Frequent itemset generation

The Apriori Algorithm

Procheta Sen

UNIVERSITY OF LIVERPOOL

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 $\mathscr{D}$, frequency threshold f)

- For every k from 1 to $|U|$
  - For every k-itemset I
    - Compute support of I

      The most expensive operation
      (depends on the size of the dataset)
    - If $\text{sup}(I) \geq f$, then add I to the family of frequent itemsets
  - If no k-itemset is frequent, then STOP

Main idea

The main idea of the Apriori algorithm
ignore those candidate $(k+1)$-itemsets that do not satisfy the Downward Closure Property. This candidates are not frequent.

- $\mathscr{C}_k$ the set of candidate k-itemsets
- $\mathscr{F}_k$ the set of frequent k-itemsets

The Apriori algorithm

`Apriori` `(`universe of items U, datatset $\mathscr{D}$, frequency threshold f`)`

1. Compute $\mathscr{F}_1$, i.e. the set of all frequent 1-itemsets; $\mathscr{F}_i = \varnothing$ for $i = 2,3,\ldots,d$
2. `for` $k = 2,3,\ldots,d$
3. `if` $\mathscr{F}_{k-1}$ is empty
4. `break;`
5. $\mathscr{C}_k$ = **`generate-candidates`**`(`$\mathscr{F}_{k-1}, k$`)`
6. `for every` $I \in \mathscr{C}_k$
7. `if` $\text{sup}(I) \geq f$
8. Add I to $\mathscr{F}_k$
9. `return` $\bigcup_{i=1}^{d} \mathscr{F}_i$

Assumptions

We assume that

- $U = \{1,2,\ldots,d\}$
- Each itemset is an **ordered** subset of U
- The transactions in the dataset are ordered lexicographically.

Example ($U = \{1,2,3,4,5\}$)

The dataset ordered lexicographically

$\{1,2,4\}$
$\{1,3,4\}$
$\{1,3,5\}$
$\{2,1,5\}$
$\{2,2,3\}$

Join representation of candidates

Downward Closure Property
Every subset of a frequent itemset is also frequent.

Let $I = \{j_1, j_2, \ldots, j_{k-2}, j_{k-1}, j_k\}$ be the frequent k-itemset, i.e. $I \in \mathcal{F}_k$. Then

1. for every element $j \in I$, the itemset $I - \{j\}$ is frequent $(k-1)$-itemset, i.e. $I - \{j\} \in \mathcal{F}_{k-1}$;
2. in particular, I can be represented as the **union** (also called **join**) of the following two $(k-1)$ -itemsets from $\mathcal{F}_{k-1}$:
   1. $\{j_1, j_2, \ldots, j_{k-2}, j_{k-1}\}$, and
   2. $\{j_1, j_2, \ldots, j_{k-2}, j_k\}$

Hence the itemset I can belong to $\mathcal{F}_k$ only if it can be represented as the union of two itemsets from $\mathcal{F}_{k-1}$

`generate-candidates` (frequent itemsets $\mathscr{F}$, size of itemsets k)

Assume that the itemsets in $\mathscr{F}$ are ordered lexicographically

$\mathscr{C} = \varnothing$

Join phase

`for each` $I \in \mathscr{F}$

  Let $I = \{j_1, j_2, \ldots, j_{k-2}, j_{k-1}\}$, such that $j_1 < j_2 < \ldots < j_{k-1}$

