Two decades of Pattern Mining

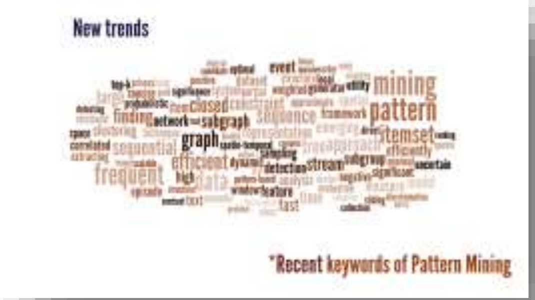
eBISS2016, Tours

*arnaud.soulet@univ-tours.fr

Outline

What is Pattern Mining?

What are the main principles?



What are the recent trends?

What is Pattern Mining?

Mining Association Rules between Sets of Items in Large Databases

Rakesh Agrawal Tomasz Imielinski* Arun Swami

IBM Almaden Research Center
650 Harry Road, San Jose, CA 95120

Abstract

We are given a large database of customer transactions. Each transaction consists of items purchased by a customer in a visit. We present an efficient algorithm that generates all significant association rules between items in the database. The algorithm incorporates buffer management and novel estimation and pruning techniques. We also present results of applying this algorithm to sales data obtained from a large retailing company, which shows the effectiveness of the algorithm.

1 Introduction

Consider a supermarket with a large collection of items. Typical business decisions that the management of the supermarket has to make include what to put on sale, how to design coupons, how to place merchandise on shelves in order to maximize the profit, etc. Analysis of past transaction data is a commonly used approach in order to improve the quality of such decisions. Until recently, however, only global data about the cumulative sales during some time period (a day, a week, a month, etc.) was available on the computer. Progress in bar-code technology has made it possible to store the so called basket data that stores items purchased on a per-transaction basis. Basket data type transactions do not necessarily consist of items bought together at the same point of time. It may consist of items bought by a customer over a period of time. Examples include monthly purchases by members of a book club or a music club.

Several organizations have collected massive amounts of such data. These data sets are usually stored on tertiary storage and are very slowly migrating to database systems. One of the main reasons for the limited success of database systems in this area is that current database systems do not provide necessary functionality for a user interested in taking advantage of this information.

This paper introduces the problem of "mining" a large collection of basket data type transactions for association rules between sets of items with some minimum specified confidence, and presents an efficient algorithm for this purpose. An example of such an association rule is the statement that 90% of transactions that purchase bread and butter also purchase milk. The antecedent of this rule consists of bread and butter and the consequent consists of milk alone. The number 90% is the confidence factor of the rule.

The work reported in this paper could be viewed as a step towards enhancing databases with functionalities to process queries such as (we have omitted the confidence factor specification):

- Find all rules that have "Diet Coke" as consequent. These rules may help plan what the store should do to boost the sale of Diet Coke.
- Find all rules that have "bagels" in the antecedent. These rules may help determine what products may be impacted if the store discontinues selling bagels.
- Find all rules that have "sausage" in the antecedent and "mustard" in the consequent. This query can be phrased alternatively as a request for the additional items that have to be sold together with sausage in order to make it highly likely that mustard will also be sold.
- Find all the rules relating items located on shelves A and B in the store. These rules may help shelf planning by determining if the sale of items on shelf A is related to the sale of items on shelf B.
- Find the "best" k rules that have "bagels" in the consequent. Here, "best" can be formulated in terms of the confidence factors of the rules, or in terms

*Current address: Computer Science Department, Rutgers University, New Brunswick, NJ 08903

SIGMOD /5/93/Washington, DC,USA
© 1993 ACM 0-89791-592-5/93/0005/0207...$1.50

20 years ago...

*SIGMOD Conference 1993

New problem

Mining Association Rules between Sets of Items in Large Databases

We introduced the problem of mining association rules between sets of items in a large database of customer transactions. Each transaction consists of items purchased by a customer in a visit. We are interested in finding those rules that have:

- Minimum transactional support s — the union of items in the consequent and antecedent of the rule is present in a minimum of s% of transactions in the database.
- Minimum confidence c — at least c% of transactions in the database that satisfy the antecedent of the rule also satisfy the consequent of the rule.

monthly purchases by members of a book club or a music club.
Several organizations have collected massive amounts of such data. These data sets are usually stored

and "mustard" in the consequent. This query can be phrased alternatively as a request for the additional items that have to be sold together with sausage in order to make it highly likely that mustard will also

An example of such an association rule is the statement that 90% of transactions that purchase bread and butter also purchase milk. The antecedent of this rule consists of bread and butter and the consequent consists of milk alone. The number 90% is the confidence factor of the rule.

*Discovering all relevant association rules

New solution

Mining Association Rules between Sets of Items in Large Databases

We introduced the problem of mining association rules between sets of items in a large database of customer transactions. Each transaction consists of items purchased by a customer in a visit. We are interested in finding those rules that have:

- Minimum transactional support s — the union of items in the consequent and antecedent of the rule is present in a minimum of s% of transactions in the database.
- Minimum confidence c — at least c% of transactions in the database that satisfy the antecedent of the rule also satisfy the consequent of the rule.

monthly purchases by members of a book club or a music club.
Several organizations have collected massive amounts

and "mustard" in the consequent. This query can be phrased alternatively as a request for the additional items that have to be sold together with sausage in

We solve this problem by decomposing it into two subproblems:

1. Finding all itemsets, called *large* itemsets, that are present in at least s% of transactions.
2. Generating from each large itemset, rules that use items from the large itemset.

*Enumerating all frequent itemsets

Leonardo DiCaprio Plays the Same Character Over and Over

*Source: mic.com

Leonardo DiCaprio Plays the Same Character Over and Over

Troubled Romantic	Rich	Dies	Hidding Secret

Troubled romantic + Rich

(supp = 0.4)

*Itemset

Leonardo DiCaprio Plays the Same Character Over and Over

Troubled romantic → Rich

(supp = 0.4 / conf = 0.5)

*Association rule

Leonardo DiCaprio Plays the Same Character Over and Over

Troubled romantic → Rich

(supp = 0.4 / conf = 0.5)

Troubled romantic → Dies

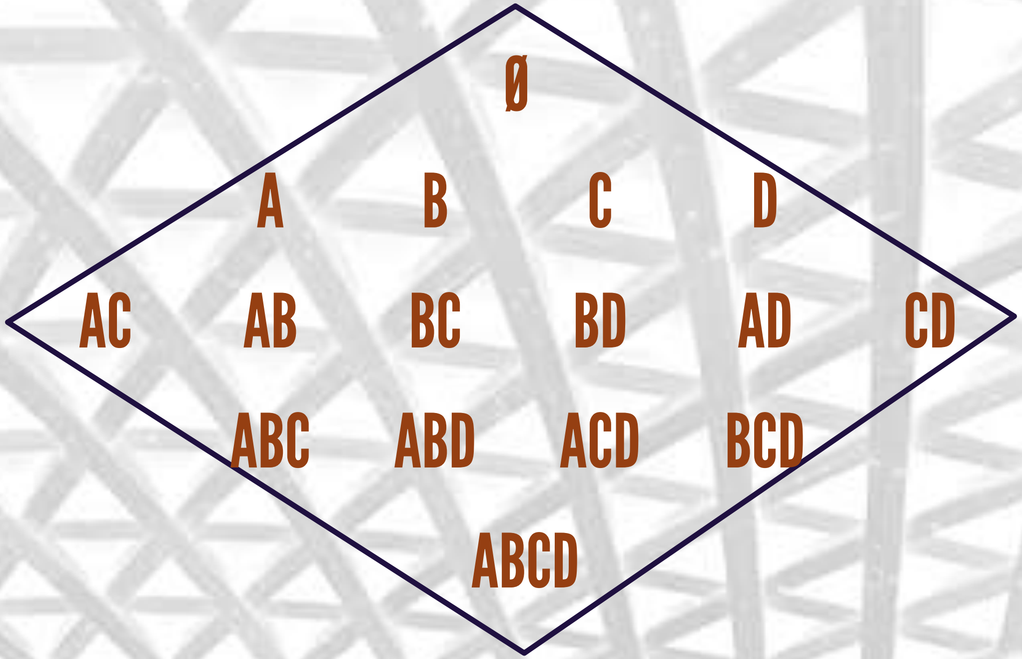
(supp = 0.6 / conf = 0.75)

Leonardo DiCaprio Plays the Same Character Over and Over

	Troubled Romantic	Rich	Dies	Hidding Secret
	A		C	
	A			D
	A	B	C	
		B	C	
	A	B	C	D

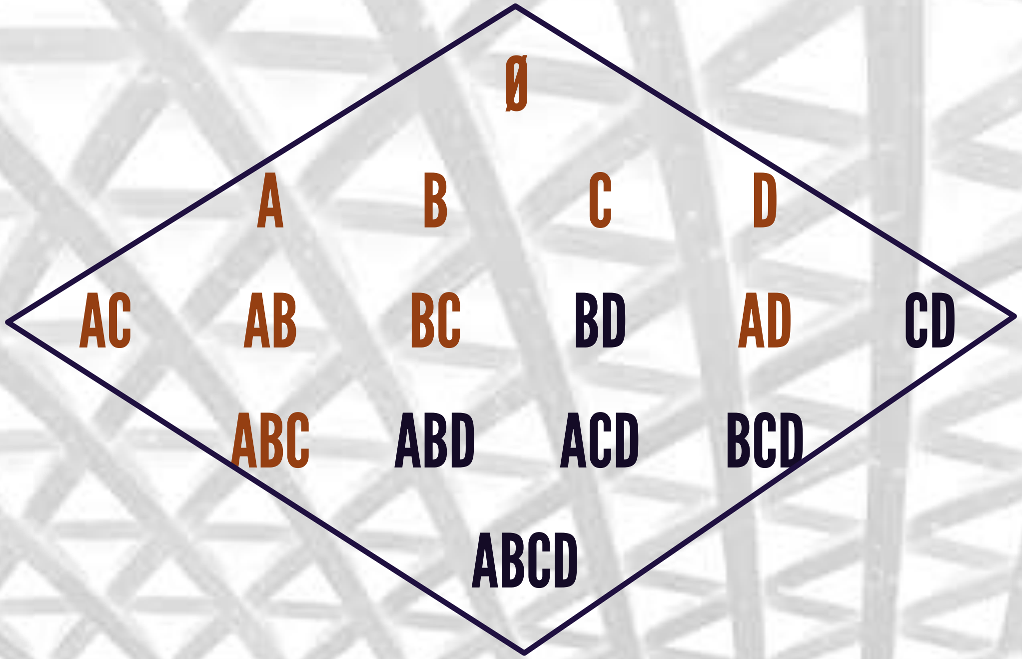
	Troubled Romantic	Rich	Dies	Hidding Secret
	A		C	
	A			D
	A	B	C	
		B	C	
	A	B	C	D

Frequent patterns



*Minimal frequency threshold = 1

	Troubled Romantic	Rich	Dies	Hidding Secret
	A		C	
	A			D
	A	B	C	
		B	C	
	A	B	C	D

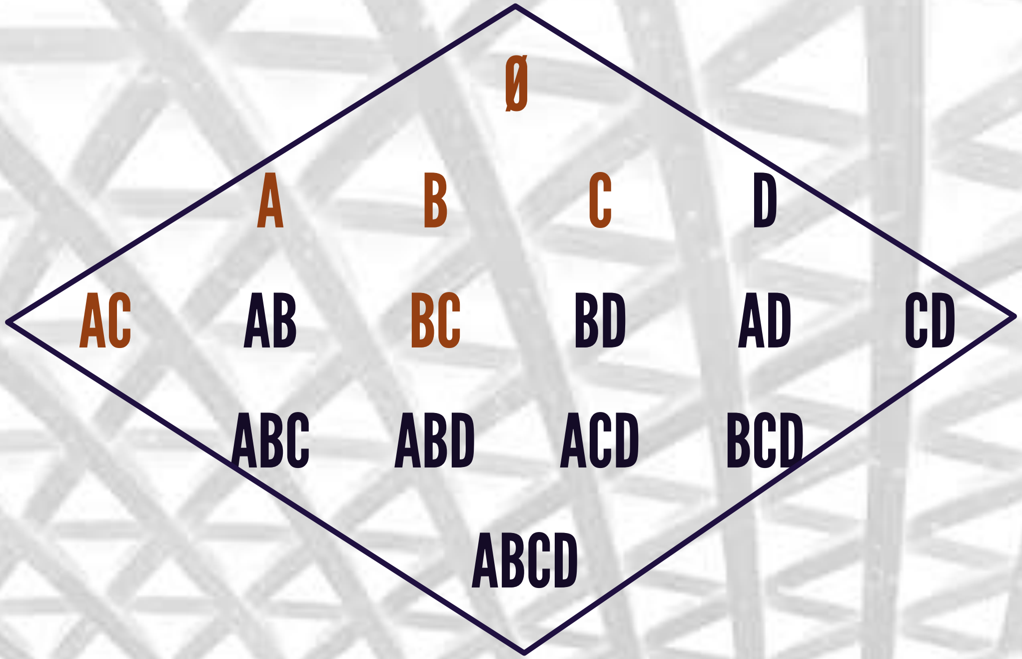


Frequent patterns

*Minimal frequency threshold = 2

	Troubled Romantic	Rich	Dies	Hidding Secret
	A		C	
	A			D
	A	B	C	
		B	C	
	A	B	C	D

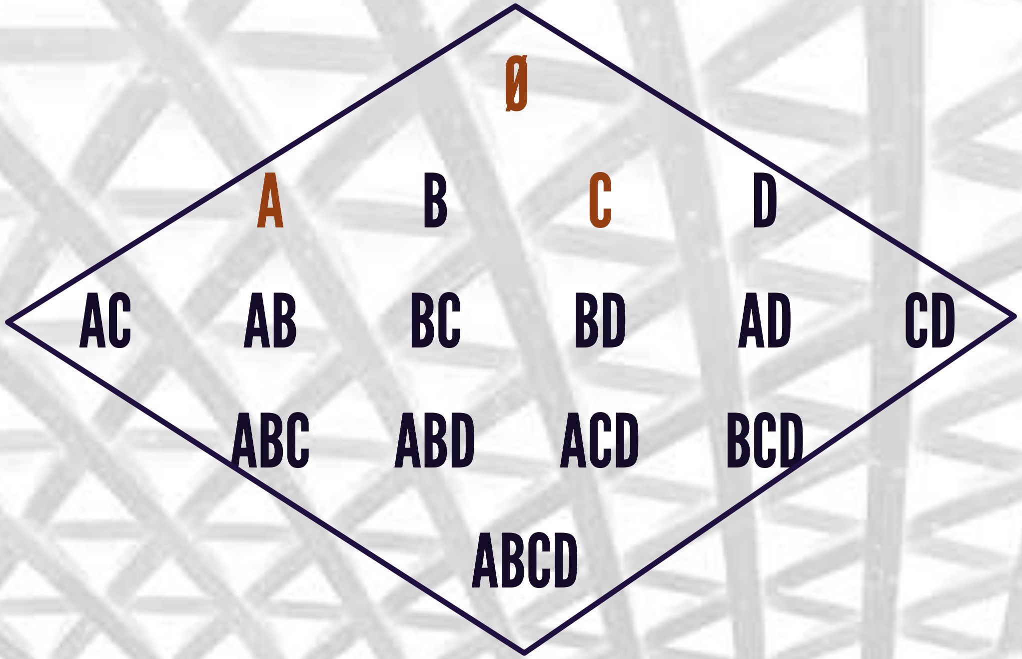
Frequent patterns



*Minimal frequency threshold = 3

	Troubled Romantic	Rich	Dies	Hidding Secret
	A		C	
	A			D
	A	B	C	
		B	C	
	A	B	C	D

Frequent patterns



*Minimal frequency threshold = 4

Exact solution

Exhaustive search

Speed of answer

Pattern Mining

*The footprint of databases

Exact solution

Exhaustive search

Speed of answer

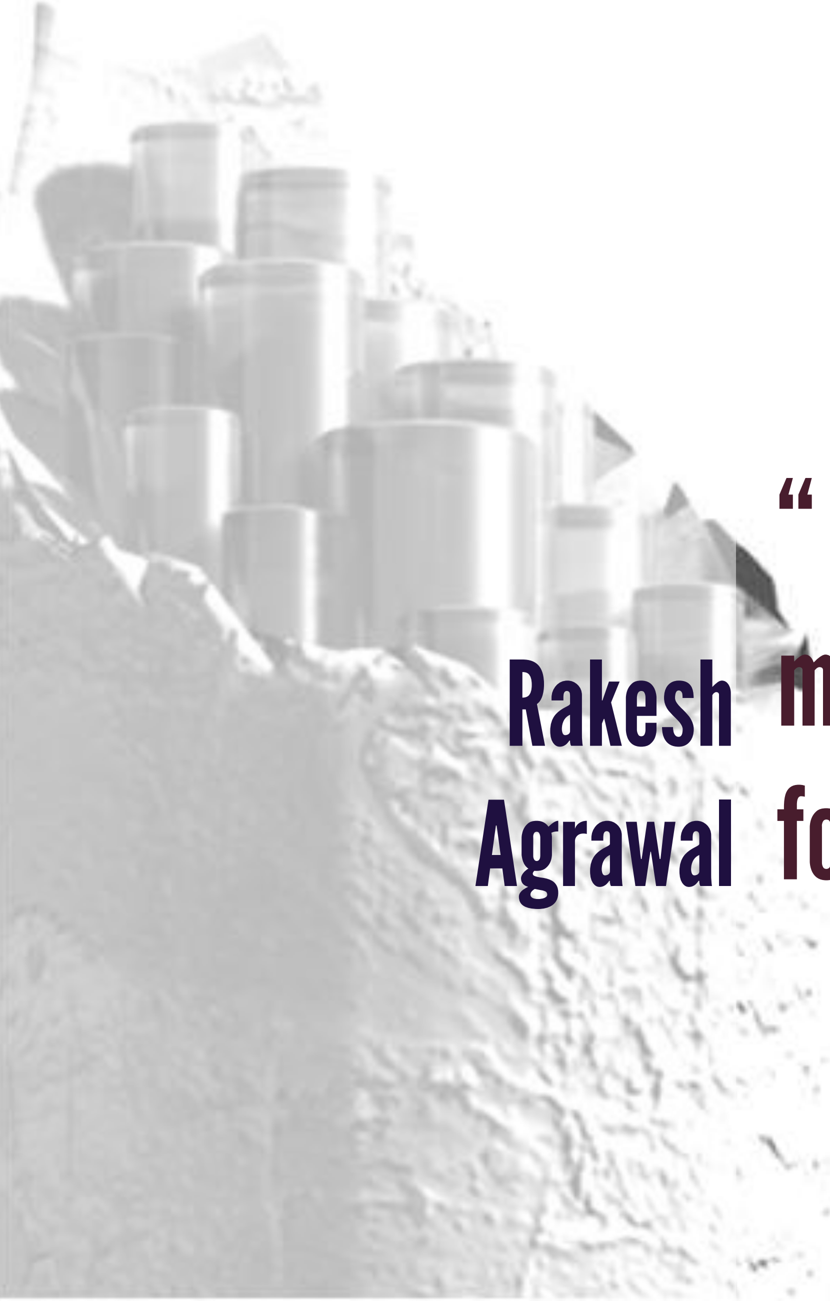
Approximate solution

Heuristic search

Quality of solution

Pattern Mining vs Artificial Intelligence

*The footprint of databases



Rakesh Agrawal

" I'm a database person, so my view of data mining has been that it is essentially a richer form of querying."

*The footprint of databases

Impact of this seminal paper?

Mining Association Rules between Sets of Items in Large Databases

Rakesh Agrawal Tomasz Imielinski* Arun Swami

IBM Almaden Research Center
650 Harry Road, San Jose, CA 95120

Abstract

We are given a large database of customer transactions. Each transaction consists of items purchased by a customer in a visit. We present an efficient algorithm that generates all significant association rules between items in the database. The algorithm incorporates buffer management and novel estimation and pruning techniques. We also present results of applying this algorithm to sales data obtained from a large retailing company, which shows the effectiveness of the algorithm.

1 Introduction

Consider a supermarket with a large collection of items. Typical business decisions that the management of the supermarket has to make include what to put on sale, how to design coupons, how to place merchandise on shelves in order to maximize the profit, etc. Analysis of past transaction data is a commonly used approach in order to improve the quality of such decisions. Until recently, however, only global data about the cumulative sales during some time period (a day, a week, a month, etc.) was available on the computer. Progress in bar-code technology has made it possible to store the so called basket data that stores items purchased on a per-transaction basis. Basket data type transactions do not necessarily consist of items bought together at the same point of time. It may consist of items bought by a customer over a period of time. Examples include monthly purchases by members of a book club or a music club.

Several organizations have collected massive amounts of such data. These data sets are usually stored on tertiary storage and are very slowly migrating to database systems. One of the main reasons for the limited success of database systems in this area is that current database systems do not provide necessary functionality for a user interested in taking advantage of this information.

This paper introduces the problem of "mining" a large collection of basket data type transactions for association rules between sets of items with some minimum specified confidence, and presents an efficient algorithm for this purpose. An example of such an association rule is the statement that 90% of transactions that purchase bread and butter also purchase milk. The antecedent of this rule consists of bread and butter and the consequent consists of milk alone. The number 90% is the confidence factor of the rule.

The work reported in this paper could be viewed as a step towards enhancing databases with functionalities to process queries such as (we have omitted the confidence factor specification):

- Find all rules that have "Diet Coke" as consequent. These rules may help plan what the store should do to boost the sale of Diet Coke.
- Find all rules that have "bagels" in the antecedent. These rules may help determine what products may be impacted if the store discontinues selling bagels.
- Find all rules that have "sausage" in the antecedent and "mustard" in the consequent. This query can be phrased alternatively as a request for the additional items that have to be sold together with sausage in order to make it highly likely that mustard will also be sold.
- Find all the rules relating items located on shelves A and B in the store. These rules may help shelf planning by determining if the sale of items on shelf A is related to the sale of items on shelf B.
- Find the "best" k rules that have "bagels" in the consequent. Here, "best" can be formulated in terms of the confidence factors of the rules, or in terms

*Current address: Computer Science Department, Rutgers University, New Brunswick, NJ 08903

SIGMOD /5/93/Washington, DC,USA
© 1993 ACM 0-89791-592-5/93/0005/0207...$1.50


A Survey on Condensed Representations for Frequent Sets

1 Introduction

55 references


Survey on Frequent Pattern Mining

Bart Goethals
HIIT Basic Research Unit
Department of Computer Science
University of Helsinki
P.O. box 26, FIN-00014 Helsinki
Finland

1 Introduction

45 references

Classical survey

*Dozens of references

Classical survey

*Dozens of references

Bibliometric survey

*Thousands of references

Materials

KDD
1994

PAKDD
PKDD
1997

ICDM
SDM
2001

1993

*Data mining conferences ranked A

Materials

Fast Algorithms for Mining Association Rules

Rakesh Agrawal   Ramakrishnan Srikant*

IBM Almaden Research Center
650 Harry Road, San Jose, CA 95120

Abstract

We consider the problem of discovering association rules between items in a large database of sales transactions. We present two new algorithms for solving this problem that are fundamentally different from the known algorithms. Empirical evaluation shows that these algorithms outperform the known algorithms by factors ranging from three for small problems to more than an order of magnitude for large problems. We also show how the best features of the two proposed algorithms can be combined into a hybrid algorithm, called AprioriHybrid. Scale-up experiments show that AprioriHybrid scales linearly with the number of transactions. AprioriHybrid also has excellent scale-up properties with respect to the transaction size and the number of items in the database.

1 Introduction

Progress in bar-code technology has made it possible for retail organizations to collect and store massive amounts of sales data, referred to as the *basket* data. A record in such data typically consists of the transaction date and the items bought in the transaction. Successful organizations view such databases as important pieces of the marketing infrastructure. They are interested in instituting information-driven marketing processes, managed by database technology, that enable marketers to develop and implement customized marketing programs and strategies [6].

The problem of mining association rules over basket data was introduced in [4]. An example of such a rule might be that 98% of customers that purchase tires and auto accessories also get automotive services done. Finding all such rules is valuable for cross-marketing and attached mailing applications. Other applications include catalog design, add-on sales, store layout, and customer segmentation based on buying patterns. The databases involved in these applications are very large. It is imperative, therefore, to have fast algorithms for this task.

The following is a formal statement of the problem [4]: Let $\mathcal{I} = \{i_1, i_2, \ldots, i_m\}$ be a set of literals, called items. Let $\mathcal{D}$ be a set of transactions, where each transaction T is a set of items such that $T \subseteq \mathcal{I}$. Associated with each transaction is a unique identifier, called its *TID*. We say that a transaction T *contains* X, a set of some items in $\mathcal{I}$, if $X \subseteq T$. An *association rule* is an implication of the form $X \Longrightarrow Y$, where $X \subset \mathcal{I}$, $Y \subset \mathcal{I}$, and $X \cap Y = \emptyset$. The rule $X \Longrightarrow Y$ holds in the transaction set $\mathcal{D}$ with *confidence* c if $c\%$ of transactions in $\mathcal{D}$ that contain X also contain Y. The rule $X \Longrightarrow Y$ has *support* s in the transaction set $\mathcal{D}$ if $s\%$ of transactions in $\mathcal{D}$ contain $X \cup Y$. Our rules are somewhat more general than in [4] in that we allow a consequent to have more than one item.

Given a set of transactions $\mathcal{D}$, the problem of mining association rules is to generate all association rules that have support and confidence greater than the user-specified minimum support (called *minsup*) and minimum confidence (called *minconf*) respectively. Our discussion is neutral with respect to the representation of $\mathcal{D}$. For example, $\mathcal{D}$ could be a data file, a relational table, or the result of a relational expression.

An algorithm for finding all association rules, [illegible] In this paper, we present two new algorithms, *Apriori* and *AprioriTid*, that differ fundamentally from these algorithms. We present experimental results showing

*Visiting from the Department of Computer Science, University of Wisconsin, Madison.

Proceedings of the 20th VLDB Conference
Santiago, Chile, 1994

[Agrawal and Srikant, 1994]

*Not consider VLDB, CIKM, ICDE,...

Materials

KDI
199

ICDM
SDM
2001

Data Mining and Knowledge Discovery 1, 241–258 (1997)
© 1997 Kluwer Academic Publishers. Manufactured in The Netherlands.

Levelwise Search and Borders of Theories in Knowledge Discovery

HEIKKI MANNILA heikki.mannila@cs.helsinki.fi
HANNU TOIVONEN hannu.toivonen@cs.helsinki.fi
Department of Computer Science, P.O. Box 26, FIN-00014 University of Helsinki, Finland

Editor: Usama Fayyad

Received January 28, 1997; Revised July 10, 1997; Accepted July 11, 1997

Abstract. One of the basic problems in knowledge discovery in databases (KDD) is the following: given a data set r, a class $\mathcal{L}$ of sentences for defining subgroups of r, and a selection predicate, find all sentences of $\mathcal{L}$ deemed interesting by the selection predicate. We analyze the simple levelwise algorithm for finding all such descriptions. We give bounds for the number of database accesses that the algorithm makes. For this, we introduce the concept of the border of a theory, a notion that turns out to be surprisingly powerful in analyzing the algorithm. We also consider the verification problem of a KDD process: given r and a set of sentences $\mathcal{S} \subseteq \mathcal{L}$, determine whether $\mathcal{S}$ is exactly the set of interesting statements about r. We show strong connections between the verification problem and the hypergraph transversal problem. The verification problem arises in a natural way when using sampling to speed up the pattern discovery step in KDD.

Keywords: theory of knowledge discovery, association rules, episodes, integrity constraints, hypergraph transversals

1. Introduction

Knowledge discovery in databases (KDD), also called data mining, has recently received wide attention from practitioners and researchers. There are several attractive application areas for KDD, and it seems that techniques from machine learning, statistics, and databases can be profitably combined to obtain useful methods and systems for KDD. See, e.g., Fayyad et al. (1996), and Piatetsky-Shapiro and Frawley (1991) for general descriptions of the area.

The KDD area is and should be largely guided by (successful) applications. Still, theoretical work in the area is needed. In this paper we take some steps towards theoretical KDD. We consider a KDD process in which the analyzer first produces lots of potentially interesting rules, subgroup descriptions, patterns, etc., and then interactively selects the truly interesting ones from these. In this paper we analyze the first stage of this process

if φ then θ,

[Manila and Toivonen, 1997]

*Not consider DMKD, TKDE, KAIS,...

