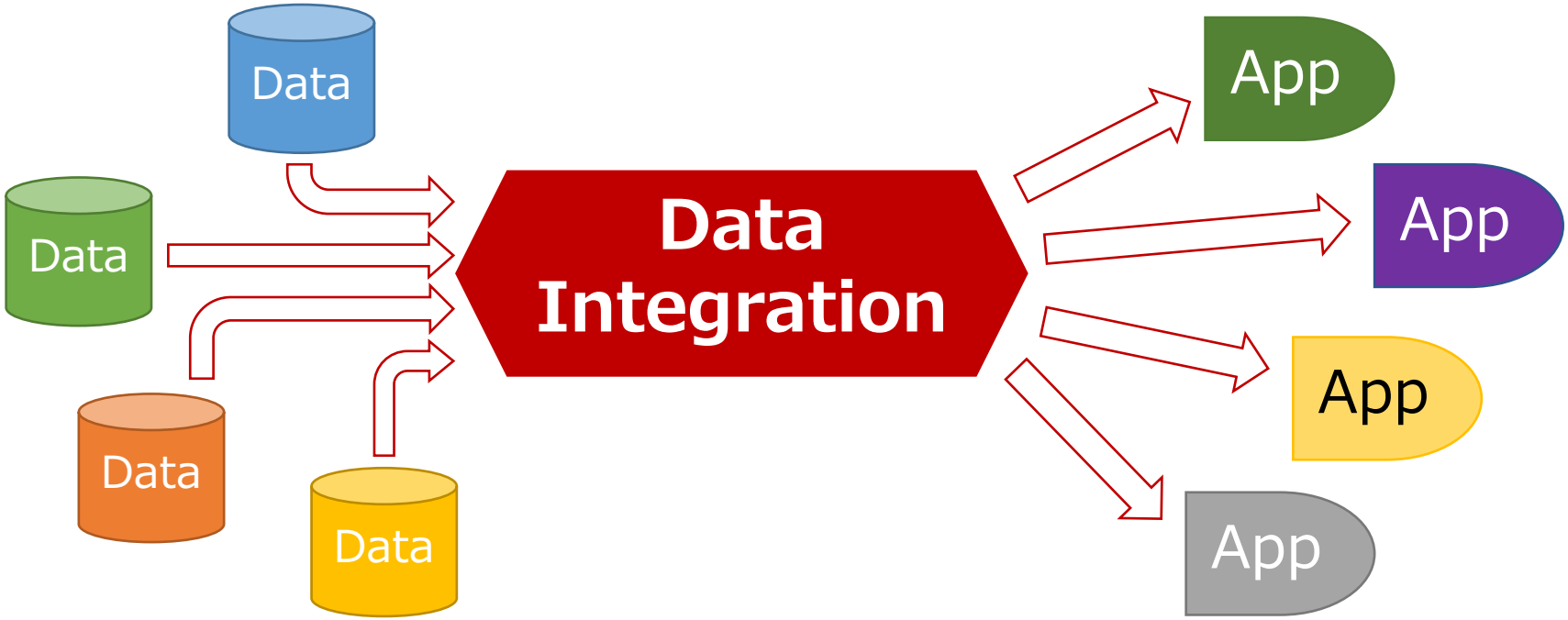
Implicit Order Join:
Joining Log Data with Property Data by Discovering Implicit Order-oriented Keys with Human Assistance

