Open-World Probabilistic Databases

Guy Van den Broeck

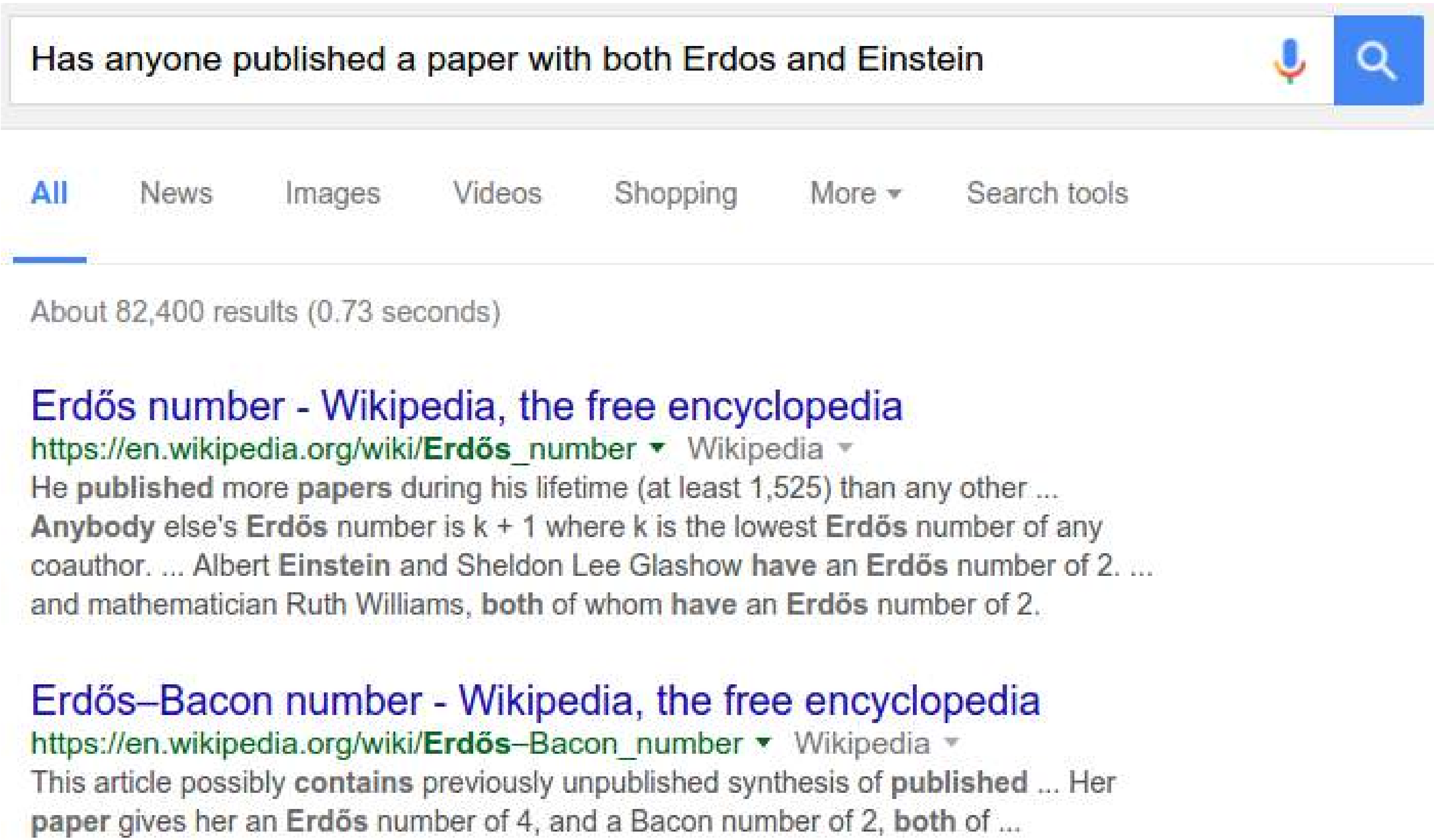
GCAI
Oct 21, 2017

Overview

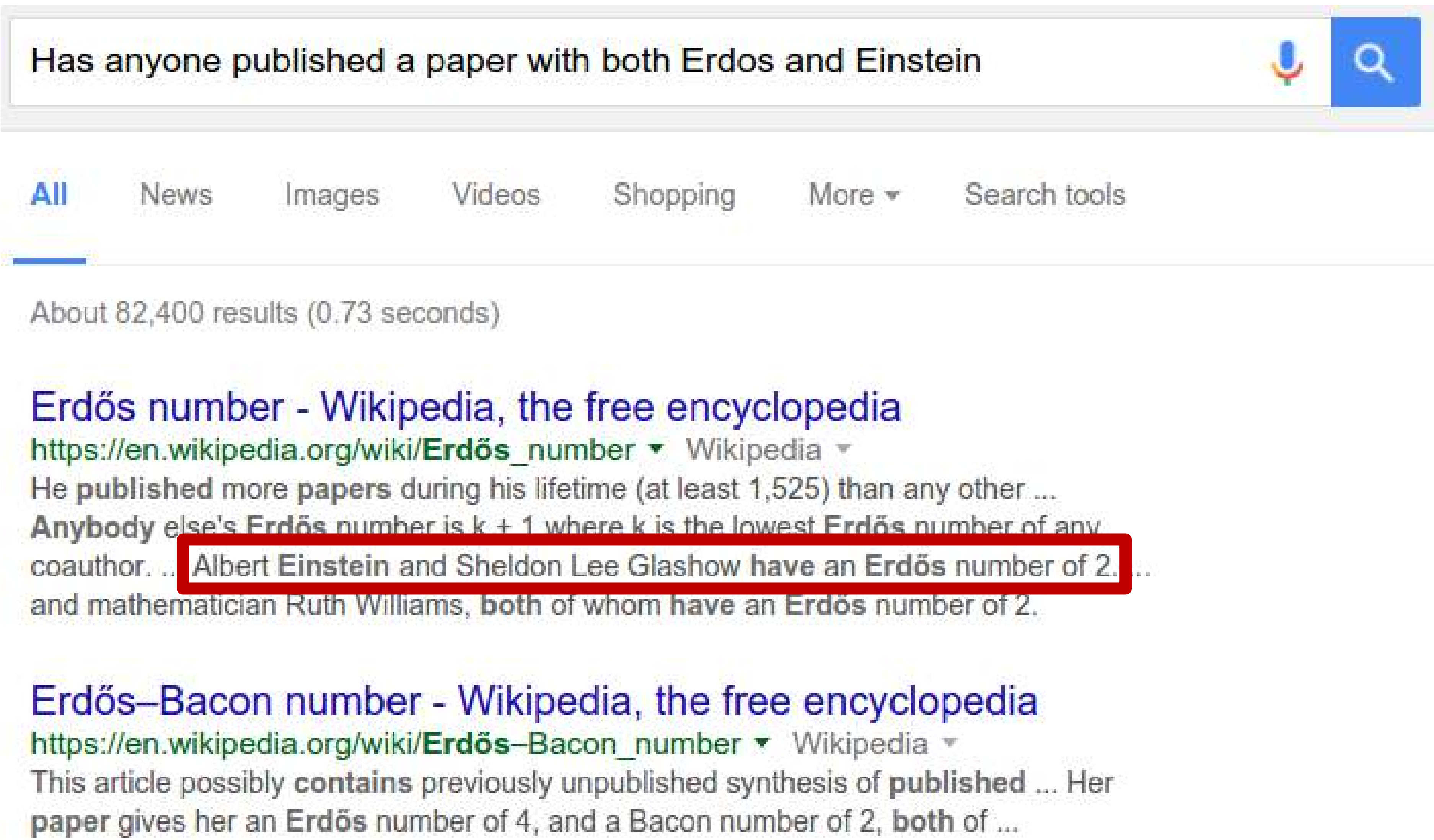
1. ***Why** probabilistic databases?*
2. ***How** probabilistic query evaluation?*
3. ***Why** open world?*
4. ***How** open-world query evaluation?*
5. ***What** is the broader picture?*
   First-order model counting!

Why *probabilistic databases?*

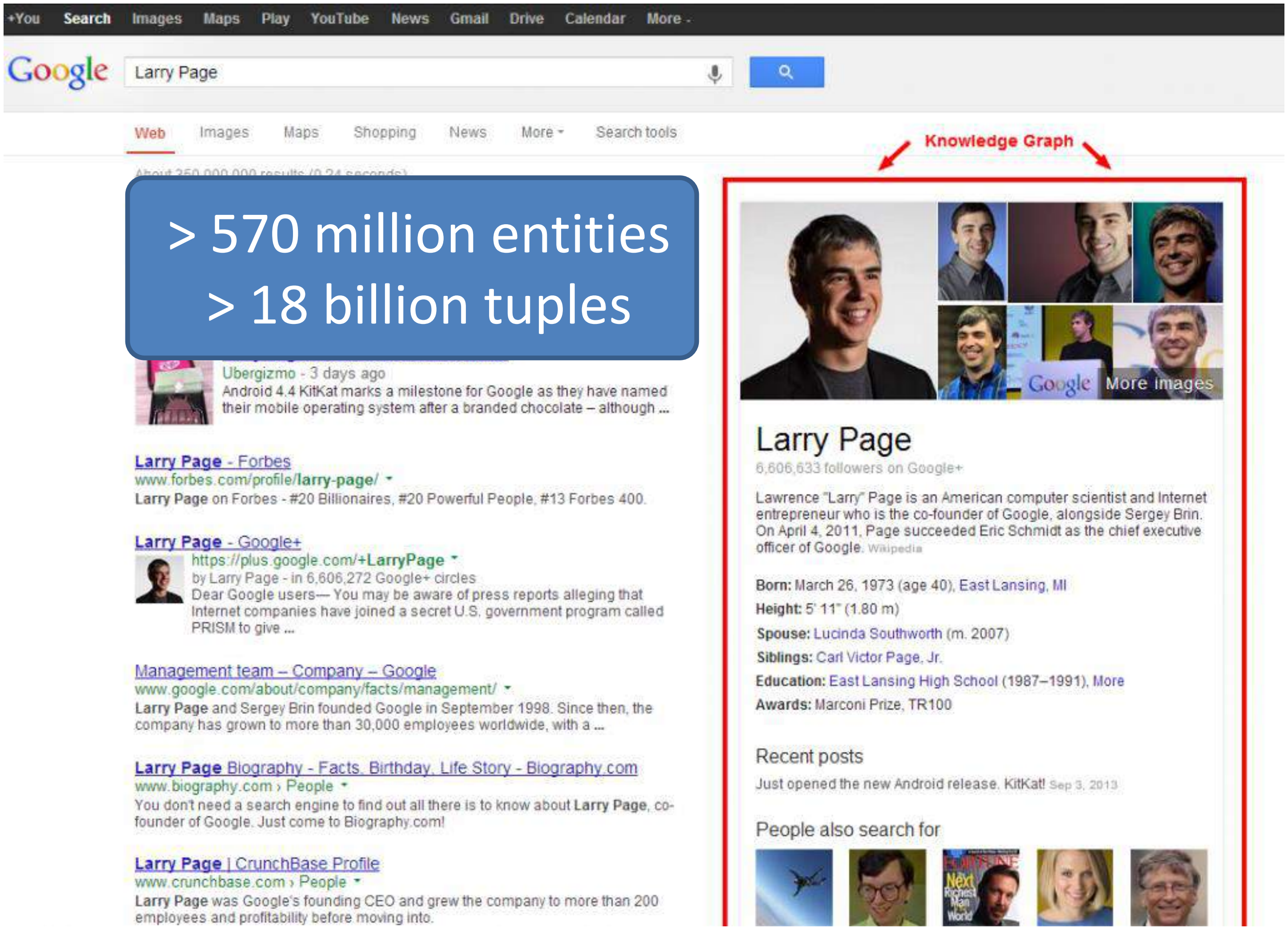
What we'd like to do…



What we’d like to do…



Google Knowledge Graph



> 570 million entities

> 18 billion tuples

Probabilistic Databases

- Tuple-independent probabilistic database

Scientist

x	P
Erdos	0.9
Einstein	0.8
Pauli	0.6

Coauthor

x	y	P
Erdos	Renyi	0.6
Einstein	Pauli	0.7
Obama	Erdos	0.1

- Learned from the web, large text corpora, ontologies, etc., using **statistical** machine learning.

[VdB&Suciu'17]

Information Extraction is Noisy!