KDD : 1,905 since 1995

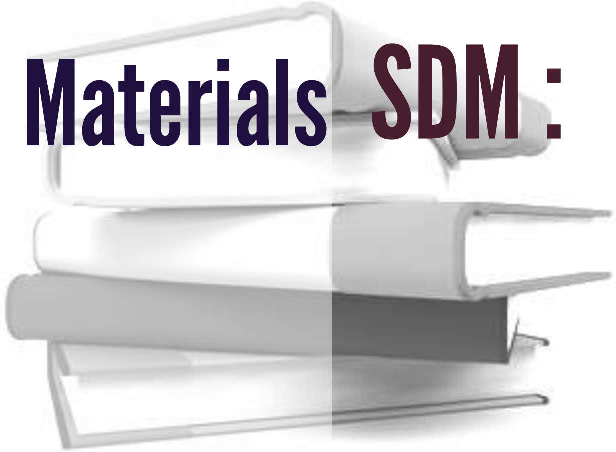
PKDD : 1,295 since 1997

PAKDD : 1,277 since 1998

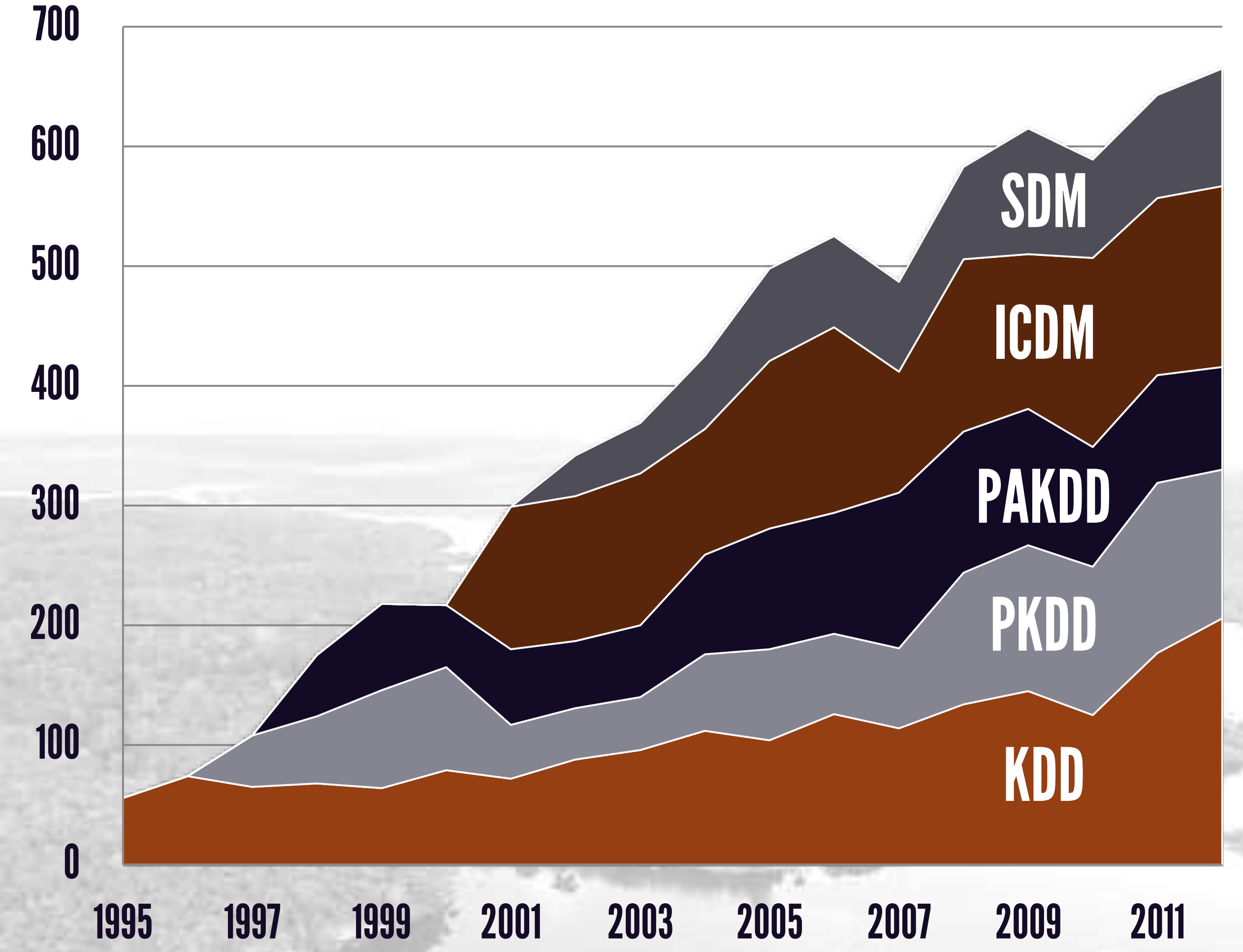
ICDM : 1,598 since 2001

SDM : 813 since 2002

Materials



*6,888 publications from DBLP (1995-2012)



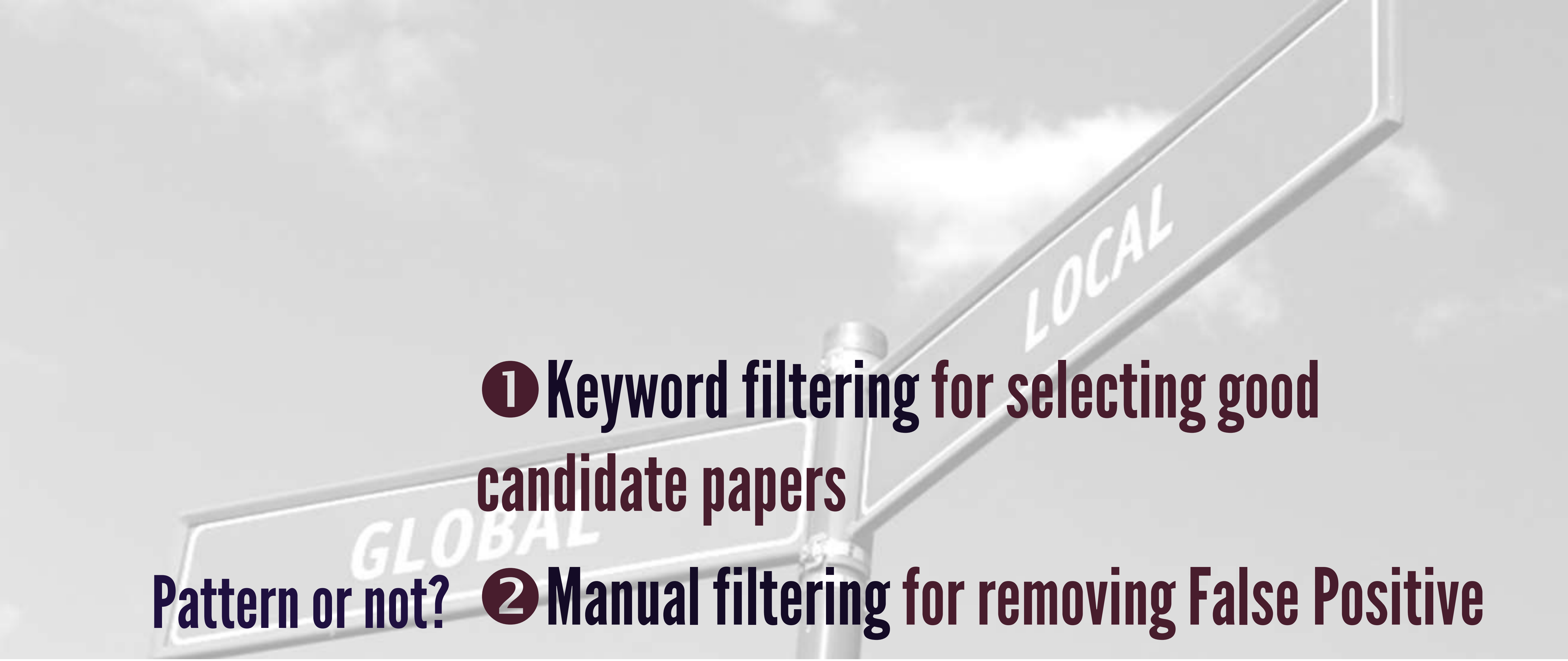
Faster, Higher, Stronger


mining
data
clustering
learning
classification
pattern
algorithm
graph
application
database
discovery
hierarchical
network
model
approach
itemset
decision
semi-supervised
constraint
detection
distributed
measure
kernel
ensemble
online
sampling
subspace
event
dataset
new
system
time
concept
prediction
machine
technique
information
relational
finding
bayesian
random
efficient
similarity
semantic
space
attribute
high
tree
vector
discovering
selection
serie
cluster
topic
local
large
text
set
structure
matrix
support
study
fast
social
probabilistic
outlier
feature
filtering
effective
latent
scalable
rule
association
incremental
sequential
web
frequent
stream
adaptive
method
spatial
extraction
active
framework
regression
modeling
classifier
temporal
ranking
dynamic
multiple
document
sequence
problem
search
reduction
knowledge
analysis

*Titles only

Pattern or not? Fuzzy limit

*Discovery and use



❶ Keyword filtering for selecting good candidate papers

Pattern or not? ❷ Manual filtering for removing False Positive

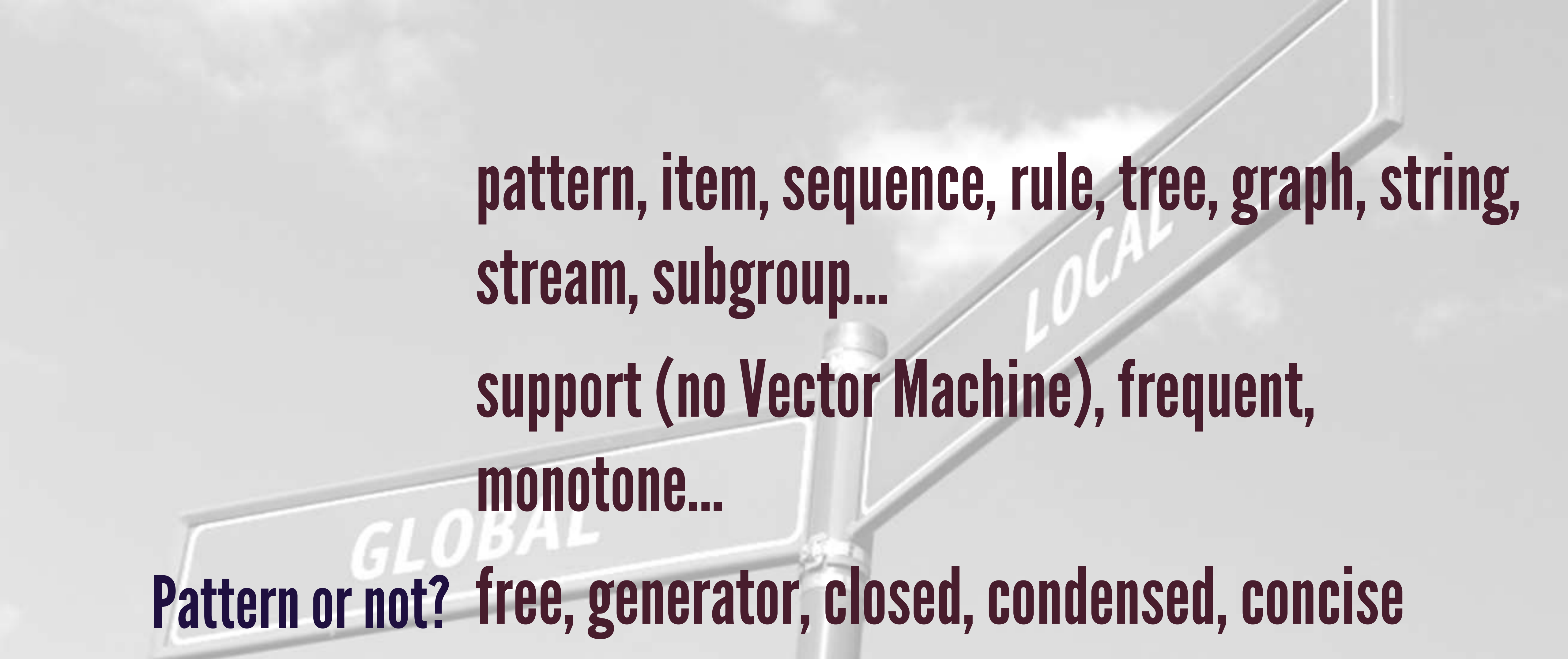
*Semi-automated topic assignment

Language

Constraint

Pattern or not? Condensed Representation

*Dimensions of Pattern Mining



pattern, item, sequence, rule, tree, graph, string, stream, subgroup...

support (no Vector Machine), frequent, monotone...

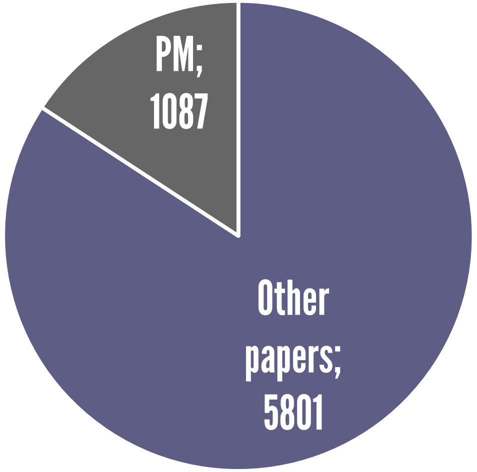
Pattern or not? free, generator, closed, condensed, concise

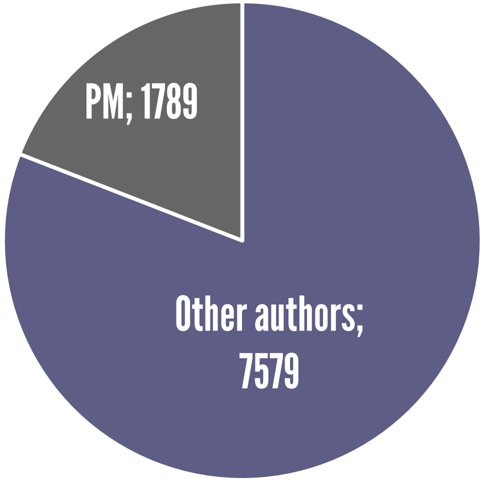
*Keywords of Pattern Mining

Keyword filtering	1,732
Manual filtering	1,087
Pattern or not?	(148 consulted abstracts)

*5% of False Negative, around 258 papers

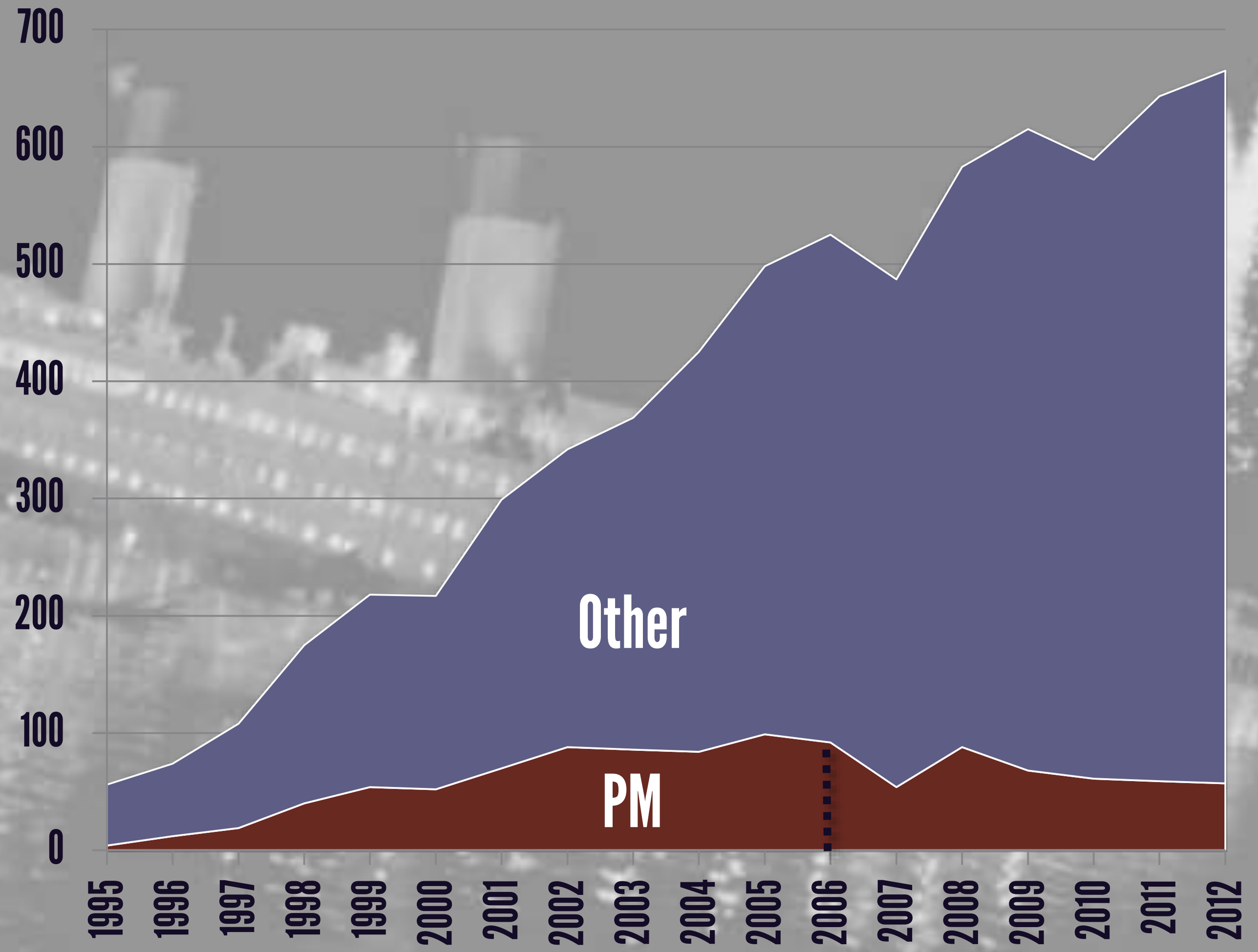
1/5th of authors and 1/6th of KDD publications





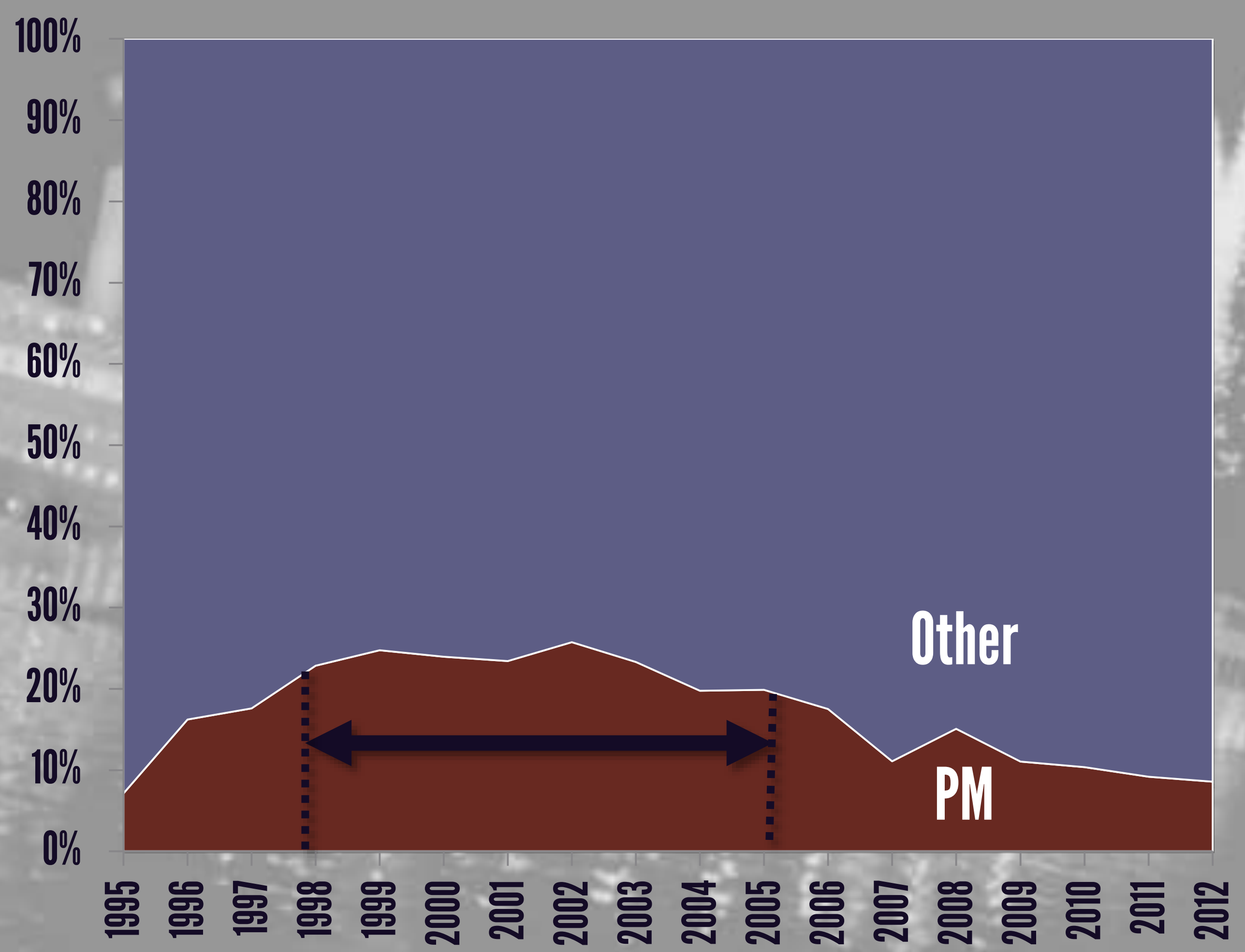
*PM is a true subfield of KDD

The slowdown of Pattern Mining



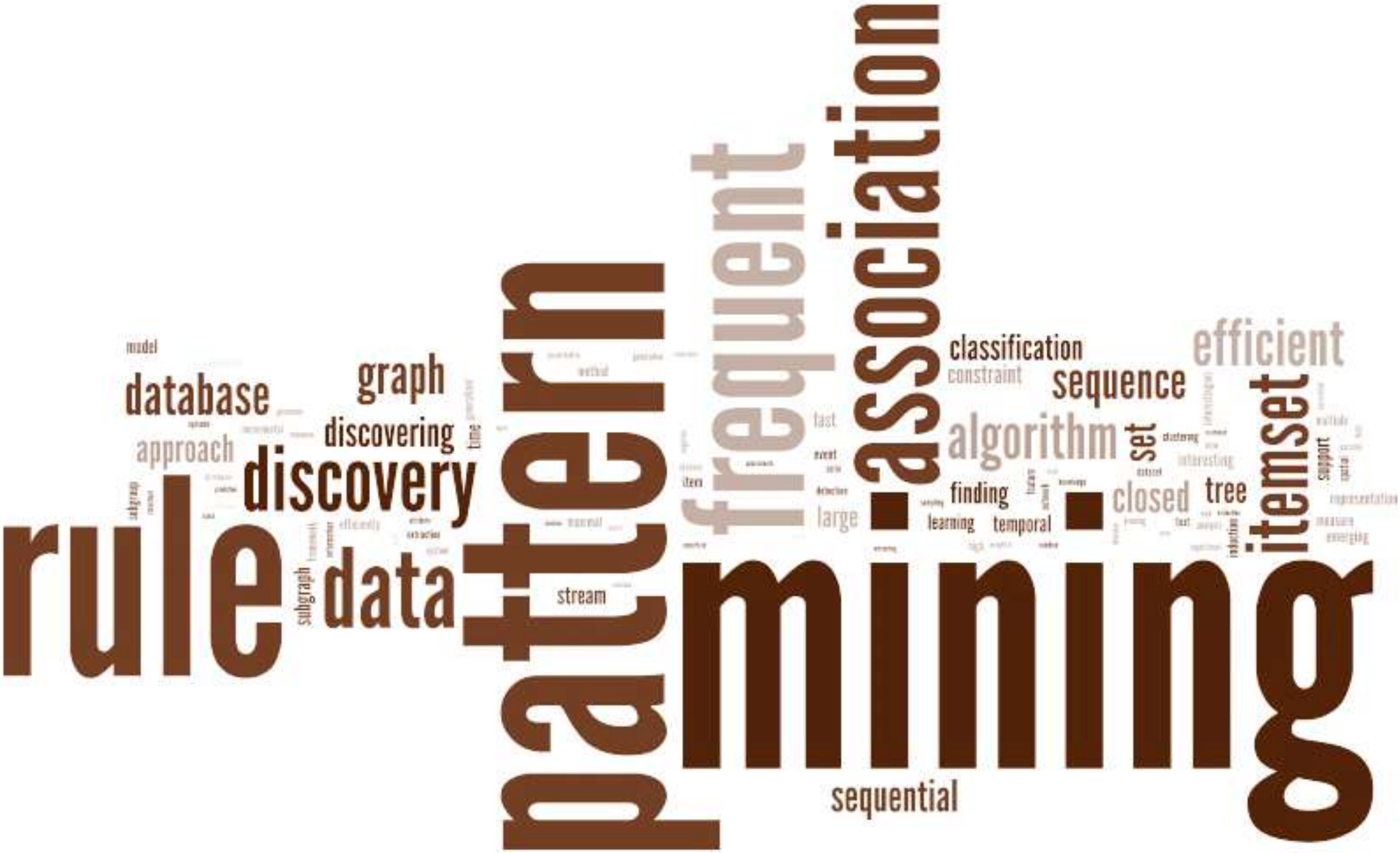
*Turning point: 2006

The slowdown of Pattern Mining



*Golden age: 1998-2005 (1 paper out of 5)

What are the main principles of Pattern Mining?


mining
pattern
rule
association
frequent
itemset
discovery
data
database
graph
discovering
approach
efficient
classification
constraint
sequence
algorithm
set
closed
tree
finding
learning
temporal
large
fast
sequential
stream
subgraph
model
time
support
interesting
clustering
representation
measure
emerging
event
items
efficiently
induction

Typology of publications


Data Mining and Knowledge Discovery 1, 241–258 (1997)
© 1997 Kluwer Academic Publishers. Manufactured in The Netherlands.

Levelwise Search and Borders of Theories in Knowledge Discovery

HEIKKI MANNILA heikki.mannila@cs.helsinki.fi
HANNU TOIVONEN hannu.toivonen@cs.helsinki.fi
Department of Computer Science, P.O. Box 26, FIN-00014 University of Helsinki, Finland

Editor: Usama Fayyad

Received January 28, 1997; Revised July 10, 1997; Accepted July 11, 1997

1. Introduction

Knowledge discovery in databases (KDD), also called data mining, has recently received wide attention from practitioners and researchers. There are several attractive application areas for KDD, and it seems that techniques from machine learning, statistics, and databases can be profitably combined to obtain useful methods and systems for KDD. See, e.g., Fayyad et al. (1996), and Piatetsky-Shapiro and Frawley (1991) for general descriptions of the area.

The KDD area is and should be largely guided by (successful) applications. Still, theoretical work in the area is needed. In this paper we take some steps towards theoretical KDD. We consider a KDD process in which the analyzer first produces lots of potentially interesting rules, subgroup descriptions, patterns, etc., and then interactively selects the truly interesting ones from these. In this paper we analyze the first stage of this process: how to find all the potentially interesting rules in the database.

The intuitive idea behind this work is as follows. A lot of work in data mining can be formulated in terms of finding all rules of the form

if φ then θ.

