



# Relational Redescription Mining

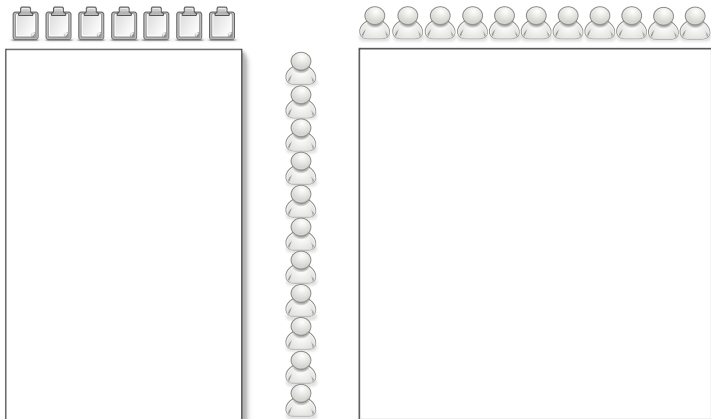
Esther Galbrun

joint work with  
Pauli Miettinen and Angelika Kimmig

Helsinki Institute for Information Technology  
Department of Computer Science, University of Helsinki



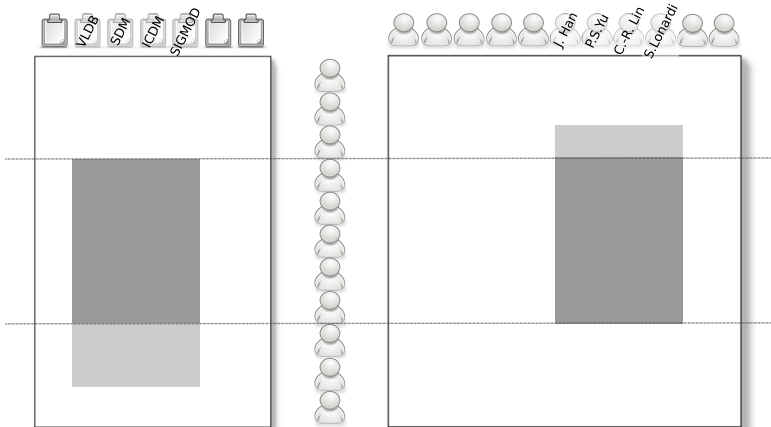
## Example: DBLP Data





## Example: DBLP Data

$VLDB \wedge ICDM \wedge SDM \wedge SIGMOD$   
 $(J. Han \wedge P.S. Yu) \vee C.-R. Lin \vee S. Lonardi$





## Definition

Redescription Given two datasets over the same entities, a **redescription** is a pair of queries ( $q_L, q_R$ ) over the two dataset respectively, characterizing approximately the same sets of entities.



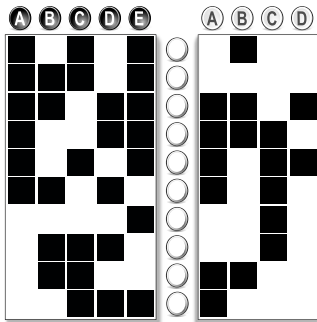
## Aims

- Find coherent sets of objects
- Find sets of related attributes
- View the same objects under different perspectives



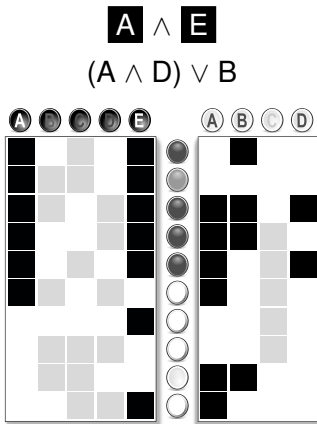
# Boolean Redescriptions

Dataset Boolean matrices





# Boolean Redescriptions



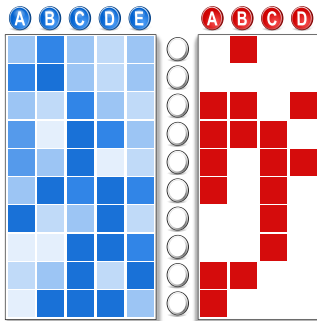
Dataset Boolean matrices  
 Queries Boolean formulae  
 Accuracy Jaccard coefficient