PhD Students Luc De Raedt

- Laura-Andrea Antanas(co-promotor Tinne Tuytelaars)
- Dries Van Daele (co-promotor Kathleen Marchal)
- Thanh Le Van (co-promotor Kathleen Marchal)
- Bogdan Moldovan
- Davide Nitti (co-promotor Tinne De Laet)
- José Antonio Oramas Mogrovejo (key supervisor Tinne Tuytelaars)
- Francesco Orsini (co-supervisor Paol Frasconi)
- Sergey Paramonov
- Joris Renkens
- Mathias Verbeke (with Bettina Berendt)
- Jonas Vlasselaer

Coauthor

x	y	P
Luc	Laura	0.7
Luc	Hendrik	0.6
Luc	Kathleen	0.3
Luc	Paol	0.3
Luc	Paolo	0.1

ebay Shop by category | Search... | All Categories | Search | Advanced

Back to home page | Listed in category: Books, Magazines > Non-Fiction Books > See more Probabilistic Inductive Logic Programming by S...

This is a private listing. Sign in to view your status or learn more about private listings.

Probabilistic Inductive Logic Programming De Raedt, Luc (Editor)/ Frasconi, Paol

Item condition: **Brand new**

Time left: 18d 13h (22 Feb. 2016 04:40:52 AEDST)

| Add to watch list

Seller information

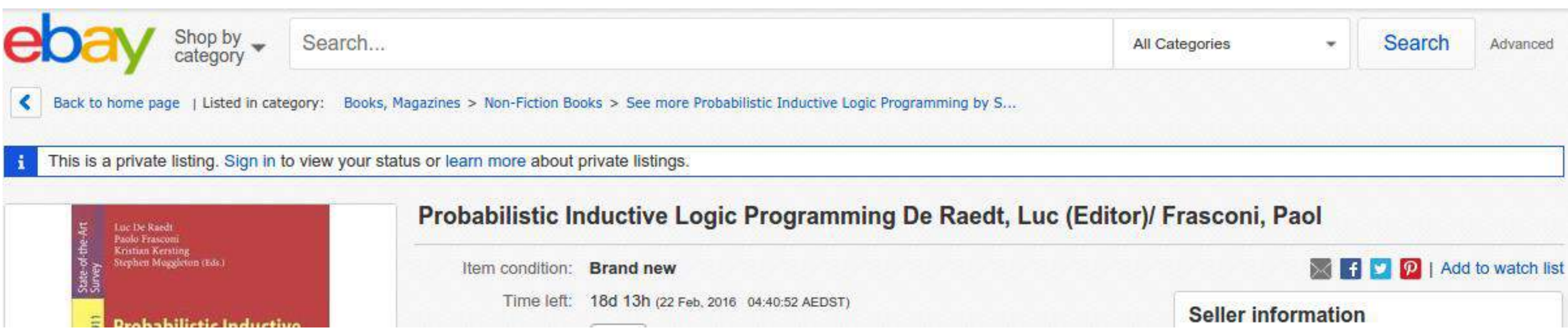
Information Extraction is Noisy!

PhD Students Luc De Raedt

- Laura-Andrea Antanas(co-promotor Tinne Tuytelaars)
- Dries Van Daele (co-promotor Kathleen Marchal)
- Thanh Le Van (co-promotor Kathleen Marchal)
- Bogdan Moldovan
- Davide Nitti (co-promotor Tinne De Laet)
- José Antonio Oramas Mogrovejo (key supervisor Tinne Tuytelaars)
- Francesco Orsini (co-supervisor Paol Frasconi)
- Sergey Paramonov
- Joris Renkens
- Mathias Verbeke (with Bettina Berendt)
- Jonas Vlasselaer

Coauthor

x	y	P
Luc	Laura	0.7
Luc	Hendrik	0.6
Luc	Kathleen	0.3
Luc	Paol	0.3
Luc	Paolo	0.1



Information Extraction is Noisy!

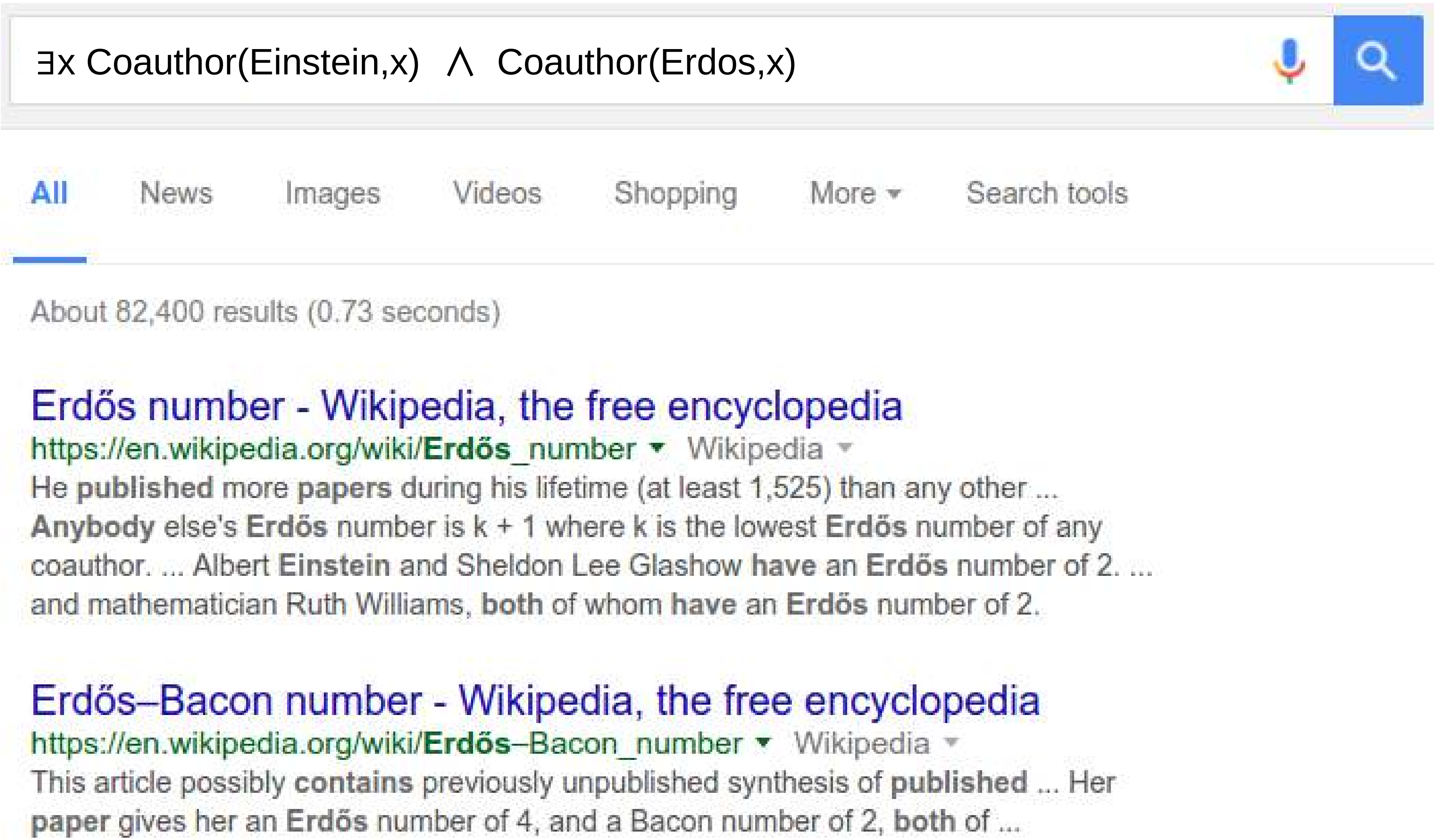
PhD Students Luc De Raedt

- Laura-Andrea Antanas(co-promotor Tinne Tuytelaars)
- Dries Van Daele (co-promotor Kathleen Marchal)
- Thanh Le Van (co-promotor Kathleen Marchal)
- Bogdan Moldovan
- Davide Nitti (co-promotor Tinne De Laet)
- José Antonio Oramas Mogrovejo (key supervisor Tinne Tuytelaars)
- Francesco Orsini (co-supervisor Paol Frasconi)
- Sergey Paramonov
- Joris Renkens
- Mathias Verbeke (with Bettina Berendt)
- Jonas Vlasselaer

Coauthor

x	y	P
Luc	Laura	0.7
Luc	Hendrik	0.6
Luc	Kathleen	0.3
Luc	Paol	0.3
Luc	Paolo	0.1

What we'd like to do…



Einstein is in the Knowledge Graph

Albert Einstein

All News Images Books Videos More ▾ Search tools

About 82,800,000 results (0.45 seconds)

The Official Licensing Site of Albert Einstein
einstein.biz/ ▾
Welcome to the Official Licensing Site of **Albert Einstein**. Learn more about **Albert Einstein** and contact us today for any commercial licensing inquiries.

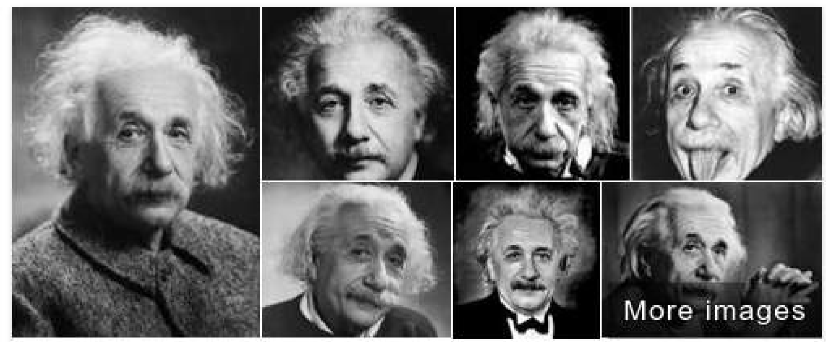
Albert Einstein - Wikipedia, the free encyclopedia
https://en.wikipedia.org/wiki/**Albert_Einstein** ▾ Wikipedia ▾
Albert Einstein (/ˈaɪnstaɪn/; German: [ˈalbɛʁt ˈaɪnʃtaɪn] (listen); 14 March 1879 – 18 April 1955) was a German-born theoretical physicist.
Hans Albert Einstein - Mass–energy equivalence - Eduard Einstein - Elsa Einstein

Albert Einstein (@AlbertEinstein) | Twitter
https://twitter.com/AlbertEinstein

16 hours ago - View on Twitter
ICYMI, Albert Einstein knew a thing or two about being romantic. Learn about the love letters he wrote. guff.com/didnt-know-einst…

20 hours ago - View on Twitter
An interesting read on Einstein's superstar status. What are your thoughts? twitter.com/aeonmag/statu…

Albert Einstein - Biographical - Nobelprize.org
www.nobelprize.org/nobel_prizes/physics/.../**einstein**-bio.htm... ▾ Nobel Prize ▾
Albert Einstein was born at Ulm, in Württemberg, Germany, on March 14, 1879. ...
Later, they moved to Italy and Albert continued his education at Aarau



Albert Einstein

Theoretical Physicist

Albert Einstein was a German-born theoretical physicist. He developed the general theory of relativity, one of the two pillars of modern physics. Einstein's work is also known for its influence on the philosophy of science. Wikipedia

Born: March 14, 1879, Ulm, Germany

Died: April 18, 1955, Princeton, NJ

Influenced by: Isaac Newton, Mahatma Gandhi, More

Children: Eduard Einstein, Lieserl Einstein, Hans Albert Einstein

Spouse: Elsa Einstein (m. 1919–1936), Mileva Marić (m. 1903–1919)

Erdős is in the Knowledge Graph

Paul Erdos

All | Images | Videos | Books | News | More ▾ | Search tools

About 333,000 results (0.35 seconds)

Paul Erdős - Wikipedia, the free encyclopedia
https://en.wikipedia.org/wiki/**Paul_Erdős** ▾ Wikipedia ▾
Paul Erdős was a Hungarian Jewish mathematician. He was one of the most prolific mathematicians of the 20th century. He was known both for his social ...
Fan Chung - Ronald Graham - Béla Bollobás - Category:Paul Erdős

The Man Who Loved Only Numbers - The New York Times
https://www.nytimes.com/books/.../hoffman-man.ht... ▾ The New York Times ▾
Paul Erdős was one of those very special geniuses, the kind who comes along only once in a very long while yet he chose, quite consciously I am sure, to share ...

Paul Erdos | Hungarian mathematician | Britannica.com
www.britannica.com/biography/**Paul-Erdos** ▾ Encyclopaedia Britannica ▾
Paul Erdős, (born March 26, 1913, Budapest, Hungary—died September 20, 1996, Warsaw, Poland), Hungarian “freelance” mathematician (known for his work ...

Paul Erdős - University of St Andrews
www-groups.dcs.st-and.ac.uk/~history/Biographies/**Erdos**.html ▾
Paul Erdős came from a Jewish family (the original family name being Engländer) although neither of his parents observed the Jewish religion. Paul's father ...

