WARSAW UNIVERSITY OF TECHNOLOGY
DEVELOPMENT PROGRAMME

# Finding Frequent Itemsets and Association Rules with Apriori

Marzena Kryszkiewicz

HUMAN CAPITAL
HUMAN – BEST INVESTMENT!
EUROPEAN UNION
EUROPEAN SOCIAL FUND
Project is co-financed by European Union within European Social Fund

WARSAW UNIVERSITY OF TECHNOLOGY
DEVELOPMENT PROGRAMME

## Finding Frequent Itemsets

- Within each iteration *i*:
  - Determine supports of candidate itemsets of length *i*.
  - From those candidates of length *i* that turned out frequent, create candidates of length *i* + 1.

2

WARSAW UNIVERSITY OF TECHNOLOGY
DEVELOPMENT PROGRAMME

## Example: Frequent 1-Itemsets

| Tid | Items |
|---|---|
| 1 | *abce* |
| 2 | *abcef* |
| 3 | *abch* |
| 4 | *abe* |
| 5 | *acfh* |
| 6 | *bef* |
| 7 | *h* |
| 8 | *af* |

- Let minSup = 1
- Iteration 1:
  **$C_0 \rightarrow F_0$:** $\varnothing_8$
  **$C_1 \rightarrow F_1$:** $a_6$ $b_5$ $c_4$ $e_4$ $f_4$ $h_3$

3

WARSAW UNIVERSITY OF TECHNOLOGY
DEVELOPMENT PROGRAMME

## Example: Frequent 2-Itemsets

| Tid | Items |
|---|---|
| 1 | *abce* |
| 2 | *abcef* |
| 3 | *abch* |
| 4 | *abe* |
| 5 | *acfh* |
| 6 | *bef* |
| 7 | *h* |
| 8 | *af* |

- Let minSup = 1
- After iteration 1:
  **$F_1$:** $a_6$ $b_5$ $c_4$ $e_4$ $f_4$ $h_3$
- Iteration 2:
  **$C_2 \rightarrow F_2$:** $ab_4$ $ac_4$ $ae_3$ $af_3$ $ah_2$ $bc_3$ $be_4$ $bf_2$ $bh_1$ $ce_2$ $cf_2$ $ch_2$ $ef_2$ $eh_0$ $fh_1$

Itemsets found as infrequent after support calculation.

4

WARSAW UNIVERSITY OF TECHNOLOGY
DEVELOPMENT PROGRAMME

## Example: Frequent 3-Itemsets

| Tid | Items |
|---|---|
| 1 | *abce* |
| 2 | *abcef* |
| 3 | *abch* |
| 4 | *abe* |
| 5 | *acfh* |
| 6 | *bef* |
| 7 | *h* |
| 8 | *af* |

- Let minSup = 1
- After iteration 2:
  **$F_2$:** $ab_4$ $ac_4$ $ae_3$ $af_3$ $ah_2$ $bc_3$ $be_4$ $bf_2$ $ce_2$ $cf_2$ $ch_2$ $ef_2$
- Iteration 3:
  **$C_3 \rightarrow F_3$:** $abc_3$ $abe_3$ $abf_1$ $abh$ $ace_2$ $acf_2$ $ach_2$ $aef_1$ $aeh$ $afh$ $bce_2$ $bcf_1$ $bef_2$ $cef_1$ $ceh$ $cfh$

Itemsets found as infrequent as supersets of infrequent itemsets.

5

WARSAW UNIVERSITY OF TECHNOLOGY
DEVELOPMENT PROGRAMME

## Example: Frequent 4-Itemsets

| Tid | Items |
|---|---|
| 1 | *abce* |
| 2 | *abcef* |
| 3 | *abch* |
| 4 | *abe* |
| 5 | *acfh* |
| 6 | *bef* |
| 7 | *h* |
| 8 | *af* |

- Let minSup = 1
- After iteration 3:
  **$F_3$:** $abc_3$ $abe_3$ $ace_2$ $acf_2$ $ach_2$ $bce_2$ $bef_2$
- Iteration 4:
  **$C_4 \rightarrow F_4$:** $abce_2$ $acef$ $aceh$ $acfh$