❶ Language

❷ Constraint

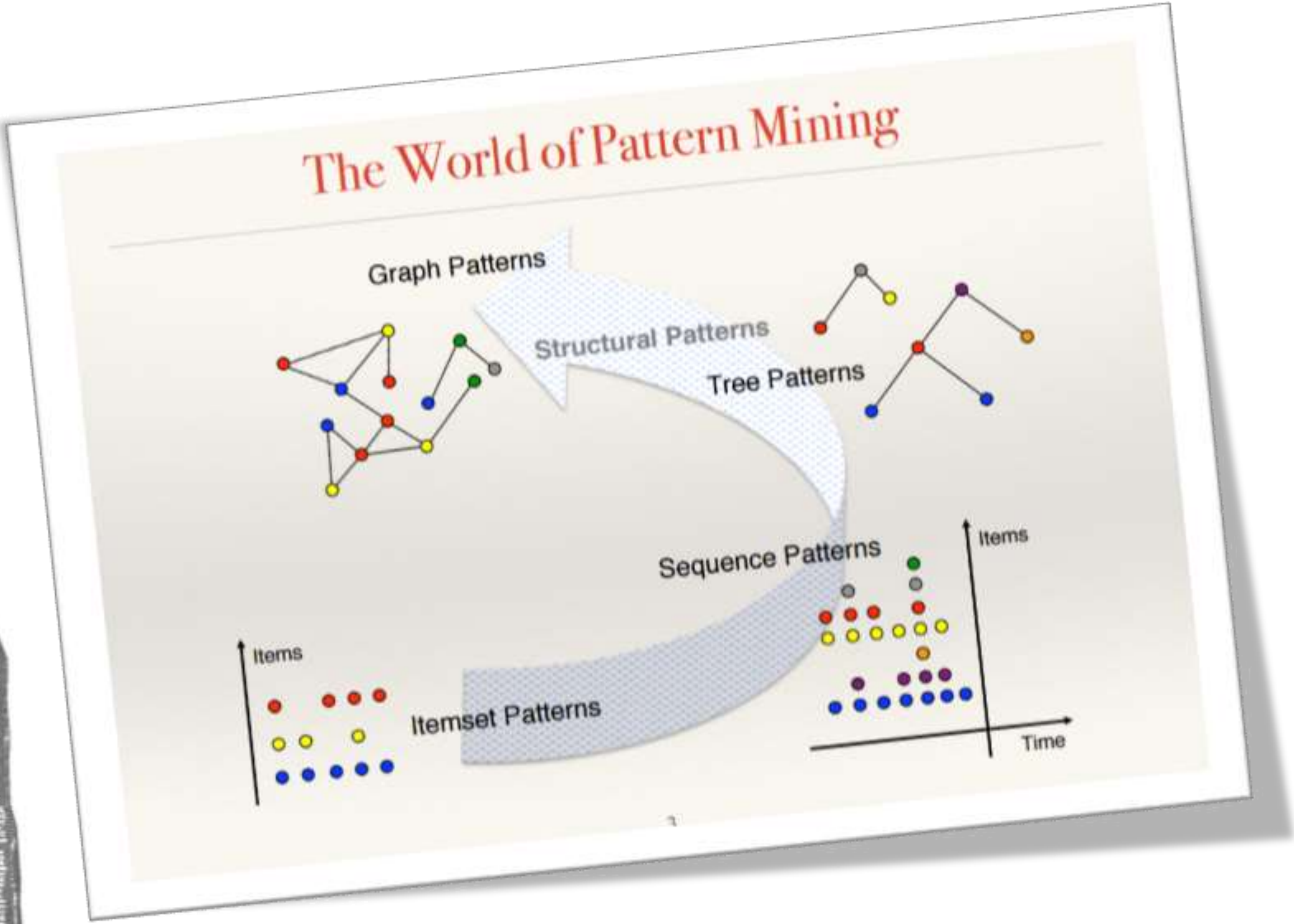
❸ Condensed Representation (CR)

*Levelwise Search and Borders of Theories in Knowledge Discovery

[Manila and Toivonen, 1997]

1

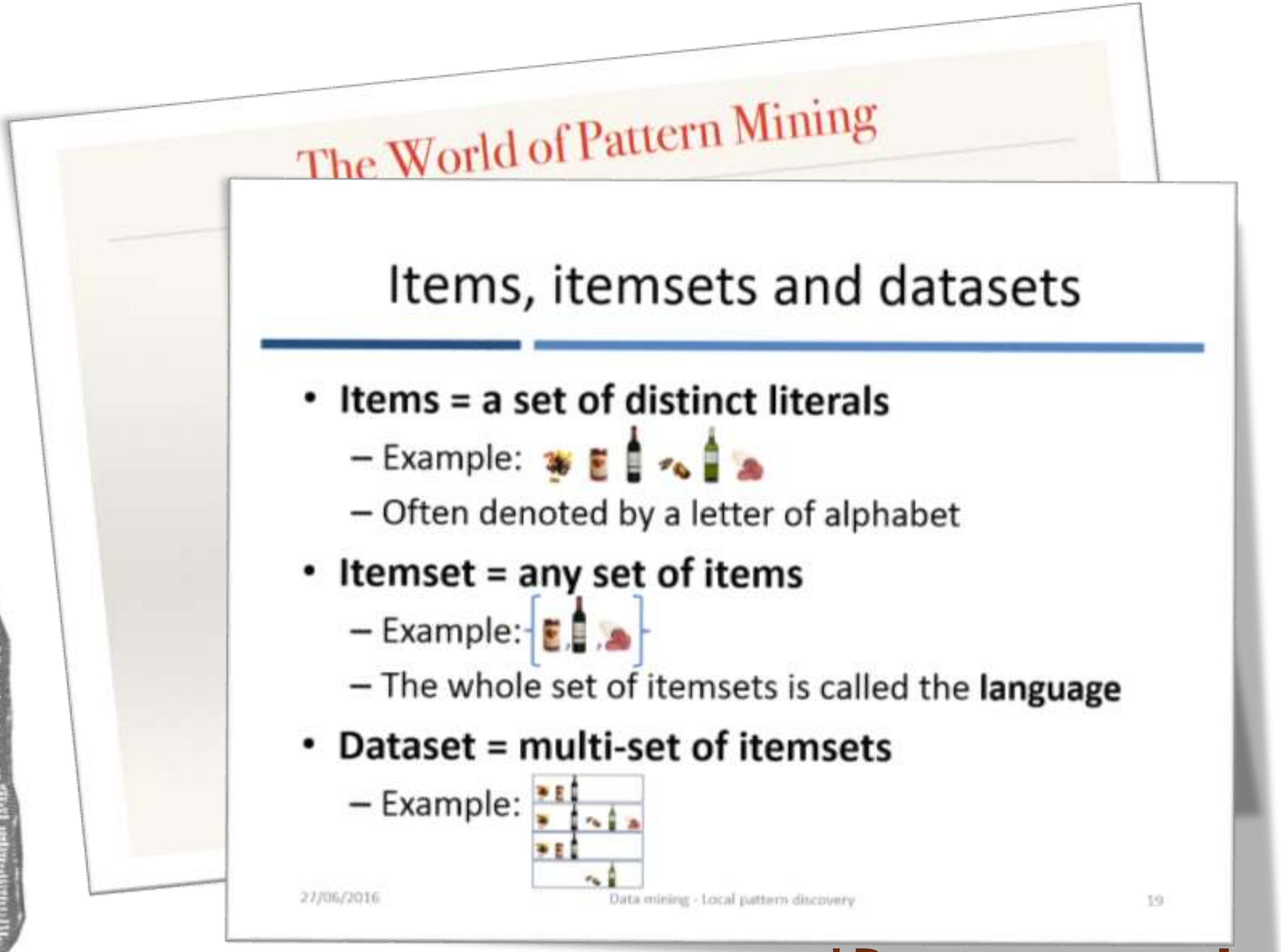
Language?



*Do you remember?

1

Language?



*Do you remember?

1

Language?

The World of Pattern Mining

Items, itemsets

Formal Notations of Sequence Data

- An *item* is denoted by lower-case letters

  a, b, c, d, …

- An *itemset* is denoted by upper-case letters

  $I = (abc)$, $I' = (acd)$, $I_1 = (bcef)$, …

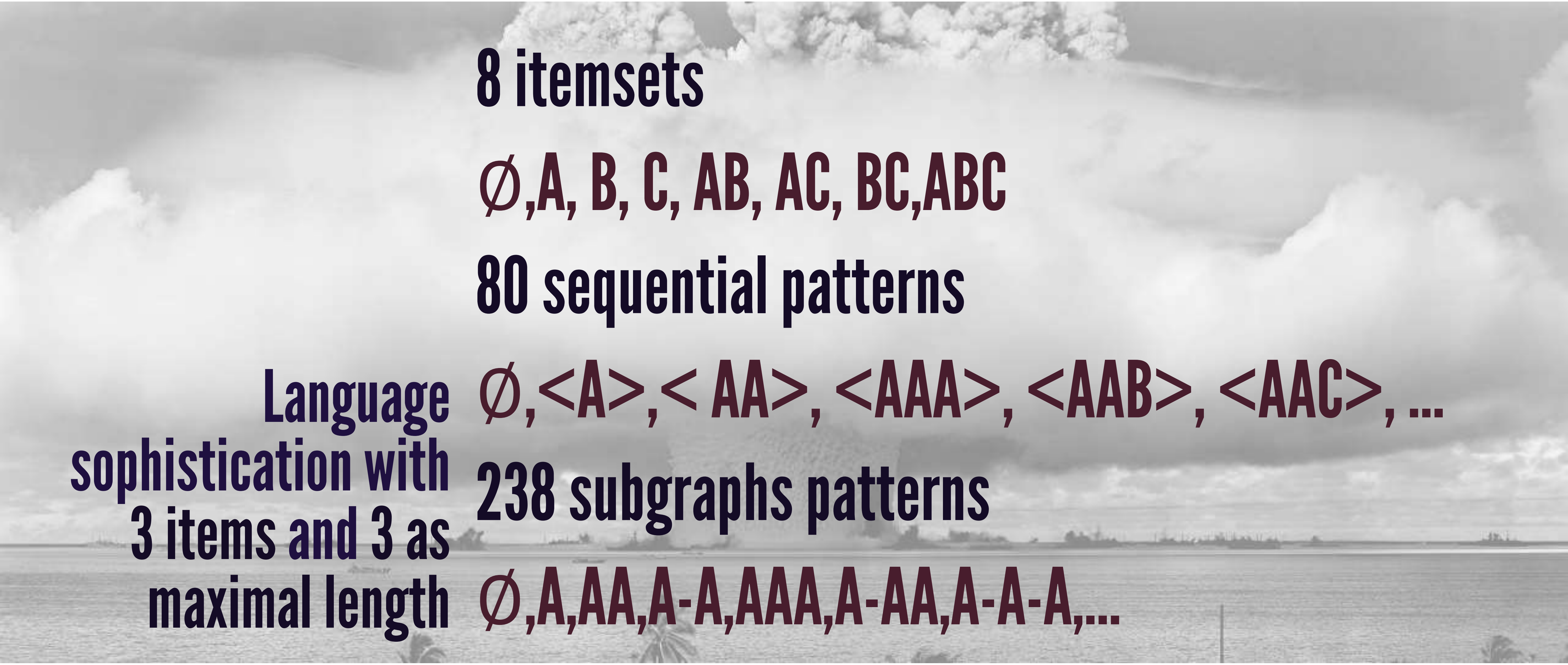
- A *sequence* is denoted by s (with prime and/or indice)

  $s = \langle I_1 I_2 I_3 I_4 \rangle$, $s' = \langle (ab)(c)(bcd)(ac) \rangle$, $s_1 = \langle (ab)c(bc) \rangle$…

  (We may ignore the parentheses for 1-item itemsets)

- A *sequence database* is a (large) set of sequences

ber?



8 itemsets

∅,A, B, C, AB, AC, BC,ABC

80 sequential patterns

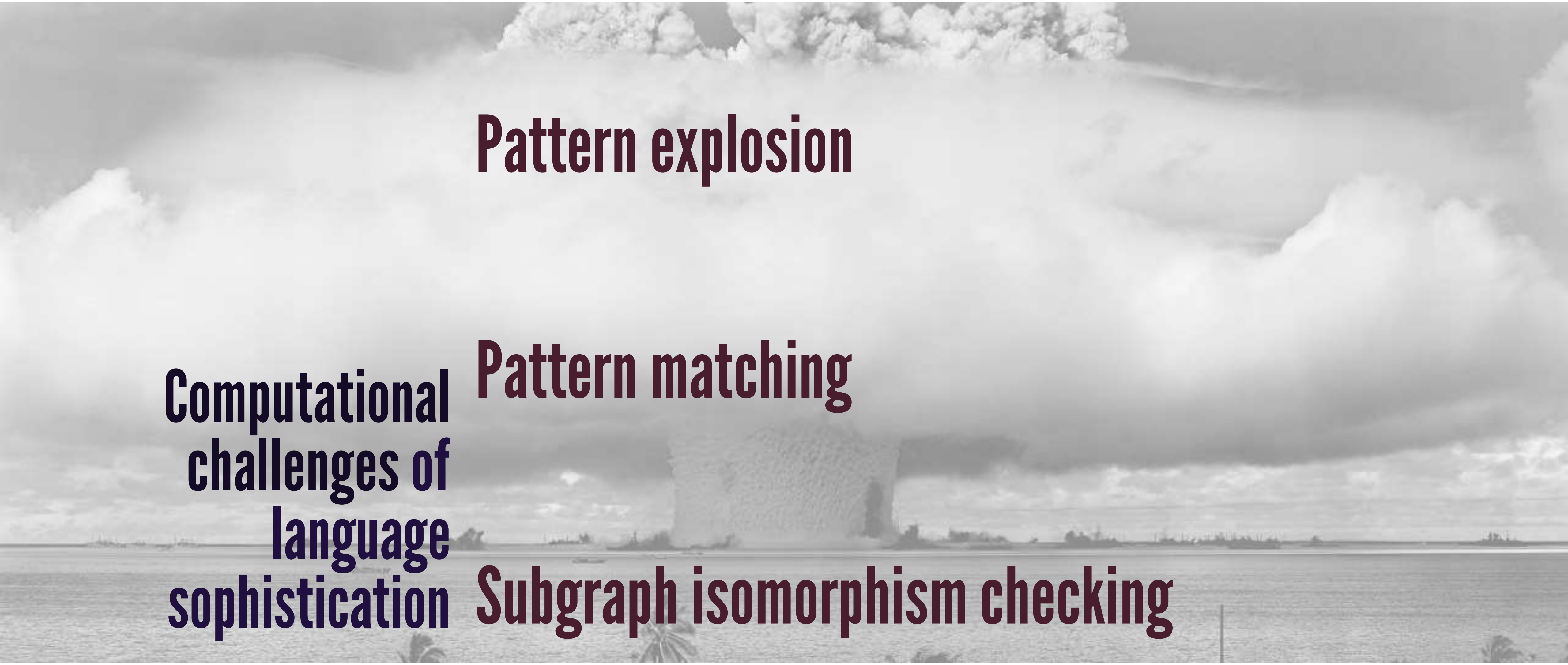
∅,<A>,< AA>, <AAA>, <AAB>, <AAC>, ...

238 subgraphs patterns

∅,A,AA,A-A,AAA,A-AA,A-A-A,...

Language sophistication with 3 items and 3 as maximal length

*Pattern explosion

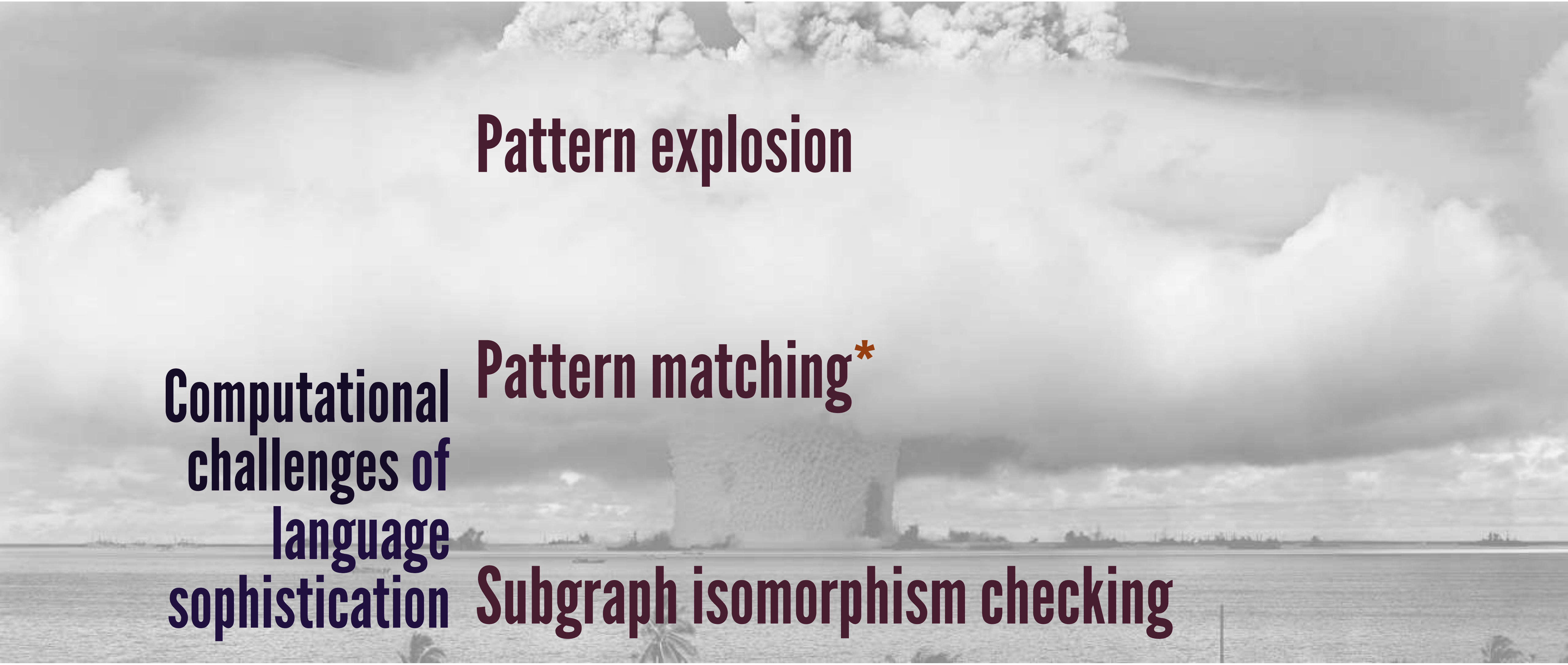


Computational challenges of language sophistication

Pattern explosion

Pattern matching

Subgraph isomorphism checking



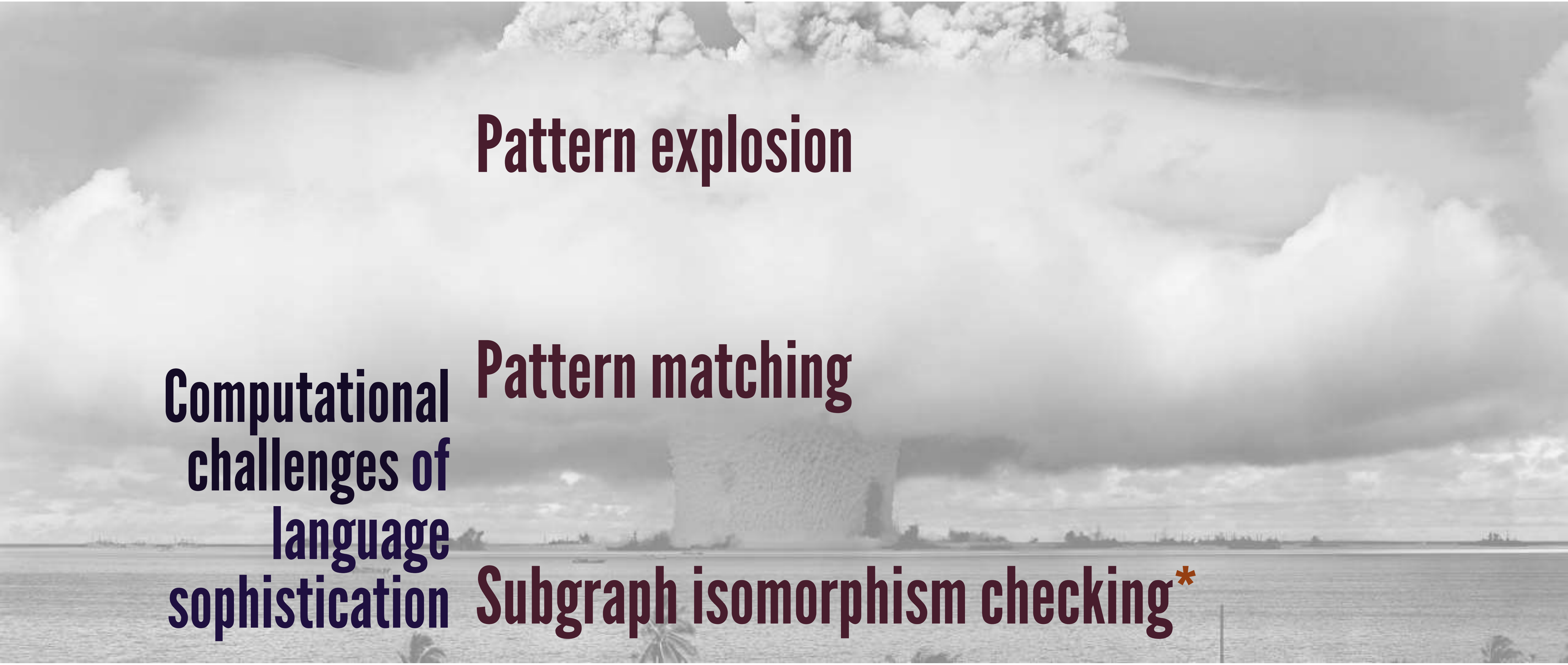
Computational challenges of language sophistication

Pattern explosion

Pattern matching*

Subgraph isomorphism checking

*Does a database entry contain a pattern?



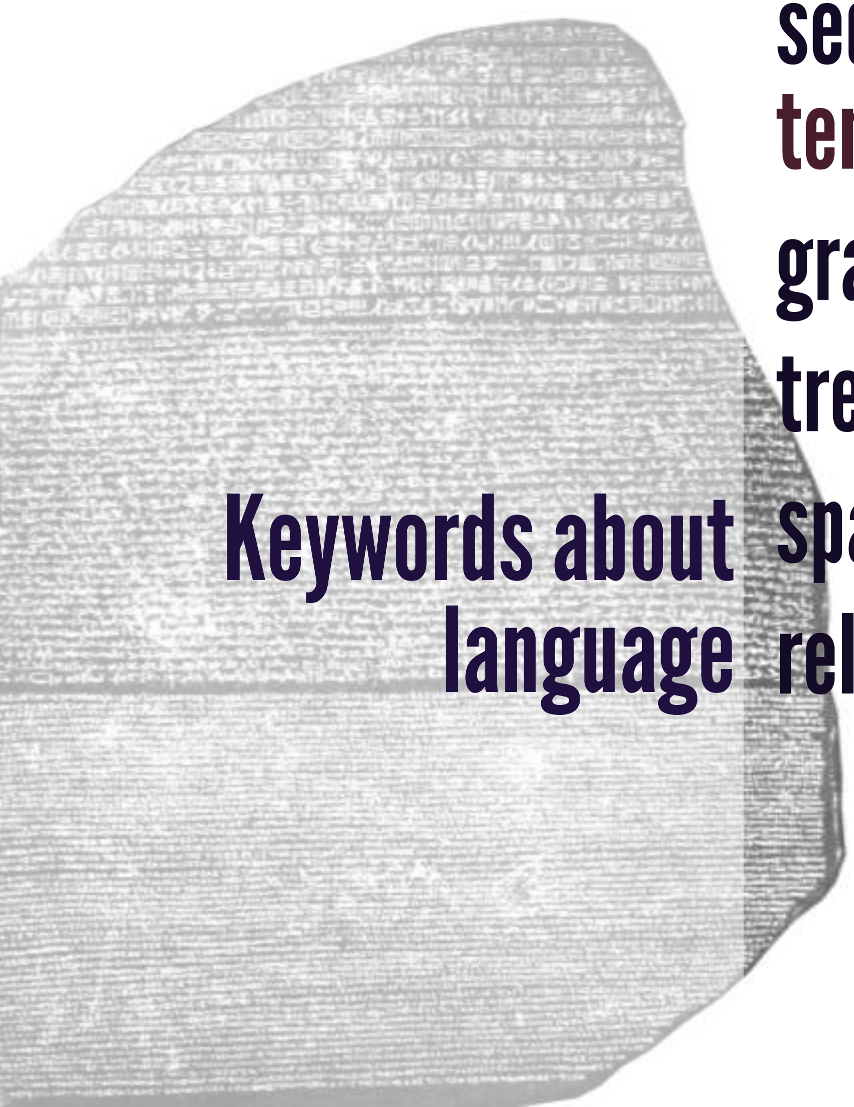
Computational challenges of language sophistication

Pattern explosion

Pattern matching

Subgraph isomorphism checking*

*Does a graph contain a subgraph isomorphic to another graph?



