

# Frequent Itemsets

Stefan Schaefer, Nils Rethmeier

# Motivation

**Task:** Find items that frequently occur together.

**Example:** In a store, what *items* do people buy together?

milk & cornflakes (expected)



diapers & beer (unexpected)

# Background

## Concepts

$i_k :=$  item

$B :=$  Basket, a tuple  
of items  $i_k$

$I :=$  itemset

$I = \{i_1, i_2, \dots, i_n\}$

$n :=$  set size

## Example

(milk, corn flakes, bread, eggs)  
(milk, corn flakes, beer, diapers, chips)  
(milk, bread, bread, butter, cheese, salad)  
(milk, corn flakes, beans, bacon)

...

$I = \{\text{milk}\}, I = \{\text{milk, corn flakes}\}$

$n = 1, \quad n = 2$

## Concepts

$s(I)$  := support/count  
of an items set  $I$

$s$  := support threshold

$I$  is a Frequent Itemset  $I_F$   
if  $s(I) \geq s$ .

## Example

$s(\{\text{milk}\}) = 4$   
 $s(\{\text{milk, corn flakes}\}) = 3$   
 $s(\{\text{milk, corn flakes, bread}\}) = 1$

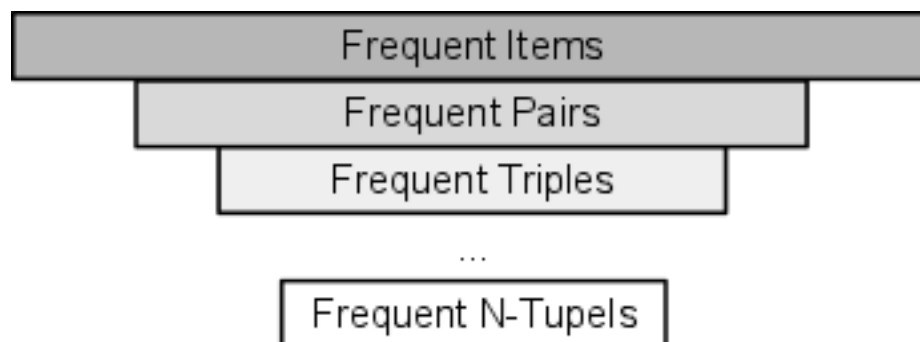
$s = 3$

$I_F = \{\text{milk}\}$   
 $I_F = \{\text{milk, corn flakes}\}$

## Concepts

### Monotonicity

If  $I$  is not frequent, then  
its supersets cannot be  
frequent either



## Example

{milk, corn flakes, bread}

is not frequent  $\rightarrow$

{milk, cornflakes, bread, eggs}

is not frequent

$$s(\{\text{milk}\}) = 4$$

$$s(\{\text{milk, corn flakes}\}) = 3$$

$$s(\{\text{milk, corn flakes, bread}\}) = 1$$

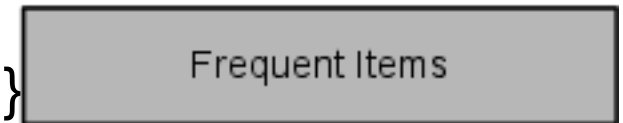
# A-Priori Algorithm

**Given:** a list of baskets with items

(milk, corn flakes, bread, eggs)  
(milk, corn flakes, beer, diapers, chips)  
(milk, bread, bread, butter, cheese, salad)  
(milk, corn flakes, beans, bacon)  
...



1<sup>st</sup> pass: find **frequent items** in the baskets.  
■ 4 x {milk}, 3 x {corn flakes}, 3 x {bread}

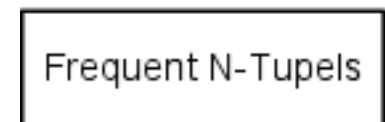


2<sup>nd</sup> pass: find **frequent pairs** based on frequent items.  
■ 3 x {milk, corn flakes}



...

