Data Mining ▷ Frequent Pattern Mining

Philippe Lenca et Romain Billot
philippe.lenca@imt-atlantique.fr
IMT Atlantique

Outline

A very popular topic in computer science

Since 1993...

Among the most cited across computer science

- AIS [AIS93]
- APRIORI [AS94]...top 10 in data mining [WKRQ+07]

Thousands of papers

- many improvements
- many extensions

Thousands of applications

- works for `examples`×`properties` datasets
- e.g. basket×products, patient×symptoms, vehicle×defaults

⇒ Interesting surveys [Goe03, HCXY07].

A very popular topic in computer science

Since 1993...

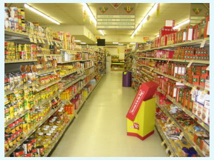
...but before

- GUHA
- CHARADE

⇒ But with APRIORI [AS94]: monotonicity property.

Roots: AIS [AIS93] & APRIORI [AS94]

Find associations between products (Supermarket basket analysis)



- which products are frequently bought together?
- do some products influence the sales of other products?

Roots: AIS [AIS93] & APRIORI [AS94]

Find associations:

- co-occurrence of properties
- applications:
  - supermarket: basket × products
  - health care: patients × symptoms
  - web: pages × keywords
  - text: texts × words
  - quality: products × defaults
  - ...

id_1	A_1	A_2	A_3	A_5	A_6	A_7	A_8
id_2	A_2	A_4	A_5	A_6			
id_3	A_1	A_2	A_3	A_4	A_6	A_7	A_8
id_5	A_2	A_4	A_5	A_7	A_8		
id_6	A_1	A_2	A_3	A_6	A_7		
id_7	A_2	A_3	A_4	A_5	A_6	A_7	A_8

	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8
id_1	1	1	1	0	1	1	1	1
id_2	0	1	0	1	1	1	0	0
id_3	1	1	1	1	0	1	1	1
id_4	1	1	1	0	0	1	1	1
id_5	0	1	0	1	1	0	1	1
id_6	1	1	1	0	0	1	1	0
id_7	0	1	1	1	1	1	1	1

Roots: AIS [AIS93] & APRIORI [AS94]

A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8
1	1	1	0	1	1	1	1
0	1	0	1	1	1	0	0
1	1	1	1	0	1	1	1
1	1	1	0	0	1	1	1
0	1	0	1	1	0	1	1
1	1	1	0	0	1	1	0
0	1	1	1	1	1	1	1

⇒ Works for `examples×properties` datasets... monotonicity property [AS94]: great success.

Boolean data bases

Discrete attibute:

id_1	P
id_2	A
id_3	P
id_4	P
id_5	E

	P	E	A
id_1	1		
id_2			1
id_3	1		
id_4	1		
id_5		1	

Continuous attribute:

id_1	1100
id_2	0
id_3	2200
id_4	800
id_5	3500

	[0..500]	[501..1000]	[1001..2500]	[2501..]
id_1			1	
id_2	1			
id_3			1	
id_4		1		
id_5				1

↪ One can always encode in binary attributes.

Outline

Binary attributes and frequent itemsets

A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8
1	1	1	0	1	1	1	1
0	1	0	1	1	1	0	0
1	1	1	1	0	1	1	1
1	1	1	0	0	1	1	1
0	1	0	1	1	0	1	1
1	1	1	0	0	1	1	0
0	1	1	1	1	1	1	1