itemset, set

rule, association

sequence, episode, string, stream, protein, periodic, temporal

graph, molecular, structure, network

tree, xml

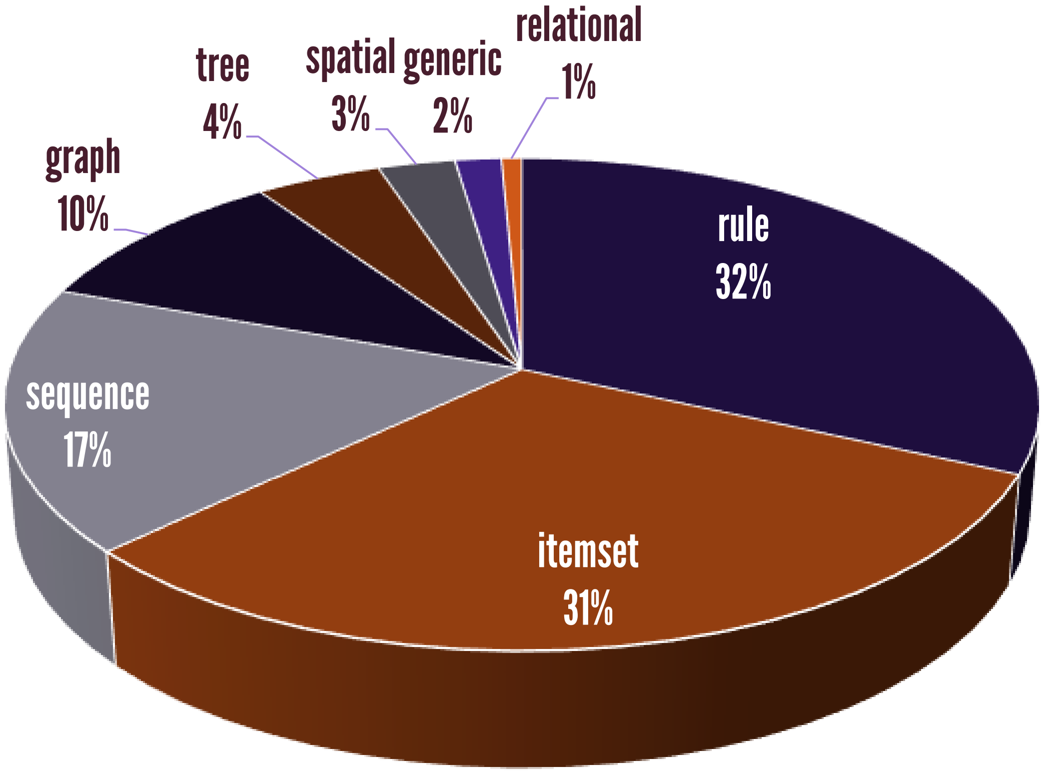
Keywords about language

spatial, spatio-temporal

relational

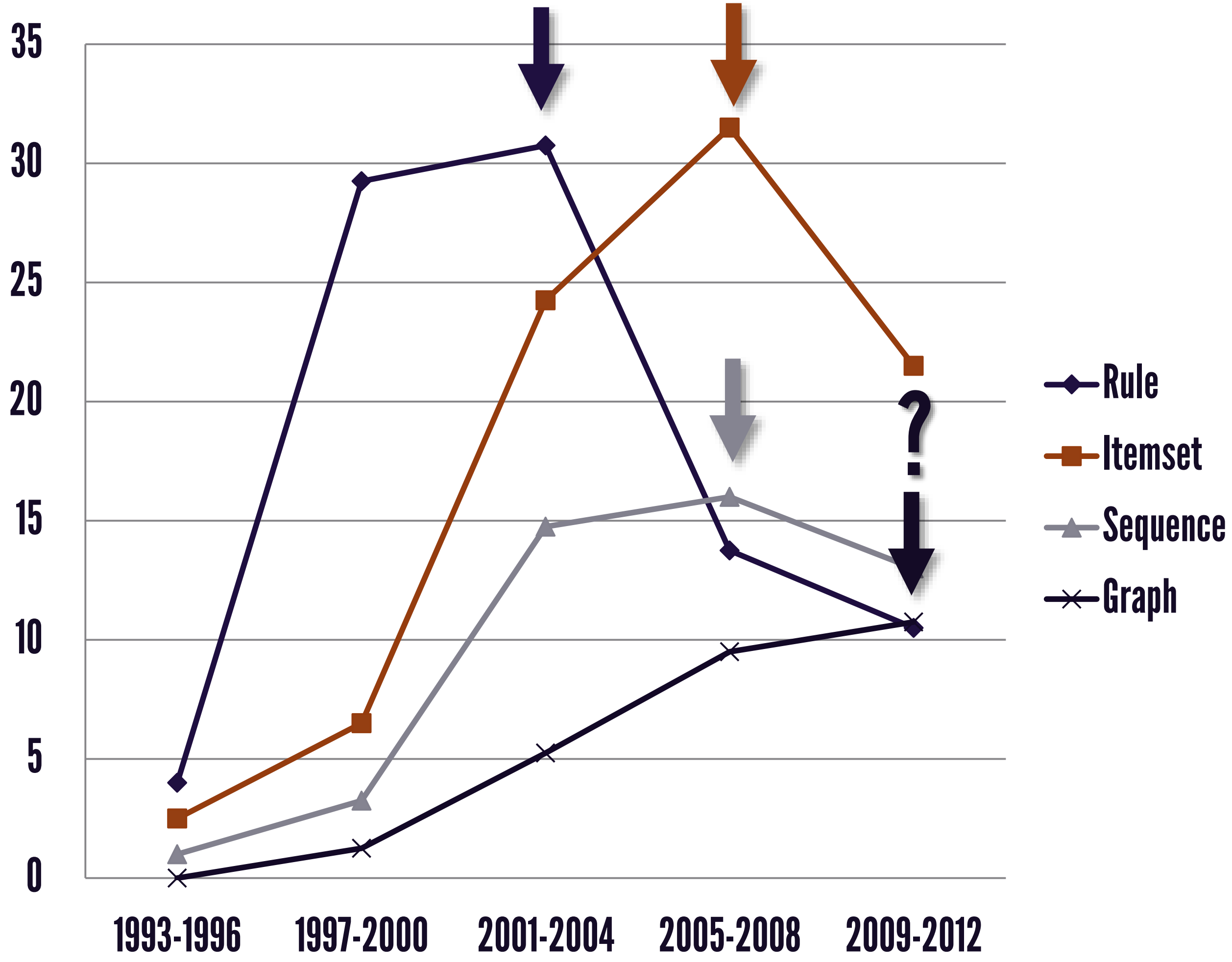
*Semi-automated topic assignment

Itemsets or rules for 63% of publications



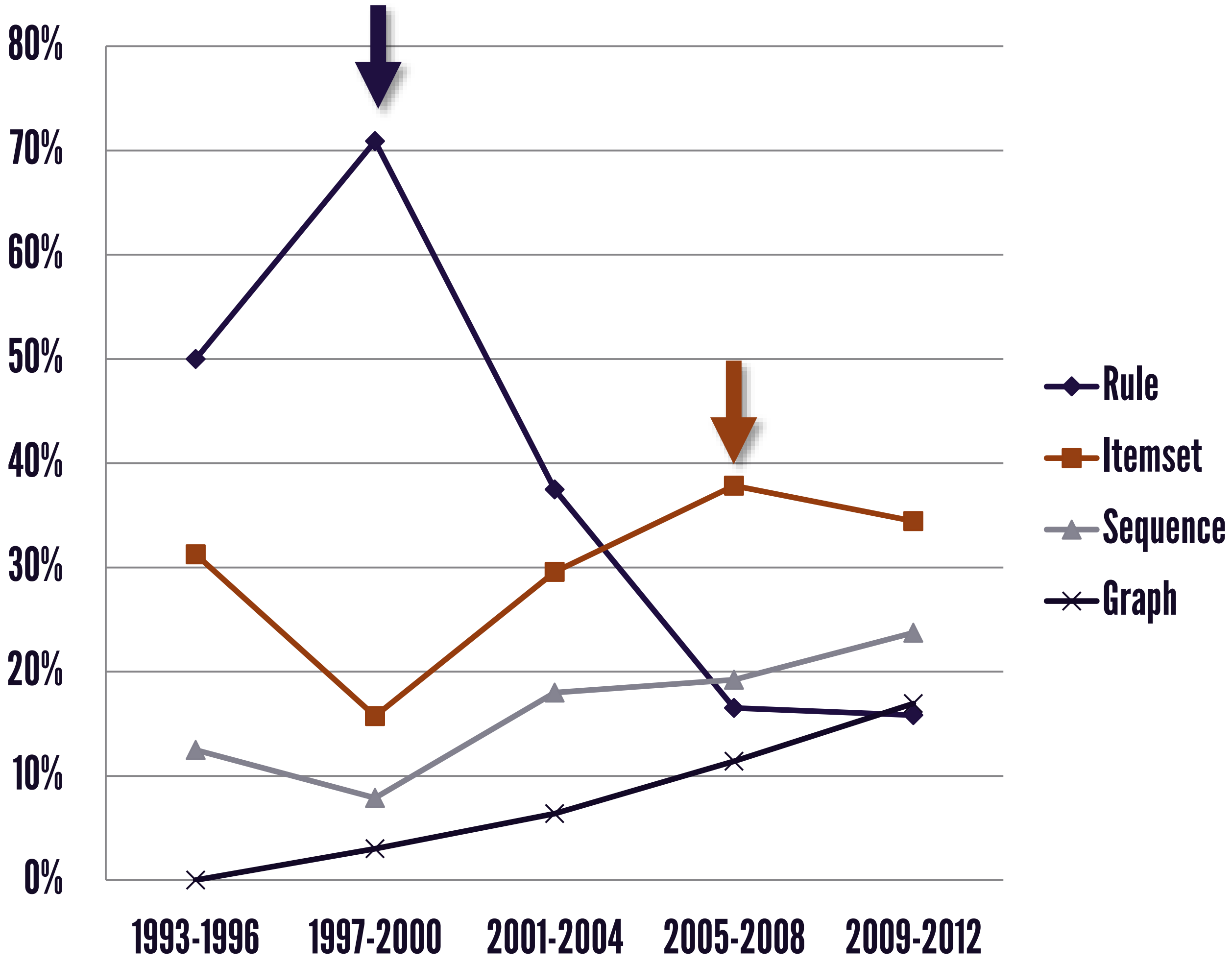
*Only 18 papers with generic language

Sophistication and marginalization of languages

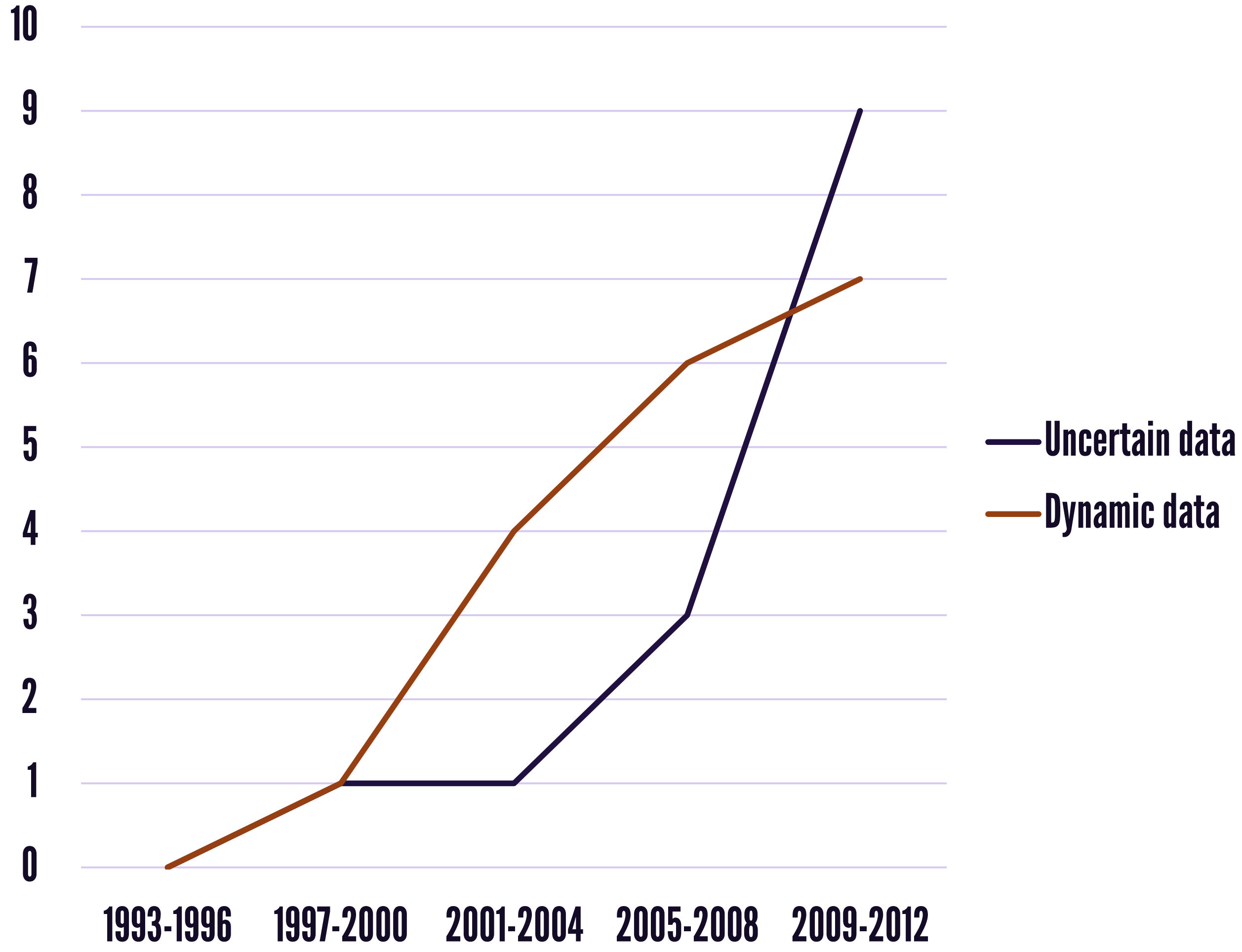


*Next language: spatio-temporal patterns?

Progress of sequences and graphs

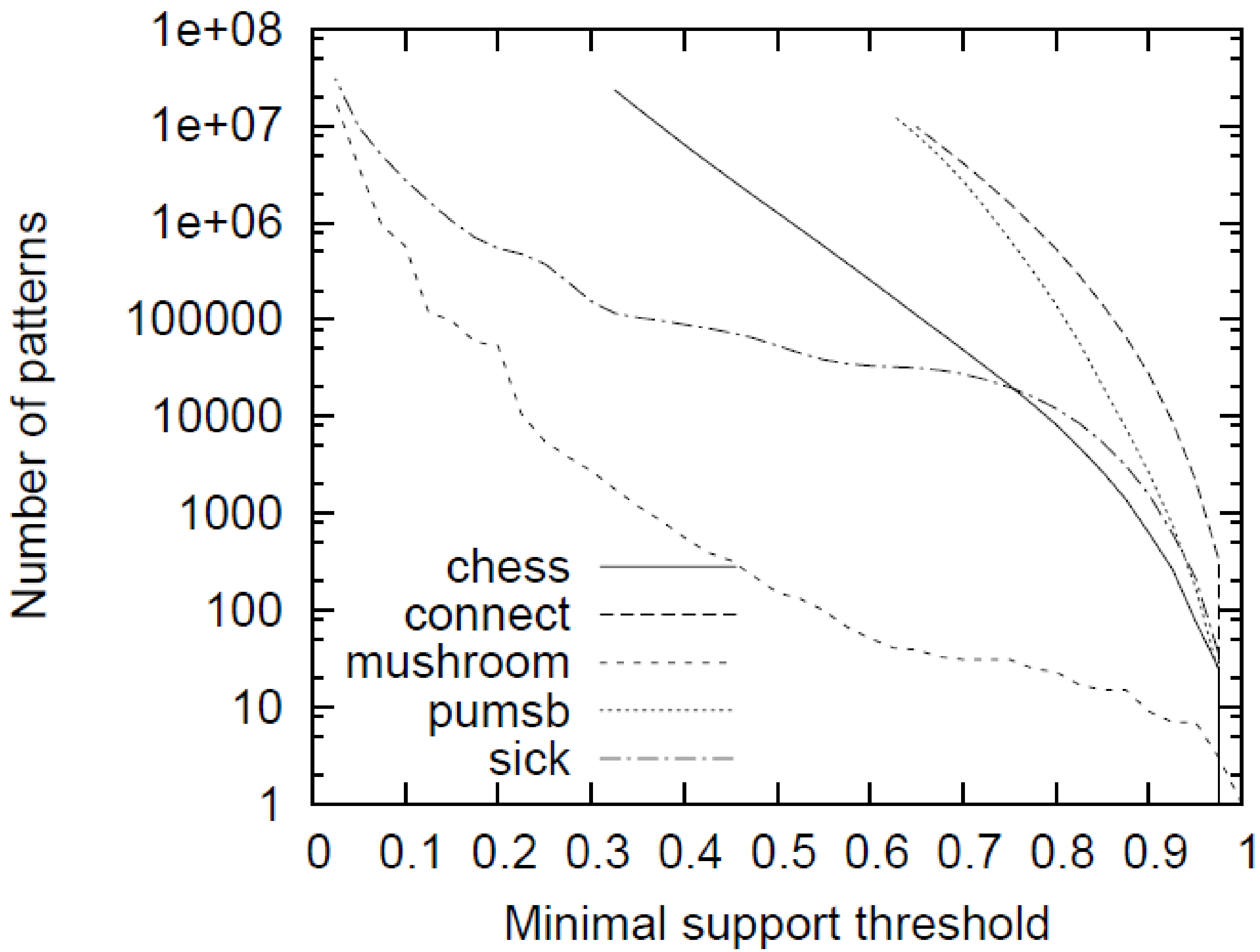


Next challenge: complex data



*Uncertain, dynamic, massive, heterogeneous data

Pattern explosion of frequent patterns (even with itemsets)



*How to reduce the number of patterns?

Two strategies against pattern explosion

Useful Patterns (UP'10)
ACM SIGKDD Workshop Report

Jilles Vreeken
Department of Mathematics and Computer Science
University of Antwerp
Belgium
jilles.vreeken@ua.ac.be

Nikolaj Tatti
Department of Mathematics and Computer Science
University of Antwerp
Belgium
nikolaj.tatti@ua.ac.be

Bart Goethals
Department of Mathematics and Computer Science
University of Antwerp
Belgium
bart.goethals@ua.ac.be

ABSTRACT

1. MOTIVATION

Pattern mining in action, an example

Making pattern mining useful

[Vreeken et al., 2010]

Constraint

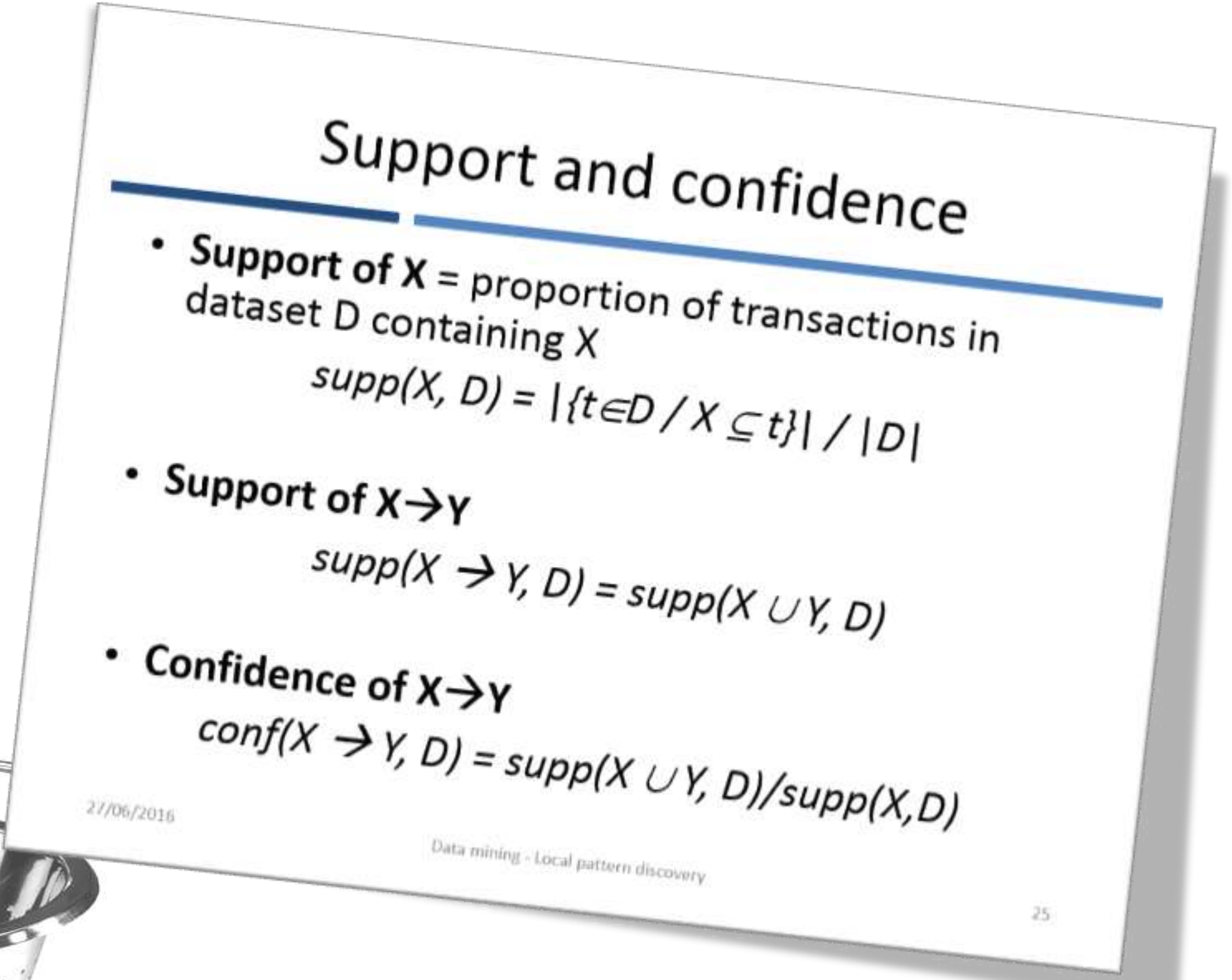
Focusing on the most useful patterns for the data expert

Condensed Representation

Removing all redundant patterns

*Useful Patterns (UP'10) ACM SIGKDD Workshop

2


Support and confidence

- **Support of X** = proportion of transactions in dataset D containing X

$$supp(X, D) = |\{t \in D \,/\, X \subseteq t\}| \,/\, |D|$$

- **Support of X→Y**

$$supp(X \rightarrow Y, D) = supp(X \cup Y, D)$$

- **Confidence of X→Y**

$$conf(X \rightarrow Y, D) = supp(X \cup Y, D)/supp(X,D)$$

27/06/2016

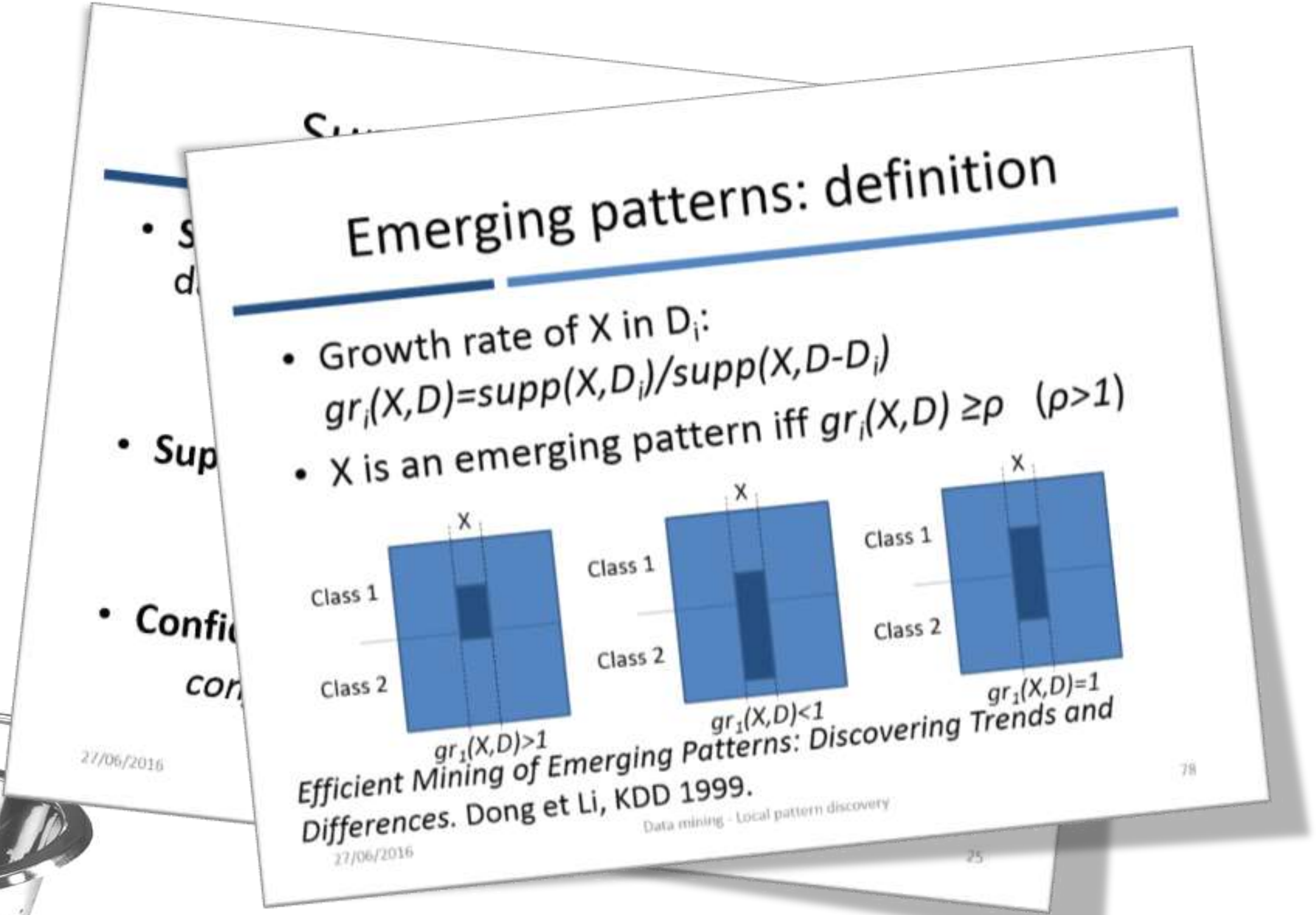
Data mining - Local pattern discovery

25

Constraint?

*Do you remember, again?

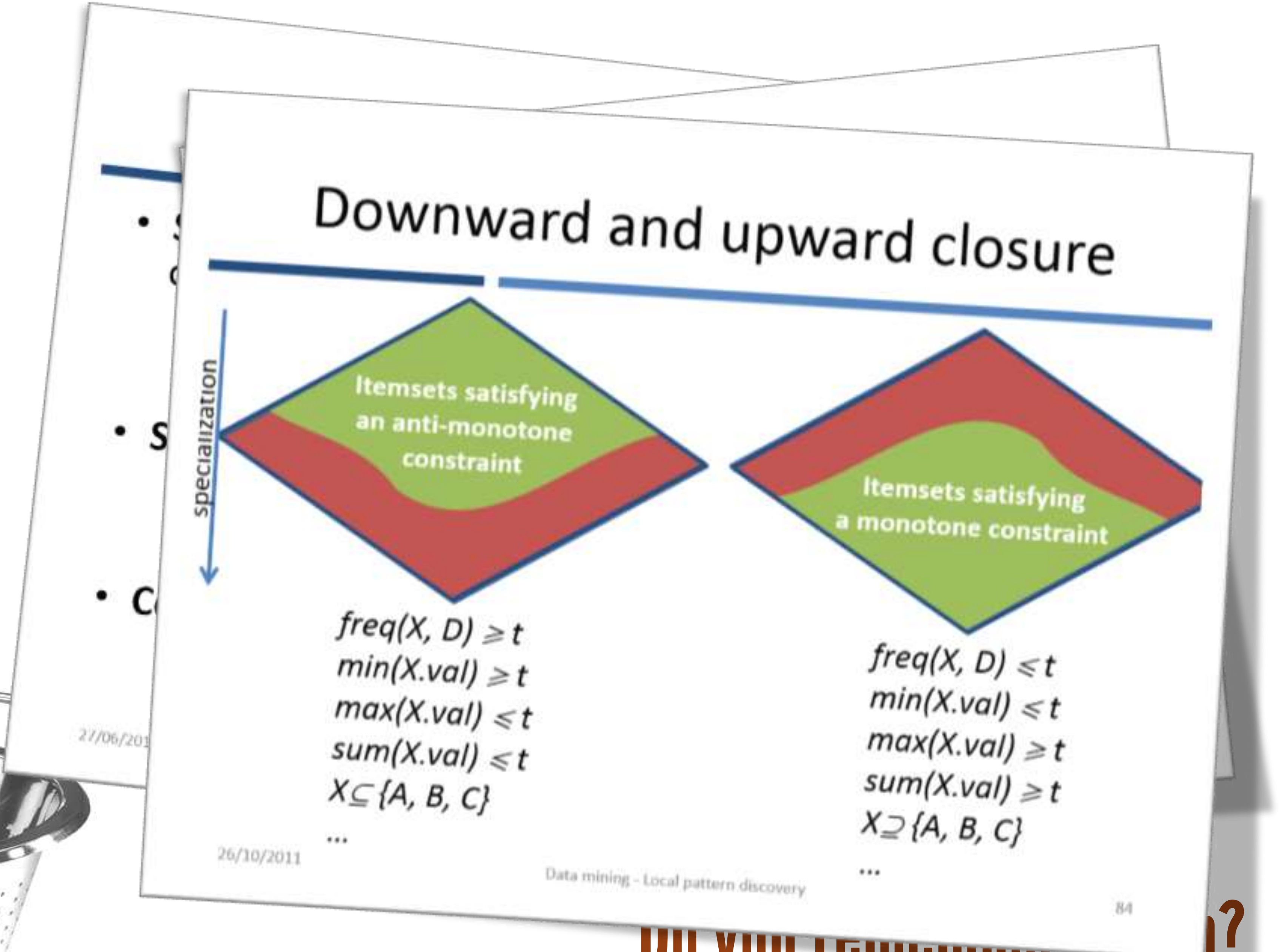
2



Constraint?

*Do you remember, again?

2



Constraint?

Interest of constraints

Troubled romantic ➔ Rich
(conf = 0.5)

Troubled romantic ➔ Dies
(conf = 0.75)

Troubled romantic → Rich

(conf = 0.5 / lift = 0,8)

Troubled romantic → Dies

(conf = 0.75 / lift = 0,6)

Interest of constraints

	Troubled Romantic	Rich	Dies	Hidding Secret
	A		C	
	A			D
	A	B	C	
		B	C	
	A	B	C	D

$$freq(X) \geq minfreq$$

Ø

A B C D

AC AB BC BD AD CD

ABC ABD ACD BCD

ABCD

Challenge of constraints

*How to prune the search space for frequency?

	Troubled Romantic	Rich	Dies	Hidding Secret
	A		C	
	A			D
	A	B	C	
		B	C	
	A	B	C	D

$$freq(X) \geq minfreq$$

Ø

A B C D

AC AB BC BD AD CD

ABC ABD ACD BCD

ABCD

Challenge of constraints

*How to prune the search space for frequency?

Easy due to the downard closure

	Troubled Romantic	Rich	Dies	Hidding Secret
	A		C	
	A			D
	A	B	C	
		B	C	
	A	B	C	D

$$freq(X) \times |X| \geq minarea$$

Ø

A B C D

AC AB BC BD AD CD

ABC ABD ACD BCD

ABCD

Challenge of constraints

*How to prune the search space for area?

	Troubled Romantic	Rich	Dies	Hidding Secret
	A		C	
	A			D
	A	B	C	
		B	C	
	A	B	C	D

$$freq(X) \times |X| \geq minarea$$

Ø

A B C D

AC AB BC BD AD CD

ABC ABD ACD BCD

ABCD

Challenge of constraints

*How to prune the search space for area?

