第七章 关联规则挖掘

7.1 关联规则挖掘基本概念

7.2 频繁集挖掘算法：Apriori

7.3 规则生成

超市购物车挖掘

通过对超市交易记录挖掘得到以下关联规则(Association Rule)：

{Diaper} → {Beer} [支持度support =2%, 置信度 confidence =60%]

表示所有交易记录的2%中尿布和啤酒被同时购买，购买尿布的记录中有60%也购买了啤酒。

注意：以上箭头表示“共同出现”的关系，而不是因果关系!

关联规则有什么用？

根据某些商品的出现来预测其他商品的出现，应用：

- 进行货架摆放优化，让用户更快找到经常一起购买的商品；
- 捆绑营销技巧。比如对尿布打折但提高啤酒的价格。

相关定义

关联规则

- 表示为 X → Y的对商品集之间的关联性描述，这里 X 和 Y 都是商品集或项集(itemsets)。若X包含k个item, 则称为k项集(k-itemset)。
- 例如: {Milk，Diaper} → {Beer}

规则评估度量

- 支持度Support (s)
  - 同时包含 X和 Y的交易占总交易数的百分比。

  Support(X->Y)=P(X,Y)
- 置信度Confidence (c)
  - 包含 X的交易中有多少也包含了Y，即同时包含X和Y的交易数占所有包含X的交易的百分比。

  Confidence(X->Y)=P(Y|X)

TID	*Items*
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

例如:

$$\{\text{Milk}, \text{Diaper}\} \Rightarrow \text{Beer}$$

$$s = \frac{\sigma(\text{Milk}, \text{Diaper}, \text{Beer})}{|\text{T}|} = \frac{2}{5} = 0.4$$

$$c = \frac{\sigma(\text{Milk}, \text{Diaper}, \text{Beer})}{\sigma(\text{Milk}, \text{Diaper})} = \frac{2}{3} = 0.67$$

$\sigma()$表示一个项集出现的次数, |T|表示交易记录的总数

关联规则挖掘任务

给定一个交易记录集合T，关联规则挖掘的目标是找出所有符合以下条件的规则：

- support ≥ MinSup （支持度阈值）
- confidence ≥ MinConf （置信度阈值）

同时满足以上两个条件，怎么找？

穷举所有可能规则，然后计算每条规则的support和confidence?

规则空间，即可能的规则数目太大，计算量太大！

TID	Items
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

例子:

{Milk,Diaper} → {Beer} (s=0.4, c=0.67)
{Milk,Beer} → {Diaper} (s=0.4, c=1.0)
{Diaper,Beer} → {Milk} (s=0.4, c=0.67)
{Beer} → {Milk,Diaper} (s=0.4, c=0.67)
{Diaper} → {Milk,Beer} (s=0.4, c=0.5)
{Milk} → {Diaper,Beer} (s=0.4, c=0.5)

发现:

- 以上所有规则都是从同一个itemset即{Milk, Diaper, Beer}导出；
- 从同一个itemset导出的规则的 support值一样但confidence值不一样；

