Algorithms: Apriori (association rules)

Statement for Audio and Video Learning Resources

Video and audio content at the University uses closed captions generated by automatic speech recognition (ASR). The ASR process is based on machine learning algorithms which automatically transcribe voice to text. According to our technology providers, this process is approximately 70-90% accurate depending on the quality of the audio, and consequently video and audio closed captions may include some transcription errors. It is therefore important to recognise that the original recording is the most accurate reflection of the content, and not the captions.

If you require accurate captions as part of your reasonable adjustments, please contact the Inclusion Centre to discuss your requirements.

Contents

- What are Association Rules?
- Itemsets and Rules
- Apriori Algorithm
- Summary

Classification Rule (Revision)

Predicts value of pre-specified attribute (class or label)

- **If** Outlook=overcast **then** play=yes

Label (class)

Outlook	Temp	…	Play
Overcast	Cool	…	Yes
…	…	…	…
Rainy	Warm	…	Yes

Attributes (Features)

Association Rules

Predicts value of arbitrary attribute or combination

- **If** temperature=cool **then** play=yes
- **If** outlook=sunny **and** play=no **then** humidity=high
- **If** windy=false **and** play=no **then** outlook=sunny **and** humid=high

Typical example: market basket associations

- milk orange_juice ⇒ bread
- baby_food nappies ⇒ beer crisps

Association Rules – what are they good for?

The find groups and patterns within the dataset. This could tell you:

- What items are bought together in a supermarket (so you can position them appropriately)
- What terms typically occur together in spam emails (and whether the same terms co-occur in regular emails, too)
- What shortcuts your machine learning algorithm might take
- **They are associative NOT causative**

Association Rules & Itemsets (Jargon)

Item: attribute-value pair

- (e.g. temperature = cool)

Itemset: set of items occurring in data

- e.g. temperature = cool & humidity = high

An Itemset with k items is a *k-itemset*

Association rule

relationship between two **disjoint itemsets** X and Y

$X \Rightarrow Y$

if X occurs then Y also occurs

Association Rules (more jargon)

Itemset

Item

Item

If windy=false **and** play=no

then

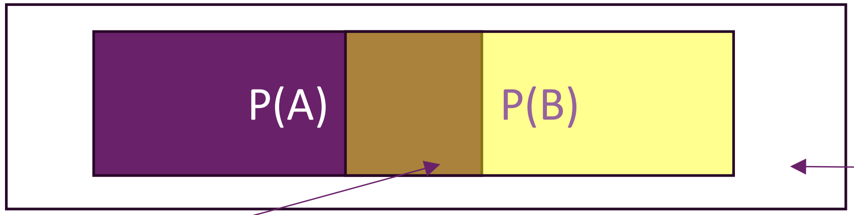
outlook=sunny **and** humid=high

Antecedent (condition, premise)

Consequent (conclusion)

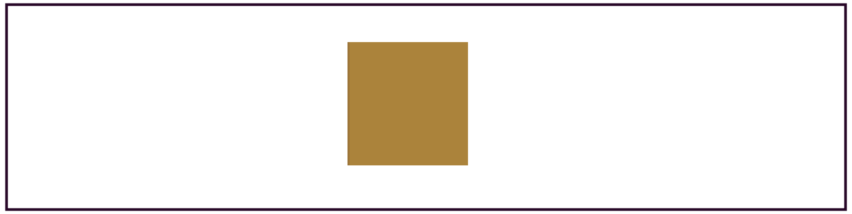
Support: Of all the instances in the dataset, what percentage does this rule apply to?
Confidence: Out of all the instances in the dataset where the antecedent applies, what percentage also have the consequent?

Support and confidence pictorially

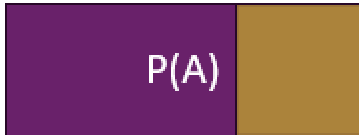


Rest of dataset
(neither A nor B)

Rule that A $\Rightarrow$ B



Support: % of instances where rules apply out of all data



Confidence: % of antecedent instances where consequent applies

Association Rule Generation

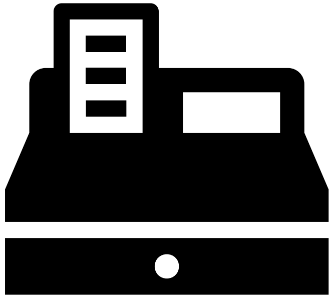
- Given a set of transactions
- STEP1
  - Generate itemsets with specified minimum support (coverage)
- STEP2
  - Determine rules that have specified minimum confidence (accuracy)

Market Basket Analysis

- Supermarkets, collect and store massive amounts of sales data, called *market basket data.* A record consist of transaction date and items bought.
- "90% of transactions that purchase bread and butter also purchase milk"
  - Antecedent (condition, premise): bread and butter
  - Consequent (conclusion): milk
  - Confidence factor: 90%
  - Support?

$$\text{bread butter} \Rightarrow \text{milk (90\%)}$$

Example : Market Basket



I: itemset

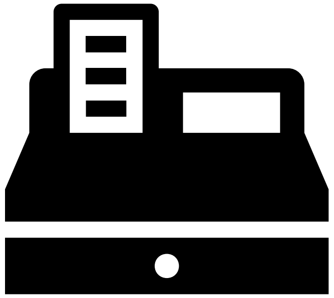
- {tomato, cucumber}
- {parsley, onion}

{tomato, cucumber} $\Rightarrow$ {parsley, onion}

Data: set of transactions

1. {cucumber, parsley, onion, tomato, salt, bread}
2. {tomato, cucumber, parsley}
3. {tomato, cucumber, olives, onion, parsley}
4. {tomato, cucumber, onion, bread}
5. {tomato, salt, onion}
6. {bread, cheese}
7. {tomato, cheese, cucumber}
8. {bread, butter}

Example : Market Basket



I: itemset

- {tomato, cucumber}
- {parsley, onion}

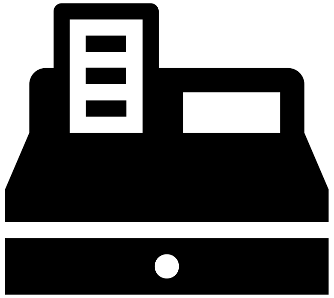
{tomato, cucumber} ⇒ {parsley, onion}

Data: set of transactions

1. **{cucumber, parsley, onion, tomato, salt, bread}**
2. **{tomato, cucumber, parsley}**
3. **{tomato, cucumber, olives, onion, parsley}**
4. **{tomato, cucumber, onion, bread}**
5. {tomato, salt, onion}
6. {bread, cheese}
7. **{tomato, cheese, cucumber}**
8. {bread, butter}

5 instances that contain our precedent {tomato, cucumber}

Example : Market Basket



{tomato, cucumber} $\Rightarrow$ {parsley, onion}

I: itemset

- {tomato, cucumber}
- {parsley, onion}

Data: set of transactions

1. **{cucumber, parsley, onion, tomato, salt, bread}**
2. {tomato, cucumber, parsley}
3. **{tomato, cucumber, olives, onion, parsley}**
4. {tomato, cucumber, onion, bread}
5. {tomato, salt, onion}
6. {bread, cheese}
7. {tomato, cheese, cucumber}
8. {bread, butter}

Confidence:
2 out of 5 instances (40%) that contain our precedent also contain our consequent {parsley, onion}