Takahiro Komamizu, Toshiyuki Amagasa, Hiroyuki Kitagawa

University of Tsukuba
Japan

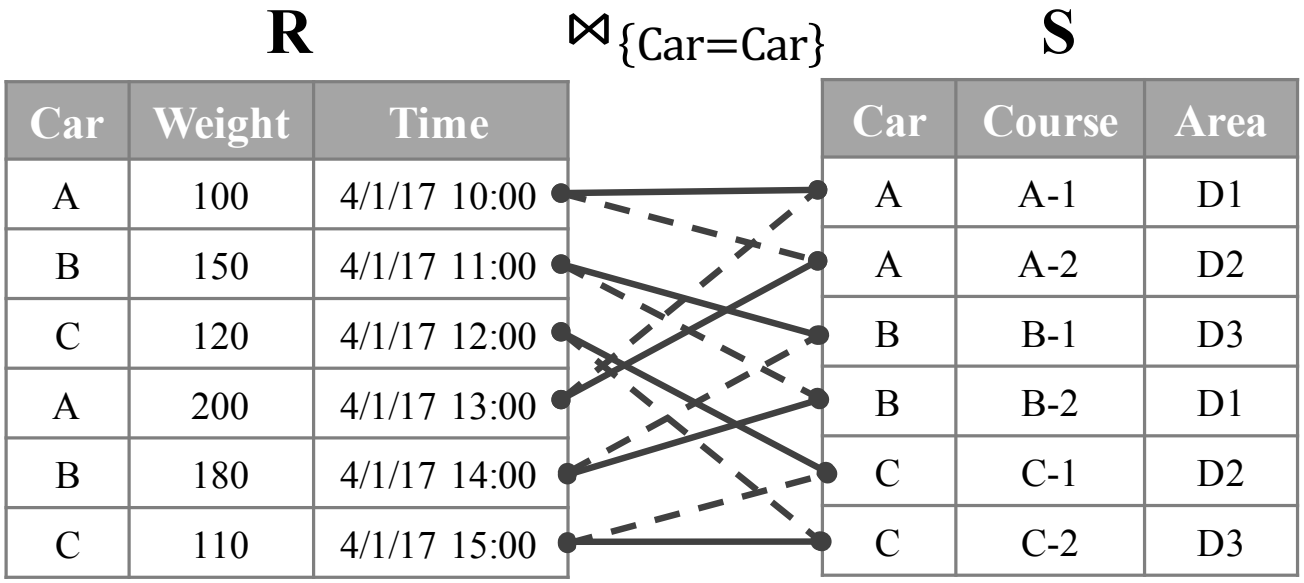
Data Integration

- Fundamental task for data analysis
- Combining data from multiple sources



Missing Key Problem

- Inconsistency of data



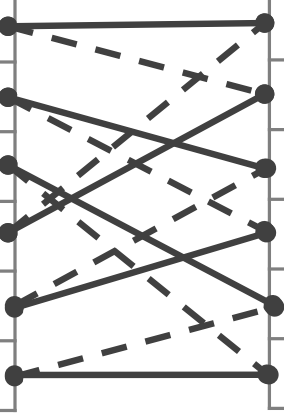
Formal Definition

Definition 1 (Missing Key Problem): Given relations R, S, join condition J and expected join results U^*, no query over $R \bowtie_J S$ provides U^*, and there is no auxiliary relation which enables to join R and S to provide U^*. $\square$

R $\bowtie_{\{Car=Car\}}$ **S**

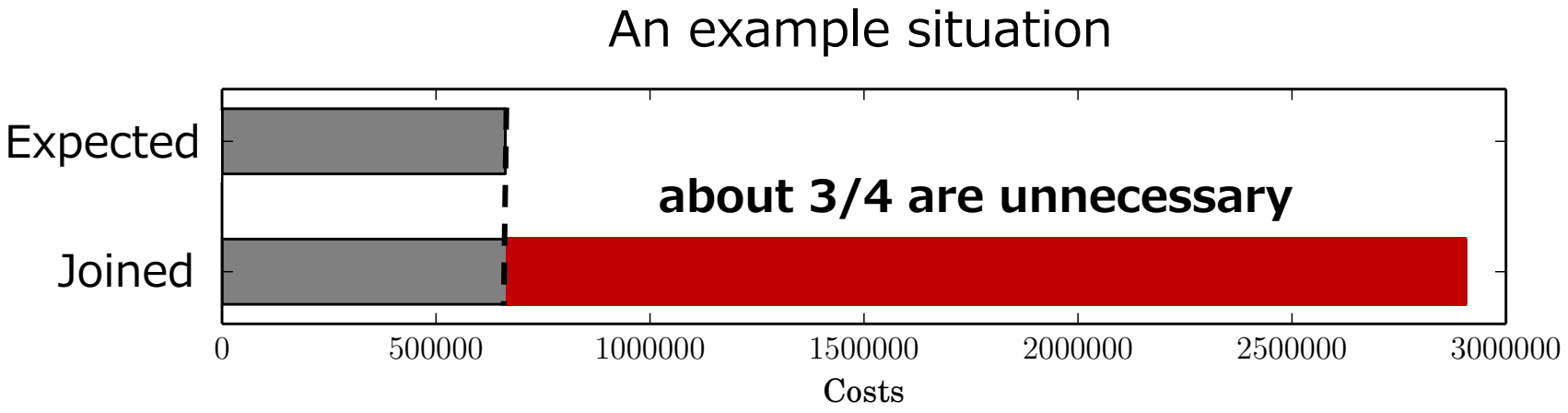
Car	Weight	Time
A	100	4/1/17 10:00
B	150	4/1/17 11:00
C	120	4/1/17 12:00
A	200	4/1/17 13:00
B	180	4/1/17 14:00
C	110	4/1/17 15:00