6

## Example: Frequent 5-Itemsets

| Tid | Items |
|---|---|
| 1 | *abce* |
| 2 | *abcef* |
| 3 | *abch* |
| 4 | *abe* |
| 5 | *acfh* |
| 6 | *bef* |
| 7 | *h* |
| 8 | *af* |

- Let minSup = 1
- After iteration 4:
  **$F_4$:** $abce_2$
- Iteration 5:
  **$C_5$: -**

7

## Example: Found Frequent Itemsets

| Tid | Items |
|---|---|
| 1 | *abce* |
| 2 | *abcef* |
| 3 | *abch* |
| 4 | *abe* |
| 5 | *acfh* |
| 6 | *bef* |
| 7 | *h* |
| 8 | *af* |

$\varnothing_8$

$a_6\ b_5\ c_4\ e_4\ f_4\ h_3$

$ab_4\ ac_4\ ae_3\ af_3\ ah_2\ bc_3\ be_4\ bf_2\ ce_2\ cf_2\ ch_2\ ef_2$

$abc_3\ abe_3\ ace_2\ acf_2\ ach_2\ bce_2\ bef_2$

$abce_2$

- **Note:** Let *n* be the length of a longest frequent itemset.
  Apriori finds in either *n* or *n*+1 iterations all frequent itemsets.

8

## Discovery of Association Rules with AprioriRuleGen…

- Candidate rules are built from each non-empty frequent itemset.
- Let *Z* be a given non-empty frequent itemset. In iteration *i*, candidate rules of the form:

$$Z \setminus Y \rightarrow Y,$$

where $Y \subset Z$ and $|Y| = i$.

9

## Discovery of Association Rules with AprioriRuleGen

- **Property.** Let $r_1$: $Z\setminus Y \rightarrow Y$ and $r_2$: $Z\setminus Y' \rightarrow Y'$, where $Y \subset Y'$, be association rules.
  - $conf(r_1) \geq conf(r_2)$,
  - If $conf(r_1) \leq minConf$, then $conf(r_2) \leq minConf$.
- In order to reduce the number of candidate rules, *i* + 1 item consequents of candidate rules are built from *i* item consequents of strong association rules only.

10

## Example: Discovery of **AR**s…

Frequent itemsets (*minSup* = 1): $\varnothing_8$

$a_6\ b_5\ c_4\ e_4\ f_4\ h_3$

$ab_4\ ac_4\ ae_3\ af_3\ ah_2\ bc_3\ be_4\ bf_2\ ce_2\ cf_2\ ch_2\ ef_2$

$abc_3\ abe_3\ ace_2\ acf_2\ ach_2\ bce_2\ bef_2$

$abce_2$

Let *minConf* = 60%, *Z* = *abce*.

**Iteration 1:**

- Consequents of candidate rules: $\mathbf{Y}_1 = \{a, b, c, e\}$.
- Candidate rules:
  - *bce*→*a* [2, 2/2];
  - *ace*→*b* [2, 2/2];
  - *abe*→*c* [2, 2/3];
  - *abc*→*e* [2, 2/3].

Strong association rules:

*bce*→*a* [2, 2/2];
*ace*→*b* [2, 2/2];
*abe*→*c* [2, 2/3];
*abc*→*e* [2, 2/3].

11

## Example: Discovery of **AR**s…

Frequent itemsets: $\varnothing_8$

$a_6\ b_5\ c_4\ e_4\ f_4\ h_3$

$ab_4\ ac_4\ ae_3\ af_3\ ah_2\ bc_3\ be_4\ bf_2\ ce_2\ cf_2\ ch_2\ ef_2$

$abc_3\ abe_3\ ace_2\ acf_2\ ach_2\ bce_2\ bef_2$

$abce_2$

**Iteration 2 (*minConf* = 60%, *Z* = *abce*):**

- Consequents of **AR**s found in iteration 1: $\mathbf{Y}_1 = \{a, b, c, e\}$.
- Consequents of candidate rules: $\mathbf{Y}_2 = \{ab, ac, ae, bc, be, ce\}$.
- Candidate rules:
  - *ce*→*ab* [2, 2/2]; *ae*→*bc* [2, 2/3];
  - *be*→*ac* [2, 2/4]; *ac*→*be* [2, 2/4];
  - *bc*→*ae* [2, 2/3]; *ab*→*ce* [2, 2/4];

Strong association rules:

*ce*→*ab* [2, 2/2];
*bc*→*ae* [2, 2/3];
*ae*→*bc* [2, 2/3].