[PDF] Paul Erdős Mathematical Genius, Human - UnTruth.org
www.untruth.org/~josh/math/**Paul**%20**Erdös**%20bio-rev2.pdf ▾
by J Hill - 2004 - Related articles

Paul Erdős

Mathematician

Paul Erdős was a Hungarian Jewish mathematician. He was one of the most prolific mathematicians of the 20th century. He was known both for his social practice of mathematics and for his eccentric lifestyle.
Wikipedia

Born: March 26, 1913, Budapest, Hungary

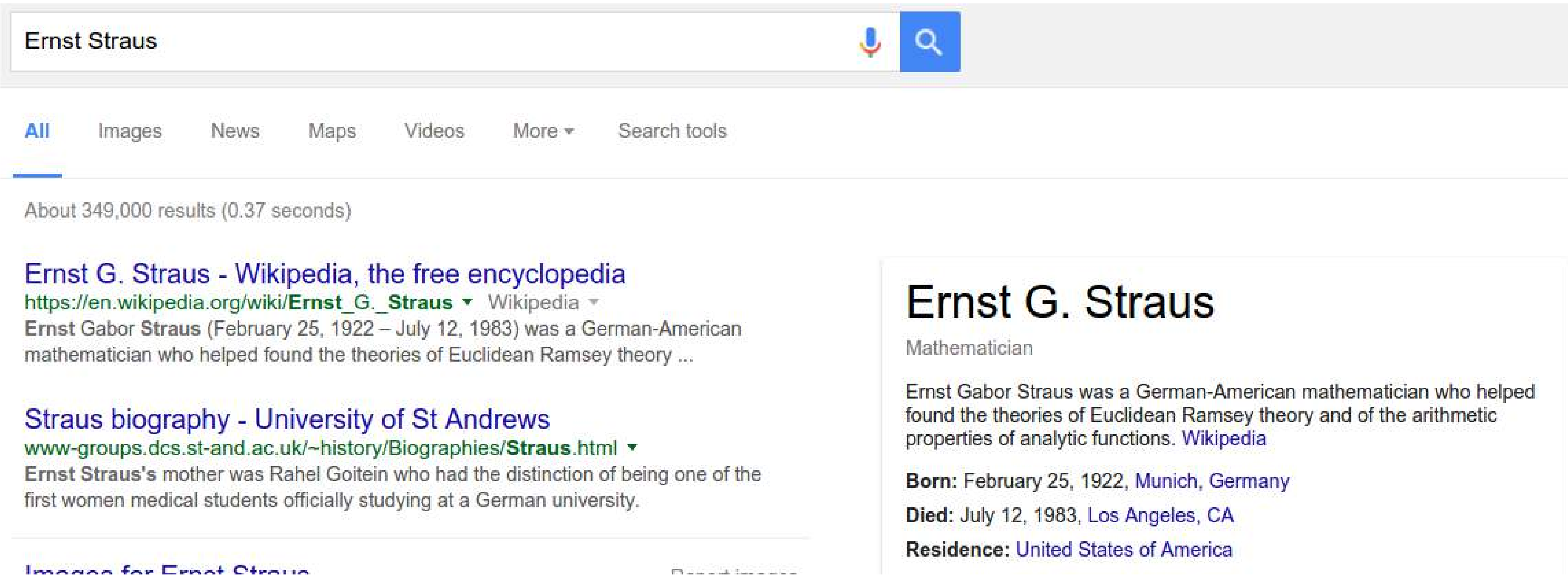
Died: September 20, 1996, Warsaw, Poland

Education: Eötvös Loránd University (1934)

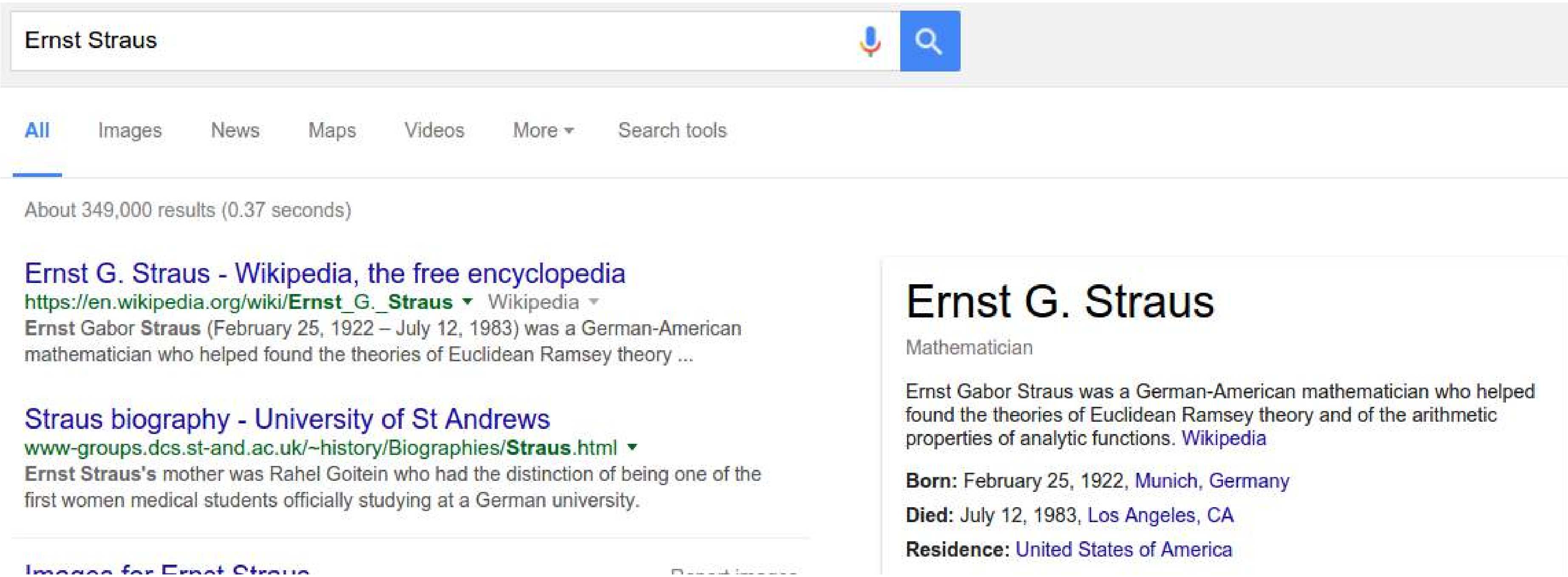
Books: Probabilistic Methods in Combinatorics, More

Notable students: Béla Bollobás, Alexander Soifer, George B. Purdy, Joseph Kruskal

This guy is in the Knowledge Graph



This guy is in the Knowledge Graph



… and he published with both Einstein and Erdos!

Desired Query Answer

Ernst Straus

Barack Obama, …

Justin Bieber, …

Desired Query Answer

Ernst Straus

Barack Obama, …

Justin Bieber, …

1. Fuse uncertain information from web

   ⇒ **Embrace probability!**

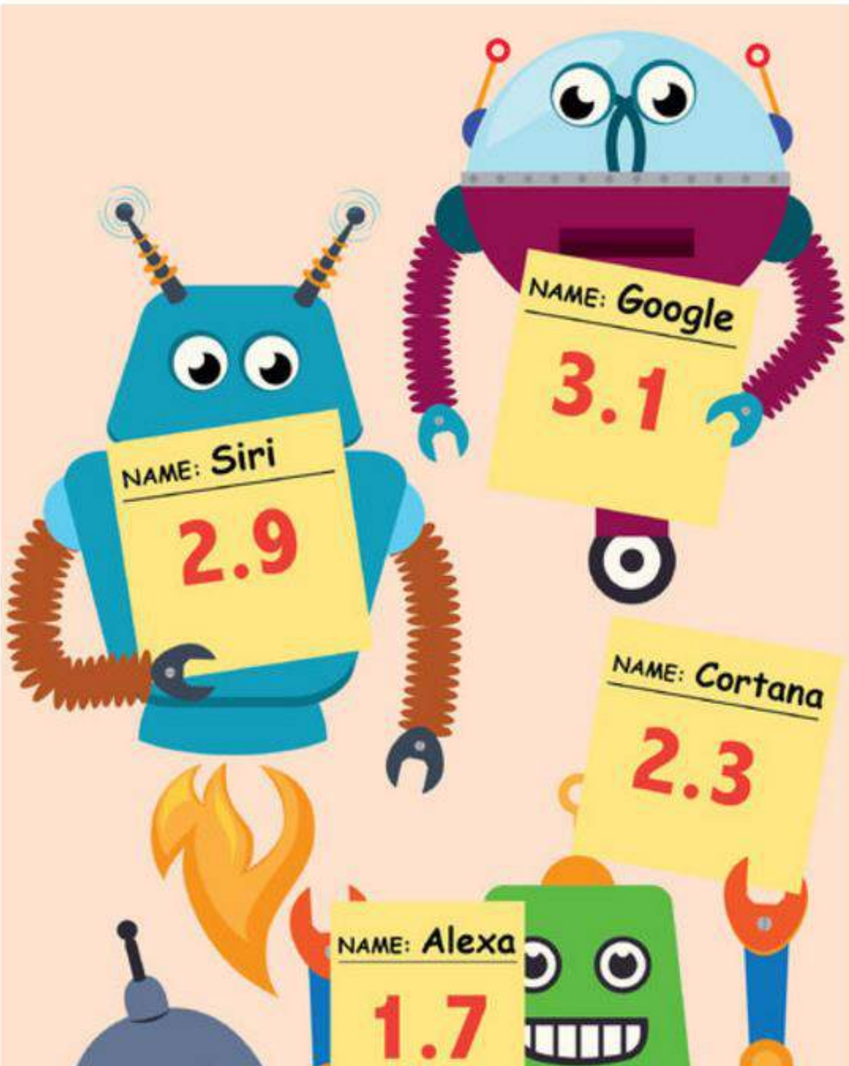
2. Cannot come from labeled data

   ⇒ **Embrace query eval!**

Siri, Alexa and Other Virtual Assistants Put to the Test

Tech Fix

By BRIAN X. CHEN JAN. 27, 2016



WHEN I asked Alexa earlier this week who was playing in the Super Bowl, she responded, somewhat monotonously, "Super Bowl 49's winner is New England Patriots."

"Come on, that's last year's Super Bowl," I said. "Even I can do better than that."

At the time, I was actually alone in my living room. I was talking to the virtual companion inside Amazon's wireless speaker, Echo, which was released last June. Known as Alexa, she has gained raves from Silicon Valley's tech-obsessed digerati and has become one of the newest members of the virtual assistants club.

All the so-called Frightful Five tech

[Chen'16]
(NYTimes)

How *probabilistic query evaluation?*

Tuple-Independent Probabilistic DB

Probabilistic database D:

Coauthor

x	y	P
A	B	p_1
A	C	p_2
B	C	p_3

[VdB&Suciu'17]

Tuple-Independent Probabilistic DB

Probabilistic database D:

Coauthor

x	y	P
A	B	p_1
A	C	p_2
B	C	p_3

Possible worlds semantics:

$p_1 p_2 p_3$

x	y
A	B
A	C
B	C

[VdB&Suciu'17]

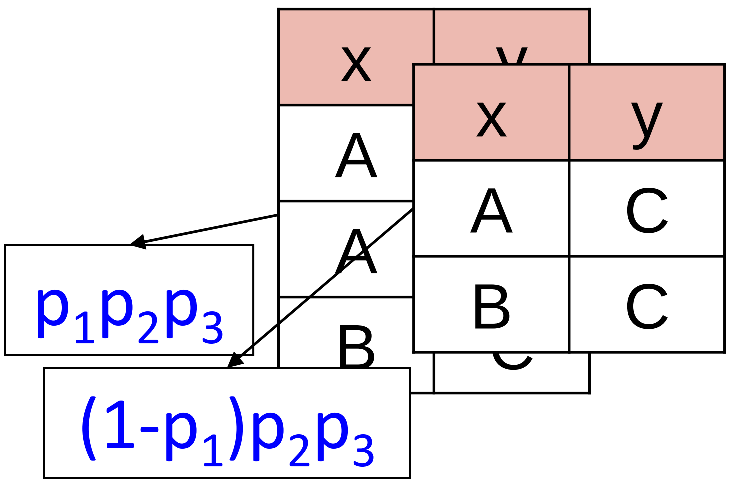
Tuple-Independent Probabilistic DB

Probabilistic database D:

Coauthor

x	y	P
A	B	p_1
A	C	p_2
B	C	p_3

Possible worlds semantics:



[VdB&Suciu'17]