Binary attributes and frequent itemsets

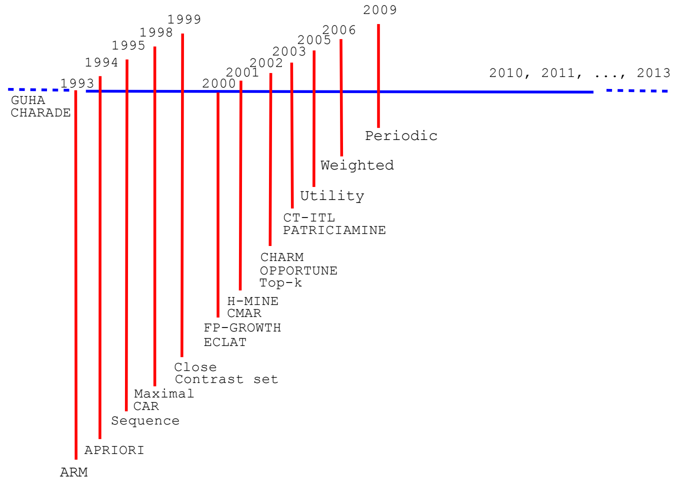
A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8
1	1	1	0	1	1	1	1
0	1	0	1	1	1	0	0
1	1	1	1	0	1	1	1
1	1	1	0	0	1	1	1
0	1	0	1	1	0	1	1
1	1	1	0	0	1	1	0
0	1	1	1	1	1	1	1

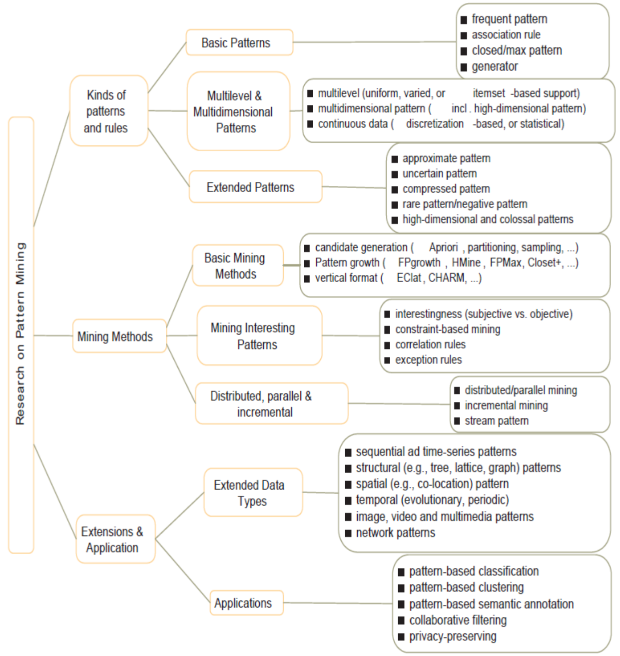
Binary attributes and frequent itemsets

A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8
1	1	1	0	1	1	1	1
0	1	0	1	1	1	0	0
1	1	1	1	0	1	1	1
1	1	1	0	0	1	1	1
0	1	0	1	1	0	1	1
1	1	1	0	0	1	1	0
0	1	1	1	1	1	1	1

$\hookrightarrow$ Co-occurrence (presence and absence): works on '1'.

Some key works





[HKP11]

Outline

Mining Association Rules

Formally [AIS93, AS94]

- $\mathcal{I} = \{i_1, i_2, \ldots, i_m\}$: a set of binary attributes, called items
- $\mathcal{D}$: a data base of transactions, where:
  - each transaction t is a set of items, $t \subseteq \mathcal{I}$
  - each transaction t is represented as a binary vector ($t[k] = 1$ if t contains i_k, and $t[k] = 0$ otherwise)
  - each transaction t has an unique associated identifier
- $\mathtt{A}$ a set of items (itemset) in $\mathcal{I}$:
  - t satisfies $\mathtt{A}$ if $\forall i_k \in \mathtt{A}$, $t[k] = 1$ (t contains $\mathtt{A}$ i.e. $\mathtt{A} \subseteq t$)

Association rule [AS94]

An association rule is an implication of the form $\mathtt{A} \rightarrow \mathtt{B}$, where $\mathtt{A} \subset \mathcal{I}$, $\mathtt{B} \subset \mathcal{I}$, and $\mathtt{A} \cap \mathtt{B} = \emptyset$ (in [AIS93] $\mathtt{B}$ is a single item).

