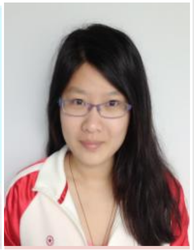
School of Computing Science
Simon Fraser University
Vancouver, Canada

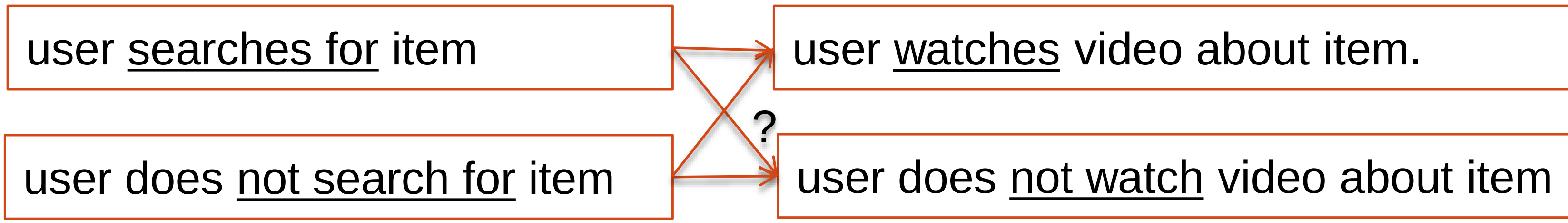
Computing Multi-Relational Sufficient Statistics for Large Databases

Zhensong Qian	Oliver Schulte	Yan Sun
		

Multi-Relational Sufficient Statistics

Why

- Find **correlations involving relationships**. e.g.



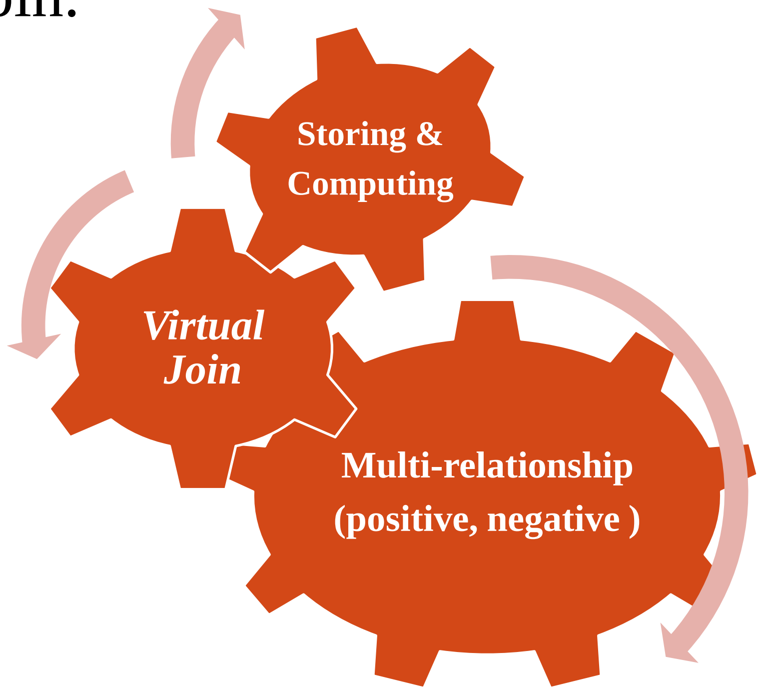
- Compactness: summarize original data by counts.

Previous Approaches

- Single-table data: row counts (σ selection only).
- Multiple tables: Table joins ⋈.

