

Fig.11 The process of general union

图 11 General union 过程

算法 4 是一般的局部聚类合并<sup>[17]</sup>算法,同时作为算法 5 的基准实验,在实验结果图中也可以看出两种算法的性能差异.观察算法 4 可以发现:Map 阶段每次都输出了核心节点及当前聚类的所有元素,导致了 IO 次数<sup>[18]</sup>快速增长,可以考虑使用最小编号的核心节点作为当前局部聚类的标记,从而对算法 4 进行优化.具体如下.

首先,Map 阶段,我们在局部聚类中找出是核心的节点,并且标注最小的核心,将所有核心分别作为 key,最小核心作为 value 构造成键值对输出.同时,以最小核心为 key,将当前局部聚类集合  $C_u$  作为 value 构造成键值对输出,MapReduce 框架会将同一个 key 的键值对分到同一个 reduce;然后,在 Reduce 阶段对相同节点的不同聚类进行合并;最后,查看迭代条件,是否需要迭代,得到最终聚类.

基于上述步骤可以看出:在两个分别以  $u, v$  标记的局部聚类里,假如局部聚类  $C_u, C_v$  可以合并,  $C_u$  中必存在一点  $v_{min}, v_{min}$  也在  $C_v$  中,利用  $v_{min}$  可以将两个聚类联系在一起,合并过程见算法 5,图示过程如图 12 所示,迭代条件与算法 4 相同.

**算法 5. Base Min\_Core Union.**

输入:维度扩展后的顶点  $u$  及其直接可达邻居的初级聚类  $C_u$ ,已知  $Neighbor[u]$ ;

输出:聚类.

1.  $Map(u, C_u)$
2.  $v_{min} \leftarrow v | v \in C_u \wedge v.isCore is True$
3. **while**  $v \in Neighbor[u]$  **do**
4.     **if**  $v.isCore is True$  **then**
5.          $Emit(v, v_{min})pairs$
6.          $Emit(v_{min}, C_u)pairs$
7. **Reduce**( $u, \{C_1 \dots C_n\}$ )
8.  $C \leftarrow Merge(C_1 \dots C_n)$
9.  $Emit(u, C)$

算法 4、算法 5 中的  $Merge(C_1 \dots C_n)$  函数是合并可以合并的两个局部聚类函数,本质上是一个集合的并<sup>[19]</sup>操作,集合元素类型是点及其核心属性构成的元组.算法 5 中,  $Neighbor[u]$  数组表示  $u$  的可达邻居节点.合并函数的具体过程见算法 6.

**算法 6.** Merge SubCluster.

输入: 维度扩展后的顶点  $u$  及其直接可达邻居的初级聚类  $C_u$ , 已知  $Neighbor[u]$ ;

输出: 聚类  $C$ .

1. **Merge**( $C_1, \dots, C_n$ )
2.     **for**  $i=0 \rightarrow n$  **do**
3.         **for**  $c$  **in**  $C_i$
4.             **if**  $C$  contains  $c$  **do**
5.                 continue
6.             **else**
7.                  $C.add(c)$
8.     **return**  $C$

