3. for i=1 to h
   利用算法 1 即改进的 K-prototypes 算法对数据集 U_i 进行聚类，找到 k 个聚类中心$\{z_1^i, z_2^i, \cdots, z_k^i\}$；
   计算数据集 U 中每一个对象 x_j 与聚类中心$\{z_1^i, z_2^i, \cdots, z_k^i\}$的距离，将 x_j 划分到与其最近的聚类中心所在的类中，得到一个聚类结果$\{C_1^i, C_2^i, \cdots, C_k^i\}$；
4. 利用算法 3 即类排列算法将 U 中每一个对象 x_j 的每一次聚类结果重新标记；
5. 利用投票法对聚类结果$\{C_1^i, C_2^i, \cdots, C_k^i\}$进行集成，得到最终的聚类结果$\{C_1, C_2, \cdots, C_k\}$。

3 实验分析

从 UCI 数据集中选取了不同规模的 3 个数据集，对本文提出的基于抽样的大规模混合数据集成聚类算法与改进的 K-prototypes 算法在所耗时间以及有效性方面进行了比较分析。

3.1 聚类有效性指标

为了评价算法的有效性，本文选取了两个聚类有效性指标：ARI(调整德兰指标)和 CUM。

(1)ARI 是一种外部评价指标，需要数据的真实划分结果，它用于评价聚类结果与实际类标签的相似程度。ARI 的表达式如下：

$$ARI=\frac{\sum_{ij} C_{n_{ij}}^2 - (\sum_i C_{b_i}^2 \sum_j C_{d_j}^2)/C_n^2}{(\sum_i C_{b_i}^2 + \sum_j C_{d_j}^2)/2 - (\sum_i C_{b_i}^2 \sum_j C_{d_j}^2)/C_n^2} \tag{6}$$

其中，数据集 U 的聚类结果为$\{C_1, C_2, \cdots, C_k\}$，$U$ 的真实划分为$\{P_1, P_2, \cdots, P_k\}$，$n_{ij}=|C_i \cap P_j|$，$b_i=|C_i|$，$d_j=|P_j|$，$n=|U|$。

如果 ARI 的值越高，说明该算法的聚类结果越接近真实划分。

(2)CUM 是一种针对混合数据聚类结果的内部评价指标，CUM 的表达式如下：

$$CUM=\frac{|A^r|}{|A|}CUN+\frac{|A^c|}{|A|}CUC \tag{7}$$

$$CUN=\frac{1}{k}\sum_{l=1}^{|A^r|}(\delta_l^2-\sum_j^k p_j\delta_{jl}^2)$$

$$CUC=\frac{1}{k}\sum_{a\in A^c} Q_a$$

$$Q_a=\sum_{X\in U/IND(\{a\})}\sum_{i=1}^{k}\frac{|C_i|}{|U|}\left(\frac{|X\cap C_i|^2}{|C_i|^2}-\frac{|X|^2}{|U|^2}\right)$$

其中

$$\delta_l^2=\sum_{x\in U}(f(x,a_l)-m_l)^2/|U|$$

$$\delta_{jl}^2=\sum_{x\in C_j}(f(x,a_l)-m_{jl})^2/|C_j|$$

$$m_l=\sum_{x\in U}f(x,a_l)/|U|$$

$$m_{jl}=\sum_{x\in C_j}f(x,a_l)/|C_j|$$

$$IND(a)=\{(x,y)\,|\,f(x,a)=f(y,a), x\in U, y\in U, a\in A^c\}$$

如果 CUM 的值越高，说明该算法的聚类结果越好。

3.2 实验及结果分析

为了验证在不同抽样比例的情况下算法的效率和有效性，设计了如下 4 个实验方案：

方案一：随机抽取总体样本的 1%作为公共样本子集 U_0，在剩余样本中不重复地随机抽取 3 组样本，每组占总体样本比例为 3%，分别与 U_0 构成样本子集 U_1, U_2, U_3，即 $r=4\%$，$s=25\%$；