Car	Course	Area
A	A-1	D1
A	A-2	D2
B	B-1	D3
B	B-2	D1
C	C-1	D2
C	C-2	D3

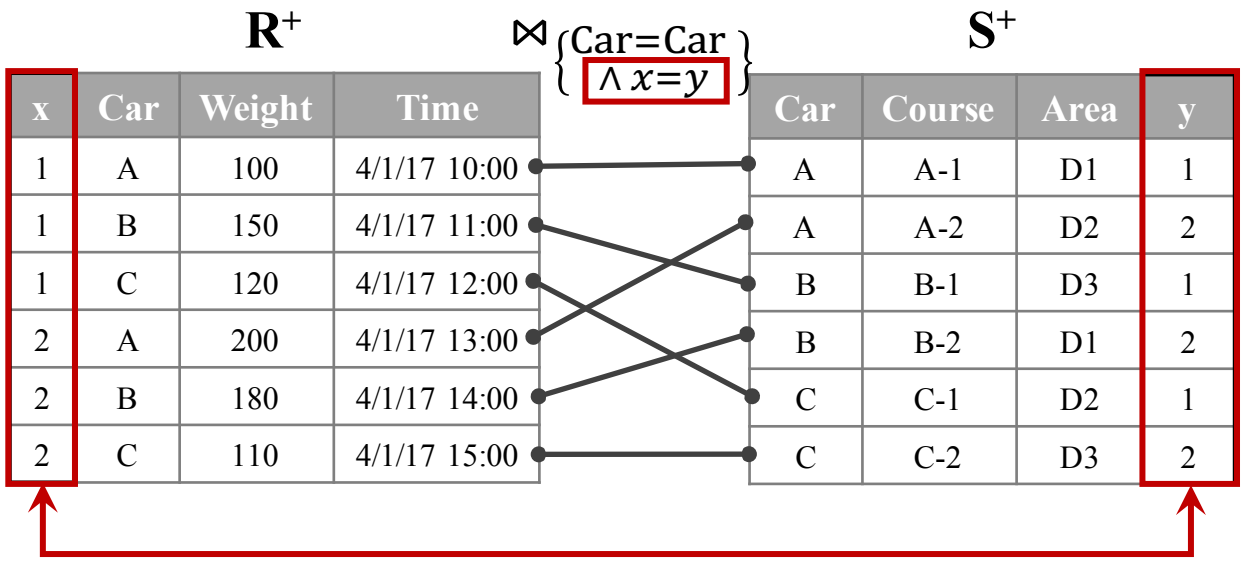


Trouble from Missing Key Prob.

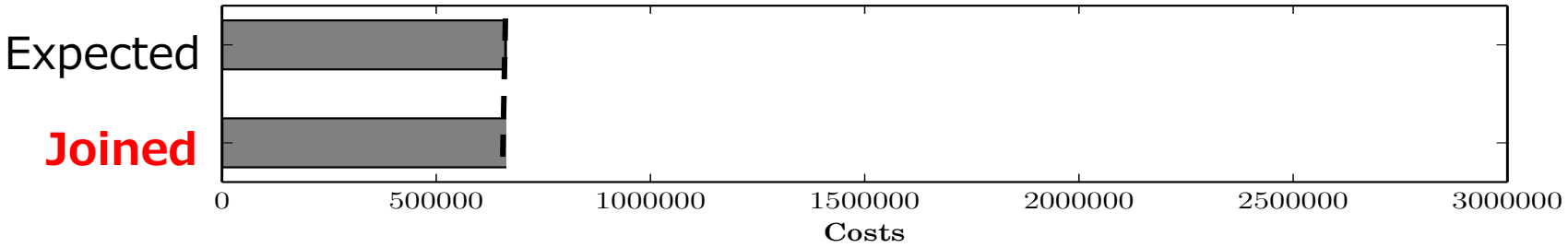
- Joined results include large number of unnecessary tuples.
- To use the results for applications, (automatic/manual) data cleansing is required.



