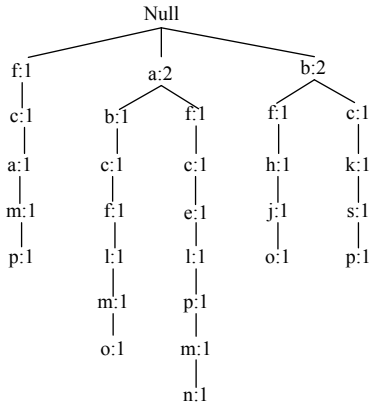

图 6 将排序后的路径插入 T 中

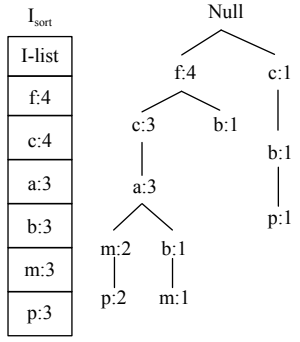

图 7 重构后的最终树

表 6 数据库特征

Database	Items	Rcords	Max|T|	Avg|T|
mushroom	119	8124	23	23
T10I4D100K	870	100 000	29	10

在相同条件下, 算法独立实验 10 次, 取其运行时间的均值作为算法最终结果. 两个数据集在不同算法和不同最小支持度阈值下的运行时间对比如表 7 和表 8 所示, 单位为秒 (即, s). 从表中可以看出, 本文改进算法在运行时间上明显优于其他几个算法, 说明本文改进算法有效提高了频繁模式的挖掘速度. 从图 8 和图 9 中可以看出本文改进算法比原始 APFT 算法更高效, 主要原因有两个: 1) 在算法的 apriori_gen() 函数前加入了连接预处理步骤, 对进入连接步的频繁 k-项集进行剪枝, 减少了两两频繁项集前 (k–1) 项的比较次数, 因此减小了连接步的时间消耗, 并且候选项集的数量也减少了, 减小了剪枝步的时间开销; 2) 新提出的树构只需对数据库进行一次扫描就可以构造出一颗紧凑的模式树, 虽然增加了树重构过程但这一过程基本上不会花费太多时间, 并且在挖掘过程中不需要额外的空间, 因此本文改进算法具有更好的空间可扩展性.

表 7 mushroom 上不同算法运行时间对比

最小支持度阈值 (%)	4	5	6	7	8	9
Apriori 算法	675	300	130	90	75	45
FP-Growth 算法	110	80	65	40	35	20
APFT 算法	90	75	50	30	15	9
本文改进算法	**60**	**53**	**33**	**24**	**10**	**5**

表 8 T10I4D100K 上不同算法运行时间对比

最小支持度阈值 (%)	1	2	3	4	5	6
Apriori 算法	6.8	4.3	3.4	3.1	2.9	2.7
FP-Growth 算法	5.7	3.5	2.9	2.7	2.5	2.2
APFT 算法	4.8	3.1	2.9	2.8	2.4	2.2
本文改进算法	**3.8**	**2.7**	**2.3**	**1.9**	**1.7**	**1.3**

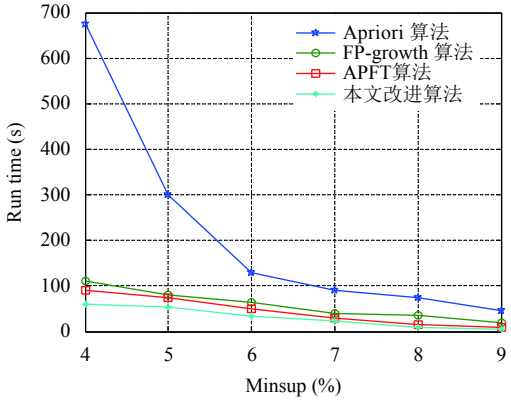

图 8 mushroom 上运行时间对比

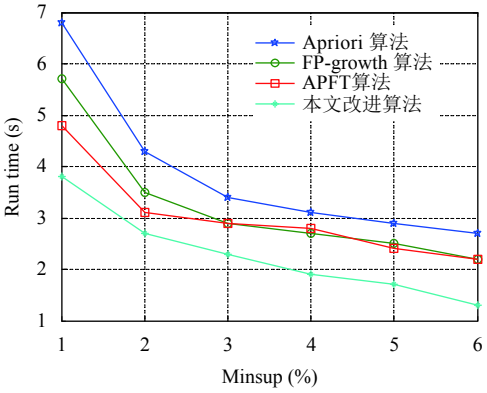

图 9 T10I4D100K 上运行时间对比

4 结束语

本文针对传统频繁模式挖掘算法存在的固有缺陷, 提出了一种基于 APFT 算法的改进频繁模式挖掘算法. 首先, 在算法的连接步前加入预处理过程, 对参与连接的频繁 k-项集进行有效剪枝, 大大减少了连接步与剪枝步的时间开销; 其次, 对 CP-tree 进行扩展提出了一