$$J(q_L, q_R) = \frac{|\text{supp}(q_L) \cap \text{supp}(q_R)|}{|\text{supp}(q_L) \cup \text{supp}(q_R)|}$$

$$= \frac{|E_{1,1}|}{|E_{1,0}| + |E_{1,1}| + |E_{0,1}|}$$



# Real-Valued Redescriptions



Dataset Real-valued matrices

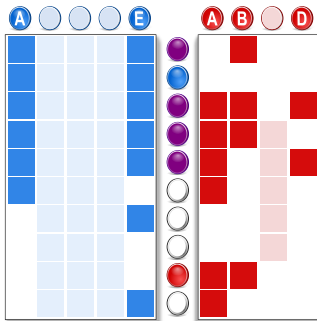




# Real-Valued Redescriptions

$$[\square \leq A \leq \square] \wedge [\square \leq E \leq \square]$$

$$(A \wedge D) \vee B$$



**Dataset** Real-valued matrices

**Queries** Intervals

**Accuracy** Jaccard coefficient

$$J(q_L, q_R) = \frac{|\text{supp}(q_L) \cap \text{supp}(q_R)|}{|\text{supp}(q_L) \cup \text{supp}(q_R)|}$$

$$= \frac{|E_{1,1}|}{|E_{1,0}| + |E_{1,1}| + |E_{0,1}|}$$



# Geospatial Redescriptions

SIREN :: tools

File Redescriptions Window Help

LHS Variables RHS Variables Redescriptions Expanding

	Query LHS	Query RHS	Acc ↑	p-Value	Support
1	<input type="checkbox"/> Polar bear	$[1.0 \leq t^* \leq 3.5]$	0.973	0.0	36
2	<input type="checkbox"/> Polar bear	$[-9.6 \leq t^* \leq -5.6]$	0.973	0.0	36
3	<input checked="" type="checkbox"/> Polar bear	$[-7.0727 \leq t^* \leq -3.375]$			
4	<input type="checkbox"/> Polar bear	$[-4.5 \leq t^* \leq -1.0]$			
5	<input type="checkbox"/> Polar bear	$[-16.694 \leq t^* \leq -11.462]$			
6	<input type="checkbox"/> Polar bear	$[-11.9 \leq t^* \leq -7.3]$			
7	<input checked="" type="checkbox"/> Wood mouse v Azores Noctule	$(( [3.0 \leq t^* ] \wedge [9.8 \leq t^* ] ))$			
8	<input checked="" type="checkbox"/> Wood mouse v Azores Noctule v Harp Seal	$(( [2.9 \leq t^* ] \vee [9.7 \leq t^* ] ))$			
9	<input checked="" type="checkbox"/> Bank Vole v Northern Red-backed Vole v Steppe Mouse v	$[-9.2 \leq t^* \leq 12.8] \wedge [7.15 \dots]$			
10	<input type="checkbox"/> Wood mouse v Azores Noctule	$(( [2.9 \leq t^* ] \wedge [8.3 \leq t^* ] ))$			
11	<input type="checkbox"/> Wood mouse v Azores Noctule v Harp Seal	$(( [-0.8 \leq t^* ] \wedge [-0.14118 \leq \dots]$			
12	<input checked="" type="checkbox"/> Wood mouse v Harp Seal	$(( [-0.8 \leq t^* \leq 17.2] \wedge [-4.9 \dots]$			
13	<input type="checkbox"/> Wood mouse v Harp Seal	$(( [-9.4 \leq t^* \leq 8.2] \wedge [-8.3 \leq \dots]$			
14	<input checked="" type="checkbox"/> Wood mouse	$(( [3.0 \leq t^* ] \wedge [4.2 \leq t^* ] ))$			
15	<input type="checkbox"/> Wood mouse	$(( [9.7 \leq t^* \leq 13.2] \vee [-5.16 \dots]$			
16	<input type="checkbox"/> Bank Vole v Northern Red-backed Vole v Steppe Mouse v	$(( [11.2 \leq t^* \leq 13.4] \vee [13. \dots]$			
17	<input checked="" type="checkbox"/> Arctic Fox v Stoat	$(( [2.6 \leq t^* \leq 8.5] \vee [7.2 \leq \dots]$			
18	<input checked="" type="checkbox"/> Stoat v Walrus	$(( [7.2 \leq t^* \leq 22.2] \vee [21.1 \dots]$			
19	<input type="checkbox"/> Stoat v Walrus	$(( [11.6 \leq t^* \leq 25.3] \vee [21.1 \dots]$			
20	<input type="checkbox"/> Arctic Fox v Stoat	$(( [t^* \leq 25.5] \wedge [0.68824 \leq \dots]$			
21	<input checked="" type="checkbox"/> Arctic Fox v Stoat	$(( [0.8 \leq t^* \leq 13.9] \wedge [t^* \leq \dots]$			
22	<input checked="" type="checkbox"/> Cape Horn v European Harp v Alaskan Moose	$(( [12.0 \leq t^* \leq 8.0] \wedge [15.5 \dots]$			

