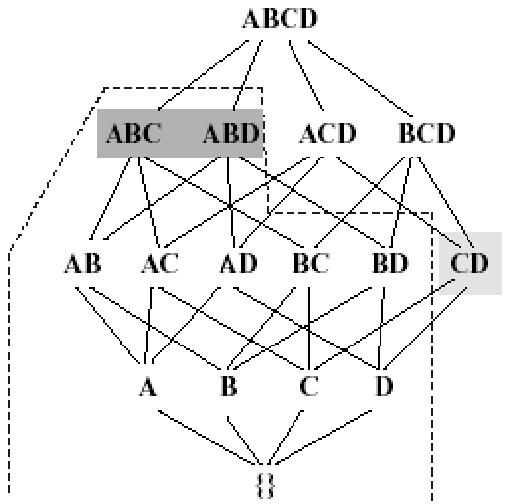
The Levelwise Search Algorithm



Tamás Horváth

University of Bonn &
Fraunhofer IAIS, Sankt Augustin, Germany
tamas.horvath@iais.fraunhofer.de

universitätbonn
RHEINISCHE FRIEDRICH-WILHELMS-UNIVERSITÄT

Fraunhofer
IAIS

Local Pattern Mining as Theory Extraction

Given

- a *database* D,
- a *pattern language* $\mathcal{L}$,
  - e.g., expressing properties or defining subgroups of the data
  - elements of $\mathcal{L}$ are referred to as *sentences* or *patterns*
- an *interestingness predicate* $q_D : \mathcal{L} \to \{\mathrm{true}, \mathrm{false}\}$
  - e.g., evaluating whether a sentence $\varphi \in \mathcal{L}$ is "interesting" w.r.t. D

compute $Th(\mathcal{L}, D, q_D) = \{\varphi \in \mathcal{L} : q_D(\varphi) = \mathrm{true}\}$

- i.e., set of "interesting" sentences
- $Th(\mathcal{L}, D, q_D)$: theory of D with respect to $\mathcal{L}$ and q_D

Theory Extraction Problem – Additional Restrictions

in many **practical cases**, we have further properties of the theory extraction problem that can algorithmically be utilized

1. there is a (natural) partial order $\preccurlyeq$ on $\mathcal{L}$
   - $\varphi, \theta \in \mathcal{L}$; $\varphi \preccurlyeq \theta$ referred to as φ is more general than θ (or θ is more specific than φ)
2. q_D is anti-monotone with respect to $\preccurlyeq$
   - i.e., $\varphi \preccurlyeq \theta$ and $q_D(\theta) = \mathrm{true}$ implies $q_D(\varphi) = \mathrm{true}$

Examples

some data mining problems which are instances of the theory extraction problem:

- association rule mining
- frequent set mining
- frequent episode mining
- subgroup discovery
- frequent connected subgraph mining
- track mining
- ...

Example I: Association Rule Mining

- discovery of interesting relations between binary attributes, called *items*, in large databases

example of an association rule extracted from supermarket sales**:**

“Customers who buy cereals and sugar also ***tend*** *to buy milk.”*

- only rules with support and confidence above some minimal thresholds are extracted
  - support: proportion of customers who bought the three items among **all** customers
  - confidence: proportion of customers who bought milk among the customers who bought cereals and sugar

Example I: Association Rule Mining

special case of the theory extraction problem:

- *database* D: binary matrix,
  - *rows*: transactions
  - *columns*: items
- *pattern language* $\mathcal{L}$:
  rules of the form $X \rightarrow Y$, where X and Y are disjoint sets of items
- *interestingness predicate* $q_D : \mathcal{L} \rightarrow \{\text{true}, \text{false}\}$:

$$q_D(X \rightarrow Y) = \text{true} \iff$$
$$\text{support}(X \rightarrow Y) \geq minsup \text{ and } \text{confidence}(X \rightarrow Y) \geq minconf$$

Example II: Frequent Itemset Mining

- discovery of sets of items (columns) that are subsets of at least *t* transactions (rows) in a binary matrix

Example:

TID	*Items*
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Milk, Bread, Diaper, Beer
5	Milk, Bread, Diaper, Coke

- $\{Milk, Bread, Diaper\}$ is frequent for the frequency thershold $t = 2$

Example II: Frequent Itemset Mining

special case of the theory extraction problem:

- *database* D: binary matrix,
  - *rows*: transactions
  - *columns*: set I of items
- *pattern language* $\mathcal{L}$: 2^I
- *interestingness predicate* $q_D : \mathcal{L} \to \{\mathrm{true}, \mathrm{false}\}$:

$$q_D(X) = \mathrm{true} \iff \text{support}(X) \geq minsup$$

Example III: Frequent Episodes

episode mining: list collection of events that occur within a time interval of a given size in a given partial order

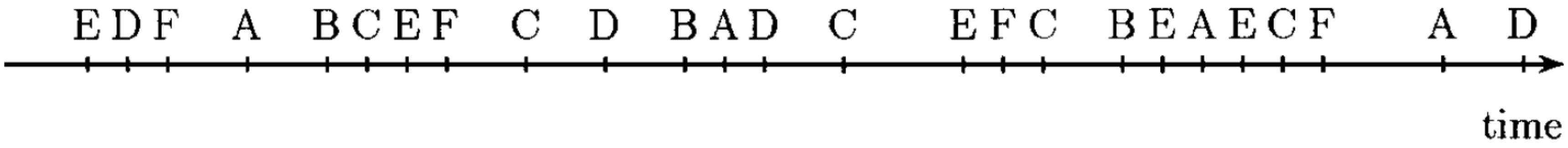
episode: $\varphi = (V, \leq, g)$ with

- V: set of nodes,
- $\leq$: partial order on V,
- $g : V \to E$: associates each node with an event type in E

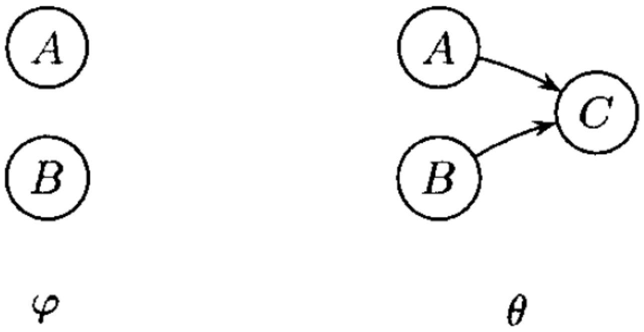
problem definition: *Given* an event sequence S over E, a window width win, and a frequency threshold $t > 0$, *list* all episodes φ that are “contained” in at least t windows of size win

- “contained”: the total order on the events in the window is consistent with the partial order of φ

Example III: Frequent Episodes



(a)



(b)

(a) An event sequence; (b) two episodes.

Example III: Frequent Episodes

frequent episode mining: special case of the theory extraction problem

- *exercise*

Example IV: Frequent Connected Subgraph Mining

frequent connected subgraph mining problem:

Given a set D of labeled graphs and an integer $t > 0$, *list* the set of t-frequent connected subgraphs w.r.t. D

- $t > 0$ integer: *frequency threshold*
- t*-frequent subgraph:* subgraph isomorphic to at least t graphs in D

instance of theory extraction problem:

- D : set of labeled graphs
- $\mathcal{L}$: set of all labeled *connected* graphs
- *interestingness predicate* q_D: for a pattern $H \in \mathcal{L}$, $q_D(H)$ is true iff H is subgraph isomorphic to at least t graphs in D
  - q_D is *anti-monotone*: any connected subgraph of a t-frequent connected graph is also t-frequent

Some Remarks on the Theory Extraction Problem

remarks on the theory extraction problem restricted to anti-monotone interestingness predicate

- enumeration problem
- size of the problem is defined by the *size of the input* database D
- size of the output can be exponentially large in the size of the input
  - e.g., for $D = \{I\}$ with $I = \{1, \ldots, n\}$, $\mathcal{L} = 2^I$, and $q_D : \varphi \mapsto \mathrm{true}$ for every $\varphi \in \mathcal{L}$, we have the theory $Th(\mathcal{L}, D, q_D) = \mathcal{L}$

  $\Rightarrow$ **hopeless** to compute $Th(\mathcal{L}, D, q_D)$ in time polynomial in the input parameter

  $\Rightarrow$ the size of the output is also taken into account in the analyses of the time and space complexity

The Levelwise Algorithm

- [Agrawal, Mannila, Srikant, Toivonen, & Verkamo, 1996]
- developed for the theory extraction problem restricted to anti-monotone interestingness predicates w.r.t. the partial order on the pattern language
- starting from the most generel sentences, generate and evaluate more and more special sentences
  - breadth-first search
  - do not evaluate those sentences that cannot be interesting given the set of interesting sentences computed earlier