先根据support后根据confidence进行过滤

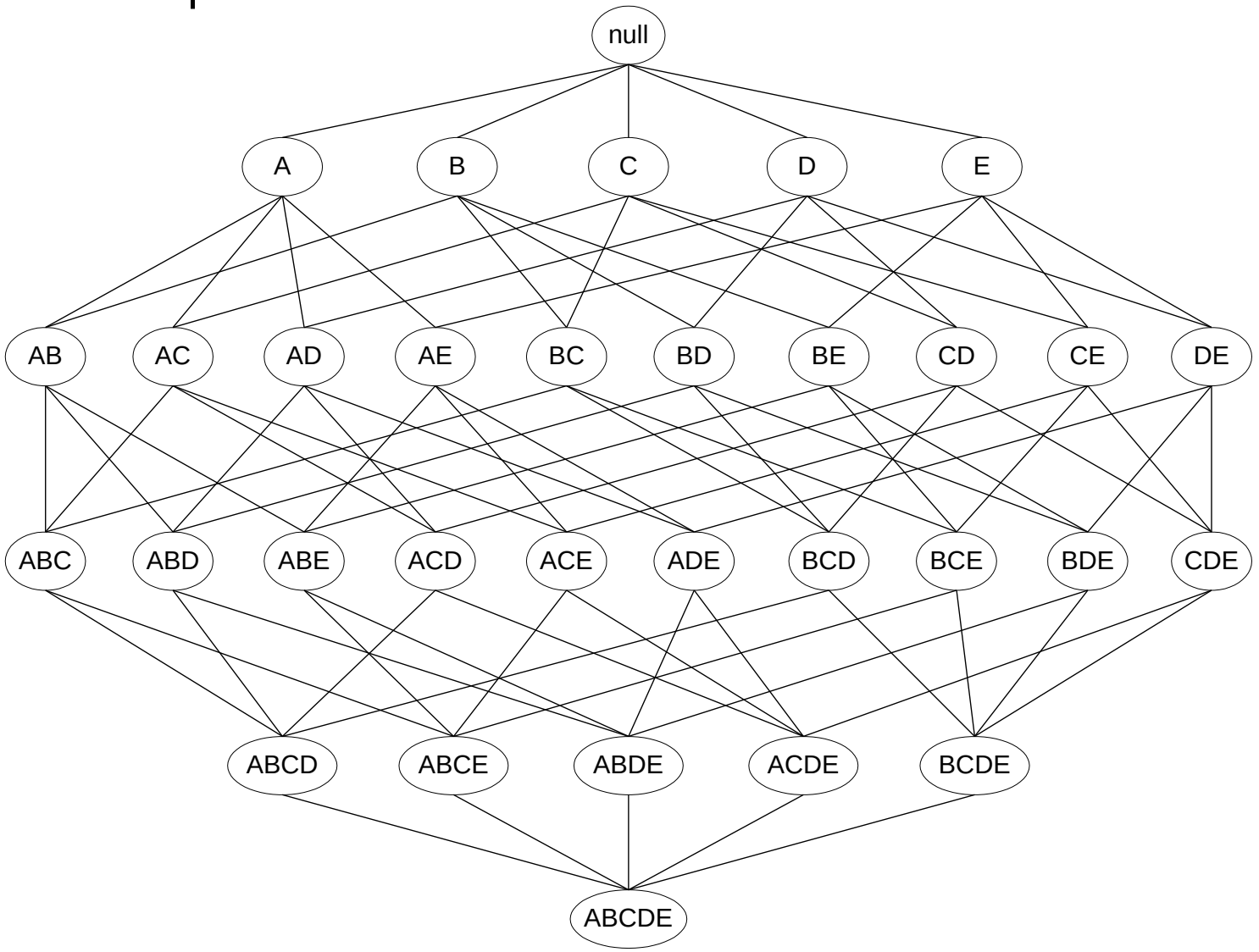
两步法解决关联规则挖掘

两步法:

1. **频繁项集frequent itemset挖掘：**得到 support ≥ MinSup的所有itemsets。
2. **生成高置信度的规则：**从每个frequent itemset生成confidence ≥ MinConf的规则。