Objective:
Implicit Key Discovery



Implicit keys



Observation: Order-oriented Correlation

- Assumed real-world situation: Joining log records with supplemental information
  - e.g., garbage collection logs and colleting routes of garbage cars

Garbage collection log

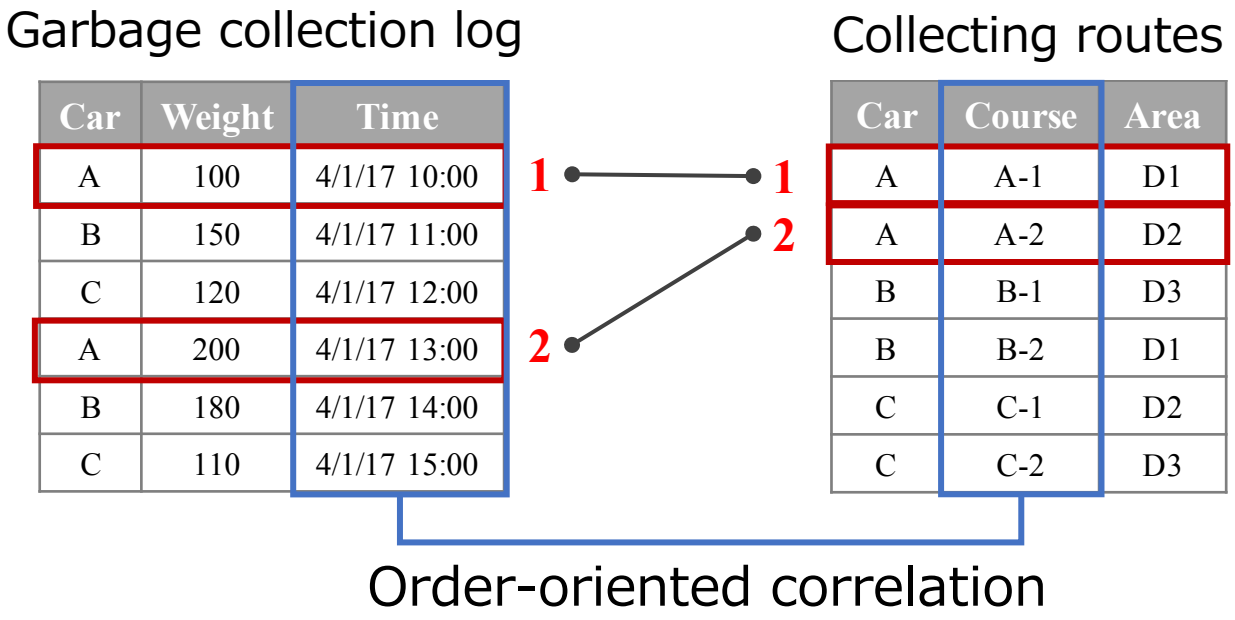
Car	Weight	Time
A	100	4/1/17 10:00
B	150	4/1/17 11:00
C	120	4/1/17 12:00
A	200	4/1/17 13:00
B	180	4/1/17 14:00
C	110	4/1/17 15:00

Collecting routes in a day

Car	Course	Area
A	A-1	D1
A	A-2	D2
B	B-1	D3
B	B-2	D1
C	C-1	D2
C	C-2	D3

Observation: Order-oriented Correlation

- Order-oriented correlation: an order of records in log data is corresponding with the of supplemental information.



Tackling Issue

- Discovery of attribute set pair with order-oriented correlation with help of human judged samples

$\widehat{U}^*$

Car	Weight	Time	Car	Course	Area
A	100	4/1/17 10:00	A	A-1	D1
B	150	4/1/17 11:00	A	A-2	D2
C	120	4/1/17 12:00	B	B-1	D3
A	200	4/1/17 13:00	B	B-2	D1
B	180	4/1/17 14:00	C	C-1	D2
C	110	4/1/17 15:00	C	C-2	D3