方案二：随机抽取总体样本的 2%作为公共样本子集 U_0，在剩余样本中不重复随机抽取 3 组样本，每组占总体样本比例为 3%，分别与 U_0 构成样本子集 U_1, U_2, U_3，即 $r=5\%$，$s=40\%$；

方案三：随机抽取总体样本的 1%作为公共样本子集 U_0，在剩余样本中不重复地随机抽取 3 组样本，每组占总体样本比例为 4%，分别与 U_0 构成样本子集 U_1, U_2, U_3，即 $r=5\%$，$s=20\%$；

方案四：随机抽取总体样本的 2%作为公共样本子集 U_0，在剩余样本中不重复地随机抽取 3 组样本，每组占总体样本比例为 4%，分别与 U_0 构成样本子集 U_1, U_2, U_3，即 $r=6\%$，$s=33.3\%$。

同时从 UCI 数据集中选取了以下 3 个数据集：

(1)Adult 数据集，该数据集中共有 32561 个对象、14 个条件属性，其中 4 个数值型属性、10 个符号型属性，分为 2 类；

(2)Covertype 数据集，该数据集中包含 581012 个样本、14 个条件属性，其中 10 个数值型属性、4 个符号型属性，分为 7 类；

(3)Poker Hand 数据集，该数据集中包含 1000000 个样本、10 个条件属性，其中 5 个数值型属性、5 个符号型属性，分为 10 类。

在这 3 个数据集上分别采用上述 5 个方案进行实验，并与改进的 K-prototypes 算法就耗时以及有效性指标 ARI 和 CUM 进行比较。实验结果如表 5—表 7 所列。

表 5 Adult 数据集实验结果

算法	抽样集成算法				改进的
方案	一	二	三	四	K-prototypes
耗时(s)	2.508	2.794	2.814	2.827	12.813
ARI	0.412	0.412	0.435	0.407	0.410
CUM	0.211	0.206	0.223	0.208	0.190

表 6 Covertype 数据集实验结果

算法	抽样集成算法				改进的
方案	一	二	三	四	K-prototypes
耗时(s)	936.6	947.1	989.2	1013	2766
ARI	0.168	0.182	0.179	0.189	0.172
CUM	2.298	2.373	2.394	2.407	2.360

表 7 Poker Hand 数据集实验结果

算法	抽样集成算法				改进的
方案	一	二	三	四	K-prototypes
耗时(s)	1117	1283	1475	1733	6034
ARI	0.120	0.126	0.119	0.128	0.114
CUM	1.363	1.395	1.351	1.369	1.357

实验表明，与改进的 K-prototypes 算法相比，在有效性指标基本相同的情况下，算法所耗时间大大减少，证明该算法对大规模混合数据聚类是有效的。

结束语 本文针对 K-prototypes 算法以及改进的 K-prototypes 算法在对大规模混合数据进行聚类时时间效率不高的问题，提出了一种基于抽样的大规模混合数据集成聚类算法，并在 UCI 数据集中选取的 3 个不同规模的数据集上分别进行了实验。实验表明，该算法在时间效率和聚类精度上与原有算法相比有较大提升，证明该算法是有效的。

参 考 文 献

[1] MacQueen J B. Some methods for classification and analysis of multivariate observations [C] // Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability. Berkeley: University of California, 1967: 281-297

[2] Ruspini E R. A new Approach to clustering [J]. Information and Control, 1969, 15(1): 22-32

[3] Camastra F, Verri A. A novel kernel method for clustering [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005, 27(5): 801-805

[4] Zhang T, Ramakrishnan R, Livny M. BIRCH [C] // Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data. Quebec: ACM, 1996: 103-114

[5] Guha S, Rastogi R, Shim K. CURE: An efficient clustering algorithm for clustering large databases [C] // Proceedings of the Symposium on Management of Data (SIGMOD). Seattle: ACM, 1998: 73-84

[6] Ester M, Kriegel H P, Sander J, et al. A density-based algorithm for discovering clusters inlarge spatial databases with noise [C] // Proceedings of the 2th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. USA: AAAI, 1996: 226-231

