Tutorial:
Causality and Explanations in Databases

Alexandra Meliou

Sudeepa Roy

Dan Suciu

VLDB 2014
Hangzhou, China

We need to understand unexpected or interesting behavior of systems, experiments, or query answers to gain knowledge or troubleshoot

Unexpected results

```
select     distinct g.genre
from       Director d, Movie_Directors md,
           Movie m, Genre g
where      d.lastName like 'Burton'
           and g. mid=m.mid
           and m. mid=md.mid
           and md. did=d.did
order by   g.genre
```

genre
...
Fantasy
History
Horror
Music
Musical
Mystery
Romance
...

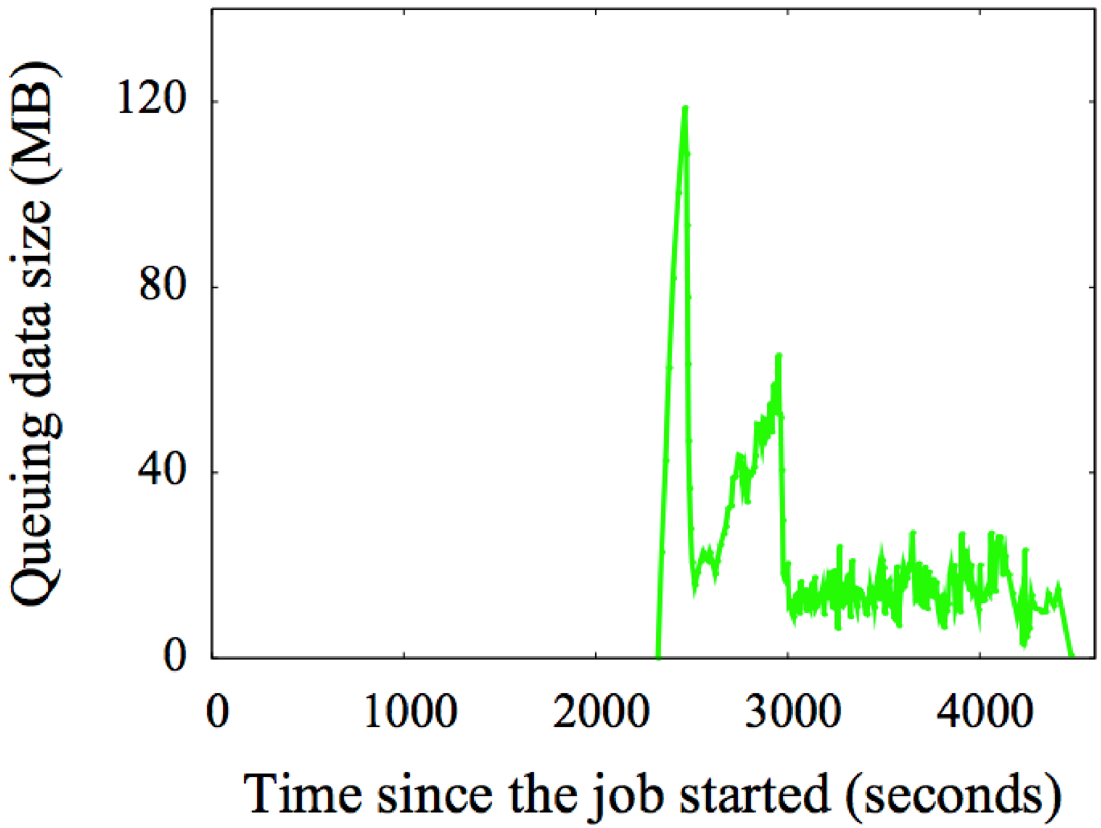
Unexpected results

```
select      distinct g.genre
from        Director d, Movie_Directors md,
            Movie m, Genre g
where       d.lastName like ‘Burton’
            and g. mid=m.mid
            and m. mid=md.mid
            and md. did=d.did
order by    g.genre
```

genre
. . .
Fantasy
History
Horror
Music
Musical
My ery
R nce

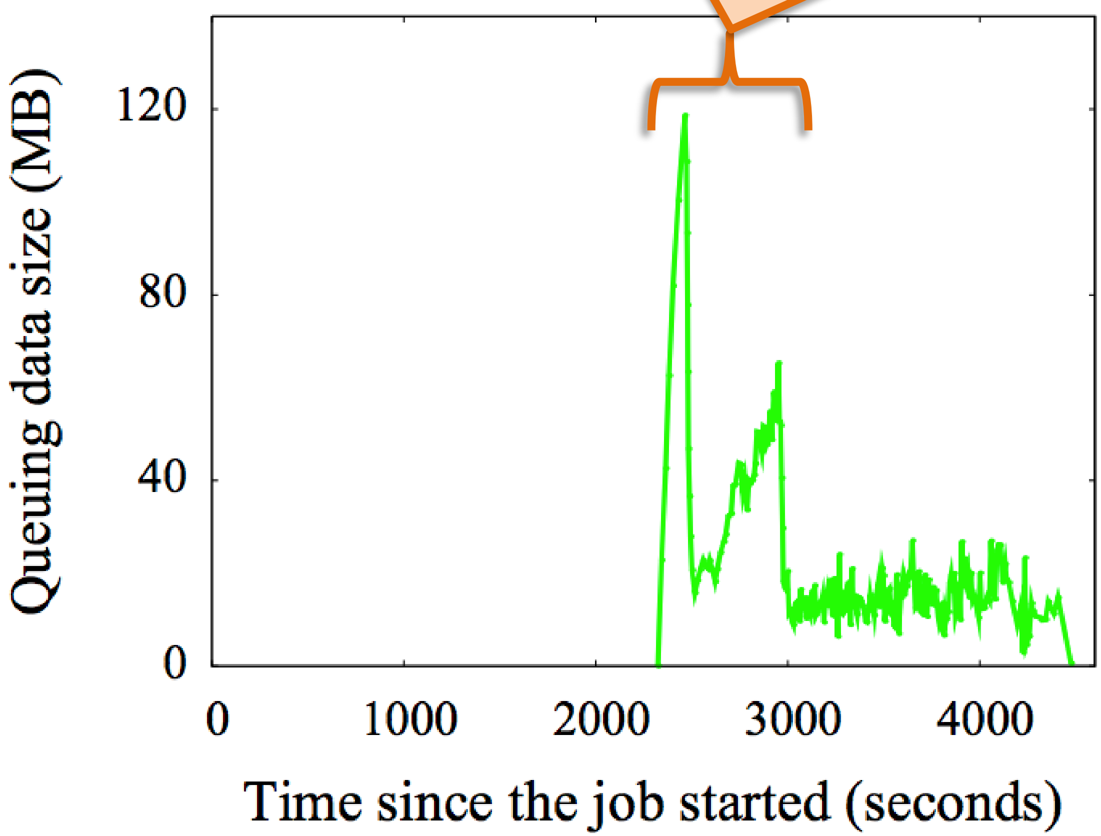
I didn’t know that Tim Burton directs Musicals!
Why are these items in the result of my query?

Inconsistent performance

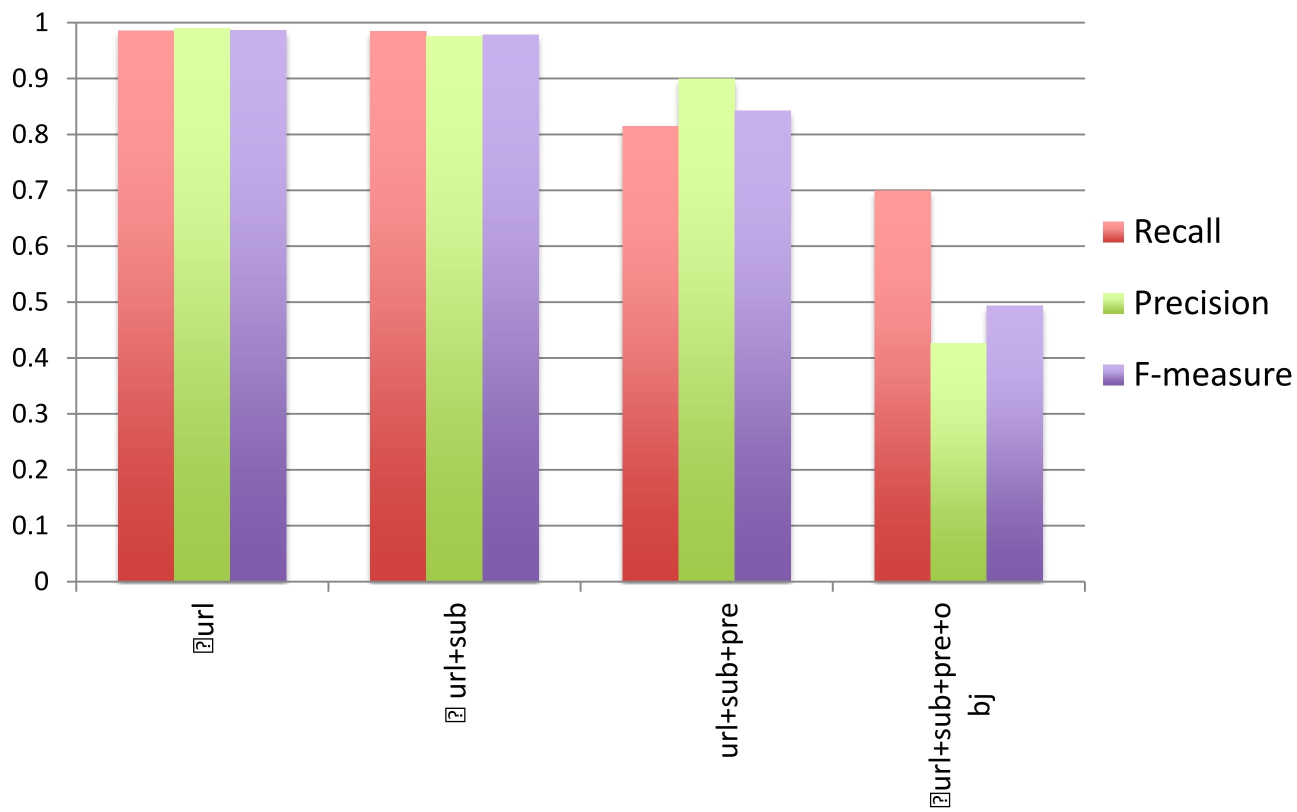


Inconsistent performance

Why is there such variability during this time interval?

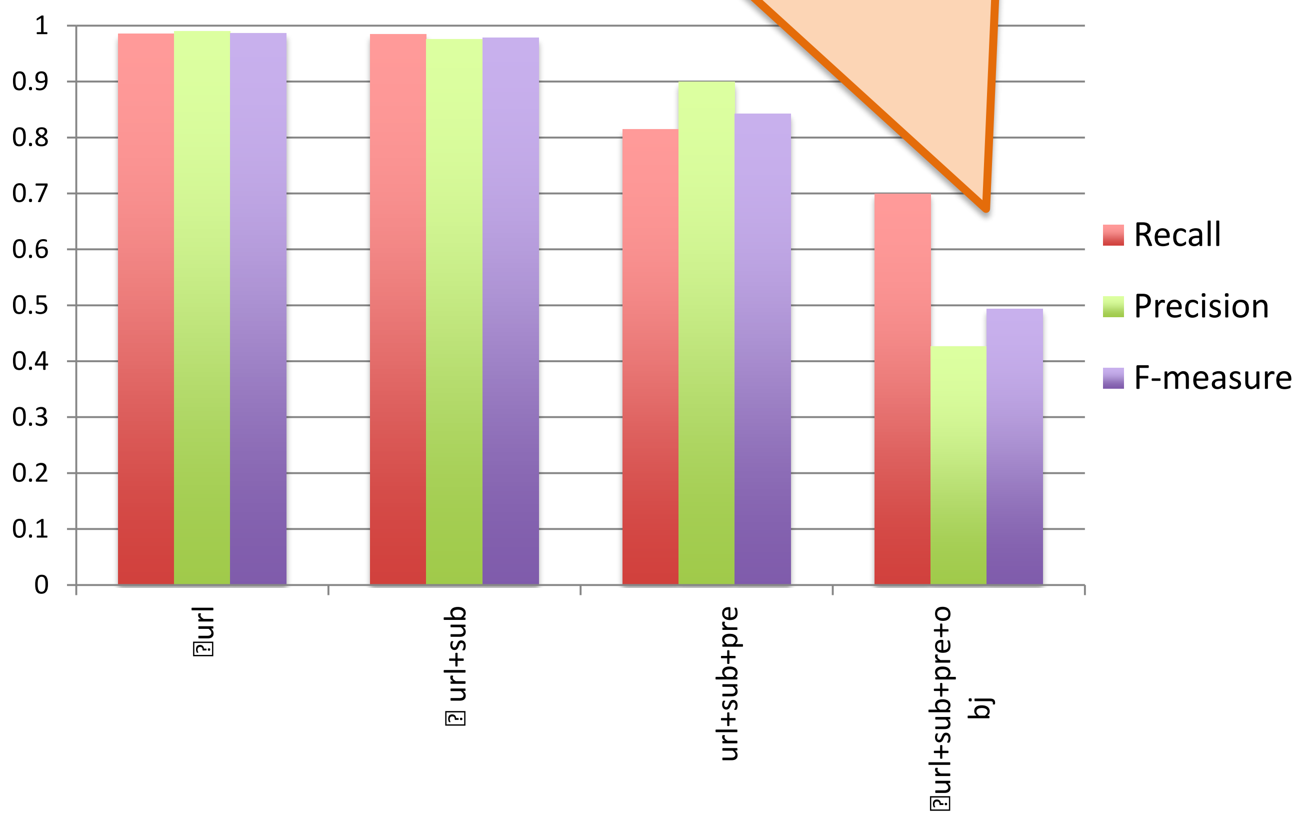


Understanding results



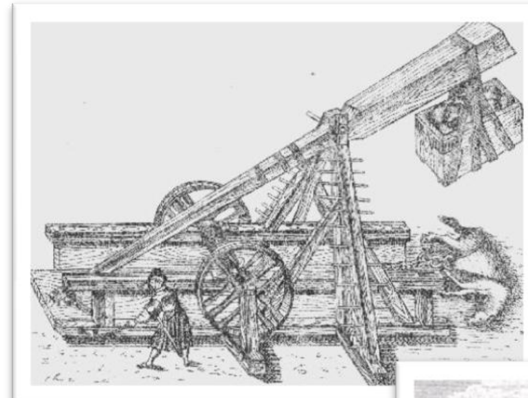
Understanding results

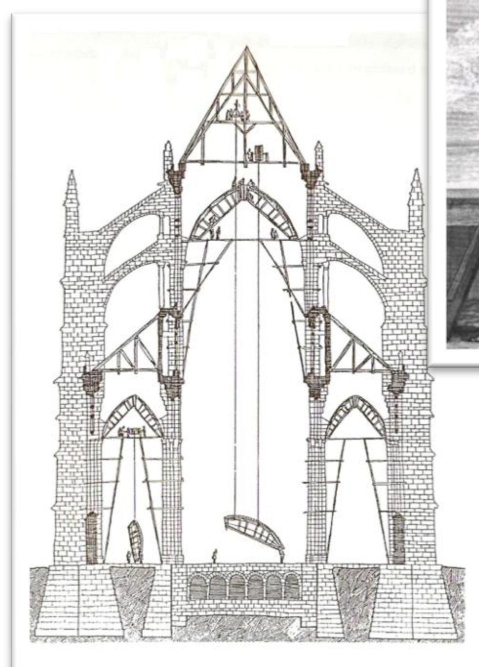
Why does the performance of my algorithm drop when I consider additional dimensions?

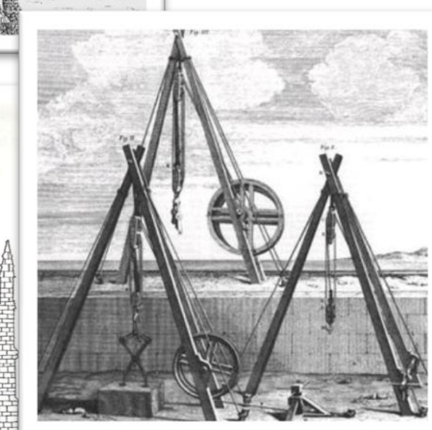


Causality in science

- Science seeks to understand and explain physical observations
  - *Why* doesn't the wheel turn?
  - *What if* I make the beam half as thick, will it carry the load?
  - *How* do I shape the beam so it will carry the load?



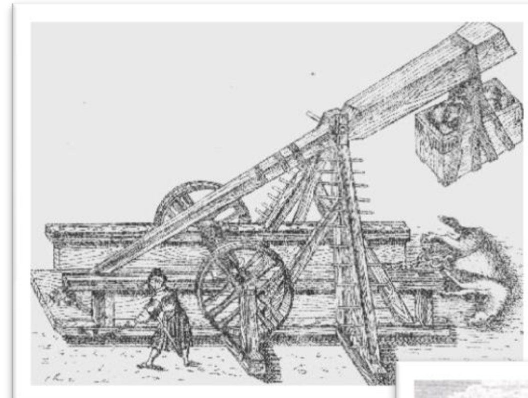


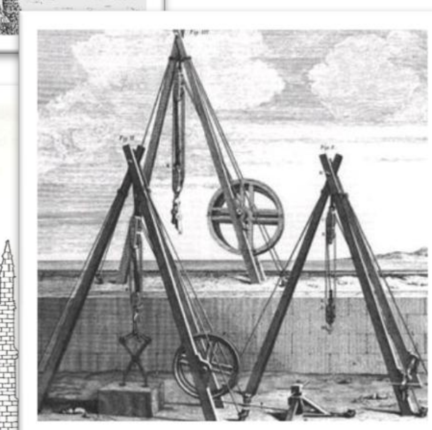


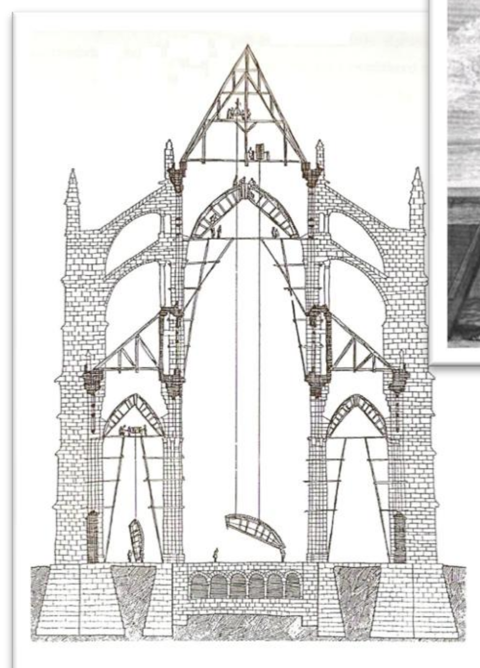
Causality in science

- Science seeks to understand and explain physical observations
  - *Why* doesn't the wheel turn?
  - *What if* I make the beam half as thick, will it carry the load?
  - *How* do I shape the beam so it will carry the load?

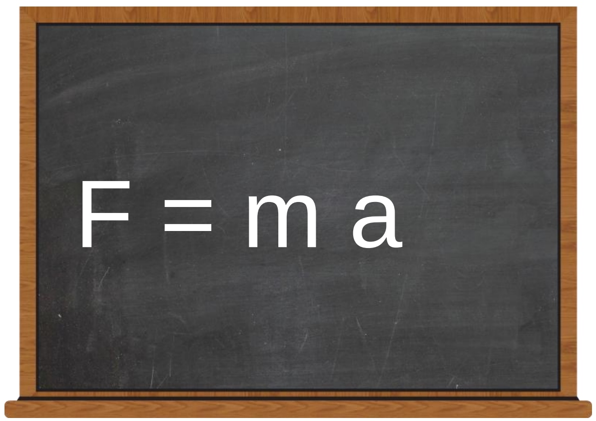
- We now have similar questions in databases!

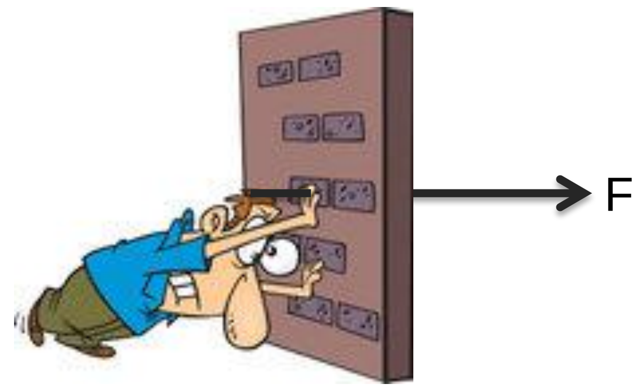






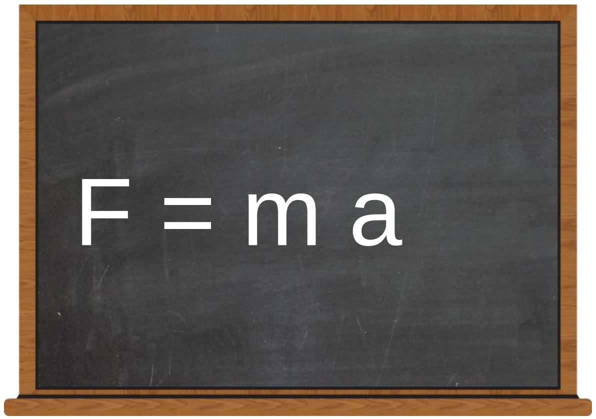
What is causality?

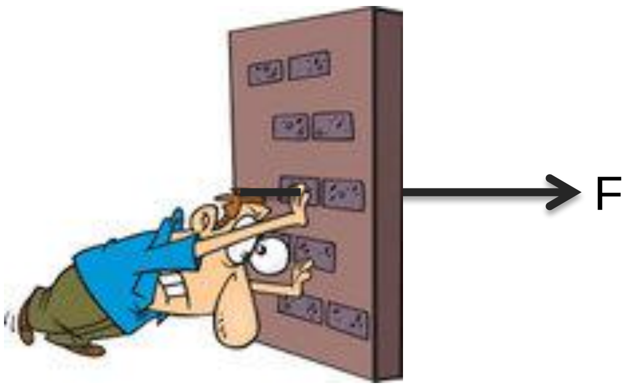




- Does acceleration cause the force?
- Does the force cause the acceleration?
- Does the force cause the mass?

What is causality?





- Does acceleration cause the force?
- Does the force cause the acceleration?
- Does the force cause the mass?

We cannot derive causality from data, yet we have developed a perception of what constitutes a cause.

Some history

David Hume (1711-1776)

Causation is a matter of perception

We remember seeing the <u>flame</u>, and feeling a sensation called <u>heat</u>; without further ceremony, we call the one <u>cause</u> and the other <u>effect</u>

Some history

David Hume (1711-1776)

Causation is a matter of perception

We remember seeing the flame, and feeling a sensation called heat; without further ceremony, we call the one cause and the other effect

Statistical ML

Forget causation! Correlation is all you should ask for.

Karl Pearson (1857-1936)

Some history

David Hume (1711-1776)

Causation is a matter of perception

We remember seeing the flame, and feeling a sensation called heat; without further ceremony, we call the one cause and the other effect

Statistical ML

Forget causation! Correlation is all you should ask for.

Karl Pearson (1857-1936)

Judea Pearl (1936-)

A mathematical definition of causality

Forget empirical observations! Define causality based on a network of known, physical, causal relationships

Tutorial overview

Part 1: Causality

- Basic definitions
- Causality in AI
- Causality in DB

Part 2: Explanations

- Explanations for DB query answers
- Application-specific approaches

Part 3: Related topics and Future directions

- Connections to lineage/provenance, deletion propagation, and missing answers
- Future directions

Part 1: Causality

a. Basic Definitions
b. Causality in AI
c. Causality in DB

Part 1.a

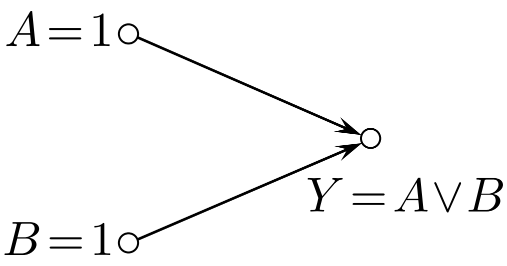
- **BASIC DEFINITIONS**

Basic definitions: overview

- Modeling causality
  - Causal networks
- Reasoning about causality
  - Counterfactual causes
  - Actual causes (Halpern & Pearl)
- Measuring causality
  - Responsibility

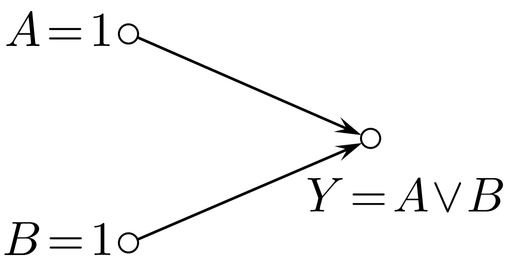
Causal networks

- Causal structural models:
  - Variables: A, B, Y
  - Structural equations: Y = A v B



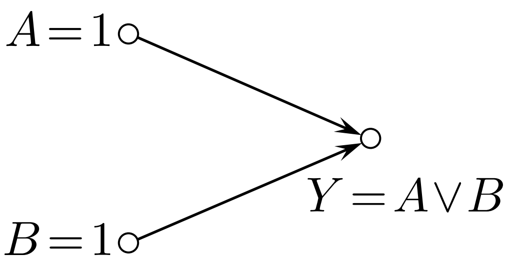
Causal networks

- Causal structural models:
  - Variables: A, B, Y
  - Structural equations: Y = A v B



- Modeling problems:
  - *E.g., A bottle breaks if either Alice or Bob throw a rock at it.*

Causal networks

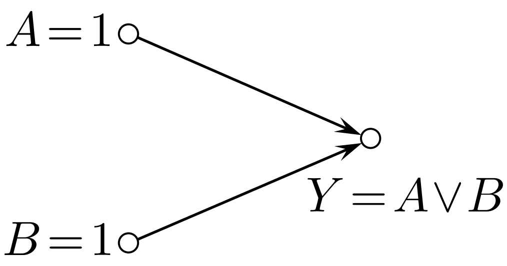


- Causal structural models:
  - Variables: A, B, Y
  - Structural equations: Y = A v B
- Modeling problems:
  - *E.g., A bottle breaks if either Alice or Bob throw a rock at it.*
  - Endogenous variables:
    - Alice throws a rock (A)
    - Bob throws a rock (B)
    - The bottle breaks (Y)

[Pearl, 2000]

Causal networks

- Causal structural models:
  - Variables: A, B, Y
  - Structural equations: Y = A v B

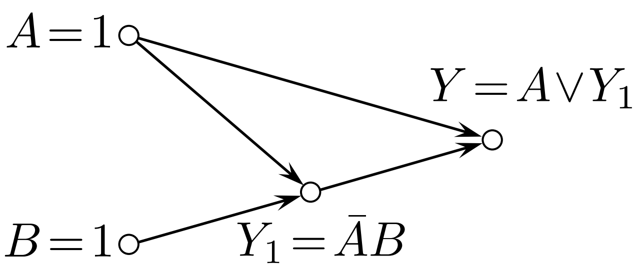


- Modeling problems:
  - *E.g., A bottle breaks if either Alice or Bob throw a rock at it.*
  - Endogenous variables:
    - Alice throws a rock (A)
    - Bob throws a rock (B)
    - The bottle breaks (Y)
  - Exogenous variables:
    - Alice’s aim, speed of the wind, bottle material etc.

[Woodward, 2003] [Hagmeyer, 2007]

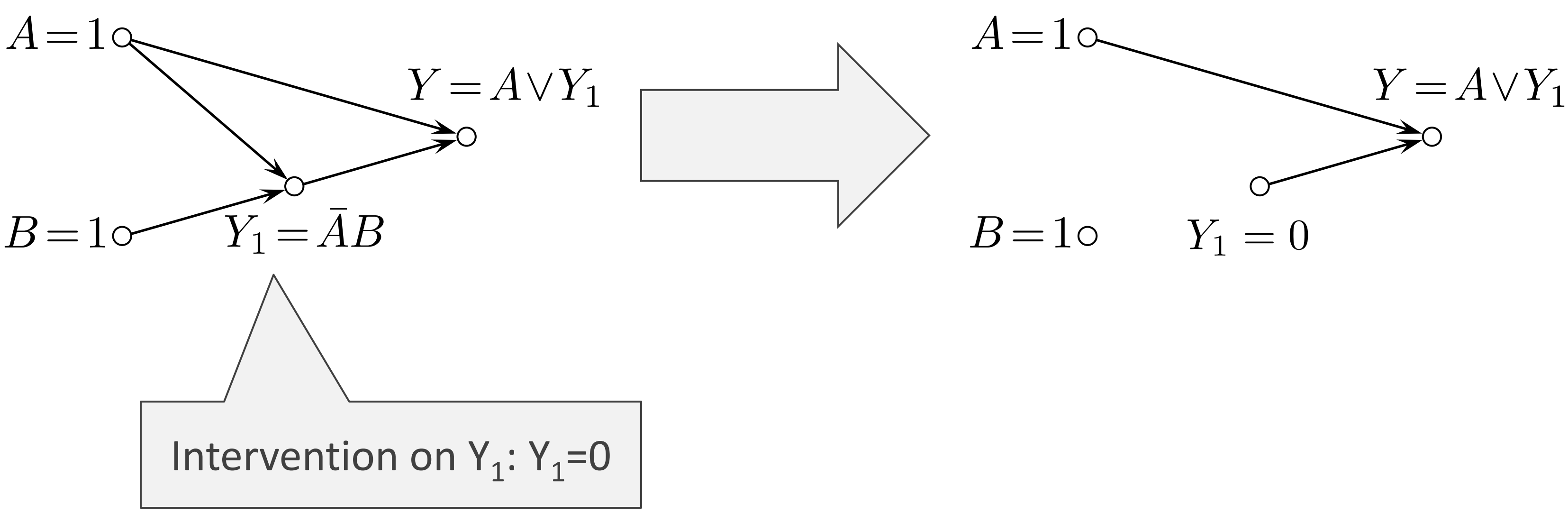
Intervention / contingency

- External interventions modify the structural equations or values of the variables.



Intervention / contingency

- External interventions modify the structural equations or values of the variables.



[Hume, 1748] [Menzies, 2008] [Lewis, 1973]

Counterfactuals

- If *not A* then *not φ*
  - In the absence of a cause, the effect doesn't occur

$$C = A \wedge B, \quad A = 1 \wedge B = 1 \longleftarrow \text{Both counterfactual}$$

[Hume, 1748] [Menzies, 2008] [Lewis, 1973]

Counterfactuals

- If *not A* then *not φ*
  - In the absence of a cause, the effect doesn't occur

$$C = A \wedge B, \quad A = 1 \wedge B = 1 \longleftarrow \text{Both counterfactual}$$

- Problem: Disjunctive causes
  - If Alice doesn't throw a rock, the bottle still breaks (because of Bob)
  - Neither Alice nor Bob are counterfactual causes

$$C = A \vee B, \quad A = 1 \wedge B = 1$$

[Hume, 1748] [Menzies, 2008] [Lewis, 1973]

Counterfactuals

- If *not A* then *not φ*
  - In the absence of a cause, the effect doesn't occur

$$C = A \land B, \quad A = 1 \land B = 1 \longleftarrow \text{Both counterfactual}$$

- Problem: Disjunctive causes
  - If Alice doesn't throw a rock, the bottle still breaks (because of Bob)
  - Neither Alice nor Bob are counterfactual causes

$$C = A \lor B, \quad A = 1 \land B = 1 \longleftarrow \text{No counterfactual causes}$$

[Halpern-Pearl, 2001] [Halpern-Pearl, 2005]

Actual causes

[simplification]

A variable X is an <u>actual cause</u> of an effect Y if there exists a contingency that makes X counterfactual for Y.

$$C = A \vee B, \quad A = 1 \wedge B = 1 \longleftarrow$$ A is a cause under the contingency B=0

Example 1

$$Y = X_1 \wedge X_2 \qquad X_1 = X_2 = 1 \Rightarrow Y = 1$$

X_1=1 is counterfactual for Y=1

Example 1

$$Y = X_1 \wedge X_2 \qquad X_1 = X_2 = 1 \Rightarrow Y = 1$$

X_1=1 is counterfactual for Y=1

Example 2

$$Y = X_1 \vee X_2 \qquad X_1 = X_2 = 1 \Rightarrow Y = 1$$

X_1=1 is not counterfactual for Y=1

X_1=1 is an <u>actual</u> cause for Y=1, with contingency X_2=0

Example 1

$$Y = X_1 \wedge X_2 \qquad X_1 = X_2 = 1 \Rightarrow Y = 1$$

X_1=1 is counterfactual for Y=1

Example 2

$$Y = X_1 \vee X_2 \qquad X_1 = X_2 = 1 \Rightarrow Y = 1$$

X_1=1 is not counterfactual for Y=1

X_1=1 is an <u>actual</u> cause for Y=1, with contingency X_2=0

Example 3

$$Y = (\neg X_1 \wedge X_2) \vee X_3 \qquad X_1 = X_2 = X_3 = 1 \Rightarrow Y = 1$$

X_1=1 is not counterfactual for Y=1

X_1=1 is not an actual cause for Y=1

[Chockler-Halpern, 2004]

Responsibility

A measure of the degree of causality

$$\rho = \frac{1}{1 + \min_{\Gamma} |\Gamma|}$$

size of the contingency set

[Chockler-Halpern, 2004]

Responsibility

A measure of the degree of causality

$$\rho = \frac{1}{1 + \min_{\Gamma} |\Gamma|}$$

size of the contingency set

Example

$Y = A \wedge (B \vee C)$ $\quad A = B = C = 1 \Rightarrow Y = 1$

A=1 is counterfactual for Y=1 (ρ=1)

B=1 is an actual cause for Y=1, with contingency C=0 (ρ=0.5)

Basic definitions: summary

- Causal networks model the known variables and causal relationships
- Counterfactual causes have direct effect to an outcome
- Actual causes extend counterfactual causes and express causal influence in more settings
- Responsibility measures the contribution of a cause to an outcome

Part 1.b

• CAUSALITY IN AI

Causality in AI: overview

- Actual causes: going deeper into the Halpern-Pearl definition
- Complications of actual causality and solutions
- Complexity of inferring actual causes

Dealing with complex settings

- The definition of actual causes was designed to capture complex scenarios

Permissible contingencies

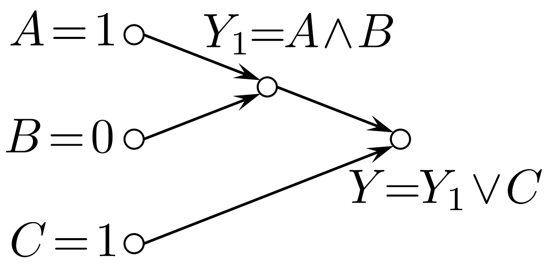
Not all contingencies are valid => Restrictions in the Halpern-Pearl definition of actual causes.