Human judged samples

R

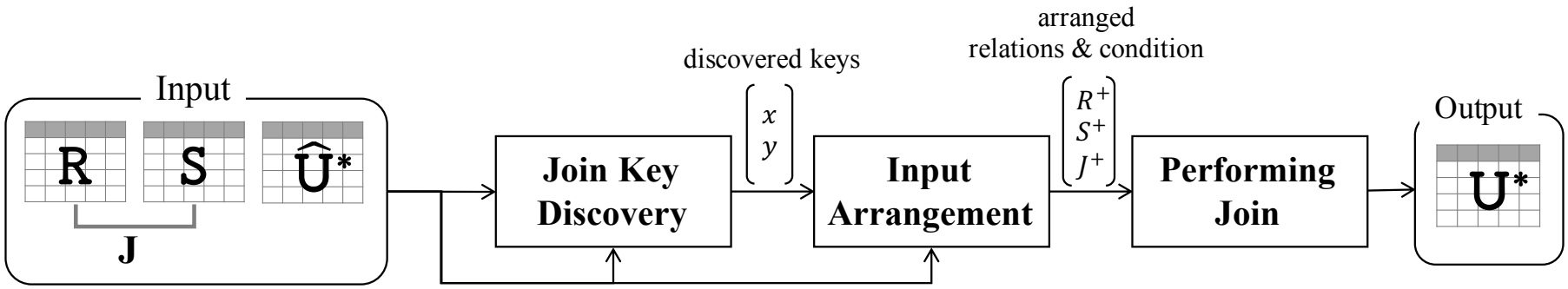
Car	Weight	Time
A	100	4/1/17 10:00
B	150	4/1/17 11:00
C	120	4/1/17 12:00
A	200	4/1/17 13:00
B	180	4/1/17 14:00
C	110	4/1/17 15:00

S

Car	Course	Area
A	A-1	D1
A	A-2	D2
B	B-1	D3
B	B-2	D1
C	C-1	D2
C	C-2	D3

Order-oriented correlation

Implicit Order Join Framework



1. Discover order-oriented attribute pair.
2. Generate complemental attributes.
3. Arrange relations and join conditions.
4. Perform join operation.

Combinatorial Problem

- Tremendous number of candidates of attribute set pairs.

$$\mathcal{O}(N_R!N_S!)$$

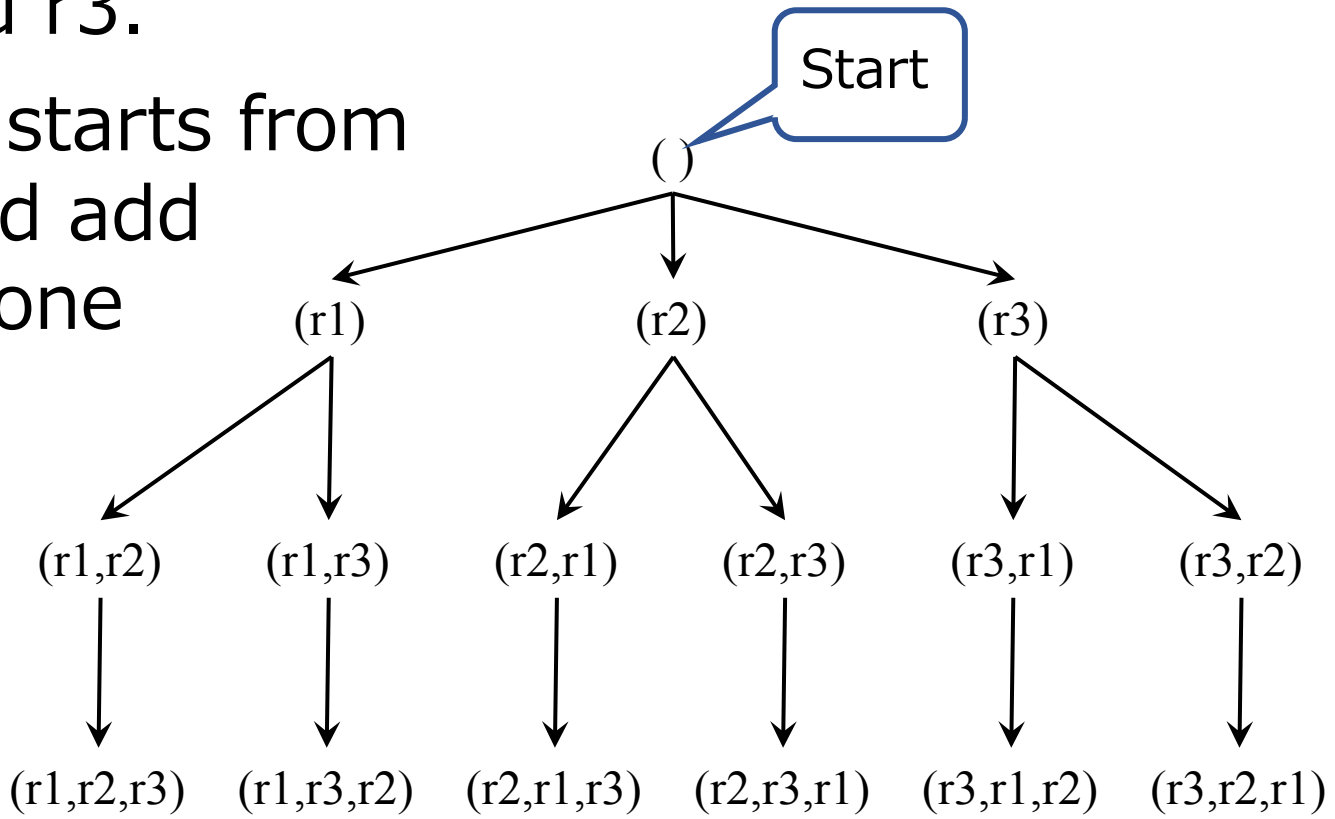
  - where N_R (or N_S) are the number of attributes of relation R (resp. S).
  - $N_X!$ is the number of enumerations of attributes in relation X.
  - Taking subsequences into account, the number of each enumeration becomes $\sum_{i=1}^{N} \binom{N}{i} i!$