然而，快速得到Frequent itemset 计算量依然太大！

Frequent Itemset 生成

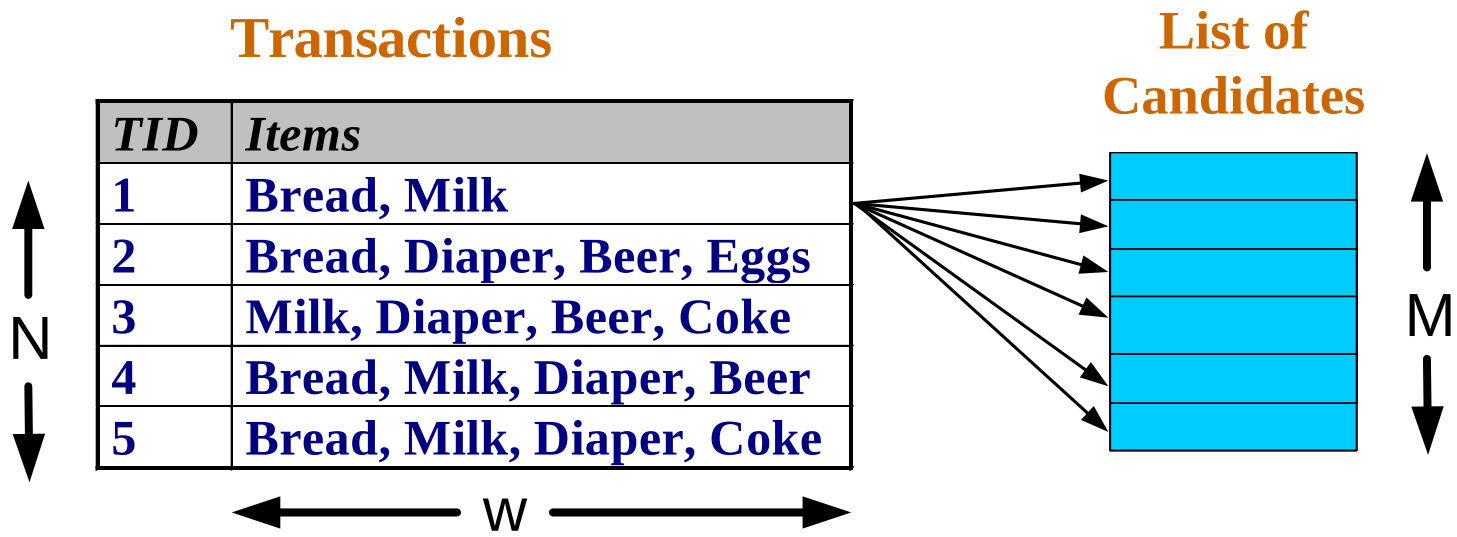


给定 d个item, 有 2^d 可能的 itemsets

沃尔玛超市出售的商品数目为100,000，保留的交易记录达数十亿条。

Frequent Itemset 生成

- 简单暴力的方法:
  - 假设每一个itemset都是候选的frequent itemset
  - 对数据集扫描得到每个itemset的support。



- 把每条交易记录扫描所有候选itemset，更新对应itemset的support；
- 复杂度 ~ O(NMw) => 计算量太大因为 $M = 2^d$!!!

第七章 关联规则挖掘

7.1 关联规则挖掘基本概念

7.2 频繁集挖掘算法：Apriori

7.3 规则生成

减少候选frequent item的数目

- Apriori 原理:

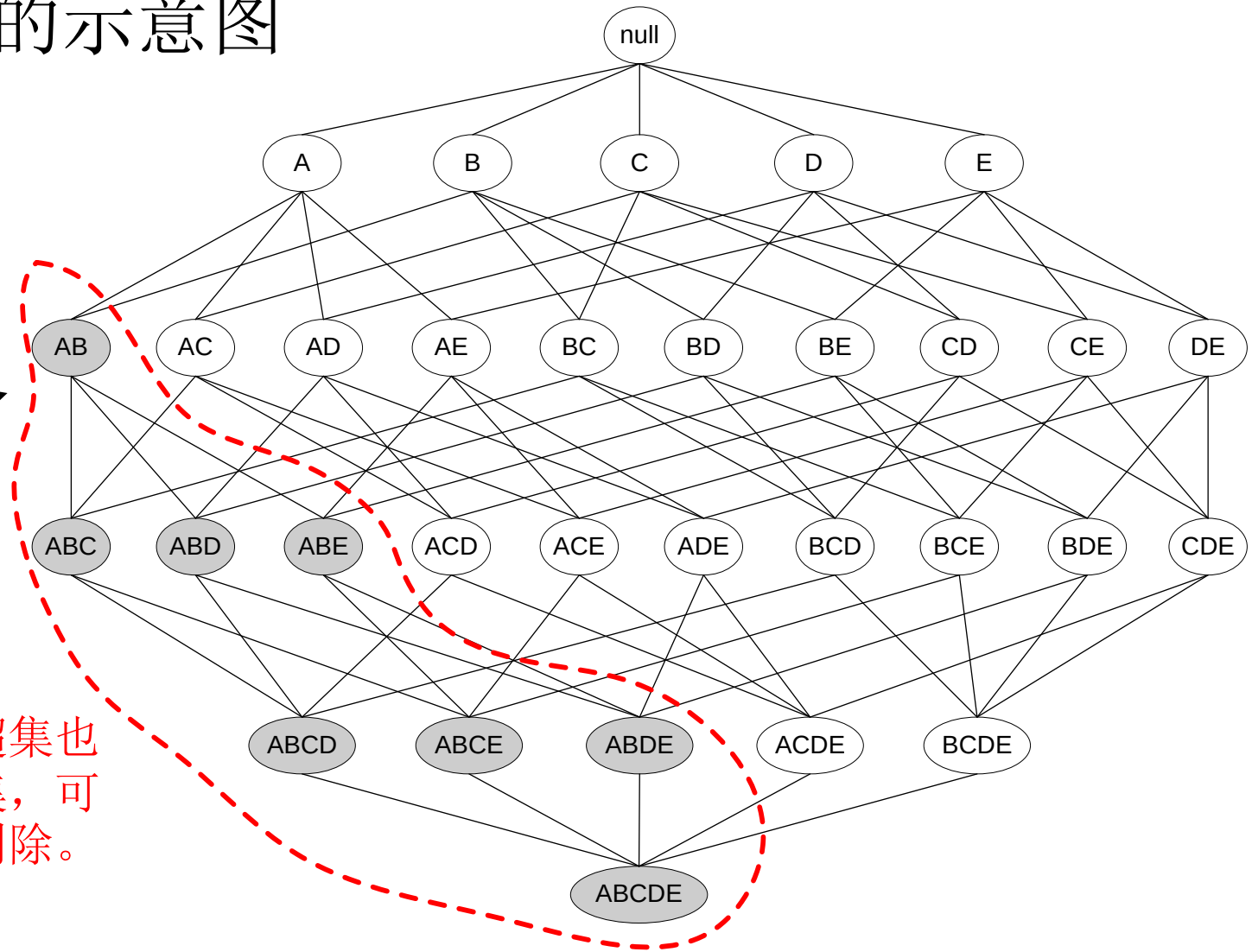
  如果一个 itemset是频繁的，那么它的所有非空子集也是频繁集。

- Apriori 原理成立因为support具有反单调性性质，即一个itemset的support不会超过它任何一个子集的support：

$$\forall X, Y : (X \subseteq Y) \Rightarrow s(X) \geq s(Y)$$

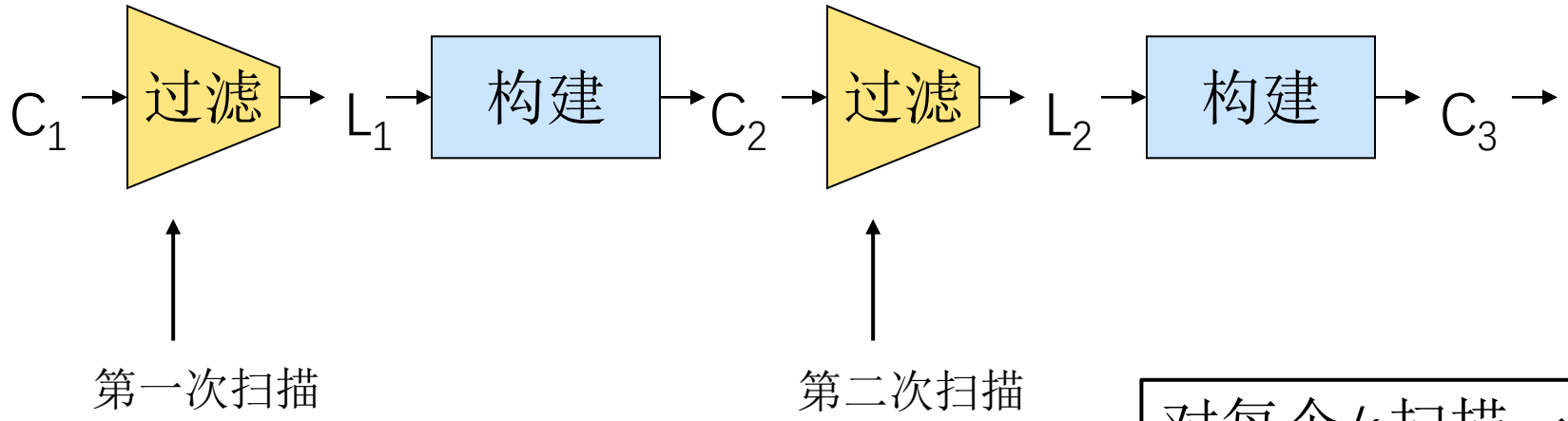
倒过来，如果一个itemset已经被判定为不频繁，那么所有包含它的集合（超集）也都不是频繁集。

Apriori 原理的示意图



若{AB}不满足频繁集条件，即其支持度小于MinSup。

则它的所有超集也都不是频繁集，可以从候选集剔除。



C_1 = 所有的单个item

L_1 = 通过第一轮扫描得到的support ≥ MinSup 的1-itemsets.

一般地

C_k =基于 L_{k-1}构建的k–itemsets 候选集

L_k = support ≥ MinSup 的k-itemsets.

Apriori 算法

- 初始化 k=1，把所有item作为候选 1-itemsets即C_1；
- 重复以下两个步骤直到没有新的频繁集被找到。

  Step1.扫描过滤

  ① 扫描一遍数据集，得到C_k中每个候选k-itemsets的support;

  ② 删除support<MinSup的项,只保留频繁k-itemsets L_k;

  Step2.构建候选k+1项集：

  ① 从L_k中产生初始C'_{k+1}：合并L_k中具有相同k-1个前缀的两个k-itemsets;

  ② 从C'_{k+1}中剔除包含不频繁k-itemset子集的项，得到C_{k+1};