Support
2 out of 8 instances (25%) contain both precedent and consequent

Example : Spam Filtering

- **I: itemset (set of keywords)**
  - {porn, viagra, mail}
  - {mortgage, apply, mail, ham}
- **Data: set of emails**
  1. {language, maths, mail, ham}
  2. {maths, language, apply, ham}
  3. {language, apply, free, spm}
  4. {apply, mortgage, free, mail, spm}
  5. {porn, free, spm}
  6. {maths, language, apply, ham}

Example: Weather Data

Outlook	Temperature	Humidity	Windy	Play
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Cloudy	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
Cloudy	Cool	Normal	True	Yes
Sunny	Mild	High	False	No
Sunny	Cool	Normal	False	Yes
Rainy	Mild	Normal	False	Yes
Sunny	Mild	Normal	True	Yes
Cloudy	Mild	High	True	Yes
Cloudy	Hot	Normal	False	Yes
Rainy	Mild	High	True	No

Support and Confidence

- Outlook = sunny $\Rightarrow$ temperature = hot

  A $\Rightarrow$ B

- **Support (A , B)** = $P(A \wedge B)$ = 2/14
  - Proportion of instances where A and B appear together.
- **Confidence (A $\Rightarrow$ B)** = $\frac{P(A \wedge B)}{P(A)}$ = 2/5
  - Proportion of instances where B appears out of the instances where A appears.
  - So, proportion of instances where the rule is true out of the number of times when the rule is applicable.
- **Support for an itemset, confidence for a rule.**

Probability of A and B appearing together in the data

Confidence of A $\Rightarrow$ B may be very different to confidence of B $\Rightarrow$ A

ROBERT GORDON UNIVERSITY ABERDEEN

Lift

- **Lift(A $\Rightarrow$ B) =** $\frac{P(A \wedge B)}{P(A)*P(B)} = \frac{\frac{2}{14}}{\frac{5}{14}*\frac{4}{14}} = \frac{7}{5}$
  - The ratio of the observed support that would be expected if A and B were independent.
    - The rise (or decrease) in probability of having B if we have observed A.
  - Lift (antecedent $\Rightarrow$ consequent) = 1 if antecedent and consequent are independent.
  - If there is a degree of dependency the lift is
    - > 1 if presence of A makes B more likely
    - < 1 if presence of A makes B less likely
- **Support and confidence vary between 0 and 1 (or 0% and 100%)**

Example: Support, Confidence & Lift

TID	Itemsets
T100	A D
T200	B C E
T300	A B C E
T400	B E

- Data set D = {T100,T200,T300,T400}
  - |D| = 4 (4 transactions)
- Support({B C}) = 2/4
  = 0.5 (or 50%)
- Confidence(B $\Rightarrow$ C) = 2/3
  = 0.67 (or 67%)
- Confidence(C $\Rightarrow$ B) = 2/2
  = 1 (or 100%)
- Lift ((B $\Rightarrow$ C) = $\frac{\frac{2}{4}}{\frac{3}{4}*\frac{2}{4}}$ = 4/3

B and C appear together in 2 out of the 4 transactions

When B appears, C also appears in 2 out of 3 transactions

Confidence, Support, Lift…?

You want all three to be high for a “solid” rule

- high support: should apply to a large amount of cases
- high confidence: should be correct often
- high lift: indicates it is not just a coincidence

Example Weather Data

- I: itemset (set of weather attribute-value pairs)
  - {Outlook=Sunny, Windy=true, Play=no}
  - {Outlook=Rainy, Windy=False}
- D: set of weather instances
  1. {Outlook=Sunny, Temp=Hot, Humidity=High, Windy = False, Play=No}
  2. {Outlook=Sunny, Temp=Hot, Humidity=High, Windy = True, Play=No}
  3. {Outlook=Cloudy, Temp=Hot, Humidity=High, Windy = False, Play=Yes}

  …

ROBERT GORDON UNIVERSITY ABERDEEN

Example: Weather Data

Outlook	Temperature	Humidity	Windy	Play
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Cloudy	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
Cloudy	Cool	Normal	True	Yes
Sunny	Mild	High	False	No
Sunny	Cool	Normal	False	Yes
Rainy	Mild	Normal	False	Yes
Sunny	Mild	Normal	True	Yes
Cloudy	Mild	High	True	Yes
Cloudy	Hot	Normal	False	Yes
Rainy	Mild	High	True	No

Itemsets for Weather Data with minimum support 2

one-itemsets	two-itemsets	three-itemsets	four-itemsets
outlook=sunny (5)	outlook=sunny temp=mild (2)	outlook=sunny temp=hot humid=high (2)	outlook=sunny temp=hot humid=high play=no (2)
temp=cool (4)	outlook=sunny humid=high (3)	outlook=sunny humid=high windy = false (2)	outlook=rainy temp=mild windy = false play=yes (2)
…	…	…	…
12 one-itemsets	**47 two-itemsets**	**39 three-itemsets**	**6 four-itemsets**

0 five-itemsets

Rules from Itemsets

- Now turn itemsets with sufficient support into rules
  - 3-itemset: Humidity=Normal, Windy=F, Play=Y
  - Support = 4/14
- Seven potential rules ($7=2^3-1$)

	Confidence
If Humidity = Normal & Windy = F then Play = Y	4/4
If Humidity = Normal & Play = Y then Windy = F	4/6
If Windy = F & Play = Y then Humidity = Normal	4/6
If Humidity = Normal then Windy = F & Play = Y	4/7
If Windy = F then Humidity = Normal & Play = Y	4/8
If Play = Yes then Humidity = Normal & Windy = F	4/9
If True then Humidity = Normal & Windy = F & Play = Y	4/14

ROBERT GORDON UNIVERSITY ABERDEEN

Rules for the Weather Data

- Rules with
  - support ≥ 2/14
  - confidence = 1

		Association Rule		Support	Confidence
3 rules	1	Humidity=Normal & Windy=F	⇒ Play=Y	4/14	100%
	2	Temperature=Cool	⇒ Humidity=Normal	4/14	100%
	3	Outlook=Overcast	⇒ Play=Y	4/14	100%
5 rules	4	Temperature=Cold & Play=Y	⇒ Humidity=Normal	3/14	100%
	...	...	...	...	...
50 rules	5	Outlook=Sunny	⇒ Humidity=High	2/14	100%
	8	Temperature=Hot			

Contents (3)

- What are Association Rules?
- Itemsets and Rules
- Apriori Algorithm
- Summary

Generating Rules Efficiently

- Step1: Generate itemsets efficiently
  - with specified minimum support (coverage)

- Step 2: Determine rules efficiently
  - that have specified minimum confidence (accuracy)

Itemset Generation

Brute-force approach

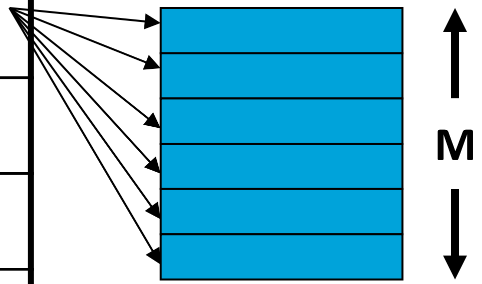
- each itemset in the lattice is a candidate itemset
- count support of each candidate by scanning dataset

