Apriori Algorithm
for
Association Rule Mining

CSED, TIET

Apriori Algorithm

- Apriori algorithm is developed by two Indians Rakesh Agarwal and Ramakrishnan Shrikant in 1994.

- Apriori algorithm works in following two phases:

    Phase 1: Identification of Frequent Item sets.

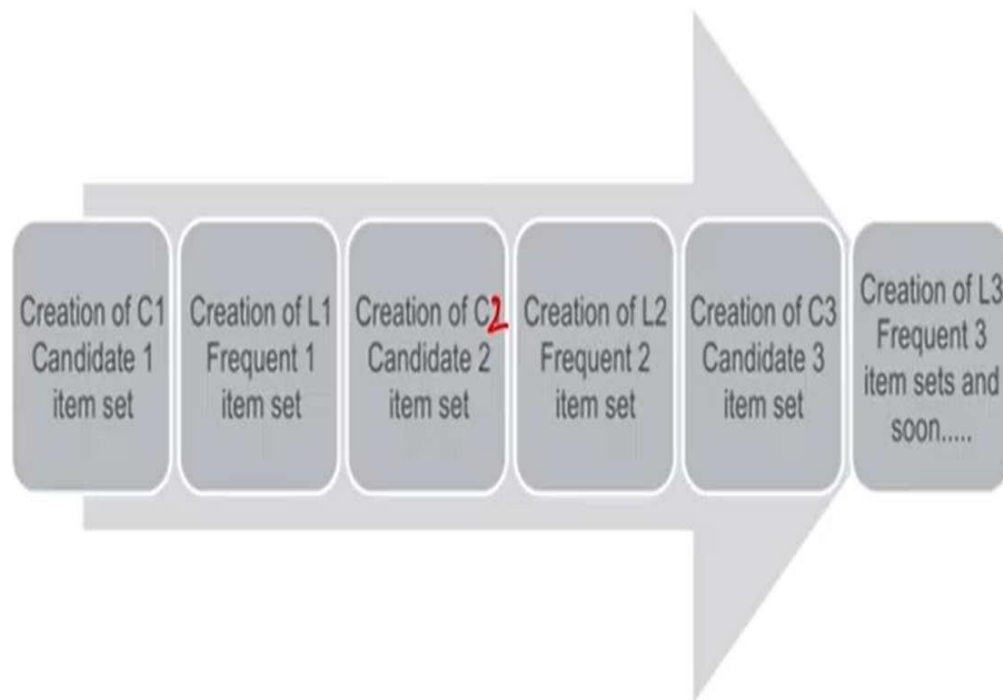   Phase 2: Generation of Association Mining Rules.

Phase 1: Identification of Frequent Item sets

▪In this phase we generate frequent item sets from transactional database that qualifies the criteria of minimum support and confidence. But, it is necessary that the items in each transaction are listed in ascending order (or sorted order).

▪ This phase generates frequent item sets in the following steps:

1. Identify Candidate One Itemset C1- i.e. all the items present in the dataset.

2. Identify Frequent One Itemset L1- i.e. all one items having frequency greater than or equal to threshold value of support.

3. Identify Candidate 2-Itemsets C2- which is generated as L1 join L1 i.e. join of L1 with L1.

4. Identify Frequent 2-Itemsets L2- all 2-Itemsets having frequency greater than or equal to threshold value of support.

5. Similarly, repeat steps 3 and 4 to generate L3, L4,…

Phase 1: Identification of Frequent Item sets

- In general, each kth candidate Itemset (C_k) is produced through L_{k-1} Join L_{k-1}. For instance, C_3 is produced through L_2 JOIN L_2.

- Two item sets are joinable if their first k-2 items are same. So, in case of C3, the first item in L2 are same. Similarly for C4, first two items in L3 should be same, and so on.

- In case of C2, there is no such requirement because k-2 is 0.