Pruning of Candidates

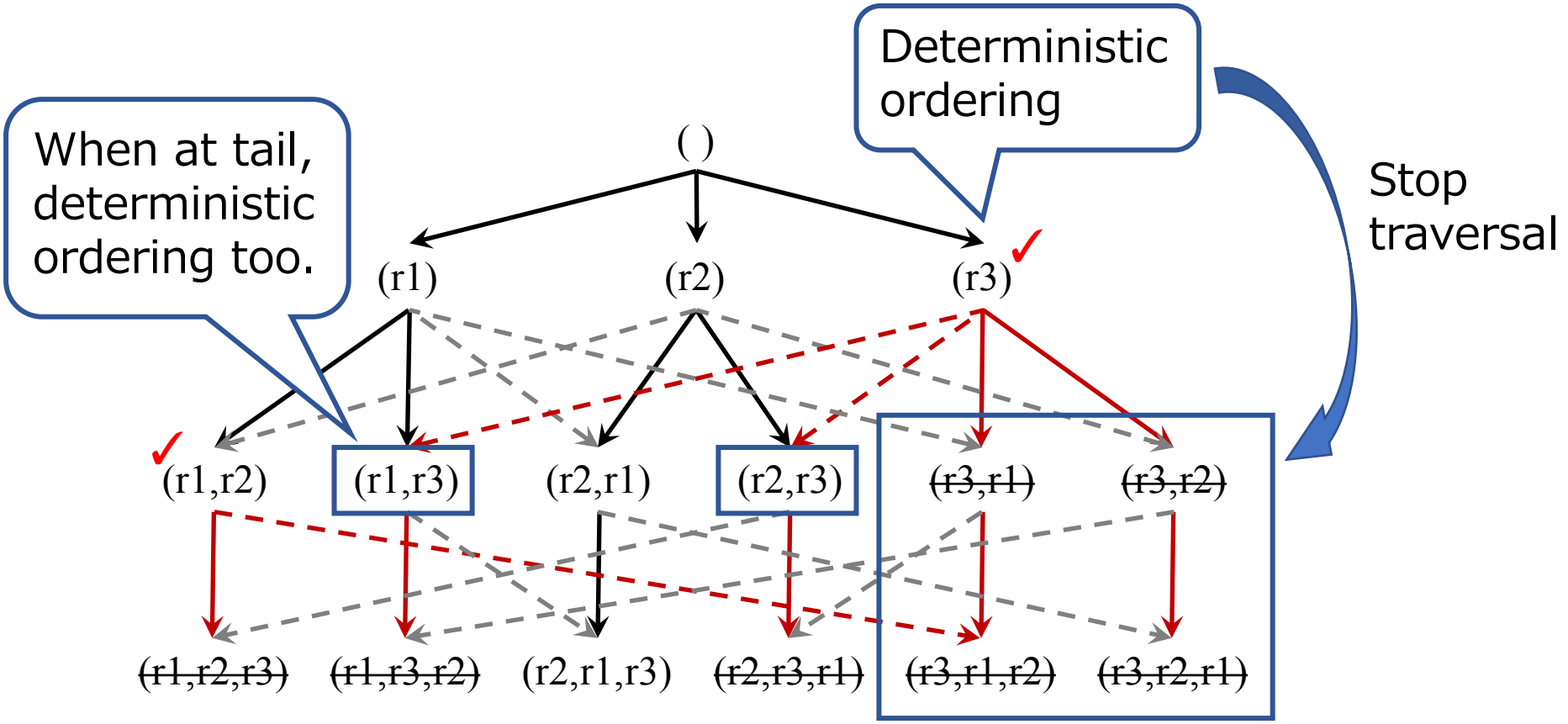
- Idea: a sequence of attribute gives deterministic ordering of records, super-sequences of it give the same ordering.
  - e.g., if (r1, r2) ➔ (d1, d2, d3),
    then (r1, r2, r3) ➔ (d1, d2, d3)
- Strategy
  - Bottom-up traversal
  - Stopping enumeration by the idea.