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

|D|

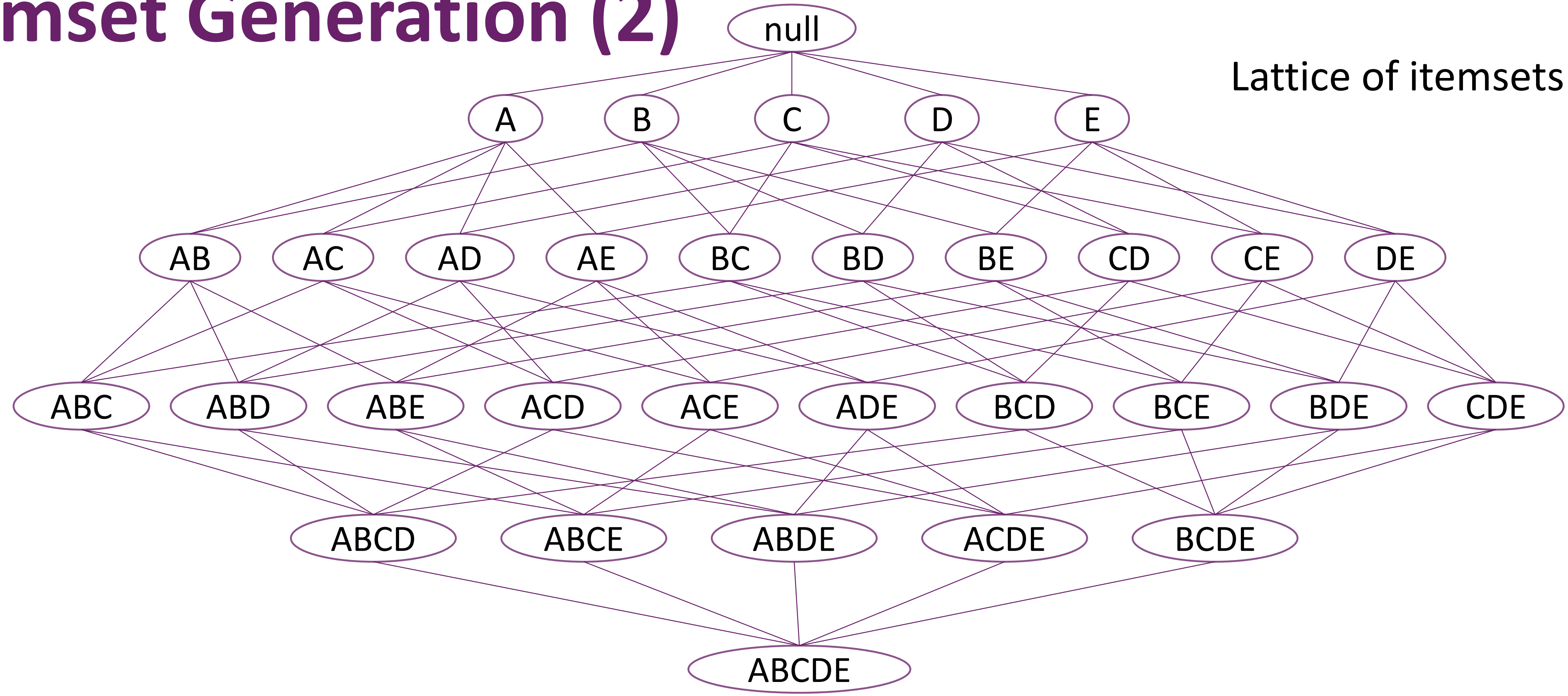
w

List of Candidates



- Match each transaction against every candidate
  - Complexity ~ O(|D|Mw) — Expensive since $M = 2^N$

Itemset Generation (2)

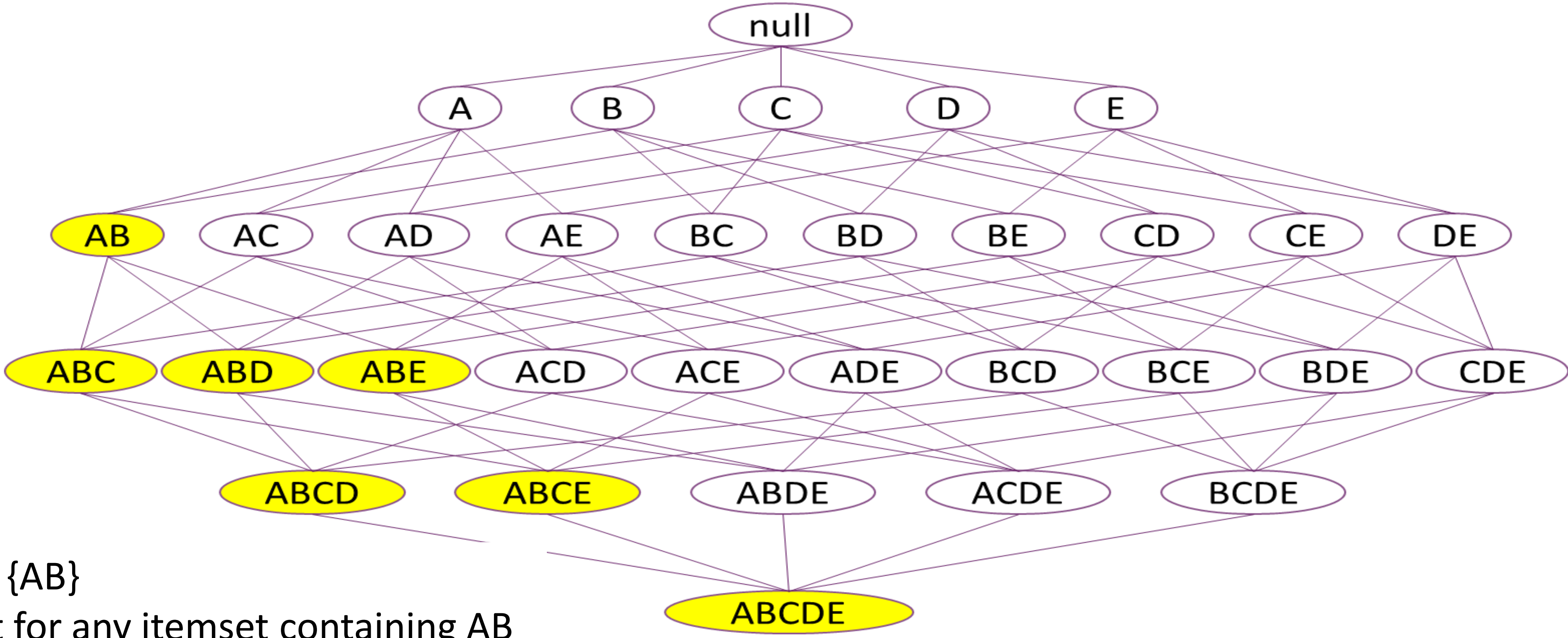


Lattice of itemsets

Given N items, there are 2^N candidate itemsets

Reduce Number of Candidates

- Ensures itemset support ≥ minimum support
- **Downward closure property**
  - any subsets of a frequent itemset are also frequent itemsets
  - any supersets of an infrequent itemset are also infrequent itemsets



If insufficient support for {AB}
then insufficient support for any itemset containing AB

