Making Sense of (Multi-)Relational Data

Part III: Exploration by Descriptive Modelling – Semi-Relational Local Approaches

Jefrey Lijffijt

Eirini Spyropoulou

Tijl De Bie

Relational Data Model

Relational Data Model

- ER data model with attributes as entities user_likes_film

users	films
u1	f1
u1	f2
u2	f1
u2	f2
u3	f2
u3	f3
u4	f2
u4	f3

film_actor

films	actors
f1	a1
f1	a3
f2	a1
f2	a3
f3	a2

films

users

	f1	f2	f3
u1	1	1	0
u2	1	1	0
u3	0	1	1
u4	0	1	1

actors

films

	a1	a2	a3
f1	1	0	1
f2	1	0	1
f3	0	1	0

Entity types

films

users

	f1	f2	f3
u1	1	1	0
u2	1	1	0
u3	0	1	1
u4	0	1	1

actors

films

	a1	a2	a3
f1	1	0	1
f2	1	0	1
f3	0	1	0

Entities

$$E = \{f1, f2, f3, u1, u2, u3, a1, a2, a3\}$$

films

users

	f1	f2	f3
u1	1	1	0
u2	1	1	0
u3	0	1	1
u4	0	1	1

actors

films

	a1	a2	a3
f1	1	0	1
f2	1	0	1
f3	0	1	0

Entities

$$E = \{f1, f2, f3, u1, u2, u3, a1, a2, a3\}$$

$t(f1) =$ films

films

users

	f1	f2	f3
u1	1	1	0
u2	1	1	0
u3	0	1	1
u4	0	1	1

actors

films

	a1	a2	a3
f1	1	0	1
f2	1	0	1
f3	0	1	0

Relationship types

$$R = \{\{users, films\}, \{films, actors\}\}$$

films

users

	f1	f2	f3
u1	1	1	0
u2	1	1	0
u3	0	1	1
u4	0	1	1

actors

films

	a1	a2	a3
f1	1	0	1
f2	1	0	1
f3	0	1	0

Relationships

films

users

	f1	f2	f3
u1	1	1	0
u2	1	1	0
u3	0	1	1
u4	0	1	1

actors

films

	a1	a2	a3
f1	1	0	1
f2	1	0	1
f3	0	1	0

Relationships

films

$R_{users,films}$

users

	f1	f2	f3
u1	1	1	0
u2	1	1	0
u3	0	1	1
u4	0	1	1

actors

$R_{films,actors}$

films

	a1	a2	a3
f1	1	0	1
f2	1	0	1
f3	0	1	0

Relationships

$$\mathcal{R} = R_{users,films} \cup R_{films,actors}$$

films

$R_{users,films}$

users

	f1	f2	f3
u1	1	1	0
u2	1	1	0
u3	0	1	1
u4	0	1	1

actors

$R_{films,actors}$

films

	a1	a2	a3
f1	1	0	1
f2	1	0	1
f3	0	1	0

Semi-relational local algorithms

- Frequent itemset mining on the join
- Smurfig

Frequent itemset mining on the join

- Approach used in practice
- Join all database relations
- Apply frequent itemset mining on the join table
- Transactions: tuples of the join table
- Items: all attribute values

Example

user_likes_film

users	films
u1	f1
u1	f2
u2	f1
u2	f2

film_actor

films	actors
f1	a1
f1	a3
f2	a1
f2	a3

Example

user_likes_film

users	films
u1	f1
u1	f2
u2	f1
u2	f2

$\bowtie$

film_actor

films	actors
f1	a1
f1	a3
f2	a1
f2	a3

transactions

users	films	actors
u1	f1	a1
u1	f1	a3
u1	f2	a1
u1	f2	a3
u2	f1	a1
u2	f1	a3
u2	f2	a1
u2	f2	a3

items

	u1	u2	f1	f2	a1	a3
t1	1	0	1	0	1	0
t2	1	0	1	0	0	1
t3	1	0	0	1	1	0
t4	1	0	0	1	0	1
t5	0	1	1	0	1	0
t6	0	1	1	0	0	1
t7	0	1	0	1	1	0
t8	0	1	0	1	0	1

frequent itemset

transactions

users	films	actors
u1	f1	a1
u1	f1	a3
u1	f2	a1
u1	f2	a3
u2	f1	a1
u2	f1	a3
u2	f2	a1
u2	f2	a3

items

	u1	**u2**	**f1**	f2	a1	a3
t1	1	0	1	0	1	0
t2	1	0	1	0	0	1
t3	1	0	0	1	1	0
t4	1	0	0	1	0	1
t5	0	1	1	0	1	0
t6	0	1	1	0	0	1
t7	0	1	0	1	1	0
t8	0	1	0	1	0	1

Issues

- Join is a costly operation

Issues

- What is the support of the pattern?
    - Counting it with respect to the transactions does not have any physical meaning.
    - Counting it with respect to one attribute is affected by the replication of the values after the join.
    - Which attribute to chose?

Example

user_likes_film

users	films
u1	f1
u1	f2
u2	f1
u2	f2

$\bowtie$

film_actor

films	actors
f1	a1
f1	a3
f2	a1
f2	a3

transactions

users	films	actors
u1	f1	a1
u1	f1	a3
u1	f2	a1
u1	f2	a3
u2	f1	a1
u2	f1	a3
u2	f2	a1
u2	f2	a3

items

	u1	u2	f1	f2	a1	a3
t1	1	0	1	0	1	0
t2	1	0	1	0	0	1
t3	1	0	0	1	1	0
t4	1	0	0	1	0	1
t5	0	1	1	0	1	0
t6	0	1	1	0	0	1
t7	0	1	0	1	1	0
t8	0	1	0	1	0	1

Issues

- The pattern syntax does not capture all the association in the data

transactions

users	films	actors
u1	f1	a1
u1	f1	a3
u1	f2	a1
u1	f2	a3
u2	f1	a1
u2	f1	a3
u2	f2	a1
u2	f2	a3

items

	u1	u2	f1	f2	a1	a3
t1	1	0	1	0	1	0
t2	1	0	1	0	0	1
t3	1	0	0	1	1	0
t4	1	0	0	1	0	1
t5	0	1	1	0	1	0
t6	0	1	1	0	0	1
t7	0	1	0	1	1	0
t8	0	1	0	1	0	1

Smurfig (Goethals et al., 2010)

- Avoids the join by computing itemsets on every single table and the combining them.
- Solves support issue by choosing a single target attribute, counting on the original table and propagating the count.
- However still the pattern syntax only allows one attribute value per attribute.

References

1. B. Goethals, W. Le Page and Michael Mampaey. Mining interesting sets and rules in relational databases. SAC 2010.