Preemption

Model priorities of events => one event may *preempt* another

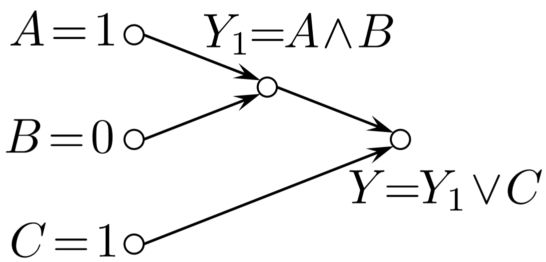
[Halpern-Pearl, 2001] [Halpern-Pearl, 2005]

Permissible contingencies



A: Alice loads Bob's gun
B: Bob shoots
C: Charlie loads and shoots his own gun
Y: the prisoner dies

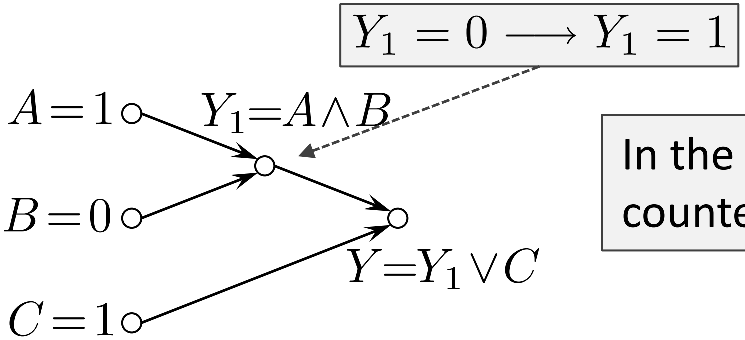
Permissible contingencies



In the contingency {A=1,B=1,C=0}, A is counterfactual, but should it be a cause?

A: Alice loads Bob's gun
B: Bob shoots
C: Charlie loads and shoots his own gun
Y: the prisoner dies

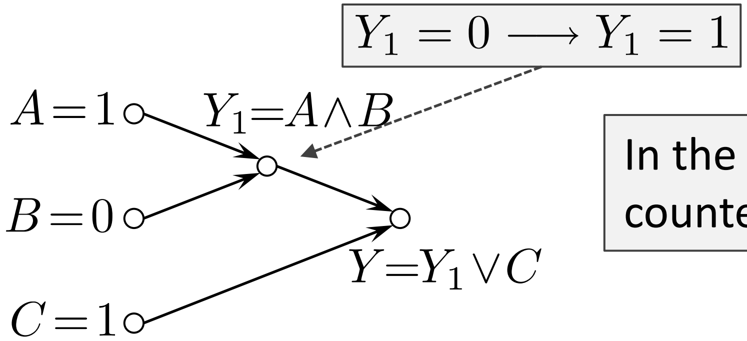
Permissible contingencies



In the contingency {A=1,B=1,C=0}, A is counterfactual, but should it be a cause?

A: Alice loads Bob’s gun
B: Bob shoots
C: Charlie loads and shoots his own gun
Y: the prisoner dies

Permissible contingencies



In the contingency {A=1,B=1,C=0}, A is counterfactual, but should it be a cause?

A: Alice loads Bob's gun

B: Bob shoots

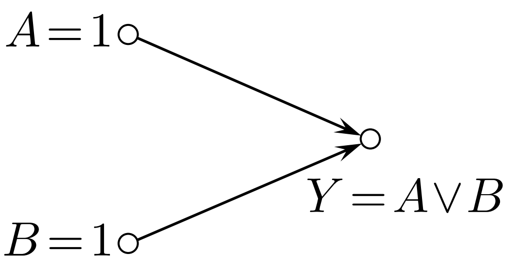
C: Charlie loads and shoots his own gun

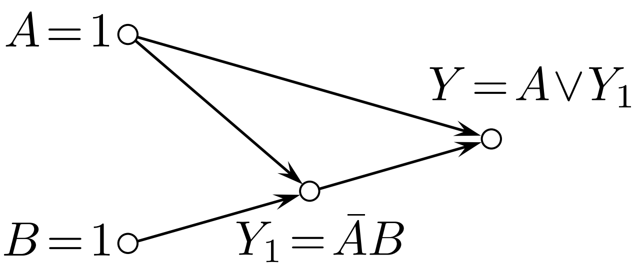
Y: the prisoner dies

Additional restriction in the HP definition:
Nodes in the causal path should not change value.

[Schaffer, 2000] [Halpern-Pearl, 2001] [Halpern-Pearl, 2005]

Causal priority: preemption





A: Alice throws a rock

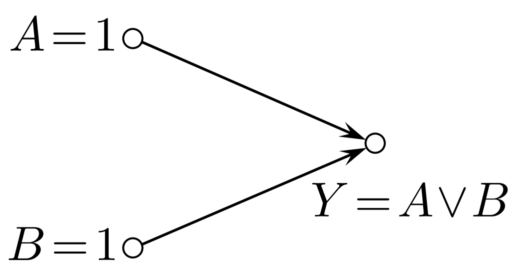
B: Bob throws a rock

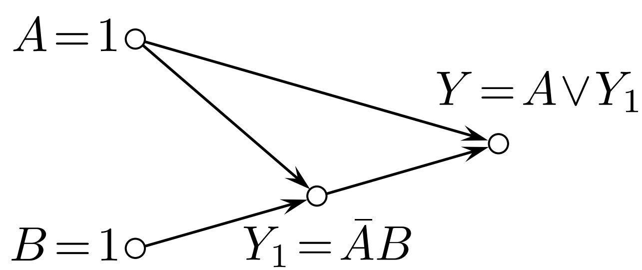
Y: the bottle breaks

[Schaffer, 2000] [Halpern-Pearl, 2001] [Halpern-Pearl, 2005]

Causal priority: preemption

$$\boxed{A \vee B = A \vee \bar{A}B}$$





A: Alice throws a rock

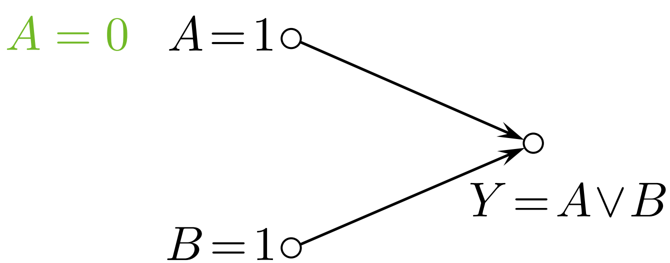
B: Bob throws a rock

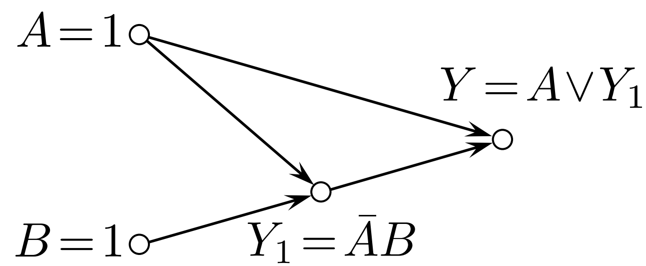
Y: the bottle breaks

[Schaffer, 2000] [Halpern-Pearl, 2001] [Halpern-Pearl, 2005]

Causal priority: preemption

$$A \lor B = A \lor \bar{A}B$$

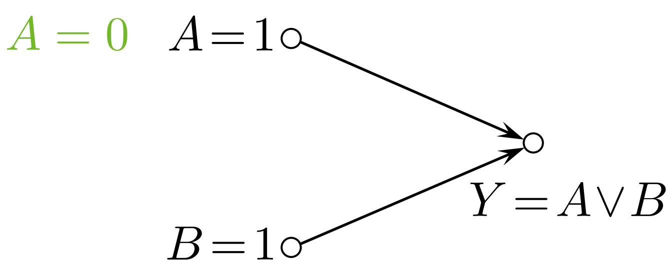


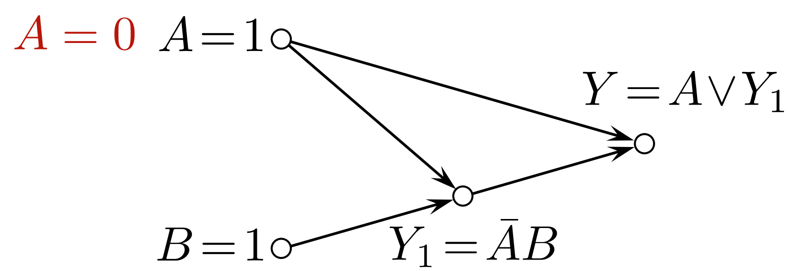


A: Alice throws a rock
B: Bob throws a rock
Y: the bottle breaks

Causal priority: preemption

$$A \vee B = A \vee \bar{A}B$$

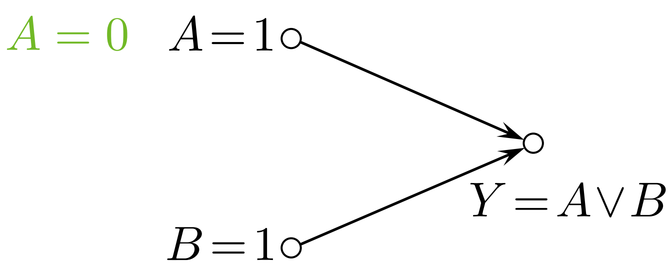


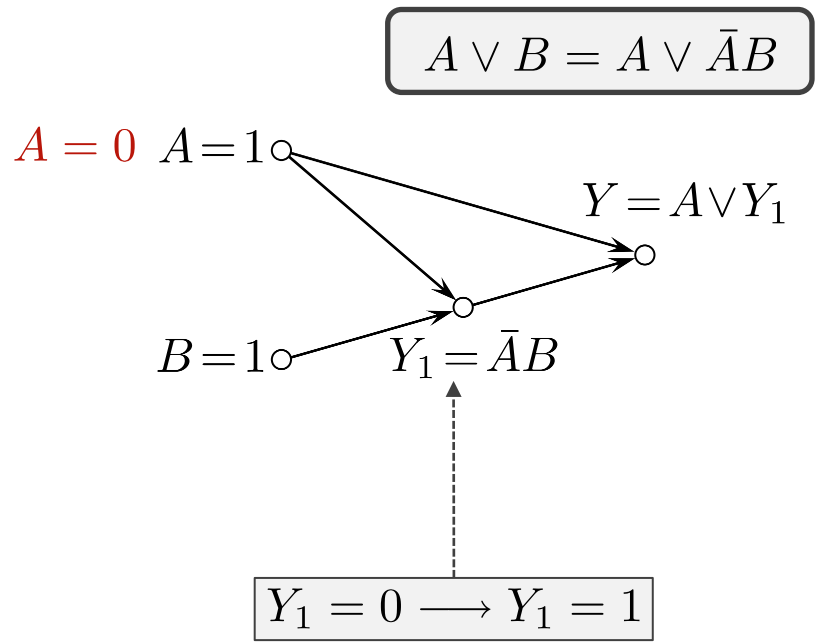


A: Alice throws a rock
B: Bob throws a rock
Y: the bottle breaks

[Schaffer, 2000] [Halpern-Pearl, 2001] [Halpern-Pearl, 2005]

Causal priority: preemption





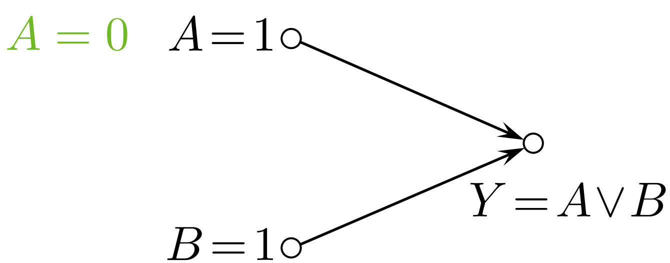
A: Alice throws a rock

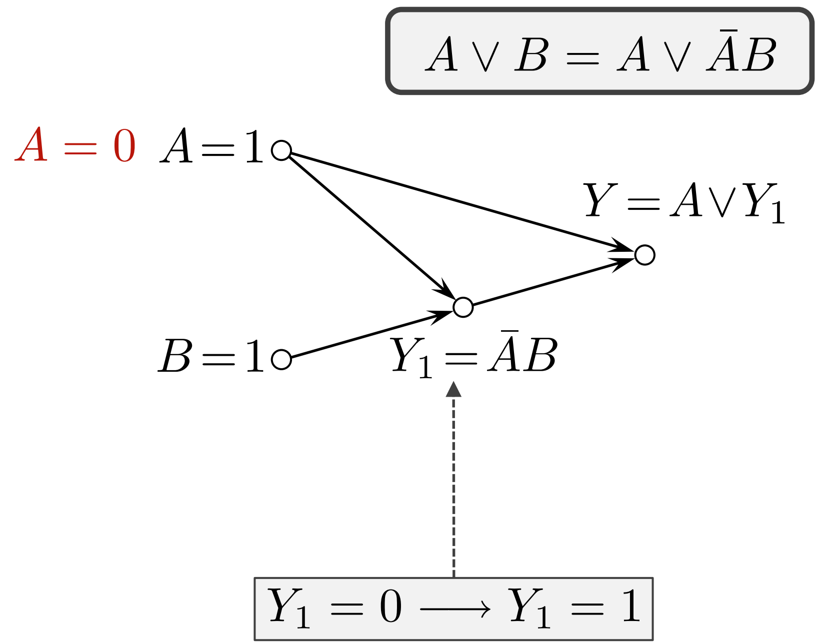
B: Bob throws a rock

Y: the bottle breaks

[Schaffer, 2000] [Halpern-Pearl, 2001] [Halpern-Pearl, 2005]

Causal priority: preemption





A: Alice throws a rock

B: Bob throws a rock

Y: the bottle breaks

Even though the structural equations for Y are equivalent, the two causal networks result in different interpretations of causality

[Meliou et al., 2010a]

Complications

- Intricacy
  - The definition has been used incorrectly in literature: [Chockler, 2008]

[Meliou et al., 2010a]

Complications

- Intricacy
  - The definition has been used incorrectly in literature: [Chockler, 2008]
- Dependency on graph structure and syntax

[Meliou et al., 2010a]

Complications

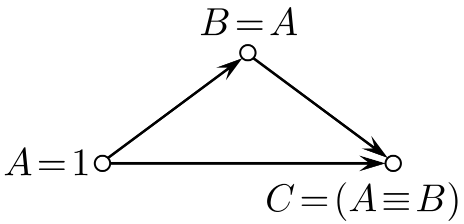
- Intricacy
  - The definition has been used incorrectly in literature: [Chockler, 2008]
- Dependency on graph structure and syntax
- Counterintuitive results

[Meliou et al., 2010a]

Complications

- Intricacy
  - The definition has been used incorrectly in literature: [Chockler, 2008]
- Dependency on graph structure and syntax
- Counterintuitive results

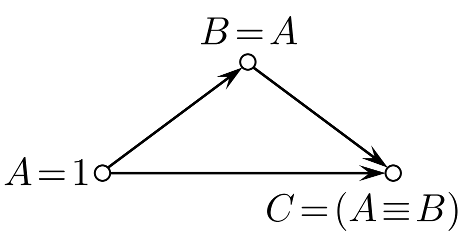
Shock C



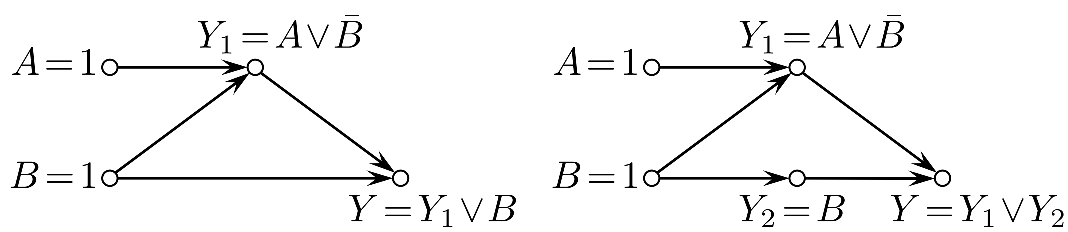
Complications

- Intricacy
  - The definition has been used incorrectly in literature: [Chockler, 2008]
- Dependency on graph structure and syntax
- Counterintuitive results

Shock C



Network expansion



Defaults and normality

- **World**: a set of values for all the variables
- **Rank:** each world has a rank; the higher the rank, the less likely the world
- **Normality:** can only pick contingencies of lower rank (more likely worlds)

[Halpern, 2008]

Defaults and normality

- **World**: a set of values for all the variables
- **Rank:** each world has a rank; the higher the rank, the less likely the world

- **Normality:** can only pick contingencies of lower rank (more likely worlds)

Addresses some of the complications, but requires ordering of possible worlds.

[Eiter- Lukasiewicz 2002]

Complexity of causality

Counterfactual cause	Actual cause
PTIME	NP-complete

Proof: Reduction from SAT.

Given F, F is satisfiable iff X is an actual cause for $X \wedge F$

[Eiter- Lukasiewicz 2002]

Complexity of causality

Counterfactual cause	Actual cause
PTIME	NP-complete

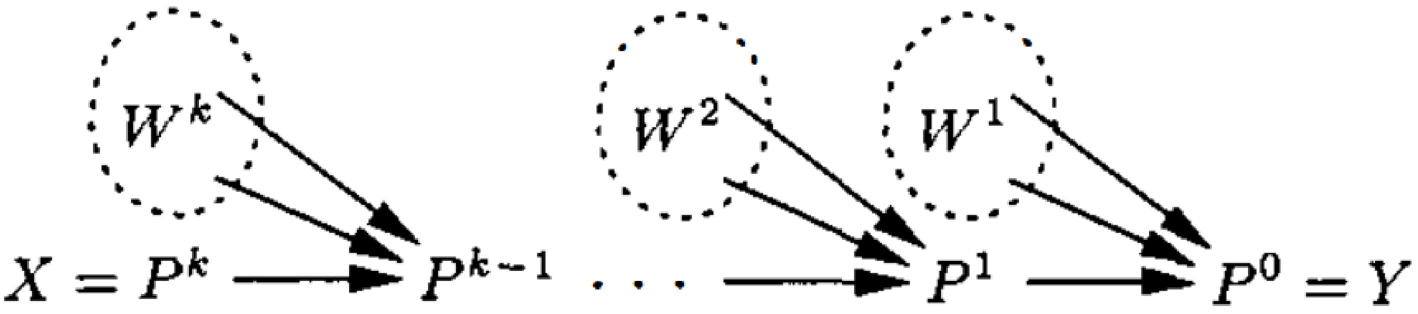
Proof: Reduction from SAT.
Given F, F is satisfiable iff X is an actual cause for $X \wedge F$

For non-binary models: Σ_2^P-complete

[Eiter- Lukasiewicz 2002]

Tractable cases

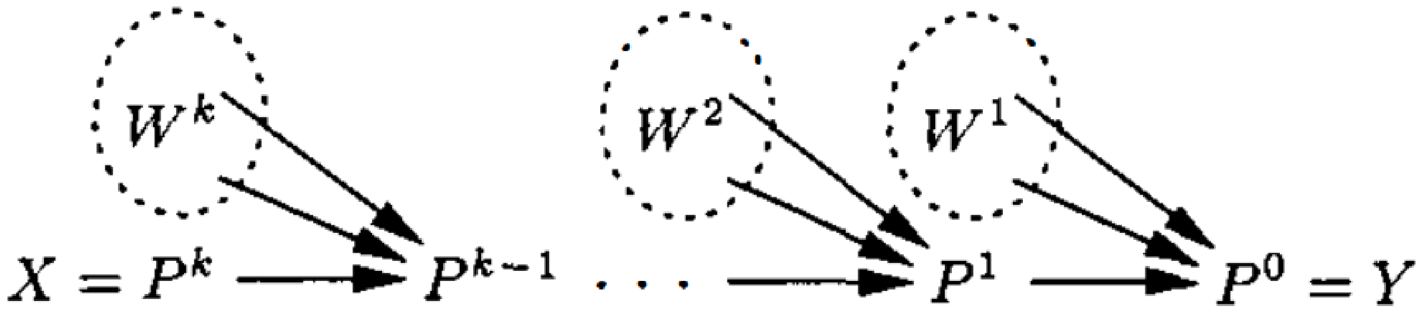
1. Causal trees



[Eiter- Lukasiewicz 2002]

Tractable cases

1. Causal trees

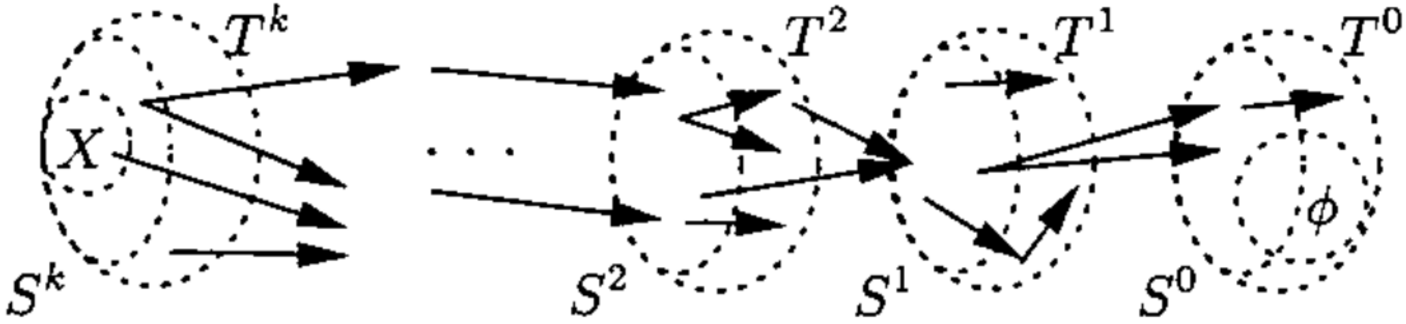


Actual causality can be determined in linear time

[Eiter- Lukasiewicz 2002]

Tractable cases

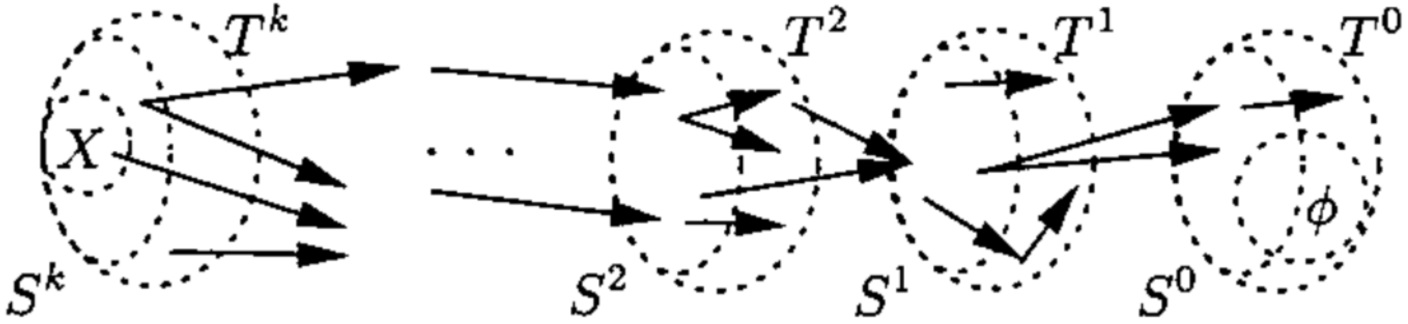
2. Width-bounded decomposable causal graphs



[Eiter- Lukasiewicz 2002]

Tractable cases

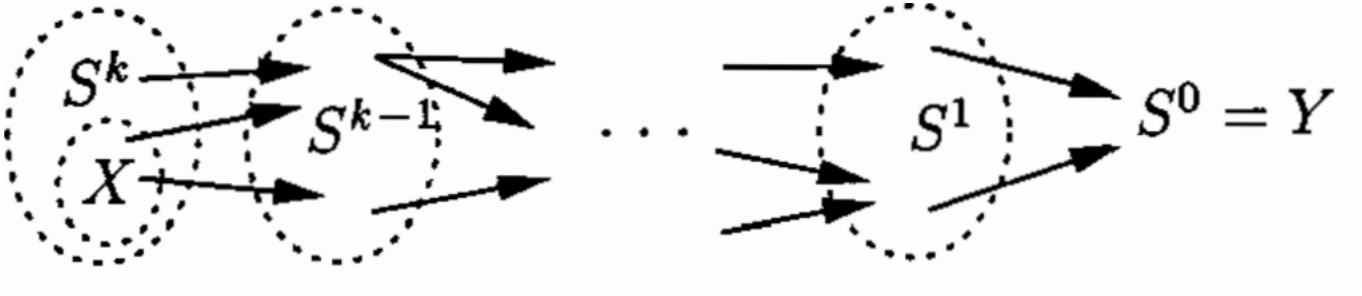
2. Width-bounded decomposable causal graphs



It is unclear whether decompositions can be efficiently computed

Tractable cases

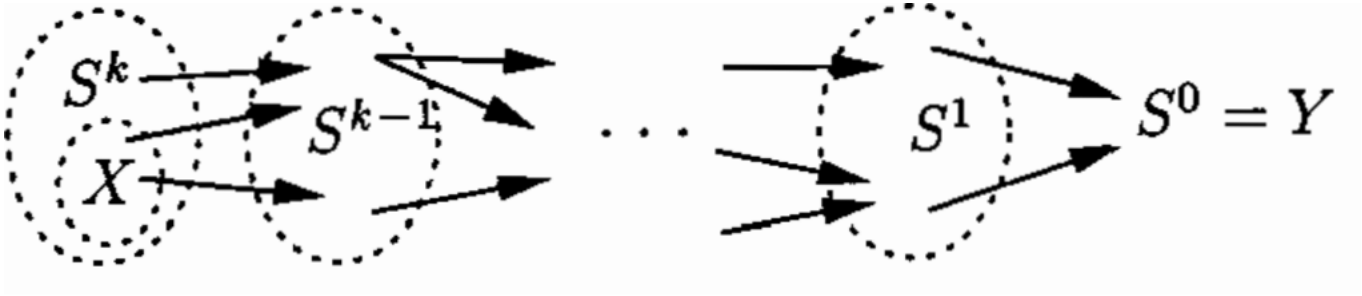
3. Layered causal graphs



[Eiter- Lukasiewicz 2002]

Tractable cases

3. Layered causal graphs



Layered graphs are decompositions that can be computed in linear time.

Causality in AI: summary

- Actual causes:
  - permissible contingencies and preemption
  - Weaknesses of the HP definition: normality

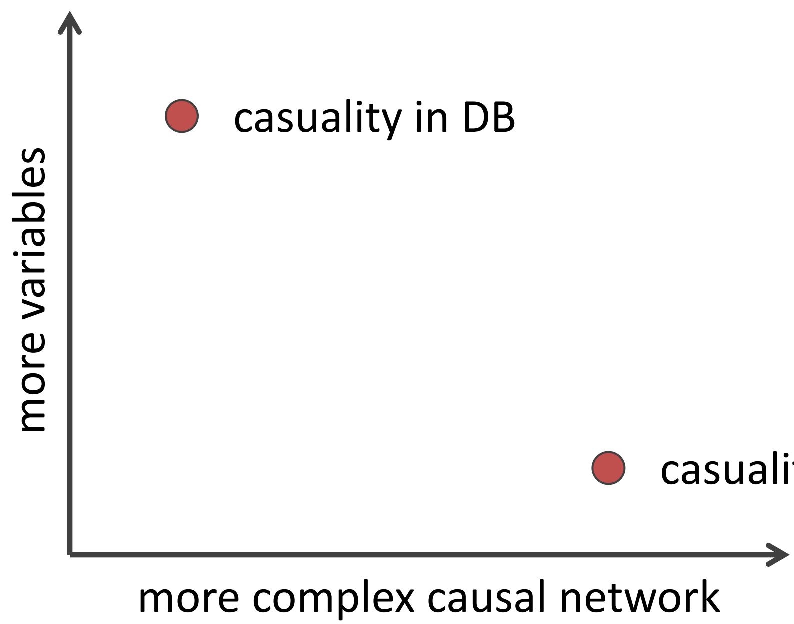
- Complexity:
  - Based on a given causal network
  - Tractable cases

Part 1.c

• CAUSALITY IN DATABASES

Causality in databases: overview

- What is the causal network, a cause, and responsibility in a DB setting?



Motivating example: IMDB dataset

IMDB Database Schema

Actor

<u>*aid*</u>	*firstName*	*lastName*

Director

<u>*did*</u>	*firstName*	*lastName*

Movie

<u>*mid*</u>	*name*	*year*	*rank*

Genre

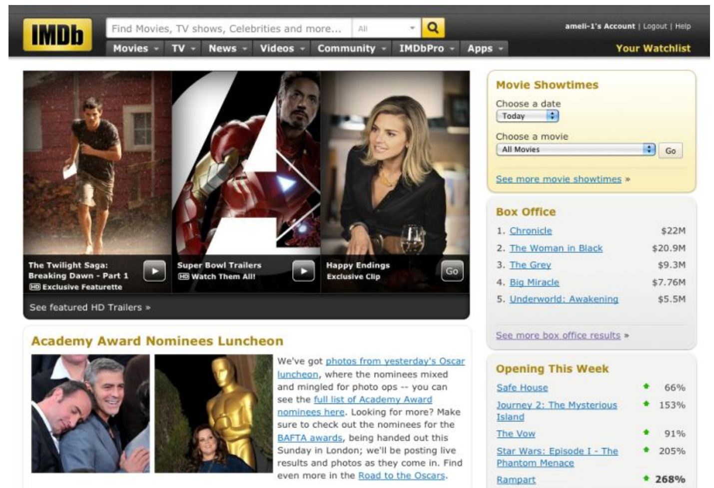
mid	*genre*

Movie_Directors

did	*mid*

Casts

aid	*mid*	*role*



Motivating example: IMDB dataset

IMDB Database Schema

Actor

<u>*aid*</u>	*firstName*	*lastName*

Director

<u>*did*</u>	*firstName*	*lastName*

Movie

<u>*mid*</u>	*name*	*year*	*rank*

Genre

mid	*genre*

Movie_Directors

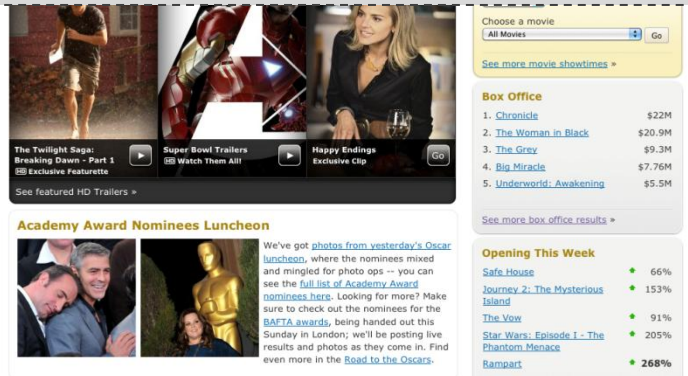
did	*mid*

Casts

aid	*mid*	*role*

Query

"What genres does Tim Burton direct?"



Motivating example: IMDB dataset

IMDB Database Schema

Actor

<u>*aid*</u>	*firstName*	*lastName*

Director

<u>*did*</u>	*firstName*	*lastName*

Movie

<u>*mid*</u>	*name*	*year*	*rank*

Genre

mid	*genre*

Movie_Directors

did	*mid*

Casts

aid	*mid*	*role*

Query

"What genres does Tim Burton direct?"


```
select     distinct g.genre
from       Director d, Movie_Directors md,
           Movie m, Genre g
where      d.lastName like 'Burton'
           and g. mid=m.mid
           and m. mid=md.mid
           and md. did=d.did
order by   g.genre
```

genre
...
Fantasy
History
Horror
Music
Musical
Mystery
Romance
...

[Meliou et al., 2010]

Motivating example: IMDB dataset

IMDB Database Schema

Actor

<u>*aid*</u>	*firstName*	*lastName*

Director

<u>*did*</u>	*firstName*	*lastName*

Movie

<u>*mid*</u>	*name*	*year*	*rank*

Genre

mid	*genre*

Movie_Directors

did	*mid*

Casts

aid	*mid*	*role*

Query

"What genres does Tim Burton direct?"


```
select     distinct g.genre
from       Director d, Movie_Directors md,
           Movie m, Genre g
where      d.lastName like 'Burton'
           and g. mid=m.mid
           and m. mid=md.mid
           and md. did=d.did
order by   g.genre
```

genre
. . .
Fantasy
History
Horror
Music
Musical
Mystery
Romance
. . .

?

Motivating example: IMDB dataset

IMDB Database Schema

Actor

aid	*firstName*	*lastName*

Director

did	*firstName*	*lastName*

Movie

mid	*name*	*year*	*rank*

Genre

mid	*genre*

Movie_Directors

did	*mid*

Casts

aid	*mid*	*role*

Query

"What genres does Tim Burton direct?"

```
select    distinct g.genre
from      Director d, Movie_Directors md,
          Movie m, Genre g
where     d.lastName like 'Burton'
          and g. mid=m.mid
          and m. mid=md.mid
          and md. did=d.did
order by  g.genre
```

genre
...
Fantasy
History
Horror
Music
Musical
Mystery
Romance
...

?

What can databases do

Provenance / Lineage:
The set of all tuples that contributed to a given output tuple

[Cheney et al. FTDB 2009], [Buneman et al. ICDT 2001], …

[Meliou et al., 2010]

Motivating example: IMDB dataset

IMDB Database Schema

Actor

aid	*firstName*	*lastName*

Director

did	*firstName*	*lastName*

Movie

mid	*name*	*year*	*rank*

Genre

mid	*genre*

Movie_Directors

did	*mid*

Casts

aid	*mid*	*role*

Query

"What genres does Tim Burton direct?"


```
select     distinct g.genre
from       Director d, Movie_Directors md,
           Movie m, Genre g
where      d.lastName like 'Burton'
           and g. mid=m.mid
           and m. mid=md.mid
           and md. did=d.did
order by   g.genre
```

genre
...
Fantasy
History
Horror
Music
Musical
Mystery
Romance
...

?