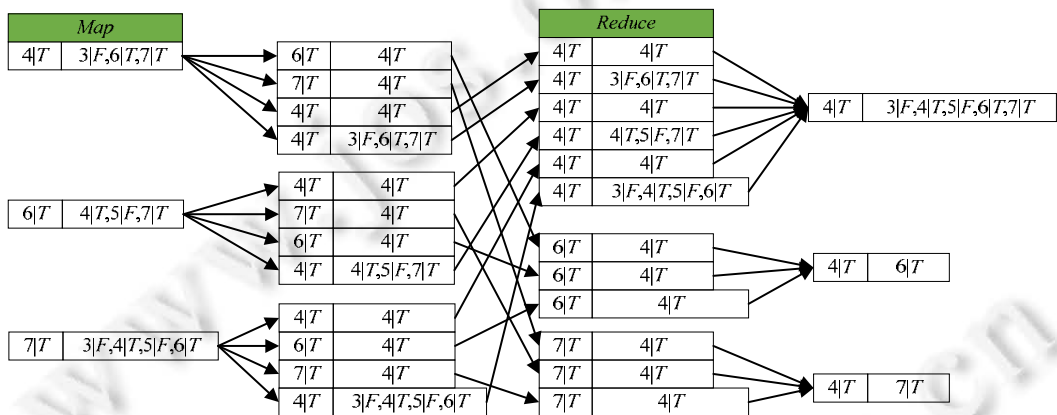


Fig.12 Base min\_coremerge subclusters

图 12 最小核局部聚类

**4 实验与结果**

在已搭建的 Hadoop 分布式集群上,使用本文提出的算法构建了一种面向无向图<sup>[20]</sup>的结构聚类模型,并且在 4 个真实的数据集上(Skitter,Pokec,LiveJournal 和 Com-Orkut)上测试本文提出的方法.

**4.1 实验平台与实验数据集**

实验环境为 8 台服务器构成的 Hadoop 集群,1 个 master,7 个 slaver,每台服务器有两个 CPU,CPU 配置参数为: Intel(R) Xeon(R) CPU E5-2620 v3@2.40GHz,每个 CPU 有 6 个核,共 12 个核,因此,集群一共有 96 个核.实验过程中,我们可以根据集群核数调整 Reduce 个数.单台服务器有 32GB 的内存,统一安装 Linux ubuntu16.04 64 位操作系统.

实验数据集来源于斯坦福大学的公开数据集.数据的详细信息见表 1,从 4 个数据集的边数可以看出,实验数据具有一定的梯度性<sup>[21]</sup>.

**Table 1** Experimental data set and its vertex number and edge number

**表 1** 实验数据集及其顶点数和边数

DataSet	Number of nodes	Number of edges
Skitter	1 696 415	11 095 298
Pokec	1 632 000	22 301 000
LiveJournal	3 997 962	34 681 189
Com-Orkut	3 072 441	117 185 083

### 4.2 实验结果及分析

实验采用 4 个数据集,相似性  $\epsilon$  和  $\epsilon$  邻居数  $\mu$  这两个参数分别取(0.4,3)和(0.6,6),实验收集得到 8 个实验结果图如图 13 所示.