SIREN :: maps

Moose

$[ -10.0 \leq t^* \leq 0.0 ] \wedge [ 12.0 \leq t^* \leq 25.0 ]$

J= 0.74513  
pVal= 0.00000

|LHS ∩ RHS|= 535  
|LHS ∪ RHS|= 718

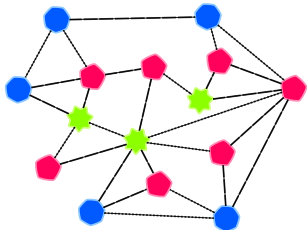
|LHS \ RHS|= 130  
|RHS \ LHS|= 53

Expand Stop



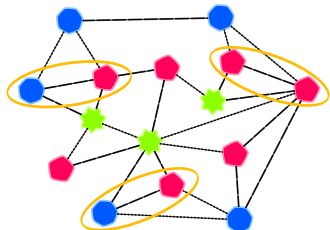
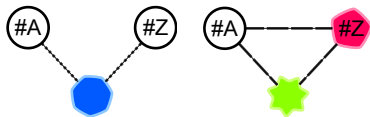
# Relational Redescriptions

**Dataset** A network with node and edge attributes





# Relational Redescriptions



**Dataset** A network with node and edge attributes

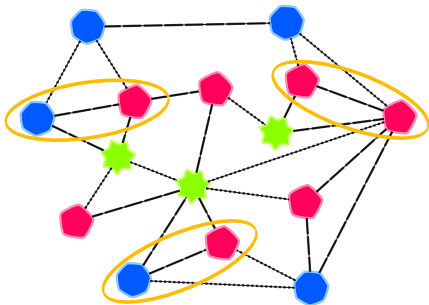
**Queries** Connection patterns

**Accuracy** Jaccard coefficient

$$J(q_L, q_R) = \frac{|\text{supp}(q_L) \cap \text{supp}(q_R)|}{|\text{supp}(q_L) \cup \text{supp}(q_R)|}$$

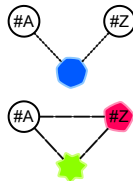


## Definition



**Dataset** A network with node and edge attributes

**Task** Find structurally different patterns covering (almost) the same pairs of nodes.



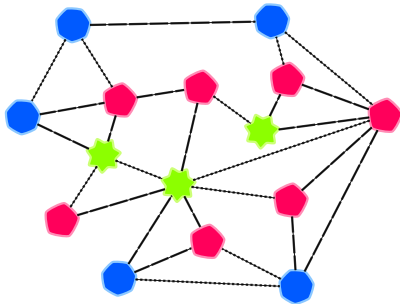


## Entities and queries

- ✓ **pairs of nodes** and their connections.
- ✗ *individual nodes* and surrounding relations.
- ✗ *a transactional graph* and occurring subgraphs.



