

研究对象

频繁项集 Frequent Itemset → e.g. 特征 X 和 Y 经常同时出现

Association Rule → e.g. 有特征 X 推出有特征 Y

概念

Support: probability or count that a transaction contains $X \cup Y$

Confidence: probability or count that transaction contains X also contains Y

Ch5 的思路:

- 从 Frequent Itemset 出发，有了 Frequent Itemset 可以简单地得到 Association (Itemset 又叫 Pattern)

→ 一条数据有多个特征，结合起来可能的 Itemset 总数很多！

Closed Itemset → 把所有子集包含关系且 support 相等的 Itemset 合并

Max Itemset → 把所有子集包含关系的 Itemset 合并

Frequent Itemset Mining Algorithms

- 1' Apriori → 扫描多次遍历，Horizontal，Hash Candidates 建立
- 2' FP-Growth → 只需两次遍历，Horizontal，递归求解
- 3' ECLAT → Vertical
- 3.5 DifferSet → ECLAT 的兄弟算法
- 4' CLOSET → 基础得到 Closed Frequent Itemset.

下面来体味看 4 种算法。

Apriori :

算法流程：

Root (L_0)

↓

Gen $L_{k+1} - C_k$

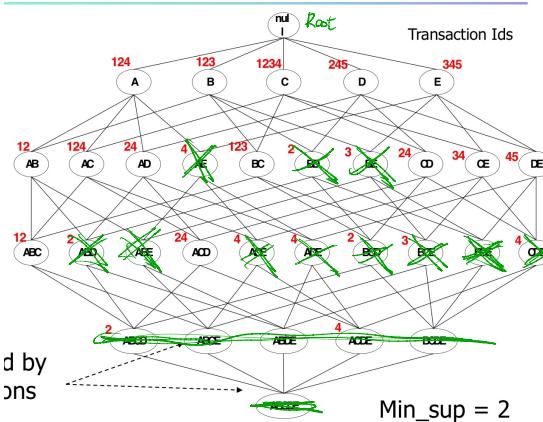
↓

Count $C_k \rightarrow L_k$

TID	Items
1	ABC
2	ABCD
3	BCE
4	ACDE
5	DE

Min-support = 2

⇒



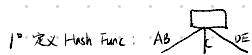
⇒ 绿色部分是 Apriori 中被剪枝掉的部分子树

Apriori 加法:

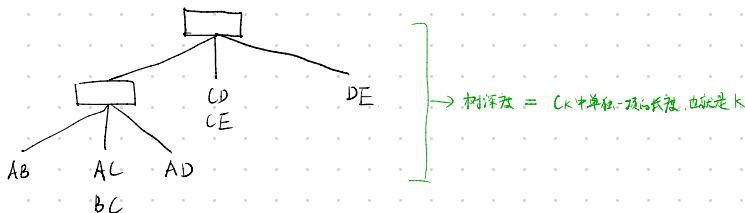
朴素的 Apriori 算法每次 Count C_k 都会需要一次数据集遍历。



Hashed Candidates 方法:



2) 对 C_K 进行 Hash, 以上图 C₂ 为例

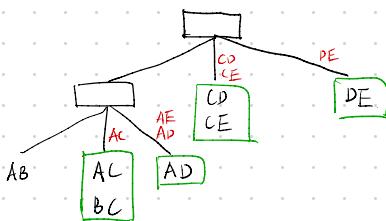


3) 遍历数据集，对于单独的一条数据 t，得到所有 K 长 Itemset

进行计数即可。

比如以 TID=4 的 ACDE 为例，组合共 4 项为 2 位 Itemset 有：AC AD AE CD CE DE

把这些 Itemset 再 hash：



原来这 6 个 Itemset 各一个都跟和 C₂ 中所有 Candidate 进行比较

现在只有绿框部分需要比较

FP-Growth

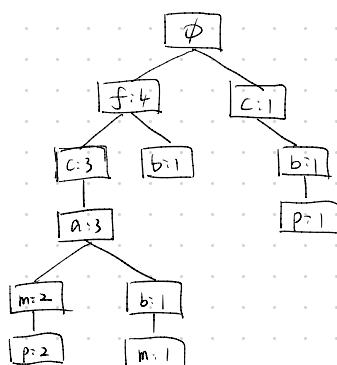
算法流程：

- 1' 得到 F-list
- 2' 根据 F-list 对数据再生成
- 3' 构造 FP-Tree
- 4' 递归求解.

