Constraint-Based Pattern Set Mining

Luc De Raedt, *Albrecht Zimmermann*

DTAI Lab
Katholieke Universiteit Leuven

April 26 2007 - 7th Siam International Conference on Data Mining

Rough Outline

1. Motivation
2. Constrained Pattern Set Mining
   - The theory
   - The practice
3. Evaluation

The First Step - Local Pattern Mining

Given:

- A pattern language $\mathcal{L}$
- A dataset $\mathcal{D}$ described in $\mathcal{L}$
- Some constraint q

Find:

- $Th(\mathcal{L}, \mathcal{D}, q)$ - the theory (set of patterns) $\in 2^{\mathcal{L}}$ whose members $\varphi \subseteq \mathcal{L}$ satisfy the constraint

The First Step - Local Pattern Mining

Given:

- A pattern language $\mathcal{L}$
- A dataset $\mathcal{D}$ described in $\mathcal{L}$
- Some constraint q

Find:

- $Th(\mathcal{L}, \mathcal{D}, q)$ - the theory (set of patterns) $\in 2^{\mathcal{L}}$ whose members $\varphi \subseteq \mathcal{L}$ satisfy the constraint

Apriori Calling

- $\mathcal{L}$ a set of items $\mathcal{I}$
- $\mathcal{D}$ a transaction database $\mathcal{T} \subset 2^{\mathcal{I}}$
- $q := sup_{\varphi \subseteq \mathcal{I}}(\varphi) \geq 1\%$
- $Th(\mathcal{I}, \mathcal{T}, q)$ - frequent itemset mining

The Second Step

$Th \Leftrightarrow$ Set of Patterns

- Patterns capture local anomalies
- Many (hundreds to millions), possibly redundant, dubious informational value

Usage

- Visual inspection by the user
- Features in e.g. classification task
- Cluster description
- Building blocks for database compression
- A.k.a. building global model

The Second Step

$Th \Leftrightarrow$ Set of Patterns

- Patterns capture local anomalies
- Many (hundreds to millions), possibly redundant, dubious informational value

Usage

- Visual inspection by the user
- Features in e.g. classification task
- Cluster description
- Building blocks for database compression
- A.k.a. building global model
- Sabotaged by solution characteristics

Possible Solutions

Ad-hoc Restriction

- No redundance - closed, free patterns
- Most specific/general ones - maximal patterns, version space borders
- Non-derivable patterns
- Correlating patterns

Post-Processing

- Prune based on database coverage - CBA
- Select optimally (MDL) compressing patterns - Siebes *et al.*

Possible Solutions

Ad-hoc Restriction

- No redundance - closed, free patterns
- Most specific/general ones - maximal patterns, version space borders
- Non-derivable patterns
- Correlating patterns

Post-Processing

- Prune based on database coverage - CBA
- Select optimally (MDL) compressing patterns - Siebes *et al.*

We Say!

More control to the user - also in Step 2!

The Framework

Pattern Sets

- Combination of patterns, interpreted as acting "together" (logical OR)
- User decides on desirable properties, e.g.
  - Maximum size - interpretability
  - Maximum overlap of coverage - redundancy
  - Inclusion/exclusion of certain patterns - knowledge gain

The Framework

Pattern Sets

- Combination of patterns, interpreted as acting "together" (logical OR)
- User decides on desirable properties, e.g.
  - Maximum size - interpretability
  - Maximum overlap of coverage - redundancy
  - Inclusion/exclusion of certain patterns - knowledge gain

Attention - Spoiler information follows! :)

- $\mathbf{L} := Th(\mathcal{I}, \mathcal{T}, \min_{\varphi \subseteq \mathcal{I}} \geq 1\%)$
- $\mathcal{T}$
- p - $\max red_{\Phi \subseteq \mathbf{L}}(\Phi) \leq 5$
- $\mathbf{Th}(\mathbf{L}, \mathcal{T}, p)$ - set of pattern sets whose members Φ satisfy the constraint

Some Primitives

Support

- "Meaning" - User wants certain number of instances covered by disjunction of patterns in Φ
- Supports reachable that no single pattern attains

Redundancy

- "Meaning" - Certain number of instances covered by union of any pair of patterns in Φ
- Large overlaps $\longrightarrow$ same information presented to user again and again

Size

- "Meaning" - Certain number of patterns included in Φ
- Humans can (and will) only explore so many patterns (Google effect...)