Final Process of Phase-1

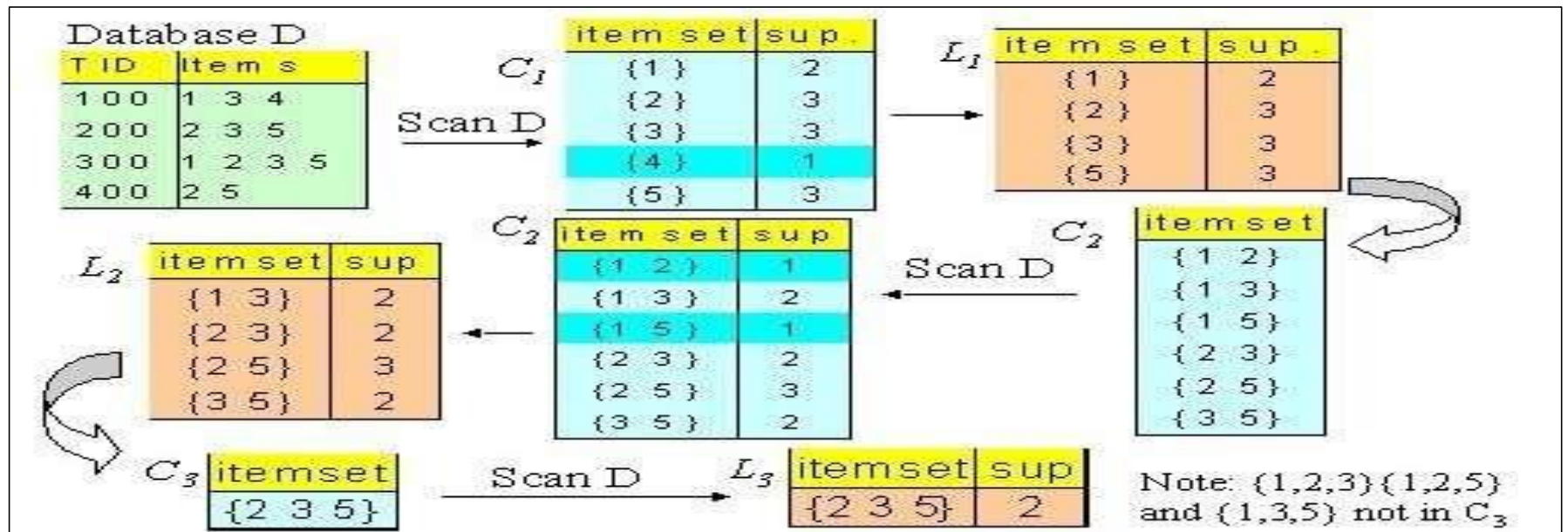# Phase 2: Generation of Association Mining Rules

- In this phase, association mining rules are generated.

- This works in following phase:

1. For each itemset L in the last frequent iItemsets, generate all possible non null subsets S.

2. Generate all possible rules s → L-s for s εS.

3. Compute confidence of each rule.

4. Add all the rules in the final set that qualifies the minimum confidence criteria.

5. For all the added rules, add rules that are implicitly generated from them and satisfy the minimum confidence threshold criteria.

Apriori Algorithm- Example 1

Find association rules with minimum support of 50% and confidence of 75%

T ID	Items
100	1 3 4
200	2 3 5
300	1 2 3 5
400	2 5

Database D

Example 1-Solution (Phase 1)

Database D

TID	Items
100	1 3 4
200	2 3 5
300	1 2 3 5
400	2 5

Scan D →

C_1

itemset	sup.
{1}	2
{2}	3
{3}	3
{4}	1
{5}	3

L_1

itemset	sup.
{1}	2
{2}	3
{3}	3
{5}	3

C_2

itemset
{1 2}
{1 3}
{1 5}
{2 3}
{2 5}
{3 5}

C_2

itemset	sup
{1 2}	1
{1 3}	2
{1 5}	1
{2 3}	2
{2 5}	3
{3 5}	2

Scan D

L_2

itemset	sup
{1 3}	2
{2 3}	2
{2 5}	3
{3 5}	2

C_3

itemset
{2 3 5}

Scan D →

L_3

itemset	sup
{2 3 5}	2

Note: {1,2,3}{1,2,5} and {1,3,5} not in C_3

Example 1-Solution (Phase 2)