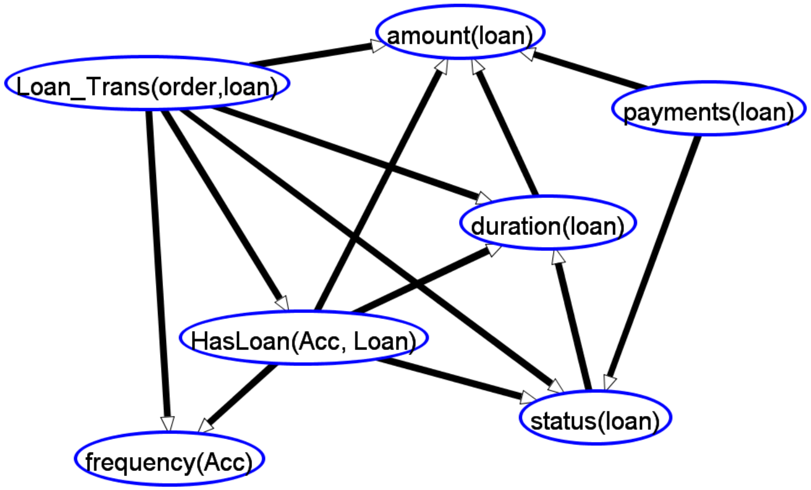
Contribution

- New approach for ***storing*** and ***computing*** multi-relational sufficient statistics including ***Negative relationships***.
- ***Virtual Join*** Algorithm: compute cross-table counts without materializing join.



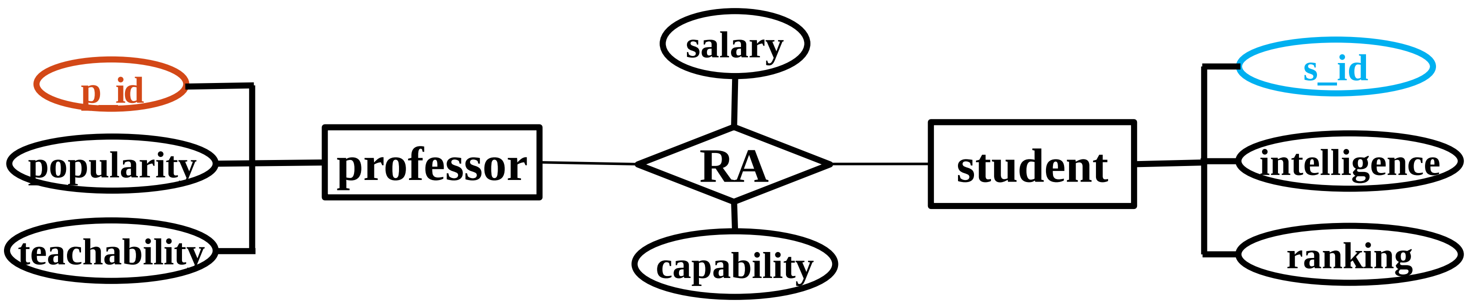
Applications

- Feature Selection.
  - Does **frequency of bank statement** predict whether **customer has loan**?
- Association Rules.
  - statement freq.(Acc) = monthly → HasLoan(Acc, Loan) = **?** .
- Bayesian Network Learning.
- …



E-R Diagram: Single Relationship

- We assume a database in Entity-Relationship format.
- Example for University domain with single Relationship.



Professor		
p_id	popularity	teachingability
Jim	2	1
Oliver	3	1
David	2	2

RA			
p_id	s_id	salary	capability
Oliver	Jack	High	3
Oliver	Kim	Low	1
Jim	Paul	Med	2
David	Kim	High	2

Student		
s_id	intelligence	ranking
Jack	3	1
Kim	2	1
Paul	1	2

Entity table: primary key; Relationship table: many-many, many-one

Contingency Tables (ct-table)

- Counts for **conjunctive queries**:
  - capability = value1, intelligence = value2.
  - capability = n/a: wasn't RA.
- **Conditional** ct-table :
  - e.g. given capability = 1.

Entity Table	Primary Key	# Tuples
Professor	p_id	6
Student	s_id	38
	Cross Product	228

capability	intelligence	count
1	2	3
1	3	2
2	2	3
2	3	1
3	1	2
3	2	4
3	3	1
4	1	1
4	3	4
5	1	1
5	2	3
N/A	1	80
N/A	2	65
N/A	3	58