What can databases do

Provenance / Lineage:
The set of all tuples that contributed to a given output tuple

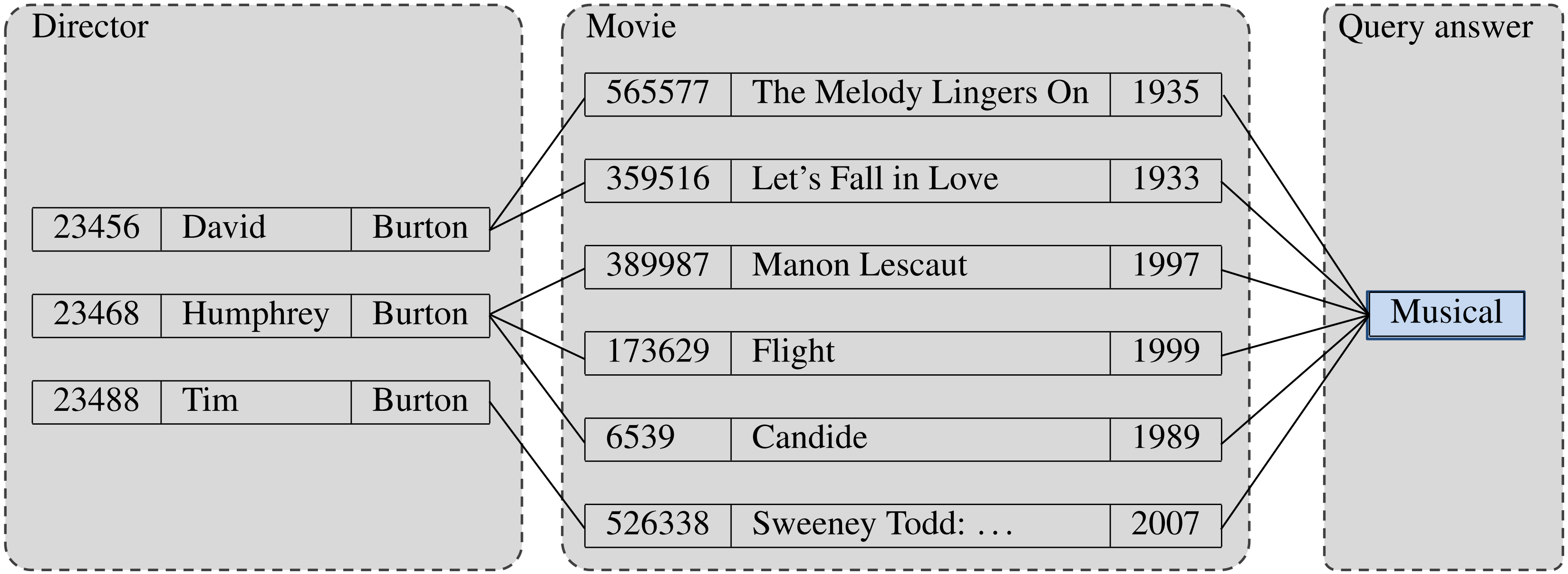
[Cheney et al. FTDB 2009], [Buneman et al. ICDT 2001], …

But

In this example, the lineage includes **137 tuples** !!

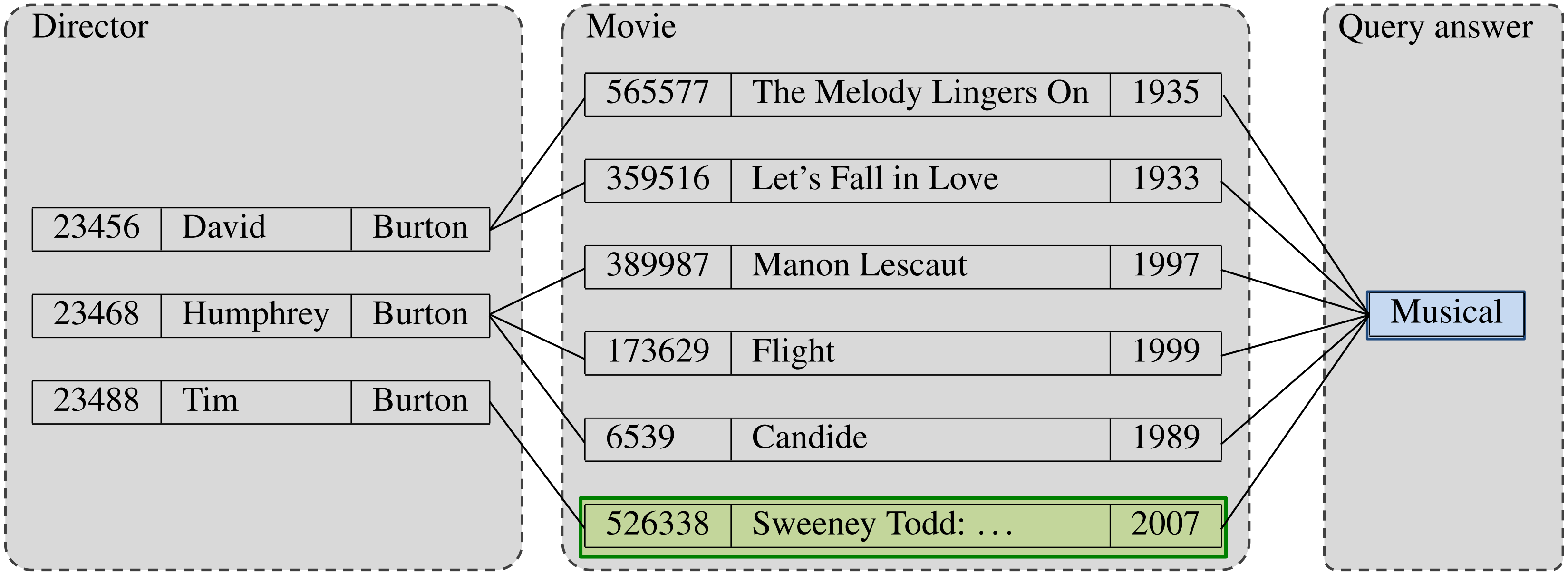
[Meliou et al., 2010]

From provenance to causality



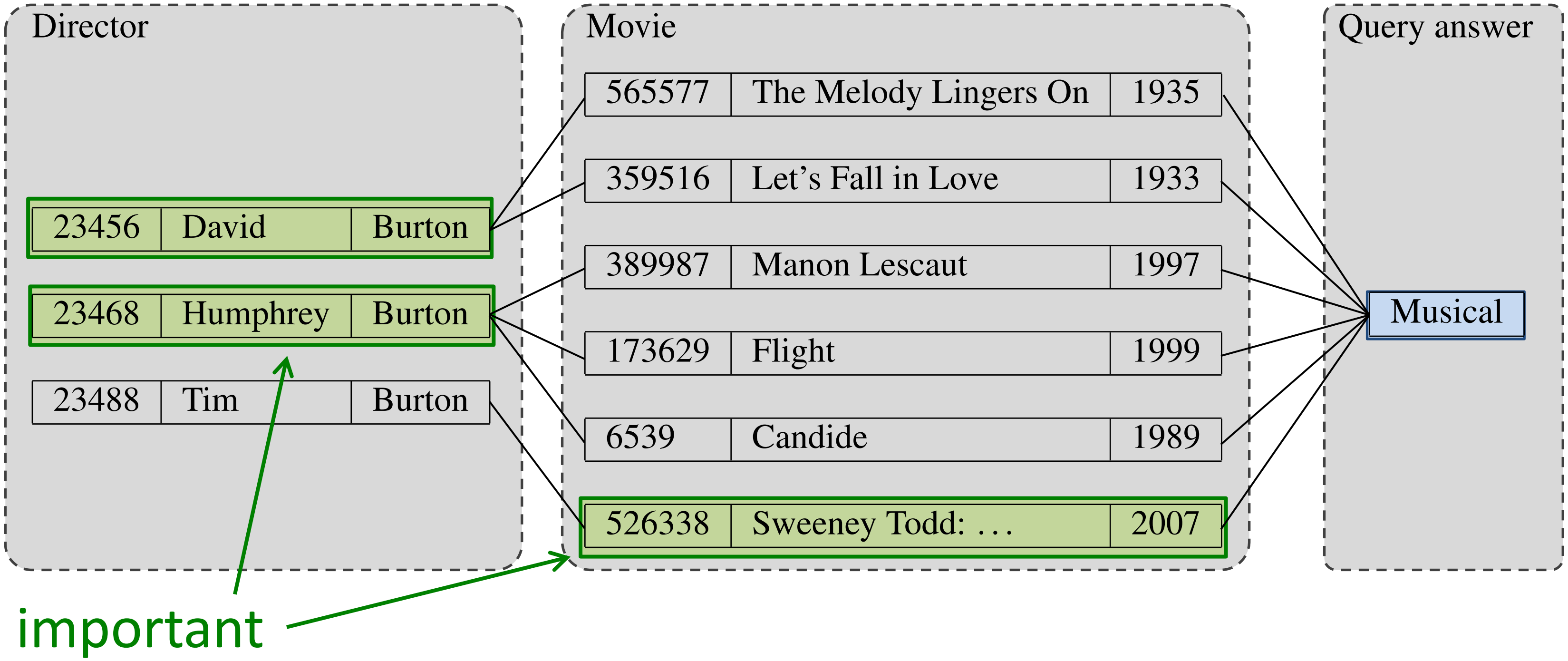
[Meliou et al., 2010]

From provenance to causality



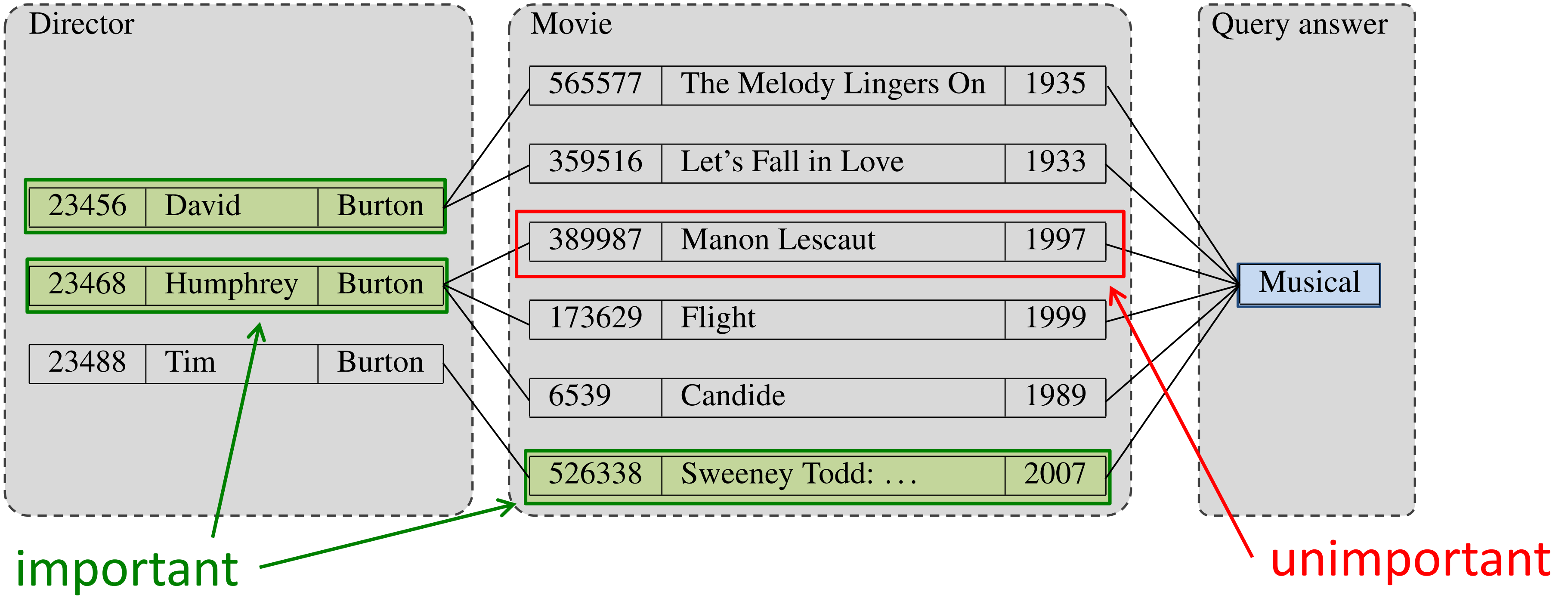
[Meliou et al., 2010]

From provenance to causality



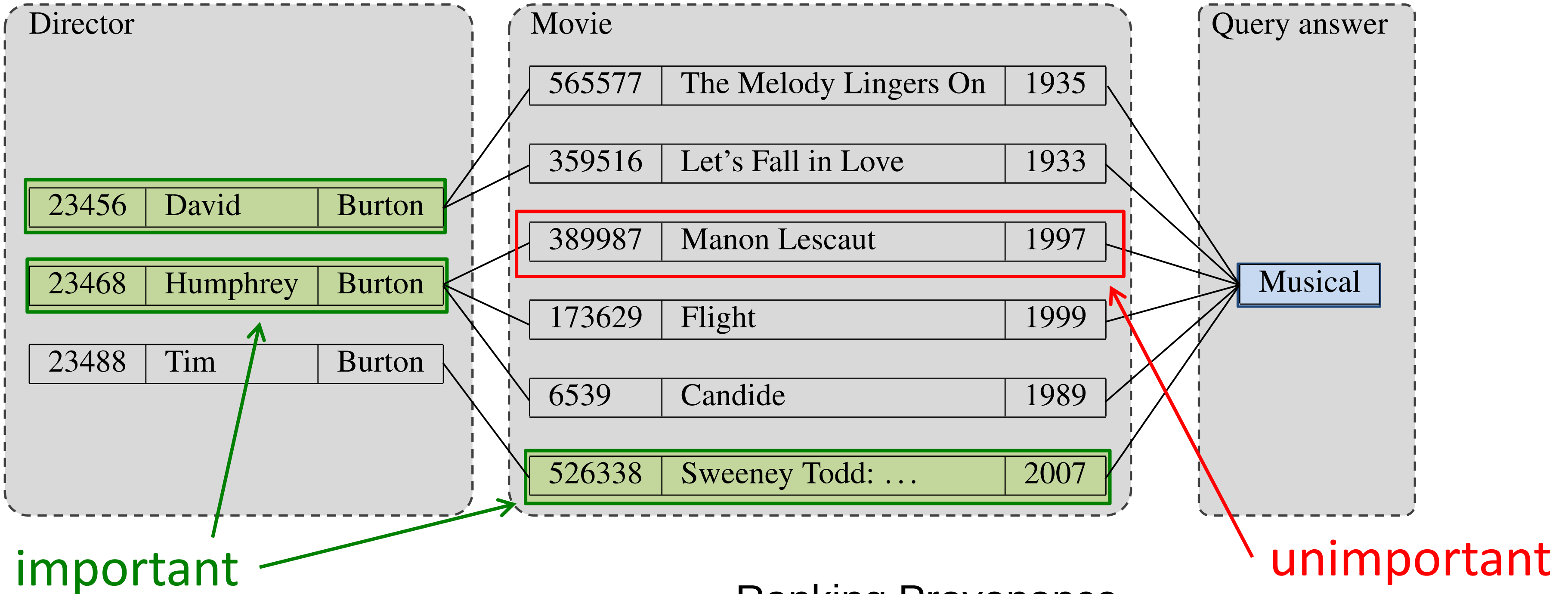
[Meliou et al., 2010]

From provenance to causality



[Meliou et al., 2010]

From provenance to causality



Goal:

Rank tuples in order of importance

Ranking Provenance

Answer tuple
Movie(526338, "Sweeney Todd", 2007)
Director(23456, David, Burton)
Director(23468, Humphrey, Burton)
Director(23488, Tim, Burton)
Movie(359516, "Let's Fall in Love", 1933)
Movie(565577, "The Melody Lingers On", 1935)
Movie(6539, "Candide", 1989)
Movie(173629, "Flight", 1999)
Movie(389987, "Manon Lescaut", 1997)

Causality for database queries

Input: database D and query Q. Output: D’=Q(D)

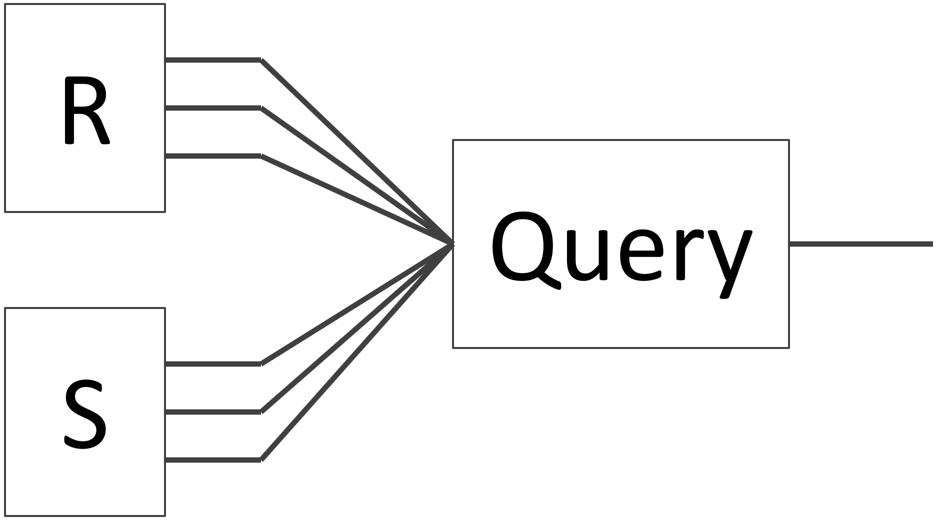
- Exogenous tuples: D^x
  - Not considered for causality: external sources, trusted sources, certain data
- Endogenous tuples: D^n
  - Potential causes: untrusted sources or tuples

[Meliou et al., 2010]

Causality for database queries

Input: database D and query Q. Output: D’=Q(D)

- Causal network:
  - Lineage of the query



$r_1 s_1 \vee r_2 s_1$

[Meliou et al., 2010]

Causality of a query answer

Input: database D and query Q. Output: D'=Q(D)

- $t \in D^n$ is a counterfactual cause for answer α
  - If $\alpha \in Q(D)$ and $\alpha \notin Q(D-t)$
- $t \in D^n$ is an actual cause for answer α
  - If $\exists \Gamma \subset D^n$ such that t is counterfactual in $D - \Gamma$

contingency set

Relationship with Halpern-Pearl causality

- Simplified definition:
  - No preemption
  - More permissible contingencies

- Open problems:
  - More complex query pipelines and reuse of views may require preemption
  - Integrity and other constraints may restrict permissible contingencies

Complexity

- Do the results of Eiter and Lukasiewicz apply?

Complexity

- Do the results of Eiter and Lukasiewicz apply?
  - Specific causal network → specific data instance

Complexity

- Do the results of Eiter and Lukasiewicz apply?
  - Specific causal network → specific data instance

- What is the complexity for a given query?
  - A given query produces a family of possible lineage expressions (for different data instances)
  - Data complexity:

    the query is fixed, the complexity is a function of the data

[Meliou et al., 2010]

Complexity

- For every conjunctive query, causality is: Polynomial, expressible in FO

Complexity

- For every conjunctive query, causality is: Polynomial, expressible in FO
- Responsibility is a harder problem

[Meliou et al., 2010]

Responsibility: example

Directors

did	firstName	lastName
28736	Steven	Spielberg
67584	Quentin	Tarantino
23488	Tim	Burton
72648	Luc	Besson

Movie_Directors

did	mid
28736	82754
67584	17653
72648	17534
23488	27645
23488	81736
67584	18764

Query: (Datalog notation)

```
q :- Directors(did,'Tim','Burton'),Movie_Directors(did,mid)
```

[Meliou et al., 2010]

Responsibility: example

Directors

	did	firstName	lastName
	28736	Steven	Spielberg
	67584	Quentin	Tarantino
s_1	23488	Tim	Burton
	72648	Luc	Besson

Movie_Directors

did	mid	
28736	82754	
67584	17653	
72648	17534	
23488	27645	r_1
23488	81736	r_2
67584	18764	

Query: (Datalog notation)

```
q :- Directors(did,'Tim','Burton'),Movie_Directors(did,mid)
```

Lineage expression: $s_1 r_1 \vee s_1 r_2$

[Meliou et al., 2010]

Responsibility: example

Directors

	did	firstName	lastName
	28736	Steven	Spielberg
	67584	Quentin	Tarantino
s_1	23488	Tim	Burton
	72648	Luc	Besson

Movie_Directors

did	mid	
28736	82754	
67584	17653	
72648	17534	
23488	27645	r_1
23488	81736	r_2
67584	18764	

Query: (Datalog notation)

```
q :- Directors(did,'Tim','Burton'),Movie_Directors(did,mid)
```

Lineage expression: $s_1 r_1 \vee s_1 r_2$

Responsibility: $\rho_t = \dfrac{1}{1 + \min_\Gamma |\Gamma|}$

[Meliou et al., 2010]

Responsibility: example

Directors

	did	firstName	lastName
	28736	Steven	Spielberg
	67584	Quentin	Tarantino
s_1	23488	Tim	Burton
	72648	Luc	Besson

Movie_Directors

did	mid	
28736	82754	
67584	17653	
72648	17534	
23488	27645	r_1
23488	81736	r_2
67584	18764	

Query: (Datalog notation)

```
q :- Directors(did,'Tim','Burton'),Movie_Directors(did,mid)
```

Lineage expression: $s_1 r_1 \vee s_1 r_2$

Responsibility: $\rho_t = \dfrac{1}{1 + \min_\Gamma |\Gamma|}$

$\rho_{s_1} = 1 \qquad \Gamma = \emptyset$

[Meliou et al., 2010]

Responsibility: example

Directors

did	firstName	lastName
28736	Steven	Spielberg
67584	Quentin	Tarantino
s_1 23488	Tim	Burton
72648	Luc	Besson

Movie_Directors

did	mid	
28736	82754	
67584	17653	
72648	17534	
23488	27645	r_1
23488	81736	r_2
67584	18764	

Query: (Datalog notation)

```
q :- Directors(did,'Tim','Burton'),Movie_Directors(did,mid)
```

Lineage expression: $s_1 r_1 \vee s_1 r_2$

Responsibility: $\rho_t = \dfrac{1}{1 + \min_\Gamma |\Gamma|}$

$$\rho_{s_1} = 1 \qquad \Gamma = \emptyset$$

$$\rho_{r_1} = \frac{1}{2} \qquad \Gamma = \{r_2\}$$

[Meliou et al., 2010]

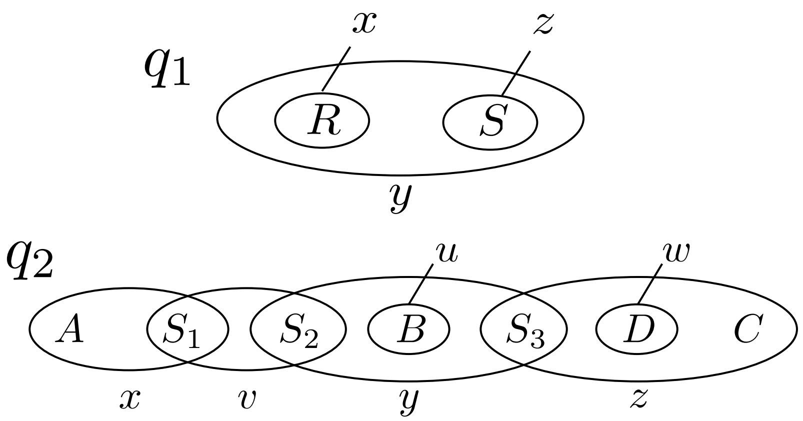
Responsibility dichotomy

PTIME	NP-hard
$q_1 :- \quad R(x,y), S(y,z)$	$h_1^* :- \quad A(x), B(y), C(z), W(x,y,z)$
$q_2 :- \quad A(x) S_1(x,v), S_2(v,y), B(y,u), S_3(y,z), D(z,w), C(z)$	$h_2^* :- \quad R(x,y), S(y,z), T(z,x)$
	$h_3^* :- \quad A(x), B(y), C(z), R(x,y), S(y,z), T(z,x)$

[Meliou et al., 2010]

Responsibility dichotomy

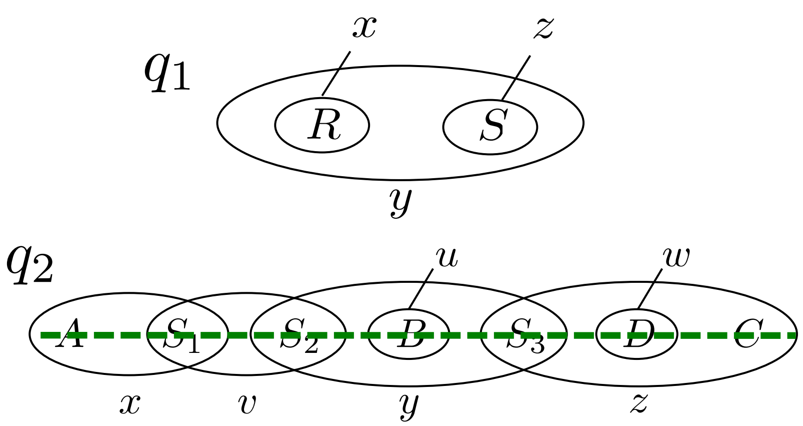
PTIME	NP-hard
$q_1 :- \quad R(x,y), S(y,z)$ $q_2 :- \quad A(x) S_1(x,v), S_2(v,y),$ $B(y,u), S_3(y,z), D(z,w), C(z)$	$h_1^* :- \quad A(x), B(y), C(z), W(x,y,z)$ $h_2^* :- \quad R(x,y), S(y,z), T(z,x)$ $h_3^* :- \quad A(x), B(y), C(z),$ $R(x,y), S(y,z), T(z,x)$



[Meliou et al., 2010]

Responsibility dichotomy

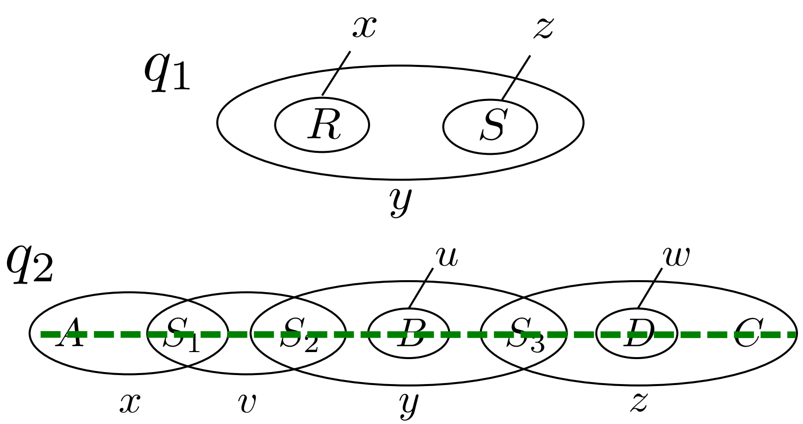
PTIME	NP-hard
q_1 :– $R(x,y), S(y,z)$	h_1^* :– $A(x), B(y), C(z), W(x,y,z)$
q_2 :– $A(x)S_1(x,v), S_2(v,y),$ $B(y,u), S_3(y,z), D(z,w), C(z)$	h_2^* :– $R(x,y), S(y,z), T(z,x)$
	h_3^* :– $A(x), B(y), C(z),$ $R(x,y), S(y,z), T(z,x)$

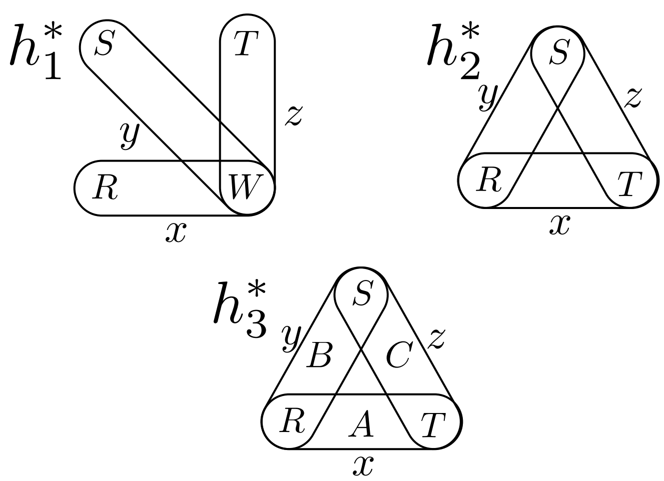


[Meliou et al., 2010]

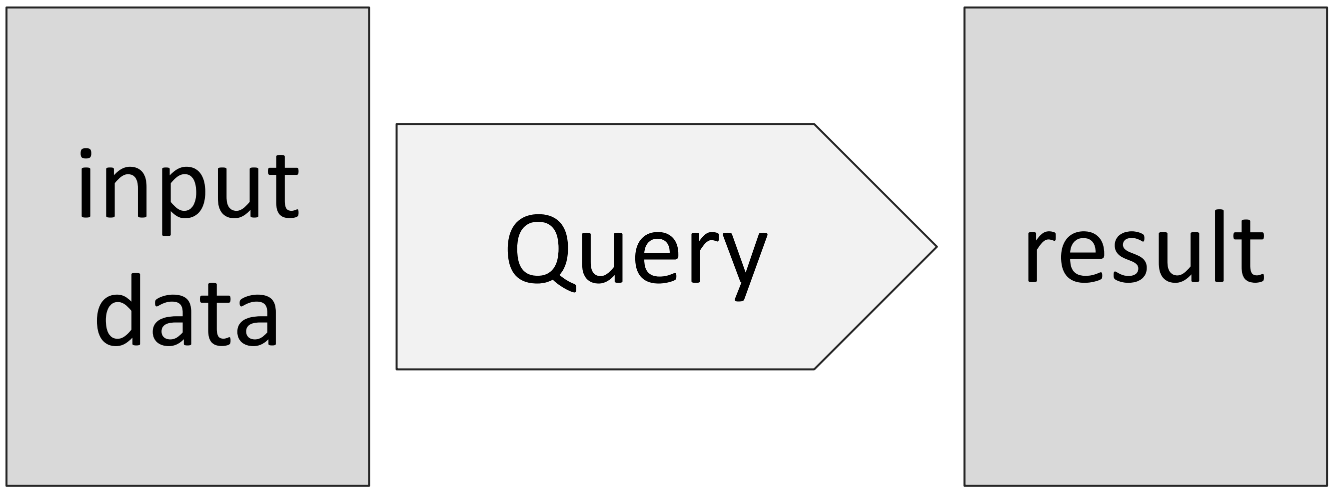
Responsibility dichotomy

PTIME	NP-hard
q_1 :– $R(x,y), S(y,z)$ q_2 :– $A(x)S_1(x,v), S_2(v,y),$ $B(y,u), S_3(y,z), D(z,w), C(z)$	h_1^* :– $A(x), B(y), C(z), W(x,y,z)$ h_2^* :– $R(x,y), S(y,z), T(z,x)$ h_3^* :– $A(x), B(y), C(z),$ $R(x,y), S(y,z), T(z,x)$