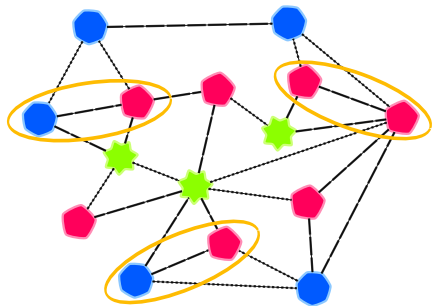
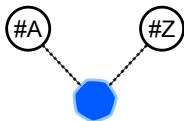
# Alternating Scheme





# Alternating Scheme

1. Fix a pattern to obtain examples

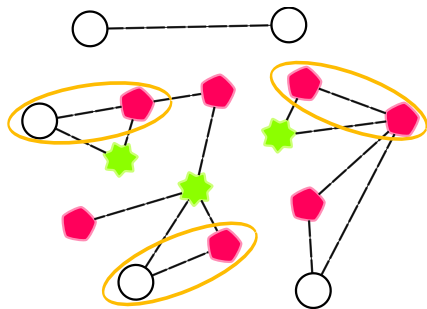
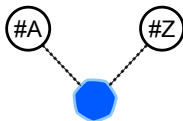






## Alternating Scheme

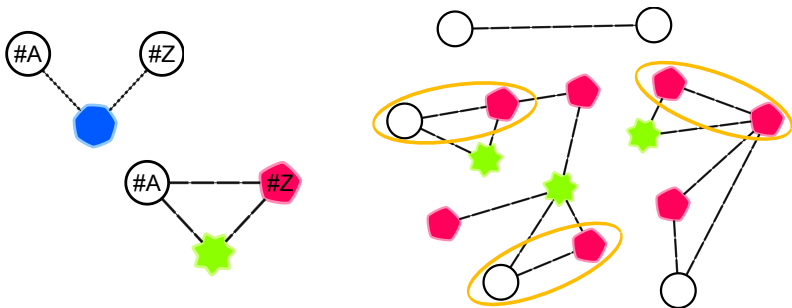
1. Fix a pattern to obtain examples
2. Consider remaining attributes





## Alternating Scheme

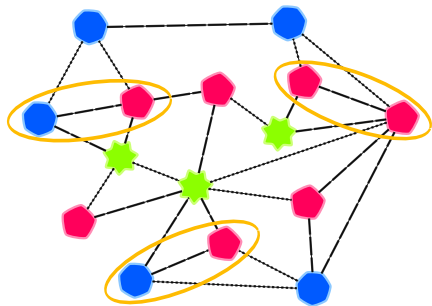
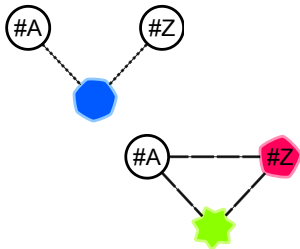
1. Fix a pattern to obtain examples
2. Consider remaining attributes
3. Find a matching pattern





## Alternating Scheme

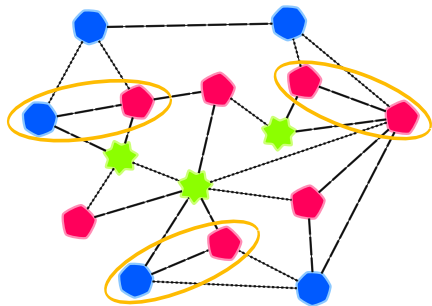
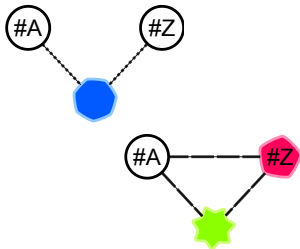
1. Fix a pattern to obtain examples
2. Consider remaining attributes
3. Find a matching pattern
4. Swap roles and iterate





## Alternating Scheme

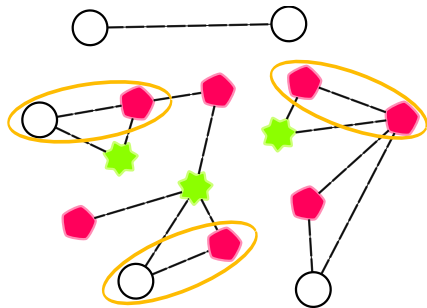
1. Fix a pattern to obtain examples
2. Consider remaining attributes
3. **Find a matching pattern**
4. Swap roles and iterate