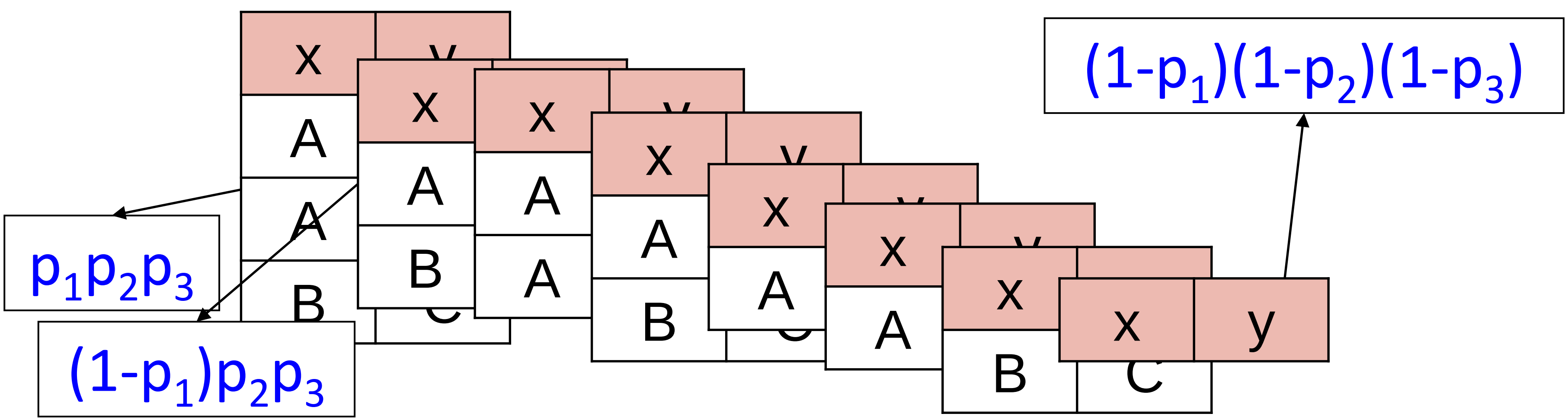
Tuple-Independent Probabilistic DB

Probabilistic database D:

Coauthor

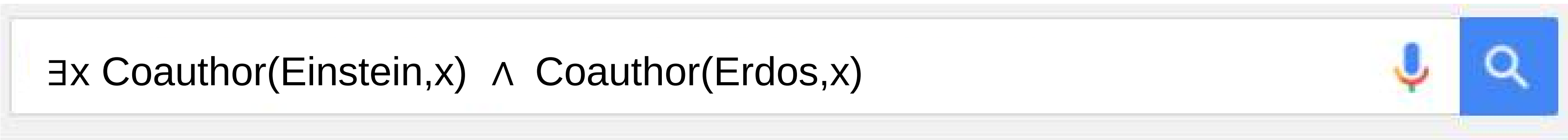
x	y	P
A	B	p_1
A	C	p_2
B	C	p_3

Possible worlds semantics:



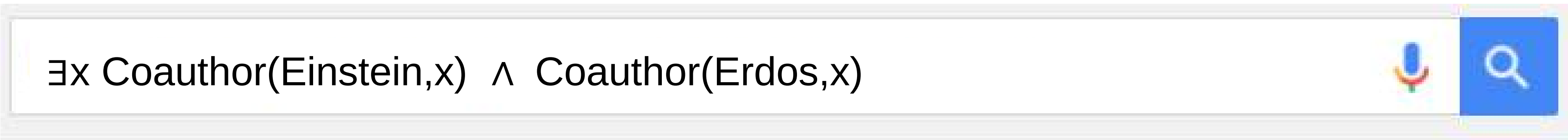
[VdB&Suciu'17]

Probabilistic Databases Queries



- Conjunctive queries (CQ)
  - $\exists$ + $\wedge$ + positive literals

Probabilistic Databases Queries



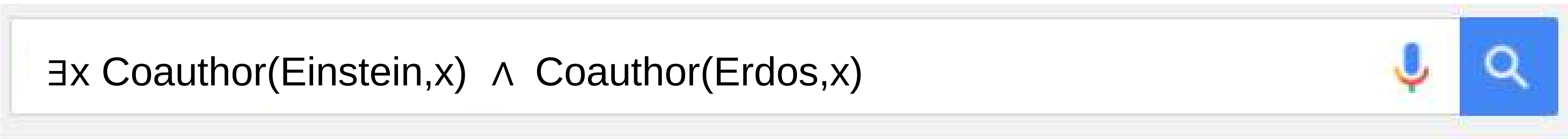
- Conjunctive queries (CQ)

  ∃ + ∧ + positive literals

- Unions of conjunctive queries (UCQ)

  ∨ of ∃ + ∧ + positive literals

Probabilistic Databases Queries



- Conjunctive queries (CQ)

  ∃ + ∧ + positive literals

- Unions of conjunctive queries (UCQ)

  ∨ of ∃ + ∧ + positive literals

- Duality
  - Negation of CQ is monotone ∀-clause
  - Negation of UCQ is monotone ∀-CNF

Probabilistic Query Evaluation

$Q = \exists x \exists y\ \text{Scientist}(x) \wedge \text{Coauthor}(x,y)$

$P(Q) =$

Scientist

x	P
A	p_1
B	p_2
C	p_3

Coauthor

x	y	P
A	D	q_1
A	E	q_2
B	F	q_3
B	G	q_4
B	H	q_5

Probabilistic Query Evaluation

$Q = \exists x \exists y \; \text{Scientist}(x) \wedge \text{Coauthor}(x,y)$

$P(Q) = \quad 1-(1-q_1)*(1-q_2)$

Scientist

x	P
A	p_1
B	p_2
C	p_3

Coauthor

x	y	P
A	D	q_1
A	E	q_2
B	F	q_3
B	G	q_4
B	H	q_5

Probabilistic Query Evaluation

$Q = \exists x \exists y\ \text{Scientist}(x) \land \text{Coauthor}(x,y)$

$$P(Q) = p_1 * [\ 1-(1-q_1)*(1-q_2)\ ]$$

Scientist

x	P
A	p_1
B	p_2
C	p_3

Coauthor

x	y	P
A	D	q_1
A	E	q_2
B	F	q_3
B	G	q_4
B	H	q_5

Probabilistic Query Evaluation

Q = ∃x∃y Scientist(x) ∧ Coauthor(x,y)

$$P(Q) = p_1 * [\ 1-(1-q_1)*(1-q_2)\ ]$$

$$1-(1-q_3)*(1-q_4)*(1-q_5)$$

Scientist

x	P
A	p_1
B	p_2
C	p_3

Coauthor

x	y	P
A	D	q_1
A	E	q_2
B	F	q_3
B	G	q_4
B	H	q_5

Probabilistic Query Evaluation

$Q = \exists x \exists y \; \text{Scientist}(x) \wedge \text{Coauthor}(x,y)$

$$P(Q) = \quad p_1 * [\; 1-(1-q_1)*(1-q_2) \;]$$
$$p_2 * [\; 1-(1-q_3)*(1-q_4)*(1-q_5) \;]$$

Scientist

x	P
A	p_1
B	p_2
C	p_3

Coauthor

x	y	P
A	D	q_1
A	E	q_2
B	F	q_3
B	G	q_4
B	H	q_5

Probabilistic Query Evaluation

Q = ∃x∃y Scientist(x) ∧ Coauthor(x,y)

$$P(Q) = 1 - \{1 - p_1 * [\ 1 - (1-q_1)*(1-q_2)\ ]\} * \{1 - p_2 * [\ 1 - (1-q_3)*(1-q_4)*(1-q_5)\ ]\}$$

Scientist

x	P
A	p_1
B	p_2
C	p_3

Coauthor

x	y	P
A	D	q_1
A	E	q_2
B	F	q_3
B	G	q_4
B	H	q_5

Lifted Inference Rules

Preprocess Q (omitted),
Then apply rules (some have preconditions)

[VdB&Suciu'17]

Lifted Inference Rules

Preprocess Q (omitted),
Then apply rules (some have preconditions)

$\boxed{P(\neg Q) = 1 - P(Q)}$

Negation

[VdB&Suciu'17]

Lifted Inference Rules

Preprocess Q (omitted),
Then apply rules (some have preconditions)

$P(\neg Q) = 1 - P(Q)$

Negation

$P(Q1 \wedge Q2) = P(Q1)\, P(Q2)$
$P(Q1 \vee Q2) = 1 - (1 - P(Q1))\, (1 - P(Q2))$

Decomposable ∧,∨

[VdB&Suciu'17]

Lifted Inference Rules

Preprocess Q (omitted),
Then apply rules (some have preconditions)

$P(\neg Q) = 1 - P(Q)$ — Negation

$P(Q1 \wedge Q2) = P(Q1)\, P(Q2)$
$P(Q1 \vee Q2) = 1 - (1 - P(Q1))\, (1 - P(Q2))$ — Decomposable $\wedge, \vee$

$P(\forall z\, Q) = \prod_{A \in Domain} P(Q[A/z])$
$P(\exists z\, Q) = 1 - \prod_{A \in Domain} (1 - P(Q[A/z]))$ — Decomposable $\exists, \forall$

[VdB&Suciu'17]

Lifted Inference Rules

Preprocess Q (omitted),
Then apply rules (some have preconditions)

$$P(\neg Q) = 1 - P(Q)$$

Negation

$$P(Q1 \wedge Q2) = P(Q1)\, P(Q2)$$
$$P(Q1 \vee Q2) = 1 - (1 - P(Q1))\, (1 - P(Q2))$$

Decomposable ∧,∨

$$P(\forall z\, Q) = \Pi_{A \in Domain}\ P(Q[A/z])$$
$$P(\exists z\, Q) = 1 - \Pi_{A \in Domain}\ (1 - P(Q[A/z]))$$

Decomposable ∃,∀

$$P(Q1 \wedge Q2) = P(Q1) + P(Q2) - P(Q1 \vee Q2)$$
$$P(Q1 \vee Q2) = P(Q1) + P(Q2) - P(Q1 \wedge Q2)$$

Inclusion/
exclusion

[VdB&Suciu'17]

Closed-World Lifted Query Eval

$$Q = \exists x \, \exists y \; \mathrm{Scientist}(x) \wedge \mathrm{Coauthor}(x,y)$$

$$P(Q) = 1 - \prod_{A \in \mathrm{Domain}} (1 - P(\mathrm{Scientist}(A) \wedge \exists y \; \mathrm{Coauthor}(A,y)))$$

Closed-World Lifted Query Eval

$$\boxed{Q = \exists x\, \exists y\ Scientist(x) \wedge Coauthor(x,y)}$$

Decomposable ∃-Rule

$$P(Q) = 1 - \Pi_{A \in Domain} (1 - P(Scientist(A) \wedge \exists y\ Coauthor(A,y))$$

Closed-World Lifted Query Eval

$\boxed{Q = \exists x\, \exists y\, \text{Scientist}(x) \wedge \text{Coauthor}(x,y)}$

Decomposable ∃-Rule

$$P(Q) = 1 - \prod_{A \in \text{Domain}} (1 - P(\text{Scientist}(A) \wedge \exists y\, \text{Coauthor}(A,y)))$$

Check independence:

$\text{Scientist}(A) \wedge \exists y\, \text{Coauthor}(A,y)$

$\text{Scientist}(B) \wedge \exists y\, \text{Coauthor}(B,y)$

Closed-World Lifted Query Eval

$\boxed{Q = \exists x\, \exists y\, \text{Scientist}(x) \wedge \text{Coauthor}(x,y)}$

Decomposable ∃-Rule

$$P(Q) = 1 - \Pi_{A \in \text{Domain}} (1 - P(\text{Scientist}(A) \wedge \exists y\, \text{Coauthor}(A,y))$$

Check independence:

$\text{Scientist}(A) \wedge \exists y\, \text{Coauthor}(A,y)$

$\text{Scientist}(B) \wedge \exists y\, \text{Coauthor}(B,y)$

$$\begin{aligned}
= 1 - &(1 - P(\text{Scientist}(A) \wedge \exists y\, \text{Coauthor}(A,y)) \\
&\times (1 - P(\text{Scientist}(B) \wedge \exists y\, \text{Coauthor}(B,y)) \\
&\times (1 - P(\text{Scientist}(C) \wedge \exists y\, \text{Coauthor}(C,y)) \\
&\times (1 - P(\text{Scientist}(D) \wedge \exists y\, \text{Coauthor}(D,y)) \\
&\times (1 - P(\text{Scientist}(E) \wedge \exists y\, \text{Coauthor}(E,y)) \\
&\times (1 - P(\text{Scientist}(F) \wedge \exists y\, \text{Coauthor}(F,y)) \\
&\quad \ldots
\end{aligned}$$