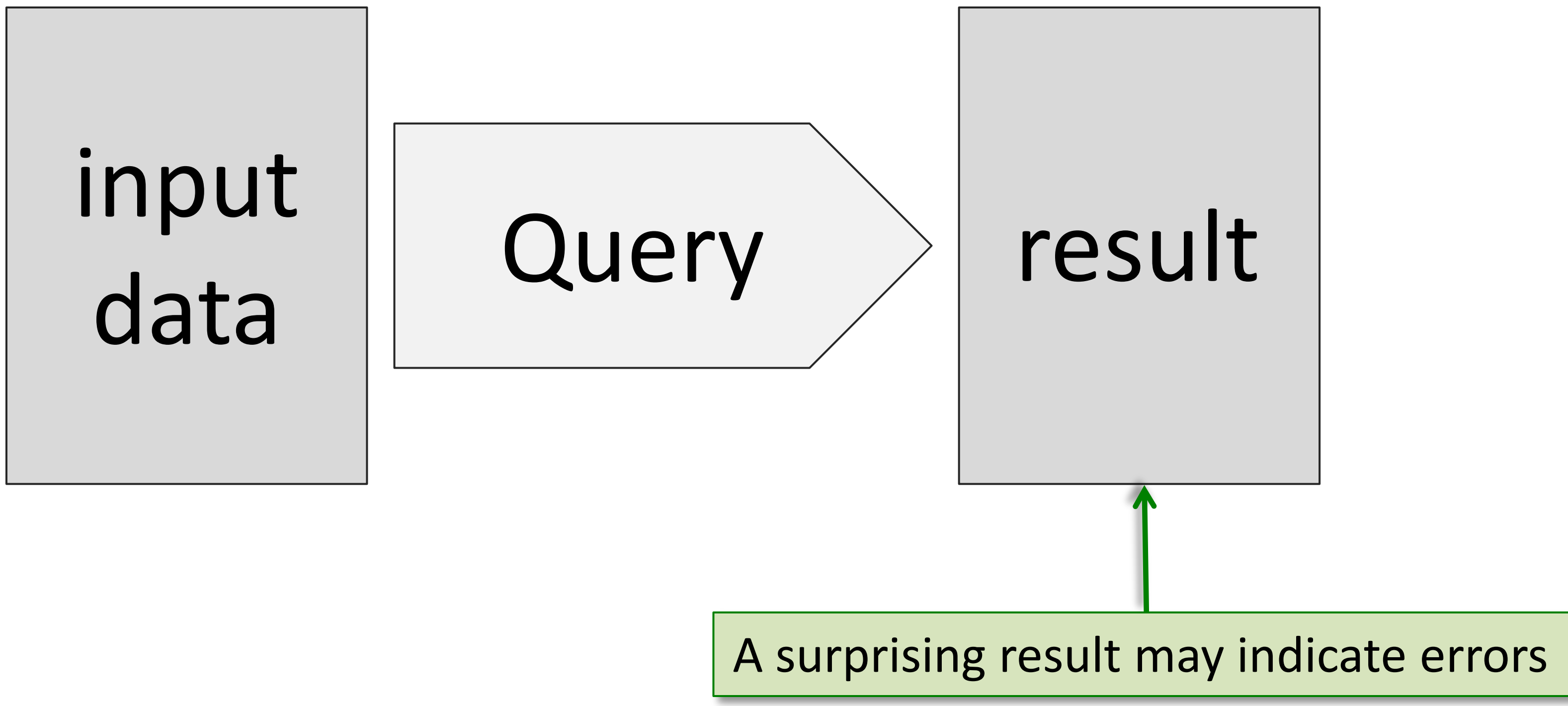




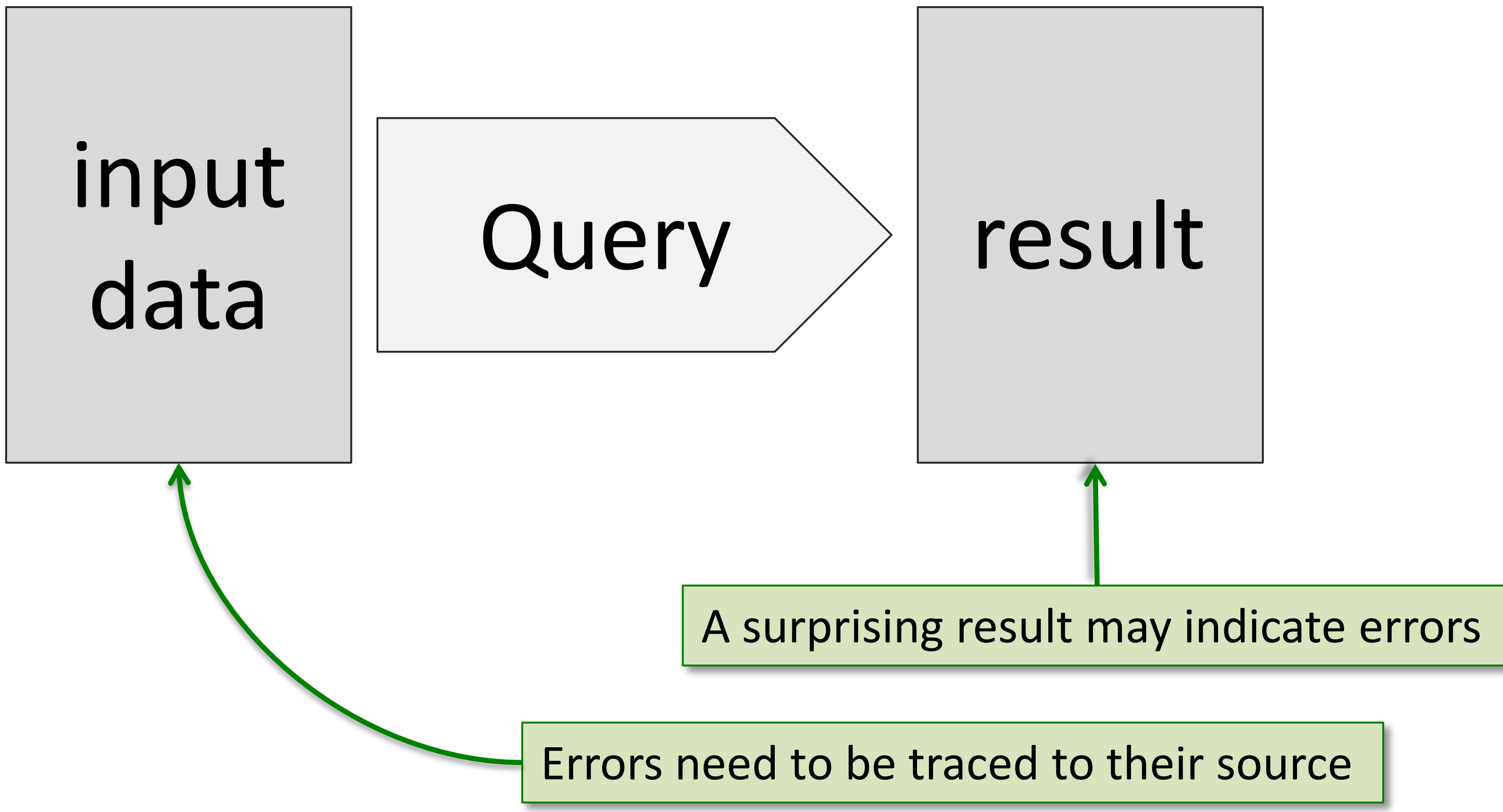
Responsibility in practice



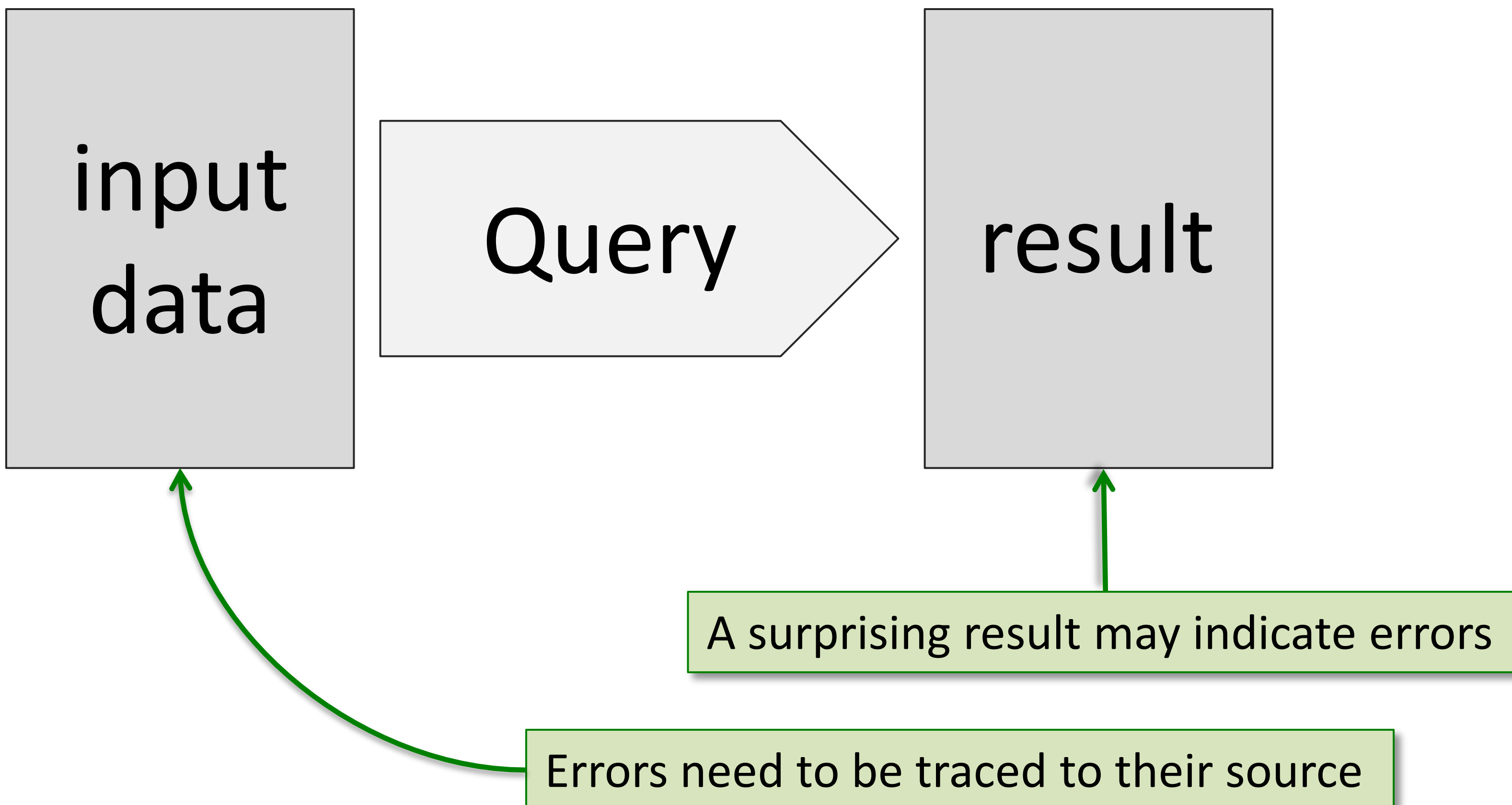
Responsibility in practice



Responsibility in practice



Responsibility in practice



Post-factum data cleaning

[Meliou et al., 2011]

Context Aware Recommendations

Data

[Meliou et al., 2011]

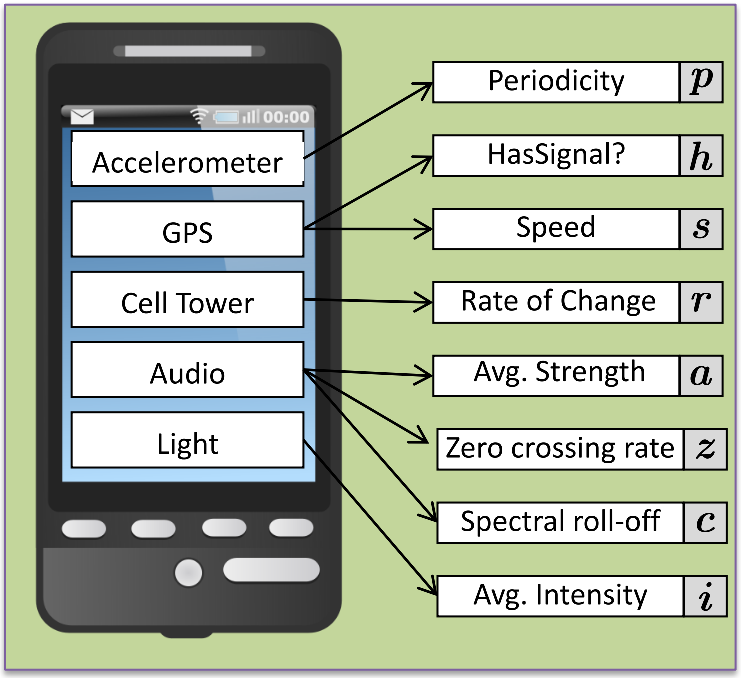
Context Aware Recommendations

Data

[Meliou et al., 2011]

Context Aware Recommendations

Data

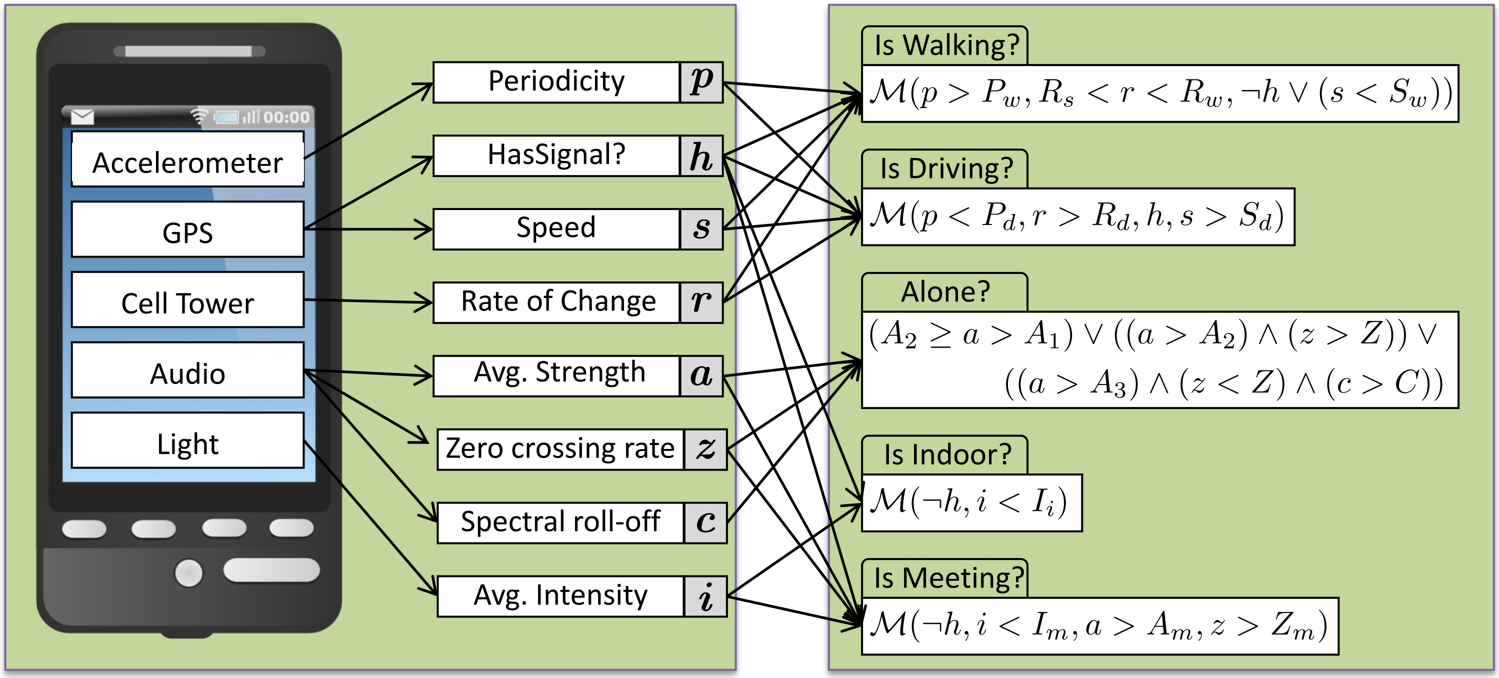


[Meliou et al., 2011]

Context Aware Recommendations

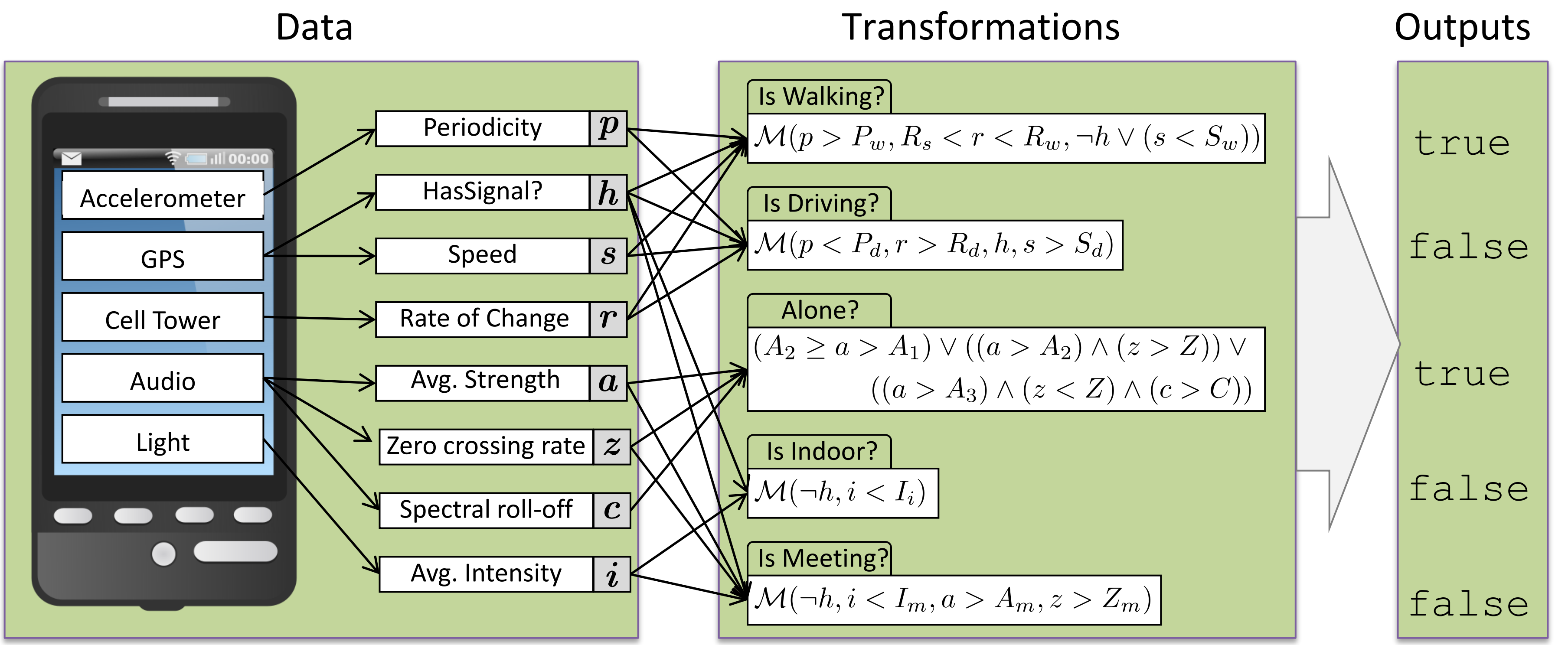
Data

Transformations

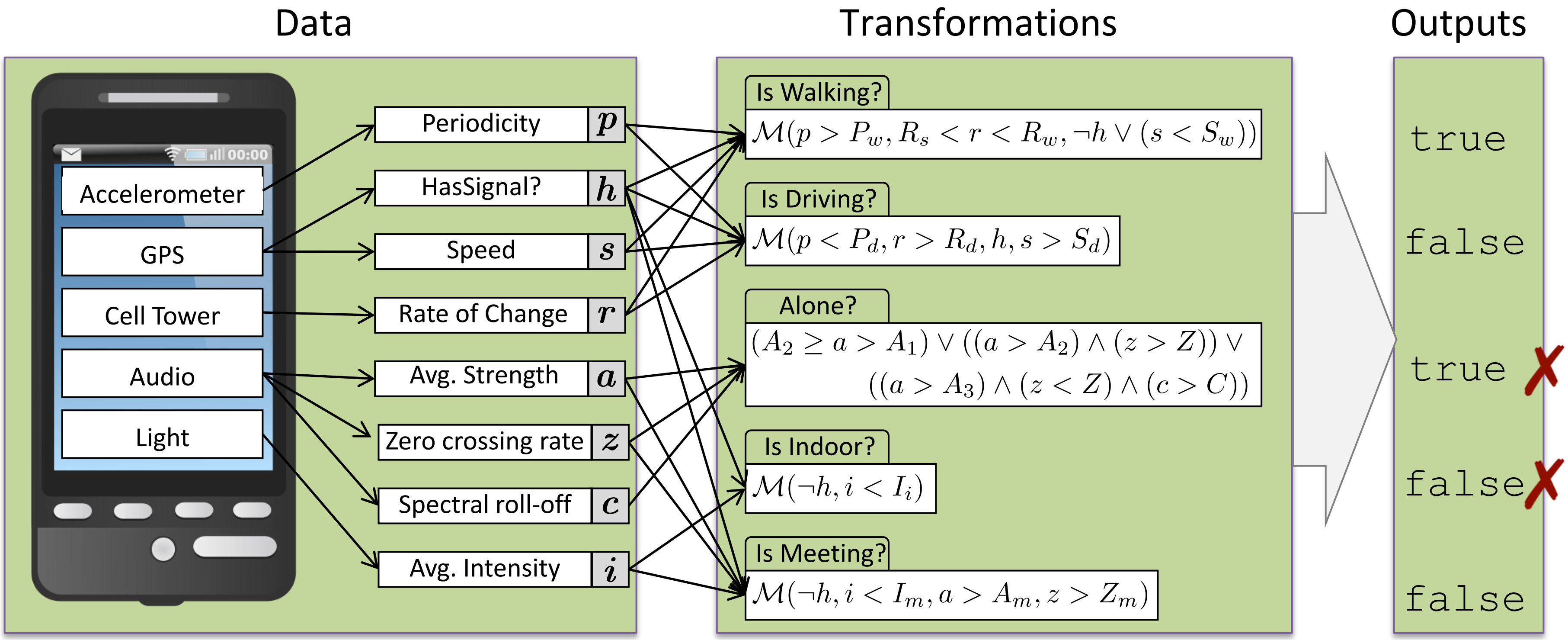


[Meliou et al., 2011]

Context Aware Recommendations

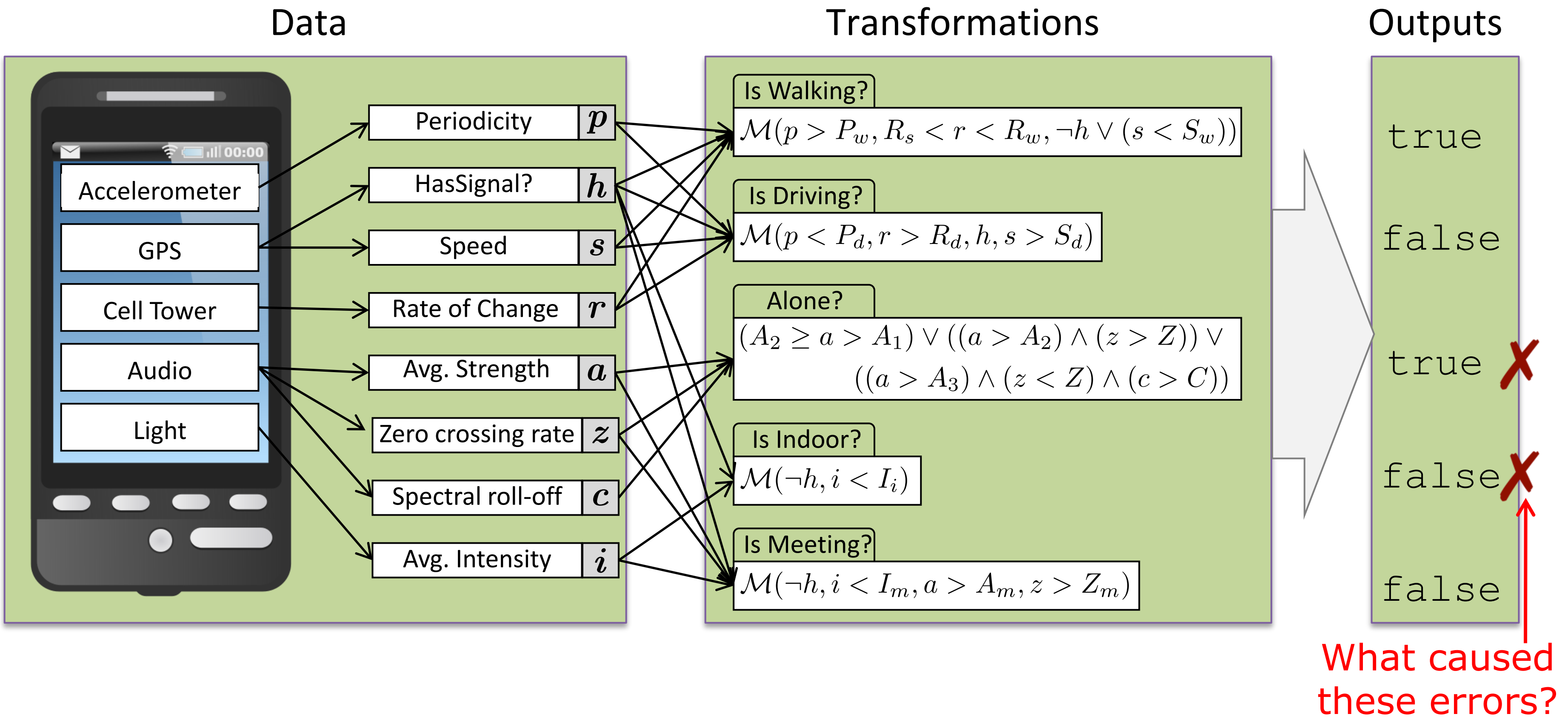


Context Aware Recommendations



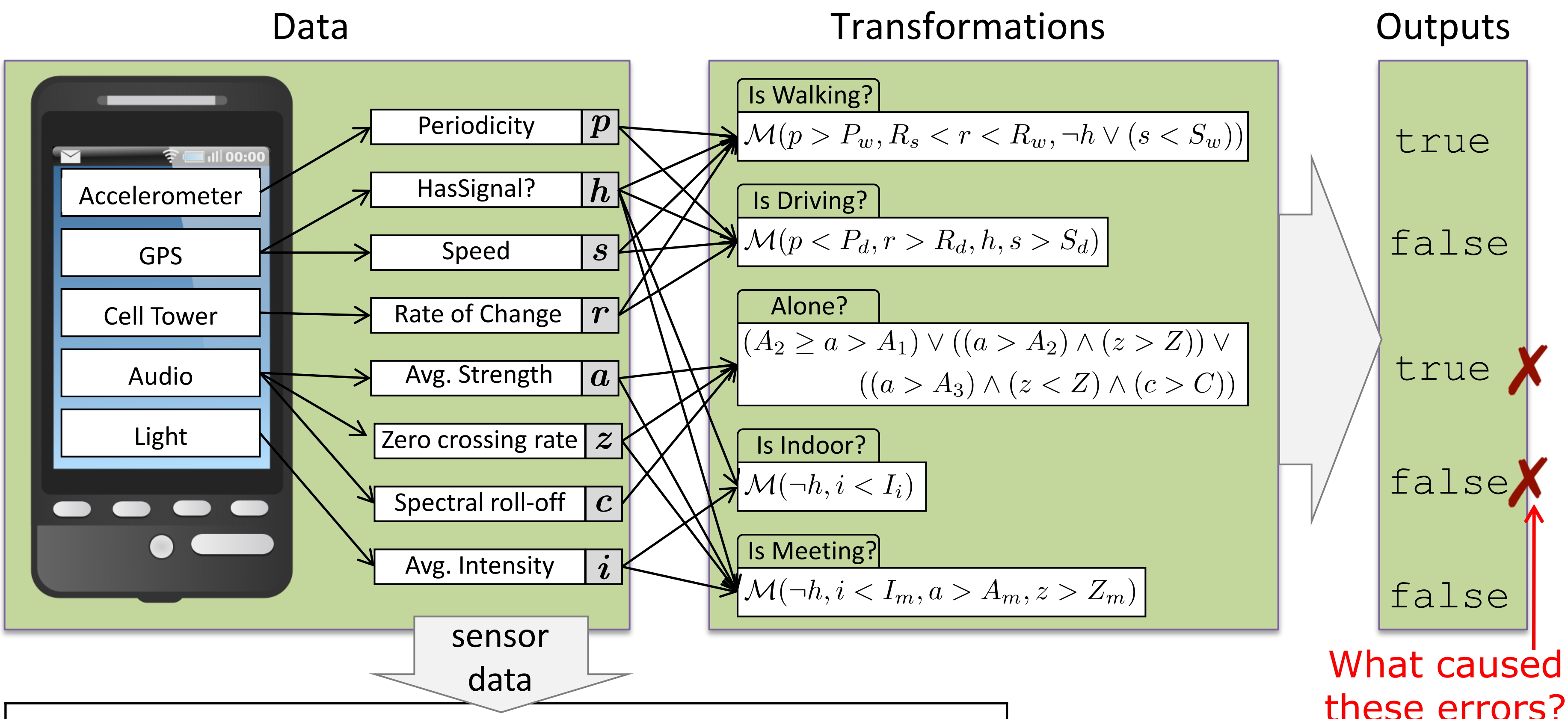
[Meliou et al., 2011]

Context Aware Recommendations



[Meliou et al., 2011]

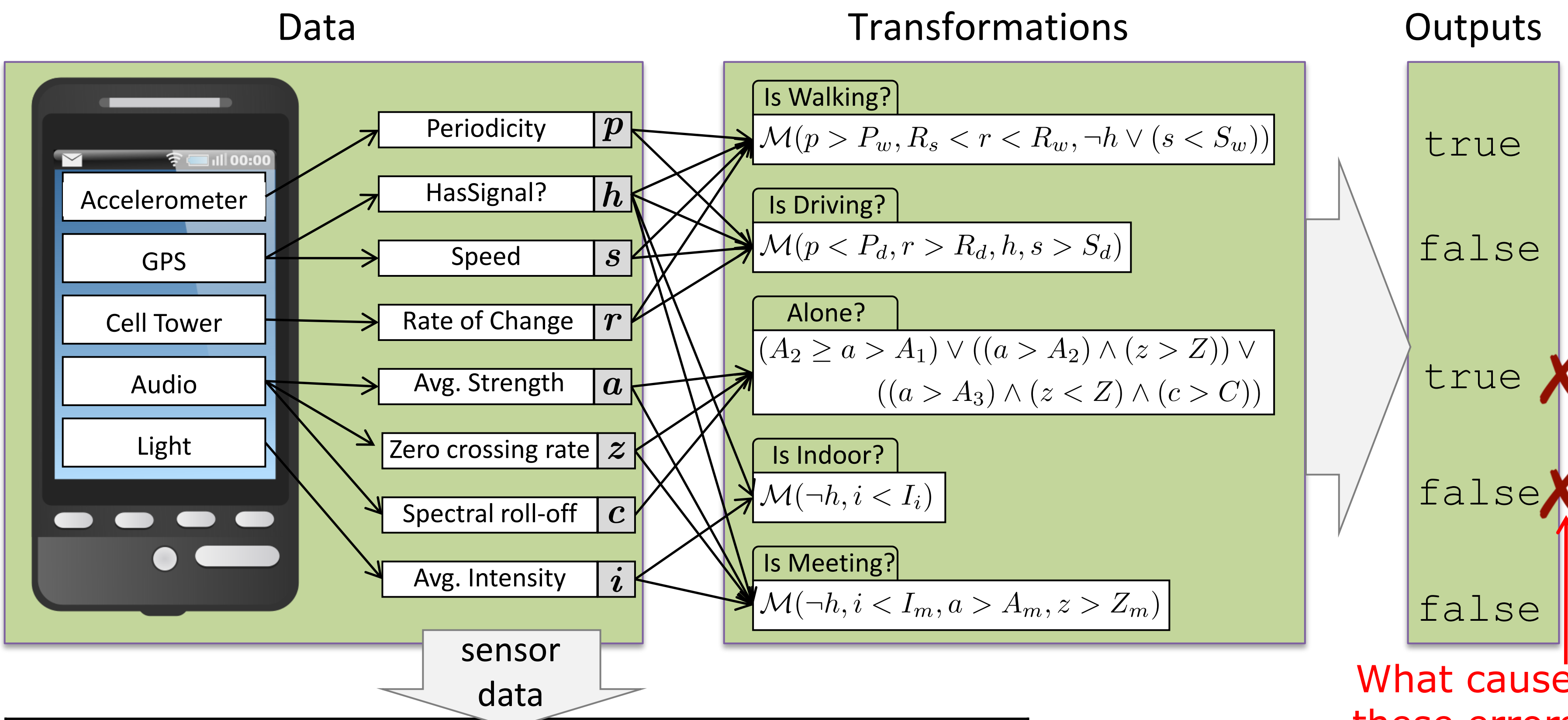
Context Aware Recommendations



0.016	True	0.067	0	0.4	0.004	0.86	0.036	10
0.0009	False	0	0	0.2	0.0039	0.81	0.034	68
0.005	True	0.19	0	0.03	0.003	0.75	0.033	17
0.0008	True	0.003	0	0.1	0.003	0.8	0.038	18

[Meliou et al., 2011]

Context Aware Recommendations



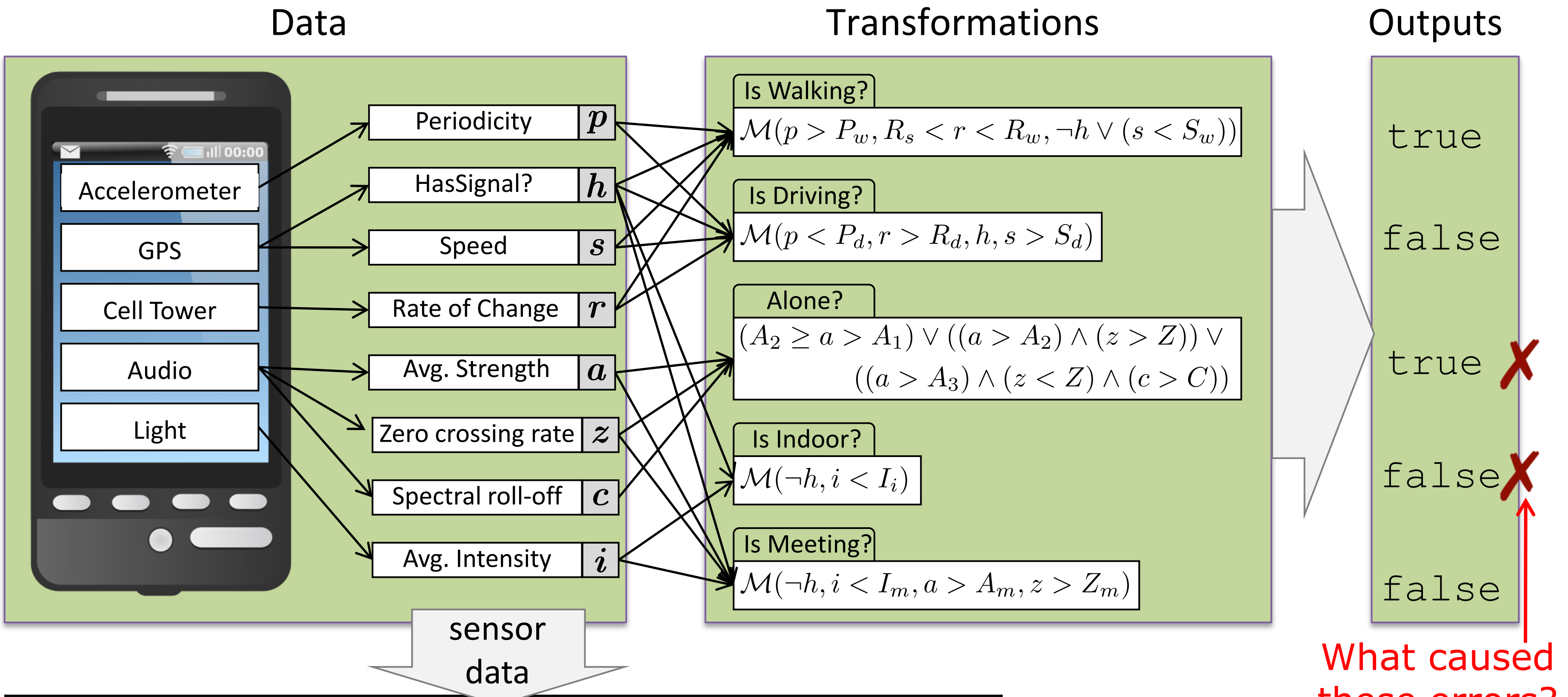
0.016	True	0.067	0	0.4	0.004	0.86	0.036	10
0.0009	False	0	0	0.2	0.0039	0.81	0.034	68
0.005	True	0.19	0	0.03	0.003	0.75	0.033	17
0.0008	True	0.003	0	0.1	0.003	0.8	0.038	18

What caused these errors?