## Subproblem: Query mining

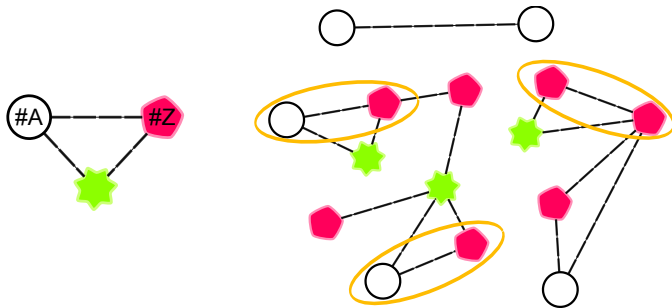
- ✓ Given a set of examples  
and a subset of attributes
- **Find a matching pattern**





## Subproblem: Query mining

- ✓ Given a set of examples  
and a subset of attributes
- **Find a matching pattern**





## FpQm: Stepwise Approach

1. Enumerate connecting paths  
and mine frequent path patterns
2. Build graph patterns from path patterns
3. Select a subset of graph patterns

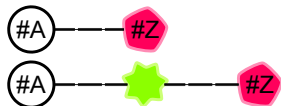
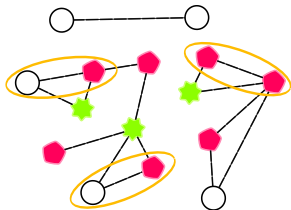


# 1. Find path patterns

Starting with paths of length  $k = 1$

1. Enumerate connecting paths
2. Mine frequent path patterns
3. Increase  $k$  by one and iterate

**Until** all examples are connected  
or  $k$  exceeds a chosen threshold



**Outcome** a set of frequent path patterns





## 2. Build graph patterns

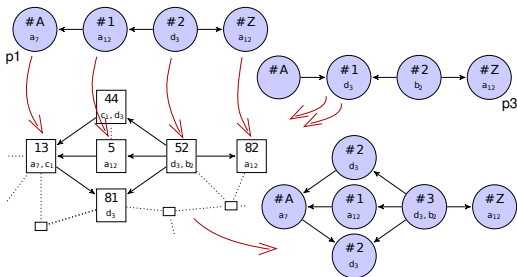
- ✓ Given a set of path patterns and of examples
- Combine paths to build graph patterns



## 2. Build graph patterns

- ✓ Given a set of path patterns and of examples
- Combine paths to build graph patterns

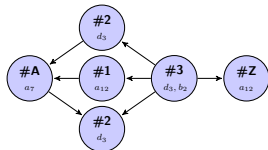
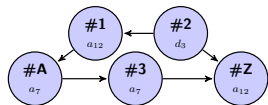
Combination based on the instances





## 2. Build graph patterns

- ✓ Given a set of path patterns and of examples
- Combine paths to build graph patterns



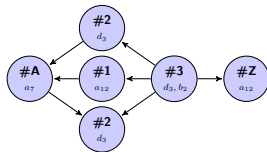
Outcome a set of graph patterns



### 3. Select graph patterns

- ✓ Given a set of graph patterns and of examples
- Select a good cover

**Outcome** a small set of graph patterns best matching the examples





## FpQm

1. Enumerate connecting paths  
and mine frequent path patterns
2. Build graph patterns from path patterns
3. Select a subset of graph patterns

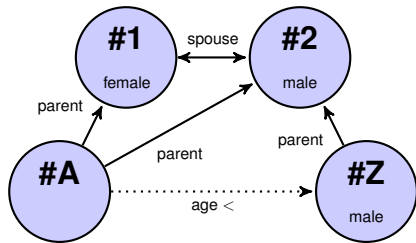
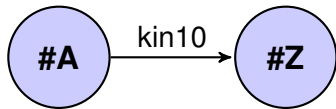


## Alternating Scheme

- 1: initialize candidates
- 2: **for** each candidate **do**
- 3:     **for** each matching clause found with FpQm **do**
- 4:         **if** turns limit not reached and no equivalent clause **then**
- 5:             add to candidates
- 6: extract good pairs of adjacent clauses from the exploration tree