- The 3- frequent item sets generated in phase 1 are (2,3,5)
- Confidence for each possible rule are:

$$confidence\,(2 \rightarrow 3,5) = \frac{n(2,3,5)}{n(2)} = \frac{2}{3} = 66.7\%$$

$$confidence\,(3,5 \rightarrow 2) = \frac{n(2,3,5)}{n(3,5)} = \frac{2}{2} = 100\%$$

$$confidence\,(3 \rightarrow 2,5) = \frac{n(2,3,5)}{n(3)} = \frac{2}{3} = 66.7\%$$

$$confidence\,(2,5 \rightarrow 3) = \frac{n(3,2,5)}{n(2,5)} = \frac{2}{3} = 66.7\%$$

$$confidence\,(5 \rightarrow 2,3) = \frac{n(2,3,5)}{n(5)} = \frac{2}{3} = 66.7\%$$

$$confidence\,(2,3 \rightarrow 5) = \frac{n(5,2,3)}{n(2,3)} = \frac{2}{2} = 100\%$$

Example 1-Solution (Phase 2)

Rule	Confidence
2, 3→5	1.0
2→5	$S(2 \cap 5) / S(2) = 3/3 = 1.0$
3→5	$S(2 \cap 5) / S(3) = 3/3 = 1.0$
3, 5→2	1.0
3→2	$S(3 \cap 2) / S(3) = 2/3 = 0.67$
5→2	$S(5 \cap 2) / S(5) = 3/3 = 1.0$

Thus, final rules having confidence more than the threshold limit, i.e., 75% are as follows.

Selected Association rules are
2, 3→5
2→5
3→5
3, 5→2
5→2

Improvement of Apriori Algorithm- Pruning

- Apriori algorithm can be improved using Apriori property.

- *Apriori property states that all non empty subsets of a frequent itemset must also be frequent.*

- The itemset which does not satisfy Apriori property should be removed from the candidate set.

- This step is called *pruning* of the candidate set.

- It will help in improvement in performance of Apriori algorithm as it will reduce the search space.

Improved Apriori Algorithm- Example 2

Generate association rules according to Apriori algorithm (with pruning)

- Threshold value of Support 15% and Confidence 70%

TID	List of Items
10	I1, I2, I5
20	I2, I4
30	I2, I3
40	I1, I2, I4
50	I1, I3
60	I2, I3
70	I1, I3
80	I1, I2, I3, I5
90	I1, I2, I4

Table 9.41   C1

C1	Count
I1	6
I2	7
I3	5
I4	3
I5	2

Example 2 – Solution (Phase-1)

TID	List of Items
T100	I1, I2, I5
T200	I2,I4
T300	I2,I3
T400	I1,I2,I4
T500	I1,I3
T600	I2,I3
T700	I1,I3
T800	I1,I2,I3,I5
T900	I1,I2,I3

L1

C1	Count
I1	6
I2	7
I3	5
I4	3
I5	2

Example 2 – Solution (Phase-1 Contd…)

Process of Generation of C2 and L2

C1

C1
I1
I2
I3
I4
I5

Generate C_2 Candidates from L_1 →

C_2

Itemset
{I1,I2}
{I1,I3}
{I1,I4}
{I1,I5}
{I2,I3}
{I2,I4}
{I2,I5}
{I3,I4}
{I3,I5}
{I4,I5}

Scan D for count of each candidate →

C_2

Itemset	Sup. count
{I1,I2}	4
{I1,I3}	4
{I1,I4}	1
{I1,I5}	2
{I2,I3}	4
{I2,I4}	2
{I2,I5}	2
{I3,I4}	0
{I3,I5}	1
{I4,I5}	0

Compare candidate support count with minimum support count

L_2

Itemset	Sup. count
{I1,I2}	4
{I1,I3}	4
{I1,I5}	2
{I2,I3}	4
{I2,I4}	2
{I2,I5}	2

Example 2 – Solution (Phase-1 Contd...)

Generation of C3

L_2

Itemset	Sup. count
{I1,I2}	4
{I1,I3}	4
{I1,I5}	2
{I2,I3}	4
{I2,I4}	2
{I2,I5}	2

Generation of C3

C3
I1, I2, I3
I1, I2, I5
I1, I3, I5
I2, I3, I4
I2, I3, I5
I2, I4, I5

Example 2 – Solution (Phase-1 Contd...)