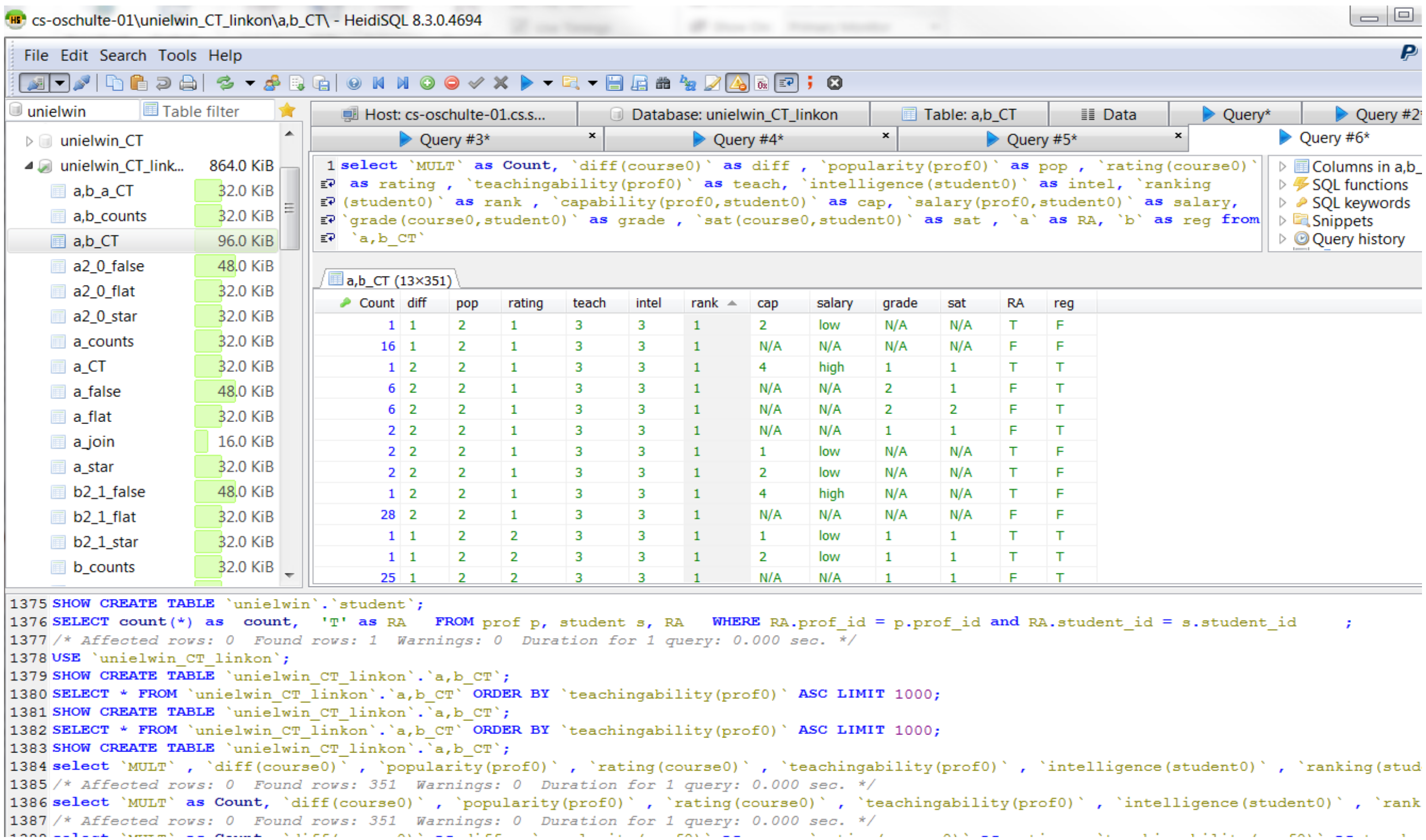
Sum(count) : 228

Total Tuples : 14

Storing Sufficient Statistics in Database Tables

- **New**: large contingency table **stored** as database table.

  Manipulate using SQL, Index, ...



Computing Sufficient Statistics: Positive Relationships only (e.g.RA=True)

CREATE TABLE **$ct_T(RA)$** **AS**

SELECT count(*) as count, pop, teach, intel, rank, cap, salary, 'T' **as** RA

FROM **Professor P, Student S, RA** cross-table count

WHERE RA.p_id = P.p_id **AND** RA.s_id = S.s_id

GROUP BY pop, teach, intel, rank, cap, salary

count	pop	teach	intel	rank	cap	salary	RA
2	2	2	3	1	4	high	T
2	2	3	1	4	3	med	T
1	1	2	2	2	1	med	T
1	1	2	2	2	2	med	T
1	1	2	2	2	3	low	T
1	**1**	**2**	**3**	**1**	**3**	**high**	**T**
…	…	…	…	…	…	…	…

Negative Relationships: Contingency Table Algebra

- Novel Contingency Table Algebra (ct-algebra):
  - Selection, Projection, **Conditioning**, Addition, **Subtraction**, **Cross Product**
  - Like relational algebra but with **count** column
- Implemented using SQL queries.
- New contingency algebra equation: basis for **virtual join**.
- Think "1-minus trick": P(not R) = 1 - P(R).