12

WARSAW UNIVERSITY OF TECHNOLOGY DEVELOPMENT PROGRAMME

## Example: Discovery of **AR**s…

Frequent itemsets (*minSup* = 1): $\emptyset_8$

$a_6\ b_5\ c_4\ e_4\ f_4\ h_3$

$ab_4\ ac_4\ ae_3\ af_3\ ah_2\ bc_3\ be_4\ bf_2\ ce_2\ cf_2\ ch_2\ ef_2$

$abc_3\ abe_3\ ace_2\ acf_2\ ach_2\ bce_2\ bef_2$

$abce_2$

**Iteration 3** (*minConf* = 60%, *Z* = *abce*)**:**

- Consequents of **AR**s found in iteration 2: $\mathbf{Y}_2 = \{ab, ae, bc\}$.
- Consequents of candidate rules: $\mathbf{Y}_3 = \{abe\}$.

- Candidate rules:
  - $c \rightarrow abe$ [2, 2/4];

Strong association rules:

*None*

13

WARSAW UNIVERSITY OF TECHNOLOGY DEVELOPMENT PROGRAMME

## Example: Found **AR**s

Frequent itemsets (*minSup* = 1): $\emptyset_8$

$a_6\ b_5\ c_4\ e_4\ f_4\ h_3$

$ab_4\ ac_4\ ae_3\ af_3\ ah_2\ bc_3\ be_4\ bf_2\ ce_2\ cf_2\ ch_2\ ef_2$

$abc_3\ abe_3\ ace_2\ acf_2\ ach_2\ bce_2\ bef_2$

$abce_2$

Strong association rules (*minConf* = 60%, *Z* = *abce*):

- $bce \rightarrow a$ [2, 2/2];
- $ace \rightarrow b$ [2, 2/2];
- $abe \rightarrow c$ [2, 2/3];
- $abc \rightarrow e$ [2, 2/3];
- $ce \rightarrow ab$ [2, 2/2];
- $bc \rightarrow ae$ [2, 2/3];
- $ae \rightarrow bc$ [2, 2/3].

14

WARSAW UNIVERSITY OF TECHNOLOGY DEVELOPMENT PROGRAMME

## Important Operations in Apriori and AprioriRuleGen

- An important time-consuming operation in *Apriori* is searching *i* item candidates supported by a given transaction.
- An important time-consuming operation in *AprioriRuleGen* is searching frequent *i* itemsets (candidate rule consequents) of a given frequent itemset in order to learn their supports.
- Thus, in both cases *i* item subsets of a given itemset are searched.

15

WARSAW UNIVERSITY OF TECHNOLOGY DEVELOPMENT PROGRAMME

## Usage of a Hash Tree

- A hash tree is used in order to make the identification of *i* item subsets of a given itemset efficient.

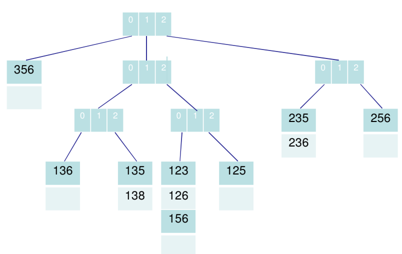
- In particular, all *i* item candidate sets are stored in a hash tree.