Relax the area constraint by $freq(X) \times 4 \geq minarea$

regularity, frequent, support

contrast, emerging, discriminative

exception, abnormal, surprising, anomaly, unexpected

utility

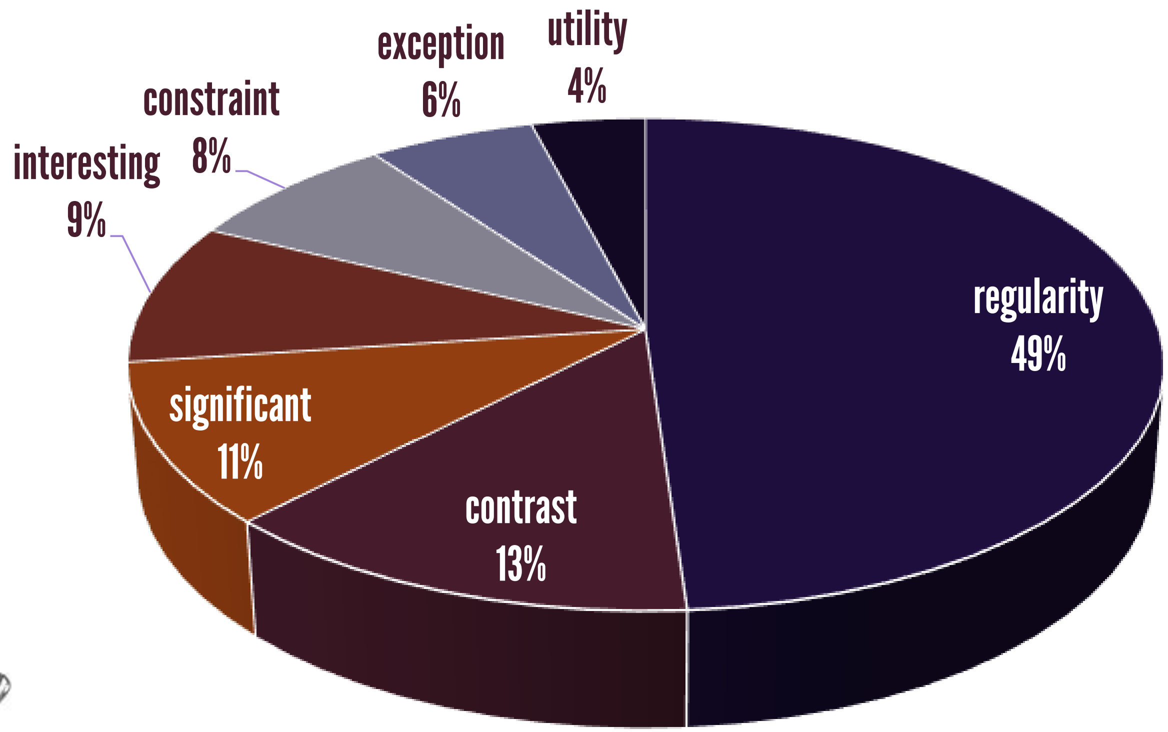
significant, chi-square, correlated

interesting, relevant

generic, monotone, anti-monone, constrained

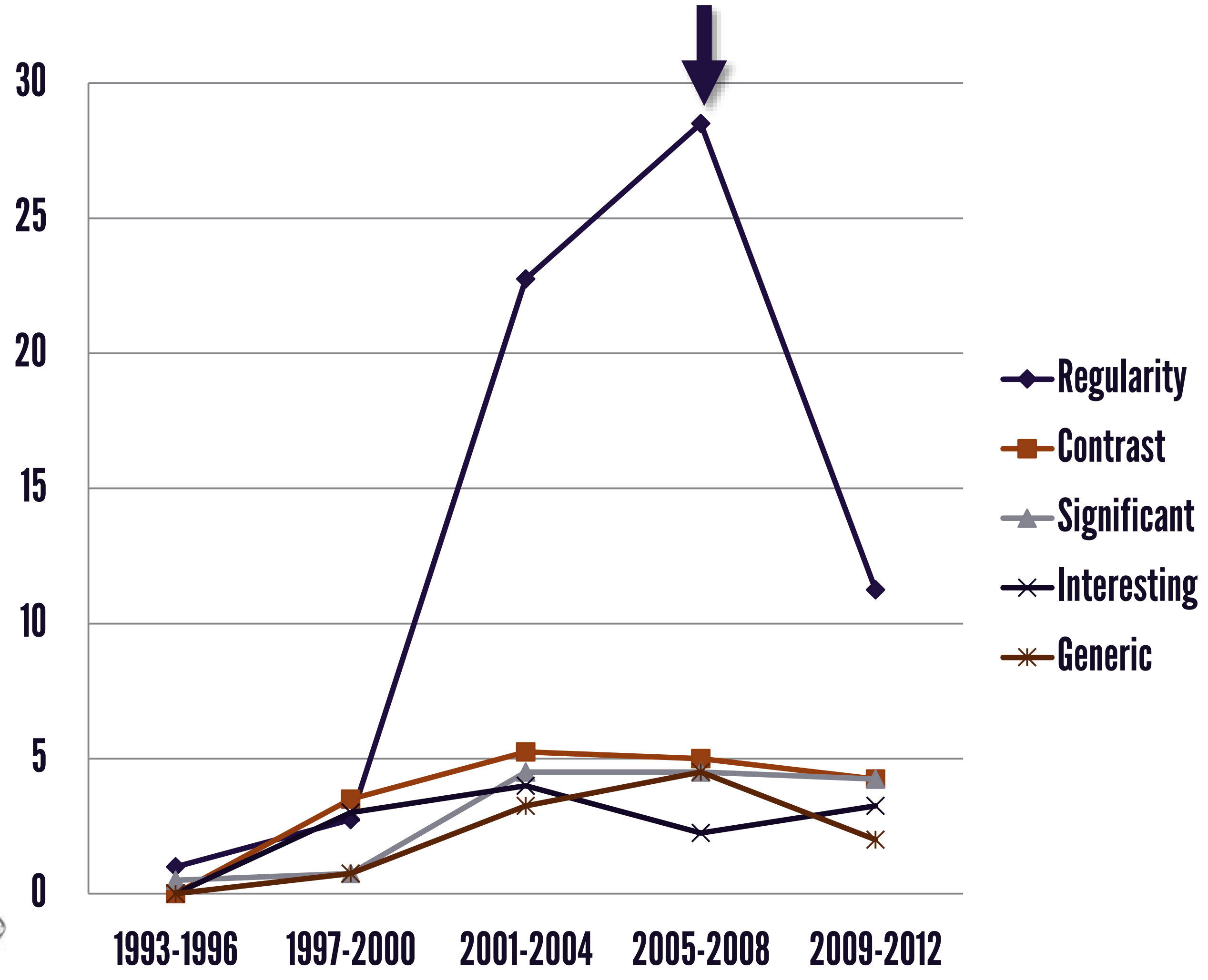
Keywords about constraints

*Semi-automated assignment topic



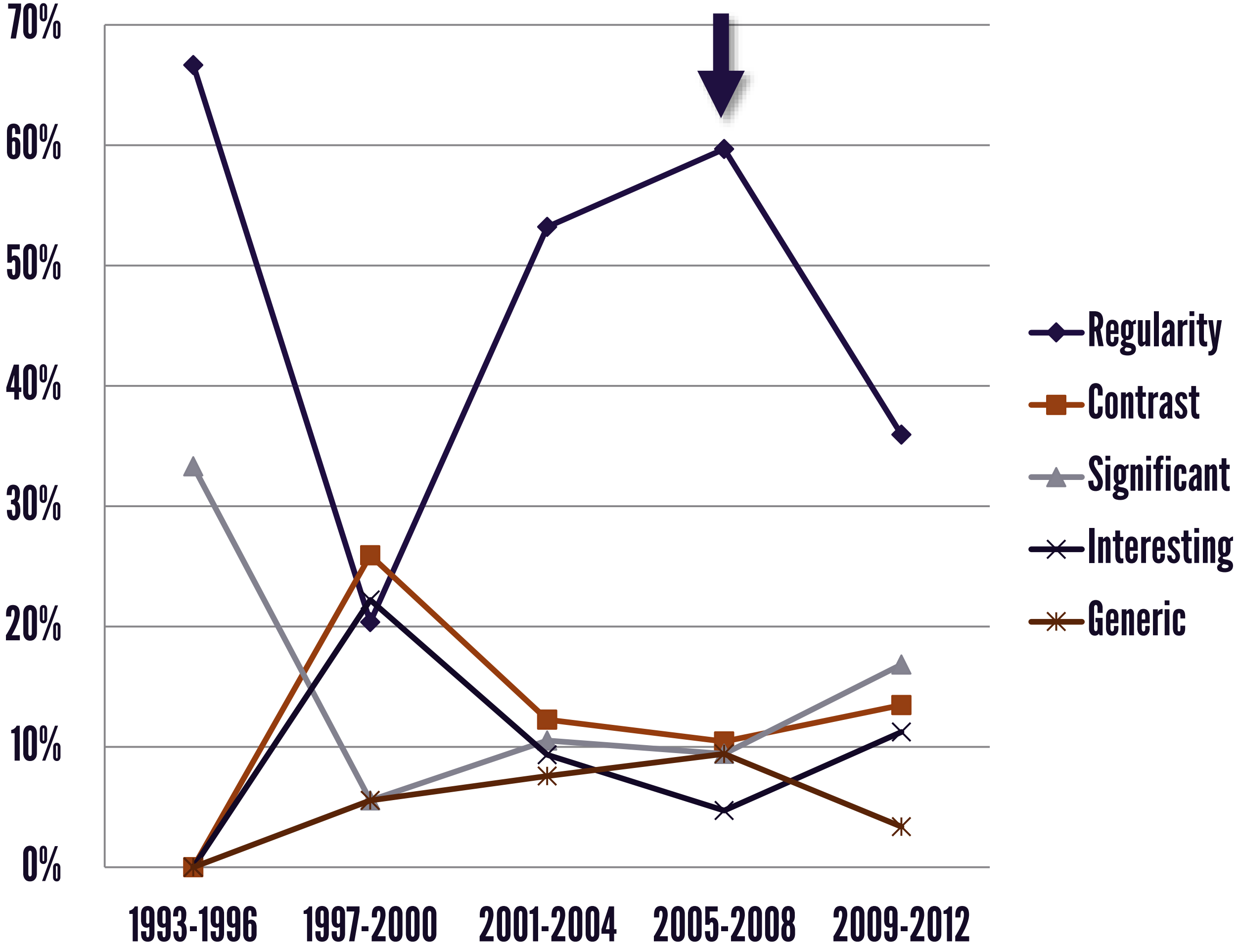
Frequent patterns in 49% of publications

*Speed preferred to quality



Stable except for frequency

Progress of contrastive and significant patterns



*Only 42 papers with generic constraints

Interestingness

Piatetsky-Shapiro
Agrawal
2003

Han et al.
2007

Rakesh Agrawal Speaks Out
on Where the Data Mining Field Is Going, Where It Came From, How to Choose Problems and Open Up New Fields, Our Responsibilities to Society as Technologists, What Industry Owes Academia, and More

by Marianne Winslett

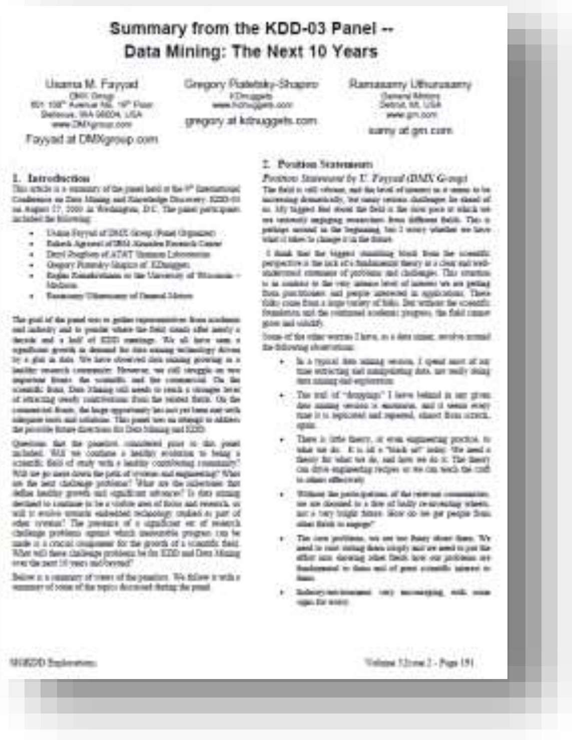
"we need work to bring in some notion of 'here is my idea of what is interesting,' and pruning the generated rules

based on that input."

Interestingness

Piatetsky-Shapiro
Agrawal
2003

Han et al.
2007


Summary from the KDD-03 Panel --
Data Mining: The Next 10 Years

Usama M. Fayyad
Fayyad at DMXgroup.com

Gregory Piatetsky-Shapiro
gregory at kdnuggets.com

Ramasamy Uthurusamy
samy at gm.com

1. Introduction

2. Position Statements

SIGKDD Explorations.

The Grand Challenges Today

“A General Theory of Interestingness. What makes this rule, pattern more interesting than another?”

Interestingness

Piatetsky-Shapiro
Agrawal
2003

Han et al.
2007

Data Min Knowl Disc (2007) 15:55–86
DOI 10.1007/s10618-006-0059-1

Frequent pattern mining: current status and future directions

Jiawei Han · Hong Cheng · Dong Xin · Xifeng Yan

Received: 22 June 2006 / Accepted: 8 November 2006 / Published online: 27 January 2007

Abstract Frequent pattern mining has been a focused theme in data mining research for over a decade. Abundant literature has been dedicated to this research and tremendous progress has been made, ranging from efficient and scalable algorithms for frequent itemset mining in transaction databases to numerous research frontiers, such as sequential pattern mining, structured pattern mining, correlation mining, associative classification, and frequent pattern-based clustering, as well as their broad applications. In this article, we provide a brief overview of the current status of frequent pattern mining and discuss a few promising research directions. We believe that frequent pattern mining research has substantially broadened the scope of data analysis and will have deep impact on data mining methodologies and applications in the long run. However, there are still some challenging research issues that need to be solved before frequent pattern mining can claim a cornerstone approach in data mining applications.

Keywords Frequent pattern mining · Association rules · Data mining research · Applications

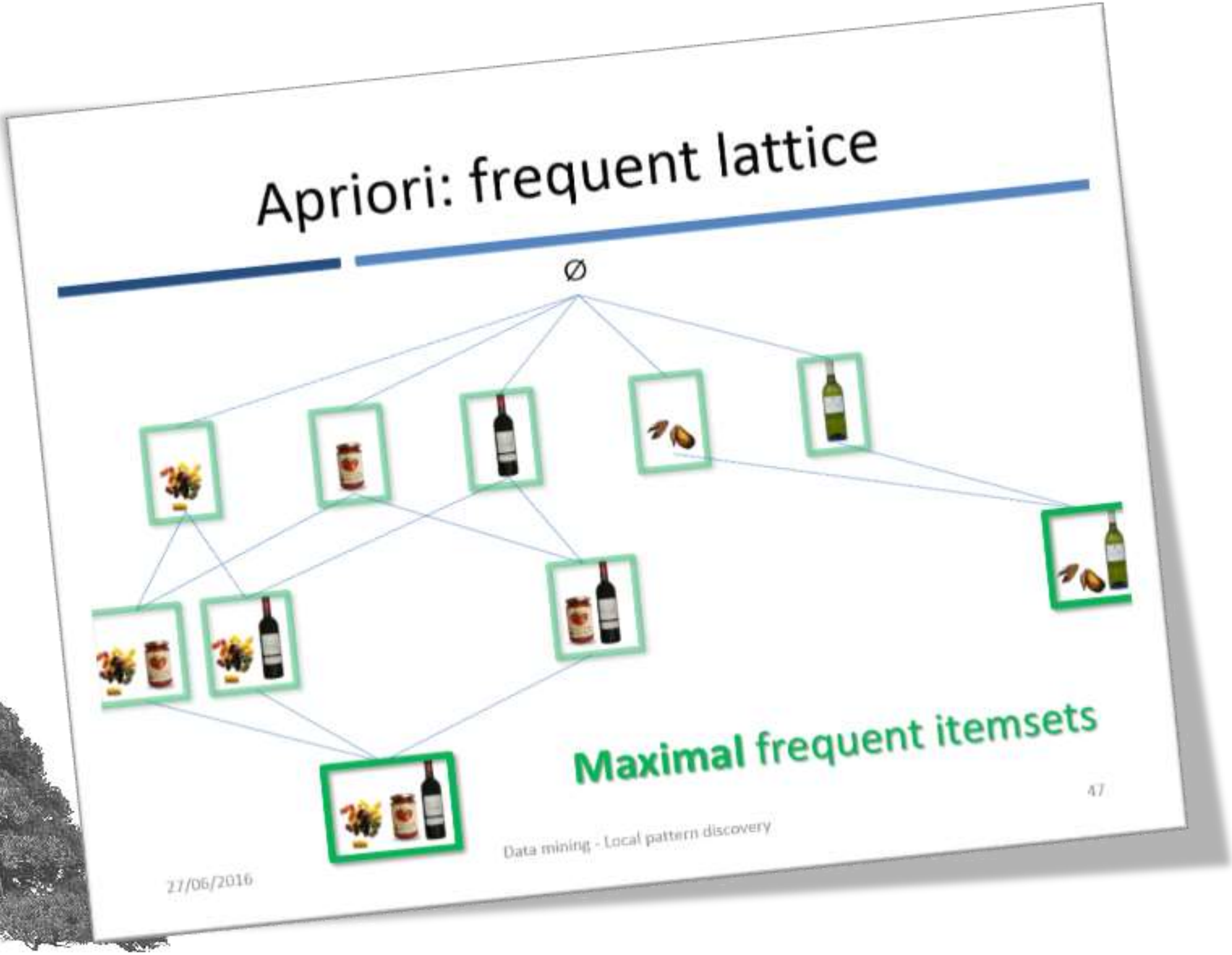
"it is still not clear what kind of patterns will give us satisfactory pattern sets in both compactness and representative quality"

1993

now*

*Quality of solution more important than speed of answer

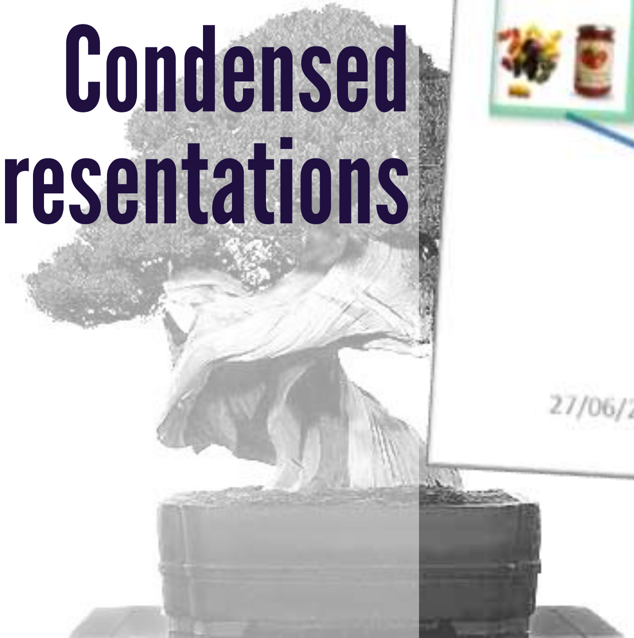
3



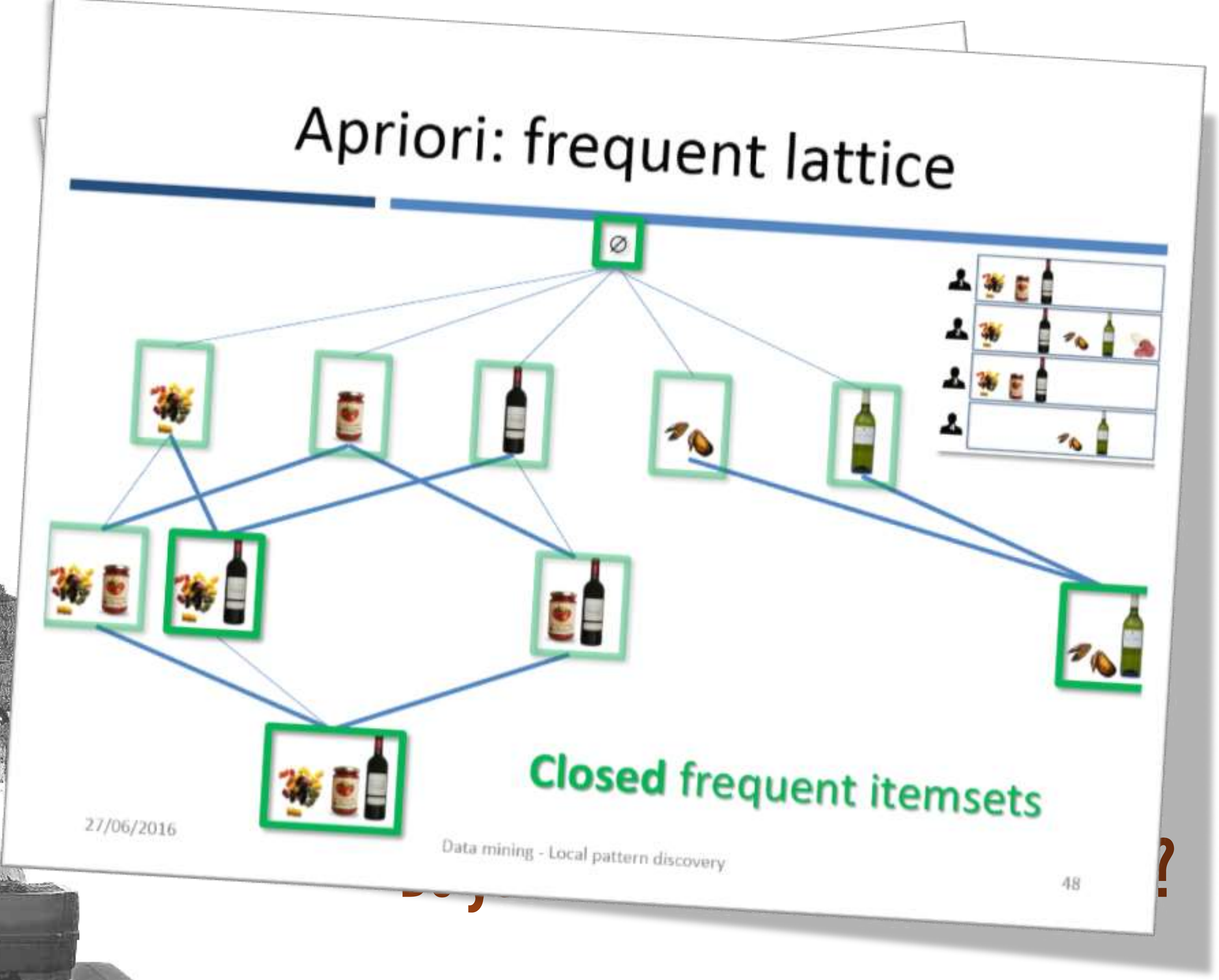
Condensed representations

*Do you remember, again and again?

3

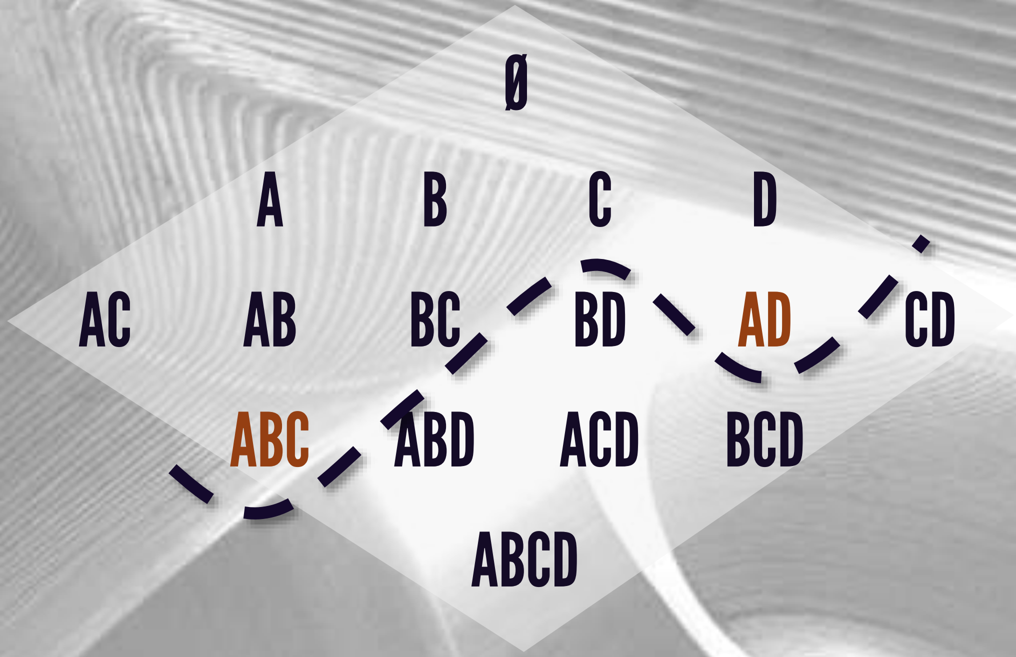


Condensed representations



Troubled Romantic	Rich	Dies	Hidding Secret
A		C	
A			D
A	B	C	
	B	C	
A	B	C	D

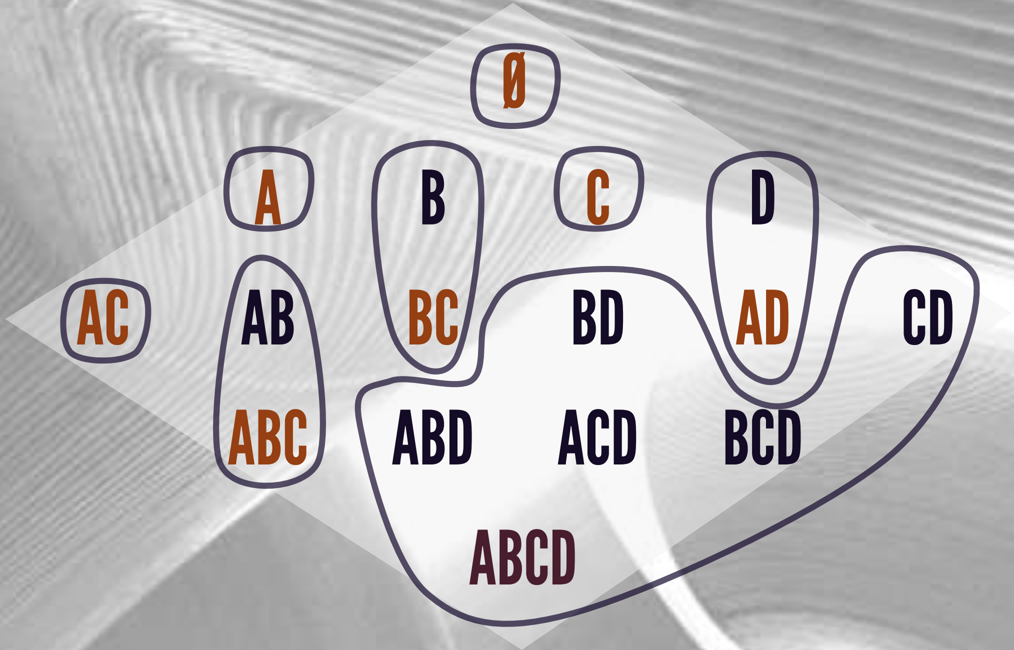
Maximal patterns



2 patterns with minfreq = 2

*larger frequent patterns w.r.t inclusion

	Troubled Romantic	Rich	Dies	Hidding Secret
	A		C	
	A			D
	A	B	C	
		B	C	
	A	B	C	D



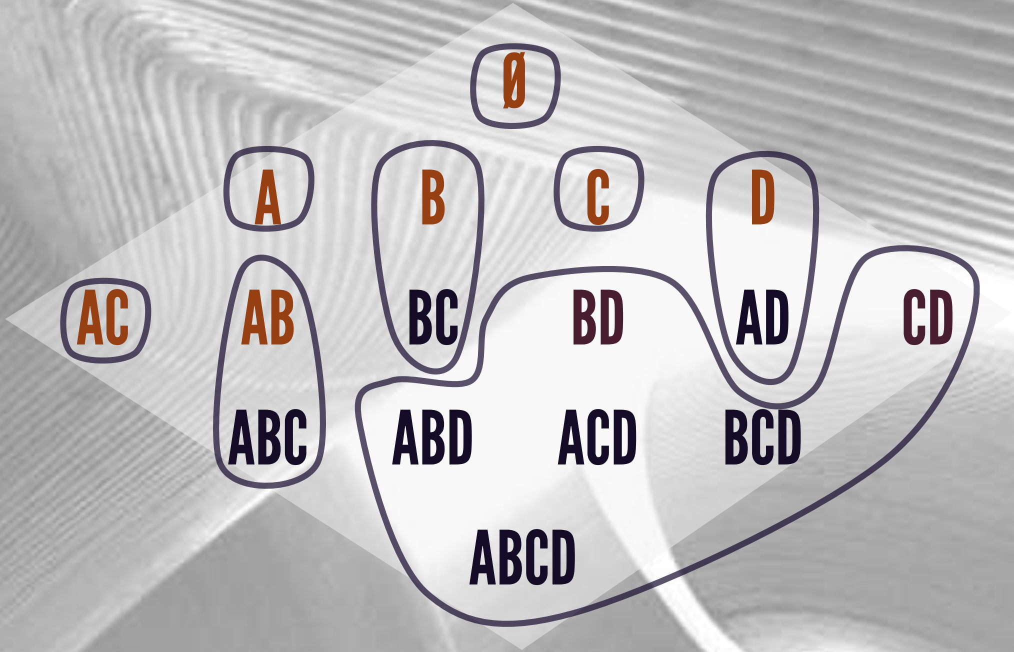
Closed patterns

7 patterns with minfreq = 2

*maximal patterns of equivalence classes

	Troubled Romantic	Rich	Dies	Hidding Secret
	A		C	
	A			D
	A	B	C	
		B	C	
	A	B	C	D

Free patterns



7 patterns with minfreq = 2

*minimal patterns of equivalence classes

More condensed representations

Non-derivable itemsets
2002

K-free itemsets
2003

Mining All Non-Derivable Frequent Itemsets

Toon Calders*
University of Antwerp, Belgium
Bart Goethals
University of Limburg, Belgium