Confidence and support of $A \rightarrow B$ (in $\mathcal{D}$)

- $A \rightarrow B$ holds with confidence c
  if c% of transactions in $\mathcal{D}$ that contain A also contain B.
- $A \rightarrow B$ has support s
  if s% of transactions in $\mathcal{D}$ contain $A \cup B$.

Problem of mining association rules [AS94]

To generate all association rules that have support and confidence greater than the user-specified minimum support and minimum confidence respectively.
Syntactic constraints were introduced in [AIS93].

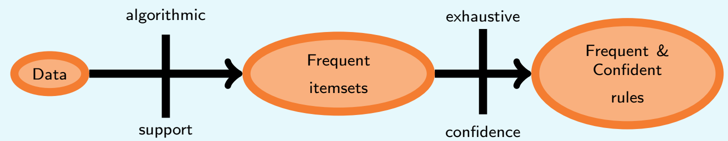
Which products are frequently bought together?

How to solve the problem of mining association rules?

- very exhaustive
  - generate all possible rules
  - count their supports and compute confidence
  - but... $\mathcal{O}(3^n)$
- more clever
  - first, find all frequent itemsets
  - second, split every frequent itemset I in two parts A and B, such that $A \rightarrow B$ is confident

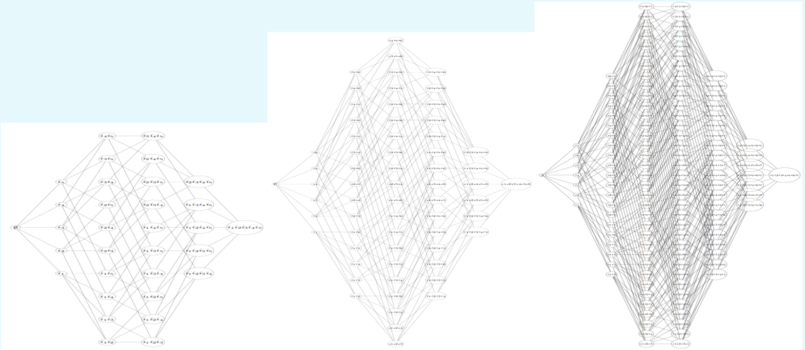
How to find all frequent itemsets?

- very exhaustive
  - generate all possible itemsets, count their support
  - but... $\mathcal{O}(2^n)$
- very clever
  - APRIORI [AS94]
  - key-point: downward-closure property of support (also called anti-monotonicity)
    - all subsets of a frequent itemset are also frequent
    - all supersets of an infrequent itemset are infrequent



Mining Association Rules

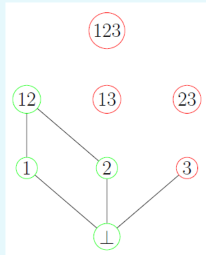
Poset of itemsets



Mining Association Rules

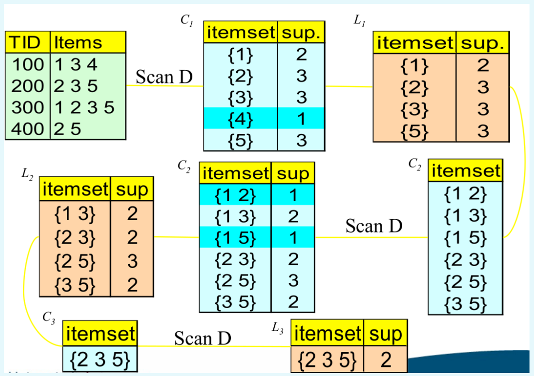
APRIORI [AS94]

- an itemset is called a candidate itemset if all of its subsets are known to be frequent
- so iteratively find frequent itemsets with cardinality from 1 to k (k-itemset); level-wise search



Mining Association Rules

Example (Rastogi and Shim)



⇒ Problem: every count step needs a (very) costly scan over the complete database.