Towards Pattern Set Mining

Properties

- Involving user leverages background knowledge/intuition
- Constraints show well-known properties:
  - (Anti-)Monotone - (Minimum) Maximum redundancy
  - Boundable - Minimum χ^2
  - Convertible - Average support
- Allows reuse of local pattern mining research
- Probably local pattern mining problems will appear
- Relationship size and generality is "flipped"

Possible Algorithms

- Level-wise
- Branch-and-bound
- "Closed/Free" pattern set mining

"Flipped" Relationship

The Phenomenon

- Local pattern mining
  - Conjunction of "items"
  - More items $\Rightarrow$ probability of matching goes down
  - Shorter itemsets more general than their extensions
- Pattern set mining
  - Disjunction of patterns
  - More patterns $\Rightarrow$ probability of matching goes up
  - Smaller pattern sets more specific than their extensions

"Flipped" Relationship

The Phenomenon

- Local pattern mining
  - Conjunction of "items"
  - More items $\Rightarrow$ probability of matching goes down
  - Shorter itemsets more general than their extensions
- Pattern set mining
  - Disjunction of patterns
  - More patterns $\Rightarrow$ probability of matching goes up
  - Smaller pattern sets more specific than their extensions

The Effect

- Goal: small (item- or pattern) sets (don't overwhelm the user)
- Start at most specific (general) for pattern (item-) sets
- Use (anti-)monotone selective constraint

Experimental Evaluation

Does It Work?

- Exhibiting the same characteristics as known pattern mining?
- Useful for anything?

Set-Ups

- Level-wise algorithm
- Frequent patterns mined on sequence, itemset data
  - Several queries run
  - Varying thresholds
  - Observing size, number pattern sets
- *CAR*s on balance-scale
  - Looking for best low-redundance, high-accuracy pattern set
  - Compare to CBA's database coverage pruning

Experiments - General Mining

Results

Numbers are in the paper

Observations

- Loose thresholds - many result sets
- Relatively short - redundancy helps keep combinatorials down
- Dense DS $\longrightarrow$ many patterns $\longrightarrow$ very dense DS for pattern set mining

Experiments - General Mining

Results

Numbers are in the paper

Observations

- Loose thresholds - many result sets
- Relatively short - redundancy helps keep combinatorials down
- Dense DS $\longrightarrow$ many patterns $\longrightarrow$ very dense DS for pattern set mining

Conclusion

Behaves as expected

Experiments - Classifier Building

Results

- Smaller pattern set: 17.4 ± 5.1 vs 4.9 ± 1.66
- Same training accuracy: 0.8443 ± 0.019
- Same testing accuracy: 0.7918 ± 0.0459

Observations

- First pattern (rule) not included!
- Heuristic method good accuracy
- Pattern set mining $\rightarrow$ easier to understand classifier
- Longer running times

Experiments - Classifier Building

Results

- Smaller pattern set: 17.4 ± 5.1 vs 4.9 ± 1.66
- Same training accuracy: 0.8443 ± 0.019
- Same testing accuracy: 0.7918 ± 0.0459

Observations

- First pattern (rule) not included!
- Heuristic method good accuracy
- Pattern set mining $\rightarrow$ easier to understand classifier
- Longer running times

Conclusion

Does what it's supposed to do

Summary

- Introduced new framework for more global pattern mining
- Identified meaningful constraints for new context
- Mining task shows many characteristics of local pattern mining
- Allows leveraging of existing knowledge
- Need for more (selective) constraints

Summary

- Introduced new framework for more global pattern mining
- Identified meaningful constraints for new context
- Mining task shows many characteristics of local pattern mining
- Allows leveraging of existing knowledge
- Need for more (selective) constraints
- Use pattern set mining experiences to modify the underlying task...

Thank you for your attention

?