Sensors may be faulty or inhibited

[Meliou et al., 2011]

Context Aware Recommendations



0.016	True	0.067	0	0.4	0.004	0.86	0.036	10
0.0009	False	0	0	0.2	0.0039	0.81	0.034	68
0.005	True	0.19	0	0.03	0.003	0.75	0.033	17
0.0008	True	0.003	0	0.1	0.003	0.8	0.038	18

Sensors may be faulty or inhibited

It is not straightforward to spot such errors in the provenance

[Meliou et al., 2011]

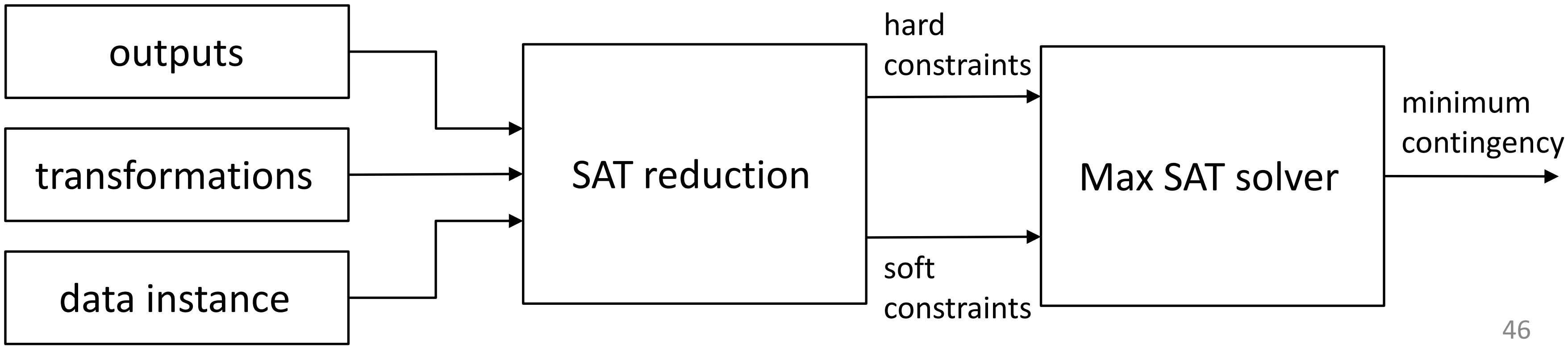
Solution

- Extension to view-conditioned causality
  - Ability to condition on multiple correct or incorrect outputs

[Meliou et al., 2011]

Solution

- Extension to view-conditioned causality
  - Ability to condition on multiple correct or incorrect outputs
- Reduction of computing responsibility to a Max SAT problem
  - Use state-of-the-art tools



Reasoning with causality
vs
Learning causality

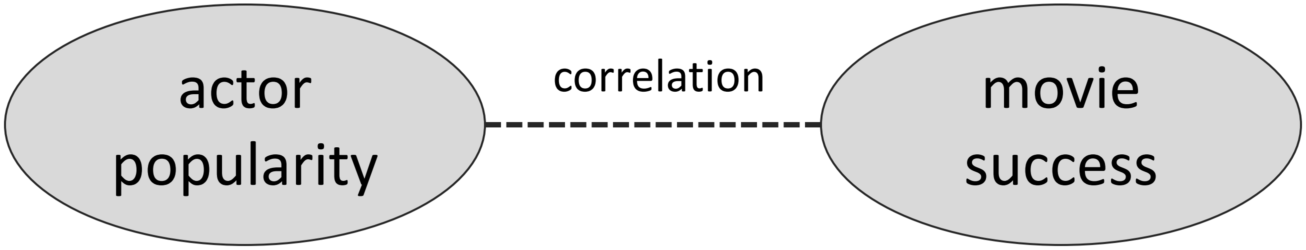
Reasoning with causality

vs

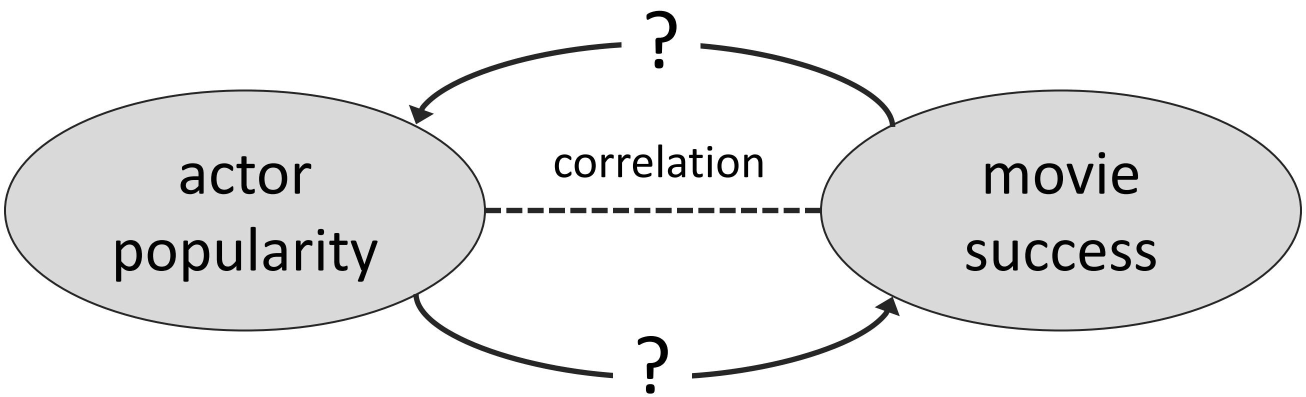
Learning causality

[Silverstein et al., 1998] [Maier et al., 2010]

Learning causal structures

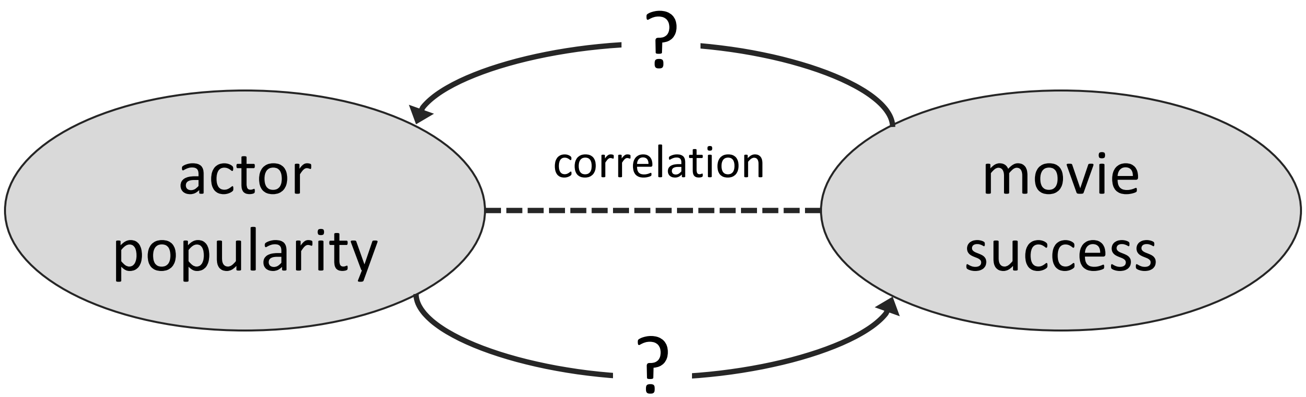


Learning causal structures



[Silverstein et al., 1998] [Maier et al., 2010]

Learning causal structures



Conditional independence:
Is one actor's popularity conditionally independent of the popularity of other actors appearing in the same movie, given that movie's success

Application of the Markov condition

[Mayrhofer et al., 2008]

Learning causal structures

Causal intuition in humans:
Understand it to discover better causal models from data

- Experimentally test how humans make associations
- Discovery: Humans use context, often violating Markovian conditions

Causality in databases: summary

- Provenance as causal network, tuples as causes
- Complexity for a query (rather than a data instance)
  - Many tractable cases
- Inferring causal relationships in data

Part 2: Explanations

a. Explanations for general DB query answers

b. Application-Specific DB Explanations

Part 2.a

• EXPLANATIONS FOR GENERAL DB QUERY ANSWERS

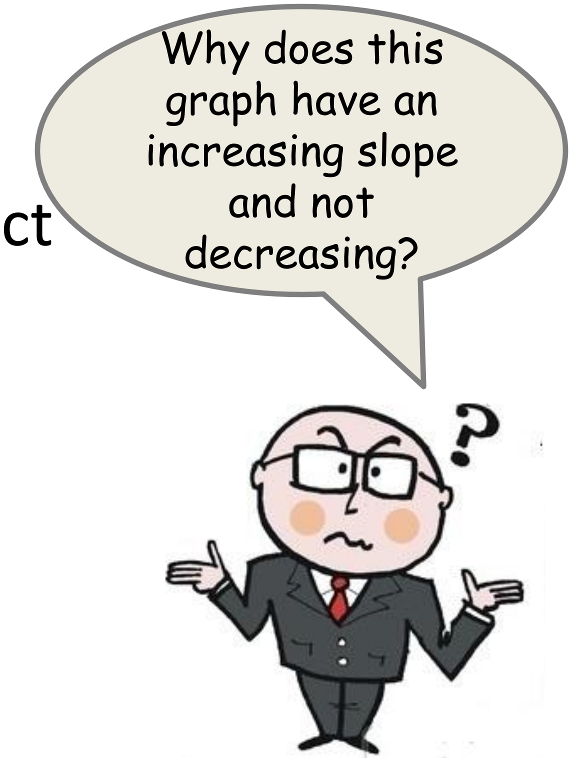
So far,

Fine-grained Actual Cause = Tuples

- Causality in AI and DB
  - defined by intervention

- In DB, goal was to compute the “responsibility” of individual input tuples in generating the output and rank them accordingly

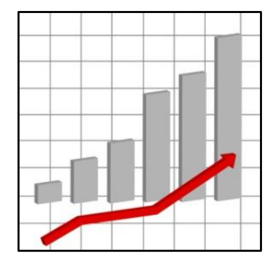
Coarse-grained Explanations = Predicates



- For "big data",
  individual input tuples may have little effect in explaining outputs. We need broader, coarse-grained explanations,
  e.g., given by predicates

- More useful to answer questions on aggregate queries visualized as graphs

- Less formal concept than causality
  - definition and ranking criteria sometimes depend on applications (more in part 2.b)

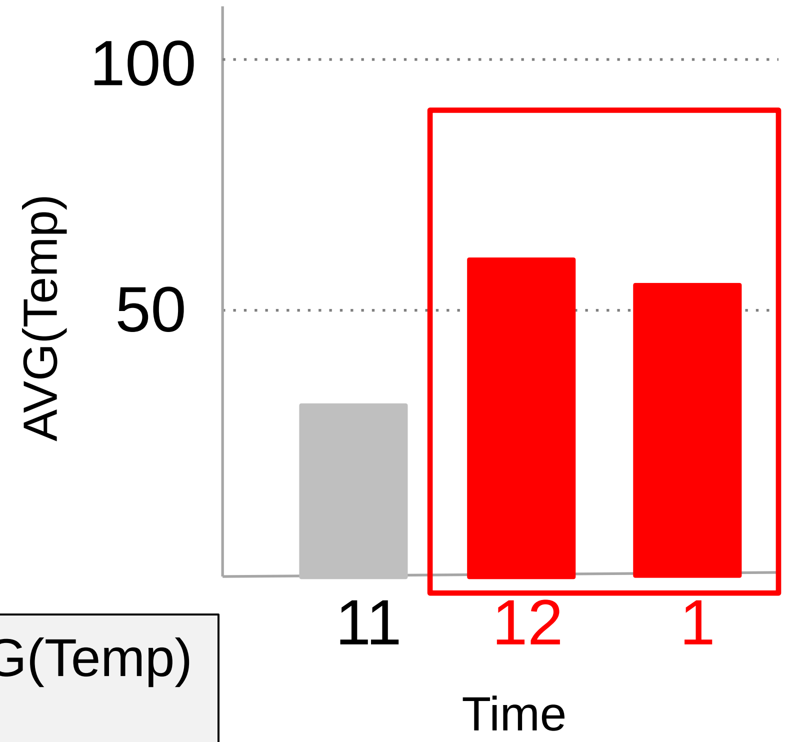


[Wu-Madden, 2013]

Example Question #1

Question on aggregate output

Time	Sensor	Volt	Humid	Temp
11	1	2.64	0.4	34
11	2	2.65	0.3	40
11	3	2.63	0.3	35
12	1	2.7	0.5	35
12	2	2.7	0.4	38
12	3	2.2	0.3	100
1	1	2.7	0.5	35
1	2	2.65	0.5	38
1	3	2.3	0.5	80

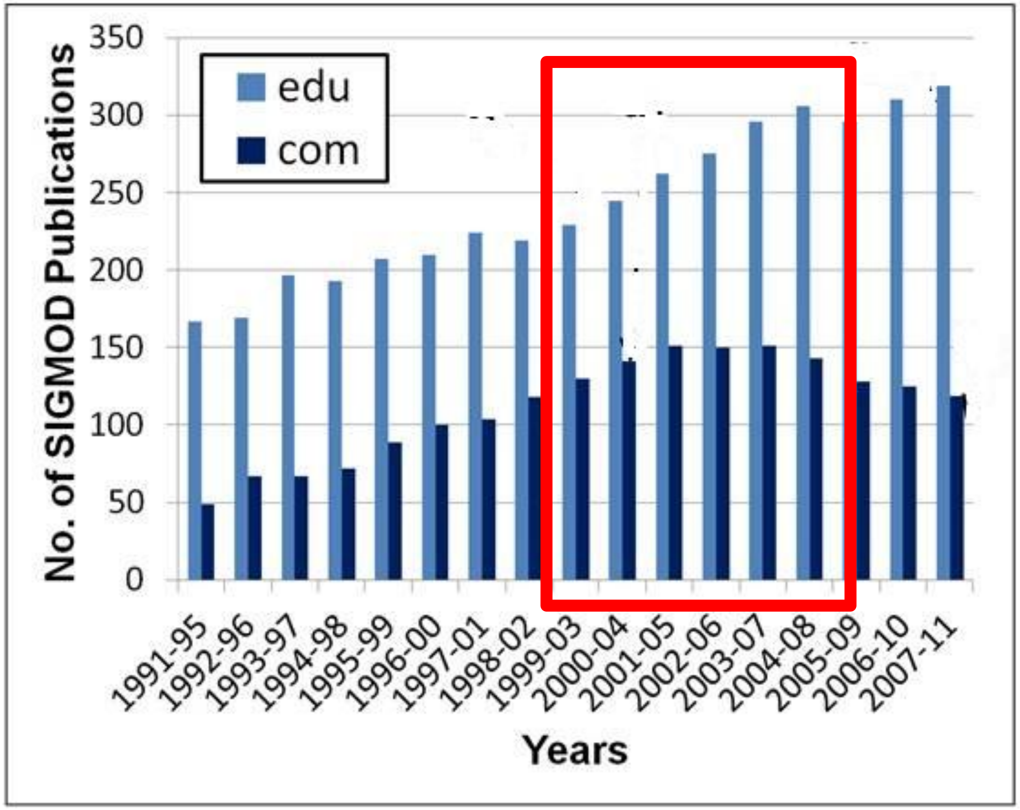



```
SELECT time, AVG(Temp)
FROM readings
GROUP BY time
```

Why is the avg. temp. high at time 12 pm and 1 pm, and low at time 11 am?

[Roy-Suciu, 2014]

Example Question #2



Question on aggregate output

Dataset:
Pre-processed DBLP
+ Affiliation data

(not all authors have affiliation info)

Why is there a peak for #sigmod papers from industry in 2000-06, while #academia papers kept increasing?

Ideal goal: Why $\equiv$ Causality

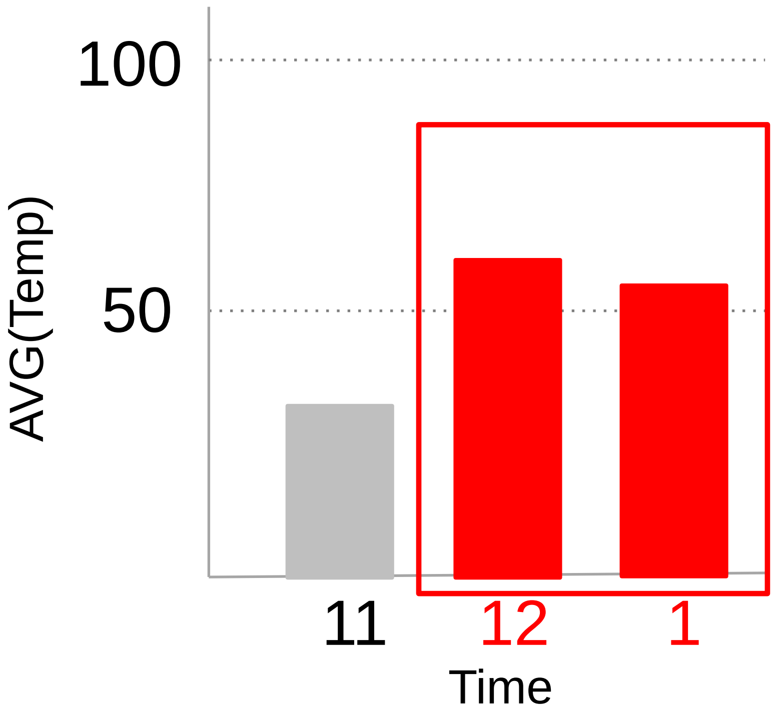
But, TRUE causality is difficult…

- True causality needs controlled, randomized experiments (repeat history)
- The database often does not even have all variables that form actual causes
- Given a limited database, broad explanations are more informative than actual causes (next slide)

Broad Explanations are more informative than Actual Causes

- We cannot repeat history and individual tuples are less informative

Time	Sensor	Volt	Humid	Temp
11	1	2.64	0.4	34
11	2	2.65	0.3	40
11	3	2.63	0.3	35
12	1	2.7	0.5	35
12	2	2.7	0.4	38
12	**3**	**2.2**	**0.3**	**100**
1	1	2.7	0.5	35
1	2	2.65	0.5	38
1	**3**	**2.3**	**0.5**	**80**

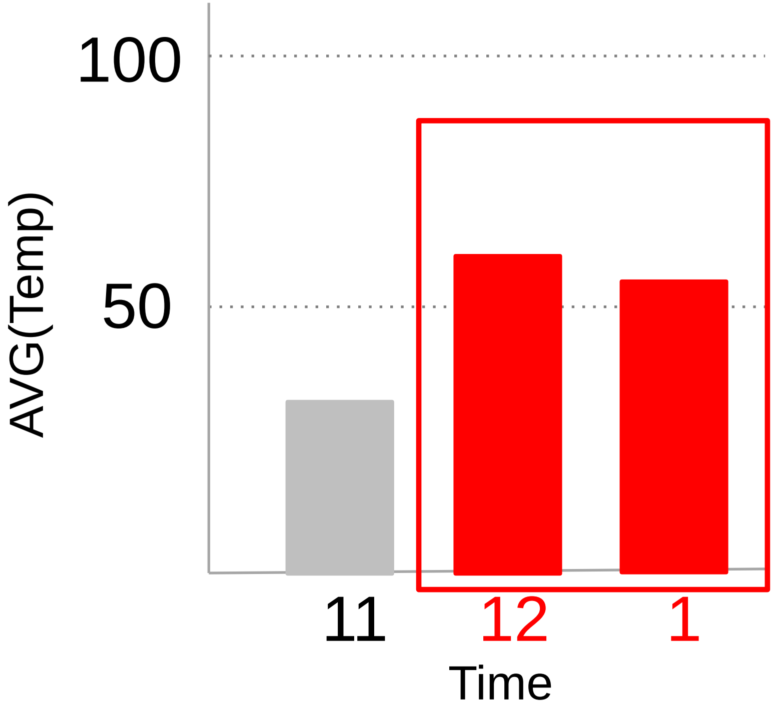


Less informative

Broad Explanations are more informative than Actual Causes

- We cannot repeat history and individual tuples are less informative

Time	Sensor	Volt	Humid	Temp
11	1	2.64	0.4	34
11	2	2.65	0.3	40
11	3	2.63	0.3	35
12	1	2.7	0.5	35
12	2	2.7	0.4	38
12	**3**	**2.2**	**0.3**	**100**
1	1	2.7	0.5	35
1	2	2.65	0.5	38
1	**3**	**2.3**	**0.5**	**80**



More informative

predicate:
Volt < 2.5 & Sensor = 3

Explanation can still be defined using "intervention" like causality!

Explanation by Intervention

- **Causality (in AI) by intervention:**

X is

a cause of Y,

if removal of X

also removes Y

keeping other conditions unchanged

Explanation by Intervention

- **Causality (in AI) by intervention:**

X is

a cause of Y,

if removal of X

also removes Y

keeping other conditions unchanged

- **Explanation (in DB) by intervention:**

Explanation by Intervention

- **Causality (in AI) by intervention:**

X is

a cause of Y,

if removal of X

also removes Y

keeping other conditions unchanged

- **Explanation (in DB) by intervention:**

A predicate X is

Explanation by Intervention

- **Causality (in AI) by intervention:**

X is

a cause of Y,

if removal of X

also removes Y

keeping other conditions unchanged

- **Explanation (in DB) by intervention:**

A predicate X is

an explanation of one or more outputs Y,

Explanation by Intervention

- **Causality (in AI) by intervention:**

X is

a cause of Y,

if removal of X

also removes Y

keeping other conditions unchanged

- **Explanation (in DB) by intervention:**

A predicate X is

an explanation of one or more outputs Y,

if removal of tuples satisfying predicate X

Explanation by Intervention

- **Causality (in AI) by intervention:**

X is

a cause of Y,

if removal of X

also removes Y

keeping other conditions unchanged

- **Explanation (in DB) by intervention:**

A predicate X is

an explanation of one or more outputs Y,

if removal of tuples satisfying predicate X

also changes Y

Explanation by Intervention

- **Causality (in AI) by intervention:**

X is

a cause of Y,

if removal of X

also removes Y

keeping other conditions unchanged

- **Explanation (in DB) by intervention:**

A predicate X is

an explanation of one or more outputs Y,

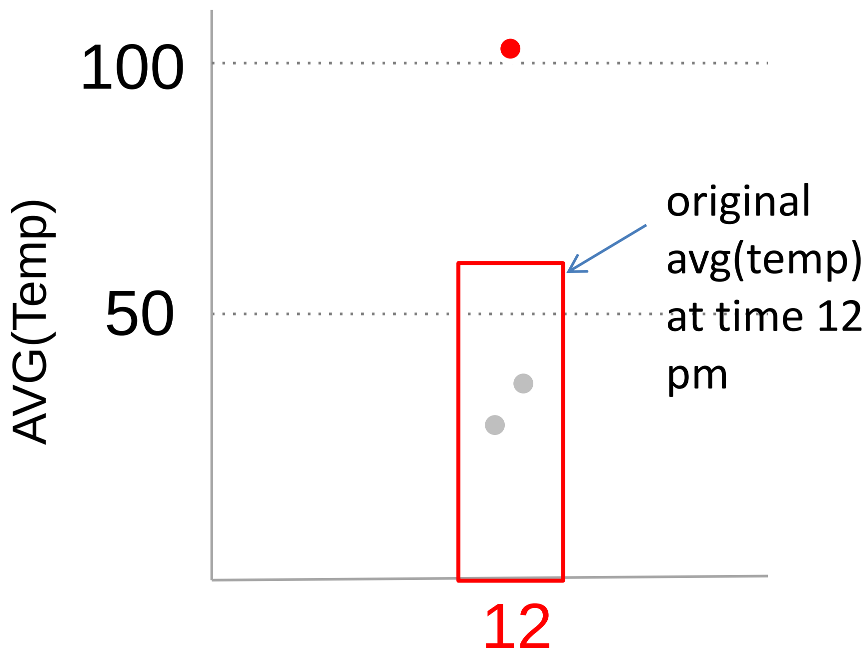
if removal of tuples satisfying predicate X

also changes Y

keeping other tuples unchanged

[Wu-Madden, 2013]

Time	Sensor	Volt	Humid	Temp
12	1	2.7	0.5	35
12	2	2.7	0.4	38
12	3	2.2	0.3	100



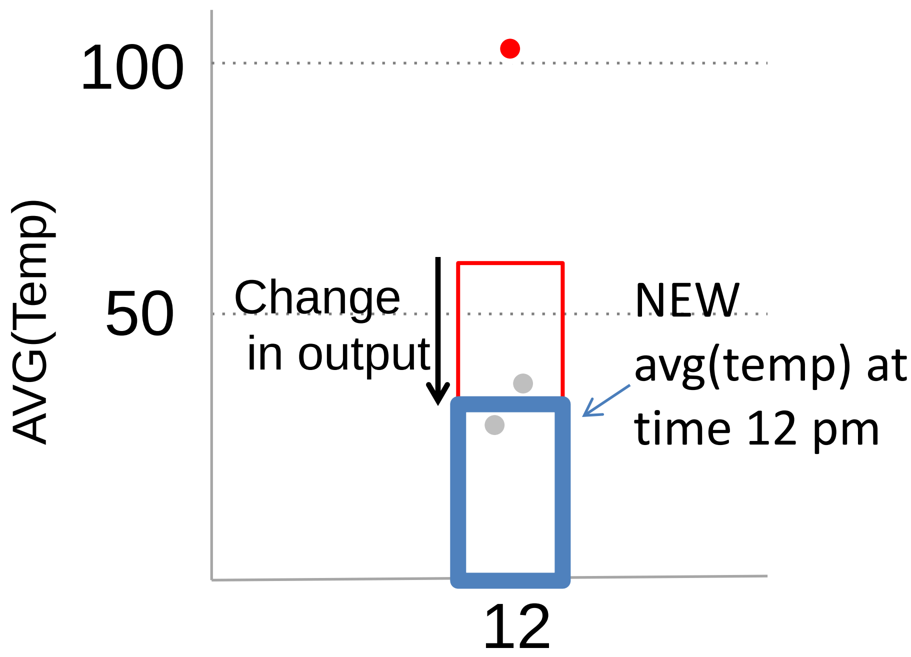
Why is the AVG(temp.) at 12pm so <u>high</u>?

predicate: Sensor = 3

[Wu-Madden, 2013]

Time	Sensor	Volt	Humid	Temp
12	1	2.7	0.5	35
12	2	2.7	0.4	38
~~12~~	~~3~~	~~2.2~~	~~0.3~~	~~100~~

Intervention!



Why is the AVG(temp.) at 12pm so <u>high</u>?

predicate: Sensor = 3

Now lower!

We need a scoring function for ranking and returning top explanations…

Scoring Function: Influence

$$\text{infl}_{\text{agg}}(p) \quad = \quad \frac{\text{Change in output}}{\text{(\# of records to make the change)}}$$

Scoring Function: Influence

$$\text{infl}_{agg}(p) = \frac{\text{Change in output}}{\text{(\# of records to make the change)}}$$

Sensor = 3

$$\frac{21.1}{1} = 21.1$$

One tuple
causes the change

[Wu-Madden, 2013]

Scoring Function: Influence

$$\text{infl}_{agg}(p) = \frac{\text{Change in output}}{\text{(\# of records to make the change)}}$$

Sensor = 3

$$\frac{21.1}{1} = 21.1$$

One tuple
causes the change

Sensor = 3 or 2

$$\frac{22.6}{2} = 11.3$$

Two tuples
cause the change

[Wu-Madden, 2013]

Scoring Function: Influence

$$\mathrm{infl}_{agg}(p) = \frac{\text{Change in output}}{\text{(\# of records to make the change)}}$$

Sensor = 3

$$\frac{21.1}{1} = 21.1$$

One tuple
causes the change

Sensor = 3 or 2

$$\frac{22.6}{2} = 11.3$$

Two tuples
cause the change

Leave the choice to the user

[Wu-Madden, 2013]

Scoring Function: Influence

$$\text{infl}_{agg}(p) = \frac{\text{Change in output}}{(\text{\# of records to make the change})^{\lambda}}$$

Sensor = 3

$$\frac{21.1}{1} = 21.1$$

One tuple
causes the change

Sensor = 3 or 2

$$\frac{22.6}{2} = 11.3$$

Two tuples
cause the change

Leave the choice to the user

[Wu-Madden, 2013]

Scoring Function: Influence

$$\text{infl}_{agg}(p) = \frac{\text{Change in output}}{(\text{\# of records to make the change})^{\lambda}}$$

Top explanation for λ = 1

Sensor = 3

$$\frac{21.1}{1} = 21.1$$

One tuple
causes the change

Sensor = 3 or 2

$$\frac{22.6}{2} = 11.3$$

Two tuples
cause the change

Leave the choice to the user

[Wu-Madden, 2013]

Scoring Function: Influence

$$\text{infl}_{agg}(p) = \frac{\text{Change in output}}{(\text{\# of records to make the change})^{\lambda}}$$

Top explanation for λ = 1

Sensor = 3

$$\frac{21.1}{1} = 21.1$$

One tuple
causes the change

Top explanation for λ = 0

Sensor = 3 or 2

$$\frac{22.6}{2} = 11.3$$

Two tuples
cause the change

Leave the choice to the user

Summary: System "Scorpion"

- Input: SQL query, outliers, normal values, λ, …
- Output: predicate p having highest influence

[Wu-Madden, 2013]

Summary: System “Scorpion”

- Input: SQL query, outliers, normal values, λ, …
- Output: predicate p having highest influence
- Uses a top-down decision tree-based algorithm that recursively partitions the predicates and merges similar predicates
  - Naïve algo is too slow as the search space of predicates is huge

[Wu-Madden, 2013]

Summary: System “Scorpion”

- Input: SQL query, outliers, normal values, λ, …
- Output: predicate p having highest influence
- Uses a top-down decision tree-based algorithm that recursively partitions the predicates and merges similar predicates
  - Naïve algo is too slow as the search space of predicates is huge
- Simple notion of intervention (implicit):

  Delete tuples that satisfy a predicate

[Roy-Suciu, 2014]

More Complex Intervention: Causal Paths in Data

Intervention in general due to a given predicate:

Delete the tuples that satisfy the predicate,

also delete tuples that directly or indirectly depend on them through causal paths

[Roy-Suciu, 2014]

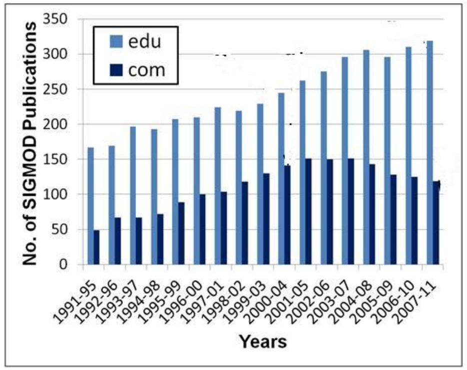
More Complex Intervention: Causal Paths in Data

Intervention in general due to a given predicate:

Delete the tuples that satisfy the predicate,

also delete tuples that directly or indirectly depend on them through causal paths

- Causal path is inherent to the data and is independent of the DB query or question asked by the user



- Next: Illustration with the DBLP example

[Roy-Suciu, 2014]

Causal Paths by Foreign Key Constraints

- Causal path $X \rightarrow Y$: removing X removes Y
- Analogy in DB:

Foreign key constraints and cascade delete semantics

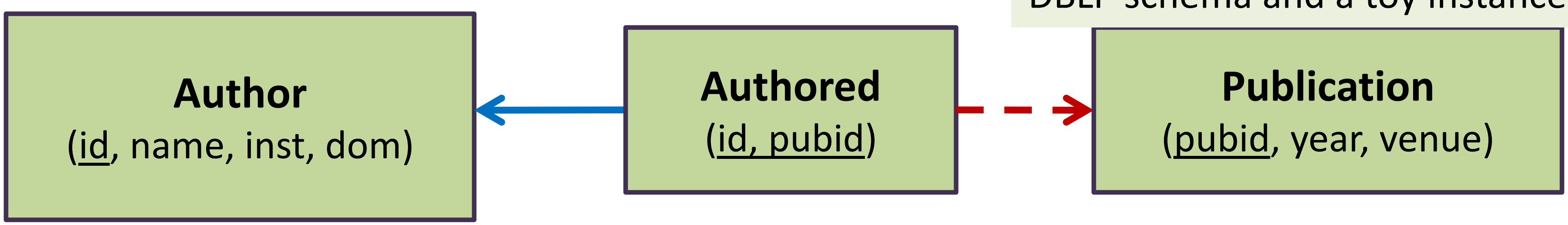
[Roy-Suciu, 2014]

Causal Paths by Foreign Key Constraints

- Causal path X $\rightarrow$ Y: removing X removes Y
- Analogy in DB:

  Foreign key constraints and cascade delete semantics

DBLP schema and a toy instance



Author

id	name	inst	dom
A1	JG	C.edu	edu
A2	RR	M.com	com
A3	CM	I.com	com

Publication

pubid	year	venue
P1	2001	SIGMOD
P2	2011	VLDB
P3	2001	SIGMOD

Authored

id	pubid
A1	P1
A2	P1
A1	P2
A3	P2
A2	P3
A3	P3

[Roy-Suciu, 2014]

Causal Paths by Foreign Key Constraints

- Causal path X $\rightarrow$ Y: removing X removes Y
- Analogy in DB:

Foreign key constraints and cascade delete semantics

DBLP schema and a toy instance

Author (id, name, inst, dom) $\leftarrow$ **Authored** (id, pubid) $\dashrightarrow$ **Publication** (pubid, year, venue)

Standard F.K. (cascade delete)

Author

id	name	inst	dom
A1	JG	C.edu	edu
A2	RR	M.com	com
A3	CM	I.com	com

Publication

pubid	year	venue
P1	2001	SIGMOD
P2	2011	VLDB
P3	2001	SIGMOD

Authored

id	pubid
A1	P1
A2	P1
A1	P2
A3	P2
A2	P3
A3	P3

[Roy-Suciu, 2014]

Causal Paths by Foreign Key Constraints

- Causal path X → Y: removing X removes Y
- Analogy in DB:

  Foreign key constraints and cascade delete semantics

DBLP schema and a toy instance

Author
(id, name, inst, dom)

Authored
(id, pubid)

Publication
(pubid, year, venue)

Standard F.K.
(cascade delete)

Author

id	name	inst	dom
~~A1~~	~~JG~~	~~C.edu~~	~~edu~~
A2	RR	M.com	com
A3	CM	I.com	com

Publication

pubid	year	venue
P1	2001	SIGMOD
P2	2011	VLDB
P3	2001	SIGMOD

Authored

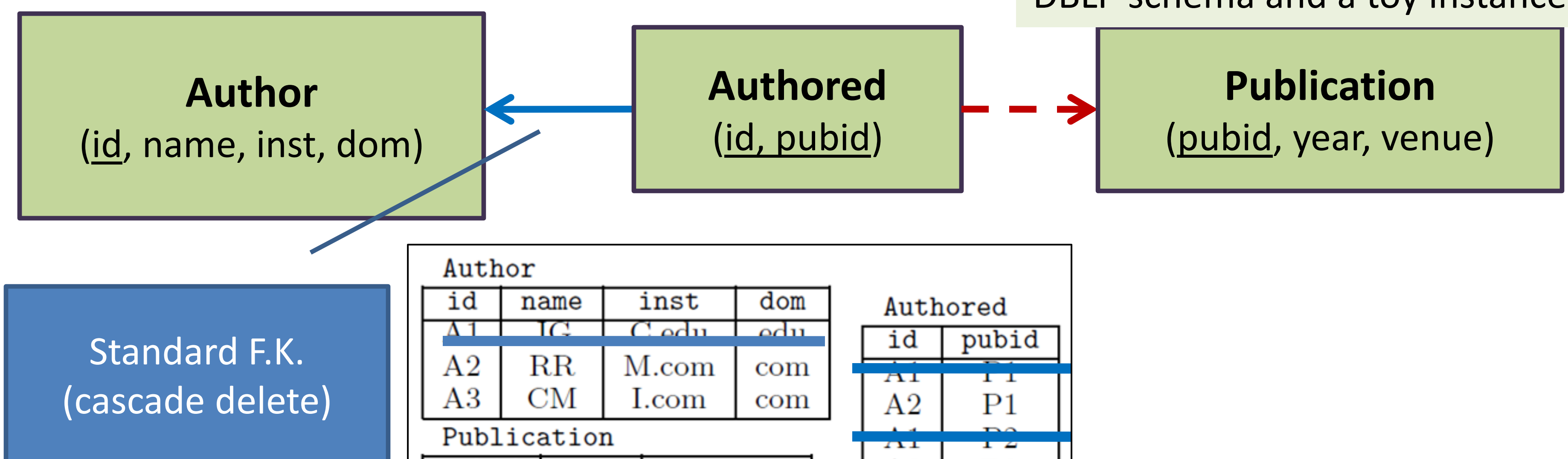
id	pubid
A1	P1
A2	P1
A1	P2
A3	P2
A2	P3
A3	P3