例子：

<u>TID</u>	<u>Items bought (or)</u>	<u>F-list:</u>	<u>(ed) frequent items</u>
100	{f, a, c, d, g, i, m, p}	f: 4	{f, c, a, m, p}
200	{a, b, c, f, l, m, o}	c: 4	{f, c, a, b, m}
300	{b, f, h, j, o, w}	a: 3	{f, b}
400	{b, c, k, s, p}	b: 3	{c, b, p}
500	{a, f, c, e, l, p, m, n}	m: 3 p: 3	{f, c, a, m, p}

构造 FP-Tree



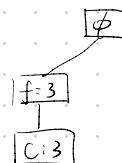
递归求解

def solve_fp (FP-Tree: T)

i 从后向前遍历 F-list.

if $i = a$

此时得到 T 中的子树 Ta



对 T_i 进行：

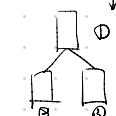
1° TEST

2° 若为单键 → 组合

若非单键 → 分治 + 组合

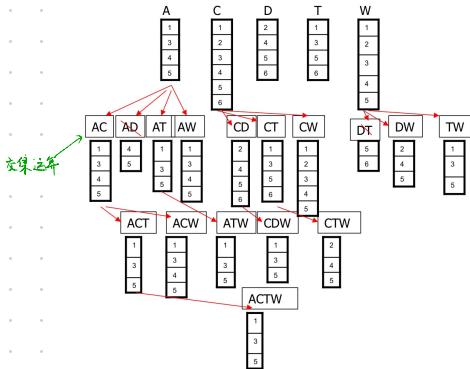
比 i 大上面 $i = a$

frequent item set: fa, ca, cfa



$\emptyset + (\emptyset \cup \emptyset) + i$

ECLAT



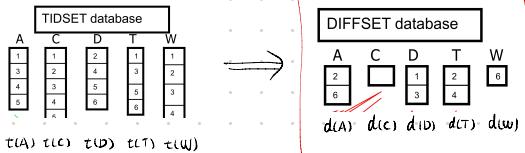
Diffset

思想和ECLAT基本一样，ECLAT算法会遇到的问题是，求交的时候

因而想到求 $\frac{A}{\cap}$ 而非



太长，求交麻烦



$$d(XY) = d(Y) - d(X)$$

$$t(XY) = t(X) - d(XY)$$

$$d(XYZ) = d(Z) - d(Y) - d(X)$$

$$t(XYZ) = t(X) - t(XY) - t(XYZ)$$

CLOSESET

1° 生成 Flist

2° 生成 TDB，由小到大生成 TDB[i]

3° 汇总 frequent closed itemset

例：

Min_sup=2

TID	Items
10	a, c, d, e, f
20	a, b, e
30	c, e, f
40	a, c, d, f
50	c, e, f

Flist:
 $d-a-f-e-c \Rightarrow$

TDB
 cefad
 ea
 cef
 cfad
 cef

\Rightarrow

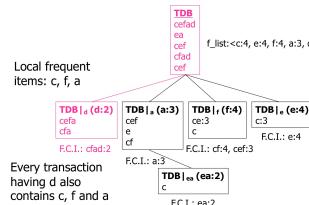
一个一个 FCI 地找

先找含 d 的 FCI

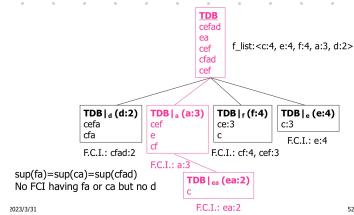
再找含 a 不含 d 的 FCI

不含 a, 不含 d 的 FCI

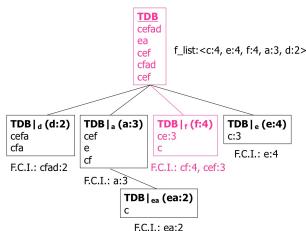
含 d 的 FCI



不含 d 的 FCI



不含 a, 不含 d 的 FCI



Association Rule.

有了频繁项集之后，Association Rule 就会很简单

比如得到频繁项集 AB: 3 A: 4 B: 5 数据总量 5

$$\text{支持度 } \text{support}(AB) = \frac{3}{5}$$

$$\text{置信度 } \text{confidence}(A \rightarrow B) = \frac{3}{4}$$