Computing Sufficient Statistics: Negative Relationship (e.g.RA=False)

- Equation for ct-table given RA = False :
  ct(Pop, Teach, Intelligence, Rank | **RA = False**) =

  ct(Prof) x ct(Student) −
  ct(Pop, Teach, Intelligence, Rank | **RA = True**)
- Compute counts for Negative relationship from true relationship and unspecified.
- Instantiates a **general equation** for arbitrary number of positive and negative relationships.

Step 1: Contingency Table Cross Product

- Example: ct(Professor) x ct(Student) → **ct$_*$(RA)**

ct(Professor)

Count	pop	teach
2	1	2
1	2	2
3	2	3

ct(Student)

Count	intel	rank
6	1	4
8	1	5
5	2	2
7	2	3
1	2	4
8	3	1
3	3	2

x

ct$_*$(RA)

Count	pop	teach	intel	rank
12	1	2	1	4
16	1	2	1	5
10	1	2	2	2
14	1	2	2	3
2	1	2	2	4
16	1	2	3	1
6	1	2	3	2
6	2	2	1	4
8	2	2	1	5
5	2	2	2	2
7	2	2	2	3
…	…	…	…	…

Step 2: Contingency Table Subtraction

$ct_*(RA)$

Count	Pop	Teach	Intel	Rank
16	1	2	3	1
15	2	3	2	2
18	2	3	1	4
⋮	⋮	⋮	⋮	⋮

$ct_T(RA)$

Count	Pop	Teach	Intel	Rank
1	1	2	3	1
1	1	2	2	2
1	2	2	2	4
⋮	⋮	⋮	⋮	⋮

RA
T

—

Count	Pop	Teach	Intel	Rank
15	1	2	3	1
16	2	3	2	3
21	2	3	3	1
⋮	⋮	⋮	⋮	⋮

RA
F

∪

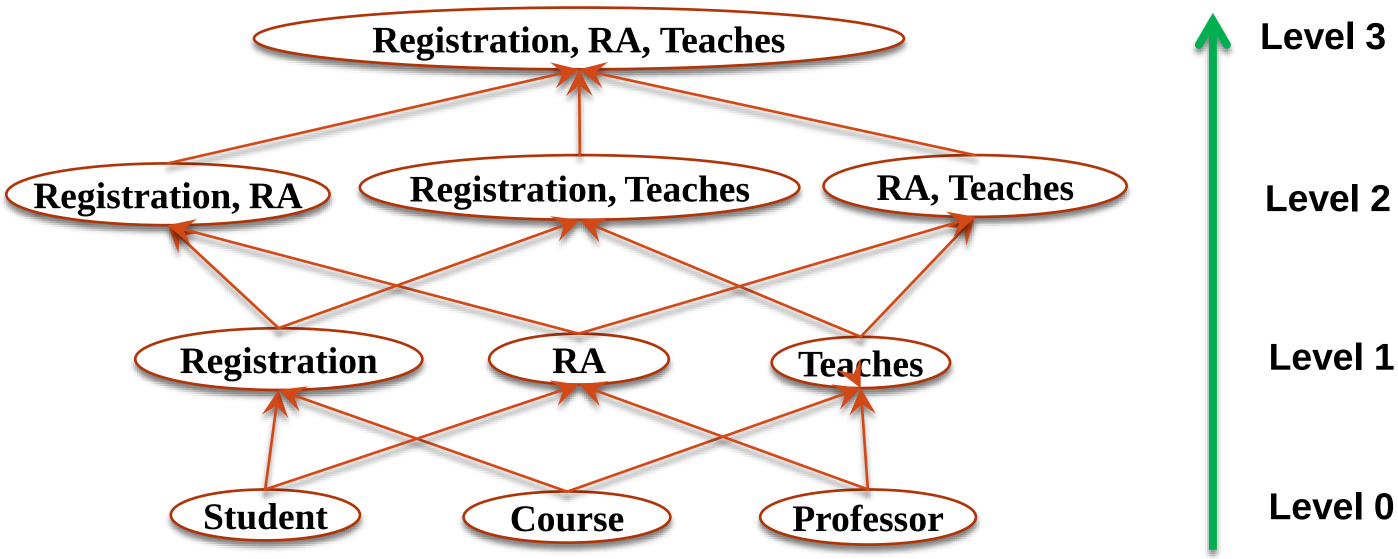
$ct(RA)$

Count	Pop	Teach	Intel	Rank	RA
15	1	2	3	1	F
1	1	2	3	1	T
1	1	2	2	2	T
⋮	⋮	⋮	⋮	⋮	⋮

Final Result: Contingency Table for RA relationship

Computation for Multiple Relationships: Dynamic Programming

- Build contingency tables for larger relationship chains from smaller ones using ct-algebra equation.



Lattice of Relationship Chains (Metapaths)

Datasets for Evaluation

7 Real-world Datasets (over 1M rows).

Dataset	#Relationship Tables/ Total	# Columns	# Rows
UW-CSE	2/4	14	712
Mondial	2/4	18	870
Hepatitis	3/7	19	12,927
Mutagenesis	2/4	11	14,540
Financial	3/7	15	225,932
Movielens	1/3	7	1,010,051
IMDB	3/7	17	1,354,134

Computation Time

- Never enumerates cross product of primary keys.
- Complexity: **nearly linear** in size of the required output.

  (non-trivial) #ct_operation = O(#SS * log (#SS))

Dataset	#Sufficient Statistics (SS)	Cross Product Time	Our Dynamic Program Time
Movielens	252	703.99	**2.70**