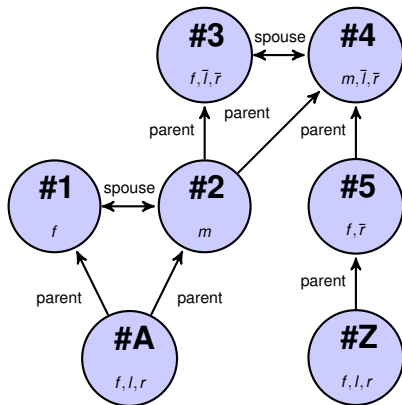
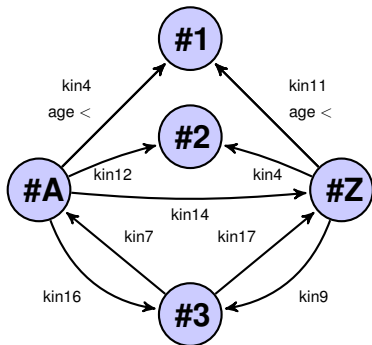
## Examples from Kinship



"Older brother"

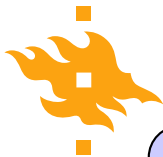


# Examples from Kinship

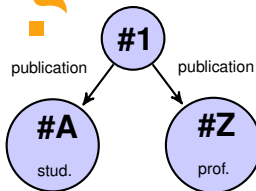


$$|E_{1,1}| = 23, J = 0.54$$

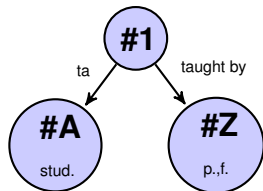
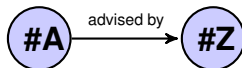




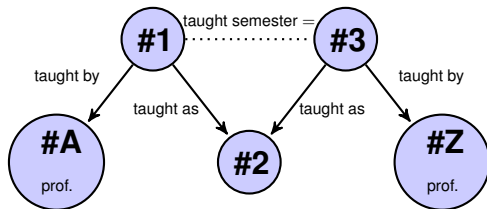
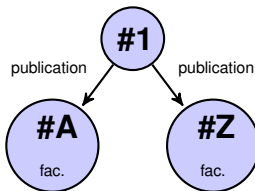
## Examples from UW-CSE



$$|E_{1,1}| = 41, J = 0.24$$



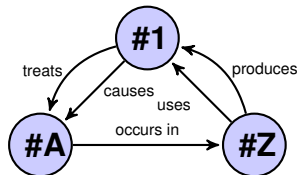
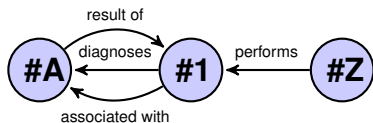
$$|E_{1,1}| = 21, J = 0.10$$



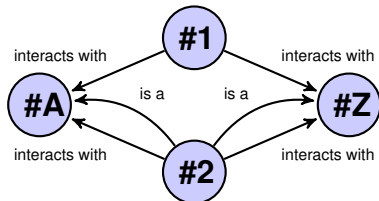
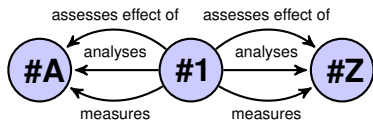
$$|E_{1,1}| = 24, J = 0.14$$



## Examples from UMLS



$$|E_{1,1}| = 23, J = 0.54$$



$$|E_{1,1}| = 506, J = 0.62$$



## But this is just ILP!?!...

### ILP tools (*from my uninitiated point of view*)

- general approach, encompassing various strategies
- progressive generalization / refinement of clauses
- heavy use of background knowledge, bias, types and co.

### FpQM

- adapted to finding linked patterns
- purely data based, no additional knowledge
- relies on frequent paths

Experimental comparison: our approach out-performed c-armr on this task



## How does it scale?

Dataset	$ N $	$ E $	#np.	#ep.	#cp.	$ R $	$ M $	Tot. T	T/clause	
									max	avg
Kinship	381	24053	3	31	1	96	340	3h 36min	254s	38s
Umls	135	4181	–	46	–	15	81	13min 29s	79s	10s
Uwcse	1042	1674	6	7	5	8	25	39s	4s	2s

Strong impact on the running times:

- Network density
- Presence of symmetries



## Relational Redescription Mining

- Find structurally different patterns covering (almost) the same pairs of nodes.
- An expressive tool for finding corresponding connections patterns in a network.



## Relational Redescription Mining

- Find structurally different patterns covering (almost) the same pairs of nodes.
- An expressive tool for finding corresponding connections patterns in a network.

*Thank you ...*