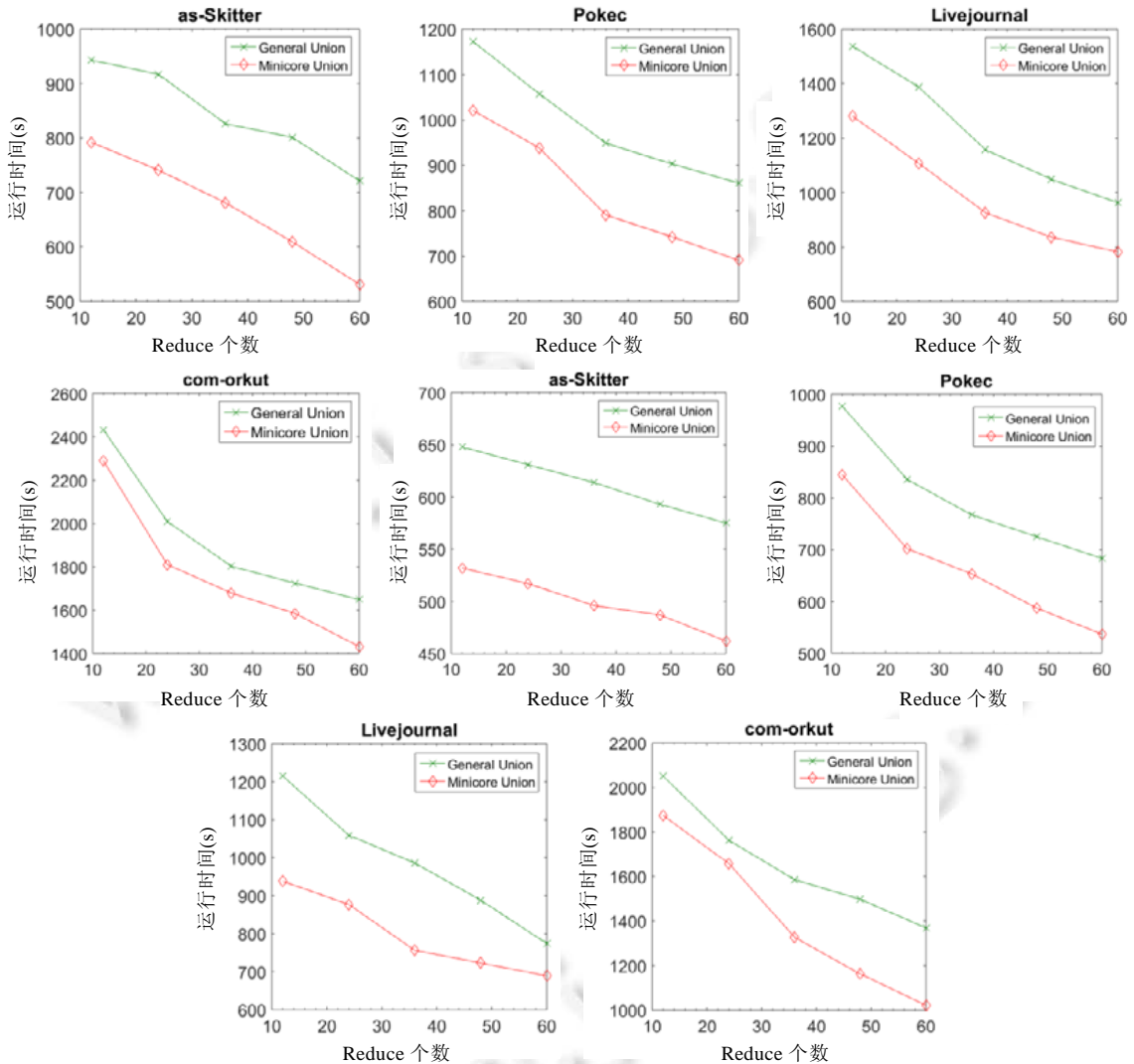


Fig.13 Efficiency comparison varying number of reduce

图 13 不同 reduce 数量运行效率对比

在图 13 的实验结果中:前 4 个结果图采用了统一的  $\epsilon$  和  $\mu$ ,取值(0.6,6),后 4 个结果图取值(0.4,3),并且让 reduce 个数线性增加(12,24,36,48,60)来观察实验结果.这里不需要设置 Map 个数,Map 个数是 Hadoop 集群根据数据集大小自动进行分配.

- 观察 1.单独观察图 13 每个实验结果可以发现:当数据集不变时,reduce 个数增加,算法的运行时间大致呈线性降低;
- 观察 2.根据实验结果可以发现,算法 5(Base Min\_Core Union)比算法 4(General Union)的合并效率高;
- 观察 3.当 reduce 个数一样时,数据集的边数越大,算法时间越长;

- 观察  $4.\varepsilon$ 和 $\mu$ 的取值越大,剪枝效率越高.

## 5 总 结

为了解决图结构聚类算法的可扩展性问题,本文提出一种基于 MapReduce 的结构聚类算法 MRSCAN.具体地,本文设计了一套计算核心节点以及两种有效的合并聚类的 MapReduce 算法.最后,在多个真实的大规模图数据集上进行测试,实验结果验证了本文算法的正确性、有效性以及可扩展能力.未来的工作包括:将我们实现的算法推广至其他的分布式计算框架,例如 Pregel<sup>[22]</sup>计算框架.