Bottom-up Traversal

- Relation R has three attributes r1, r2 and r3.
- Traversal starts from empty and add attribute one by one.



Pruning

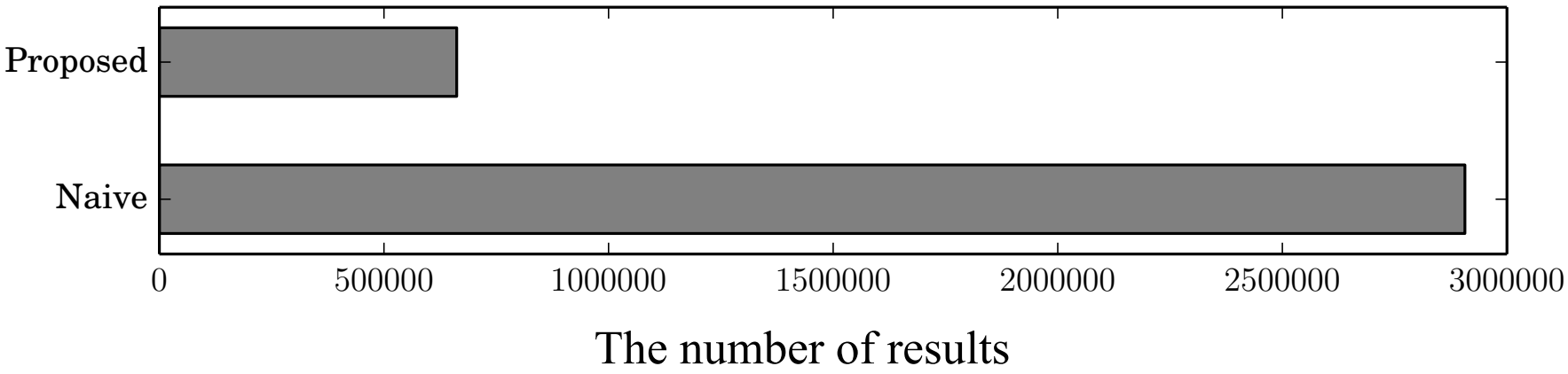

When at tail, deterministic ordering too.
Deterministic ordering
Stop traversal
( )
(r1)
(r2)
(r3)
(r1,r2)
(r1,r3)
(r2,r1)
(r2,r3)
(r3,r1)
(r3,r2)
(r1,r2,r3)
(r1,r3,r2)
(r2,r1,r3)
(r2,r3,r1)
(r3,r1,r2)
(r3,r2,r1)

Experimental Evaluation

- Objective
  1. Check effectiveness of the implicit order join.
  2. Check efficiency of the pruning.
- Datasets
  1. Real-world data from Fujisawa city, Japan.
     - Garbage collection logs and routing info.
  2. Synthetic data*
     - Tunable parameters
       - #attributes: total number of attributes
       - #oo-attributes: size of order-oriented attribute set