The Levelwise Algorithm

Input : database D, a language $\mathcal{L}$ associated with a specialization relation $\preceq$, and a selection predicate q_D

Output: $Th(\mathcal{L}, D, q_D)$

$\mathcal{C}_1 := \{\varphi \in \mathcal{L} :$ there is no $\varphi' \in \mathcal{L}$ such that $\varphi' \prec \varphi\}$

$i := 1$

while $\mathcal{C}_i \neq \emptyset$ **do**

  $\mathcal{F}_i := \{\varphi \in \mathcal{C}_i : q_D(\varphi)\}$

  $\mathcal{C}_{i+1} := \{\varphi \in \mathcal{L} :$ for all $\varphi' \prec \varphi$ we have $\varphi' \in \bigcup_{j \leq i} \mathcal{F}_j\} \setminus \bigcup_{j \leq i} \mathcal{C}_j$

  $i := i + 1$

endwhile

print $\bigcup_{j<i} \mathcal{F}_j$

Proposition: The levelwise algorithm computes $Th(\mathcal{L}, D, q_d)$ correctly.

Proof: *exercise*

Complexity of Finding All Interesting Sentences

- we consider the restricted theory extraction problem
  - i.e., partially ordered pattern language and anti-monotone interestingness predicate
- real-world applications:
  - main effort in generating the theory is in the evaluation of the interestingness predicate q_D against the database

⇨ we want to analyse the complexity of generating all interesting sentences in terms of the number of evaluations of the interestingness predicate

  - we show that it depends not only on the cardinality of the theory (i.e., set of interesting sentences), but also on the cardinality of the **border** of theory

Borders of Theories

- $(\mathcal{L}, \preccurlyeq)$: poset
- $S \subseteq \mathcal{L}$ is closed downwards under $\preccurlyeq$ if

$$\forall \varphi' (\exists \varphi \in S \text{ such that } \varphi' \preccurlyeq \varphi \implies \varphi' \in S)$$

- let $S \subseteq \mathcal{L}$ be closed downwards under $\preccurlyeq$
  - the border $Bd(S)$ of S is defined by

$$Bd(S) = \{\varphi \in \mathcal{L} : \forall \gamma (\gamma \prec \varphi \implies \gamma \in S) \land \forall \theta (\varphi \prec \theta \implies \theta \notin S)\}$$

  i.e., all generalizations of φ are in S and none of the specializations of φ is in S

- if S is not closed then $Bd(S) = Bd(S')$, where S' is the downward closure of S

Positive and Negative Borders

- the positive border $Bd^+(S)$ of S is defined by

$$Bd^+(S) = \{\varphi \in S : \forall\theta(\varphi \prec \theta \implies \theta \notin S)\}$$

  i.e., those sentences $\varphi \in Bd(S)$ that are in S

- the negative border $Bd^-(S)$ of S is defined by

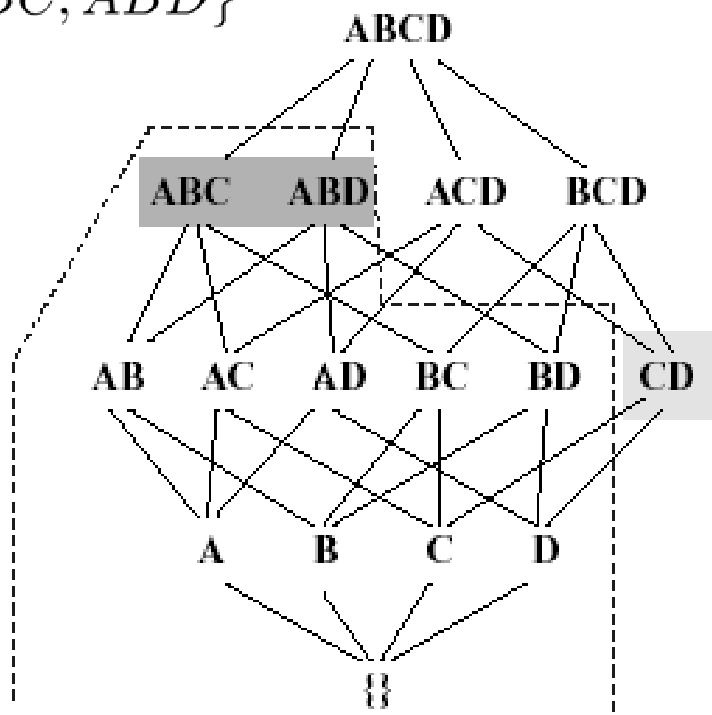
$$Bd^-(S) = \{\varphi \in \mathcal{L} \setminus S : \forall\gamma(\gamma \prec \varphi \implies \gamma \in S)\}$$