### References:

- [1] Mayer-Schoenberger V, Cukier K, Wrote; Zhou T, *et al.* Trans. Big Data: A Revolution That Will Transform How We Live, Work, and Think. John Murray, 2013 (in Chinese).
- [2] Shvachko K, Kuang H, Radia S, Chansler R. The hadoop distributed file system. In: Proc. of the IEEE Symp. on MASS Storage Systems and Technologies. IEEE Computer Society, 2010. 1–10. [doi: 10.1109/MSST.2010.5496972]
- [3] Shiokawa H, Fujiwara Y, Onizuka M. SCAN++: Efficient algorithm for finding clusters, hubs and outliers on large-scale graphs. Proc. of the VLDB Endowment, 2015. [doi: 10.14778/2809974.2809980]
- [4] Chang L, Li W, Qin L, Zhang W. pSCAN: Fast and exact structural graph clustering. IEEE Trans. on Knowledge & Data Engineering, 2017,29(2):387–401. [doi: 10.1109/TKDE.2016.2618795]
- [5] Li JJ, Cui J, Wang D, Yan L, Huang YS. Survey of MapReduce parallel programming model. Acta Electronic Journal, 2011,39(11):2635–2642 (in Chinese with English abstract).
- [6] Wang F, Lei BH. Model analysis of hadoop distributed file system. Telecommunications Science, 2010,26(12):95–99 (in Chinese with English abstract). [doi: 10.3969/j.issn.1000-0801.2010.12.019]
- [7] Chen F, Kodialam M, Lakshman TV. Joint scheduling of processing and Shuffle phases in MapReduce systems. In: Proc. of the IEEE INFOCOM. IEEE, 2012. 1143–1151. [doi: 10.1109/INFOCOM.2012.6195473]
- [8] Xu X, Yuruk N, Feng Z, Schweiger TAJ. SCAN: A structural clustering algorithm for networks. In: Proc. of the ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. ACM Press, 2007. 824–833. [doi: 10.1145/1281192.1281280]
- [9] Zhou FF, Li JC, Huang W, Wang JW, Zhao Y. Based on dimension expansion Radviz visual clustering analysis method. Ruan Jian Xue Bao/ Journal of Software, 2016,27(5):1127–1139 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/4951.htm> [doi: 10.13328/j.cnki.jos.004951]
- [10] Guo QK. Study on connection method based on MapReduce [Ph.D. Thesis]. Jilin University, 2014 (in Chinese with English abstract).
- [11] Cohen J. Graph twiddling in a MapReduce world. Computing in Science & Engineering, 2009,11(4):29–41. [doi: 10.1109/MCSE.2009.120]
- [12] Wang HY, Zhou L. Study on the maximum mission problem based on divide, prune and ant colony algorithm. Journal of Hefei Teachers College, 2011,29(3):59–62 (in Chinese with English abstract). [doi: 10.3969/j.issn.1674-2273.2011.03.020]
- [13] Qin L, Yu J X, Chang L, Cheng H, Zhang C, Lin XM. Scalable big graph processing in MapReduce. In: Proc. of the SIGMOD. 2014. 827–838. [doi: 10.1145/2588555.2593661]
- [14] Feng XJ. Research on iterative distributed data processing based on MapReduce [Ph.D. Thesis]. Shandong University, 2013 (in Chinese with English abstract).
- [15] Liu T, Wu SH, Qiang Y. Task scheduling algorithm for multiple MapReduce jobs. Microelectronics & Computer, 2013,30(12): 156–159 (in Chinese with English abstract).
- [16] Gao SL. Research on Web page parallel de-emphasis algorithm based on MapReduce framework. Heilongjiang Science, 2010,1(5): 13–18 (in Chinese with English abstract).
- [17] Que X, Wang Y, Xu C, Yu WK. Hierarchical merge for scalable MapReduce. In: Proc. of the Workshop on Management of Big Data Systems. 2012. 1–6. [doi: 10.1145/2378356.2378358]