[7] Huang Zhe-xue. Extensions to the k-means algorithm for clustering large data sets with categorical values [J]. Data Mining and Knowledge Discovery, 1998, 2(3): 283-304

[8] Liang Ji-ye, Zhao Xing-wang, Li De-yu, et al. Determining the number of clusters using information entropy for mixed data [J]. Pattern Recognition, 2012, 45(6): 2251-2265

[9] He Zeng-you, Xu Xiao-fei, Deng Sheng-chun. Clustering Mixed Numeric and Categorical Data: A Cluster Ensemble Approach [J]. Computer Science Artificial Intelligence, 2005, 5(4): 225-268

[10] Luo Hui-lan, Wei Hui. Clustering Algorithm for Mixed Data Based on Clustering Ensemble Technique [J]. Computer Science, 2010, 37(11): 234-238(in Chinese)
罗慧兰，危辉. 一种基于聚类集成技术的混合型数据聚类方法 [J]. 计算机科学，2010，37(11)：234-238

[11] Zhou Zhi-hua, Tang Wei. Clusterer ensemble [J]. Knowledge-Based Systems, 2006, 19(1): 77-83

[12] Yang Cao-yuan, Liu Da-you, Yang Bo, et al. Research on Cluster Aggregation Approaches [J]. Computer Science, 2011, 38(2): 166-170(in Chinese)
杨草原，刘大有，杨博，等. 聚类集成方法研究 [J]. 计算机科学，2011，38(2)：166-170

(上接第 202 页)

杜方，陈跃国，杜小勇. RDF 数据查询处理技术综述 [J]. 软件学报，2013，24(6)：1222-1242

[2] Auer S, Bizer C, Kobilarov G, et al. Dbpedia: A nucleus for a web of open data [M]. Springer Berlin Heidelberg, 2007: 722-735

[3] Apweiler R, Bairoch A, Wu C H, et al. UniProt: the universal protein knowledgebase [J]. Nucleic Acids Research, 2004, 32 (suppl 1): D115-D119

[4] Stadler C, Lehmann J, Höffner K, et al. Linkedgeodata: A core for a web of spatial open data [J]. Semantic Web, 2012, 3(4): 333-354

[5] Goodman E L, Jimenez E, Mizell D, et al. High-performance computing applied to semantic databases [M] // The Semanic Web: Research and Applications. Springer Berlin Heidelberg, 2011: 31-45

[6] Long Cheng, Malik A, Kotoulas S, et al. Efficient parallel dictionary encoding for RDF data [C] // Proceedings of the 17th International Workshop on the Web and Databases (WebDB). 2014

[7] Long Cheng, Malik A, Kotoulas S, et al. Scalable RDF Data Compression using X10 [J/OL]. http://rian.ie/ga/item/viem/109810.html

[8] Urbani J, Maassen J, Drost N, et al. Scalable RDF data compression with MapReduce [J]. Concurrency and Computation: Practice and Experience, 2013, 25(1): 24-39

[9] Wu Bu-wen, Jin Hai, Yuan Ping-peng. Scalable SAPRQL querying processing on large RDF data in cloud computing environment [M] // Pervasive Computing and the Networked World. Springer Berlin Heidelberg, 2013: 631-646

[10] Liu Liu, Yin Jiang-tao, Gao Li-xin. Efficient social network data query processing on mapreduce [C] // Proceedings of the 5th ACM workshop on HotPlanet. ACM, 2013: 27-32

[11] Lee D, Kim J S, Maeng S. Large-scale incremental processing with MapReduce [J]. Future Generation Computer Systems, 2014, 36: 66-79

[12] Thomas H, Cormen Charles E, Leiserson Ronald L, et al. Introduction to Algorithms(第 3 版) [M]. 殷建平，徐云，等译. 北京：机械工业出版社，2013：593-599

[13] Guo Yuan-bo, Pan Zheng-xiang, Heflin J. LUBM: A benchmark for OWL knowledge base systems [J]. Web Semantics: Science, Services and Agents on the World Wide Web, 2005, 3(2): 158-182