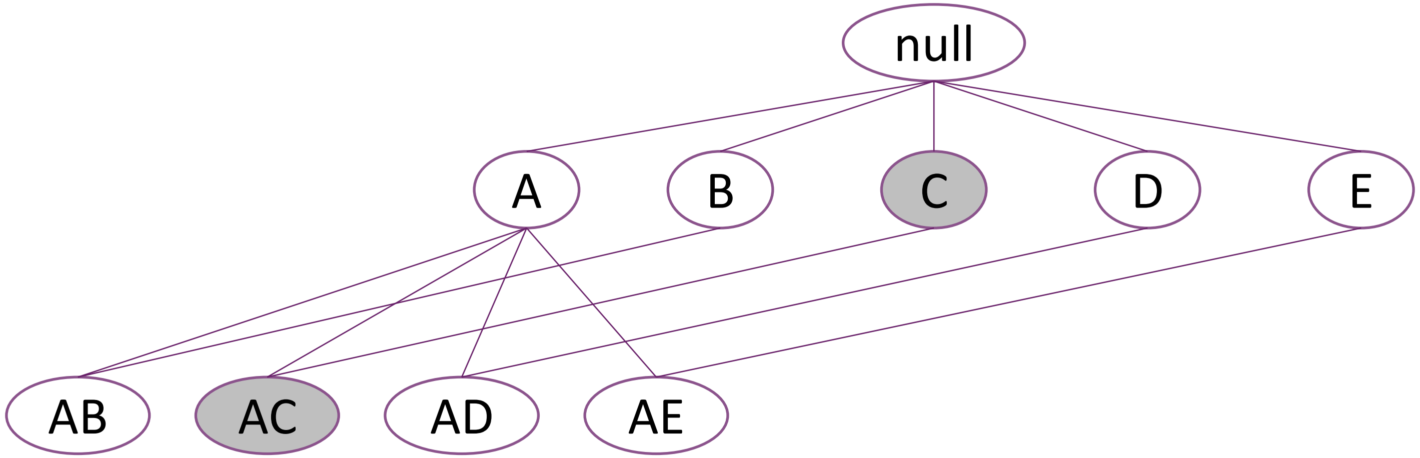
Itemsets Efficiently Generated

- Generate one-itemsets (easy)
  - remove those without minimum support (coverage)
- Generate two-itemsets from pairs of one-itemsets
  - cannot miss a frequent two-itemset
    - if (A B) is frequent itemset, then (A) and (B) are too!
  - remove those without minimum support (coverage)
- Compute k-itemset by *merging* (k-1)-itemsets
  - cannot miss a frequent k-itemset
    - if X is frequent k-itemset, then all (k-1)-itemsubsets of X are also frequent
  - remove those without minimum support (coverage)

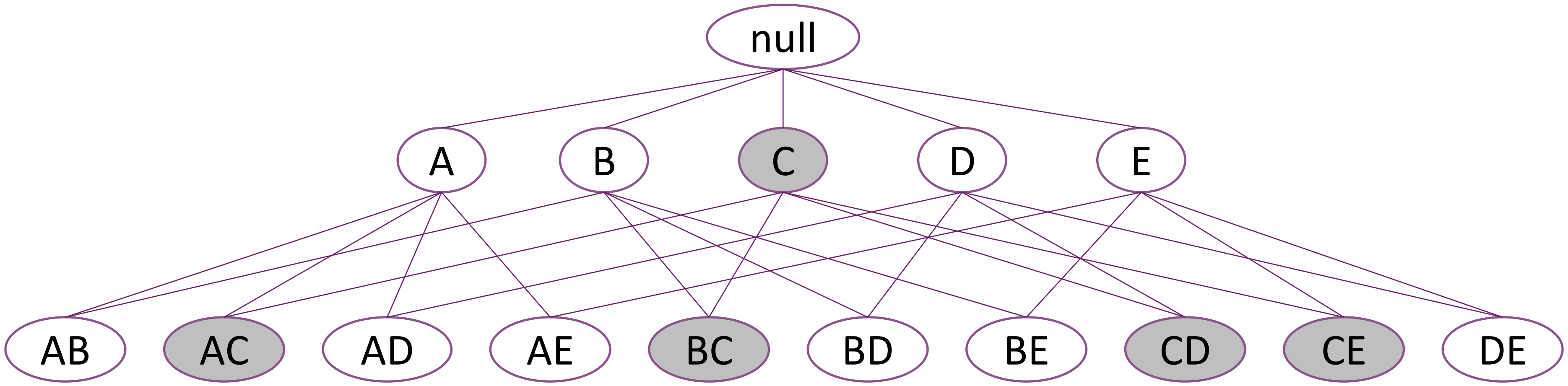
Candidate Itemsets

Assume C is infrequent (not enough support)

Then using C to generate larger itemsets is pointless

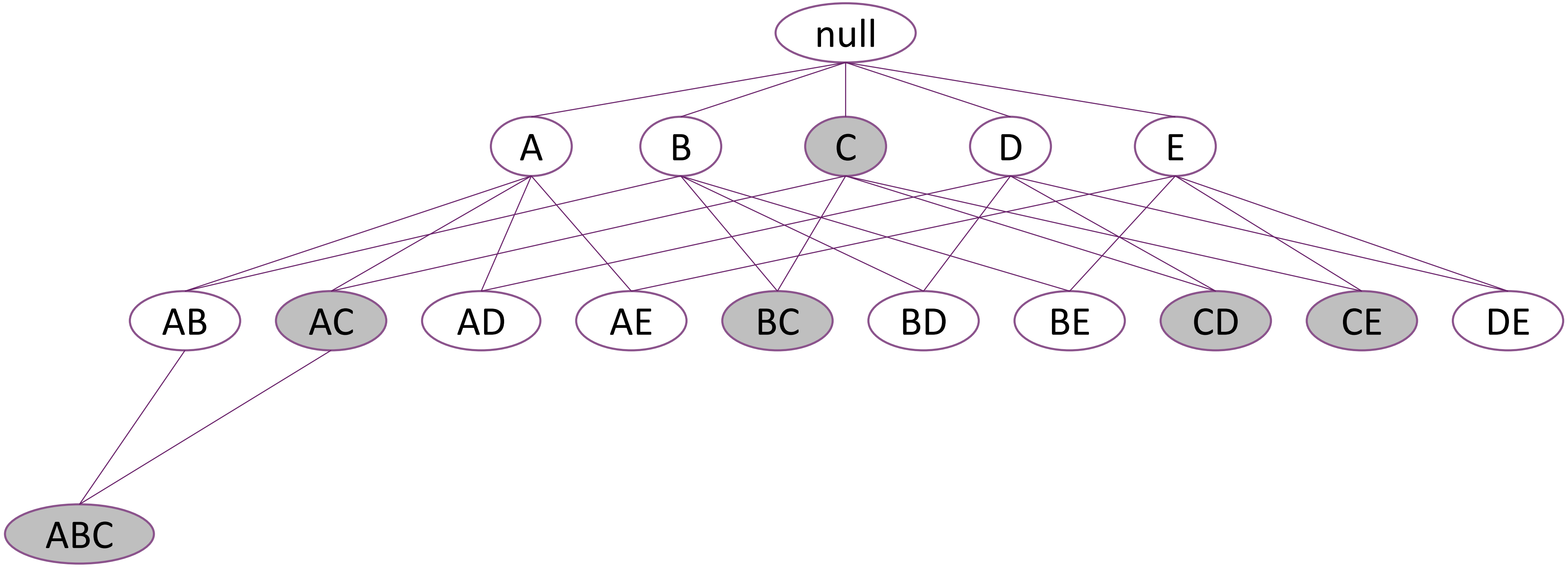


Candidate Itemsets (2)