Closed-World Lifted Query Eval

$Q = \exists x\ \exists y\ \mathrm{Scientist}(x) \wedge \mathrm{Coauthor}(x,y)$

Decomposable ∃-Rule

$$P(Q) = 1 - \prod_{A \in \mathrm{Domain}} (1 - P(\mathrm{Scientist}(A) \wedge \exists y\ \mathrm{Coauthor}(A,y))$$

Check independence:

$\mathrm{Scientist}(A) \wedge \exists y\ \mathrm{Coauthor}(A,y)$

$\mathrm{Scientist}(B) \wedge \exists y\ \mathrm{Coauthor}(B,y)$

$$\begin{aligned}
= 1 - &(1 - P(\mathrm{Scientist}(A) \wedge \exists y\ \mathrm{Coauthor}(A,y)) \\
&\times (1 - P(\mathrm{Scientist}(B) \wedge \exists y\ \mathrm{Coauthor}(B,y)) \\
&\times (1 - P(\mathrm{Scientist}(C) \wedge \exists y\ \mathrm{Coauthor}(C,y)) \\
&\times (1 - P(\mathrm{Scientist}(D) \wedge \exists y\ \mathrm{Coauthor}(D,y)) \\
&\times (1 - P(\mathrm{Scientist}(E) \wedge \exists y\ \mathrm{Coauthor}(E,y)) \\
&\times (1 - P(\mathrm{Scientist}(F) \wedge \exists y\ \mathrm{Coauthor}(F,y)) \\
&\quad \ldots
\end{aligned}$$

Complexity PTIME

Limitations

$$H_0 = \forall x \forall y\, \text{Smoker}(x) \vee \text{Friend}(x,y) \vee \text{Jogger}(y)$$

The decomposable ∀-rule: $P(\forall z\ Q) = \Pi_{A\ \in \text{Domain}}\ P(Q[A/z])$

Limitations

$H_0 = \forall x \forall y\ \text{Smoker}(x) \lor \text{Friend}(x,y) \lor \text{Jogger}(y)$

The decomposable ∀-rule: $P(\forall z\ Q) = \Pi_{A\ \in \text{Domain}}\ P(Q[A/z])$

… does not apply:

$H_0[\text{Alice}/x]$ and $H_0[\text{Bob}/x]$ are dependent:

Dependent

$\forall y\ (\text{Smoker}(\text{Alice}) \lor \text{Friend}(\text{Alice},y) \lor \text{Jogger}(y))$

$\forall y\ (\text{Smoker}(\text{Bob}) \lor \text{Friend}(\text{Bob},y) \lor \text{Jogger}(y))$

Limitations

$H_0 = \forall x \forall y\ Smoker(x) \lor Friend(x,y) \lor Jogger(y)$

The decomposable ∀-rule: $P(\forall z\ Q) = \Pi_{A\ \in Domain}\ P(Q[A/z])$

… does not apply:

$H_0[Alice/x]$ and $H_0[Bob/x]$ are dependent:

Dependent

$\forall y\ (Smoker(Alice) \lor Friend(Alice,y) \lor Jogger(y))$

$\forall y\ (Smoker(Bob) \lor Friend(Bob,y) \lor Jogger(y))$

Lifted inference sometimes fails.

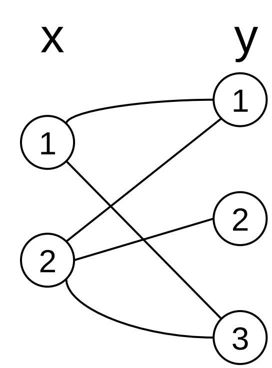
Background: Positive Partitioned 2CNF

A PP2CNF is:

$$F = \bigwedge_{(i,j) \in E} (x_i \vee y_j)$$

where E = the edge set of a bipartite graph

$$F = (x_1 \vee y_1) \wedge (x_2 \vee y_1) \wedge (x_2 \vee y_3) \wedge (x_1 \vee y_3) \wedge (x_2 \vee y_2)$$



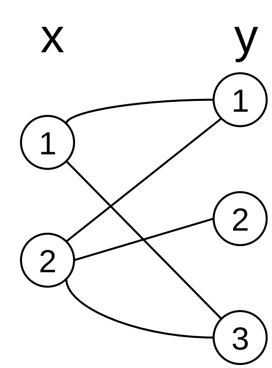
Background: Positive Partitioned 2CNF

A PP2CNF is:

$$F = \bigwedge_{(i,j) \in E} (x_i \vee y_j)$$

where E = the edge set of a bipartite graph

$$F = (x_1 \vee y_1) \wedge (x_2 \vee y_1) \wedge (x_2 \vee y_3) \wedge (x_1 \vee y_3) \wedge (x_2 \vee y_2)$$



Theorem: #PP2CNF is **#P**-hard [Provan'83]

Our Problematic Clause

$H_0 = \forall x \forall y\ \text{Smoker}(x) \lor \text{Friend}(x,y) \lor \text{Jogger}(y)$

Our Problematic Clause

$H_0 = \forall x \forall y\ \text{Smoker}(x) \vee \text{Friend}(x,y) \vee \text{Jogger}(y)$

Theorem. Computing $P(H_0)$ is **#P**-hard in the size of the database.

[Dalvi&Suciu'04]

Our Problematic Clause

$H_0 = \forall x \forall y\ \mathrm{Smoker}(x) \lor \mathrm{Friend}(x,y) \lor \mathrm{Jogger}(y)$

Theorem. Computing $P(H_0)$ is **#P**-hard in the size of the database.

[Dalvi&Suciu'04]

Proof: PP2CNF: $F = (X_{i1} \lor Y_{j1}) \land (X_{i2} \lor Y_{j2}) \land \ldots$ reduce #F to computing $P(H_0)$

By example:

Our Problematic Clause

$H_0 = \forall x \forall y \; \text{Smoker}(x) \vee \text{Friend}(x,y) \vee \text{Jogger}(y)$

Theorem. Computing $P(H_0)$ is **#P**-hard in the size of the database.

[Dalvi&Suciu'04]

Proof: PP2CNF: $F = (X_{i1} \vee Y_{j1}) \wedge (X_{i2} \vee Y_{j2}) \wedge \ldots$ reduce #F to computing $P(H_0)$

By example:

$F = (X_1 \vee Y_1) \wedge (X_1 \vee Y_2) \wedge (X_2 \vee Y_2)$

Our Problematic Clause

$H_0 = \forall x \forall y\ \text{Smoker}(x) \vee \text{Friend}(x,y) \vee \text{Jogger}(y)$

Theorem. Computing $P(H_0)$ is **#P**-hard in the size of the database.

[Dalvi&Suciu'04]

Proof: PP2CNF: $F = (X_{i1} \vee Y_{j1}) \wedge (X_{i2} \vee Y_{j2}) \wedge \ldots$ reduce #F to computing $P(H_0)$

By example:

$F = (X_1 \vee Y_1) \wedge (X_1 \vee Y_2) \wedge (X_2 \vee Y_2)$

Probabilities (tuples not shown have P=1)

Smoker

X	P
x_1	0.5
x_2	0.5

Friend

X	Y	P
x_1	y_1	0
x_1	y_2	0
x_2	y_2	0

Jogger

Y	P
y_1	0.5
y_2	0.5

Our Problematic Clause

$H_0 = \forall x \forall y\ \text{Smoker}(x) \lor \text{Friend}(x,y) \lor \text{Jogger}(y)$

Theorem. Computing $P(H_0)$ is **#P**-hard in the size of the database.

[Dalvi&Suciu'04]

Proof: PP2CNF: $F = (X_{i1} \lor Y_{j1}) \land (X_{i2} \lor Y_{j2}) \land \ldots$ reduce #F to computing $P(H_0)$

By example:

$F = (X_1 \lor Y_1) \land (X_1 \lor Y_2) \land (X_2 \lor Y_2)$

$P(H_0) = P(F)$; hence $P(H_0)$ is #P-hard

Probabilities (tuples not shown have P=1)

Smoker

X	P
x_1	0.5
x_2	0.5

Friend

X	Y	P
x_1	y_1	0
x_1	y_2	0
x_2	y_2	0

Jogger

Y	P
y_1	0.5
y_2	0.5

Are the Lifted Rules Complete?

You already know:

- Inference rules: PTIME data complexity
- Some queries: #P-hard data complexity

[Dalvi and Suciu;JACM'11]

Are the Lifted Rules Complete?

You already know:

- Inference rules: PTIME data complexity
- Some queries: #P-hard data complexity

Dichotomy Theorem for UCQ / Mon. CNF

- If lifted rules succeed, then PTIME query
- If lifted rules fail, then query is #P-hard

[Dalvi and Suciu;JACM'11]

Are the Lifted Rules Complete?

You already know:

- Inference rules: PTIME data complexity
- Some queries: #P-hard data complexity

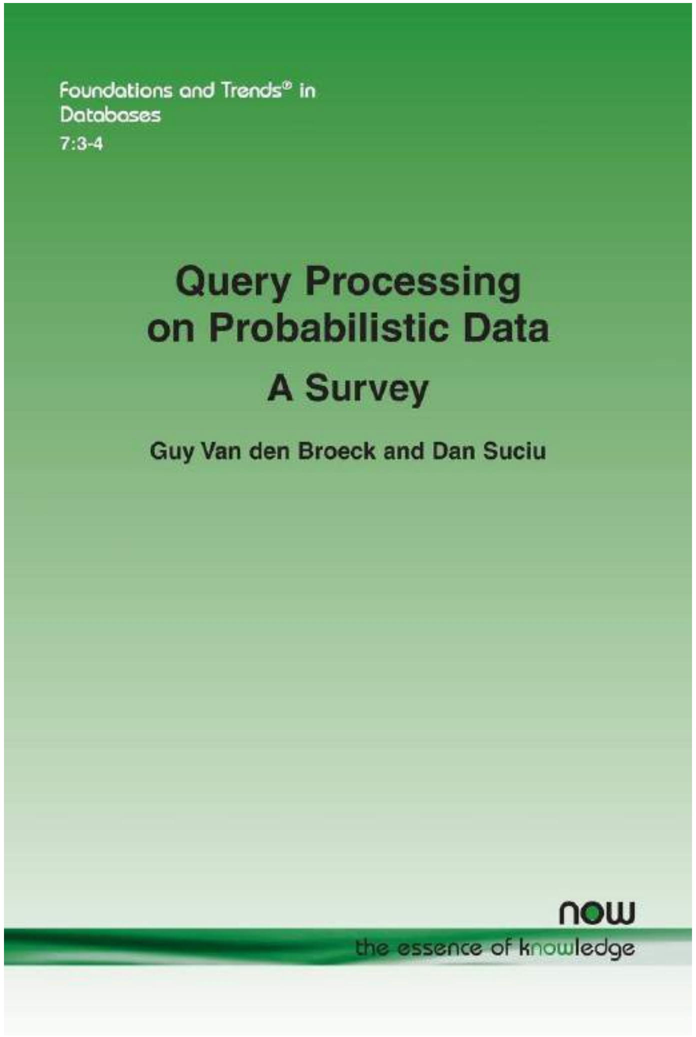
Dichotomy Theorem for UCQ / Mon. CNF

- If lifted rules succeed, then PTIME query
- If lifted rules fail, then query is #P-hard

Lifted rules are complete for UCQ!

[Dalvi and Suciu;JACM'11]

Commercial Break



- Survey book (2017)
  http://www.nowpublishers.com/article/Details/DBS-052
- IJCAI 2016 tutorial
  http://web.cs.ucla.edu/~guyvdb/talks/IJCAI16-tutorial/

Why *open world?*

Knowledge Base Completion

Given:

Coauthor

x	y	P
Einstein	Straus	0.7
Erdos	Straus	0.6
Einstein	Pauli	0.9
…	…	…

Learn:

0.8::Coauthor(x,y) :- Coauthor(z,x) ∧ Coauthor(z,y).

Complete:

x	y	P
Straus	Pauli	0.504
…	…	…

Bayesian Learning Loop

Bayesian view on learning:

1. Prior belief:

   P(Coauthor(Straus,Pauli)) = 0.01

2. Observe page

   P(Coauthor(Straus,Pauli| ) = 0.2

3. Observe page

   P(Coauthor(Straus,Pauli)|  , ) = 0.3

Principled and sound reasoning!