Pruning of C3

Generation of C3

C3
I1, I2, I3
I1, I2, I5
I1, I3, I5
I2, I3, I4
I2, I3, I5
I2, I4, I5

Pruned C3

Pruned C3
I1, I2, I3
I1, I2, I5

L_2

Itemset	Sup. count
{I1,I2}	4
{I1,I3}	4
{I1,I5}	2
{I2,I3}	4
{I2,I4}	2
{I2,I5}	2

Example 2 – Solution (Phase-1 Contd...)

Generation of L3

Generate C_3 Candidates From L_2

C_3

Itemset
{I1, I2, I3}
{I1, I2, I5}

Scan D for count of each candidate

C_3

Itemset	Sup. count
{I1, I2, I3}	2
{I1, I2, I5}	2

Compare candidate support count with minimum support count

L_3

Itemset	Sup. count
{I1, I2, I3}	2
{I1, I2, I5}	2

Example 2 – Solution (Phase-1 Contd...)

Generation of L3

Generate C_3 Candidates From L_2 →

C_3

Itemset
{I1, I2, I3}
{I1, I2, I5}

Scan D for count of each candidate →

C_3

Itemset	Sup. count
{I1, I2, I3}	2
{I1, I2, I5}	2

Compare candidate support count with minimum support count →

L_3

Itemset	Sup. count
{I1, I2, I3}	2
{I1, I2, I5}	2
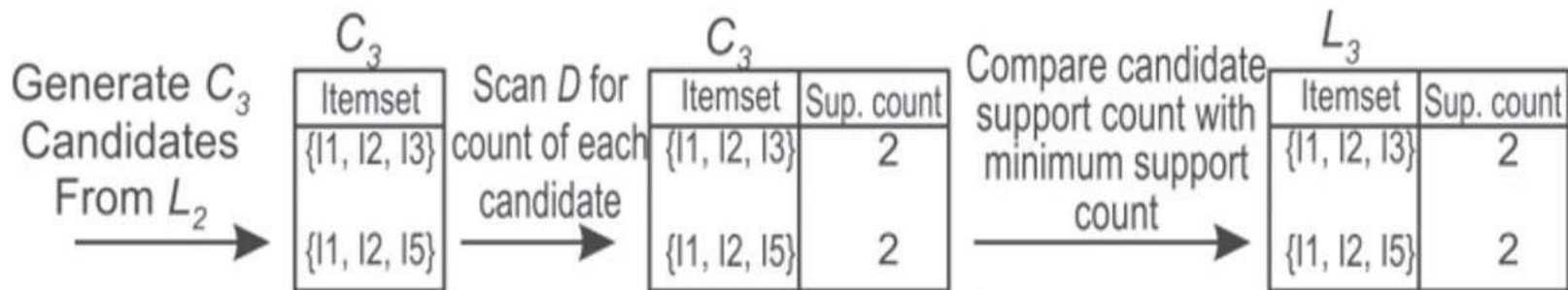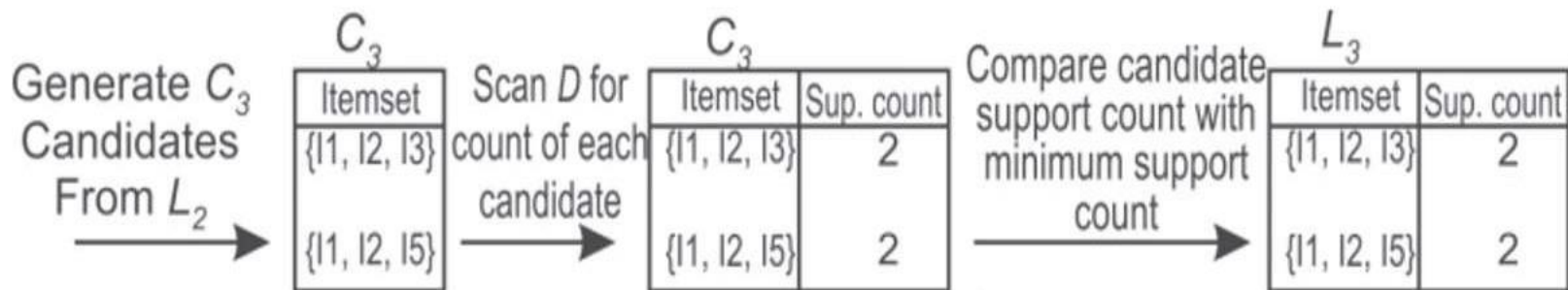
Example 2 – Solution (Phase-1 Contd…)

Generation and Pruning of C4

C4

C4
I1, I2, I3, I5

But this itemset is pruned by the Apriori property because its subset (I2, I3, I5) is not frequent as it is not present in L3. Thus, C4 is null and the algorithm terminates at this point, having found all of frequent itemsets as shown in Table :

Pruned C4

C4
NULL

Example 2 – Solution (Phase-2 Contd...)