Abstract

1 Introduction

[Calders and Goethals, 2002]

Minimal k-Free Representations of Frequent Sets

Toon Calders[1] and Bart Goethals[2]

1 Introduction

[Calders and Goethals, 2003]

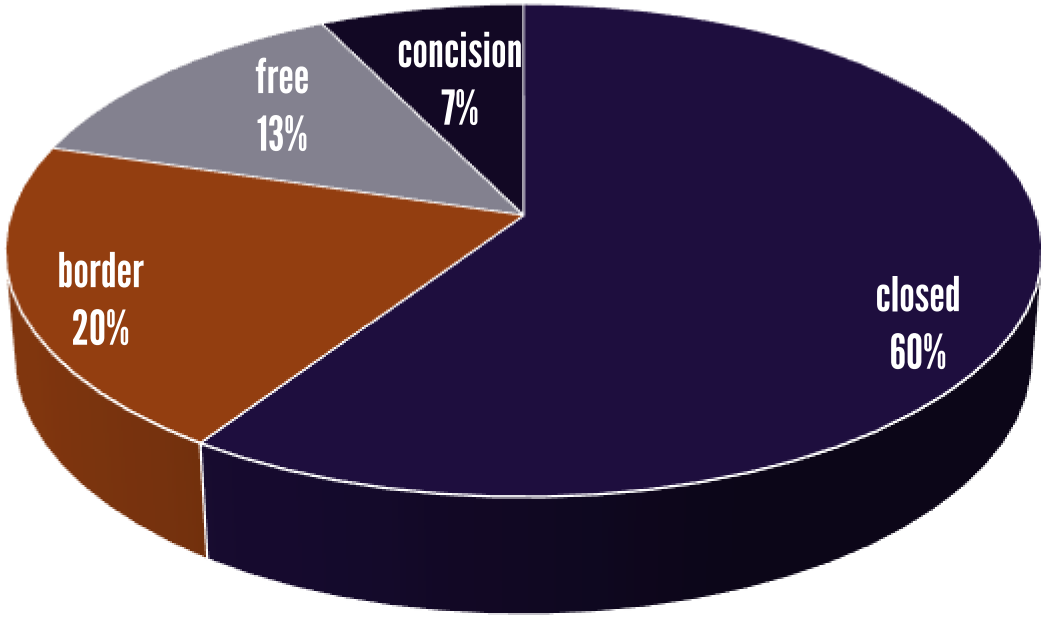
Keywords about condensed representations

closed

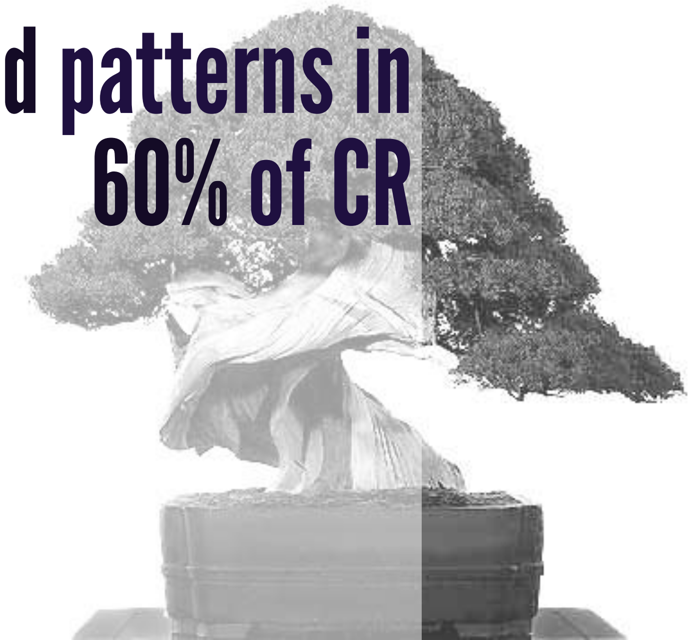
border, maximal, minimal

free, generator, non-derivable

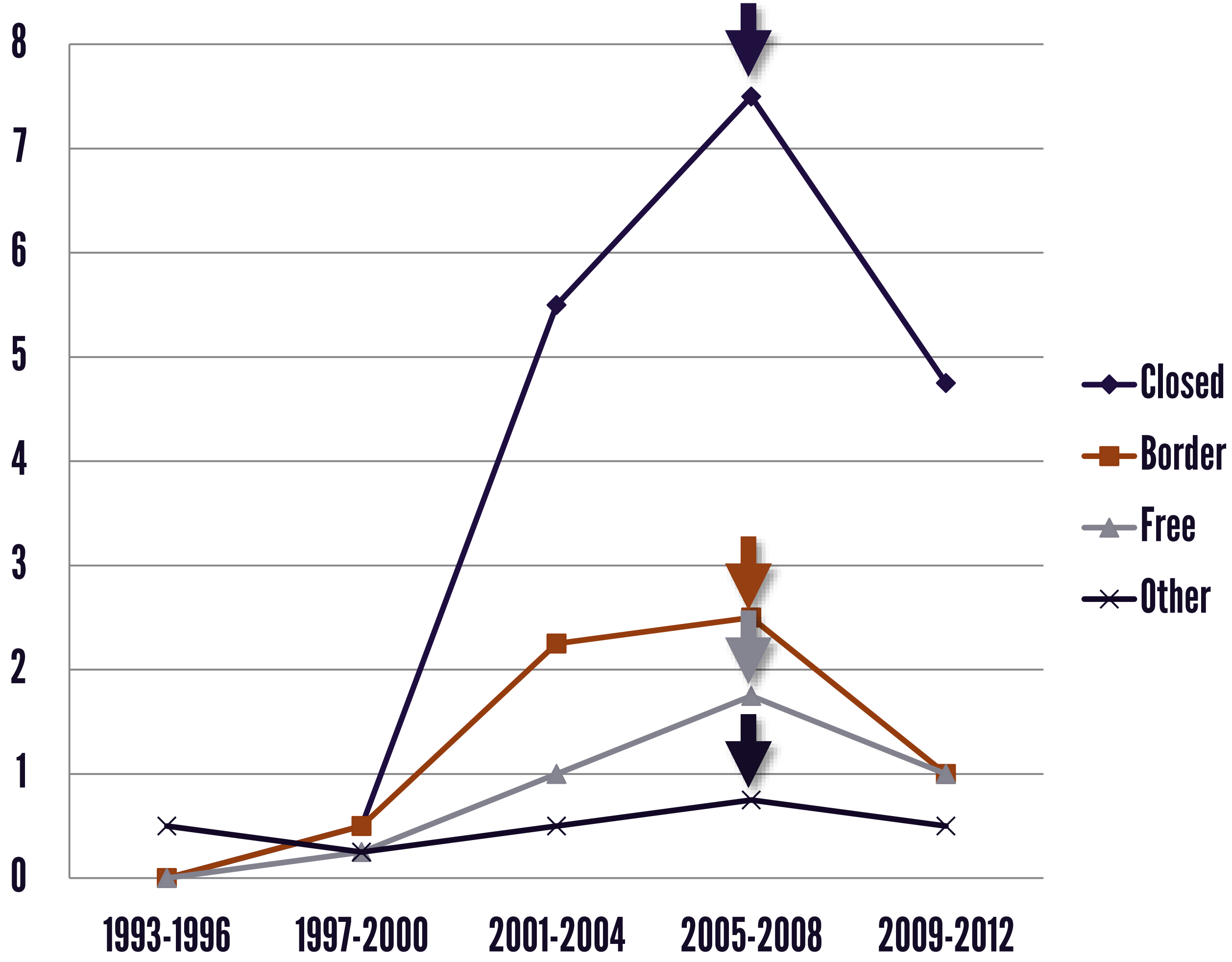
*Semi-automated topic assignment

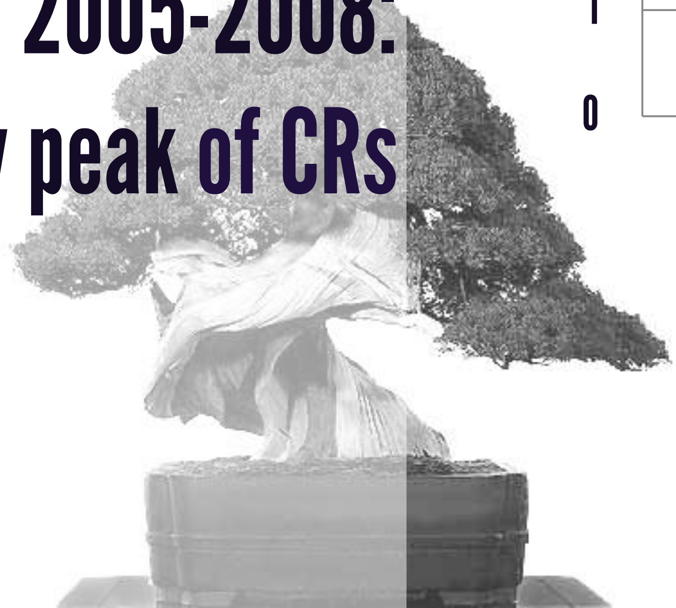


Closed patterns in 60% of CR



*Why this success?

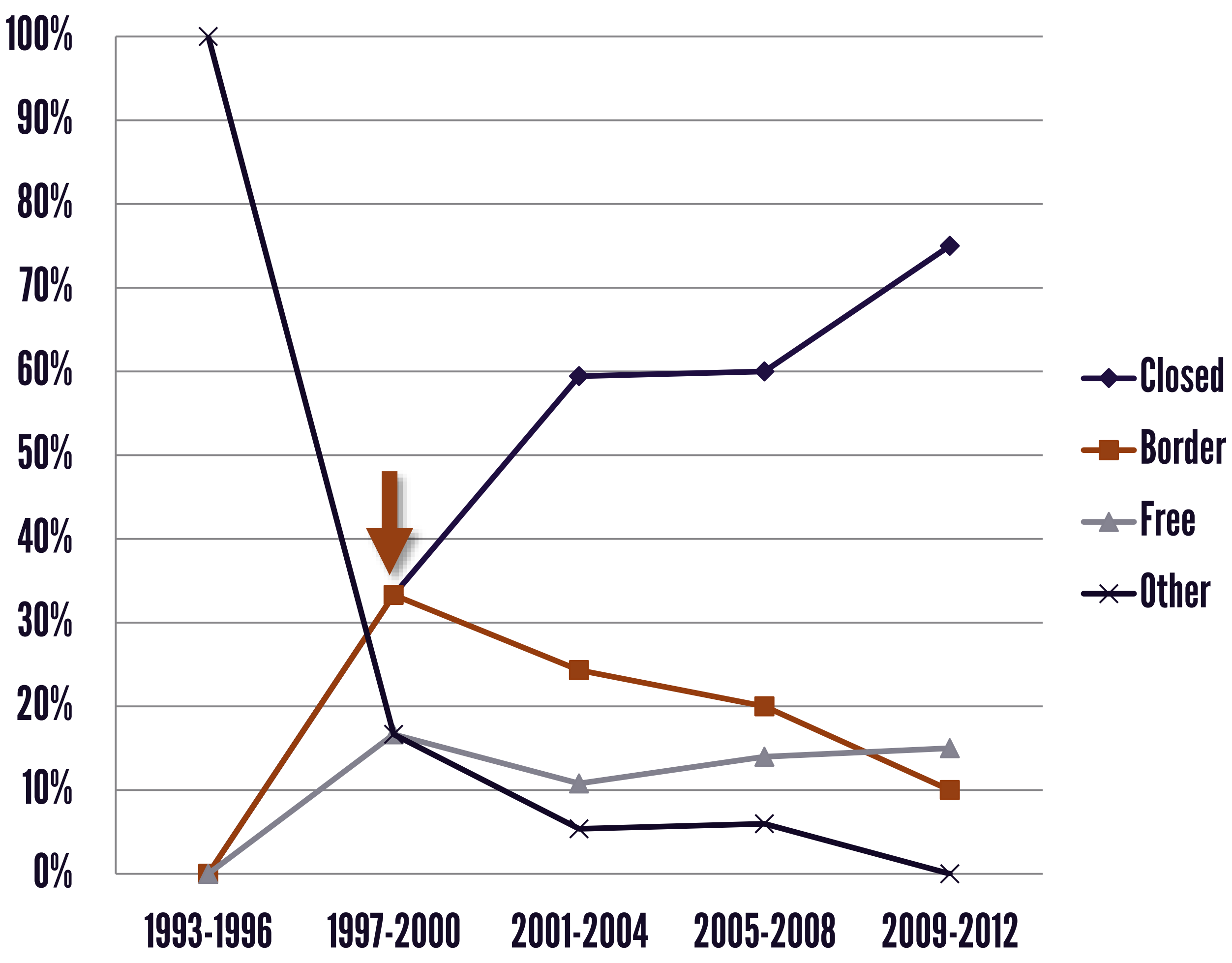




2005-2008:
Activity peak of CRs

*Pattern-based models instead of CRs

Progress of free itemsets



Pattern-based classifiers

CBA
1998

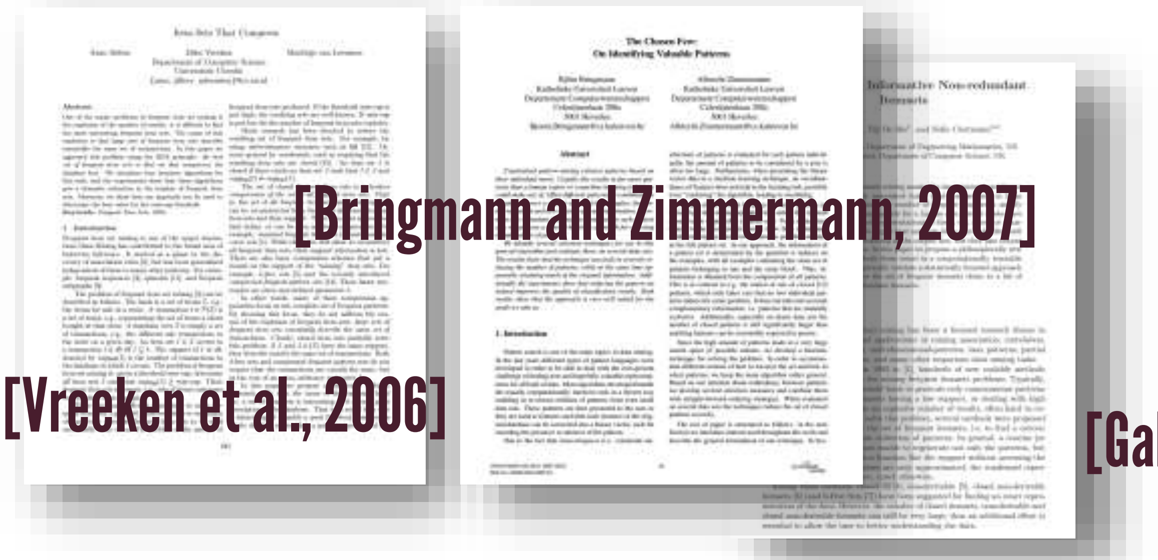

Integrating Classification and Association Rule Mining

*Two-phases: pattern extraction & model construction

Pattern-based models

Krimp
2006

The Chosen Few
MINI
2007



[Vreeken et al., 2006]

[Bringmann and Zimmermann, 2007]

[Gallo et al., 2007]

*Two-phases: pattern extraction & model construction

Pattern-based models

MUSK
2009

Local pattern sampling
2011


MUSK: Uniform Sampling of k Maximal Patterns

[Hasam and Zaki, 2009]


Direct Local Pattern Sampling by Efficient Two-Step Random Procedures

[Boley et al., 2011]

*Sampling

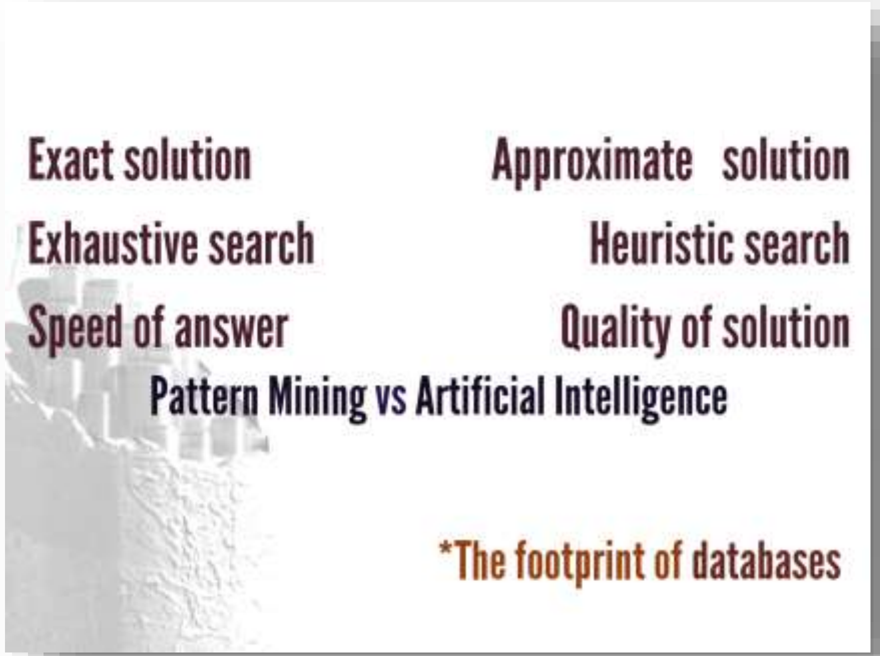
Stop completeness!

Useful Patterns
ACM SIGKDD
Workshop
2010

ECMLPKDD most-influential paper award
2012

“Please, please stop making new algorithms for mining **all** patterns”

Toon Calders



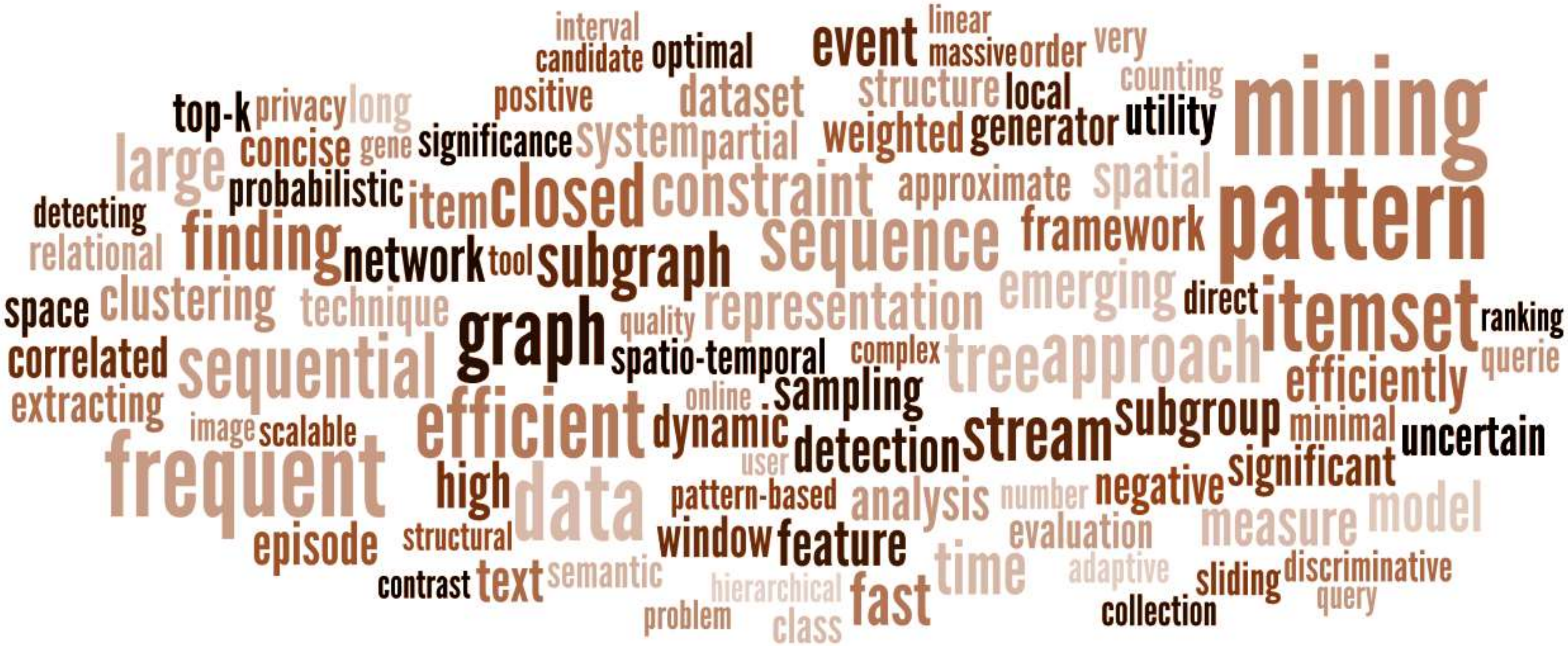
1993

now*

*Aprroximate solution and Heuristic search!

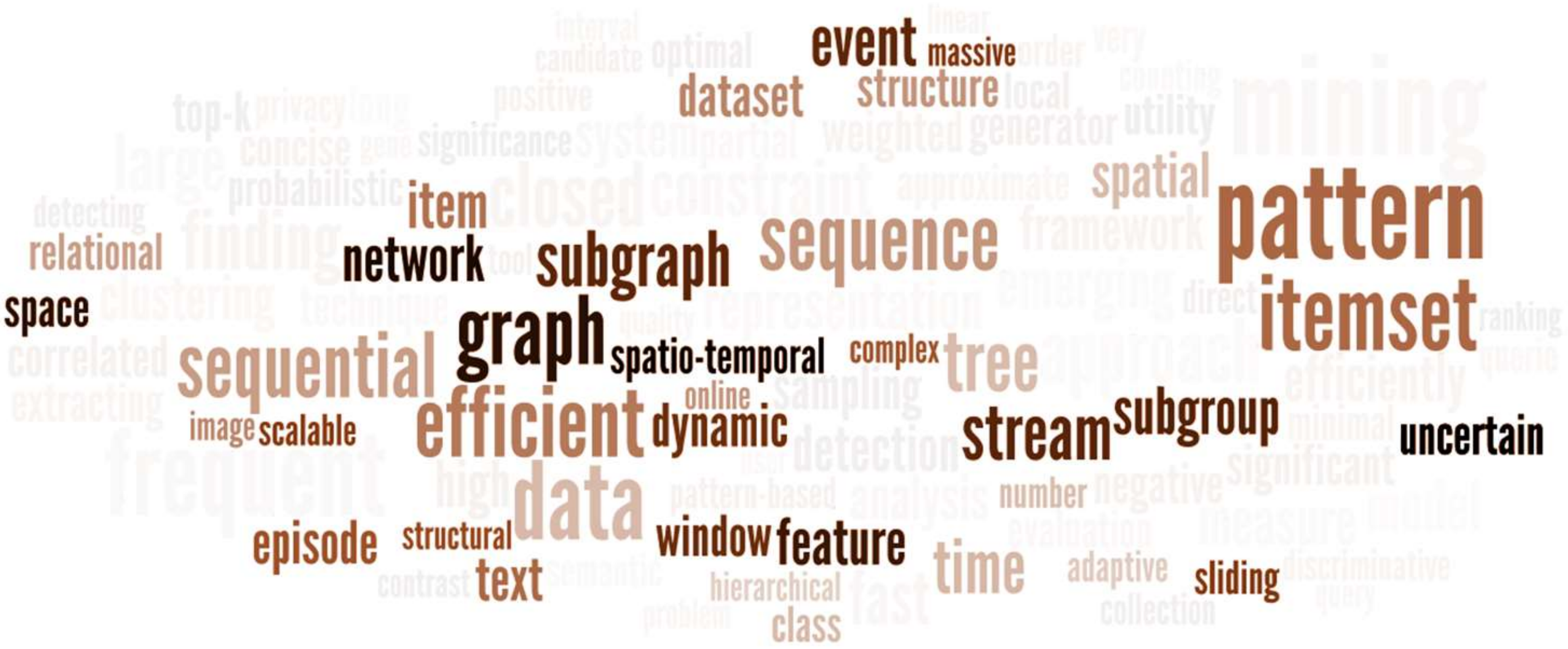
What are the recent trends of Pattern Mining?

New trends



*Recent keywords of Pattern Mining

New trends



*Complex data

New trends

*From speed to quality

Pattern mining as an optimization problem

Top-k frequent patterns
2000

Skypatterns
2011

Optimal patterns
Dominance programming
2015

Mining N-most Interesting Itemsets

[Fu et al., 2000]

Mining Dominant Patterns in the Sky

[Soulet et al., 2011]

Modeling and Mining Optimal Patterns using Dynamic CSP

[Urgate et al., 2015]

*Focusing on the best patterns

	Troubled Romantic	Rich	Dies	Hidding Secret
	A		C	
	A			D
	A	B	C	
		B	C	
	A	B	C	D

Ø

A B C D

AC AB BC BD AD CD

ABC ABD ACD BCD

ABCD

Top-k pattern mining

*Finding the k patterns maximizing an interestingness measure

	Troubled Romantic	Rich	Dies	Hidding Secret
	A		C	
	A			D
	A	B	C	
		B	C	
	A	B	C	D

Ø

A B C D

AC AB BC BD AD CD

ABC ABD ACD BCD

ABCD

Top-k pattern mining

*Finding the 3 most frequent patterns: Ø (5), A (4), C (4)

	Troubled Romantic	Rich	Dies	Hidding Secret
	A		C	
	A			D
	A	B	C	
		B	C	
	A	B	C	D

Ø

A B C D

AC AB BC BD AD CD

ABC ABD ACD BCD

ABCD

Top-k pattern mining

*Finding the 3 most frequent patterns: Ø (5), A (4), C (4)

**Easy due to anti-monotone property of frequency

Troubled Romantic	Rich	Dies	Hidding Secret
A		C	
A			D
A	B	C	
	B	C	
A	B	C	D

Ø

A B C D

AC AB BC BD AD CD

ABC ABD ACD BCD

ABCD

Top-k pattern mining

*Finding the 3 patterns maximizing area: AC (6), BC (6), ABC (6)

**Branch&Bound method

Top-k pattern mining

+	−
Compact	Not fast*
Threshold free	No diversity
Best patterns	

*Exact resolution is costly / sometimes heuristic search (beam seacrh)

Top-k pattern mining

+	−
Compact	Not fast
Threshold free	No diversity*
Best patterns	

*Diversity issue: top-k patterns often very similar

	Troubled Romantic	Rich	Dies	Hidding Secret
	A		C	
	A			D
	A	B	C	
		B	C	
	A	B	C	D

Pattern	Freq.	Area	
Ø	5	0	Top frequent
A	4	4	
C	4	4	
AC	3	6	Top area
BC	3	6	
AD	2	4	
ABC	2	6	
ABCD	1	4	

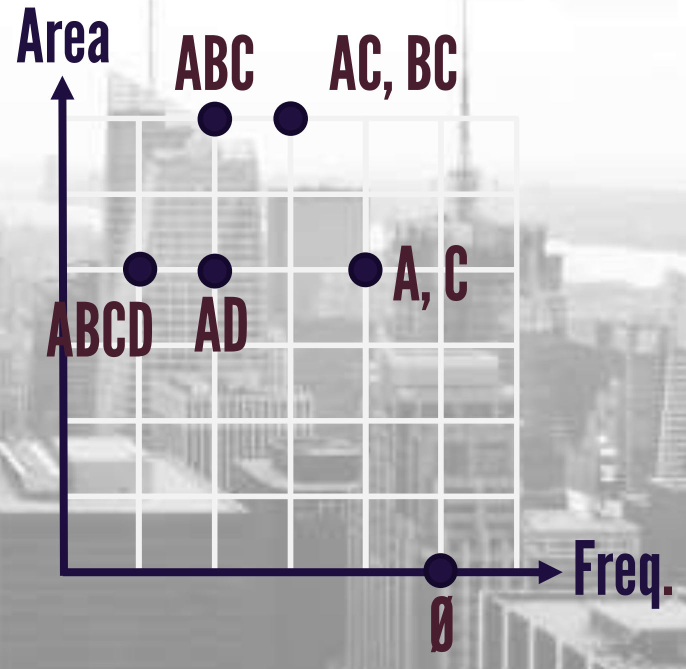
Skyline pattern mining

*How to find a trade-off between several criteria?