  `for` $j = j_{k-1}+1, j_{k-1}+2, \ldots, d$

    $I' = \{j_1, j_2, \ldots, j_{k-2}, j\}$

    `if` $I' \in \mathscr{F}$

      Add $\{j_1, j_2, \ldots, j_{k-2}, j_{k-1}, j\}$ to $\mathscr{C}$

Prune phase

`for each` $I \in \mathscr{C}$

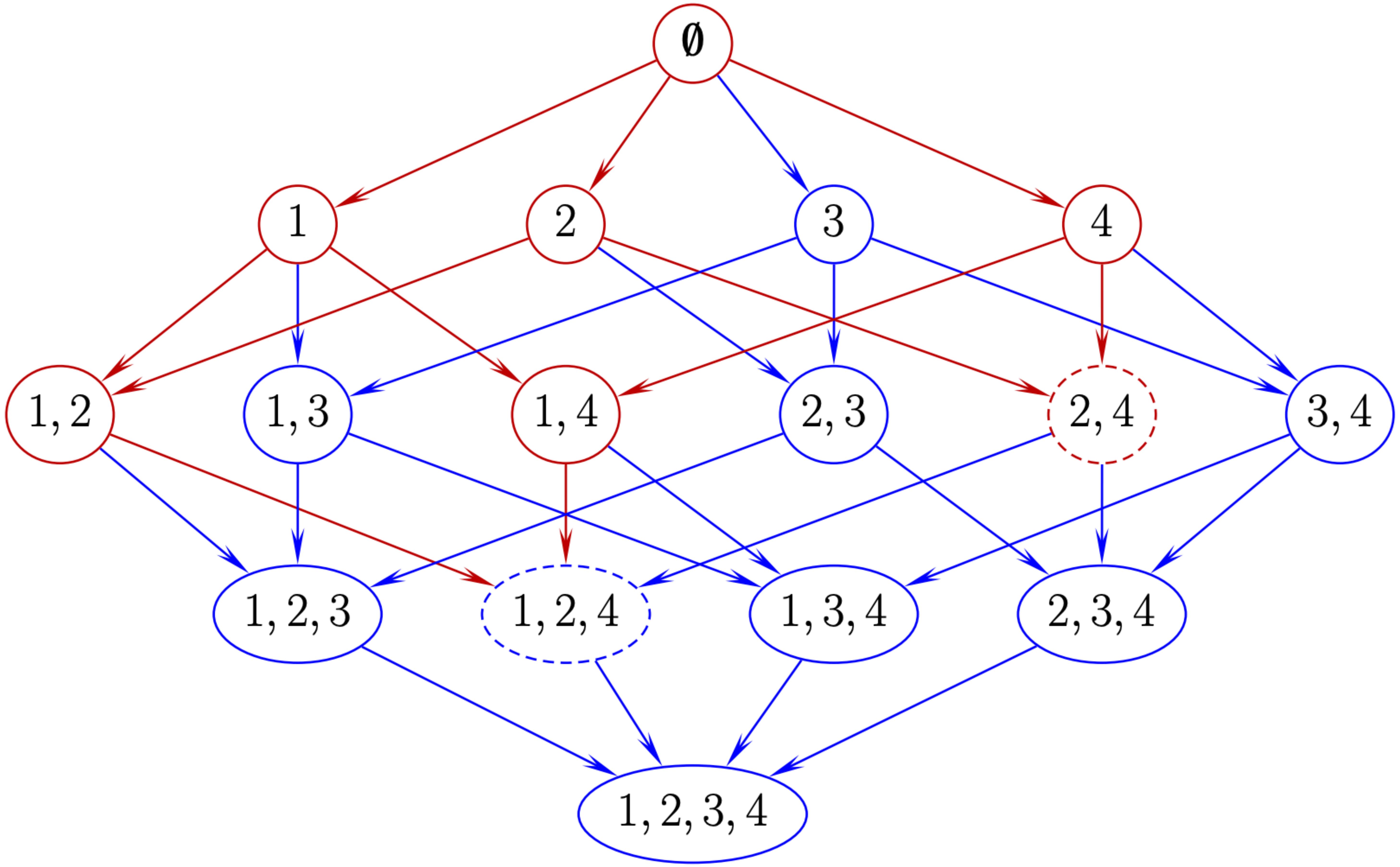
  `for each` $j \in I$

    `if` $I - \{j\} \notin \mathscr{F}$

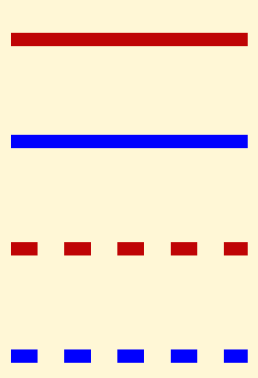
      Remove I from $\mathscr{C}$; `break`

`return` $\mathscr{C}$

Example



∅
1
2
3
4
1, 2
1, 3
1, 4
2, 3
2, 4
3, 4
1, 2, 3
1, 2, 4
1, 3, 4
2, 3, 4
1, 2, 3, 4



Frequent itemsets
Non-frequent itemsets
Generated candidates, that are not frequent
Candidates created at Join phase, but removed at Prune phase