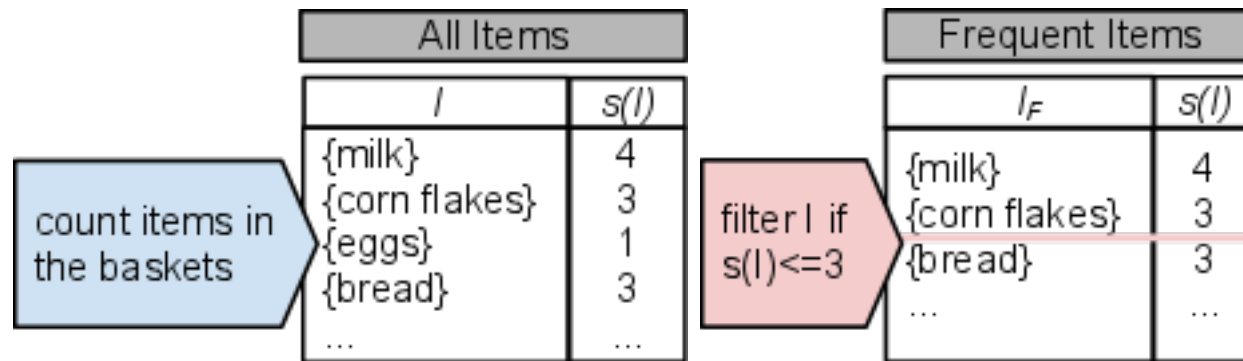
n-th pass: find frequent Itemsets of size  $n$  based on frequent  $n-1$  itemsets.



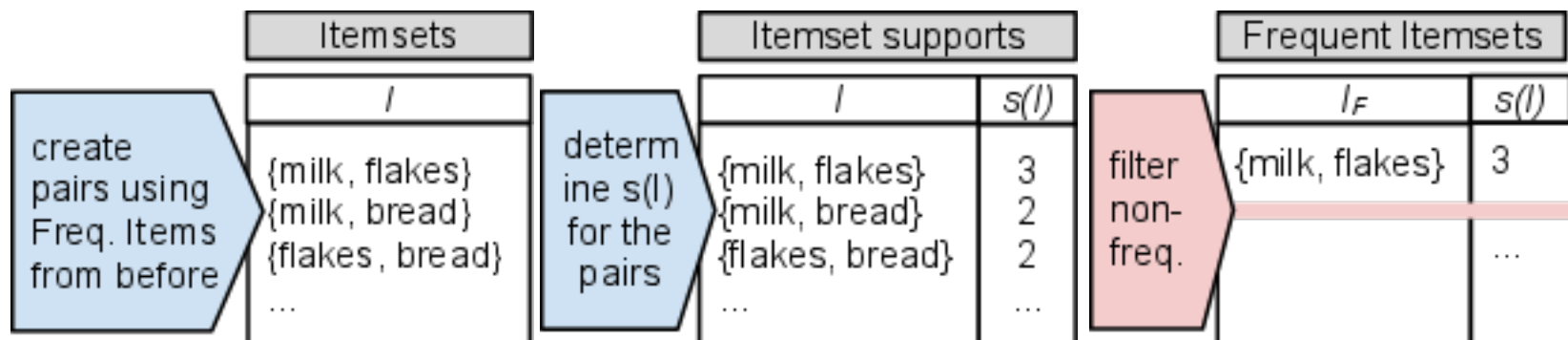
# A-Priori Algorithm

Given: a list of baskets with items

Count Itemsets  $I$  of size 1 (items) and filter result against threshold  $s$



Count Itemsets  $I$  of size 2 (pairs) and filter the results



# SON Algorithm

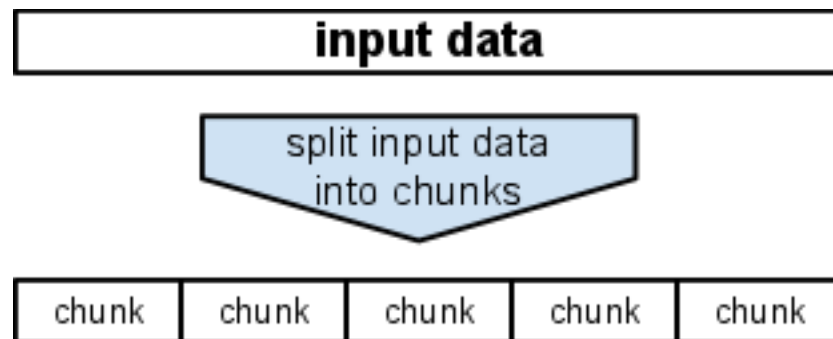
Algorithm of Savasere, Omiecinski, Navathe (*SON Algorithm*)