种新的树结构 ECP-tree, 新的树结构是一棵紧凑的前缀模式树; 然后, 再将改进点与 APFT 算法结合用于频繁模式挖掘; 最后, 利用实验验证了改进算法的有效性.

为了有效对频繁模式进行挖掘, 本文将改进点与 APFT 算法结合, 由于新提出的连接预处理方法与紧凑树结构具有较好的可移植性, 因此改进点可与其它类似算法结合, 且并不影响挖掘效率, 相应地还能增强原始算法处理数据流问题的能力.

下一步的研究工作包括: 1) 继续完善算法, 往并行计算方向扩展; 2) 将该算法的思想扩展到最大、闭频繁模式的挖掘领域.

参考文献

1 Han JW, Kamber M, Pei J. 数据挖掘: 概念与技术. 范明, 孟小峰, 译. 3 版. 北京: 机械工业出版社, 2012.

2 Agrawal R, Imieliński T, Swami A. Mining association rules between sets of items in large databases. ACM SIGMOD Record, 1993, 22(2): 207–216. [doi: 10.1145/170036]

3 刘步中. 基于频繁项集挖掘算法的改进与研究. 计算机应用研究, 2012, 29(2): 475–477. [doi: 10.3969/j.issn.1001-3695.2012.02.019]

4 邢长征, 安维国, 王星. 垂直数据格式挖掘频繁项集算法的改进. 计算机工程与科学, 2017, 39(7): 1365–1370. [doi: 10.3969/j.issn.1007-130X.2017.07.025]

5 于守健, 周羿阳. 基于前缀项集的 Apriori 算法改进. 计算机应用与软件, 2017, 34(2): 290–294. [doi: 10.3969/j.issn.1000-386x.2017.02.052]

6 王蒙, 邹书蓉, 方睿. 一种基于矩阵的 Apriori 改进算法. 信息技术, 2018, (3): 150–154.

7 Han JW, Pei J, Yin YW. Mining frequent patterns without candidate generation. Proceedings of 2000 ACM SIGMOD International Conference on Management of Data. Dallas, Texas, USA. 2000. 1–12.

8 肖继海, 崔晓红, 陈俊杰. 基于 COFI-Tree 的 N-最有兴趣项目集挖掘算法. 计算机技术与发展, 2012, 22(3): 99–102. [doi: 10.3969/j.issn.1673-629X.2012.03.026]

9 Yin M, Wang WJ, Liu Y, *et al*. An improvement of FP-Growth association rule mining algorithm based on adjacency table. MATEC Web of Conferences, 2018, 189(1): 10012.

10 Lan QH, Zhang DF, Wu B. A new algorithm for frequent itemsets mining based on apriori and FP-tree. Proceedings of 2009 WRI Global Congress on Intelligent Systems. Xiamen, China. 2009. 360–364.

11 吴倩. 基于压缩 FP-tree 的频繁项集快速挖掘算法研究[硕士学位论文]. 上海: 华东理工大学, 2015.

12 张宁. 基于 FP-tree 的 Apriori 算法的改进. 信息通信, 2015, (2): 94–95. [doi: 10.3969/j.issn.1673-1131.2015.02.056]

13 倪政君, 夏哲雷. 一种基于 fp-tree 的 Apriori 算法改进研究. 中国计量大学学报, 2018, 29(1): 50–54. [doi: 10.3969/j.issn.2096-2835.2018.01.009]

14 吴磊, 程良伦, 王涛. 基于事务映射区间求交的高效频繁模式挖掘算法. 计算机应用研究, 2019, 36(4). [doi: 10.19734/j.issn.1001-3695.2017.10.0972.]

15 Tanbeer SK, Ahmed CF, Jeong BS, *et al*. Efficient single-pass frequent pattern mining using a prefix-tree. Information Sciences, 2009, 179(5): 559–583. [doi: 10.1016/j.ins.2008.10.027]