Outline

APRIORI: optimizations

Strategies

- reduce the number of candidate itemsets
- reduce the number of transactions
- reduce the number of comparisons

Count step needs a (very) costly scan over $\mathcal{D}$: optimizations

- partition [SON95]: partition database, and mine each part separately (relative support instead of absolute support), union of all frequent itemsets of all parts are a superset of all frequent itemsets in $\mathcal{D}$, extra pruning step
- sampling [Toi96]: run APRIORI on small sample of $\mathcal{D}$, correct result
- Dynamic Itemset Counting [BMUT97]: interrupt algorithm after every x transactions and already generate larger candidates if possible

APRIORI: optimizations

Many research

- for sparse/dense data, for many/few items
- to improve the counting step
- to read efficiently the database
- to generate efficiently the candidates
- to prune the candidates
- to manage efficiently the ordering of items
- ...

APRIORI: optimizations

To improve the counting step if $\mathcal{D}$ fits in memory

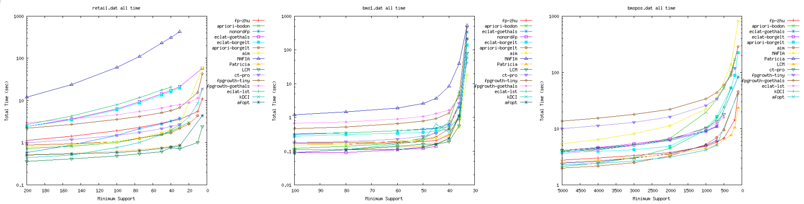
- ECLAT [Zak00]
- FP-GROWTH [HPY00]
- ...

$\Rightarrow$ Differ in counting strategy and how $\mathcal{D}$ is represented in memory.

APRIORI: optimizations

Many optimizations exist!

- no winner, it depends on implementation, on data



`http://fimi.ua.ac.be/experiments/`

$\Rightarrow$ ...Implementation matters!

FIM: extensions

Concise representations of frequent itemsets

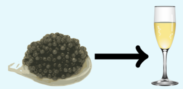
- an itemset is maximal frequent if none of its immediate supersets is frequent [Jr.98]
- an itemset is frequent closed if none of its immediate supersets has the same support as the itemset [PBTL99]
- $FIM \subset CLO \subset MAX$

$\hookrightarrow$ Closed and maximal frequent itemsets are typically by orders of magnitude fewer itemsets than all frequent itemsets. However, all frequent itemsets can be induced from these itemsets and thus algorithms mining closed and maximal frequent itemsets are often more efficient.

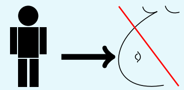
Apriori-like approach to mine association rules

Main issues: complexity and quality

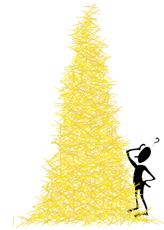
- large number of itemsets, of rules
  ... most of them uninteresting
- some infrequent patterns may be lost: nuggets



- some frequent patterns may be true but well known/obvious



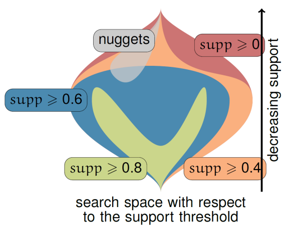
- ... invalid patterns... surprising patterns...



How to select the *good* ones?

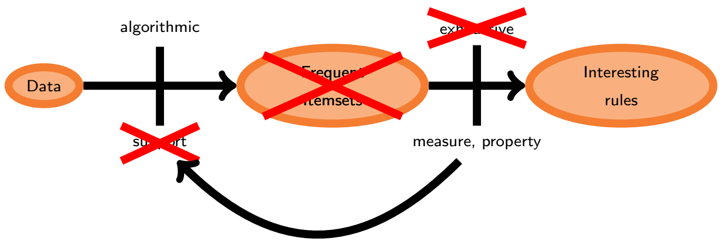
$\hookrightarrow$ Interestingness measures [LMVL08].

Next step: use of the good interestingness measure (without support)?



finding frequent itemsets is costly
can we avoid this steep?
can we reach directly interesting rules?