  i.e., those sentences $\varphi \in Bd(S)$ that are not in S

$\Rightarrow\ Bd(S) = Bd^+(S) \cup Bd^-(S)$

Example

- poset $(\mathcal{L}, \preccurlyeq)$ is $(2^{\{A,B,C,D\}}, \subseteq)$
- $S = \{ABC, ABD\}$ // $\{X_1, \ldots, X_k\}$ is also denoted by $X_1 \ldots X_k$
- $S' = \{\emptyset, A, B, C, D, AB, AC, AD, BC, BD, ABC, ABD\}$
- $Bd(S) = Bd(S') = \{ABC, ABD, CD\}$ with
  - $Bd^+(S) = Bd^+(S') = \{ABC, ABD\}$
  - $Bd^-(S) = Bd^-(S') = \{CD\}$



Complexity of Finding All Interesting Sentences

Prop.: The levelwise algorithm uses

$$|Th(\mathcal{L}, D, q_D) \cup Bd^-(Th(\mathcal{L}, D, q_D))|$$

evaluations of the interestingness predicate q_D.

Proof: *exercise*

Complexity of Finding All Interesting Sentences

Theorem: *Given* $(\mathcal{L}, \preccurlyeq)$, D, q_D, and $S \subseteq \mathcal{L}$, *deciding* whether $S = Th(\mathcal{L}, D, q_D)$

(i) requires in the worst case at least $|Bd(S)|$ evaluations of q_D and

(ii) can be done using exactly this number of evaluations of q_D.

Corollary: Any algorithm that computes $Th(\mathcal{L}, D, q_d)$ and has access to D only by means of evaluations of q_D requires *at least*

$$|Bd(Th(\mathcal{L}, D, q_D))|$$

invocations of the interestingness predicate q_D.

Proof of the Theorem

Proof of (i): suppose there exists an algorithm $\mathcal{A}$ and a problem instance $(\mathcal{L}, D, q_D)$ such that $\mathcal{A}$ computes $Th(\mathcal{L}, D, q_D)$ with less than $|Bd(S)|$ evaluations of the interestingness predicate q_D

$\Rightarrow$ $\exists \varphi \in Bd(S)$ that has not been evaluated by $\mathcal{A}$

$\Rightarrow$ consider the problem instance $(\mathcal{L}, D, q'_D)$ with

$$q'_D(\theta) := \begin{cases} \neg q_D(\theta) & \text{if } \theta = \varphi \\ q_D(\theta) & \text{o/w} \end{cases}$$

for every $\theta \in \mathcal{L}$

$\Rightarrow$ $\mathcal{A}$ computes the same theory for the two instances

$\Rightarrow$ $\mathcal{A}$ is not correct, as $Th(\mathcal{L}, D, q_D) \neq Th(\mathcal{L}, D, q'_D)$

Proof of the Theorem

Proof of (ii): it follows directly from

$$S = Th(\mathcal{L}, D, q_D) \iff$$
$$q_D(\varphi) = \text{true} \text{ for all } \varphi \in Bd^+(S) \text{ and } q_D(\theta) = \text{false} \text{ for all } \theta \in Bd^-(S)$$

("$\Rightarrow$") *trivial*

("$\Leftarrow$") *exercise*

Summary

- many practical pattern mining problems:
  - special cases of the theory extraction problem
- further restriction: partial order on the pattern language and an anti-monotone interestingness predicate w.r.t. the partial order
  - also satisfied by many practical problems
- levelwise algorithm
  - generates all interesting patterns
  - number of evaluations of the interestingness predicate is
    cardinality of the theory + cardinality of the negative border
- cardinality of the border is a sharp lower bound on the number of evaluations of the interestingness predicate by any algorithm accessing the data only by means of the evaluation of the interestingness predicate