	Troubled Romantic	Rich	Dies	Hidding Secret
	A		C	
	A			D
	A	B	C	
		B	C	
	A	B	C	D

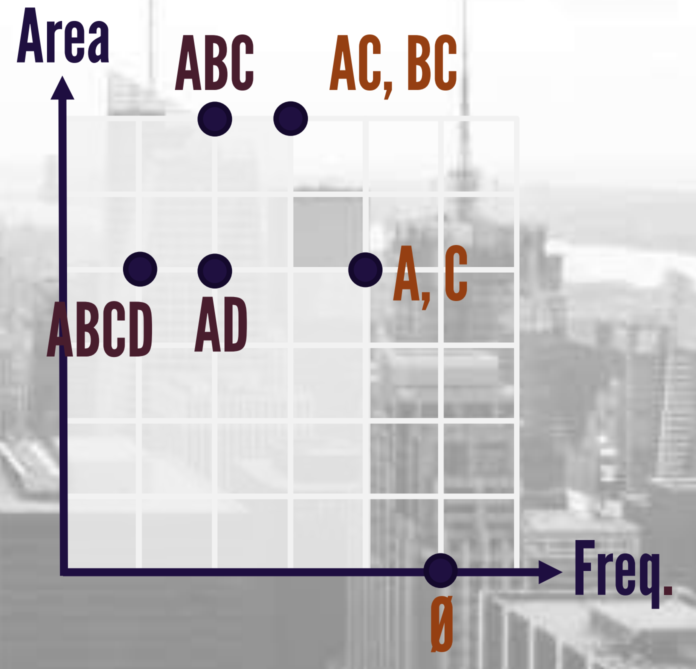
Skyline pattern mining

Pattern	Freq.	Area
Ø	5	0
A	4	4
C	4	4
AC	3	6
BC	3	6
AD	2	4
ABC	2	6
ABCD	1	4



Troubled Romantic	Rich	Dies	Hidding Secret
A		C	
A			D
A	B	C	
	B	C	
A	B	C	D

Pattern	Freq.	Area
Ø	5	0
A	4	4
C	4	4
AC	3	6
BC	3	6
AD	2	4
ABC	2	6
ABCD	1	4



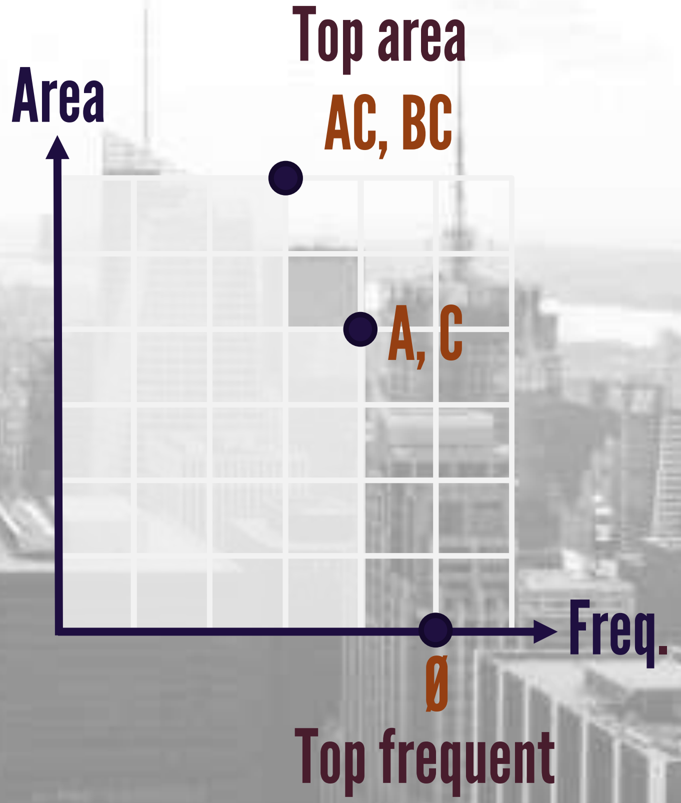
Skyline pattern mining

*Dominated space

	Troubled Romantic	Rich	Dies	Hidding Secret
	A		C	
	A			D
	A	B	C	
		B	C	
	A	B	C	D

Skyline pattern mining

Pattern	Freq.	Area
ø	5	0
A	4	4
C	4	4
AC	3	6
BC	3	6
AD	2	4
ABC	2	6
ABCD	1	4

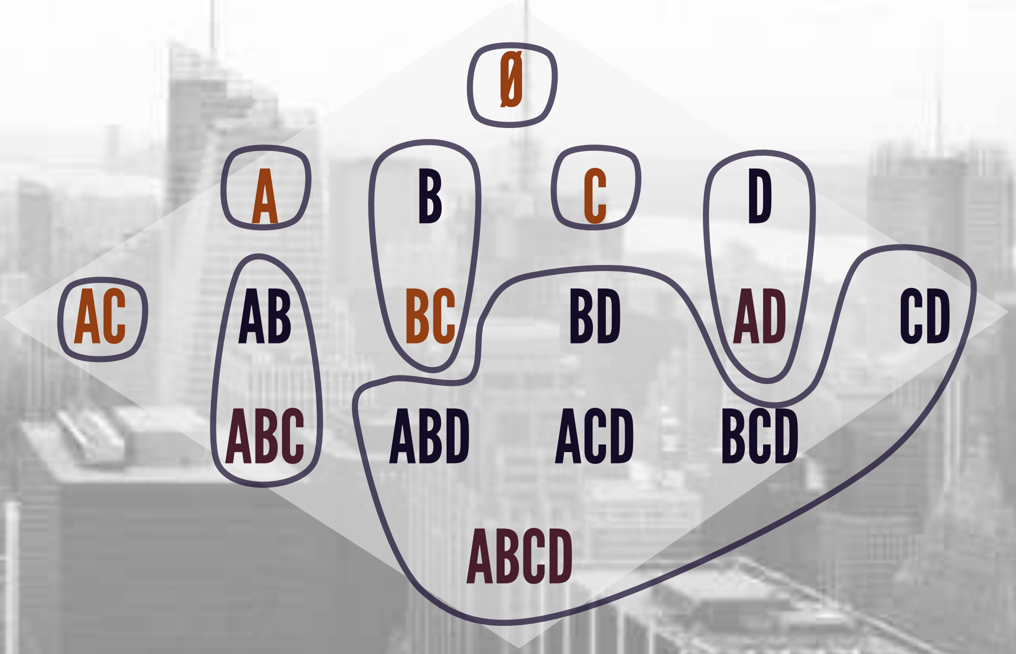


*Skypatterns = non-dominated patterns

	Troubled Romantic	Rich	Dies	Hidding Secret
	A		C	
	A			D
	A	B	C	
		B	C	
	A	B	C	D

Skyline pattern mining

Pattern	Freq.	Area
Ø	5	0
A	4	4
C	4	4
AC	3	6
BC	3	6
AD	2	4
ABC	2	6
ABCD	1	4



*Skypatterns are closed patterns

Dominance programming for optimal patterns

Maximal patterns

Closed patterns

Top-k patterns

Skypatterns

*A pattern is optimal if it is not dominated by another.

Dominance programming for optimal patterns

- Maximal patterns*
- Closed patterns
- Top-k patterns
- Skypatterns

*Dominance relation = inclusion

Dominance programming for optimal patterns

Maximal patterns

Closed patterns*

Top-k patterns

Skypatterns

*Dominance relation = inclusion at same frequency

Maximal patterns

Closed patterns

Dominance programming for optimal patterns

Top-k patterns*

Skypatterns

*Dominance relation = order induced by the interestingness measure

Dominance programming for optimal patterns

Maximal patterns

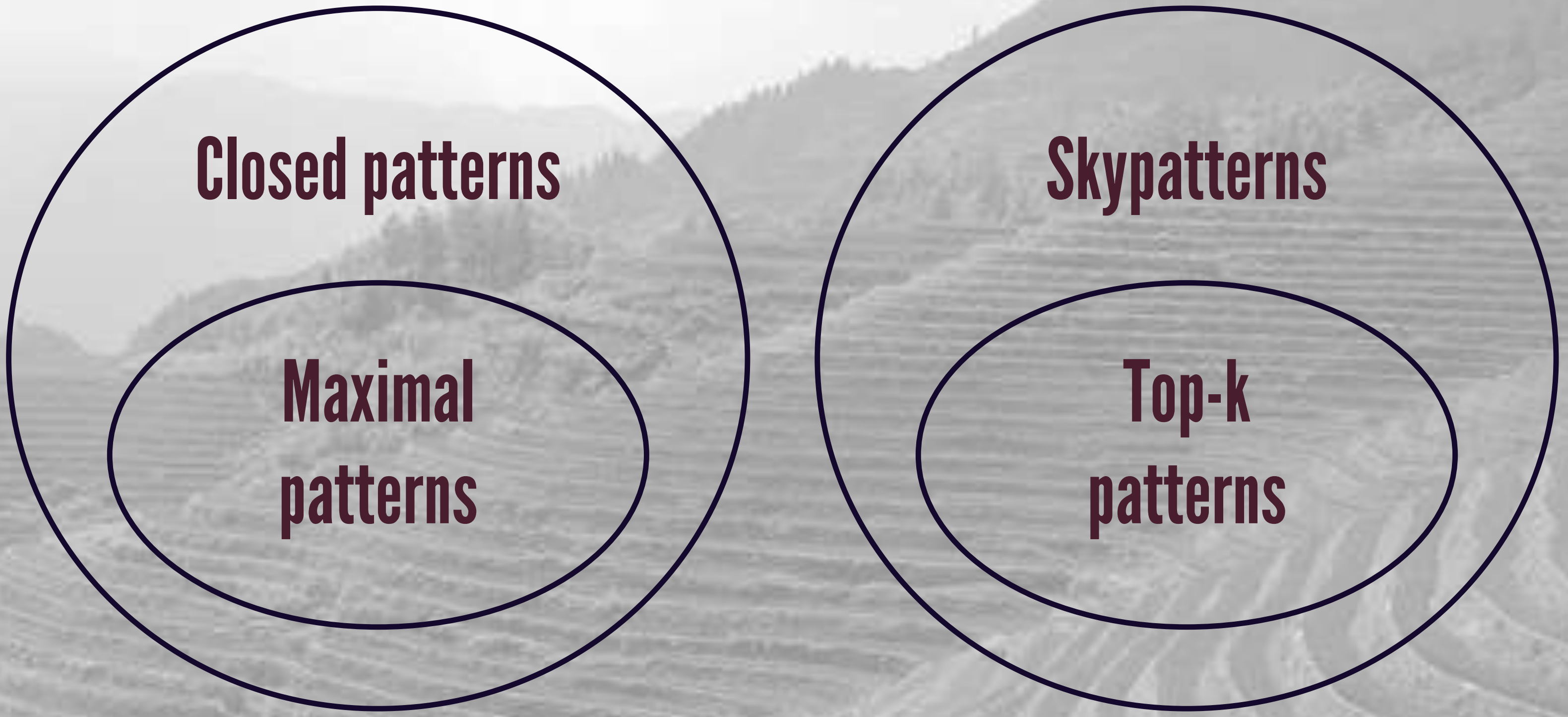
Closed patterns

Top-k patterns

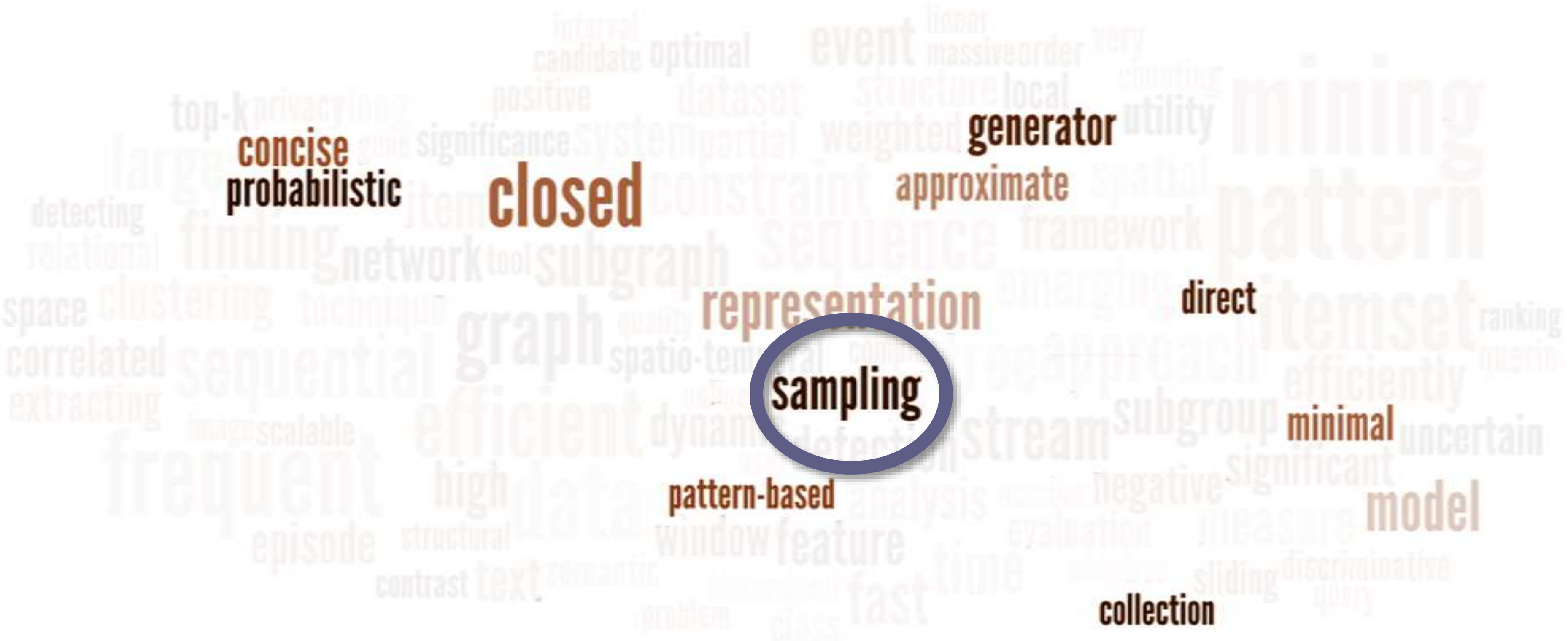
Skypatterns*

*Dominance relation = Pareto domination

Dominance programming for optimal patterns



New trends



*From exhaustive collection to models

Pattern sampling

MUSK
2009

Local pattern sampling
2011


MUSK: Uniform Sampling of k Maximal Patterns


Direct Local Pattern Sampling by Efficient Two-Step Random Procedures

*Sampling

Troubled Romantic	Rich	Dies	Hidding Secret
A		C	
A			D
A	B	C	
	B	C	
A	B	C	D

ø

A B C D

AC AB BC BD AD CD

ABC ABD ACD BCD

ABCD

Pattern A (freq = 4) has twice more chance to be drawn than pattern D (freq = 2)

Pattern sampling

*Picking k patterns randomly with a probability proportional to an interestingness measure

Troubled Romantic	Rich	Dies	Hidding Secret
A		C	
A			D
A	B	C	
	B	C	
A	B	C	D

Pattern sampling

Stochastic methods [Hasan and Zaki, 2009]

Random walk on lattice

Two-step direct method [Boley et al., 2011]

Pick a transaction + pick an itemset of this transaction

*Two main families of methods

	Troubled Romantic	Rich	Dies	Hidding Secret
	A		C	
	A			D
	A	B	C	
		B	C	
	A	B	C	D

Pattern sampling

Stochastic methods [Hasan and Zaki, 2009]

Random walk on lattice

Two-step direct method [Boley et al., 2011]*

Pick a transaction + pick an itemset of this transaction

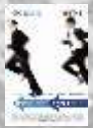
*More uniform

	Troubled Romantic	Rich	Dies	Hidding Secret
	A		C	
	A			D
	A	B	C	
		B	C	
	A	B	C	D

Direct pattern sampling

[Boley et al., 2011]

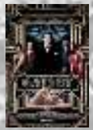
Ø, A, C, AC



Ø, A, D, AD

Ø, A, B, C, AB, AC, BC, ABC

Ø, B, C, BC



Ø, A, B, C, D, AB, AC, AD, BC, BD, CD, ABC, ABD, ACD, BCD, ABCD

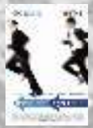
*Consider all itemsets contain in each transaction

	Troubled Romantic	Rich	Dies	Hidding Secret
	A		C	
	A			D
	A	B	C	
		B	C	
	A	B	C	D

Direct pattern sampling

[Boley et al., 2011]

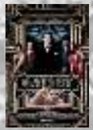
Ø, A, C, AC



Ø, A, D, AD

Ø, A, B, C, AB, AC, BC, ABC

Ø, B, C, BC

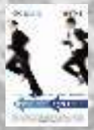


Ø, A, B, C, D, AB, AC, AD, BC, BD, CD, ABC, ABD, ACD, BCD, ABCD

Pattern A (freq = 4) appears twice more than pattern D (freq = 2)

*Consider all itemsets contain in each transaction

	Troubled Romantic	Rich	Dies	Hidding Secret
	A		C	
	A			D
	A	B	C	
		B	C	
	A	B	C	D

	4	ø, A, C, AC
	4	ø, A, D, AD
	8	ø, A, B, C, AB, AC, BC, ABC
	4	ø, B, C, BC
	16	ø, A, B, C, D, AB, AC, AD, BC, BD, CD, ABC, ABD, ACD, BCD, ABCD

Direct pattern sampling

[Boley et al., 2011]

Pattern A (freq = 4) has twice more chance to be drawn than pattern D (freq = 2)

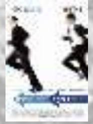
*Count the number of itemsets

	Troubled Romantic	Rich	Dies	Hidding Secret
	A		C	
	A			D
	A	B	C	
		B	C	
	A	B	C	D

Direct pattern sampling

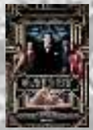
[Boley et al., 2011]


1/9 ∅, A, C, AC


1/9 ∅, A, D, AD


2/9 ∅, A, B, C, AB, AC, BC, ABC


1/9 ∅, B, C, BC


4/9 ∅, A, B, C, D, AB, AC, AD, BC, BD, CD, ABC, ABD, ACD, BCD, ABCD

Pattern A (freq = 4) has twice more chance to be drawn than pattern D (freq = 2)

*Normalize

	Troubled Romantic	Rich	Dies	Hidding Secret
	A		C	
	A			D
	A	B	C	
		B	C	
	A	B	C	D

1/9 Ø, A, C, AC

1/9 Ø, A, D, AD

2/9 Ø, A, B, C, AB, AC, BC, ABC

1/9 Ø, B, C, BC

4/9 Ø, A, B, C, D, AB, AC, AD, BC, BD, CD, ABC, ABD, ACD, BCD, ABCD

Direct pattern sampling

[Boley et al., 2011]

Pattern A (freq = 4) has twice more chance to be drawn than pattern D (freq = 2)