Apriori 原理举例

MinSup = 3

Item（C1）	Count
Bread	4
Coke	2
Milk	4
Beer	3
Diaper	4
Eggs	1

1-itemsets

Item（L1）
Bread
Milk
Beer
Diaper

Itemset（C2）	Count
{Bread,Milk}	3
{Bread,Beer}	2
{Bread,Diaper}	3
{Milk,Beer}	2
{Milk,Diaper}	3
{Beer,Diaper}	3

2-itemsets

Itemset（L2）
{Bread,Milk}
{Bread,Diaper}
{Milk,Diaper}
{Beer,Diaper}

合并只有一个item不同的两个k-itemset得到(k+1)-itemset

Itemset(C_3')
{Bread,Milk,Diaper}
{Bread,Beer, Diaper}
{Milk, Beer, Diaper}

剔除包含非频繁k-itemsets的(k+1)-itemset

Itemset(C_3)	Count
{Bread,Milk,Diaper}	3

产生候选项集的总数对比：

如果考虑所有组合：

$$C_6^1 + C_6^2 + C_6^3 = 41$$

基于Apriori方法：

$$C_6^1 + C_4^2 + 1 = 13$$

第七章 关联规则挖掘

7.1 关联规则挖掘基本概念

7.2 频繁集挖掘算法：Apriori

7.3 规则生成

规则生成

- 给定一个频繁项集 L, 找到所有非空子集 $f \subset L$ 使得$f \rightarrow L - f$ 满足最小置信度要求。
- 如果 {A,B,C,D} 是一个频繁项集, 候选规则有:

ABC →D,	ABD →C,	ACD →B,	BCD →A,
A →BCD,	B →ACD,	C →ABD,	D →ABC
AB →CD,	AC → BD,	AD → BC,	BC →AD,
BD →AC,	CD →AB,		

- 如果 $|L| = k$, 那么有 $2^k - 2$ 候选关联规则 (忽略 $L \rightarrow \varnothing$ and $\varnothing \rightarrow L$)

规则生成

- 如何从频繁 itemsets生成规则?
  - 一般地, 置信度不具有反单调性质。

    c(ABC →D) 可以大于或小于 c(AB →D)

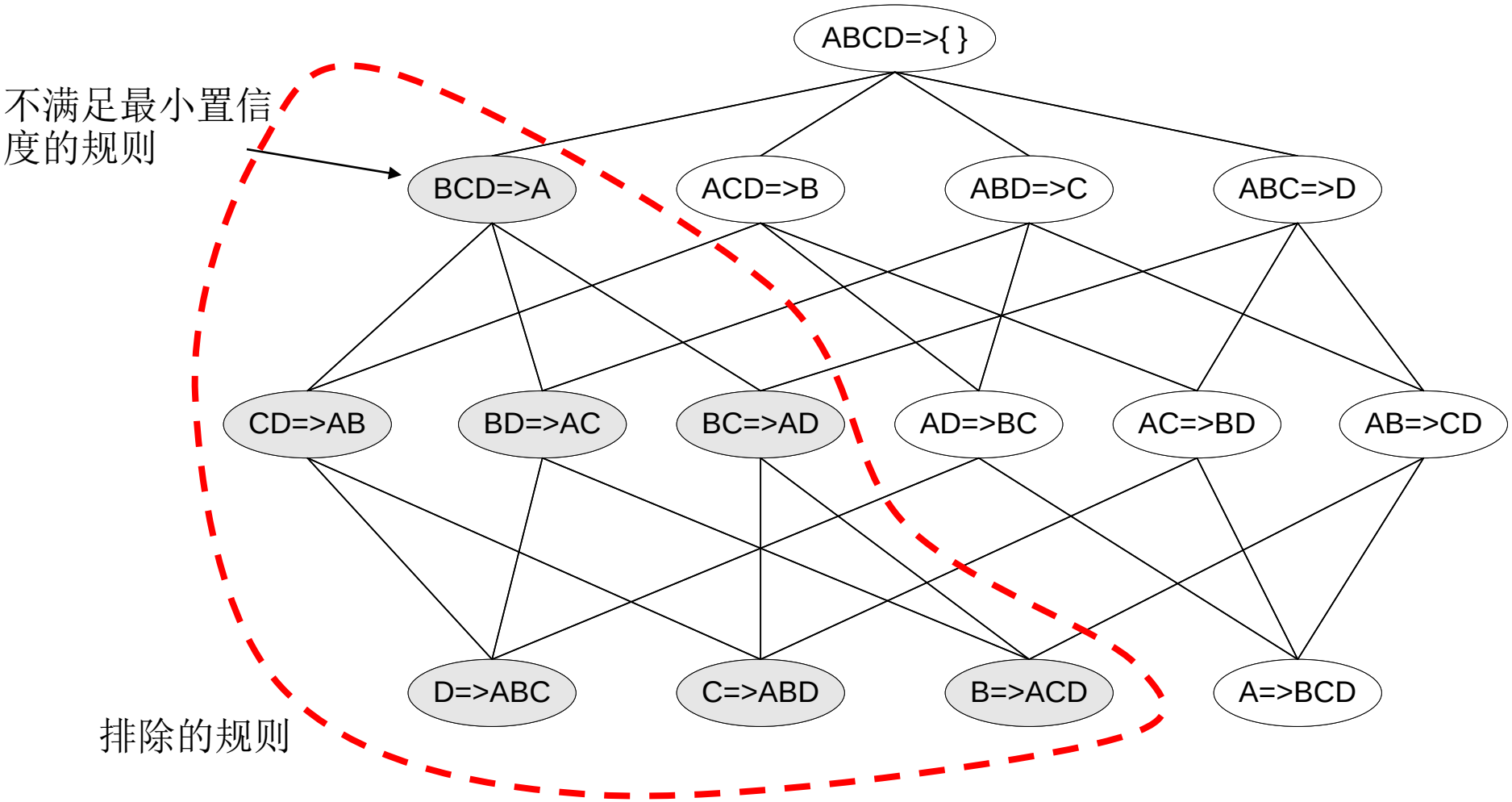
  - 但是从同一个itemset产生地规则具有反单调性质。

    如 L = {A,B,C,D}:

    $$c(ABC \rightarrow D) \geq c(AB \rightarrow CD) \geq c(A \rightarrow BCD)$$