## Problem

- input data too large for RAM
  - swapping slows down processing

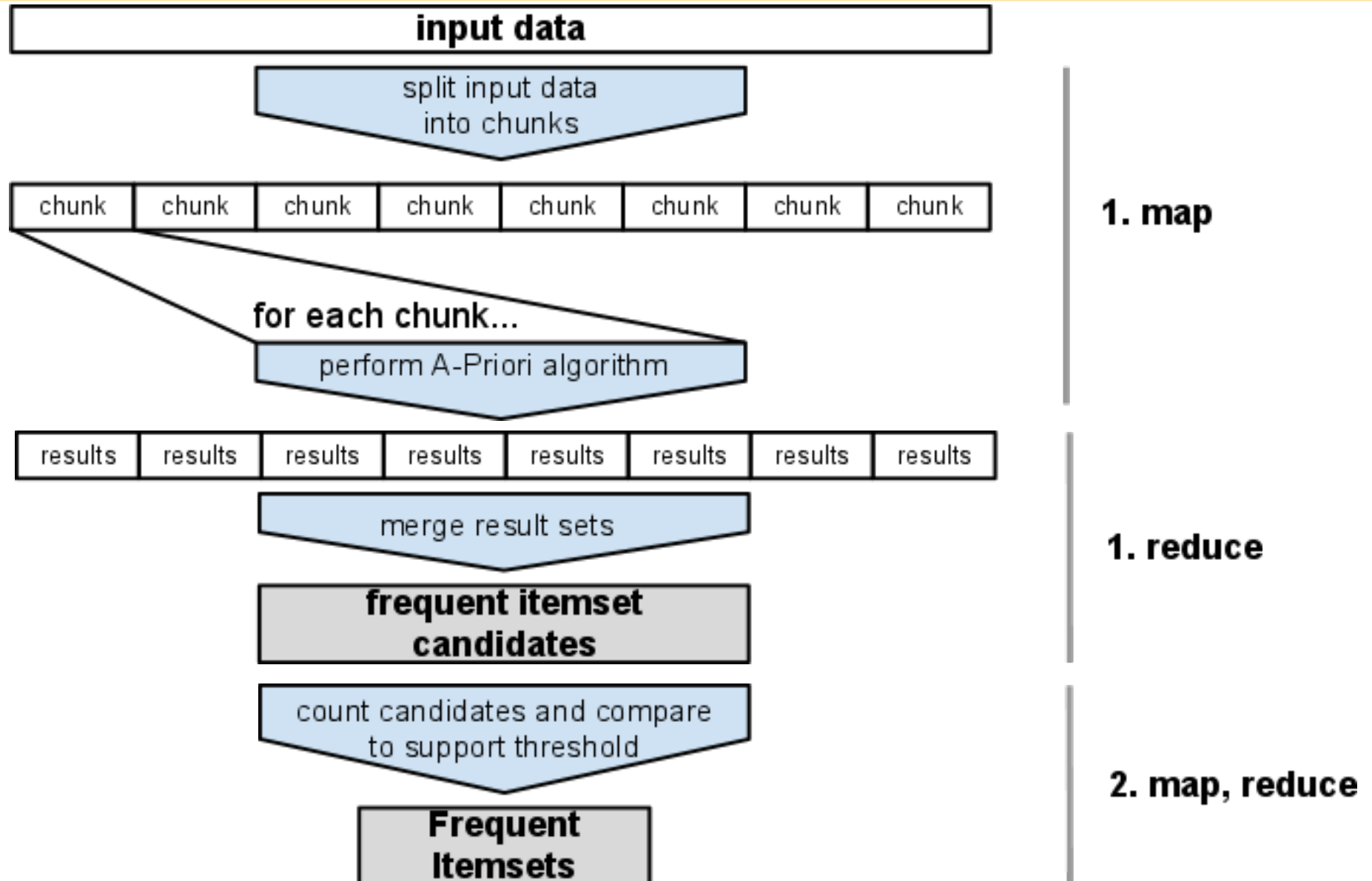
## Solution (Data parallelism)