*Pick a transaction proportionally to the distribution

**Pick uniformly an itemset within this transaction

Pattern sampling

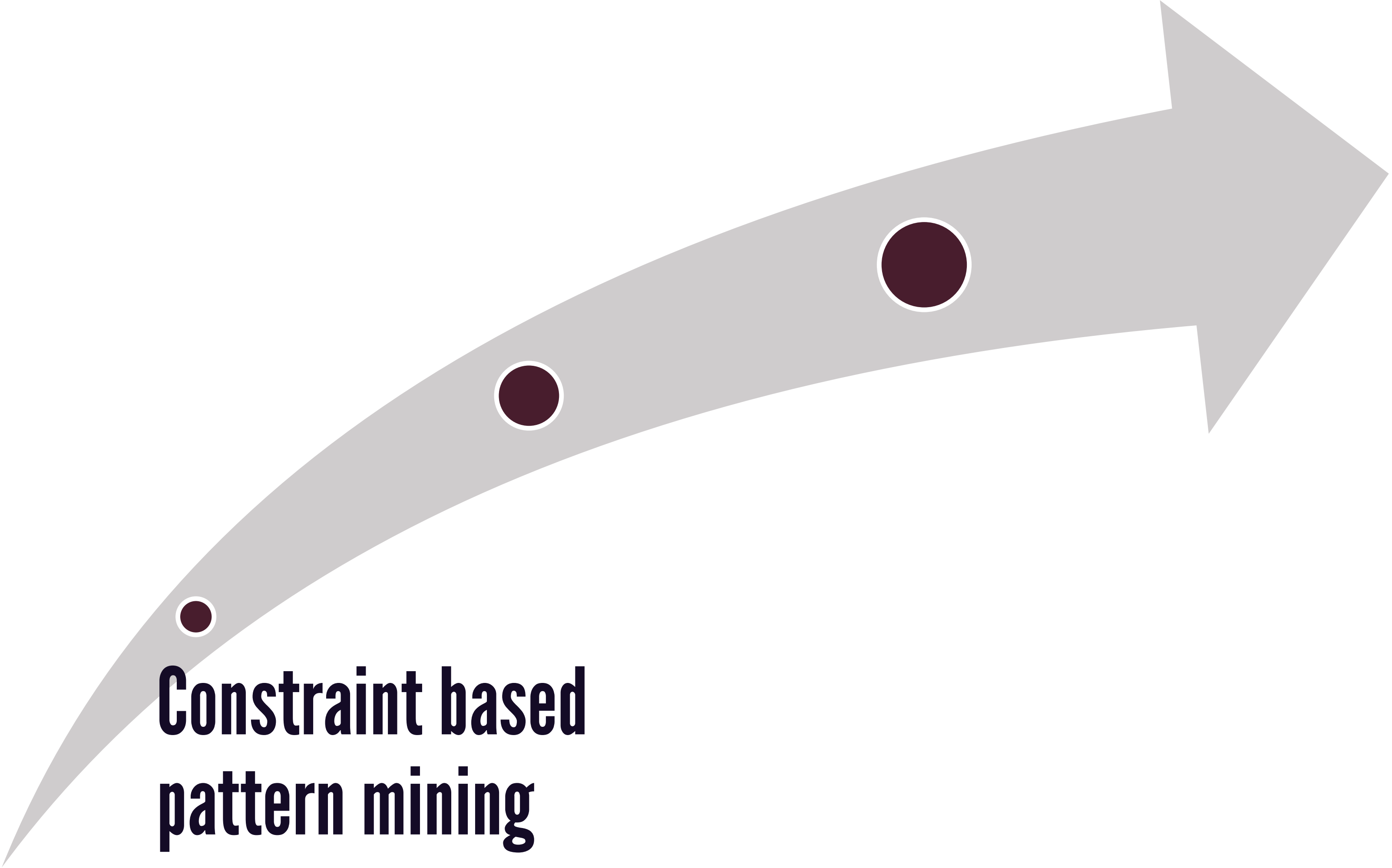
Compact

Threshold free

Diversity

Very fast

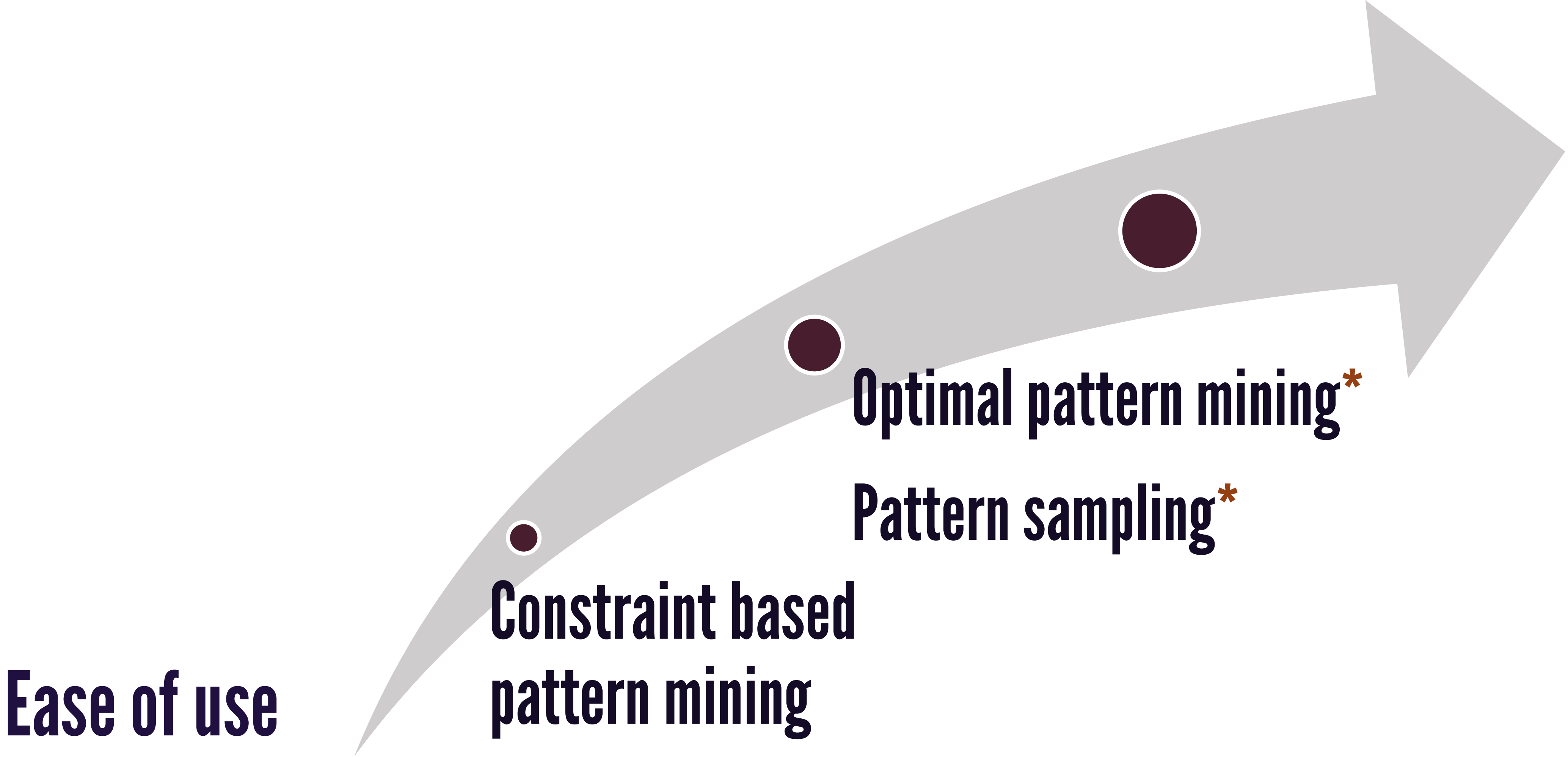
Patterns far from optimality



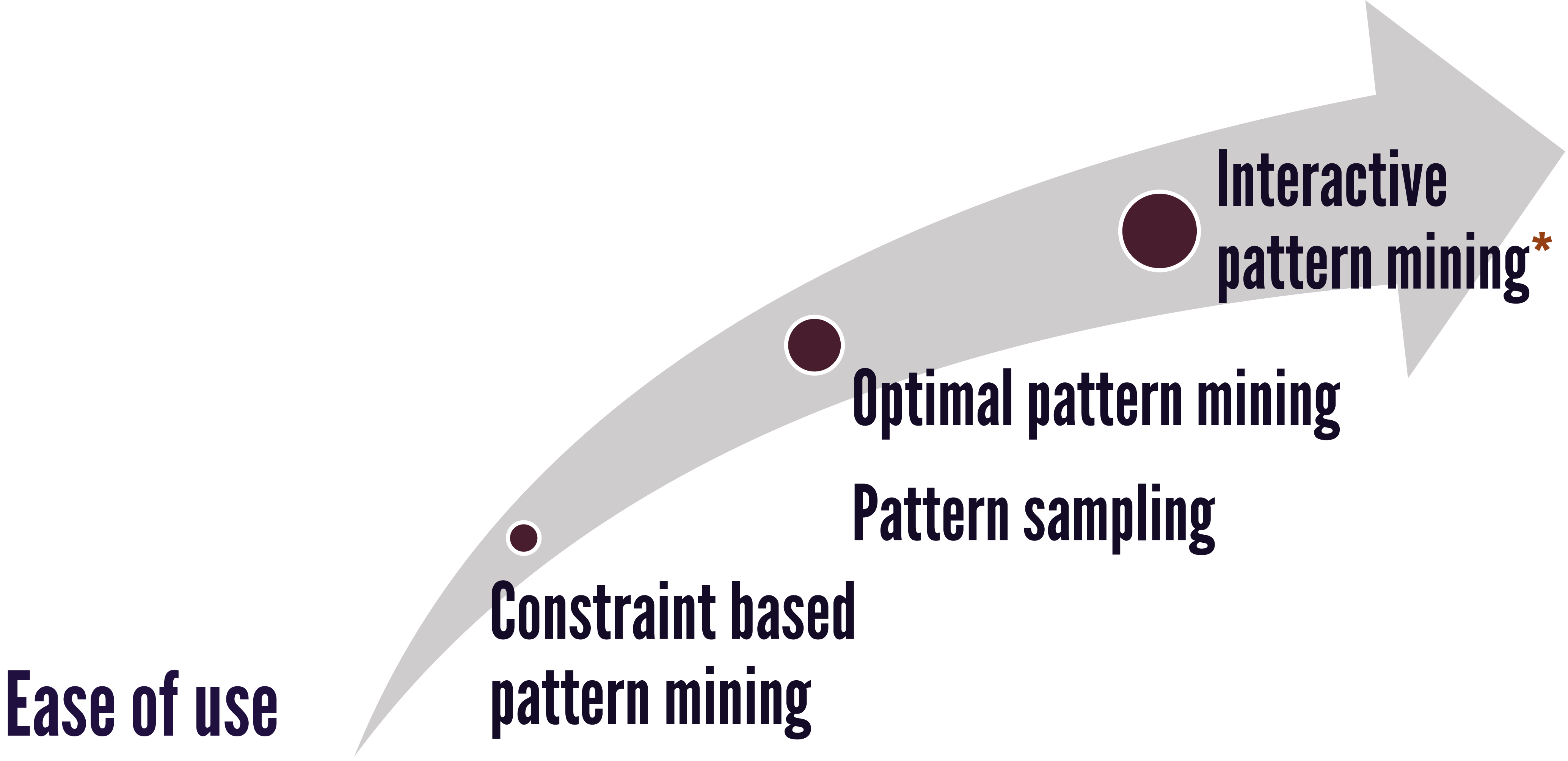
Ease of use

Constraint based pattern mining

*No algorithm specification



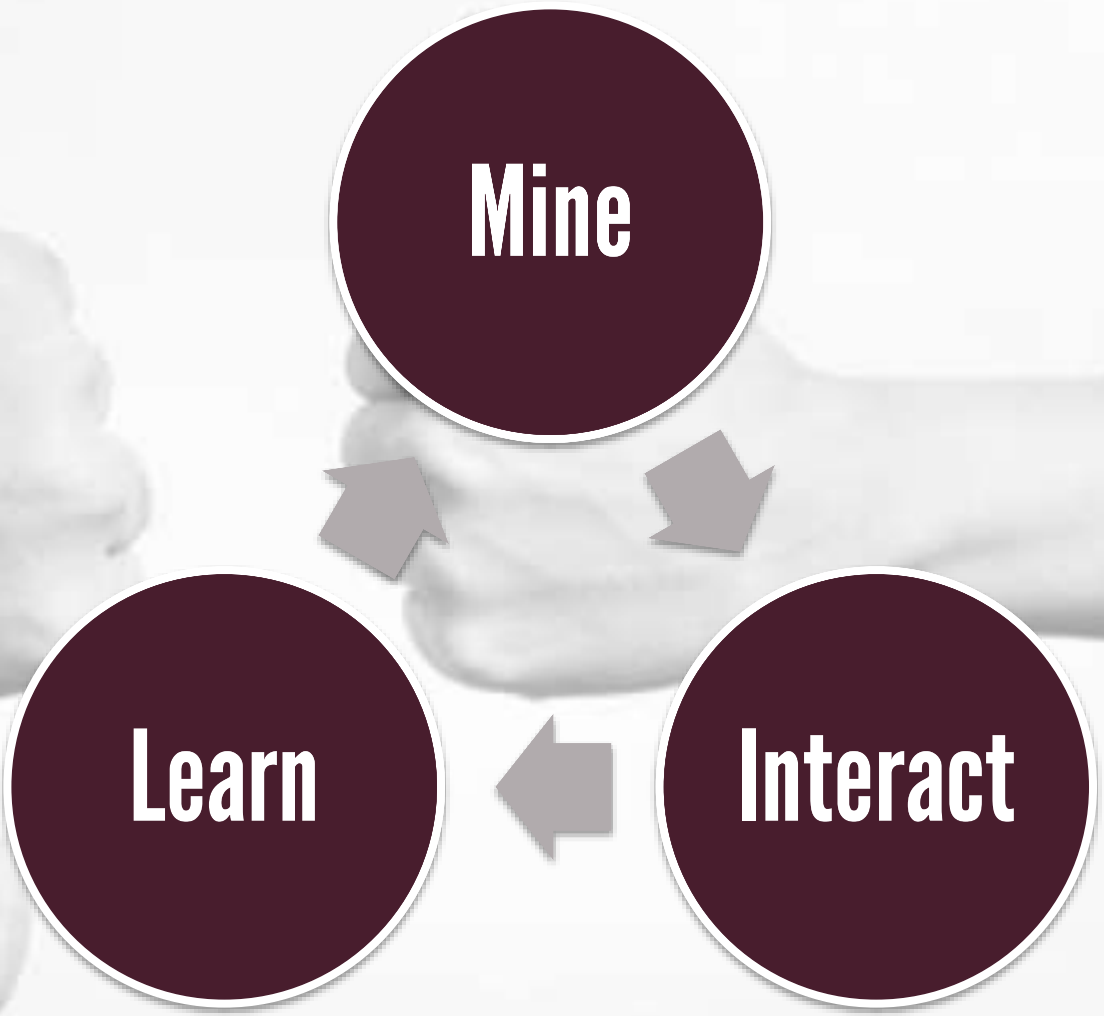
*No user-specified threshold



*No user-specified measure

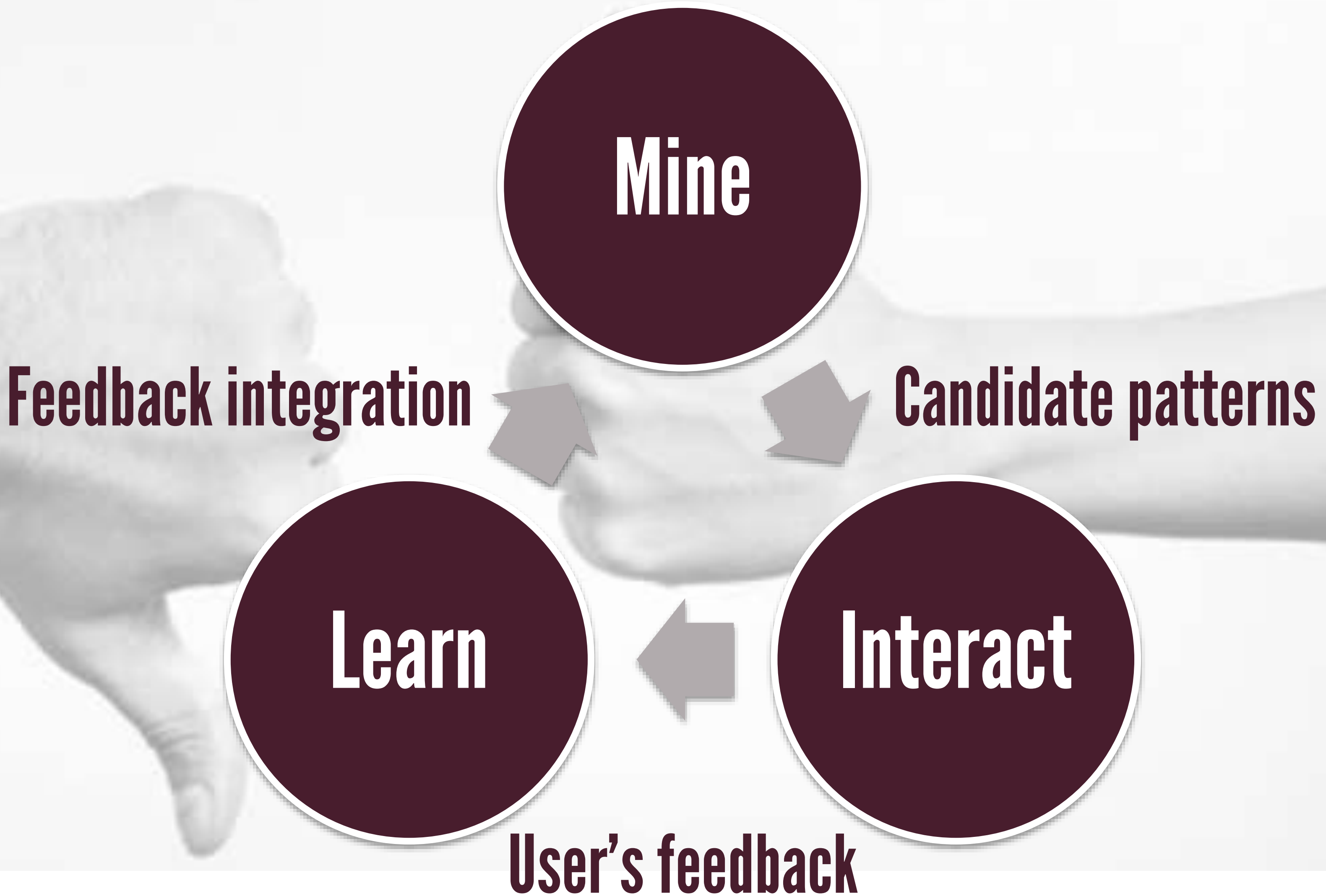
Interactive data exploration using pattern mining

[van Leeuwen 2014]



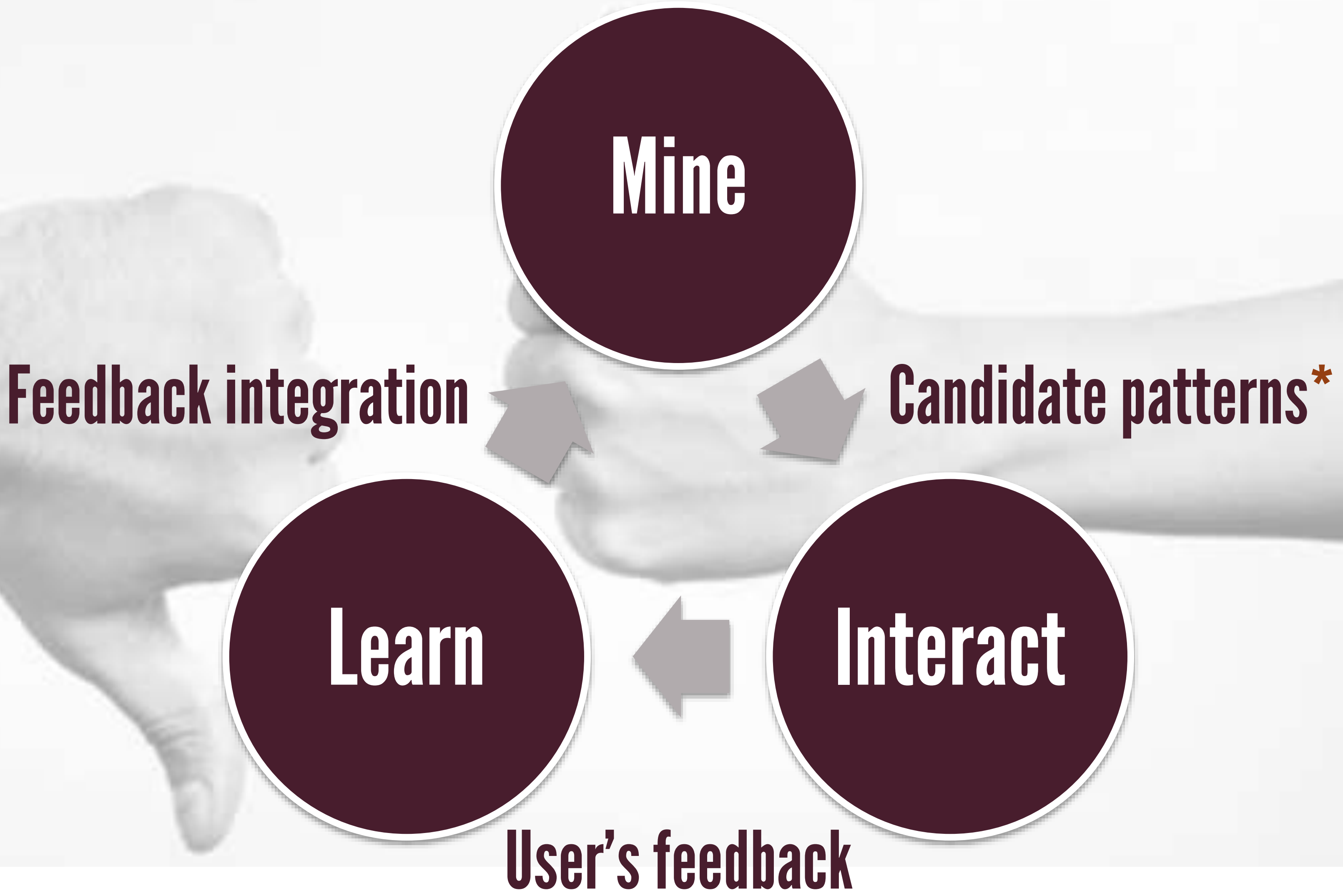
Interactive data exploration using pattern mining

[van Leeuwen 2014]



Interactive data exploration using pattern mining

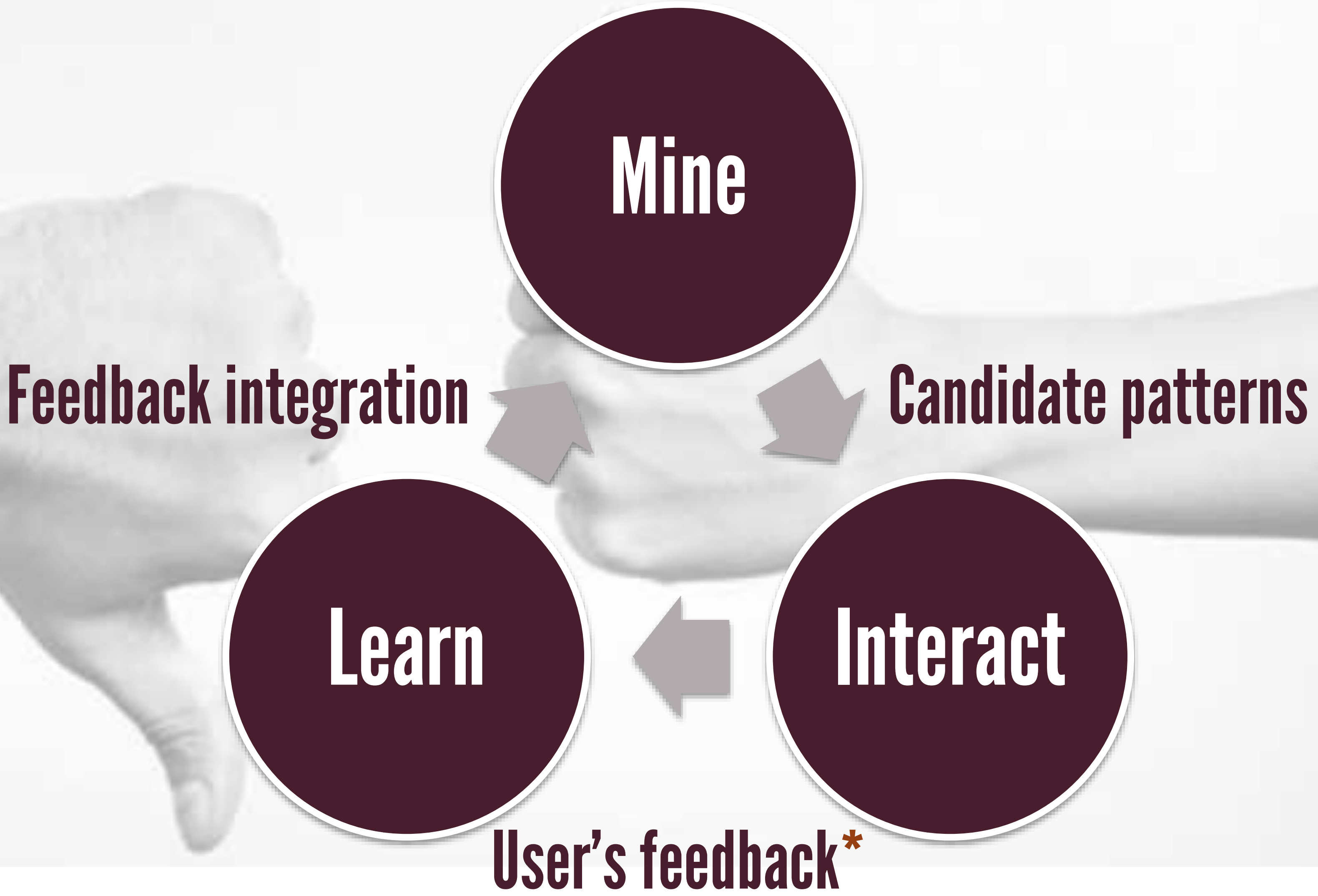
[van Leeuwen 2014]



*Active learning vs useful pattern mining

Interactive data exploration using pattern mining

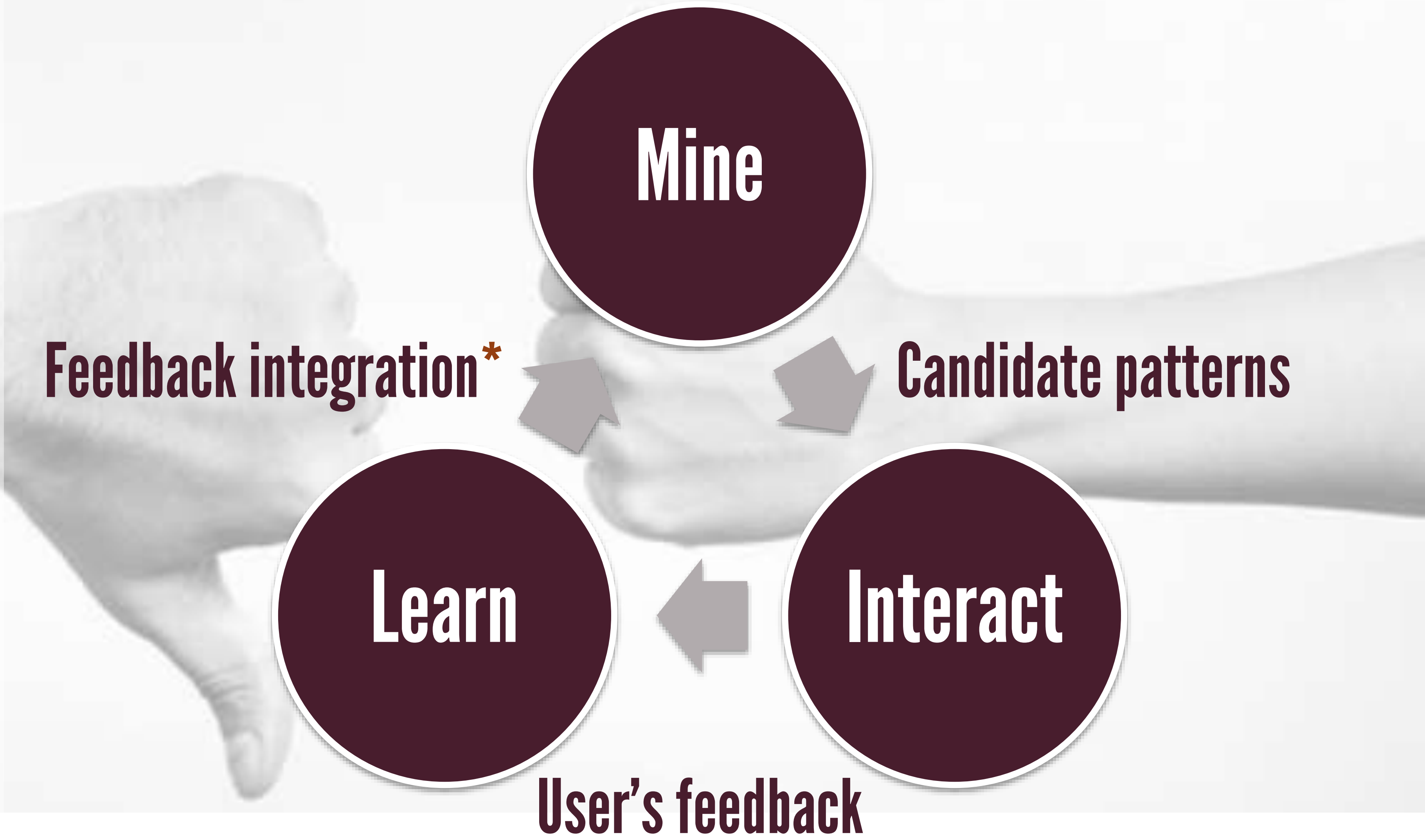
[van Leeuwen 2014]



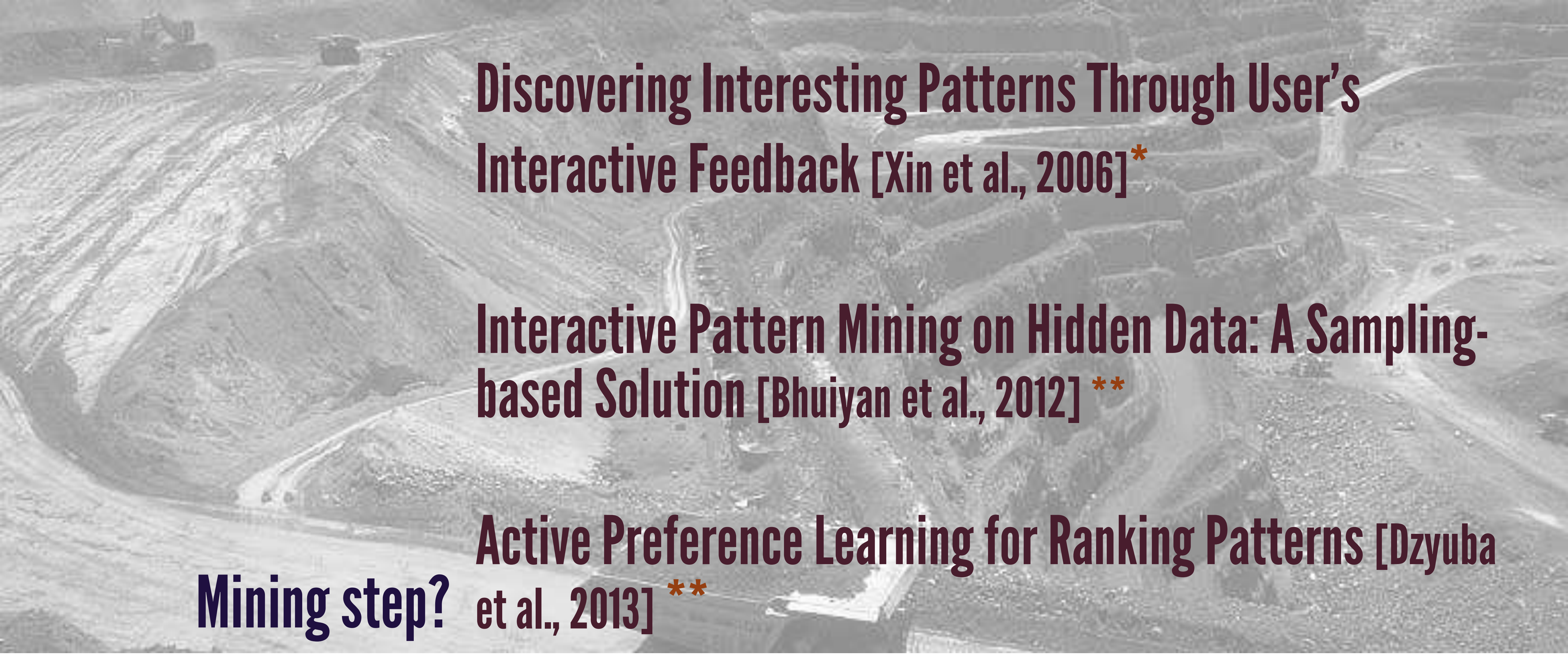
*Explicit feedback vs implicit feedback

Interactive data exploration using pattern mining

[van Leeuwen 2014]



*How to upate the target of the mining method?



Discovering Interesting Patterns Through User's Interactive Feedback [Xin et al., 2006]*

Interactive Pattern Mining on Hidden Data: A Sampling-based Solution [Bhuiyan et al., 2012] **

Active Preference Learning for Ranking Patterns [Dzyuba et al., 2013] **

Mining step?

*Offline mining of all frequent patterns

**Online mining by integrating preferences

Discovering Interesting Patterns Through User's Interactive Feedback [Xin et al., 2006]

Interactive Pattern Mining on Hidden Data: A Sampling-based Solution [Bhuiyan et al., 2012] *

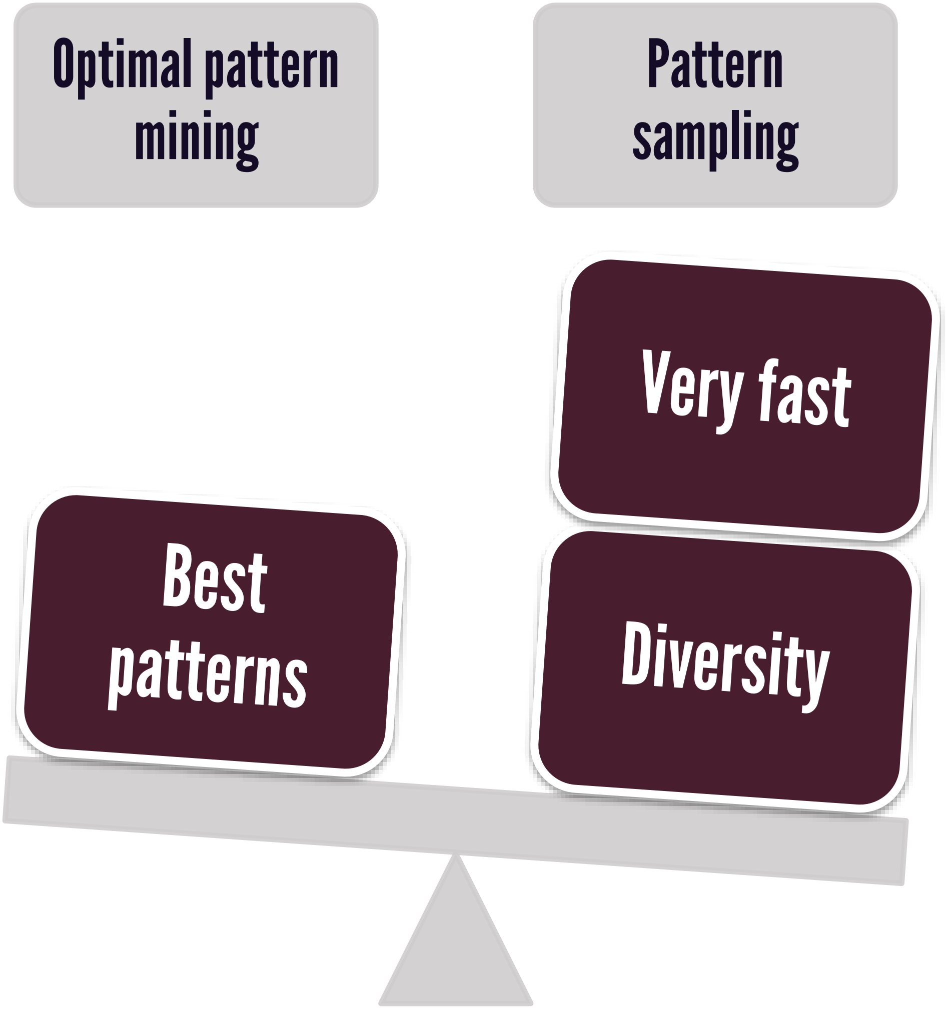
Active Preference Learning for Ranking Patterns [Dzyuba et al., 2013] **

Mining step?

*Pattern sampling

**Optimal pattern mining via beam search


Mining step?


Optimal pattern mining
Pattern sampling
Very fast
Best patterns
Diversity

Discovering Interesting Patterns Through User's Interactive Feedback [Xin et al., 2006]*

Interactive Pattern Mining on Hidden Data: A Sampling-based Solution [Bhuiyan et al., 2012]**

Active Preference Learning for Ranking Patterns [Dzyuba et al., 2013]*

Learning step?

*Ranking over all patterns = learning to rank problem

**Weight on items

Conclusion

Frequent pattern mining
1990s

Constraint-based pattern mining
2000s

Optimal pattern mining
Early 2010s

Declarative pattern mining
Early 2010s

Interactive pattern mining
Now

Retrieval era

Exploratory analysis era

Performance issue*

The more, the better

Data-driven

Quality issue

The less, the better

User-driven

*Faster

Conclusion

Frequent pattern mining
1990s

Constraint-based pattern mining
2000s

Optimal pattern mining
Early 2010s

Declarative pattern mining
Early 2010s

Interactive pattern mining
Now

Retrieval era

Exploratory analysis era

Performance issue

The more, the better

Data-driven

Quality issue*

The less, the better*

User-driven

*Faster, better

Conclusion

Frequent pattern mining
1990s

Constraint-based pattern mining
2000s

Optimal pattern mining
Early 2010s

Declarative pattern mining
Early 2010s

Interactive pattern mining
Now

Retrieval era

Exploratory analysis era

Performance issue

The more, the better

Data-driven

Quality issue

The less, the better*

User-driven*

*Faster, better, easier

Acknowledgements

Bruno Crémilleux, François Rioult

Arnaud Giacometti, Dominique Li, Patrick Marcel

Marc Plantevit

Chedy Raïssi

Thank you!