16

WARSAW UNIVERSITY OF TECHNOLOGY DEVELOPMENT PROGRAMME

## Example: Candidate 3-Itemsets in a Hash Tree

| itemset | represen-tation |
|---|---|
| *abc* | 123 |
| *abe* | 125 |
| *abf* | 126 |
| *ace* | 135 |
| *acf* | 136 |
| *ach* | 138 |
| *aef* | 156 |
| *bce* | 235 |
| *bcf* | 236 |
| *bef* | 256 |
| *cef* | 356 |



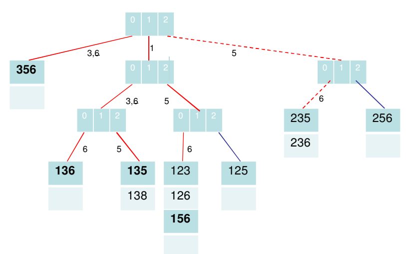

17

WARSAW UNIVERSITY OF TECHNOLOGY DEVELOPMENT PROGRAMME

## Example: Searching for Subsets in a Hash Tree

| itemset | represen-tation |
|---|---|
| *abc* | 123 |
| *abe* | 125 |
| *abf* | 126 |
| … | … |

| transac-tion | represen-tation |
|---|---|
| … | … |
| *acef* | 1356 |
| … | … |




- 4 subsets of transaction *acef* (after coding: 1356) has been found.

18

## Dealing with non-transactional data

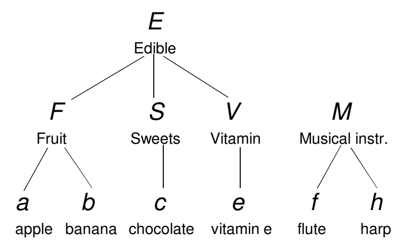
- Non-transactional data are often transformed into equivalent transactional data.





## Transactional Data Set + Taxonomies

| Tid | Items |
|---|---|
| 1 | *abce* |
| 2 | *abcef* |
| 3 | *abch* |
| 4 | *abe* |
| 5 | *acfh* |
| 6 | *bef* |
| 7 | *h* |
| 8 | *af* |

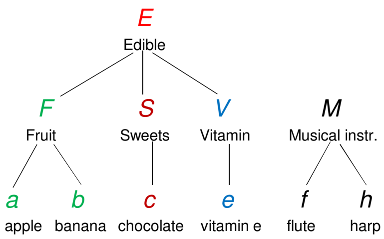







## Transactional Data Set with Taxonomies included

| Tid | Items |
|---|---|
| 1 | *abceFSVE* |
| 2 | *abcefFSVME* |
| 3 | *abchFSME* |
| 4 | *abeFVE* |
| 5 | *acfhFSME* |
| 6 | *befFVME* |
| 7 | *hM* |
| 8 | *afFME* |

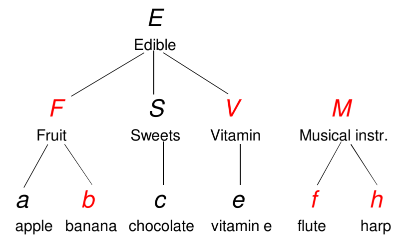







## Transactional Data Set + Taxonomies

| Tid | Items |
|---|---|
| 1 | *abce* |
| 2 | *abcef* |
| 3 | *abch* |
| 4 | *abe* |
| 5 | *acfh* |
| 6 | *bef* |
| 7 | *h* |
| 8 | *af* |








## Transactional Data Set + Negated Items

| Tid | Items |
|---|---|
| 1 | *abce* |
| 2 | *abcef* |
| 3 | *abch* |
| 4 | *abe* |
| 5 | *acfh* |
| 6 | *bef* |
| 7 | *h* |
| 8 | *af* |

6 positive items {*abcefh*}

6 negated items {*abcefh*}

| Tid | Items |
|---|---|
| 1 | *abcefh* |
| 2 | *abcefh* |
| 3 | *abchef* |
| 4 | *abecfh* |
| 5 | *acfhbe* |
| 6 | *befach* |
| 7 | *habcef* |
| 8 | *afbceh* |





## Transactional Data Set + Negated Items

| Tid | Items |
|---|---|
| 1 | *abce* |
| 2 | *abcef* |
| 3 | *abch* |
| 4 | *abe* |
| 5 | *acfh* |
| 6 | *bef* |
| 7 | *h* |
| 8 | *af* |