Let us apply this rule to frequent 3-itemsets (I1, I2, I3) and (I1, I2, I5) found in case of example

For first frequent itemset (I1, I2, I3), non-empty subsets are {{I1},{I2},{I3},{(I1, I2)}, {(I1, I3)},{(I2, I3)}.

For every non-empty set, the rule will be generated as follows:

I1→I2, I3 [Here, (I1) is s and I2 and I3 are l–s]
I2→I1, I3
I3→I1, I2
I1, I2→I3
I1, I3→I2
I2, I3→I1

The next will be to calculate the confidence for each rule as shown below.

I1→I2, I3; Confidence = S(I1∩I2∩I3) / S(I1) = 2/6 = 0.3
I2→I1, I3; Confidence = S(I1∩I2∩I3) / S(I2) = 2/7 = 0.28
I3→I1, I2; Confidence = S(I1∩I2∩I3) / S(I3) = 2/5 = 0.4
I1, I2→I3; Confidence = S(I1∩I2∩I3) / S(I1∩I2) = 2/4 = 0.5
I1, I3→I2; Confidence = S(I1∩I2∩I3) / S(I1∩I3) = 2/4 = 0.5
I2, I3→I1; Confidence = S(I1∩I2∩I3) / S(I2∩I3) = 2/4 = 0.5

Example 2 – Solution (Phase-2 Contd…)

Now, let us apply this rule to second frequent 3-itemset (I1, I2, I5). For this frequent itemset, non-empty subsets are {I1}, {I2}, {I5}, {(I1, I2)}, {(I1, I5)}, {(I2, I5)}.

For every non-empty set the rule will be generated as follows:

I1→I2, I5 (Here, (I1) is s and I2 and I5 are $l-s$]
I2→I1, I5
I5→I1, I2
I1, I2→I5
I1, I5→I2
I2, I5→I1

The next step will be to calculate the confidence for each rule as shown below.

I1→I2, I5; Confidence = S(I1∩I2∩I5) / S(I1) = 2/6 = 0.3
I2→I1, I5; Confidence = S(I1∩I2∩I5) / S(I2) = 2/7 = 0.28
I5→I1, I2; Confidence = S(I1∩I2∩I5) / S(I5) = 2/2 = 1
I1, I2→I5; Confidence = S(I1∩I2∩I5) / S(I1∩I2) = 2/4 = 0.5
I1, I5→I2; Confidence = S(I1∩I2∩I5) / S(I1∩I5) = 2/2 = 1
I2, I5→I1; Confidence = S(I1∩I2∩I5) / S(I2∩I5) = 2/2 = 1

Now, there are three rules whose confidence is more than minimum threshold value of 70%, and these rules are as follows:

I5→I1, I2
I1, I5→I2
I2, I5→I1

Example 2 – Solution (Phase-2 Contd...)

Association Rule	Discussion	Confidence	More than threshold limit Or Not
I5→I1, I2	Already identified	1.0	Yes
I5→I1	Implicit	No need to calculate it will be more than or equal to I5→I1, I2	Yes
I5→I2	Implicit	No need to calculate it will be more than or equal to I5→I1, I2	Yes
I2, I5→I1	Already identified	1.0	Yes
I2→I1	Not found earlier, confidence need to be calculated	Confidence of I2→I1 = S(I2∩I1) / S(I1) = 4/6 = 67%	No
I5→I1	Already found from I5→I1, I2 given in row 2	No need to calculate it will be more than or equal to I5→I1, I2	Yes (Already listed)
I1, I5→I2	Already identified	1.0	Yes
I1→I2	Not found earlier, so confidence needs to be calculated	Confidence of I1→I2 = S(I2∩I1) / S(I2) = 4/7 = 57%	No
I5→I2	Already found from I5→I1, I2 given in row 3		Yes (Already listed)