*https://github.com/Taka-Coma/OOJBench

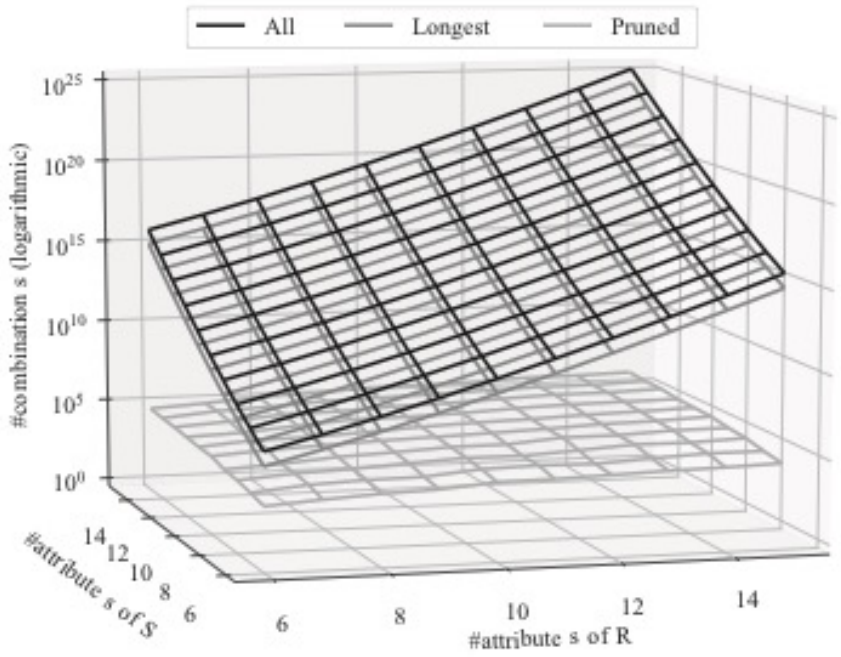
Implicit Order Join is Effective.



- 77% reduction of joined results.
- Carefully checked by human judges that the results are correct.

Efficiently prune for large #attrs.

Processing time in logarithmic scale

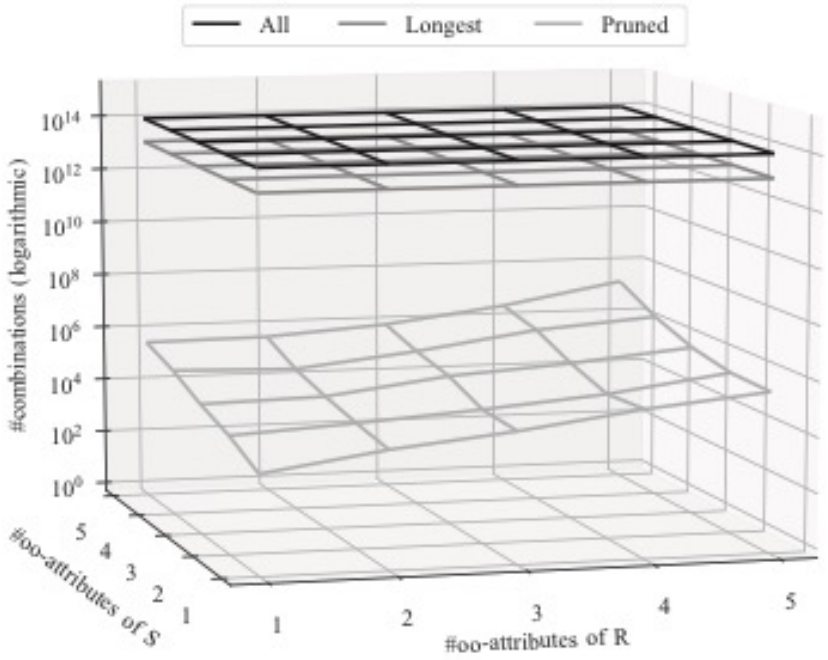


- The larger #attrs, the more #candidates in enumeration.
- Pruning effects big reduction of #candidates esp. when #attrs is large.

- Baselines
  - all: enumeration of subsequences of attributes
  - longest: enumeration of all attributes

#oo-attrs affects performance.

Processing time in logarithmic scale



- The larger #oo-attrs, the more processing time.
- Still far better than baselines.

- Baselines
  - all: enumeration of subsequences of attributes
  - longest: enumeration of all attributes

Conclusion and Future Work

- Conclusion
  - Definition: Missing key problem
  - Proposal: Implicit order join framework
    - Order-oriented correlation assumption
  - Experiment: Effectiveness and Efficiency
- Future Work
  - General approach for implicit join
    - Removal of the assumption