Causal Paths by Foreign Key Constraints

- Causal path X → Y: removing X removes Y
- Analogy in DB:

Foreign key constraints and cascade delete semantics

DBLP schema and a toy instance



Author

id	name	inst	dom
~~A1~~	~~JG~~	~~C.edu~~	~~edu~~
A2	RR	M.com	com
A3	CM	I.com	com

Publication

pubid	year	venue
P1	2001	SIGMOD
P2	2011	VLDB
P3	2001	SIGMOD

Authored

id	pubid
~~A1~~	~~P1~~
A2	P1
~~A1~~	~~P2~~
A3	P2
A2	P3
A3	P3

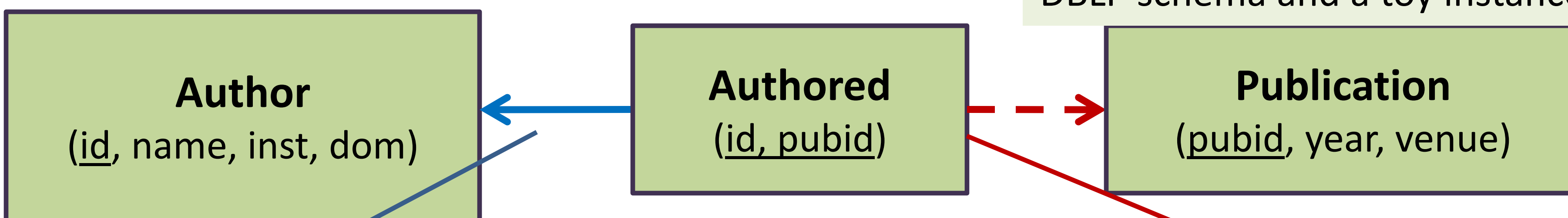
[Roy-Suciu, 2014]

Causal Paths by Foreign Key Constraints

- Causal path X → Y: removing X removes Y
- Analogy in DB:

Foreign key constraints and cascade delete semantics

DBLP schema and a toy instance



Standard F.K. (cascade delete)

Author

id	name	inst	dom
A1	JG	C.edu	edu
A2	RR	M.com	com
A3	CM	I.com	com

Publication

pubid	year	venue
P1	2001	SIGMOD
P2	2011	VLDB
P3	2001	SIGMOD

Authored

id	pubid
A1	P1
A2	P1
A1	P2
A3	P2
A2	P3
A3	P3

Back and Forth F.K.
(cascade delete
+
reverse cascade delete)

[Roy-Suciu, 2014]

Causal Paths by Foreign Key Constraints

- Causal path X → Y: removing X removes Y
- Analogy in DB:

  Foreign key constraints and cascade delete semantics

DBLP schema and a toy instance

Author
(id, name, inst, dom)

Authored
(id, pubid)

Publication
(pubid, year, venue)

Standard F.K.
(cascade delete)

Forward

Back and Forth F.K.
(cascade delete
+
reverse cascade delete)

Author

id	name	inst	dom
A1	JG	C.edu	edu
A2	RR	M.com	com
A3	CM	I.com	com

Publication

pubid	year	venue
P1	2001	SIGMOD
P2	2011	VLDB
~~P3~~	~~2001~~	~~SIGMOD~~

Authored

id	pubid
A1	P1
A2	P1
A1	P2
A3	P2
A2	P3
A3	P3

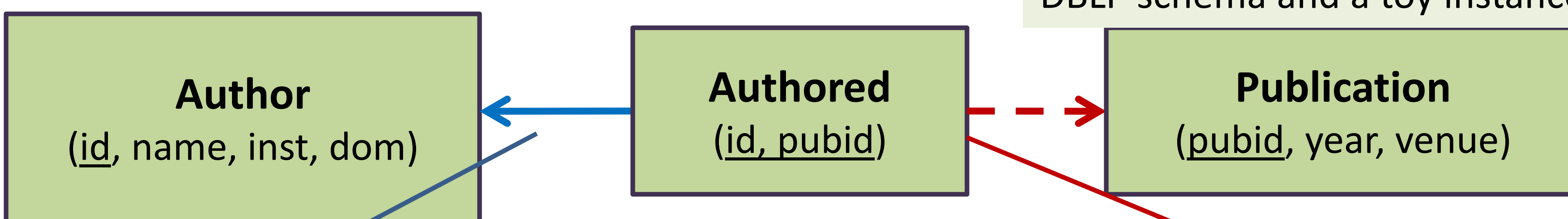
[Roy-Suciu, 2014]

Causal Paths by Foreign Key Constraints

- Causal path X $\rightarrow$ Y: removing X removes Y
- Analogy in DB:

  Foreign key constraints and cascade delete semantics

DBLP schema and a toy instance



Standard F.K.
(cascade delete)

Forward

Back and Forth F.K.
(cascade delete
+
reverse cascade delete)

Author

id	name	inst	dom
A1	JG	C.edu	edu
A2	RR	M.com	com
A3	CM	I.com	com

Publication

pubid	year	venue
P1	2001	SIGMOD
P2	2011	VLDB
~~P3~~	~~2001~~	~~SIGMOD~~

Authored

id	pubid
A1	P1
A2	P1
A1	P2
A3	P2
~~A2~~	~~P3~~
~~A3~~	~~P3~~

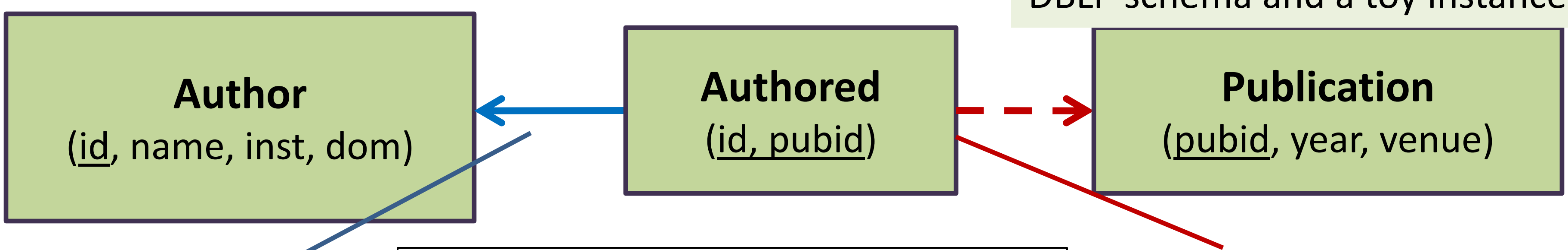
[Roy-Suciu, 2014]

Causal Paths by Foreign Key Constraints

- Causal path X → Y: removing X removes Y
- Analogy in DB:

Foreign key constraints and cascade delete semantics

DBLP schema and a toy instance



Standard F.K.
(cascade delete)

Reverse

Author

id	name	inst	dom
A1	JG	C.edu	edu
A2	RR	M.com	com
A3	CM	I.com	com

Publication

pubid	year	venue
P1	2001	SIGMOD
P2	2011	VLDB
P3	2001	SIGMOD

Authored

id	pubid
A1	P1
A2	P1
A1	P2
A3	P2
~~A2~~	~~P3~~
A3	P3

Back and Forth F.K.
(cascade delete
+
reverse cascade delete)

[Roy-Suciu, 2014]

Causal Paths by Foreign Key Constraints

- Causal path X → Y: removing X removes Y
- Analogy in DB:

Foreign key constraints and cascade delete semantics

DBLP schema and a toy instance

Author
(id, name, inst, dom)

Authored
(id, pubid)

Publication
(pubid, year, venue)

Standard F.K.
(cascade delete)

Reverse

Back and Forth F.K.
(cascade delete
+
reverse cascade delete)

Author

id	name	inst	dom
A1	JG	C.edu	edu
A2	RR	M.com	com
A3	CM	I.com	com

Publication

pubid	year	venue
P1	2001	SIGMOD
P2	2011	VLDB
~~P3~~	~~2001~~	~~SIGMOD~~

Authored

id	pubid
A1	P1
A2	P1
A1	P2
A3	P2
~~A2~~	~~P3~~
A3	P3

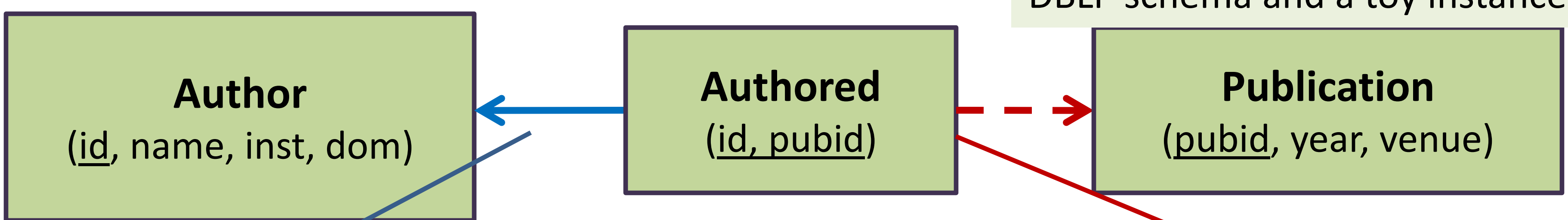
[Roy-Suciu, 2014]

Causal Paths by Foreign Key Constraints

Intuition:
- An author **can exist** if one of her papers is deleted
- A paper **cannot exist** if any of its co-authors is deleted

Note: Both F.K.s could be standard

DBLP schema and a toy instance



Standard F.K.
(cascade delete)

Reverse

Author

id	name	inst	dom
A1	JG	C.edu	edu
A2	RR	M.com	com
A3	CM	I.com	com

Publication

pubid	year	venue
P1	2001	SIGMOD
P2	2011	VLDB
~~P3~~	~~2001~~	~~SIGMOD~~

Authored

id	pubid
A1	P1
A2	P1
A1	P2
A3	P2
~~A2~~	~~P3~~
A3	P3

Back and Forth F.K.
(cascade delete
+
reverse cascade delete)

[Roy-Suciu, 2014]

Intervention through Causal Paths

Forward
Reverse

Author

id	name	inst	dom
A1	JG	C.edu	edu
A2	RR	M.com	com
A3	CM	I.com	com

Publication

pubid	year	venue
P1	2001	SIGMOD
P2	2011	VLDB
P3	2001	SIGMOD

Authored

id	pubid
A1	P1
A2	P1
A1	P2
A3	P2
A2	P3
A3	P3

[Roy-Suciu, 2014]

Intervention through Causal Paths

Candidate explanation predicate **ϕ : [name = ‘RR’]**

Forward

Reverse

Author

id	name	inst	dom
A1	JG	C.edu	edu
A2	RR	M.com	com
A3	CM	I.com	com

Authored

id	pubid
A1	P1
A2	P1
A1	P2
A3	P2
A2	P3
A3	P3

Publication

pubid	year	venue
P1	2001	SIGMOD
P2	2011	VLDB
P3	2001	SIGMOD

[Roy-Suciu, 2014]

Intervention through Causal Paths

Candidate explanation predicate **ϕ : [name = 'RR']**

Forward
Reverse

Author

id	name	inst	dom
A1	JG	C.edu	edu
~~A2~~	~~RR~~	~~M.com~~	~~com~~
A3	CM	I.com	com

Publication

pubid	year	venue
P1	2001	SIGMOD
P2	2011	VLDB
P3	2001	SIGMOD

Authored

id	pubid
A1	P1
A2	P1
A1	P2
A3	P2
A2	P3
A3	P3

[Roy-Suciu, 2014]

Intervention through Causal Paths

Candidate explanation predicate **ϕ : [name = ‘RR’]**

Forward

Reverse

Author

id	name	inst	dom
A1	JG	C.edu	edu
~~A2~~	~~RR~~	~~M.com~~	~~com~~
A3	CM	I.com	com

Authored

id	pubid
A1	P1
A2	P1
A1	P2
A3	P2
A2	P3
A3	P3

Publication

pubid	year	venue
P1	2001	SIGMOD
P2	2011	VLDB
P3	2001	SIGMOD

Intervention Δ_ϕ :
Tuples T_0 that satisfy ϕ + Tuples reachable from T_0

[Roy-Suciu, 2014]

Intervention through Causal Paths

Candidate explanation predicate **ϕ : [name = 'RR']**

Forward

Reverse

Author

id	name	inst	dom
A1	JG	C.edu	edu
~~A2~~	~~RR~~	~~M.com~~	~~com~~
A3	CM	I.com	com

Publication

pubid	year	venue
P1	2001	SIGMOD
P2	2011	VLDB
P3	2001	SIGMOD

Authored

id	pubid
A1	P1
~~A2~~	~~P1~~
A1	P2
A3	P2
~~A2~~	~~P3~~
A3	P3

Intervention Δ_ϕ :
Tuples T_0 that satisfy ϕ + Tuples reachable from T_0

[Roy-Suciu, 2014]

Intervention through Causal Paths

Candidate explanation predicate **ϕ : [name = 'RR']**

Forward
Reverse

Author

id	name	inst	dom
A1	JG	C.edu	edu
~~A2~~	~~RR~~	~~M.com~~	~~com~~
A3	CM	I.com	com

Publication

pubid	year	venue
~~P1~~	~~2001~~	~~SIGMOD~~
P2	2011	VLDB
~~P3~~	~~2001~~	~~SIGMOD~~

Authored

id	pubid
A1	P1
~~A2~~	~~P1~~
A1	P2
A3	P2
~~A2~~	~~P3~~
A3	P3

Intervention Δ_ϕ :
Tuples T_0 that satisfy ϕ + Tuples reachable from T_0

Intervention through Causal Paths

Candidate explanation predicate **ϕ : [name = 'RR']**

Forward
Reverse

Author

id	name	inst	dom
A1	JG	C.edu	edu
~~A2~~	~~RR~~	~~M.com~~	~~com~~
A3	CM	I.com	com

Publication

pubid	year	venue
~~P1~~	~~2001~~	~~SIGMOD~~
P2	2011	VLDB
~~P3~~	~~2001~~	~~SIGMOD~~

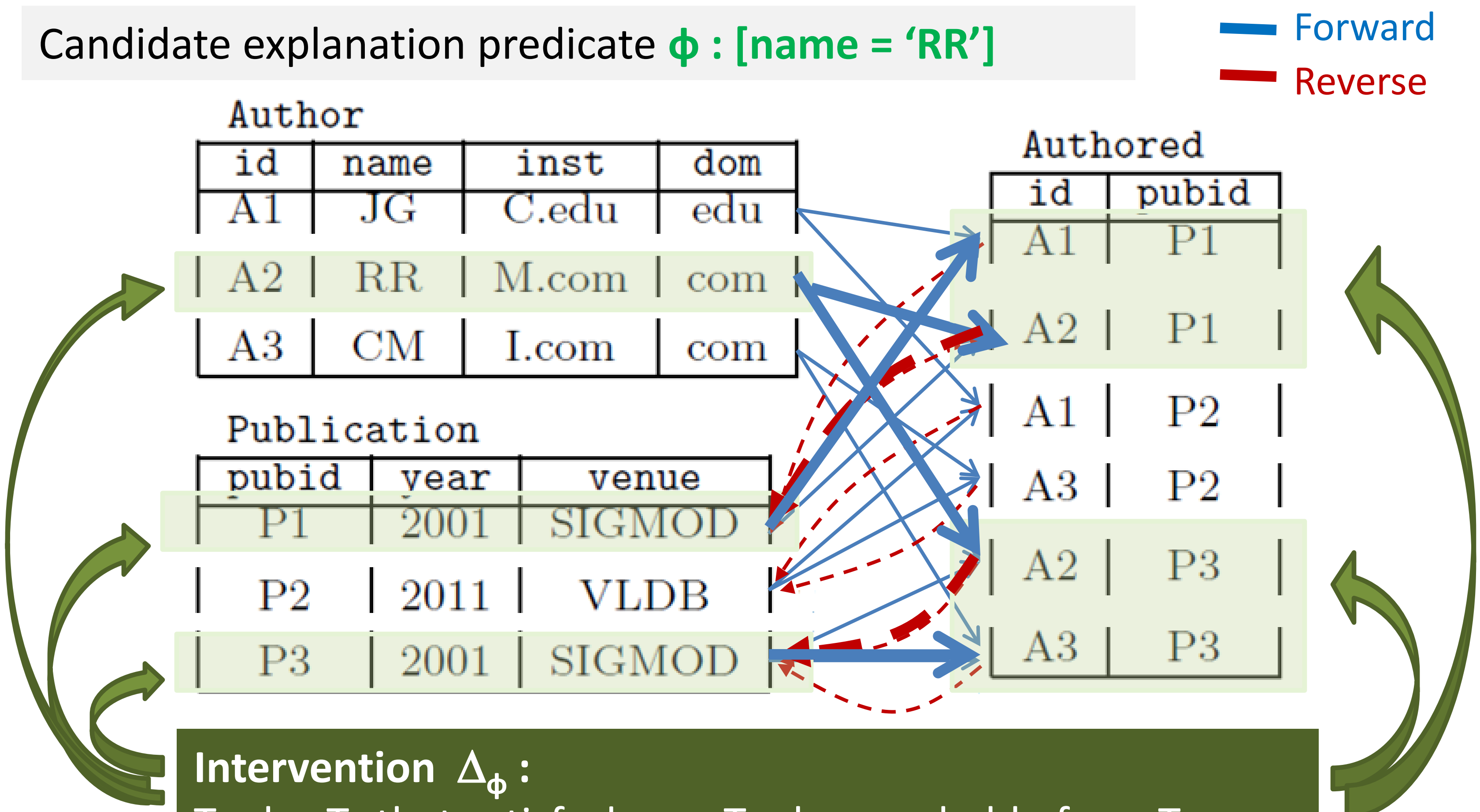
Authored

id	pubid
~~A1~~	~~P1~~
~~A2~~	~~P1~~
A1	P2
A3	P2
~~A2~~	~~P3~~
~~A3~~	~~P3~~

Intervention Δ_ϕ :
Tuples T_0 that satisfy ϕ + Tuples reachable from T_0

Intervention through Causal Paths

Candidate explanation predicate **ϕ : [name = 'RR']**



Author

id	name	inst	dom
A1	JG	C.edu	edu
A2	RR	M.com	com
A3	CM	I.com	com

Publication

pubid	year	venue
P1	2001	SIGMOD
P2	2011	VLDB
P3	2001	SIGMOD

Authored

id	pubid
A1	P1
A2	P1
A1	P2
A3	P2
A2	P3
A3	P3

Intervention Δ_ϕ :
Tuples T_0 that satisfy ϕ + Tuples reachable from T_0

[Roy-Suciu, 2014]

Intervention through Causal Paths

Candidate explanation predicate **ϕ : [name = 'RR']**

Forward

Reverse

Author

id	name	inst	dom
A1	JG	C.edu	edu
A2	RR	M.com	com
A3	CM	I.com	com

Authored

id	pubid
A1	P1
A2	P1
A1	P2
A3	P2
A2	P3
A3	P3

Publication

pubid	year	venue
P1	2001	SIGMOD
P2	2011	VLDB
P3	2001	SIGMOD

Predicates on multiple tables require **universal relation**

Intervention Δ_ϕ :

Tuples T_0 that satisfy ϕ + Tuples reachable from T_0

[Roy-Suciu, 2014]

Intervention through Causal Paths

Candidate explanation predicate **ϕ : [name = 'RR']**

Forward

Reverse

Author

id	name	inst	dom
A1	JG	C.edu	edu
A2	RR	M.com	com
A3	CM	I.com	com

Publication

pubid	year	venue
P1	2001	SIGMOD
P2	2011	VLDB
P3	2001	SIGMOD

Authored

id	pubid
A1	P1
A2	P1
A1	P2
A3	P2
A2	P3
A3	P3

Predicates on multiple tables require **universal relation**

Intervention Δ_ϕ :
Tuples T_0 that satisfy ϕ + Tuples reachable from T_0

Given ϕ, computation of Δ_ϕ requires a recursive query

[Roy-Suciu, 2014]

Two sources of complexity

1. Huge search space of predicates (standard)
2. For any such predicate, run a recursive query to compute intervention (new)
   - The recursive query is poly-time, but still not good enough

[Roy-Suciu, 2014]

Two sources of complexity

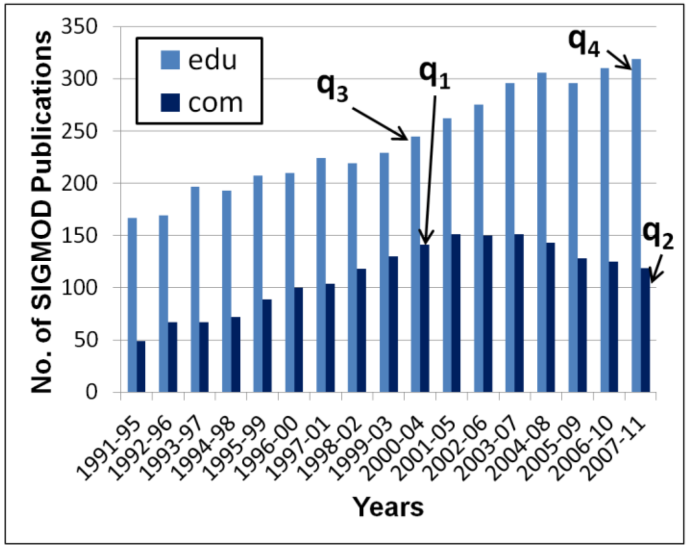
1. Huge search space of predicates (standard)
2. For any such predicate, run a recursive query to compute intervention (new)
   – The recursive query is poly-time, but still not good enough

- Data-cube-based bottom-up algorithm to address both challenges
  – Matches the semantic of recursive query for certain inputs, heuristic for others (open problem: efficient algorithm that matches the semantic for all inputs)

[Roy-Suciu, 2014]

Qualitative Evaluation (DBLP)

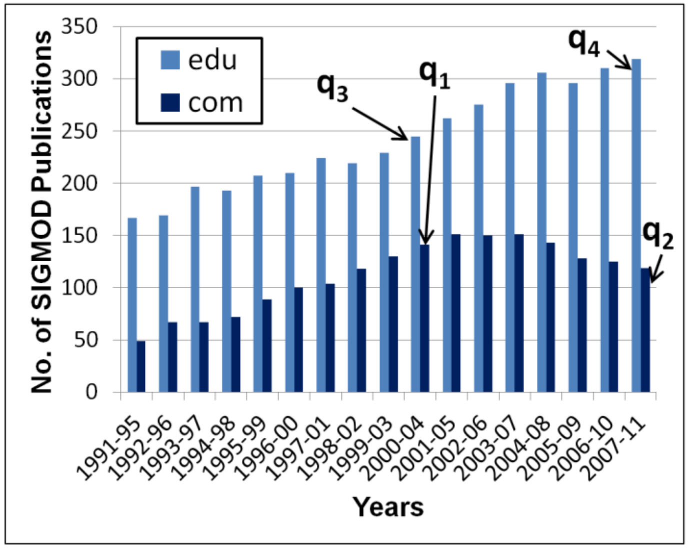
Hard due to lack of gold standard



Q. Why is there a peak for #sigmod papers from industry during 2000-06, while #academia papers kept increasing?

[Roy-Suciu, 2014]

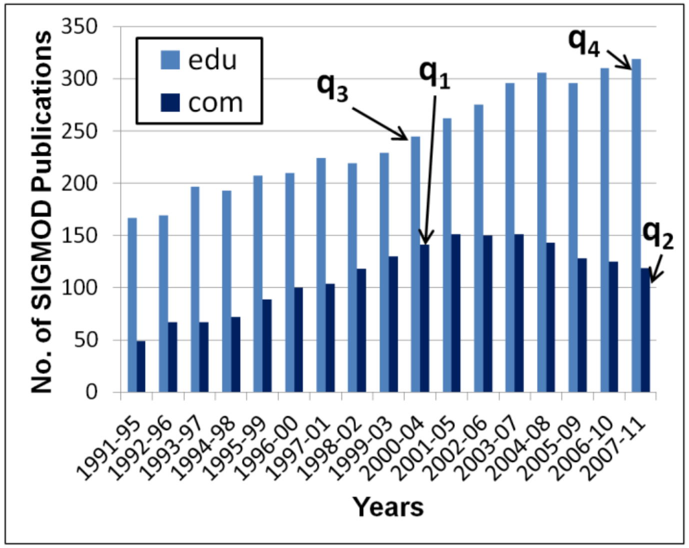
Qualitative Evaluation (DBLP)



rank	explanation (predicates)
1	[affiliation = ibm.com]
2	[affiliation = bell-labs.com]
3	[author = Rajeev Rastogi]
4	[affiliation = ucla.edu]
5	[author = Hamid Pirahesh]
6	[affiliation = asu.edu]
7	[author = Rakesh Agrawal]
8	[affiliation = utah.edu]
9	[affiliation = gwu.edu]

Q. Why is there a peak for #sigmod papers from industry during 2000-06, while #academia papers kept increasing?

Qualitative Evaluation (DBLP)



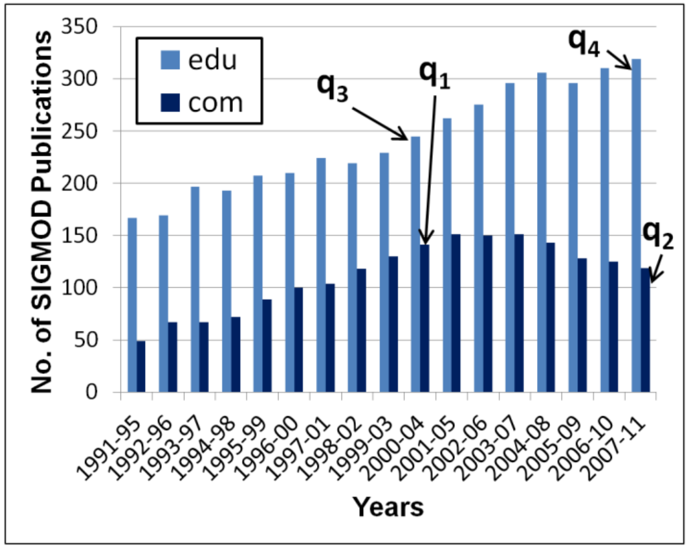
rank	explanation (predicates)
	[affiliation = ibm.com]
	[affiliation = bell-labs.com]
	[author = Rajeev Rastogi]
4	[affiliation = ucla.edu]
	[author = Hamid Pirahesh]
6	[affiliation = asu.edu]
	[author = Rakesh Agrawal]
8	[affiliation = utah.edu]
9	[affiliation = gwu.edu]

Q. Why is there a peak for #sigmod papers from industry during 2000-06, while #academia papers kept increasing?

Intuition:

1. If we remove these industrial labs and their senior researchers, the peak during 2000-04 is more flattened

Qualitative Evaluation (DBLP)



rank	explanation (predicates)
1	[affiliation = ibm.com]
2	[affiliation = bell-labs.com]
3	[author = Rajeev Rastogi]
[redacted]	[affiliation = ucla.edu]
5	[author = Hamid Pirahesh]
[redacted]	[affiliation = asu.edu]
7	[author = Rakesh Agrawal]
[redacted]	[affiliation = utah.edu]
[redacted]	[affiliation = gwu.edu]

Q. Why is there a peak for #sigmod papers from industry during 2000-06, while #academia papers kept increasing?

Intuition:

1. If we remove these industrial labs and their senior researchers, the peak during 2000-04 is more flattened

2. If we remove these universities with relatively new but highly prolific db groups, the curve for academia is less increasing

Summary: Explanations for DB

In general, follow these steps:

Summary: Explanations for DB

In general, follow these steps:

- **Define explanation**
  - Simple predicates, complex predicates with aggregates, comparison operators, …

Summary: Explanations for DB

In general, follow these steps:

- **Define explanation**
  - Simple predicates, complex predicates with aggregates, comparison operators, …
- **Define additional causal paths in the data** (if any)
  - Independent of query/user question

Summary: Explanations for DB

In general, follow these steps:

- **Define explanation**
  - Simple predicates, complex predicates with aggregates, comparison operators, …
- **Define additional causal paths in the data** (if any)
  - Independent of query/user question
- **Define intervention**
  - Delete tuples
  - Insert/update tuples (future direction)
  - Propagate through causal paths

Summary: Explanations for DB

In general, follow these steps:

- **Define explanation**
  - Simple predicates, complex predicates with aggregates, comparison operators, …
- **Define additional causal paths in the data** (if any)
  - Independent of query/user question
- **Define intervention**
  - Delete tuples
  - Insert/update tuples (future direction)
  - Propagate through causal paths
- **Define a scoring function**
  - to rank the explanations based on their intervention
- **Find top-k explanations efficiently**

Part 2.b

• APPLICATION-SPECIFIC DB EXPLANATIONS

Application-Specific Explanations

1. Map-Reduce
2. Probabilistic Databases
3. Security
4. User Rating

We will discuss their notions of explanation and skip the details

Disclaimer:

- There are many applications/research papers that address explanations in one form or another; we cover only a few of them as representatives

1. Explanations for Map Reduce Jobs

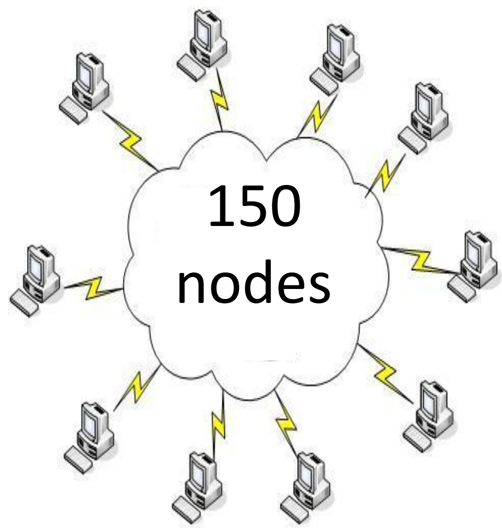
[Khoussainova et al., 2012]

[Khoussainova et al, 2012]

A MapReduce Scenario

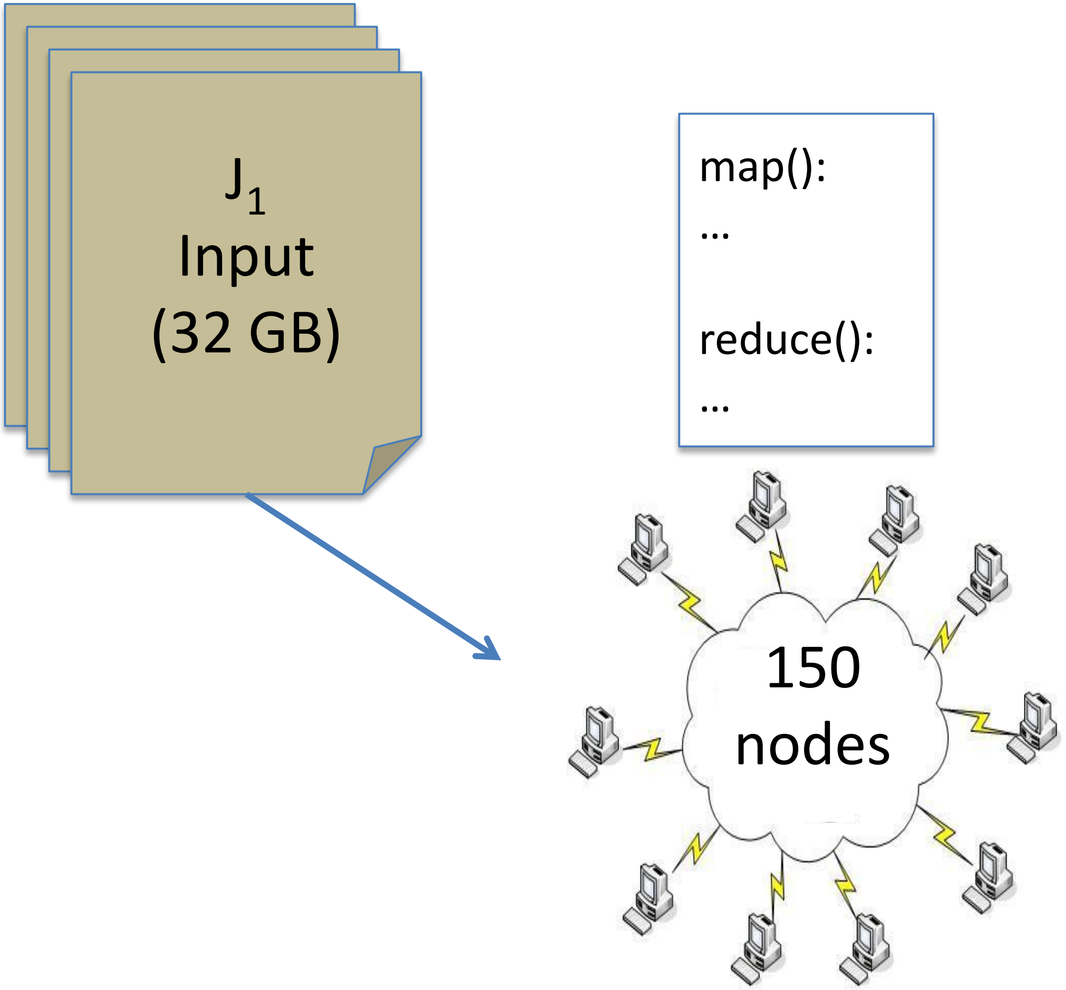
```
map():
…

reduce():
…
```

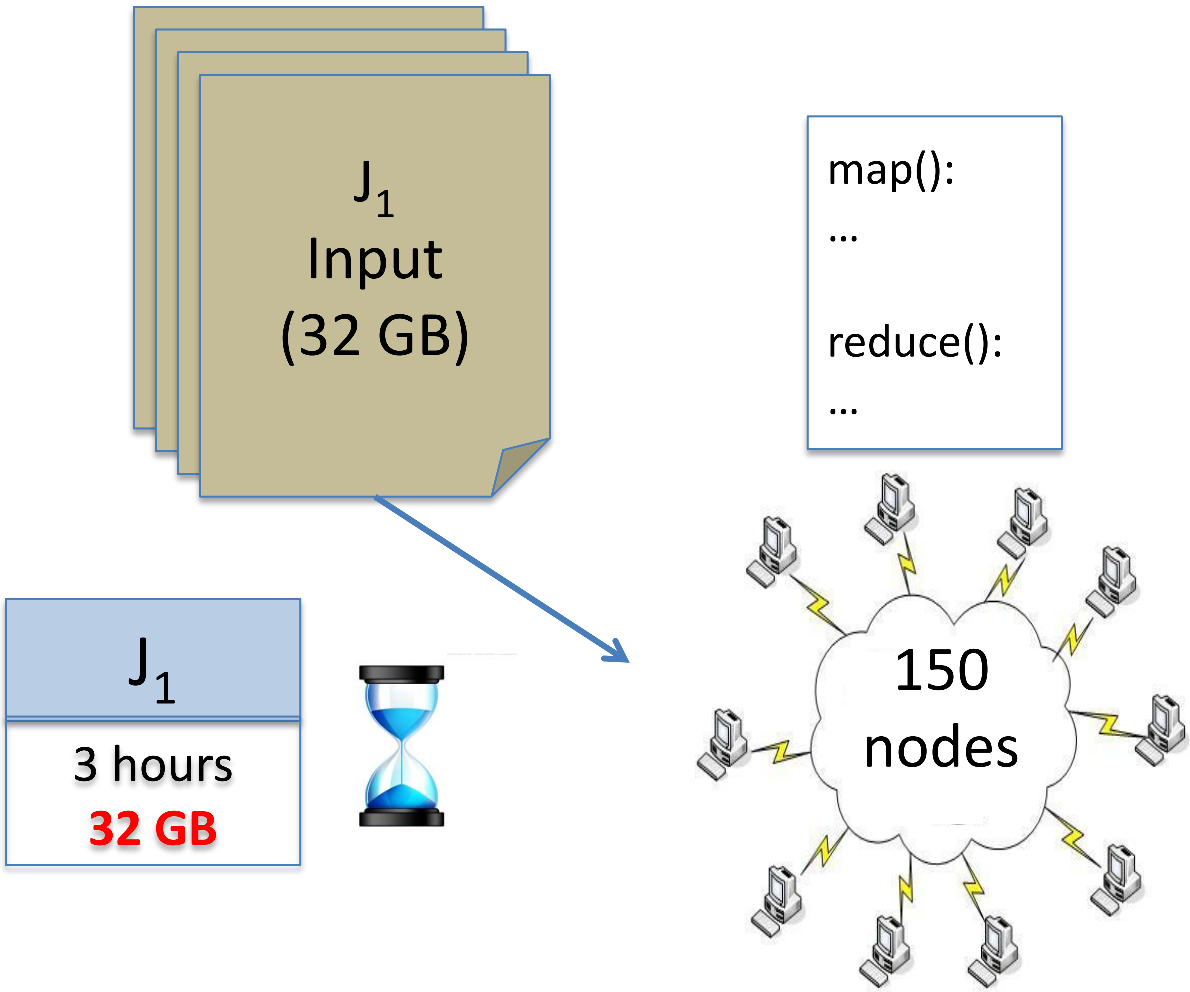


[Khoussainova et al, 2012]