Found to be Infrequent

Pruned supersets
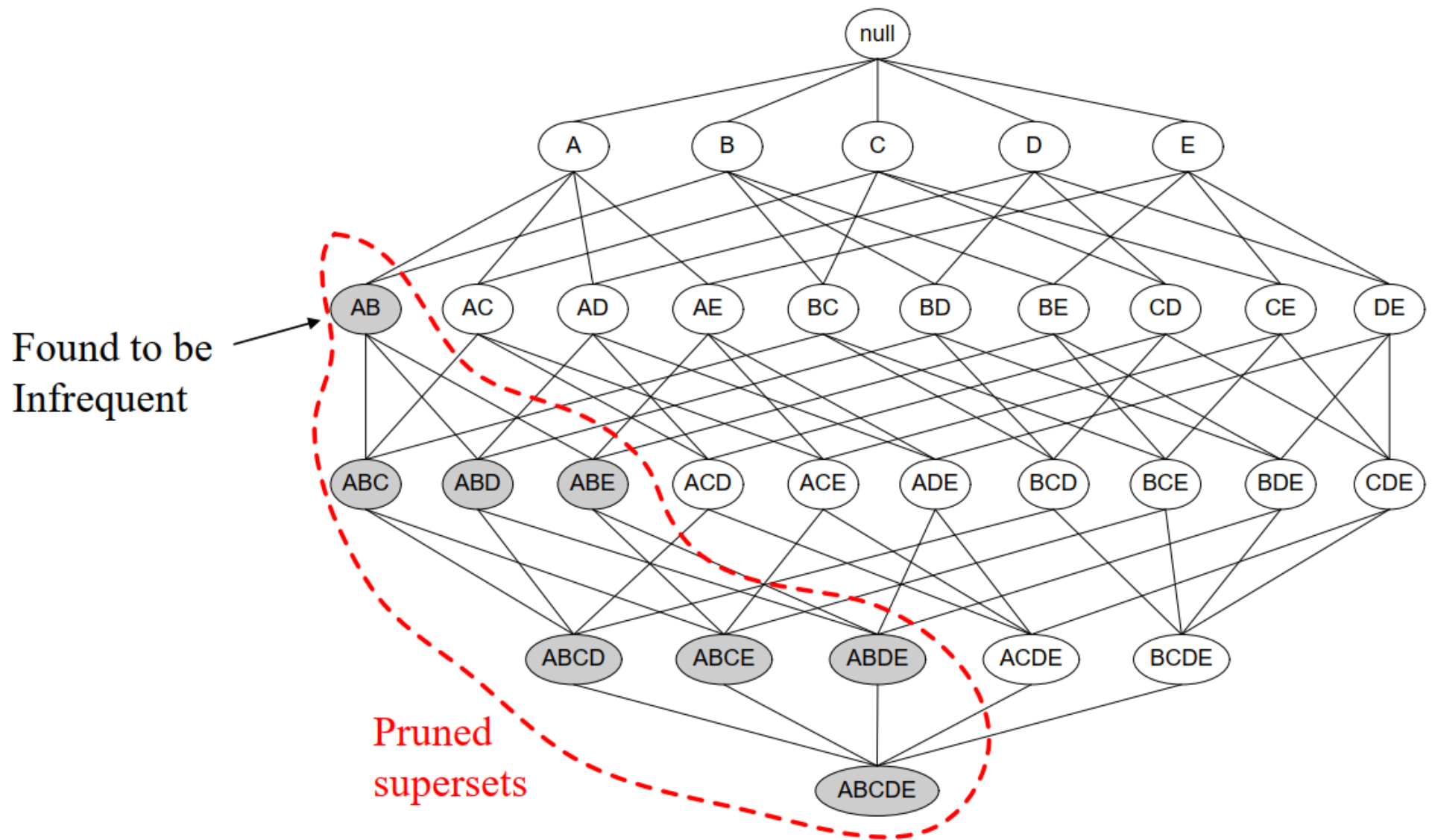
- Maximal Frequent Itemsets
- Closed Frequent Itemsets