- split input data into chunks
- process each chunk in parallel
  - Map & Reduce





# SON Algorithm



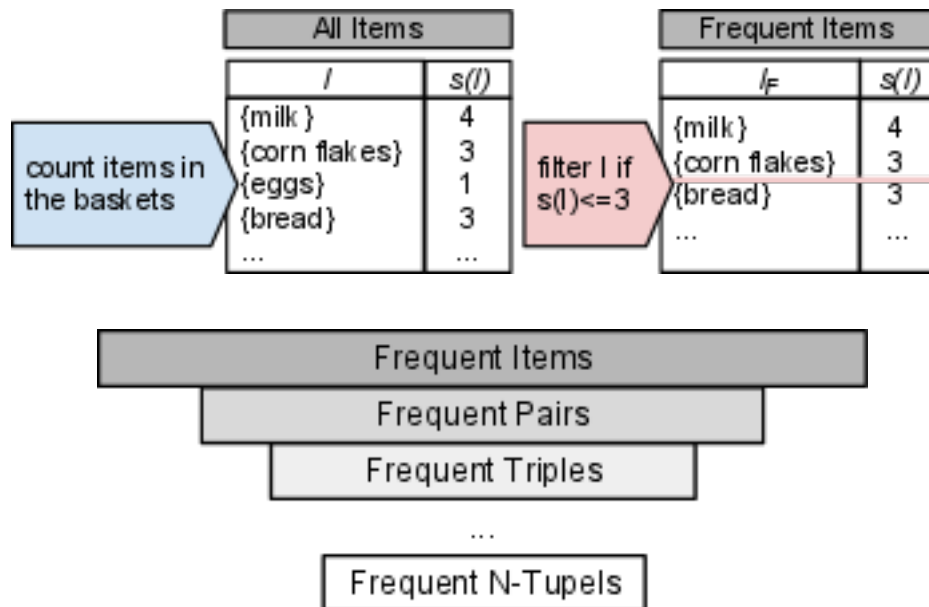
# Summary

## Frequent Itemset applications

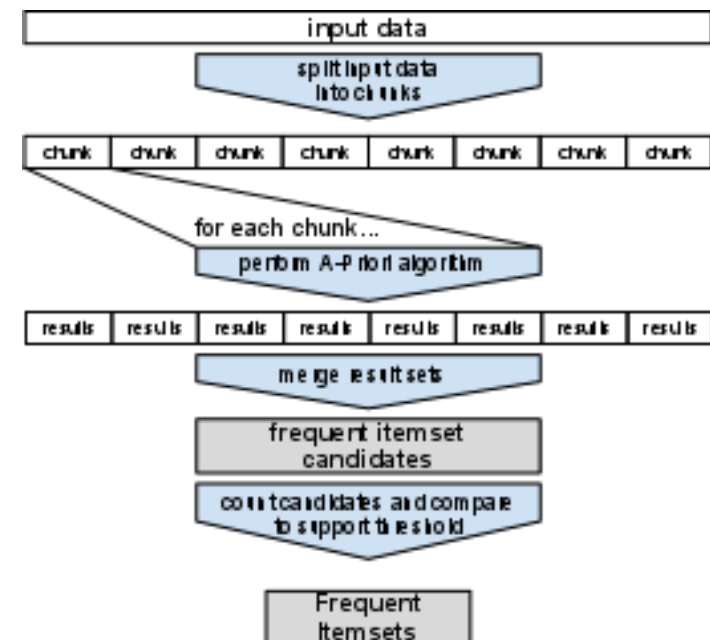
- market analysis
- plagiarism detection
- related concepts



## A Priori Algorithm



## SON Algorithm



Thank you.



# Background

## Concepts

$i_k := \text{item}$

$B := \text{Basket}$ , a tuple  
of items  $i_k$

## Example

{milk, corn flakes, bread, eggs}  
{milk, corn flakes, beer, diapers, chips}  
{milk, bread, bread, butter, cheese, salad}  
{milk, corn flakes, beans, bacon}  
...

## Concepts

$s(I)$  := support/count  
of an items set  $I$

$s$  := support threshold

$I$  is a Frequent Itemset  $I_F$   
if  $s(I) \geq s$ .

## Example

$s(\{\text{milk}\}) = 4$   
 $s(\{\text{milk, corn flakes}\}) = 3$   
 $s(\{\text{milk, corn flakes, bread}\}) = 1$

$s = 3$

$I_F = \{\text{milk}\}$   
 $I_F = \{\text{milk, corn flakes}\}$

# Background

## Concepts

$i_k :=$  item

$B :=$  Basket, a tuple of items  $i_k$

$I :=$  itemset

$I = \{i_1, i_2, \dots, i_n\}$

$n :=$  set size

## Example

{milk, corn flakes, bread, eggs}  
{milk, corn flakes, beer, diapers, chips}  
{milk, bread, bread, butter, cheese, salad}  
{milk, corn flakes, beans, bacon}  
...

$I = \{\text{milk}\}, I = \{\text{milk, corn flakes}\}$   
 $n = 1, \quad n = 2$