Mutagenesis	1,631	1,096.00	**1.67**
UW-CSE	2,828	350.30	**3.84**
Mondial	1,746,870	132.13	**1,112.84**
Financial	3,013,011	N.T.	**1,421.87**
Hepatitis	12,374,892	N.T.	**3,536.76**
IMDB	15,538,430	N.T.	**7,467.85**

(Time in seconds.)

Link Analysis Finds **New** Association Rules

- Link Analysis Off: only positive relationships occur.
- Link Analysis On: both positive and negative relationships may occur.

Dataset	MovieLens	Mutagenesis	Financial	Hepatitis	IMDB	Mondial	UW-CSE
# rules	14/20	20/20	12/20	15/20	20/20	16/20	12/20

- E.g. 12 rules with relationship correlation out of top-20 most interesting rules.
  - statement freq.(Acc) = monthly → HasLoan(Acc, Loan) = **T**.

Link Analysis Finds **New Relevant** Features for Classification

Dataset	Target Variable	# Selected Features	
		Link Analysis Off	Link Analysis On / Relationship Indicators
MovieLens	Horror	2	2/0
Mutagenesis	inda	3	3/0
Financial	**balance(trans)**	**3**	**2/1**
Hepatitis	sex	1	2/1
IMDB	avg_revenue	5	2/1
Mondial	percentage	Empty CT	4/0
UW-CSE	courseLevel	1	4/2

- New Features are selected with link analysis on.

E.g. **amount(trans), type(trans), frequency(acc)** V.S. **operation(trans), loan_trans(trans, loan)**

Link Analysis Finds **Better** Bayes Nets

- link analysis on: BNs achieve better model selection scores.

Financial	log-likelihood	#Parameter	R2R	A2R
Link Analysis Off	-10.96	11,572	0	0
Link Analysis On	**-10.74**	**2433**	2	9

IMDB	log-likelihood	#Parameter	R2R	A2R
Link Analysis Off	-13.63	181,896	0	0
Link Analysis On	**-11.39**	**60,059**	0	11

- Loan_Order(order, loan) → Has_loan(acc,loan)

  R2R: correlation between relationships.

- Loan_Order(order, loan) → Frequency(acc)

  A2R: correlation between attribute and relationship.

Conclusion

- Storing Sufficient statistics in database tables.
- Proposed contingency table algebra to support Virtual Join: compute cross-table counts without materializing joins.
- New DP Algorithm for computing multi-relational sufficient statistics with negative relationships.
- Efficient computation time.