Problem: Broken Learning Loop

Bayesian view on learning:

1. Prior belief:

   $P(\texttt{Coauthor(Straus,Pauli)}) = 0$

2. Observe page

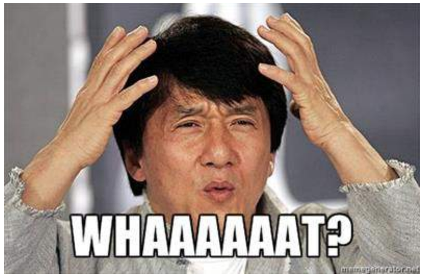
   $P(\texttt{Coauthor(Straus,Pauli} \mid$  $) = 0.2$

3. Observe page

   $P(\texttt{Coauthor(Straus,Pauli)} \mid$  ,  $) = 0.3$

[Ceylan, Darwiche, Van den Broeck; KR'16]

Problem: Broken Learning Loop



Bayesian view on learning:

1. Prior belief:

   $P(\texttt{Coauthor(Straus,Pauli)}) = 0$

2. Observe page

   $P(\texttt{Coauthor(Straus,Pauli} \mid$  $) = 0.2$

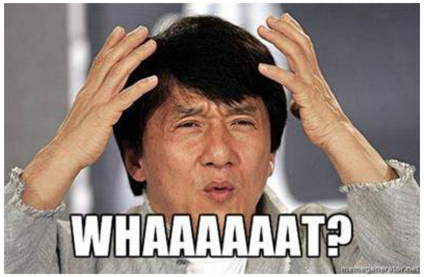
3. Observe page

   $P(\texttt{Coauthor(Straus,Pauli)} \mid$  ,  $) = 0.3$

[Ceylan, Darwiche, Van den Broeck; KR'16]

Problem: Broken Learning Loop

Bayesian view on learning:



1. Prior belief:

   $P(\mathtt{Coauthor(Straus,Pauli)}) = 0$

2. Observe page

   $P(\mathtt{Coauthor(Straus,Pauli} \mid$  $) = 0.2$

3. Observe page

   $P(\mathtt{Coauthor(Straus,Pauli)} \mid$  ,  $) = 0.3$

This is mathematical nonsense!

[Ceylan, Darwiche, Van den Broeck; KR'16]

What we'd like to do…

∃x Coauthor(Einstein,x) ∧ Coauthor(Erdos,x)

Ernst Straus

Kristian Kersting, …

Justin Bieber, …

Open World DB

- What if fact missing?

- Probability 0 for:

Q1 = ∃x Coauthor(Einstein,**x**) ∧ Coauthor(Erdos,**x**)

Coauthor

X	Y	P
Einstein	Straus	0.7
~~**Erdos**~~	~~**Straus**~~	~~**0.6**~~
Einstein	Pauli	0.9
Erdos	Renyi	0.7
Kersting	Natarajan	0.8
Luc	Paol	0.1
…	…	…

Open World DB

- What if fact missing?

- Probability 0 for:

Coauthor

X	Y	P
Einstein	Straus	0.7
~~**Erdos**~~	~~**Straus**~~	~~**0.6**~~
Einstein	Pauli	0.9
Erdos	Renyi	0.7
Kersting	Natarajan	0.8
Luc	Paol	0.1
…	…	…

$$Q1 = \exists x\ \text{Coauthor}(\text{Einstein},\mathbf{x}) \land \text{Coauthor}(\text{Erdos},\mathbf{x})$$

$$Q2 = \exists x\ \text{Coauthor}(\text{Bieber},\mathbf{x}) \land \text{Coauthor}(\text{Erdos},\mathbf{x})$$

Coauthor

X	Y	P
Einstein	Straus	0.7
~~Erdos~~	~~Straus~~	~~0.6~~
Einstein	Pauli	0.9
Erdos	Renyi	0.7
Kersting	Natarajan	0.8
Luc	Paol	0.1
…	…	…

Open World DB

- What if fact missing?

- Probability 0 for:

Q1 = ∃x Coauthor(Einstein,**x**) ∧ Coauthor(Erdos,**x**)

Q2 = ∃x Coauthor(Bieber,**x**) ∧ Coauthor(Erdos,**x**)

Q3 = Coauthor(Einstein,**Straus**) ∧ Coauthor(Erdos,**Straus**)

Open World DB

- What if fact missing?
- Probability 0 for:

Coauthor

X	Y	P
Einstein	Straus	0.7
~~**Erdos**~~	~~**Straus**~~	~~**0.6**~~
Einstein	Pauli	0.9
Erdos	Renyi	0.7
Kersting	Natarajan	0.8
Luc	Paol	0.1
…	…	…

Q1 = $\exists x$ Coauthor(Einstein,**x**) $\wedge$ Coauthor(Erdos,**x**)

Q2 = $\exists x$ Coauthor(Bieber,**x**) $\wedge$ Coauthor(Erdos,**x**)

Q3 = Coauthor(Einstein,**Straus**) $\wedge$ Coauthor(Erdos,**Straus**)

Q4 = Coauthor(Einstein,**Bieber**) $\wedge$ Coauthor(Erdos,**Bieber**)

Open World DB

- What if fact missing?
- Probability 0 for:

Coauthor

X	Y	P
Einstein	Straus	0.7
~~**Erdos**~~	~~**Straus**~~	~~**0.6**~~
Einstein	Pauli	0.9
Erdos	Renyi	0.7
Kersting	Natarajan	0.8
Luc	Paol	0.1
…	…	…

Q1 = ∃x Coauthor(Einstein,**x**) ∧ Coauthor(Erdos,**x**)

Q2 = ∃x Coauthor(Bieber,**x**) ∧ Coauthor(Erdos,**x**)

Q3 = Coauthor(Einstein,**Straus**) ∧ Coauthor(Erdos,**Straus**)

Q4 = Coauthor(Einstein,**Bieber**) ∧ Coauthor(Erdos,**Bieber**)

Q5 = Coauthor(Einstein,**Bieber**) ∧ ¬Coauthor(**Einstein**,**Bieber**)

Intuition

X	Y	P
Einstein	Straus	0.7
~~**Erdos**~~	~~**Straus**~~	~~**0.6**~~
Einstein	Pauli	0.9
Erdos	Renyi	0.7
Kersting	Natarajan	0.8
Luc	Paol	0.1
…	…	…

Q1 = ∃x Coauthor(Einstein,**x**) ∧ Coauthor(Erdos,**x**)

Q3 = Coauthor(Einstein,**Straus**) ∧ Coauthor(Erdos,**Straus**)

Q4 = Coauthor(Einstein,**Bieber**) ∧ Coauthor(Erdos,**Bieber**)

[Ceylan, Darwiche, Van den Broeck; KR'16]

Intuition

X	Y	P
Einstein	Straus	0.7
~~**Erdos**~~	~~**Straus**~~	~~**0.6**~~
Einstein	Pauli	0.9
Erdos	Renyi	0.7
Kersting	Natarajan	0.8
Luc	Paol	0.1
…	…	…

Q1 = ∃x Coauthor(Einstein,**x**) ∧ Coauthor(Erdos,**x**)

Q3 = Coauthor(Einstein,**Straus**) ∧ Coauthor(Erdos,**Straus**)

Q4 = Coauthor(Einstein,**Bieber**) ∧ Coauthor(Erdos,**Bieber**)

We know for sure that P(Q1) ≥ P(Q3), P(Q1) ≥ P(Q4)

[Ceylan, Darwiche, Van den Broeck; KR'16]

Intuition

X	Y	P
Einstein	Straus	0.7
~~**Erdos**~~	~~**Straus**~~	~~**0.6**~~
Einstein	Pauli	0.9
Erdos	Renyi	0.7
Kersting	Natarajan	0.8
Luc	Paol	0.1
…	…	…

Q1 = ∃x Coauthor(Einstein,**x**) ∧ Coauthor(Erdos,**x**)

Q3 = Coauthor(Einstein,**Straus**) ∧ Coauthor(Erdos,**Straus**)

Q4 = Coauthor(Einstein,**Bieber**) ∧ Coauthor(Erdos,**Bieber**)

Q5 = Coauthor(Einstein,**Bieber**) ∧ ¬Coauthor(**Einstein**,**Bieber**)

We know for sure that P(Q1) ≥ P(Q3), P(Q1) ≥ P(Q4)

and P(Q3) ≥ P(Q5), P(Q4) ≥ P(Q5)

[Ceylan, Darwiche, Van den Broeck; KR’16]

Intuition

X	Y	P
Einstein	Straus	0.7
~~**Erdos**~~	~~**Straus**~~	~~**0.6**~~
Einstein	Pauli	0.9
Erdos	Renyi	0.7
Kersting	Natarajan	0.8
Luc	Paol	0.1
…	…	…

$Q1 = \exists x\ \text{Coauthor}(\text{Einstein},\mathbf{x}) \land \text{Coauthor}(\text{Erdos},\mathbf{x})$

$Q3 = \text{Coauthor}(\text{Einstein},\mathbf{Straus}) \land \text{Coauthor}(\text{Erdos},\mathbf{Straus})$

$Q4 = \text{Coauthor}(\text{Einstein},\mathbf{Bieber}) \land \text{Coauthor}(\text{Erdos},\mathbf{Bieber})$

$Q5 = \text{Coauthor}(\text{Einstein},\mathbf{Bieber}) \land \neg\text{Coauthor}(\mathbf{Einstein},\mathbf{Bieber})$

We know for sure that $P(Q1) \geq P(Q3)$, $P(Q1) \geq P(Q4)$

and $P(Q3) \geq P(Q5)$, $P(Q4) \geq P(Q5)$ because $P(Q5) = 0$.

[Ceylan, Darwiche, Van den Broeck; KR'16]

Intuition

X	Y	P
Einstein	Straus	0.7
~~**Erdos**~~	~~**Straus**~~	~~**0.6**~~
Einstein	Pauli	0.9
Erdos	Renyi	0.7
Kersting	Natarajan	0.8
Luc	Paol	0.1
…	…	…

Q1 = ∃x Coauthor(Einstein,**x**) ∧ Coauthor(Erdos,**x**)

Q2 = ∃x Coauthor(Bieber,**x**) ∧ Coauthor(Erdos,**x**)

Q3 = Coauthor(Einstein,**Straus**) ∧ Coauthor(Erdos,**Straus**)

Q4 = Coauthor(Einstein,**Bieber**) ∧ Coauthor(Erdos,**Bieber**)

Q5 = Coauthor(Einstein,**Bieber**) ∧ ¬Coauthor(**Einstein**,**Bieber**)

We know for sure that P(Q1) ≥ P(Q3), P(Q1) ≥ P(Q4)

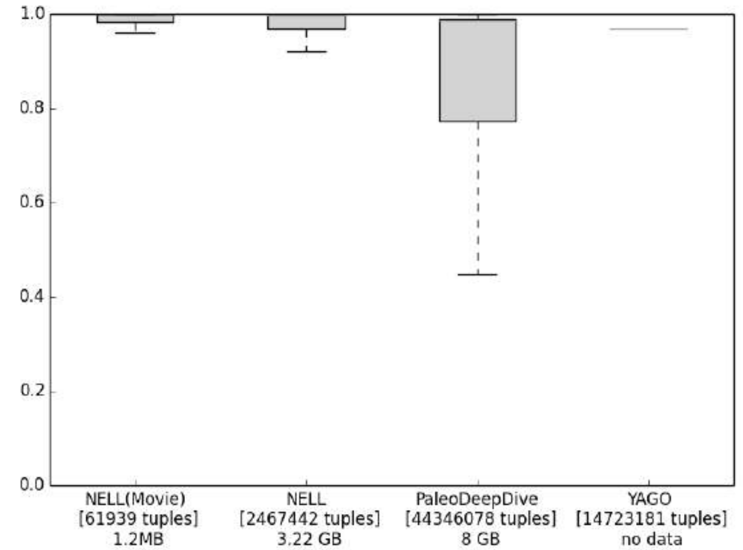
and P(Q3) ≥ P(Q5), P(Q4) ≥ P(Q5) because P(Q5) = 0.

We have strong evidence that P(Q1) ≥ P(Q2).

[Ceylan, Darwiche, Van den Broeck; KR'16]

Problem: Curse of Superlinearity

Reality is worse: tuples **intentionally** missing!



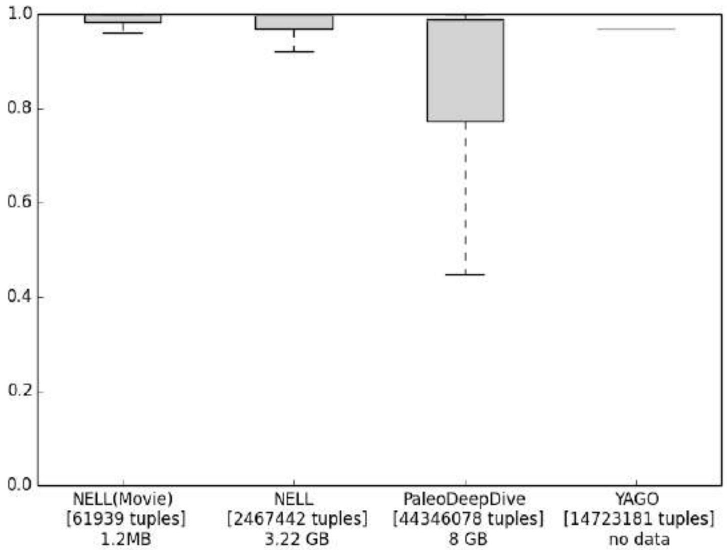
[Ceylan, Darwiche, Van den Broeck; KR'16]

Problem: Curse of Superlinearity

Reality is worse: tuples **intentionally** missing!

