Frequent itemset generation

Brute Force Algorithms

Procheta Sen

UNIVERSITY OF LIVERPOOL

Association rule generation framework

Phase 1: generate all frequent itemsets for the given frequency threshold f

- Bruteforce algorithm
- Apriori algorithm

Phase 2: from the frequent itemsets, generate the association rules at the given confidence threshold c

- For each frequent item set I:
  - partition I into all possible pairs of subsets (X, Y) such that $Y = I - X$ and $X \cup Y = I$;
  - compute the confidence of the rule $X \Rightarrow Y$. If it is at least c, store the rule $X \Rightarrow Y$.

Brute Force Algorithm

- Let U be the universe of items and $d = |U|$
- There are $2^d - 1$ distinct non-empty subsets of U (i.e. itemsets)
- Each of these itemsets is a **candidate** of being frequent (i.e. candidate itemset)

Brute Force Algorithm (universe of items U, dataset $\mathscr{D}$, frequency threshold f)

- For every non-empty subset I of U
  - Compute support of I
  - If $\mathrm{sup}(I) \geq f$, then add I to the family of frequent itemsets

Brute Force Algorithm

Brute Force Algorithm (universe of items U, dataset $\mathscr{D}$, frequency threshold f)

- For every non-empty subset I of U
  - Compute support of I
  - If $\text{sup}(I) \geq f$, then add I to the family of frequent itemsets

Major issue: exponential time complexity. If $|U| = 1000$, then there are a total of $2^{1000} > 10^{300}$ candidate itemsets.

Pruning the Search Space

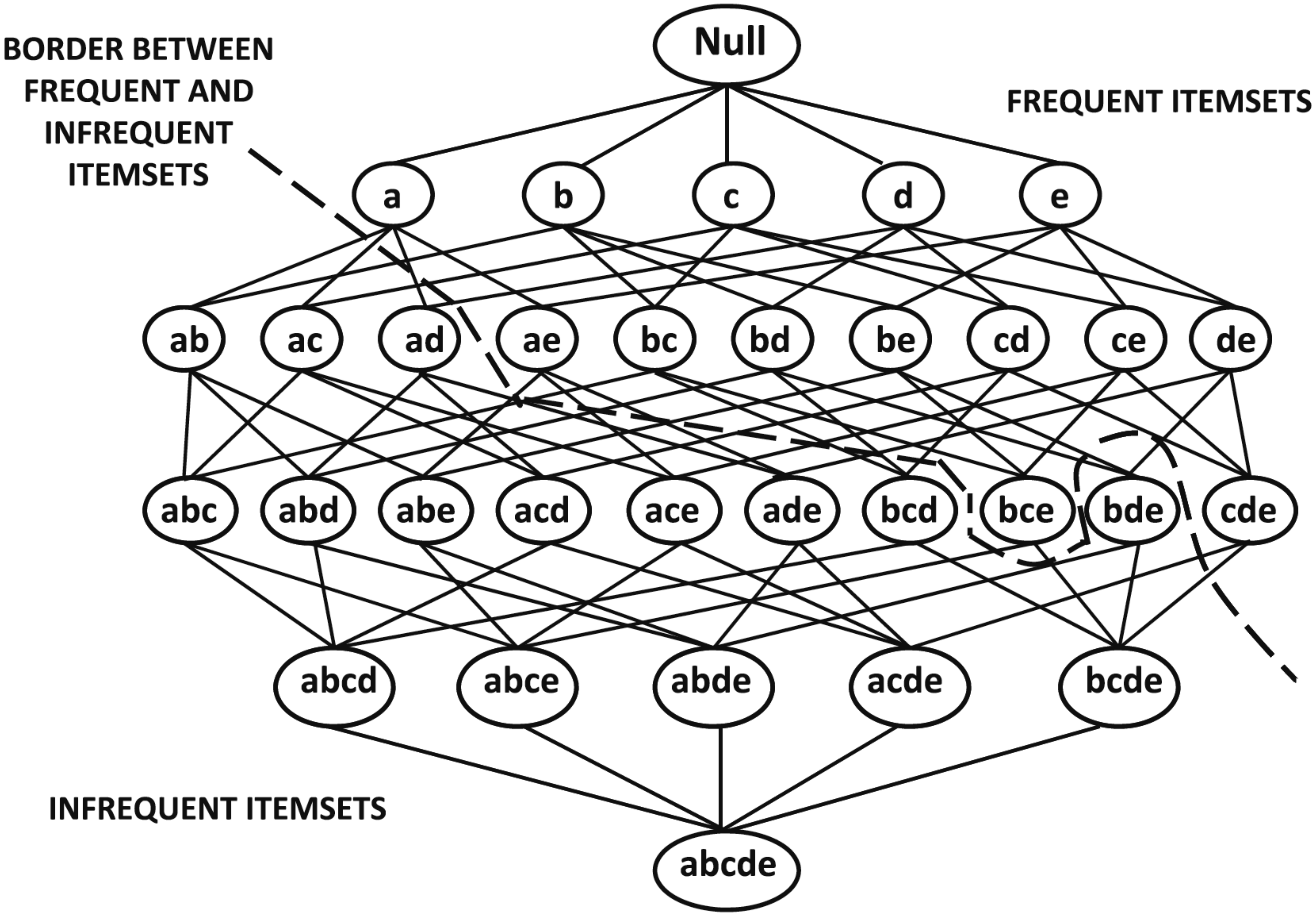
Downward Closure Property

Every subset of a frequent itemset is also frequent.

Definition: A k-itemset is an itemset that contains exactly k elements.

If no k-itemset is frequent, then no $(k+1)$-itemset is frequent.

Downward Closure Property



Improved Brute Force Algorithm

If no k-itemset is frequent, then no $(k+1)$-itemset is frequent.

Improved Brute Force Algorithm (universe of items U, dataset $\mathcal{D}$, frequency threshold f)

- For every k from 1 to $|U|$
  - For every k-itemset I
    - Compute support of I
    - If $\text{sup}(I) \geq f$, then add I to the family of frequent itemsets
  - If no k-itemset is frequent, then STOP

Improved Brute Force Algorithm

- Works much better than the plain Brute Force on sparse datasets, i.e. on datasets in which transactions have small number of items.
- Let l be the largest number of items in a transaction in the dataset.
- Then there are at most $\sum_{i=1}^{l} \binom{|U|}{i}$ candidate itemsets, which is much smaller than $2^{|U|}$, when l is much smaller than $|U|$.
- However, when $|U|$ is relatively large there are still too much candidate itemsets to consider.
  - For example, for $|U| = 1000$ and $l = 10$, the value $\sum_{i=1}^{10} \binom{|U|}{i}$ is of the order of 10^{23}.