Conclusion

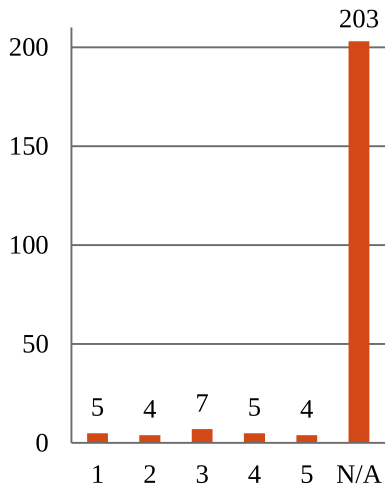
- Useful for many applications
  - association rule learning.
  - feature selection.
  - generative modelling.
  - ...
- MySQL/Java implementation available on-line.

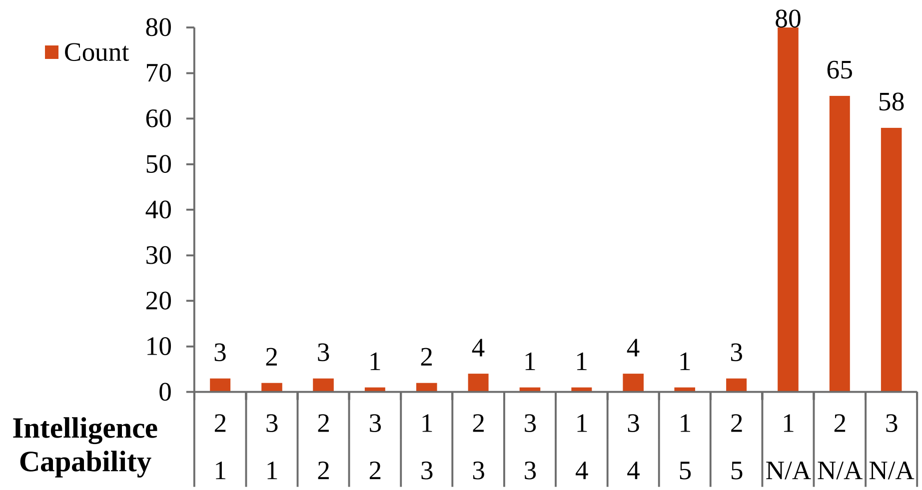
Future Work

- Scales well in number of rows in data tables.
- Does not scale well with number of columns/variables.

Thanks for your attention.

Backup Slide: count distribution





$ct_*(RA)$

ct(Professor)

Count	pop	teach
2	1	2
1	2	2
3	2	3

ct(Student)

Count	intel	rank
6	1	4
8	1	5
5	2	2
7	2	3
1	2	4
8	3	1
3	3	2

x

MULT	pop	teach	intel	rank
12	1	2	1	4
16	1	2	1	5
10	1	2	2	2
14	1	2	2	3
2	1	2	2	4
16	1	2	3	1
6	1	2	3	2
6	2	2	1	4
8	2	2	1	5
5	2	2	2	2
7	2	2	2	3
1	2	2	2	4
8	2	2	3	1
3	2	2	3	2
18	2	3	1	4
24	2	3	1	5
15	2	3	2	2
21	2	3	2	3
3	2	3	2	4
24	2	3	3	1
9	2	3	3	2