Sibling

x	y	P
…	…	…



Facebook scale

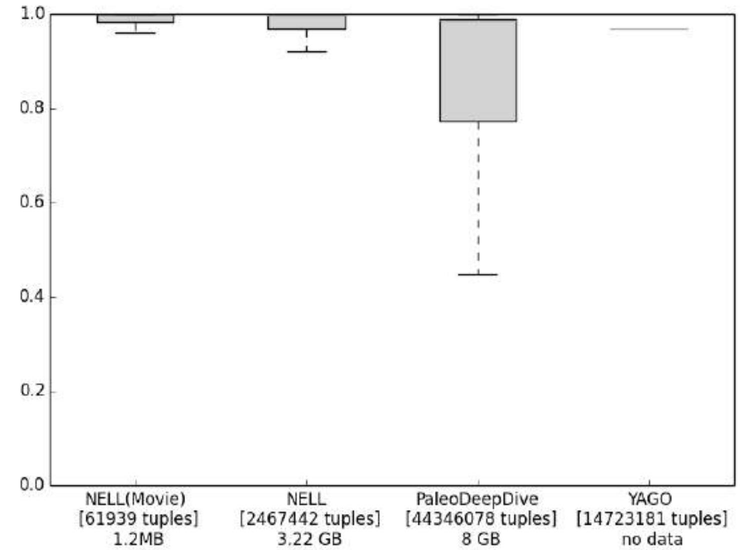
[Ceylan, Darwiche, Van den Broeck; KR'16]

Problem: Curse of Superlinearity

Reality is worse: tuples **intentionally** missing!

Sibling

x	y	P
…	…	…



Facebook scale ⇒ 200 Exabytes of data

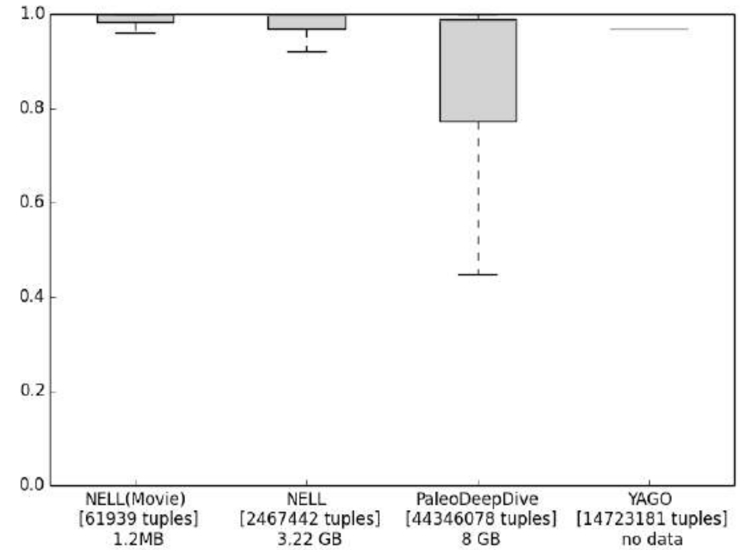
[Ceylan, Darwiche, Van den Broeck; KR'16]

Problem: Curse of Superlinearity

Reality is worse: tuples **intentionally** missing!

Sibling

x	y	P
…	…	…



Facebook scale ⇒ 200 Exabytes of data

All Google storage is 2 exabytes…

Randall Munroe. Google's datacenters on punch cards, 2015.

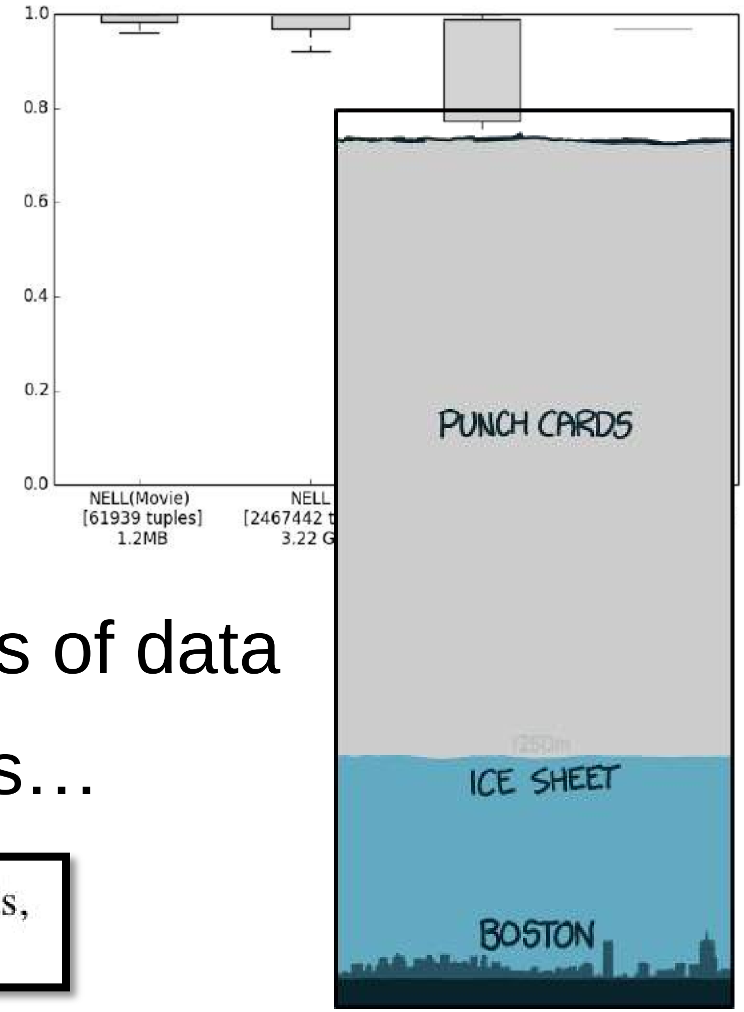
[Ceylan, Darwiche, Van den Broeck; KR'16]

Problem: Curse of Superlinearity

Reality is worse: tuples **intentionally** missing!

Sibling

x	y	P
…	…	…



Facebook scale ⇒ 200 Exabytes of data

All Google storage is 2 exabytes…

Randall Munroe. Google's datacenters on punch cards, 2015.

[Ceylan, Darwiche, Van den Broeck; KR'16]

Problem: Model Evaluation

Given:

Coauthor

x	y	P
Einstein	Straus	0.7
Erdos	Straus	0.6
Einstein	Pauli	0.9
…	…	…

Learn:

0.8::Coauthor(x,y) :- Coauthor(z,x) ∧ Coauthor(z,y).

OR

0.6::Coauthor(x,y) :- Affiliation(x,z) ∧ Affiliation(y,z).

[De Raedt et al; IJCAI'15]

Problem: Model Evaluation

Given:

Coauthor

x	y	P
Einstein	Straus	0.7
Erdos	Straus	0.6
Einstein	Pauli	0.9
…	…	…

Learn:

0.8::Coauthor(x,y) :- Coauthor(z,x) ∧ Coauthor(z,y).

OR

0.6::Coauthor(x,y) :- Affiliation(x,z) ∧ Affiliation(y,z).

What is the likelihood, precision, accuracy, …?

[De Raedt et al; IJCAI'15]

Open-World Prob. Databases

Intuition: tuples can be added with $P < \lambda$

Q2 = Coauthor(Einstein,**Straus**) $\wedge$ Coauthor(Erdos,**Straus**)

$P(Q2) \geq 0$

Coauthor

X	Y	P
Einstein	Straus	0.7
Einstein	Pauli	0.9
Erdos	Renyi	0.7
Kersting	Natarajan	0.8
Luc	Paol	0.1
…	…	…

Open-World Prob. Databases

Intuition: tuples can be added with P < λ

Q2 = Coauthor(Einstein,**Straus**) ∧ Coauthor(Erdos,**Straus**)

P(Q2) ≥ 0

Coauthor

X	Y	P
Einstein	Straus	0.7
Einstein	Pauli	0.9
Erdos	Renyi	0.7
Kersting	Natarajan	0.8
Luc	Paol	0.1
…	…	…

Coauthor

X	Y	P
Einstein	Straus	0.7
Einstein	Pauli	0.9
Erdos	Renyi	0.7
Kersting	Natarajan	0.8
Luc	Paol	0.1
…	…	…
Erdos	**Straus**	**λ**

Open-World Prob. Databases

Intuition: tuples can be added with $P < \lambda$

Q2 = Coauthor(Einstein,**Straus**) $\wedge$ Coauthor(Erdos,**Straus**)

$0.7 * \lambda \geq P(Q2) \geq 0$

Coauthor

X	Y	P
Einstein	Straus	0.7
Einstein	Pauli	0.9
Erdos	Renyi	0.7
Kersting	Natarajan	0.8
Luc	Paol	0.1
…	…	…

Coauthor

X	Y	P
Einstein	Straus	0.7
Einstein	Pauli	0.9
Erdos	Renyi	0.7
Kersting	Natarajan	0.8
Luc	Paol	0.1
…	…	…
Erdos	**Straus**	**λ**

How *open-world query evaluation?*

UCQ / Monotone CNF

- Lower bound = closed-world probability
- Upper bound = probability after **adding all tuples** with probability λ

UCQ / Monotone CNF

- Lower bound = closed-world probability
- Upper bound = probability after **adding all tuples** with probability λ

- Polynomial time ☺

UCQ / Monotone CNF

- Lower bound = closed-world probability
- Upper bound = probability after **adding all tuples** with probability λ

- Polynomial time ☺
- Quadratic blow-up ☹
- 200 exabytes … again ☹

Closed-World Lifted Query Eval

$$Q = \exists x\, \exists y\, \mathrm{Scientist}(x) \wedge \mathrm{Coauthor}(x,y)$$

$$P(Q) = 1 - \Pi_{A \in \mathrm{Domain}} (1 - P(\mathrm{Scientist}(A) \wedge \exists y\, \mathrm{Coauthor}(A,y))$$

Closed-World Lifted Query Eval

$$Q = \exists x \, \exists y \, \text{Scientist}(x) \wedge \text{Coauthor}(x,y)$$

Decomposable ∃-Rule

$$P(Q) = 1 - \prod_{A \in \text{Domain}} (1 - P(\text{Scientist}(A) \wedge \exists y \, \text{Coauthor}(A,y)))$$

Closed-World Lifted Query Eval

$\boxed{Q = \exists x\, \exists y\, \text{Scientist}(x) \land \text{Coauthor}(x,y)}$

Decomposable ∃-Rule

$$P(Q) = 1 - \Pi_{A \in \text{Domain}} (1 - P(\text{Scientist}(A) \land \exists y\, \text{Coauthor}(A,y))$$

$$\begin{aligned}
= 1 - &(1 - P(\text{Scientist}(A) \land \exists y\, \text{Coauthor}(A,y)) \\
&\times (1 - P(\text{Scientist}(B) \land \exists y\, \text{Coauthor}(B,y)) \\
&\times (1 - P(\text{Scientist}(C) \land \exists y\, \text{Coauthor}(C,y)) \\
&\times (1 - P(\text{Scientist}(D) \land \exists y\, \text{Coauthor}(D,y)) \\
&\times (1 - P(\text{Scientist}(E) \land \exists y\, \text{Coauthor}(E,y)) \\
&\times (1 - P(\text{Scientist}(F) \land \exists y\, \text{Coauthor}(F,y)) \\
&\quad \ldots
\end{aligned}$$

Closed-World Lifted Query Eval

$Q = \exists x\, \exists y\, \text{Scientist}(x) \wedge \text{Coauthor}(x,y)$

Decomposable ∃-Rule

$$P(Q) = 1 - \prod_{A \in \text{Domain}} (1 - P(\text{Scientist}(A) \wedge \exists y\, \text{Coauthor}(A,y))$$

$$\begin{aligned}
= 1 - (1 - &P(\text{Scientist}(A) \wedge \exists y\, \text{Coauthor}(A,y)) \\
\times (1 - &P(\text{Scientist}(B) \wedge \exists y\, \text{Coauthor}(B,y)) \\
\times (1 - &P(\text{Scientist}(C) \wedge \exists y\, \text{Coauthor}(C,y)) \\
\times (1 - &P(\text{Scientist}(D) \wedge \exists y\, \text{Coauthor}(D,y)) \\
\times (1 - &P(\text{Scientist}(E) \wedge \exists y\, \text{Coauthor}(E,y)) \\
\times (1 - &P(\text{Scientist}(F) \wedge \exists y\, \text{Coauthor}(F,y)) \\
&\ldots
\end{aligned}$$