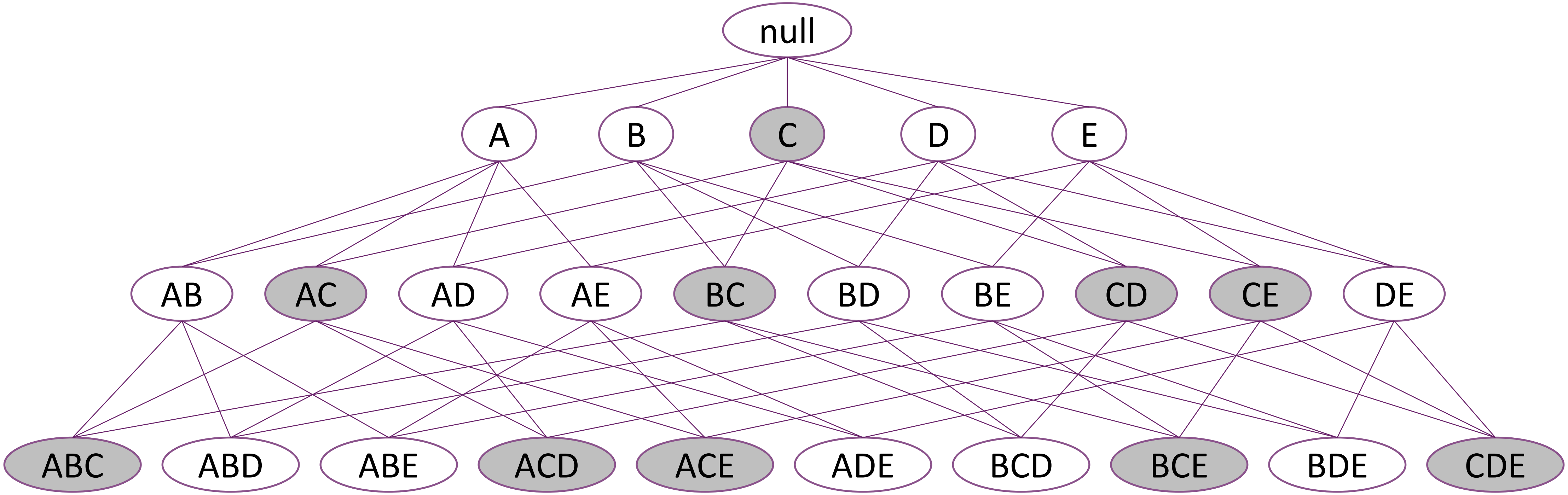
Greyed nodes containing pairs of items are NOT generated as they will be too infrequent (not enough support).

Candidate Itemsets (3)



{ABC} will not be generated, as {AC} was not generated.

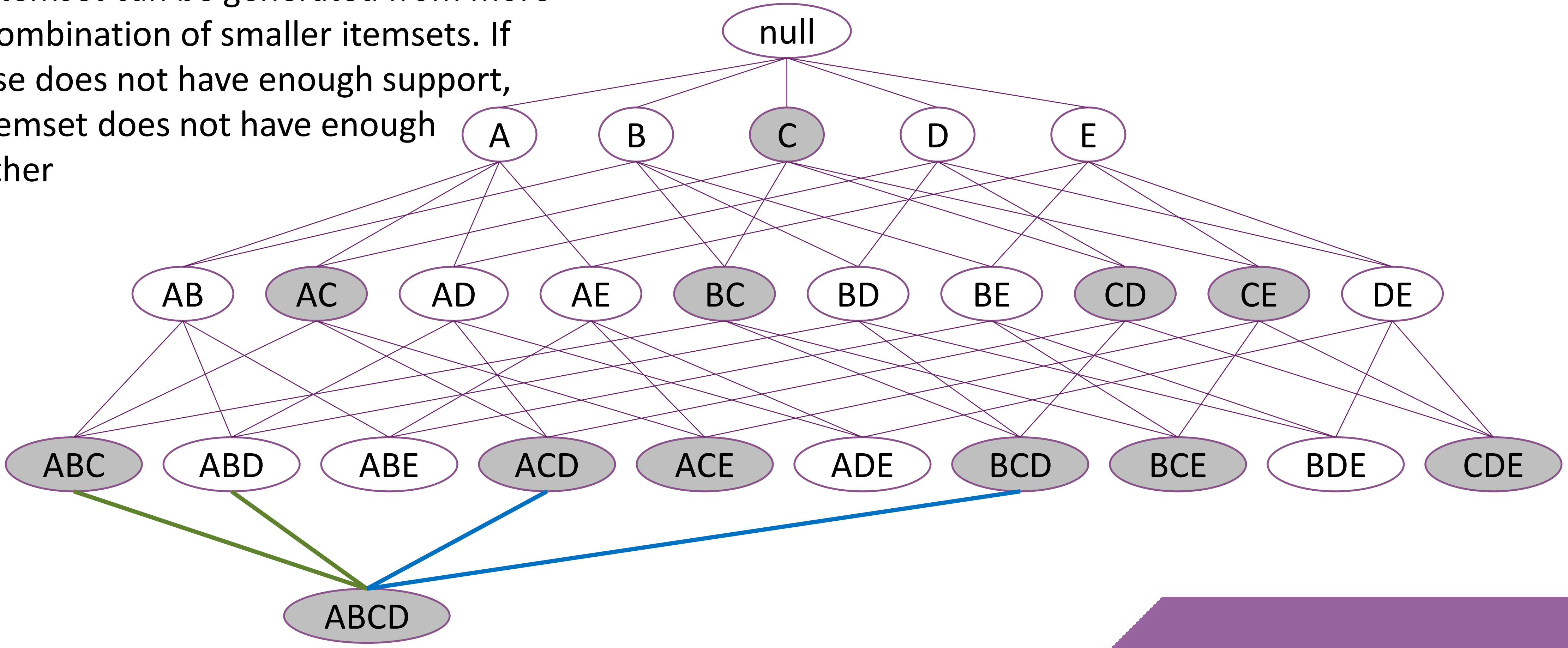
Itemsets Efficiently Generated (2)



Greyed nodes containing pairs or triplets of items are NOT generated as they will be too infrequent (not enough support).