- [18] Tiwari N, Sarkar S, Indrawan-Santiago M, Bellur U. Improving energy efficiency of IO-intensive MapReduce jobs. In: Proc. of the Int'l Conf. 2015. 1–4. [doi: 10.1145/2684464.2684484]
- [19] Zhang X. Design of DBSCAN algorithm based on query and search. Journal of Yili Normal University: Natural Science Edition, 2014,8(4):62–65 (in Chinese with English abstract). [doi: 10.3969/j.issn.1673-999X.2014.04.014]
- [20] Ma J, Iwama K, Ma SH. Search for parallel algorithm in loop of undirected graph. Ruan Jian Xue Bao/Journal of Software, 1997, 18(6):475–480 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/18/475.htm>
- [21] Hou ZL, Wei XH, Huang DN, Xu S. Application of parallel computing and its performance analysis in gravity totally gradient data inversion. Applied Geophysics, 2015,12(3):292–302 (in Chinese with English abstract).
- [22] Malewicz G, Austern MH, Bik AJC, Dehnert JC, Horn I, Leiser N, Czajkowski G. Pregel: A system for large-scale graph processing. In: Proc. of the ACM SIGMOD Int'l Conf. on Management of Data. ACM Press, 2010. 135–146. [doi: 10.1145/1582716.1582723]

#### 附中文参考文献:

- [1] Mayer-Schonberger V, Cukier K, 著;周涛,等,译.大数据时代:生活、工作与思维的大变革.杭州:浙江人民出版社,2013.
- [5] 李建江,崔健,王聃,严林,黄义双.MapReduce 并行编程模型研究综述.电子学报,2011,39(11):2635–2642.
- [6] 王峰,雷葆华.Hadoop 分布式文件系统的模型分析.电信科学,2010,26(12):95–99. [doi: 10.3969/j.issn.1000-0801.2010.12.019]
- [9] 周芳芳,李俊材,黄伟,王俊韡,赵颖.基于维度扩展的 Radviz 可视化聚类分析方法.软件学报,2016,27(5):1127–1139. <http://www.jos.org.cn/1000-9825/4951.htm>
- [10] 郭祺恺.基于 MapReduce 的连接方法研究[博士学位论文].长春:吉林大学,2014.
- [12] 王会颖,周琳.基于分治、剪枝和蚁群算法求解最大团问题.合肥师范学院学报,2011,29(3):59–62. [doi: 10.3969/j.issn.1674-2273.2011.03.020]
- [14] 冯新建.基于 MapReduce 的迭代型分布式数据处理研究[博士学位论文].济南:山东大学,2013.
- [15] 刘涛,武淑红,强彦.用于多个 MapReduce 作业的任务调度算法.微电子学与计算机,2013,30(12):156–159.
- [16] 高殊丽.基于 MapReduce 框架的网页并行去重算法研究.黑龙江科学,2010,1(5):13–18.
- [19] 张晓.基于并查集的 DBSCAN 算法设计.伊犁师范学院学报:自然科学版,2014,8(4):62–65. [doi: 10.3969/j.issn.1673-999X.2014.04.014]
- [20] 马军,岩间一雄,马绍汉.寻找无向图中回路的并行算法.软件学报,1997,18(6):475–480. <http://www.jos.org.cn/1000-9825/18/475.htm>



张伟鹏(1991—),男,广东汕头人,硕士生,主要研究领域为大规模图数据并行算法。



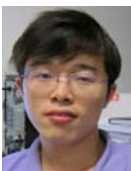
刘宇鸿(1997—),男,学士,主要研究领域为社区搜索,时态图挖掘。



李振军(1979—),男,博士,工程师,主要研究领域为数据挖掘,深度学习。



毛睿(1975—),男,博士,教授,CCF 高级会员,主要研究领域为数据挖掘,数据库,统计方法,机器学习,计算生物。



李荣华(1985—),男,博士,讲师,主要研究领域为图数据挖掘,社交网络分析。



乔少杰(1981—),男,博士,教授,CCF 高级会员,主要研究领域为轨迹数据挖掘,机器学习。