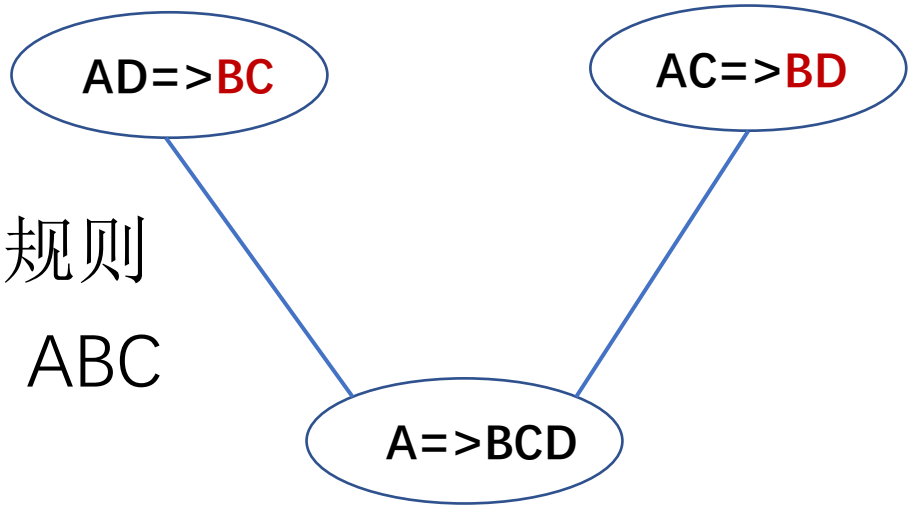
  - 置信度与箭头右边的items具有反单调性质。

Apriori 算法的规则生成



Apriori 算法的规则生成

- 候选规则通过合并箭头右边具有相同前缀的规则得到



- 合并(AD=>BC,AC=>BD，假设这两个规则的置信度满足要求)得到候选规则 D => ABC

- 排除 A=>BCD如果它的子集 AB=>CD 置信度小于阈值

购物车模型应用-文本分析（拓展）

- “购物车模型”是一个抽象模型，用于发现基于不同定义的“交易记录”和“item”两个概念之间的关联关系 ；
- 算法对多个item同时出现在某个交易记录的次数进行计算，而不是倒过来。

例子1:定义“交易记录”=>文本; “item” =>文本中的词。
关联规则对应经常一起出现的词，可用于话题检测。

例子2:定义“交易记录”=>句子; “item”=>包含这些句子的文档。
关联规则对应经常同时包含某些句子的文档，可用于发现抄袭。