$\Rightarrow$ algorithmic properties of measures [LBLL09, BLL11]



Outline

References I

[AIS93] R. Agrawal, T. Imielinski, and A. N. Swami.
Mining association rules between sets of items in large databases.
In *SIGMOD*, 1993.

[AS94] R. Agrawal and R. Srikant.
Fast algorithms for mining association rules.
In *VLDB*, pages 487–499, 1994.

[BLL11] Yannick Le Bras, Philippe Lenca, and Stéphane Lallich.
Mining classification rules without support: an anti-monotone property of jaccard measure.
In Tapio Elomaa, Jaakko Hollmén, and Heikki Mannila, editors, *Discovery Science*, volume 6926 of *Lecture Notes in Computer Science*, pages 179–193. Springer, 2011.

[BMUT97] Sergey Brin, Rajeev Motwani, Jeffrey D. Ullman, and Shalom Tsur.
Dynamic itemset counting and implication rules for market basket data.
In *SIGMOD 1997, Proceedings ACM SIGMOD International Conference on Management of Data*, pages 255–264, Tucson, Arizona, USA, May 1997.

[Goe03] B. Goethals.
Survey on frequent pattern mining.
Manuscript, 2003.

[HCXY07] J. Han, H. Cheng, D. Xin, and X. Yan.
Frequent pattern mining: current status and future directions.
Data Min. Knowl. Discov., 15(1):55–86, 2007.

[HKP11] J. Han, M. Kamber, and J. Pei.
Data Mining: Concepts and Techniques.
The Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann Publishers, 3rd edition, 2011.

[HPY00] Jiawei Han, Jian Pei, and Yiwen Yin.
Mining frequent patterns without candidate generation.
In *SIGMOD Conference*, pages 1–12, 2000.

References II

[Jr.98] Roberto J. Bayardo Jr.
Efficiently mining long patterns from databases.
In *SIGMOD Conference*, pages 85–93, 1998.

[LBLL09] Y. Le Bras, P. Lenca, and S. Lallich.
On optimal rules mining: a framework and a necessary and sufficient condition for optimality.
In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, volume 5476 of *Lecture Notes in Computer Science*, pages 705–712. Springer-Verlag Berlin Heidelberg, 2009.

[LMVL08] P. Lenca, P. Meyer, B. Vaillant, and S. Lallich.
On selecting interestingness measures for association rules: user oriented description and multiple criteria decision aid.
European Journal of Operational Research, 184(2):610–626, 2008.

[PBTL99] Nicolas Pasquier, Yves Bastide, Rafik Taouil, and Lotfi Lakhal.
Discovering frequent closed itemsets for association rules.
In Catriel Beeri and Peter Buneman, editors, *ICDT*, volume 1540 of *Lecture Notes in Computer Science*, pages 398–416. Springer, 1999.

[SON95] Ashok Savasere, Edward Omiecinski, and Shamkant Navathe.
An efficient algorithm for mining association rules in large databases.
In *Proceedings of the 21st VLDB Conference*, pages 432–443, Zurich, Switzerland, 1995.

[Toi96] Hannu Toivonen.
Sampling large databases for association rules.
In *VLDB '96: Proceedings of the 22th International Conference on Very Large Data Bases*, pages 134–145, San Francisco, CA, USA, 1996. Morgan Kaufmann Publishers Inc.

[WKRQ+07] Xindong Wu, Vipin Kumar, J. Ross Quinlan, Joydeep Ghosh, Qiang Yang, Hiroshi Motoda, Geoffrey J. McLachlan, Angus Ng, Bing Liu, Philip S. Yu, Zhi-Hua Zhou, Michael Steinbach, David J. Hand, and Dan Steinberg.
Top 10 algorithms in data mining.
Knowl. Inf. Syst., 14(1):1–37, December 2007.

References III

[Zak00] Mohammed J. Zaki.
Scalable algorithms for association mining.
IEEE Transactions on Knowledge and Data Engineering, 12(3):372–390, May/June 2000.