A MapReduce Scenario

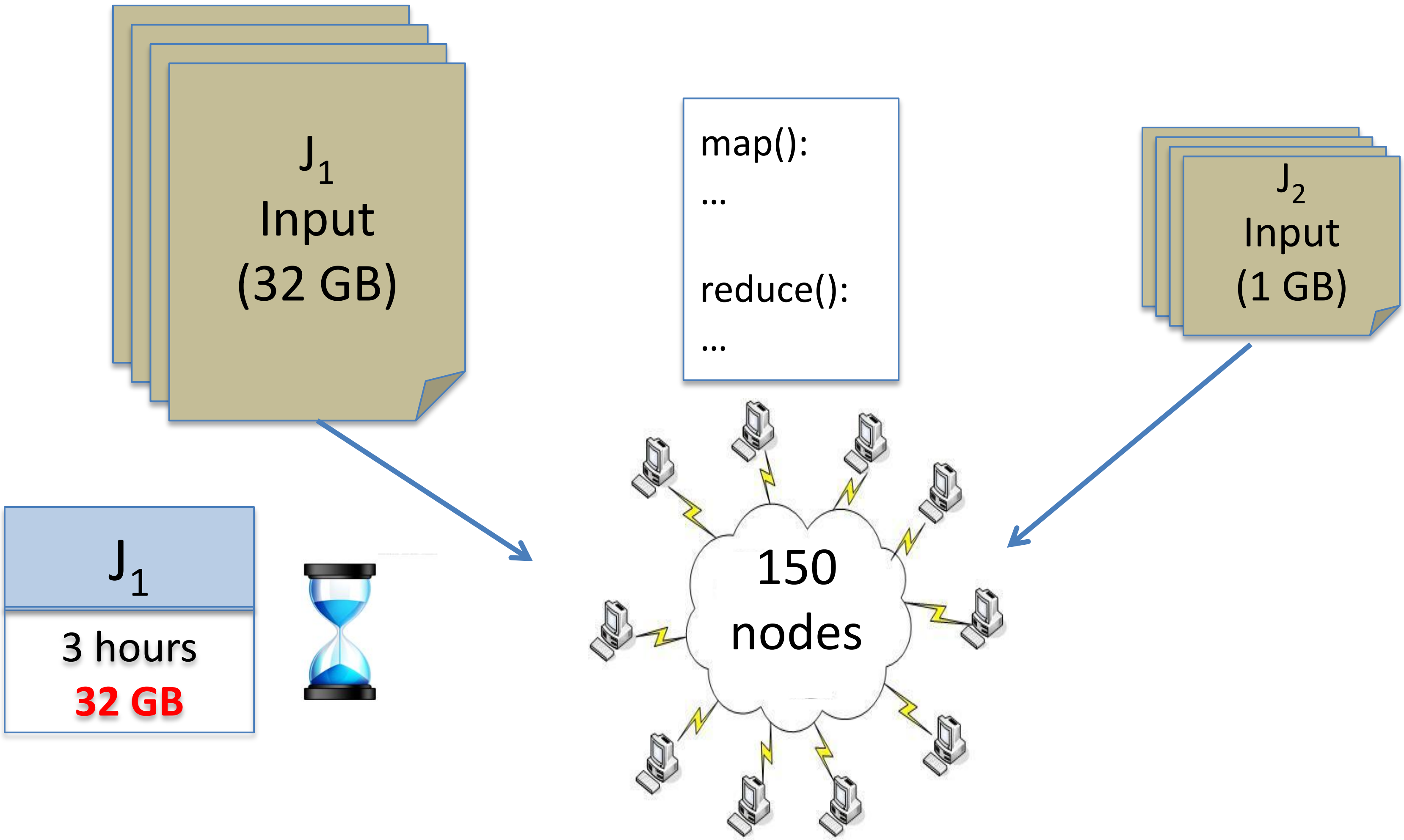


A MapReduce Scenario



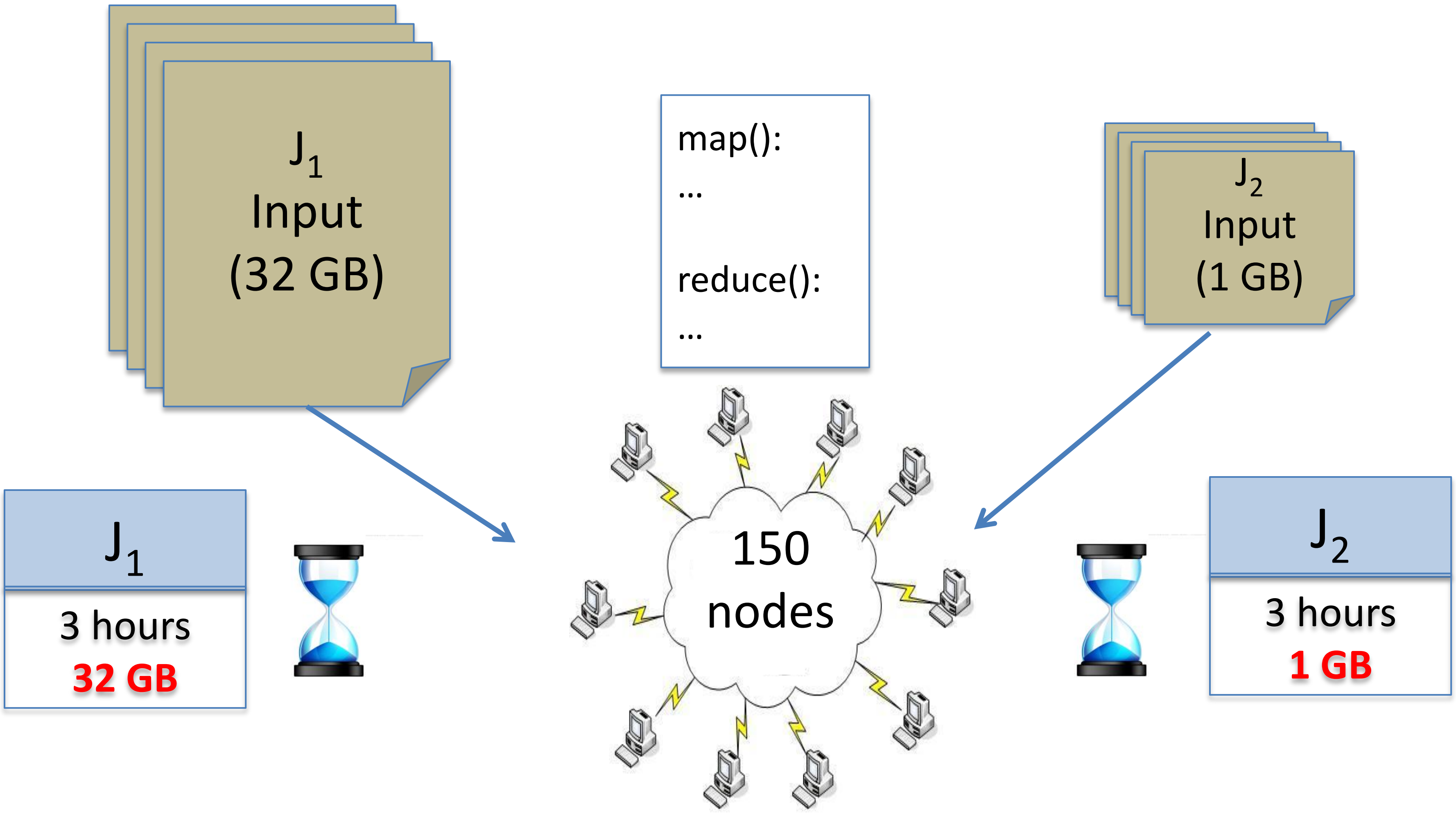
[Khoussainova et al, 2012]

A MapReduce Scenario



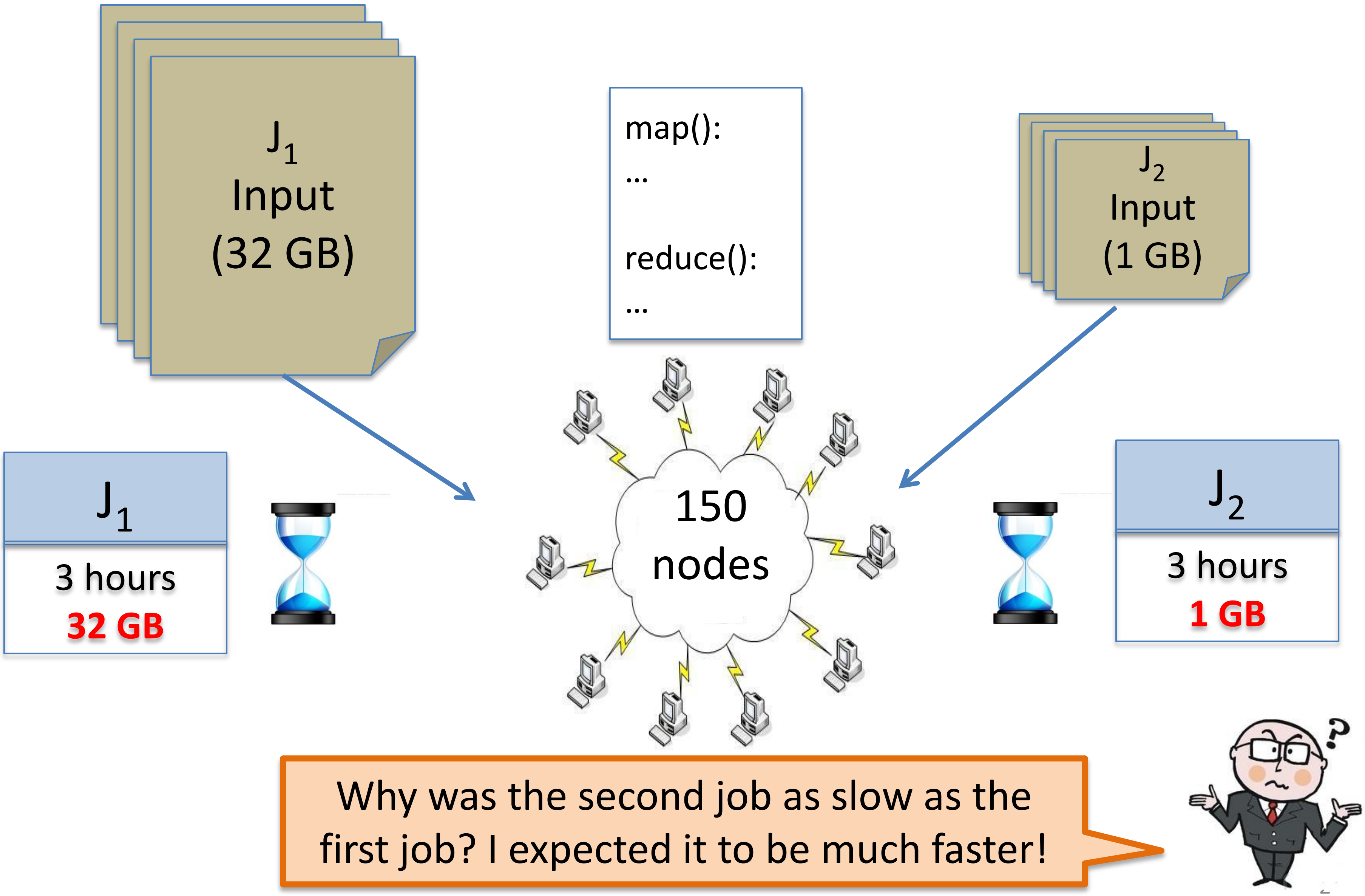
[Khoussainova et al, 2012]

A MapReduce Scenario



[Khoussainova et al, 2012]

A MapReduce Scenario



[Khoussainova et al, 2012]

Explanation by "PerfXPlain"

DFS block size >= 256 MB and #nodes = 150

J_1
3 hours
32 GB

J_2
3 hours
1 GB

Why was the second job as slow as the first job? I expected it to be much faster!

[Khoussainova et al, 2012]

Explanation by “PerfXPlain”

DFS block size >= 256 MB and #nodes = 150

32 GB / 256 MB = 128 blocks.
There are 150 nodes!
Completion time = time to process one block.

J_1
3 hours
32 GB

J_2
3 hours
1 GB

Why was the second job as slow as the first job? I expected it to be much faster!

[Khoussainova et al, 2012]

Explanation by “PerfXPlain”

DFS block size >= 256 MB and #nodes = 150

32 GB / 256 MB = 128 blocks.

There are 150 nodes!

Completion time = time to process one block.

J_1
3 hours
32 GB

=

J_2
3 hours
1 GB

Why was the second job as slow as the first job? I expected it to be much faster!

[Khoussainova et al, 2012]

Explanation by “PerfXPlain”

DFS block size >= 256 MB and #nodes = 150

J_1
3 hours **32 GB**

32 GB / 256 MB = 128 blocks.
There are 150 nodes!
Completion time = time to process one block.

=

1 GB / 256 MB = 4 blocks
Completion time = time to process one block.

J_2
3 hours **1 GB**

Why was the second job as slow as the first job? I expected it to be much faster!

[Khoussainova et al, 2012]

Explanation by "PerfXPlain"

DFS block size >= 256 MB and #nodes = 150

J_1
3 hours
32 GB

32 GB / 256 MB = 128 blocks.
There are 150 nodes!
Completion time = time to process one block.

=

1 GB / 256 MB = 4 blocks
Completion time = time to process one block.

J_2
3 hours
1 GB

PerfXPlain uses a log of past job history and returns predicates on cluster config, job details, load etc. as explanations

2. Explanations for Probabilistic Database

[Kanagal et al, 2012]

Review: Query Evaluation in Prob. DB.

Probability

	AsthmaPatient	
x_1	Ann	0.1
x_2	Bob	0.4

	Friend		
y_1	Ann	Joe	0.9
y_2	Ann	Tom	0.8
y_3	Bob	Tom	0.2

	Smoker	
z_1	Joe	0.3
z_2	Tom	0.7

Probabilistic Database D

Boolean query **Q:** $\exists x \exists y$ **AsthmaPatient(x)** $\wedge$ **Friend (x, y)** $\wedge$ **Smoker(y)**

Review: Query Evaluation in Prob. DB.

Probability

	AsthmaPatient	
x_1	Ann	0.1
x_2	Bob	0.4

	Friend		
y_1	Ann	Joe	0.9
y_2	Ann	Tom	0.8
y_3	Bob	Tom	0.2

	Smoker	
z_1	Joe	0.3
z_2	Tom	0.7

Probabilistic Database D

Boolean query **Q:** $\exists x \exists y$ **AsthmaPatient(x)** $\wedge$ **Friend (x, y)** $\wedge$ **Smoker(y)**

- Q(D) is not simply true/false, has a probability Pr[Q(D)] of being true

Review: Query Evaluation in Prob. DB.

Probability

	AsthmaPatient	
x_1	Ann	0.1
x_2	Bob	0.4

	Friend		
y_1	Ann	Joe	0.9
y_2	Ann	Tom	0.8
y_3	Bob	Tom	0.2

	Smoker	
z_1	Joe	0.3
z_2	Tom	0.7

Probabilistic Database D

Boolean query **Q:** $\exists x \exists y$ **AsthmaPatient(x)** $\wedge$ **Friend (x, y)** $\wedge$ **Smoker(y)**

- Q(D) is not simply true/false, has a probability Pr[Q(D)] of being true

Lineage: $F_{Q,D} = (x_1 \wedge y_1 \wedge z_1) \vee (x_1 \wedge y_2 \wedge z_2) \vee (x_2 \wedge y_3 \wedge z_2)$

- **Q** is true on **D** $\Leftrightarrow$ $F_{Q,D}$ is true

$$\Pr[F_{Q,D}] = \Pr[Q(D)]$$

[Kanagal et al, 2012]

Explanations for Prob. DB.

Explanation for Q(D) of size k:

- A set S of tuples in D, |S| = k, such that Pr[Q(D)] changes the most when we set the probabilities of all tuples in S to 0
  - i.e. when tuples in S are deleted (intervention)

[Kanagal et al, 2012]

Explanations for Prob. DB.

Explanation for Q(D) of size k:

- A set S of tuples in D, |S| = k, such that Pr[Q(D)] changes the most when we set the probabilities of all tuples in S to 0
  - i.e. when tuples in S are deleted (intervention)

Example

Lineage: $(a \wedge b) \vee (c \wedge d)$

Probabilities: Pr[a] = Pr[b] = **0.9**, Pr[c] = Pr[d] = **0.1**

[Kanagal et al, 2012]

Explanations for Prob. DB.

Explanation for Q(D) of size k:

- A set S of tuples in D, |S| = k, such that Pr[Q(D)] changes the most when we set the probabilities of all tuples in S to 0
  - i.e. when tuples in S are deleted (intervention)

Example

Lineage: $(a \wedge b) \vee (c \wedge d)$

Probabilities: Pr[a] = Pr[b] = **0.9**, Pr[c] = Pr[d] = **0.1**

Explanation of size 1: {a} or {b}

[Kanagal et al, 2012]

Explanations for Prob. DB.

Explanation for Q(D) of size k:

- A set S of tuples in D, |S| = k, such that Pr[Q(D)] changes the most when we set the probabilities of all tuples in S to 0
  - i.e. when tuples in S are deleted (intervention)

Example

Lineage: **$(a \wedge b) \vee (c \wedge d)$**

Probabilities: Pr[a] = Pr[b] = **0.9**, Pr[c] = Pr[d] = **0.1**

Explanation of size 1: {a} or {b}

Explanation of size 2:

Any of four combinations {a,b} x {c, d} that makes Pr[Q(D)] = 0 and **NOT** {a, b}

[Kanagal et al, 2012]

Explanations for Prob. DB.

Explanation for Q(D) of size k:

- A set S of tuples in D, |S| = k, such that Pr[Q(D)] changes the most when we set the probabilities of all tuples in S to 0
  - i.e. when tuples in S are deleted (intervention)

Example

NP-hard, but
poly-time for special cases

Lineage: **$(a \wedge b) \vee (c \wedge d)$**

Probabilities: Pr[a] = Pr[b] = **0.9**, Pr[c] = Pr[d] = **0.1**

Explanation of size 1: {a} or {b}

Explanation of size 2:

Any of four combinations {a,b} x {c, d} that makes Pr[Q(D)] = 0 and **NOT** {a, b}

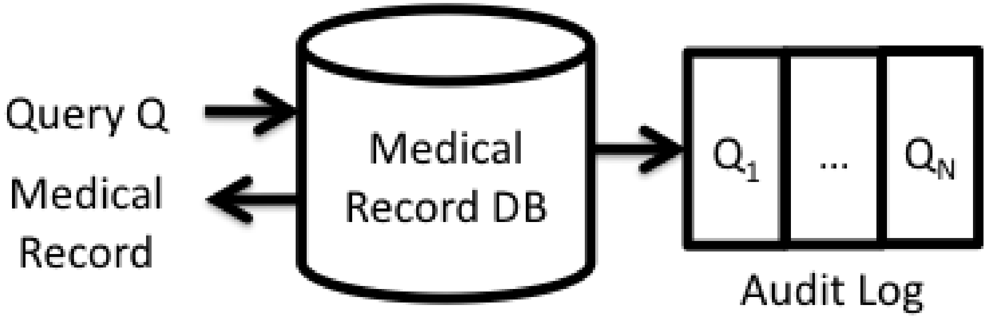
3. Explanations for Security and Access Logs

[Fabbri-LeFevre, 2011]
[Bender et al., 2014]

[Fabbri-LeFevre, 2011]

3a. Medical Record Security

- Security of patient data is immensely important
- Hospitals monitor accesses and construct an audit log
- Large number of accesses, difficult for compliance officers monitor the audit log
- Goal: Improve the auditing system so that it is easier to find inappropriate accesses by "explaining" the reason for access



[Fabbri-LeFevre, 2011]

Explanation by Existence of Paths

Consider this sample audit log and associated database:

Lid	Date	User	Patient
1	1/1/12	Dr. Bob	Alice
2	1/2/12	Dr. Mike	Alice
2	1/3/12	Dr. Evil	Alice

Audit Log

Patient	Date	Doctor
Alice	1/1/12	Dr. Bob

Appointments

Doctor	Department
Dr. Bob	Pediatrics
Dr. Mike	Pediatrics

Departments

[Fabbri-LeFevre, 2011]

Explanation by Existence of Paths

An access is explained if there exists a path:

- From the data accessed (Patient) to the user accessing the data (User)
- Through other tables/tuples stored in the DB

Lid	Date	User	Patient
1	1/1/12	Dr. Bob	Alice
2	1/2/12	Dr. Mike	Alice
2	1/3/12	Dr. Evil	Alice

Audit Log

Patient	Date	Doctor
Alice	1/1/12	Dr. Bob

Appointments

Doctor	Department
Dr. Bob	Pediatrics
Dr. Mike	Pediatrics

Departments

[Fabbri-LeFevre, 2011]

Explanation by Existence of Paths

An access is explained if there exists a path:

- From the data accessed (Patient) to the user accessing the data (User)
- Through other tables/tuples stored in the DB

Lid	Date	User	Patient
1	1/1/12	Dr. Bob	Alice
2	1/2/12	Dr. Mike	Alice
2	1/3/12	Dr. Evil	Alice

Audit Log

Patient	Date	Doctor
Alice	1/1/12	Dr. Bob

Appointments

Doctor	Department
Dr. Bob	Pediatrics
Dr. Mike	Pediatrics

Departments

Why did **Dr. Bob** access **Alice**'s record?

[Fabbri-LeFevre, 2011]

Explanation by Existence of Paths

An access is explained if there exists a path:

- From the data accessed (Patient) to the user accessing the data (User)
- Through other tables/tuples stored in the DB

Lid	Date	User	Patient
1	1/1/12	Dr. Bob	Alice
2	1/2/12	Dr. Mike	Alice
2	1/3/12	Dr. Evil	Alice

Audit Log

Patient	Date	Doctor
Alice	1/1/12	Dr. Bob

Appointments

Doctor	Department
Dr. Bob	Pediatrics
Dr. Mike	Pediatrics

Departments

Because of an appointment

Why did **Dr. Bob** access **Alice**'s record?

[Fabbri-LeFevre, 2011]

Explanation by Existence of Paths

An access is explained if there exists a path:

- From the data accessed (Patient) to the user accessing the data (User)
- Through other tables/tuples stored in the DB

Lid	Date	User	Patient
1	1/1/12	Dr. Bob	Alice
2	1/2/12	Dr. Mike	Alice
2	1/3/12	Dr. Evil	Alice

Audit Log

Patient	Date	Doctor
Alice	1/1/12	Dr. Bob

Appointments

Doctor	Department
Dr. Bob	Pediatrics
Dr. Mike	Pediatrics

Departments

Why did **Dr. Mike** access **Alice**'s record?

[Fabbri-LeFevre, 2011]

Explanation by Existence of Paths

An access is explained if there exists a path:

- From the data accessed (Patient) to the user accessing the data (User)
- Through other tables/tuples stored in the DB

Lid	Date	User	Patient
1	1/1/12	Dr. Bob	Alice
2	1/2/12	Dr. Mike	Alice
2	1/3/12	Dr. Evil	Alice

Audit Log

Patient	Date	Doctor
Alice	1/1/12	Dr. Bob

Appointments

Doctor	Department
Dr. Bob	Pediatrics
Dr. Mike	Pediatrics

Departments

Alice had an appointment with Dr. Bob, and Dr. Bob and Dr. Mike are Pediatricians *(same department)*

Why did **Dr. Mike** access **Alice**'s record?

[Fabbri-LeFevre, 2011]

Explanation by Existence of Paths

An access is explained if there exists a path:

- From the data accessed (Patient) to the user accessing the data (User)
- Through other tables/tuples stored in the DB

Lid	Date	User	Patient
1	1/1/12	Dr. Bob	Alice
2	1/2/12	Dr. Mike	Alice
2	1/3/12	**Dr. Evil**	**Alice**

Audit Log

Patient	Date	Doctor
Alice	1/1/12	Dr. Bob

Appointments

Doctor	Department
Dr. Bob	Pediatrics
Dr. Mike	Pediatrics

Departments

Why did **Dr. Evil** access **Alice**'s record?

[Fabbri-LeFevre, 2011]

Explanation by Existence of Paths

An access is explained if there exists a path:

- From the data accessed (Patient) to the user accessing the data (User)
- Through other tables/tuples stored in the DB

Lid	Date	User	Patient
1	1/1/12	Dr. Bob	Alice
2	1/2/12	Dr. Mike	Alice
2	1/3/12	**Dr. Evil**	**Alice**

Audit Log

Patient	Date	Doctor
Alice	1/1/12	Dr. Bob

Appointments

Doctor	Department
Dr. Bob	Pediatrics
Dr. Mike	Pediatrics

Departments

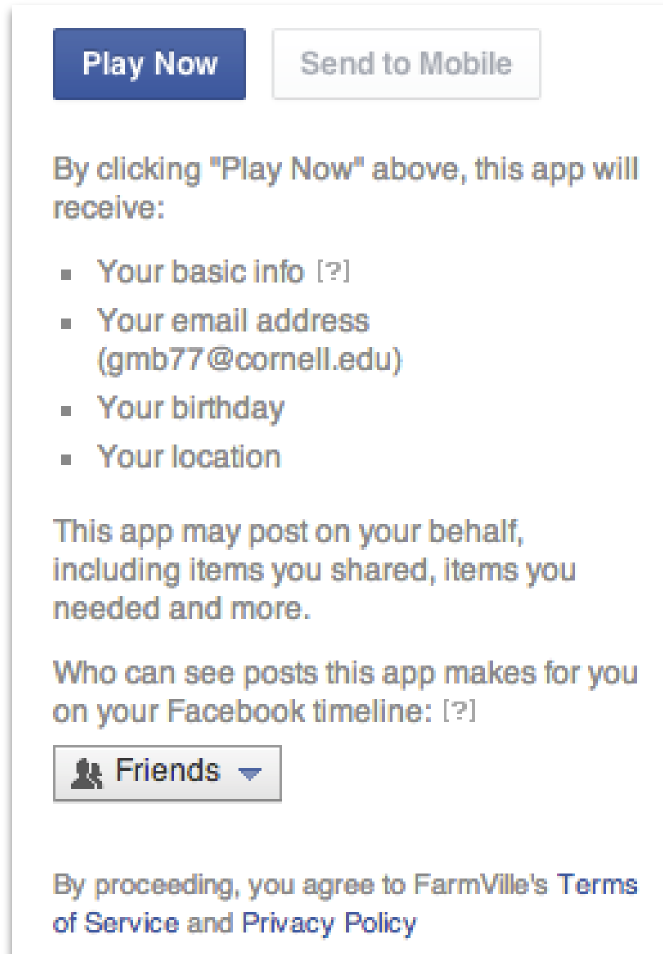
No path exists,
suspicious access!!

Why did **Dr. Evil** access **Alice**'s record?