6 positive items {*abcefh*}

6 negated items {*abcefh*}

*sup*({*befh*}) = 2

*conf*({*bf*}→{*eh*}) = 2/3

*conf*({*bh*}→{*ef*}) = 2/4

| Tid | Items |
|---|---|
| 1 | *abcefh* |
| 2 | *abcefh* |
| 3 | *abchef* |
| 4 | *abecfh* |
| 5 | *acfhbe* |
| 6 | *befach* |
| 7 | *habcef* |
| 8 | *afbceh* |

## Transactional Data Set + Negated Items

| Tid | Items |
|---|---|
| 1 | abcefh |
| 2 | abcefh |
| 3 | abchef |
| 4 | abecfh |
| 5 | acfhbe |
| 6 | befach |
| 7 | habcef |
| 8 | afbceh |

| Tid | Items |
|---|---|
| 1 | 1 2 3 4 11 12 |
| 2 | 1 2 3 4 5 12 |
| 3 | 1 2 3 6 11 12 |
| 4 | 1 2 4 10 11 12 |
| 5 | 1 3 5 6 8 10 |
| 6 | 2 4 5 7 9 12 |
| 7 | 6 7 8 9 10 11 |
| 8 | 1 5 8 9 10 12 |

| Item | a | b | c | e | f | h | a | b | c | e | f | h |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| item id | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |

## Relational Data → Transactional Data

| Height | Colour | Grade |
|---|---|---|
| tall | green | 5 |
| short | black | 4 |
| short | green | 4 |

| Tid | Items |
|---|---|
| 1 | {1, 3, 6} |
| 2 | {2, 4, 5} |
| 3 | {2, 3, 5} |

| Item | (H=tall) | (H=short) | (C=green) | (C=black) | (G=4) | (G=5) |
|---|---|---|---|---|---|---|
| item id | 1 | 2 | 3 | 4 | 5 | 6 |
| attribute | 1 | 1 | 2 | 2 | 3 | 3 |

26

## Relational Data → Transactional Data

| Height | Colour | Grade |
|---|---|---|
| tall | green | 5 |
| short | black | 4 |
| short | green | 4 |

| Tid | Items |
|---|---|
| 1 | {1, 3, 6} |
| 2 | {2, 4, 5} |
| 3 | {2, 3, 5} |

| Item | (H=tall) | (H=short) | (C=green) | (C=black) | (G=4) | (G=5) |
|---|---|---|---|---|---|---|
| item id | 1 | 2 | 3 | 4 | 5 | 6 |
| attribute | 1 | 1 | 2 | 2 | 3 | 3 |

{2} → {5} [2, 2/2]. So, (H=short) → (G=4) [2, 100%].

{3} → {5} [1, 1/2]. So, (C=green) → (G=4) [1, 50%].

{2,3} → {5} [1, 1/1]. So, (H=short)∧(C=green) → (G=4) [1, 100%].

## References…


– Agrawal R., Imielinski T., Swami A.: Mining Associations Rules between Sets of Items in Large Databases. In: Proc. of the ACM SIGMOD Conference on Management of Data, Washington, USA (1993) 207–216
– Rakesh Agrawal, Ramakrishnan Srikant: Fast Algorithms for Mining Association Rules in Large Databases. VLDB 1994: 487-499
– Agrawal R., Mannila H., Srikant R., Toivonen H., Verkamo A.I.: Fast Discovery of Association Rules. In: Fayyad U.M., Piatetsky-Shapiro G., Smyth P., Uthurusamy R. (eds.): Advances in Knowledge Discovery and Data Mining. AAAI, CA (1996) 307–328


28

## References


– Jiawei Han, Micheline Kamber, Jian Pei: Data Mining: Concept and Techniques, The Morgan Kaufmann Series in Data Management Systems, 2011
– Kryszkiewicz, M.: Concise Representations of Frequent Patterns and Association Rules. Prace Naukowe Politechniki Warszawskiej. Elektronika 142. Publishing House of the Warsaw University of Technology (2002)
– Ashok Savasere, Edward Omiecinski, Shamkant B. Navathe: An Efficient Algorithm for Mining Association Rules in Large Databases. VLDB 1995: 432-444
– Mohammed Javeed Zaki, Karam Gouda: Fast vertical mining using diffsets. KDD 2003: 326-335


29