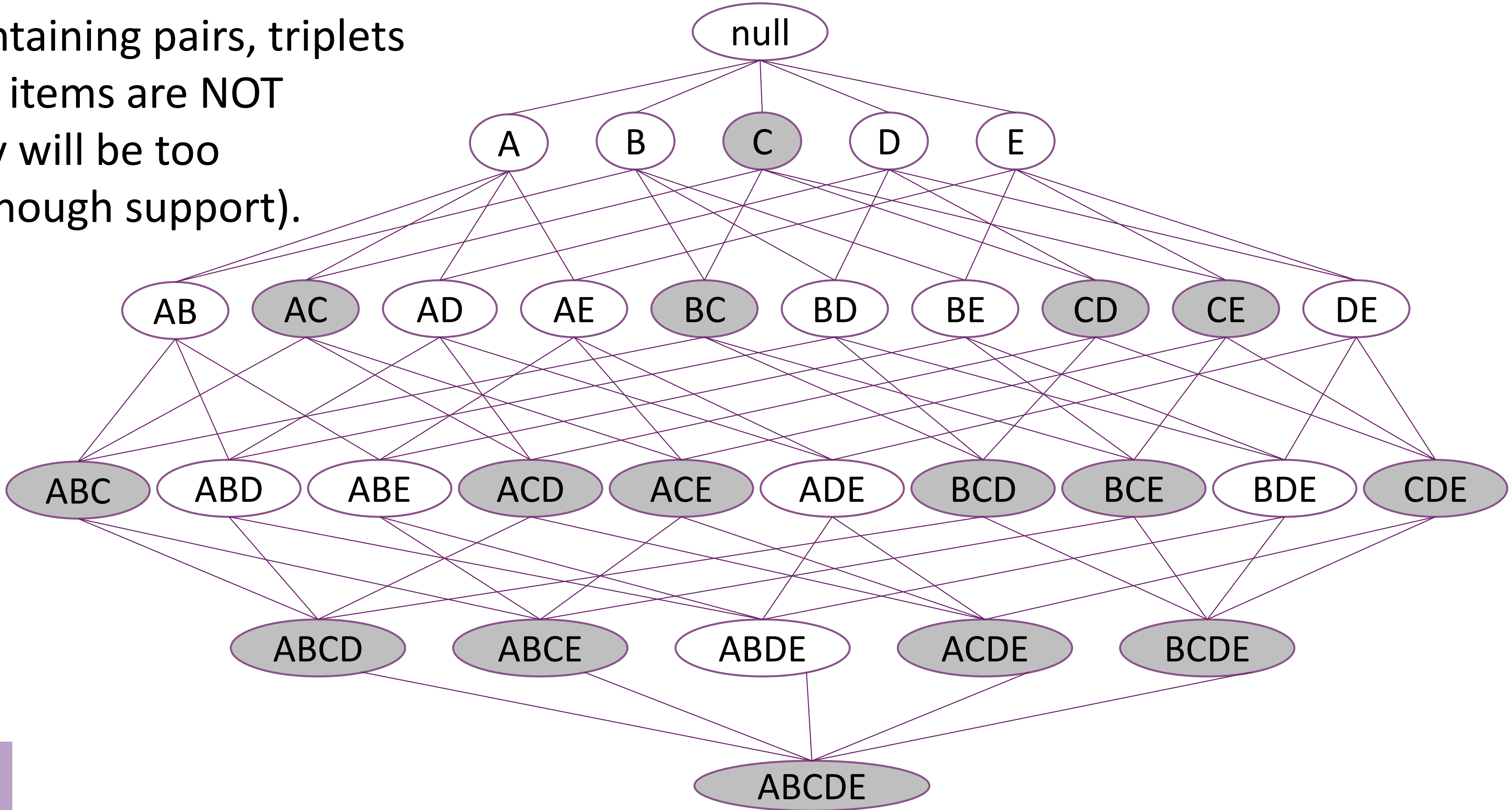
Itemsets Efficiently Generated (3)

The same itemset can be generated from more than one combination of smaller itemsets. If one of those does not have enough support, the later itemset does not have enough support either

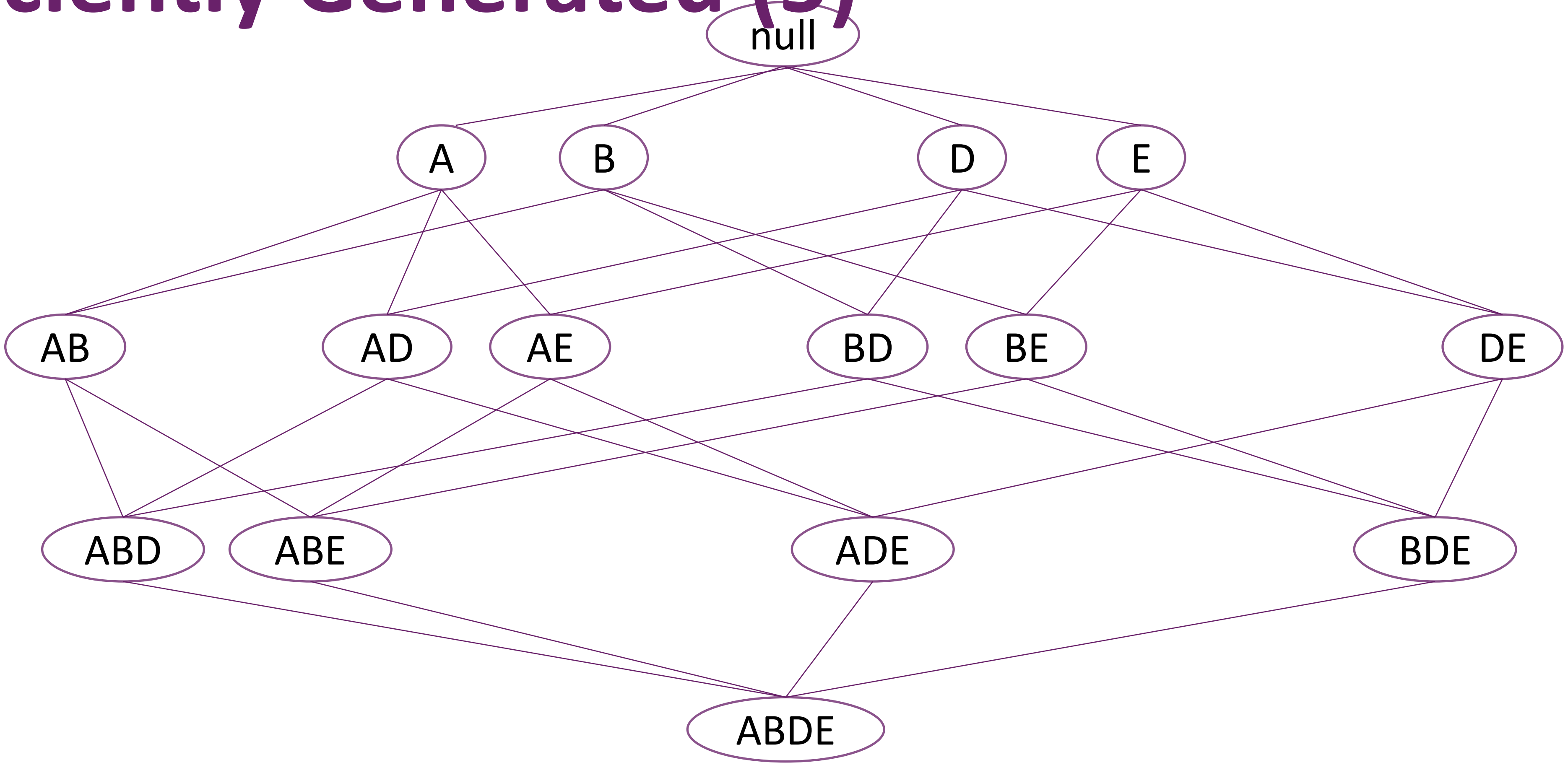


Itemsets Efficiently Generated (4)

Greyed nodes containing pairs, triplets or quadruplets of items are NOT generated as they will be too infrequent (not enough support).