[Bender et al., 2014]

3b. Explainable security permissions

- Access policies for social media/smartphone apps can be complex and fine-grained
- Difficult to comprehend for application developers
- Explain “NO ACCESS” decisions by what permissions are needed for access



[Bender et al., 2014]

Example: Base Table

User

uid	name	email
4	Zuck	zuck@fb.com
10	Marcel	marcel@fb.com
12347	Lucja	lucja@cornell.edu

[Bender et al., 2014]

Example: Security Views

```
CREATE VIEW V1 AS
SELECT * FROM User
 WHERE uid = 4

CREATE VIEW V2 AS
SELECT uid, name
  FROM User

CREATE VIEW V3 AS
SELECT name, email
  FROM User
```

User

uid	name	email
4	Zuck	zuck@fb.com
10	Marcel	marcel@fb.com
12347	Lucja	lucja@cornell.edu

[Bender et al., 2014]

Example: Security Views

```
CREATE VIEW V1 AS
SELECT * FROM User
 WHERE uid = 4
```

```
CREATE VIEW V2 AS
SELECT uid, name
  FROM User
```

```
CREATE VIEW V3 AS
SELECT name, email
  FROM User
```

User

uid	name	email
4	Zuck	zuck@fb.com
10	Marcel	marcel@fb.com
12347	Lucja	lucja@cornell.edu

[Bender et al., 2014]

Example: Security Views

```
CREATE VIEW V1 AS
SELECT * FROM User
 WHERE uid = 4
```

```
CREATE VIEW V2 AS
SELECT uid, name
  FROM User
```

```
CREATE VIEW V3 AS
SELECT name, email
  FROM User
```

User

uid	name	email
4	Zuck	zuck@fb.com
10	Marcel	marcel@fb.com
12347	Lucja	lucja@cornell.edu

[Bender et al., 2014]

Example: Security Views

```
CREATE VIEW V1 AS
SELECT * FROM User
 WHERE uid = 4

CREATE VIEW V2 AS
SELECT uid, name
  FROM User

CREATE VIEW V3 AS
SELECT name, email
  FROM User
```

User

uid	name	email
4	Zuck	zuck@fb.com
10	Marcel	marcel@fb.com
12347	Lucja	lucja@cornell.edu

[Bender et al., 2014]

Example: Security Policy

✓
```
CREATE VIEW V1 AS
SELECT * FROM User
 WHERE uid = 4
```

✗
```
CREATE VIEW V2 AS
SELECT uid, name
  FROM User
```

✓
```
CREATE VIEW V3 AS
SELECT name, email
  FROM User
```

User

uid	name	email
4	Zuck	zuck@fb.com
10	Marcel	marcel@fb.com
12347	Lucja	lucja@cornell.edu

✓ Permitted

✗ Not Permitted

[Bender et al., 2014]

Example: Security Policy Decisions

✓
```
CREATE VIEW V1 AS
SELECT * FROM User
 WHERE uid = 4
```

✗
```
CREATE VIEW V2 AS
SELECT uid, name
  FROM User
```

✓
```
CREATE VIEW V3 AS
SELECT name, email
  FROM User
```

```
SELECT name
 FROM User
 WHERE uid = 4
```

Query issued by app

User

uid	name	email
4	Zuck	zuck@fb.com
10	Marcel	marcel@fb.com
12347	Lucja	lucja@cornell.edu

✓ Permitted

✗ Not Permitted

[Bender et al., 2014]

Example: Security Policy Decisions

✓
```
CREATE VIEW V1 AS
SELECT * FROM User
 WHERE uid = 4
```

✗
```
CREATE VIEW V2 AS
SELECT uid, name
  FROM User
```

✓
```
CREATE VIEW V3 AS
SELECT name, email
  FROM User
```

User

uid	name	email
4	Zuck	zuck@fb.com
10	Marcel	marcel@fb.com
12347	Lucja	lucja@cornell.edu

✓ Permitted

✗ Not Permitted

✓
```
SELECT name
 FROM User
 WHERE uid = 4
```

Query issued by app

[Bender et al., 2014]

Example: Security Policy Decisions

✗
```
CREATE VIEW V1 AS
SELECT * FROM User
 WHERE uid = 4
```

✗
```
CREATE VIEW V2 AS
SELECT uid, name
  FROM User
```

✓
```
CREATE VIEW V3 AS
SELECT name, email
  FROM User
```

✗
```
SELECT name
 FROM User
 WHERE uid = 4
```

Query issued by app

User

uid	name	email
4	Zuck	zuck@fb.com
10	Marcel	marcel@fb.com
12347	Lucja	lucja@cornell.edu

✓ Permitted

✗ Not Permitted

[Bender et al., 2014]

Example: Why-Not Explanations

✗
```
CREATE VIEW V1 AS
SELECT * FROM User
 WHERE uid = 4
```

✗
```
CREATE VIEW V2 AS
SELECT uid, name
  FROM User
```

✓
```
CREATE VIEW V3 AS
SELECT name, email
  FROM User
```

✗
```
SELECT name
 FROM User
 WHERE uid = 4
```

Query issued by app

V1	V2	V3	Q
✗	✗	✓	✗
✗	✓	✓	✓
✓	✗	✓	✓
✓	✓	✓	✓

[Bender et al., 2014]

Example: Why-Not Explanations

✗
```
CREATE VIEW V1 AS
SELECT * FROM User
 WHERE uid = 4
```

✗
```
CREATE VIEW V2 AS
SELECT uid, name
  FROM User
```

✓
```
CREATE VIEW V3 AS
SELECT name, email
  FROM User
```

✗
```
SELECT name
 FROM User
 WHERE uid = 4
```

Query issued by app

V1	V2	V3	Q
✗	✗	✓	✗
✗	✓	✓	✓
✓	✗	✓	✓
✓	✓	✓	✓

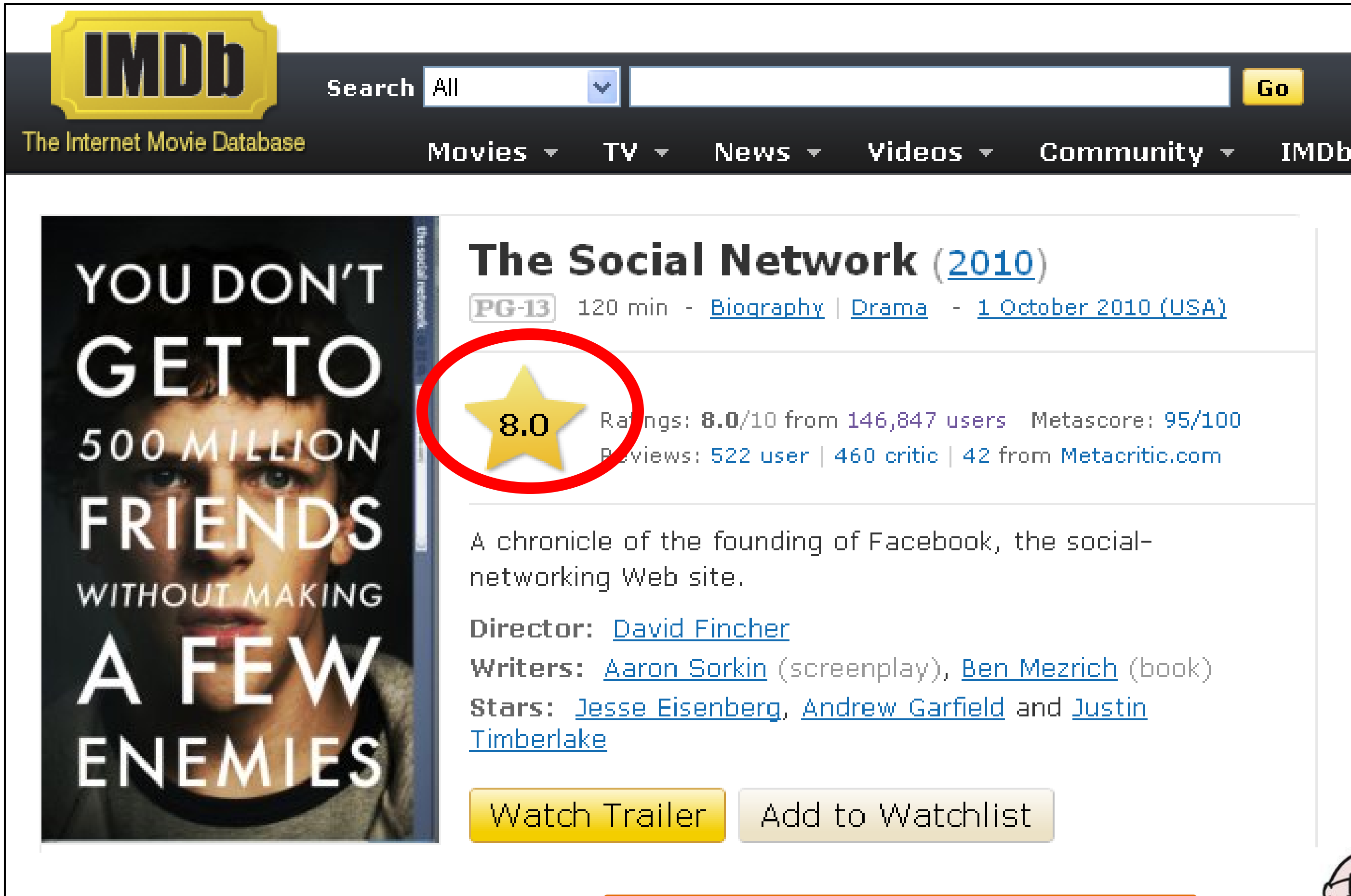
Why-not explanation:
V1 or V2

4. Explanations for User Ratings

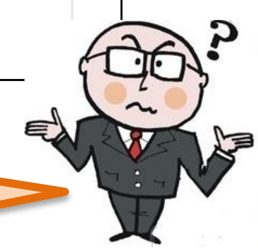
[Das et al., 2012]

[Das et al., 2012]

How to meaningfully explain user rating?

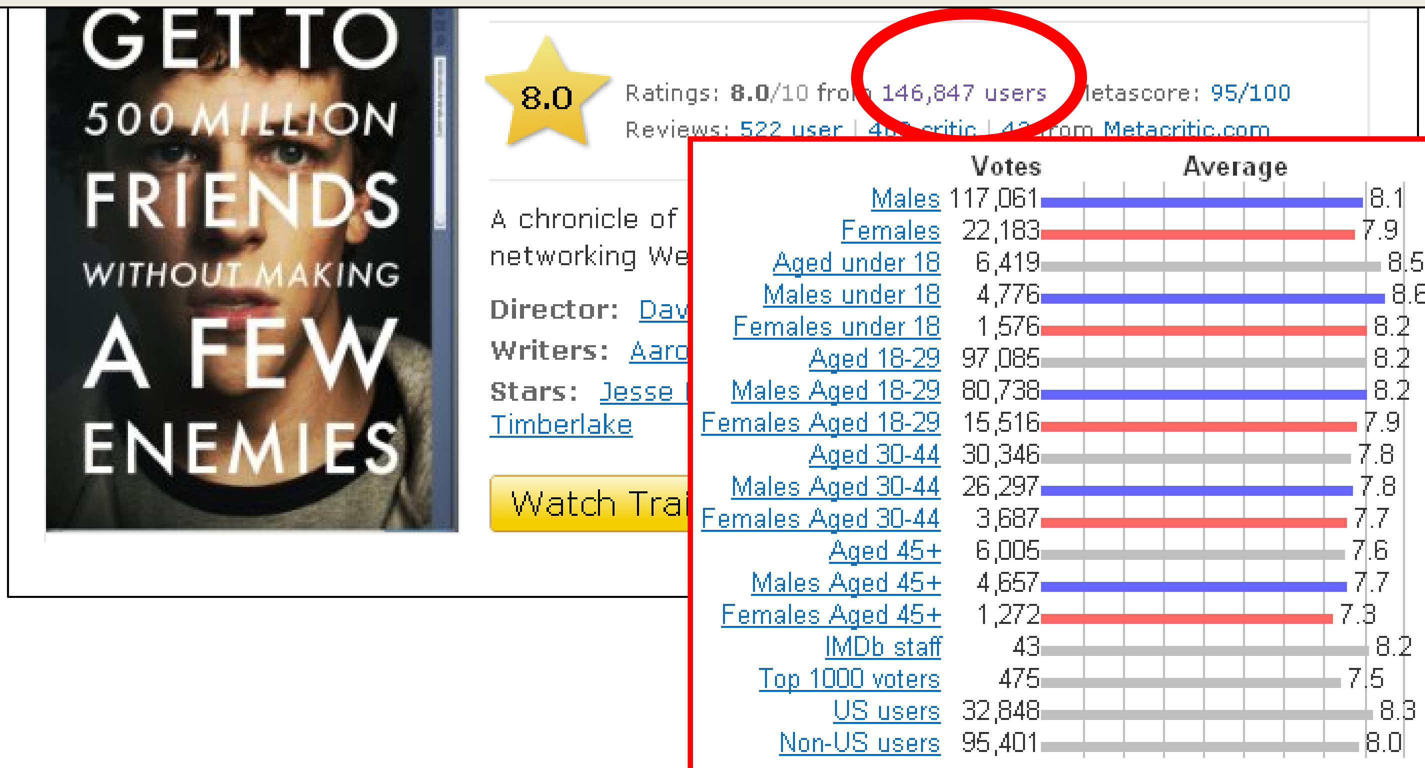


Why is the average rating 8.0?



How to meaningfully explain user rating?

- IMDB provides demographic information of the users, but it is limited
- Need a balance between individual reviews (too many) and final aggregate (less informative)



Meaningful User Rating

- **Solution:**
  Explain ratings by leveraging information about users and item attributes (data cube)

OUTPUT

Black Swan (2010)

R 108 min - Drama | Mystery | Thriller - 17 December 2010 (USA)

8.3 Ratings: **8.3**/10 from 156,148 users Metascore: **79**/100
Reviews: 892 user | 523 critic | 42 from Metacritic.com

Young female reviewers love this movie, average rating: 9.3

Reviewers from New York love this movie, average rating: 8.7

Young male student reviewers hate this movie, average rating: 6.1

Summary

- Causality is fine-grained (actual cause = single tuple), explanations for DB query answers are coarse-grained (explanation = a predicate)
  - There are other application-specific notions of explanations
- Like causality, explanation is defined by intervention

Part 3:

Related Topics
and
Future Directions

Part 3.a:

- **RELATED TOPICS**

Related Topics

- Causality/explanations:
  - how the inputs affect and explain the output(s)

- Other formalisms in databases that capture the connection between inputs and outputs:

1. Provenance/Lineage

2. Deletion Propagation

3. Missing Answers/Why-Not

[Cui et al., 2000] [Buneman et al., 2001] [EDBT 2010 keynote by Val Tannen]
[Green et al., 2007] [Cheney et al., 2009] [Amsterdamer et al. 2011] .....

1. (Boolean) Provenance/Lineage

- Tracks the source tuples that produced an output tuple and how it was produced

R

r1	a1	b1
r2	a1	b2
r3	a2	b2

S

b1	c1	s1
b2	c1	s2
b2	c2	s3

T = R ⋈ S

a1	**c1**	**r1s1 + r2s2**
a1	c2	r2s3
a2	c2	r3s3

- Why/how is T(a1, c1) produced?
- Ans: Either
  by r1 AND s1
  OR
  by r2 AND s2

Provenance vs. Causality/Explanations

- Provenance is a useful tool in finding causality/explanations
  e.g., [Meliou et al., 2010]

Provenance vs. Causality/Explanations

- Provenance is a useful tool in finding causality/explanations
  e.g., [Meliou et al., 2010]
- But, causality/explanations go beyond simple provenance
  - Causality points out the responsibility of each tuple in producing the output that helps ranking input tuples

Provenance vs. Causality/Explanations

- Provenance is a useful tool in finding causality/explanations
  e.g., [Meliou et al., 2010]
- But, causality/explanations go beyond simple provenance
  - Causality points out the responsibility of each tuple in producing the output that helps ranking input tuples
  - Explanations return high-level abstractions as predicates which also help in comparing two or more output aggregate values

Provenance vs. Causality/Explanations

- Provenance is a useful tool in finding causality/explanations
  e.g., [Meliou et al., 2010]
- But, causality/explanations go beyond simple provenance
  – Causality points out the responsibility of each tuple in producing the output that helps ranking input tuples
  – Explanations return high-level abstractions as predicates which also help in comparing two or more output aggregate values

Example
For questions of the form
"Why is avg(temp) at time 12 pm so high?"
"Why is avg(temp) at time 12 pm higher than that at time 11 am?"

Provenance returns individual tuples, whereas a predicate is more informative:
"Sensor = 3"

[Buneman et al. 2002] [Cong et al. 2011] [Kimelfeld et al. 2011]

2. Deletion propagation

- An output tuple is to be deleted
- Delete a set of source tuples to achieve this
- Find a set of source tuples, having minimum side effect in
  - output (view): delete as few other output tuples as possible, or
  - source: delete as few source tuples as possible

[Buneman et al. 2002] [Cong et al. 2011] [Kimelfeld et al. 2011]

Deletion Propagation: View Side Effect

- To delete T(a1, c1)
- Need to delete one of 4 combinations: {r1, s1} x {r2, s2}

R

r1	a1	b1
r2	a1	b2
r3	a2	b2

S

b1	c1	s1
b2	c1	s2
b2	c2	s3

T = R ⋈ S

a1	c1	**r1s1 + r2s2**
a1	c2	r2s3
a2	c2	r3s3

[Buneman et al. 2002] [Cong et al. 2011] [Kimelfeld et al. 2011]

Deletion Propagation: View Side Effect

- To delete T(a1, c1)
- Need to delete one of 4 combinations: {r1, s1} x {r2, s2}

R

r1	a1	b1
r2	a1	b2
r3	a2	b2

S

b1	c1	s1
b2	c1	s2
b2	c2	s3

T = R $\bowtie$ S

a1	c1	**r1s1 + r2s2**
a1	c2	r2s3
a2	c2	r3s3

Delete **{r1, r2}**

[Buneman et al. 2002] [Cong et al. 2011] [Kimelfeld et al. 2011]

Deletion Propagation: View Side Effect

- To delete T(a1, c1)
- Need to delete one of 4 combinations: {r1, s1} x {r2, s2}

R

r1	~~a1~~	~~b1~~
r2	a1	b2
r3	a2	b2

S

b1	c1	s1
b2	c1	s2
b2	c2	s3

T = R ⋈ S

a1	c1	**r1s1 + r2s2**
a1	c2	r2s3
a2	c2	r3s3

Delete **{r1, r2}**

[Buneman et al. 2002] [Cong et al. 2011] [Kimelfeld et al. 2011]

Deletion Propagation: View Side Effect

- To delete T(a1, c1)
- Need to delete one of 4 combinations: {r1, s1} x {r2, s2}

R

r1	~~a1~~	~~b1~~
r2	~~a1~~	~~b2~~
r3	a2	b2

S

b1	c1	s1
b2	c1	s2
b2	c2	s3

T = R ⋈ S

a1	c1	**r1s1 + r2s2**
a1	c2	r2s3
a2	c2	r3s3

Delete **{r1, r2}**

[Buneman et al. 2002] [Cong et al. 2011] [Kimelfeld et al. 2011]

Deletion Propagation: View Side Effect

- To delete T(a1, c1)
- Need to delete one of 4 combinations: {r1, s1} x {r2, s2}

R

r1	~~a1~~	~~b1~~
r2	~~a1~~	~~b2~~
r3	a2	b2

S

b1	c1	s1
b2	c1	s2
b2	c2	s3

T = R ⋈ S

~~a1~~	~~c1~~	**r1s1 + r2s2**
a1	c2	r2s3
a2	c2	r3s3

Delete **{r1, r2}**

[Buneman et al. 2002] [Cong et al. 2011] [Kimelfeld et al. 2011]

Deletion Propagation: View Side Effect

- To delete T(a1, c1)
- Need to delete one of 4 combinations: {r1, s1} x {r2, s2}

R

r1	~~a1~~	~~b1~~
r2	~~a1~~	~~b2~~
r3	a2	b2

S

b1	c1	s1
b2	c1	s2
b2	c2	s3

T =

$R \bowtie S$

~~a1~~	~~c1~~	**r1s1 + r2s2**
~~a1~~	~~c2~~	r2s3
a2	c2	r3s3

Delete **{r1, r2}**

[Buneman et al. 2002] [Cong et al. 2011] [Kimelfeld et al. 2011]

Deletion Propagation: View Side Effect

- To delete T(a1, c1)
- Need to delete one of 4 combinations: {r1, s1} x {r2, s2}

R

r1	~~a1~~	~~b1~~
r2	~~a1~~	~~b2~~
r3	a2	b2

S

b1	c1	s1
b2	c1	s2
b2	c2	s3

T = R ⋈ S

~~a1~~	~~c1~~	**r1s1 + r2s2**
~~a1~~	~~c2~~	r2s3
a2	c2	r3s3

Delete **{r1, r2}**
View Side Effect = 1
as T(a1, c2) is also deleted

[Buneman et al. 2002] [Cong et al. 2011] [Kimelfeld et al. 2011]

Deletion Propagation: View Side Effect

- To delete T(a1, c1)
- Need to delete one of 4 combinations: {r1, s1} x {r2, s2}

R

r1	a1	b1
r2	a1	b2
r3	a2	b2

S

b1	c1	s1
b2	c1	s2
b2	c2	s3

T =
R S

a1	c1	r1s1 + r2s2
a1	c2	r2s3
a2	c2	r3s3

Delete **{r1, s2}**

[Buneman et al. 2002] [Cong et al. 2011] [Kimelfeld et al. 2011]

Deletion Propagation:
View Side Effect

- To delete T(a1, c1)
- Need to delete one of 4 combinations: {r1, s1} x {r2, s2}

R

r1	~~a1~~	~~b1~~
r2	a1	b2
r3	a2	b2

S

b1	c1	s1
b2	c1	s2
b2	c2	s3

T =
R S

a1	c1	r1s1 + r2s2
a1	c2	r2s3
a2	c2	r3s3

Delete **{r1, s2}**

[Buneman et al. 2002] [Cong et al. 2011] [Kimelfeld et al. 2011]

Deletion Propagation: View Side Effect

- To delete T(a1, c1)
- Need to delete one of 4 combinations: {r1, s1} x {r2, s2}

R

r1	~~a1~~	~~b1~~
r2	a1	b2
r3	a2	b2

S

b1	c1	s1
~~b2~~	~~c1~~	s2
b2	c2	s3

T =

R S

a1	c1	r1s1 + r2s2
a1	c2	r2s3
a2	c2	r3s3

Delete **{r1, s2}**

[Buneman et al. 2002] [Cong et al. 2011] [Kimelfeld et al. 2011]

Deletion Propagation: View Side Effect

- To delete T(a1, c1)
- Need to delete one of 4 combinations: {r1, s1} x {r2, s2}

R

r1	~~a1~~	~~b1~~
r2	a1	b2
r3	a2	b2

S

b1	c1	s1
~~b2~~	~~c1~~	s2
b2	c2	s3

T =

R S

~~a1~~	~~c1~~	r1s1 + r2s2
a1	c2	r2s3
a2	c2	r3s3

Delete **{r1, s2}**
View Side Effect = 0
(optimal)

[Buneman et al. 2002] [Cong et al. 2011] [Kimelfeld et al. 2011]

Deletion Propagation: Source Side Effect

- To delete T(a1, c1)
- Need to delete one of 4 combinations: {r1, s1} x {r2, s2}

R

r1	~~a1~~	~~b1~~
r2	a1	b2
r3	a2	b2

S

b1	c1	s1
~~b2~~	~~c1~~	s2
b2	c2	s3

T = $R \bowtie S$

~~a1~~	~~c1~~	r1s1 + r2s2
a1	c2	r2s3
a2	c2	r3s3

Source side effect =
#source tuples to be deleted **= 2**
(**optimal** for any of these four combinations)

Deletion Propagation vs. Causality

- Deletion propagation with source side effects:
  - Minimum set of source tuples to delete that deletes an output tuple
- Causality:
  - Minimum set of source tuples to delete that together with a tuple t deletes an output tuple

- Easy to show that causality is as hard as deletion propagation with source side effect

  (exact relationship is an open problem)

3. Missing Answers/Why-Not

- Aims to explain why a set of tuples does not appear in the query answer

3. Missing Answers/Why-Not

- Aims to explain why a set of tuples does not appear in the query answer

- **Data-based** (explain in terms of database tuples)
  - Insert/update certain input tuples such that the missing tuples appear in the answer
    [Herschel-Hernandez, 2009] [Herschel et al., 2010] [Huang et al., 2008]

3. Missing Answers/Why-Not

- Aims to explain why a set of tuples does not appear in the query answer

- **Data-based** (explain in terms of database tuples)
  - Insert/update certain input tuples such that the missing tuples appear in the answer

    [Herschel-Hernandez, 2009] [Herschel et al., 2010] [Huang et al., 2008]

- **Query-based** (explain in terms of the query issued)
  - Identify the operator in the query plan that is responsible for excluding the missing tuple from the result

    [Chapman-Jagadish, 2009]
  - Generate a refined query whose result includes both the original result tuples as well as the missing tuples

    [Tran-Chan, 2010]

3. Why-Not vs. Causality/Explanations

- In general, why-not approaches use intervention
  - on the database, by inserting/updating tuples
  - or, on the query, by proposing a new query

3. Why-Not vs. Causality/Explanations

- In general, why-not approaches use intervention
  - on the database, by inserting/updating tuples
  - or, on the query, by proposing a new query

- **Future direction:**

  A unified framework for explaining missing tuples or high/low aggregate values using why-not techniques
  - e.g. [Meliou et al., 2010] already handles missing tuples

Other Related Work

- OLAP/Data cube exploration
  e.g. [Sathe-Sarawagi, 2001] [Sarawagi, 2000] [Sarawagi-Sathe, 2000]
  - Get insights about data by exploring along different dimensions

Other Related Work

- OLAP/Data cube exploration
  e.g. [Sathe-Sarawagi, 2001] [Sarawagi, 2000] [Sarawagi-Sathe, 2000]
  – Get insights about data by exploring along different dimensions

- Connections between causality, diagnosis, repairs, and view-updates
  [Bertossi-Salimi, 2014] [Salimi-Bertossi, 2014]

Other Related Work

- OLAP/Data cube exploration
  e.g. [Sathe-Sarawagi, 2001] [Sarawagi, 2000] [Sarawagi-Sathe, 2000]
  - Get insights about data by exploring along different dimensions
- Connections between causality, diagnosis, repairs, and view-updates [Bertossi-Salimi, 2014] [Salimi-Bertossi, 2014]
- Causal inference and learning for computational advertising e.g. [Bottou et al., 2013]
  - Uses causal inference and intervention in controlled experiments for better ad placement in search engines

Other Related Work

- OLAP/Data cube exploration
  e.g. [Sathe-Sarawagi, 2001] [Sarawagi, 2000] [Sarawagi-Sathe, 2000]
  – Get insights about data by exploring along different dimensions

- Connections between causality, diagnosis, repairs, and view-updates [Bertossi-Salimi, 2014] [Salimi-Bertossi, 2014]

- Causal inference and learning for computational advertising e.g. [Bottou et al., 2013]
  – Uses causal inference and intervention in controlled experiments for better ad placement in search engines

- Explanations in AI [Pacer et al., 2013] [Pearl, 1988] [Yuan et al., 2011]
  – Given a set of observed values of variables in a Bayesian network, find a hypothesis (an assignment to other variables) that best explains the observed values

Other Related Work

- OLAP/Data cube exploration
  e.g. [Sathe-Sarawagi, 2001] [Sarawagi, 2000] [Sarawagi-Sathe, 2000]
  – Get insights about data by exploring along different dimensions
- Connections between causality, diagnosis, repairs, and view-updates [Bertossi-Salimi, 2014] [Salimi-Bertossi, 2014]
- Causal inference and learning for computational advertising e.g. [Bottou et al., 2013]
  – Uses causal inference and intervention in controlled experiments for better ad placement in search engines
- Explanations in AI [Pacer et al., 2013] [Pearl, 1988] [Yuan et al., 2011]
  – Given a set of observed values of variables in a Bayesian network, find a hypothesis (an assignment to other variables) that best explains the observed values
- Lamport's causality [Lamport, 1978]
  – to determine the causal order of events in distributed systems

Part 3.b:

- **FUTURE DIRECTIONS**

Extending causality

- Study broader query classes
  - e.g. for aggregate queries, can we define counterfactuals/responsibility in terms of increasing/decreasing the value of an output tuple instead of deleting it totally?

- Analyze causality under the presence of constraints
  - E.g., FDs restrict the lineage expressions that a query can produce. How does this affect complexity?

Refining the definition of cause

- Do we need preemption?
  - Preemption can model intermediate results/views that perhaps cannot be modified
  - Some complexity of the Halpern-Pearl definition may be valuable

- Causality/explanations for queries:
  - Looking for causes/explanations in a query, rather than the data

Find complex explanations efficiently

- Complex explanations
  - Beyond simple predicates,
    e.g. avg(salary) $\geq$ avg(expenditure)

- Efficiently explore the huge search space of predicates
  - Pre-processing/pruning to return explanations in real time

Ranking and Visualization

- Study ranking criteria
  - for simple, general, and diverse explanations
- Visualization and Interactive platform
  - View how the returned explanations affect the original answers
  - Filter out uninteresting explanations

Conclusions

- We need tools to assist users understand “big data”.
  Providing with causality/explanation will be a critical component of these tools

Conclusions

- We need tools to assist users understand “big data”.
  Providing with causality/explanation will be a critical component of these tools
- Causality/explanation is at the intersection of AI, data management, and philosophy

Conclusions

- We need tools to assist users understand “big data”.
  Providing with causality/explanation will be a critical component of these tools
- Causality/explanation is at the intersection of AI, data management, and philosophy
- This tutorial offered a snapshot of current state of the art in causality/explanation in databases; the field is poised to evolve in the near future

Conclusions

- We need tools to assist users understand “big data”. Providing with causality/explanation will be a critical component of these tools
- Causality/explanation is at the intersection of AI, data management, and philosophy
- This tutorial offered a snapshot of current state of the art in causality/explanation in databases; the field is poised to evolve in the near future
- All references are at the end of this tutorial
- The tutorial is available to download from www.cs.umass.edu/~ameli and homes.cs.washington.edu/~sudeepa

Acknowledgements

- Authors of all papers
  - We could not cover many relevant papers due to time limit

- Big thanks to Gabriel Bender, Mahashweta Das, Daniel Fabbri, Nodira Khoussainova, and Eugene Wu for sharing their slides!

- Partially supported by

  NSF Awards IIS-0911036 and CCF-1349784.

References

1. [Bender et al., 2014] G. Bender, L. Kot, J. Gehrke: Explainable security for relational databases. SIGMOD Conference , pages1411-1422, 2014.
2. [Bertossi-Salimi, 2014] L. E. Bertossi, B. Salimi: Unifying Causality, Diagnosis, Repairs and View-Updates in Databases. CoRR abs/1405.4228, 2014.
3. [Bottou et al., 2013] L. Bottou, J. Peters, J. Quiñonero Candela, D. X. Charles, M. Chickering, E. Portugaly, D. Ray, P. Simard, E. Snelson: Counterfactual reasoning and learning systems: the example of computational advertising. Journal of Machine Learning Research 14(1): 3207-3260 , 2013.
4. [Buneman et al., 2001] P. Buneman, S. Khanna, and W. C. Tan: A characterization of data provenance. ICDT, pages 316-330, 2001.
5. [Buneman et al., 2002] P. Buneman, S. Khanna, and W. C. Tan: On propagation of deletions and annotations through views. PODS, pages 150-158, 2002.
6. [Chalamalla et al., 2014] A. Chalamalla, I. F. Ilyas, M. Ouzzani, P. Papotti: Descriptive and prescriptive data cleaning. SIGMOD, pages 445-456, 2014.
7. [Chapman-Jagadish, 2009] A. Chapman, H. V. Jagadish: Why not? SIGMOD, pages 523-534, 2009.
8. [Cheney et al., 2009] J. Cheney, L. Chiticariu, and W. C. Tan: Provenance in databases: Why, how, and where. Foundations and Trends in Databases, 1(4):379-474, 2009.
9. [Chockler-Halpern, 2004] H. Chockler and J. Y. Halpern: Responsibility and blame: A structural-model approach. J. Artif. Intell. Res. (JAIR), 22:93-115, 2004.
10. [Cong et al., 2011] G. Cong, W. Fan, F. Geerts, and J. Luo: On the complexity of view update and its applications to annotation propagation. TKDE, 2011.

References

11. [Cui et al., 2000] Y. Cui, J. Widom, and J. L. Wiener: Tracing the lineage of view data in a warehousing environment. ACM Trans. Database Syst., 25(2):179-227, 2000.
12. [Das et al., 2012] M. Das, S. Amer-Yahia, G. Das, and C. Yu. Mri: Meaningful interpretations of collaborative ratings. PVLDB, 4(11):1063-1074, 2011.
13. [Eiter- Lukasiewicz , 2002] T. Eiter and T. Lukasiewicz. Causes and explanations in the structural-model approach: Tractable cases: UAI, pages 146-153. Morgan Kaufmann, 2002.
14. [Fabbri-LeFevre, 2011] D. Fabbri and K. LeFevre: Explanation-based auditing. Proc. VLDB Endow., 5(1):1-12, Sept. 2011.
15. [Green et al., 2007] T. J. Green, G. Karvounarakis, and V. Tannen: Provenance semirings. PODS, pages 31-40, 2007.
16. [Hagmeyer, 2007] Y. Hagmayer, S. A. Sloman, D. A. Lagnado, and M. R. Waldmann: Causal reasoning through intervention. Causal learning: Psychology, philosophy, and computation, pages 86-100, 2007.
17. [Halpern-Pearl, 2001] J. Y. Halpern and J. Pearl: Causes and explanations: A structural-model approach: Part 1: Causes. UAI, pages 194-202, 2001.
18. [Halpern-Pearl, 2005] J. Y. Halpern and J. Pearl. Causes and explanations: A structural-model approach. Part I: Causes. Brit. J. Phil. Sci., 56:843-887, 2005. (Conference version in UAI, 2001).
19. [Halpern, 2008] J. Y. Halpern. Defaults and Normality in Causal Structures: KR, pages 198-208, 2008
20. [Herschel-Hernandez, 2009] M. Herschel, M. A. Hernandez, and W. C. Tan. Artemis: A system for analyzing missing answers. PVLDB, 2(2):1550-1553, 2009.

References

21. [Herschel et al., 2010] M. Herschel and M. A. Hernandez: Explaining missing answers to SPJUA queries. PVLDB, 3(1):185-196, 2010.
22. [Huang et al., 2008] J. Huang, T. Chen, A. Doan, and J. F. Naughton: On the provenance of non-answers to queries over extracted data. PVLDB, 1(1):736-747, 2008.
23. [Hume, 1748] D. Hume. An enquiry concerning human understanding: Hackett, Indianapolis, IN, 1748.
24. [Kanagal et al, 2012] B. Kanagal, J. Li, and A. Deshpande: Sensitivity analysis and explanations for robust query evaluation in probabilistic databases. SIGMOD, pages 841-852, 2011.
25. [Khoussainova et al., 2012] N. Khoussainova, M. Balazinska, and D. Suciu. Perfxplain: debugging mapreduce job performance. Proc. VLDB Endow., 5(7):598-609, Mar. 2012.
26. [Kimelfeld et al. 2011] B. Kimelfeld, J. Vondrak, and R. Williams: Maximizing conjunctive views in deletion propagation. PODS, pages 187-198, 2011.
27. [Lamport, 1978] L. Lamport. Time, clocks, and the ordering of events in a distributed system: Commun. ACM, 21(7):558-565, July 1978.
28. [Lewis, 1973] D. Lewis. Causation: The Journal of Philosophy, 70(17):556-567, 1973.
29. [Maier et al., 2010] M. E. Maier, B. J. Taylor, H. Oktay, and D. Jensen: Learning causal models of relational domains. AAAI, 2010.
30. [Mayrhofer, 2008] R. Mayrhofer, N. D. Goodman, M. R. Waldmann, and J. B. Tenenbaum: Structured correlation from the causal background. Cognitive Science Society, pages 303-308, 2008.

References

31. [Meliou et al., 2010] A. Meliou, W. Gatterbauer, K. F. Moore, and D. Suciu: The complexity of causality and responsibility for query answers and non-answers. PVLDB, 4(1):34-45, 2010.
32. [Meliou et al., 2010a] A. Meliou, W. Gatterbauer, K. F. Moore, D. Suciu: WHY SO? or WHY NO? Functional Causality for Explaining Query Answers. MUD, pages 3-17, 2010.
33. [Meliou et al., 2011] A. Meliou, W. Gatterbauer, S. Nath, and D. Suciu: Tracing data errors with view-conditioned causality. SIGMOD Conference, pages 505-516, 2011.
34. [Menzies, 2008] P. Menzies. Counterfactual theories of causation: Stanford Encylopedia of Philosophy, 2008.
35. [Pacer et al., 2013] M. Pacer, T. Lombrozo, T. Griths, J. Williams, and X. Chen: Evaluating computational models of explanation using human judgments. UAI, pages 498-507, 2013.
36. [Pearl, 1988] J. Pearl. Probabilistic reasoning in intelligent systems: networks of plausible inference. Morgan Kaufmann Publishers Inc., 1988.
37. [Pearl, 2000] J. Pearl. Causality: models, reasoning, and inference: Cambridge University Press, 2000.
38. [Roy-Suciu, 2014] S. Roy, D. Suciu: A formal approach to finding explanations for database queries: SIGMOD Conference, pages 1579-1590, 2014
39. [Salimi-Bertossi, 2014] Babak Salimi, Leopoldo E. Bertossi: Causality in Databases: The Diagnosis and Repair Connections. CoRR abs/1404.6857, 2014
40. [Sarawagi, 2000] S. Sarawagi: User-Adaptive Exploration of Multidimensional Data: VLDB: pages 307-316, 2000

References

41. [Sarawagi-Sathe, 2000] S. Sarawagi and G. Sathe. i3: Intelligent, interactive investigation of olap data cubes: SIGMOD, 2000.
42. [Sathe-Sarawagi, 2001] G. Sathe, S. Sarawagi: Intelligent Rollups in Multidimensional OLAP Data. VLDB, pages 531-540, 2001
43. [Schaffer, 2000] J. Schaffer: Trumping preemption. The Journal of Philosophy, pages 165-181, 2000
44. [Silverstein et al., 1998] C. Silverstein, S. Brin, R. Motwani, J. D. Ullman: Scalable Techniques for Mining Causal Structures. VLDB: pages 594-605, 1998
45. [Tran-Chan, 2010] Q. T. Tran and C.-Y. Chan: How to conquer why-not questions. SIGMOD, pages 15-26, 2010.
46. [Woodward, 2003] J. Woodward. Making Things Happen: A Theory of Causal Explanation. Oxford scholarship online. Oxford University Press, 2003.
47. [Wu-Madden, 2013] E. Wu and S. Madden. Scorpion: Explaining away outliers in aggregate queries. PVLDB, 6(8), 2013.
48. [Yuan et al., 2011] C. Yuan, H. Lim, and M. L. Littman: Most relevant explanation: computational complexity and approximation methods. Ann. Math. Artif. Intell., 61(3):159{183,2011.

Thank you!

Questions?