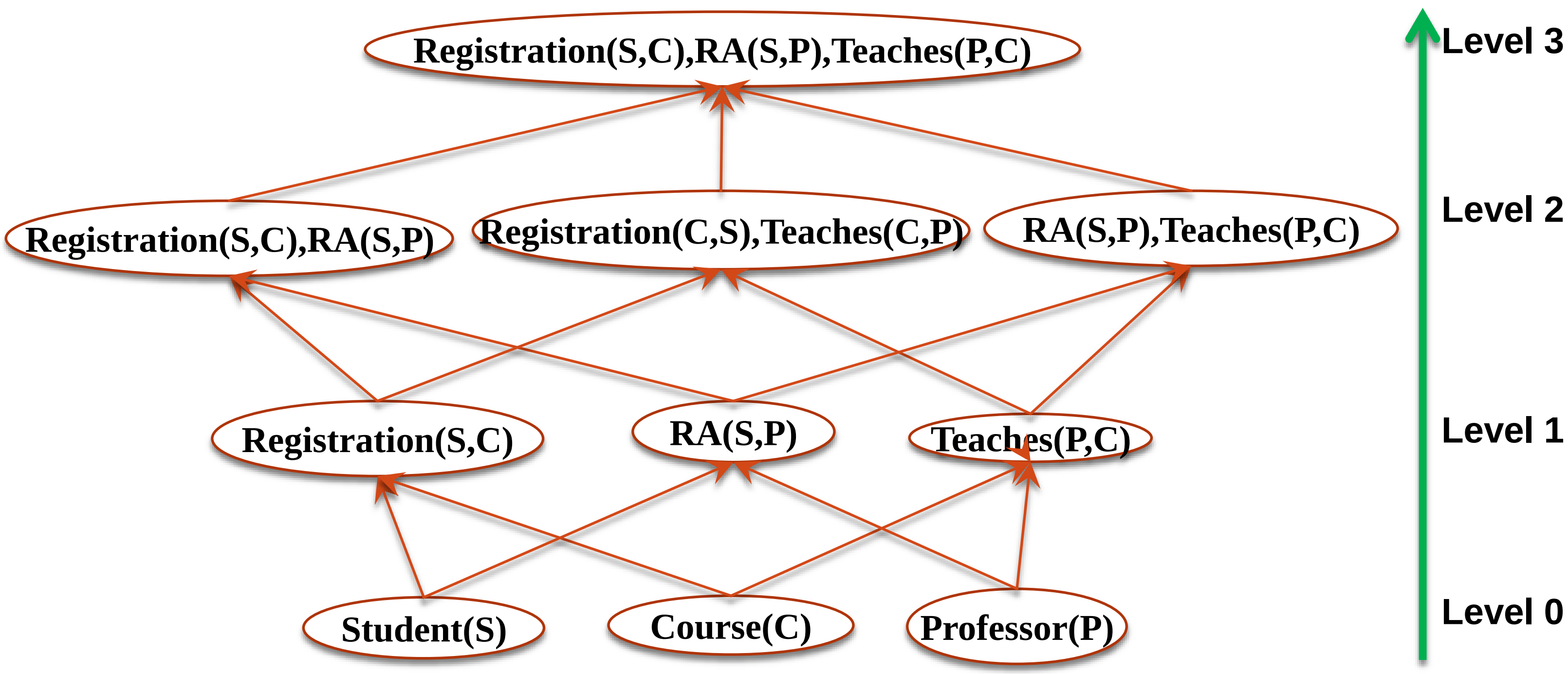
ct(Student) X ct(Professor) → $\mathbf{ct_*(RA)}$

Backup Slide: Compression Ratio

Dataset	MJ-time(s)	CP-time(s)	CP-#tuples	#Statistics	Compress Ratio
Movielens	2.70	703.99	23M	252	93,053.32
Mutagenesis	1.67	1096.00	1M	1,631	555.00
Financial	1421.87	N.T.	149,046,585M	3,013,011	49,467,653.90
Hepatitis	3536.76	N.T.	17,846M	12,374,892	1,442.19
IMDB	7467.85	N.T.	5,030,412,758M	15,538,430	323,740,092.05
Mondial	1112.84	132.13	5M	1,746,870	2.67
UW-CSE	3.84	350.30	10M	2,828	3,607.32

Table 3: Constructing the contingency table for each dataset. M = million. N.T. = non-termination. Compress Ratio = CP-#tuples/#Statistics.

Backup Slide: Lattice with functor notation



Dataset	Target Variable	# Selected Features	
		Link Analysis Off	Link Analysis On / Relationship Indicators
MovieLens	Horror	2	2/0
Mutagenesis	inda	3	3/0
Financial	balance	3	2/1
Hepatitis	sex	1	2/1
IMDB	avg_revenue	5	2/1
Mondial	percentage	Empty CT	4/0
UW-CSE	courseLevel	1	4/2

Dataset	Target variable	# Selected Attributes		Distinctness
		Link Analysis Off	Link Analysis On / Rvars	
MovieLens	Horror(M)	2	2 / 0	0.0
Mutagenesis	inda(M)	3	3 / 0	0.0
Financial	balance(T)	3	2 / 1	1.0
Hepatitis	sex(D)	1	2 / 1	0.5
IMDB	avg_revenue(D)	5	2 / 1	1.0
Mondial	percentage(C)	Empty CT	4 / 0	1.0
UW-CSE	courseLevel(C)	1	4 / 2	1.0