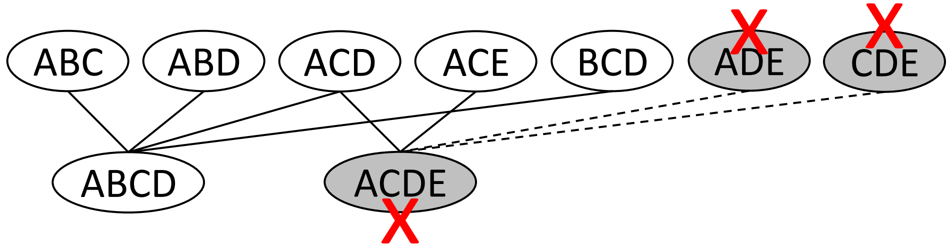
Itemsets Efficiently Generated (5)



The (much reduced!) resulting candidate itemsets if C is infrequent.

Example: Itemsets Efficiently Generated

- Given: three-itemsets (lexicographically ordered!)
  - (A B C), (A B D), (A C D), (A C E), (B C D)
- Merge those with same items except last
  - candidate four-itemsets
    - (A B C D) from (A B C) and (A B D)
    - (A C D E) from (A C D) and (A C E)
- Prune
  - remove those without minimum support
    - remove (A C D E)
      - (A D E) and (C D E) insufficient support
- Single four-itemset
  - (A B C D)



Generating Rules Efficiently …

- Step1: Generate itemsets efficiently
  - with specified minimum support (coverage)

- Step 2: Determine rules efficiently
  - that have specified minimum confidence (accuracy)

Rule Generation

Given a frequent itemset F find subsets $f \subset F$ such that

$f \Rightarrow F \setminus f$ satisfies minimum confidence requirement

- Frequent itemset {A,B,C,D}
- Candidate rules
  - $ABCD \Rightarrow \varnothing$,
  - $ABC \Rightarrow D$, $ABD \Rightarrow C$, $ACD \Rightarrow B$, $BCD \Rightarrow A$,
  - $AB \Rightarrow CD$, $AC \Rightarrow BD$, $AD \Rightarrow BC$, $BC \Rightarrow AD$, $BD \Rightarrow AC$, $CD \Rightarrow AB$,
  - $A \Rightarrow BCD$, $B \Rightarrow ACD$, $C \Rightarrow ABD$, $D \Rightarrow ABC$,
  - $\varnothing \Rightarrow ABCD$

The set of all item in F excluding f

- If |F| = N, then **$2^N - 1$** candidate association rules
  - ignoring $\{F\} \Rightarrow \varnothing$

Rule Generation Efficiently

In general, confidence does not obey the downward closure property

- $conf(ABC \Rightarrow D)$ can be larger or smaller than $conf(AB \Rightarrow D)$

But confidence of rules generated from the same itemset does have this property

- Itemset {A,B,C,D}
  - $conf(ABC \Rightarrow D) \geq conf(AB \Rightarrow CD) \geq conf(A \Rightarrow BCD)$
  - $conf(ABC \Rightarrow D) \geq conf(AC \Rightarrow BD) \geq conf(C \Rightarrow ABD)$
  - $conf(ABC \Rightarrow D) \geq conf(BC \Rightarrow AD) \geq conf(B \Rightarrow ACD)$

Conjunction fallacy $\mathrm{conf}(ABC \Rightarrow D) \geq \mathrm{conf}(AB \Rightarrow CD) \geq \mathrm{conf}(A \Rightarrow BCD)$

Linda is 31 years old, single, outspoken, and very bright. She majored in philosophy. As a student, she was deeply concerned with issues of discrimination and social justice, and also participated in anti-nuclear demonstrations.

Which is more probable?

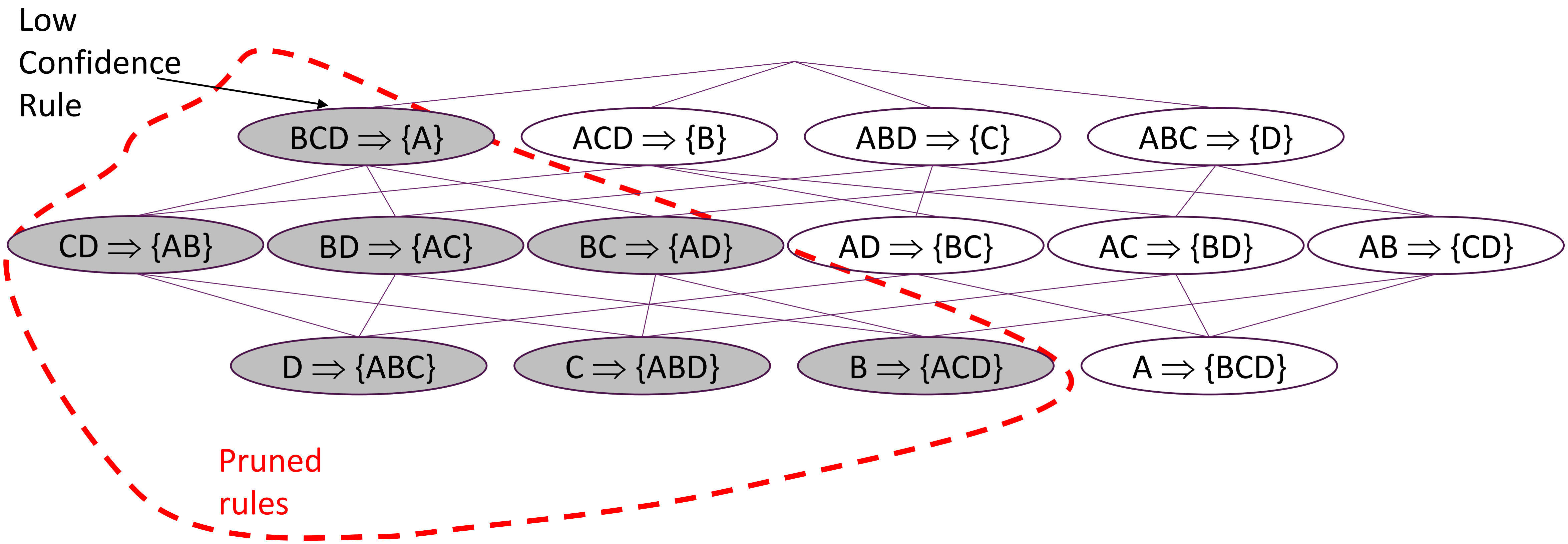
(a) Linda is a bank teller.

(b) Linda is a bank teller and is active in the feminist movement.

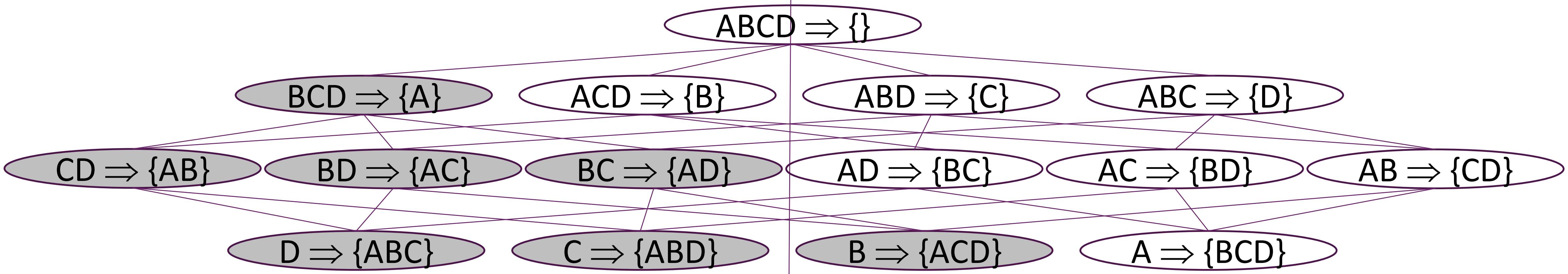
Tversky and Kahnemann: [Conjunction fallacy - Wikipedia]

Apriori Rule Generation

Lattice of rules



Apriori – generating rules



Rule Generation Efficiently

Build rules with (c+1)-consequents from rules with c-consequents

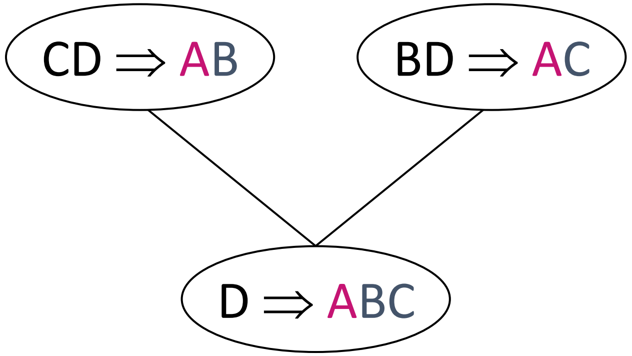
- (c+1)-consequent rule meets confidence requirement only if all corresponding c-consequent rules do

Resulting algorithm similar to procedure for large itemsets

Apriori Rule Generation

Candidate rule is generated by joining two rules which:

- are from the same itemset
- share the same prefix in the rule consequent



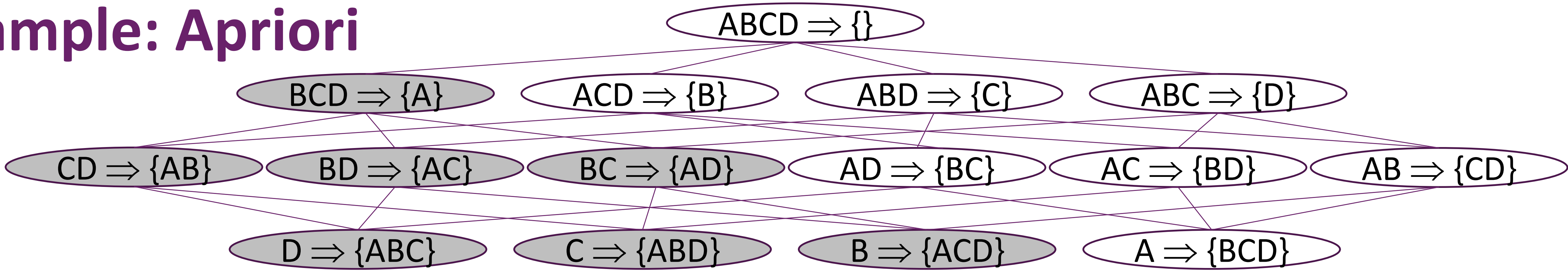
Joining (CD ⇒ AB, BD ⇒ AC)

- produces the candidate rule D ==> ABC

Prune rule D ⇒ ABC

- if it does not have minimum confidence

Example: Apriori



- To find 100% confidence rules from Itemset (A B C D)
- Suppose support for itemsets is
  - 6 (50%): (A B C D) (A B C) (A B D) (A C D) (A B) (A C) (A D)
  - 8 (75%): (B C D)
- Candidate 1-consequent rules from (A B C D)
  - ABC ⇒ D (6/6) ABD ⇒ C (6/6) ACD ⇒ B (6/6) BCD ⇒ A (6/8 <100%)
- Prune 1-consequent rules ABC ⇒ D (6/6) ABD ⇒ C (6/6) ACD ⇒ B (6/6)

Example: Apriori (cont)

Pruned 1-consequent rules:
ABC $\Rightarrow$ D   ABD $\Rightarrow$ C   ACD $\Rightarrow$ B
Support 6 (50%): (A B C D) (A B) (A C) (A D)

- Build candidate 2-consequent rules
  - AB $\Rightarrow$ CD (6/6)   AC $\Rightarrow$ BD (6/6)   AD $\Rightarrow$ BC (6/6)
- Prune 2-consequent rules
  - AB $\Rightarrow$ CD (6/6)   AC $\Rightarrow$ BD (6/6)   AD $\Rightarrow$ BC (6/6)
- Build 3-consequent rules from 2-consequent rules
  - join (AC $\Rightarrow$ BD, AD $\Rightarrow$ BC) to give A $\Rightarrow$ BCD
    - check other subset OK for confidence: AB $\Rightarrow$ CD
    - other candidates fail subset test
      - B $\Rightarrow$ ACD C $\Rightarrow$ ABD D $\Rightarrow$ ABC   (so cannot have minimum confidence)
- Prune 3-consequent rules
  - check A $\Rightarrow$ BCD for confidence

Problems

Standard format very inefficient for market basket data

- attributes represent items in a basket
- most items are usually missing
    - sparse datafiles

Confidence is not necessarily best measure

- milk occurs in almost every supermarket transaction
- other measures have been devised
    - lift measures gain in accuracy over default rule (e.g. default is everyone buys milk)

Apriori

- Given number of high support rules desirable?
  - maintain required minimum confidence (accuracy)
- Choose high desired support and generate rules
- If not enough rules repeatedly
  - Reduce minimum support (coverage)
  - Generate additional rules
- To ensure that you generate all rules of sufficient support and minimum confidence
  - Choose a large number of rules (more than those generated)

Applications

Market basket analysis

- Identification of associations between products in shopping trolleys, i.e., which products are frequently bought together
- Supermarkets gain understanding of customer shopping habits
  - Can plan product location within supermarket
  - Can do targeted marketing

Churn analysis and selective marketing - Telecoms

- Identification of behaviours and demographics of customers who are likely/unlikely to switch to other companies
- Selection of customer groups who are likely to buy an offering

Applications (2)

Stock market analysis

- Identifying **link** between individual stocks, or between stocks and economic factors
- Can help stock traders select interesting stocks and improve trading strategies

Medical diagnosis

- Discovering relationships between symptoms, test results and illness
- Can be used for diagnosis or treatment support

Applications (3)

Credit risk

- Identification of attributes of customers likely to default on payments.
- Used to assess loan or credit card applications

Health informatics – knowledge discovery

- E.g. Relationship between family medical history, medical issues and lifestyle

Census data **correlations**

Network traffic analysis

Detection of malware

Summary

- Itemsets capture frequently occurring combinations
- Rules rearrange items around $\Rightarrow$
- Apriori efficiency from downward closure
  - Generate frequent supersets from frequent subsets
  - Generate high confidence rules from supersets of consequents of high confidence rules from same itemset
- Apriori iterates
  - through high support itemsets first