Complexity PTIME

Closed-World Lifted Query Eval

$Q = \exists x\, \exists y\, Scientist(x) \wedge Coauthor(x,y)$

Decomposable ∃-Rule

$$P(Q) = 1 - \Pi_{A \in Domain} (1 - P(Scientist(A) \wedge \exists y\, Coauthor(A,y))$$

Check independence:

$Scientist(A) \wedge \exists y\, Coauthor(A,y)$

$Scientist(B) \wedge \exists y\, Coauthor(B,y)$

$$\begin{aligned} &= 1 - (1 - P(Scientist(A) \wedge \exists y\, Coauthor(A,y)) \\ &\quad \times (1 - P(Scientist(B) \wedge \exists y\, Coauthor(B,y)) \\ &\quad \times (1 - P(Scientist(C) \wedge \exists y\, Coauthor(C,y)) \\ &\quad \times (1 - P(Scientist(D) \wedge \exists y\, Coauthor(D,y)) \\ &\quad \times (1 - P(Scientist(E) \wedge \exists y\, Coauthor(E,y)) \\ &\quad \times (1 - P(Scientist(F) \wedge \exists y\, Coauthor(F,y)) \\ &\quad \quad \ldots \end{aligned}$$

Complexity PTIME

Closed-World Lifted Query Eval

$$\boxed{Q = \exists x\, \exists y\, \mathrm{Scientist}(x) \wedge \mathrm{Coauthor}(x,y)}$$

$$P(Q) = 1 - \Pi_{A \in \mathrm{Domain}} (1 - P(\mathrm{Scientist}(A) \wedge \exists y\, \mathrm{Coauthor}(A,y))$$

$$\begin{aligned}
= 1 - &(1 - P(\mathrm{Scientist}(A) \wedge \exists y\, \mathrm{Coauthor}(A,y)) \\
&\times (1 - P(\mathrm{Scientist}(B) \wedge \exists y\, \mathrm{Coauthor}(B,y)) \\
&\times (1 - P(\mathrm{Scientist}(C) \wedge \exists y\, \mathrm{Coauthor}(C,y)) \\
&\times (1 - P(\mathrm{Scientist}(D) \wedge \exists y\, \mathrm{Coauthor}(D,y)) \\
&\times (1 - P(\mathrm{Scientist}(E) \wedge \exists y\, \mathrm{Coauthor}(E,y)) \\
&\times (1 - P(\mathrm{Scientist}(F) \wedge \exists y\, \mathrm{Coauthor}(F,y)) \\
&\quad \ldots
\end{aligned}$$

Closed-World Lifted Query Eval

$\boxed{Q = \exists x \, \exists y \, \text{Scientist}(x) \wedge \text{Coauthor}(x,y)}$

$$P(Q) = 1 - \prod_{A \in \text{Domain}} (1 - P(\text{Scientist}(A) \wedge \exists y \, \text{Coauthor}(A,y)))$$

$$\begin{aligned}
= 1 - &(1 - P(\text{Scientist}(A) \wedge \exists y \, \text{Coauthor}(A,y))) \\
\times &(1 - P(\text{Scientist}(B) \wedge \exists y \, \text{Coauthor}(B,y))) \\
\times &(1 - P(\text{Scientist}(C) \wedge \exists y \, \text{Coauthor}(C,y))) \\
\times &(1 - P(\text{Scientist}(D) \wedge \exists y \, \text{Coauthor}(D,y))) \\
\times &(1 - P(\text{Scientist}(E) \wedge \exists y \, \text{Coauthor}(E,y))) \\
\times &(1 - P(\text{Scientist}(F) \wedge \exists y \, \text{Coauthor}(F,y))) \\
&\ldots
\end{aligned}$$

$\Rightarrow$ No supporting facts in database!

Closed-World Lifted Query Eval

$Q = \exists x\ \exists y\ \mathrm{Scientist}(x) \wedge \mathrm{Coauthor}(x,y)$

$$P(Q) = 1 - \Pi_{A \in \mathrm{Domain}} (1 - P(\mathrm{Scientist}(A) \wedge \exists y\ \mathrm{Coauthor}(A,y))$$

$$\begin{aligned}
= 1 - &(1 - P(\mathrm{Scientist}(A) \wedge \exists y\ \mathrm{Coauthor}(A,y)) \\
&\times (1 - P(\mathrm{Scientist}(B) \wedge \exists y\ \mathrm{Coauthor}(B,y)) \\
&\times (1 - P(\mathrm{Scientist}(C) \wedge \exists y\ \mathrm{Coauthor}(C,y)) \\
&\times (1 - P(\mathrm{Scientist}(D) \wedge \exists y\ \mathrm{Coauthor}(D,y)) \\
&\times (1 - P(\mathrm{Scientist}(E) \wedge \exists y\ \mathrm{Coauthor}(E,y)) \\
&\times (1 - P(\mathrm{Scientist}(F) \wedge \exists y\ \mathrm{Coauthor}(F,y)) \\
&\quad \ldots
\end{aligned}$$

$\Rightarrow$ No supporting facts in database!

$\Rightarrow$ Probability 0 in closed world

Closed-World Lifted Query Eval

$\boxed{Q = \exists x\ \exists y\ \text{Scientist}(x) \wedge \text{Coauthor}(x,y)}$

$$P(Q) = 1 - \prod_{A \in \text{Domain}} (1 - P(\text{Scientist}(A) \wedge \exists y\ \text{Coauthor}(A,y)))$$

$$\begin{aligned}
= 1 - &(1 - P(\text{Scientist}(A) \wedge \exists y\ \text{Coauthor}(A,y)) \\
\times &(1 - P(\text{Scientist}(B) \wedge \exists y\ \text{Coauthor}(B,y)) \\
\times &(1 - P(\text{Scientist}(C) \wedge \exists y\ \text{Coauthor}(C,y)) \\
\times &(1 - P(\text{Scientist}(D) \wedge \exists y\ \text{Coauthor}(D,y)) \\
\times &(1 - P(\text{Scientist}(E) \wedge \exists y\ \text{Coauthor}(E,y)) \\
\times &(1 - P(\text{Scientist}(F) \wedge \exists y\ \text{Coauthor}(F,y)) \\
&\ldots
\end{aligned}$$

⇨ No supporting facts in database!

⇨ Probability 0 in closed world

⇨ Ignore these sub-queries!

Closed-World Lifted Query Eval

$Q = \exists x\, \exists y\, \text{Scientist}(x) \wedge \text{Coauthor}(x,y)$

$$P(Q) = 1 - \Pi_{A \in \text{Domain}} (1 - P(\text{Scientist}(A) \wedge \exists y\, \text{Coauthor}(A,y))$$

$$\begin{aligned}
= 1 - &(1 - P(\text{Scientist}(A) \wedge \exists y\, \text{Coauthor}(A,y)) \\
&\times (1 - P(\text{Scientist}(B) \wedge \exists y\, \text{Coauthor}(B,y)) \\
&\times (1 - P(\text{Scientist}(C) \wedge \exists y\, \text{Coauthor}(C,y)) \\
&\times (1 - P(\text{Scientist}(D) \wedge \exists y\, \text{Coauthor}(D,y)) \\
&\times (1 - P(\text{Scientist}(E) \wedge \exists y\, \text{Coauthor}(E,y)) \\
&\times (1 - P(\text{Scientist}(F) \wedge \exists y\, \text{Coauthor}(F,y)) \\
&\quad \ldots
\end{aligned}$$

⇨ No supporting facts in database!

⇨ Probability 0 in closed world

⇨ Ignore these sub-queries!

Complexity linear time!

Open-World Lifted Query Eval

$Q = \exists x\ \exists y\ \text{Scientist}(x) \land \text{Coauthor}(x,y)$

$$P(Q) = 1 - \prod_{A \in \text{Domain}} (1 - P(\text{Scientist}(A) \land \exists y\ \text{Coauthor}(A,y))$$

$$\begin{aligned}
= 1 - &(1 - P(\text{Scientist}(A) \land \exists y\ \text{Coauthor}(A,y)) \\
&\times (1 - P(\text{Scientist}(B) \land \exists y\ \text{Coauthor}(B,y)) \\
&\times (1 - P(\text{Scientist}(C) \land \exists y\ \text{Coauthor}(C,y)) \\
&\times (1 - P(\text{Scientist}(D) \land \exists y\ \text{Coauthor}(D,y)) \\
&\times (1 - P(\text{Scientist}(E) \land \exists y\ \text{Coauthor}(E,y)) \\
&\times (1 - P(\text{Scientist}(F) \land \exists y\ \text{Coauthor}(F,y)) \\
&\quad \ldots
\end{aligned}$$

$\Rightarrow$ No supporting facts in database!

Open-World Lifted Query Eval

$$Q = \exists x\ \exists y\ \text{Scientist}(x) \wedge \text{Coauthor}(x,y)$$

$$P(Q) = 1 - \prod_{A \in \text{Domain}} (1 - P(\text{Scientist}(A) \wedge \exists y\ \text{Coauthor}(A,y))$$

$$\begin{aligned}
= 1 - &(1 - P(\text{Scientist}(A) \wedge \exists y\ \text{Coauthor}(A,y)) \\
&\times (1 - P(\text{Scientist}(B) \wedge \exists y\ \text{Coauthor}(B,y)) \\
&\times (1 - P(\text{Scientist}(C) \wedge \exists y\ \text{Coauthor}(C,y)) \\
&\times (1 - P(\text{Scientist}(D) \wedge \exists y\ \text{Coauthor}(D,y)) \\
&\times (1 - P(\text{Scientist}(E) \wedge \exists y\ \text{Coauthor}(E,y)) \\
&\times (1 - P(\text{Scientist}(F) \wedge \exists y\ \text{Coauthor}(F,y)) \\
&\quad \ldots
\end{aligned}$$

⇨ No supporting facts in database!

⇨ Probability λ in open world

Open-World Lifted Query Eval

$Q = \exists x\ \exists y\ \text{Scientist}(x) \land \text{Coauthor}(x,y)$

$$P(Q) = 1 - \Pi_{A \in \text{Domain}} (1 - P(\text{Scientist}(A) \land \exists y\ \text{Coauthor}(A,y))$$

$$
\begin{aligned}
= 1 - &(1 - P(\text{Scientist}(A) \land \exists y\ \text{Coauthor}(A,y)) \\
&\times (1 - P(\text{Scientist}(B) \land \exists y\ \text{Coauthor}(B,y)) \\
&\times (1 - P(\text{Scientist}(C) \land \exists y\ \text{Coauthor}(C,y)) \\
&\times (1 - P(\text{Scientist}(D) \land \exists y\ \text{Coauthor}(D,y)) \\
&\times (1 - P(\text{Scientist}(E) \land \exists y\ \text{Coauthor}(E,y)) \\
&\times (1 - P(\text{Scientist}(F) \land \exists y\ \text{Coauthor}(F,y)) \\
&\ldots
\end{aligned}
$$

⇨ No supporting facts in database!

⇨ Probability λ in open world

Complexity PTIME!

Open-World Lifted Query Eval

$Q = \exists x\, \exists y\ \mathrm{Scientist}(x) \wedge \mathrm{Coauthor}(x,y)$

$$P(Q) = 1 - \prod_{A \in \mathrm{Domain}} (1 - P(\mathrm{Scientist}(A) \wedge \exists y\ \mathrm{Coauthor}(A,y))$$

$$\begin{aligned}
= 1 - &(1 - P(\mathrm{Scientist}(A) \wedge \exists y\ \mathrm{Coauthor}(A,y)) \\
&\times (1 - P(\mathrm{Scientist}(B) \wedge \exists y\ \mathrm{Coauthor}(B,y)) \\
&\times (1 - P(\mathrm{Scientist}(C) \wedge \exists y\ \mathrm{Coauthor}(C,y)) \\
&\times (1 - P(\mathrm{Scientist}(D) \wedge \exists y\ \mathrm{Coauthor}(D,y)) \\
&\times (1 - P(\mathrm{Scientist}(E) \wedge \exists y\ \mathrm{Coauthor}(E,y)) \\
&\times (1 - P(\mathrm{Scientist}(F) \wedge \exists y\ \mathrm{Coauthor}(F,y)) \\
&\quad \ldots
\end{aligned}$$

⇒ No supporting facts in database!

⇒ Probability p in closed world

Open-World Lifted Query Eval

$$Q = \exists x\, \exists y\, \text{Scientist}(x) \land \text{Coauthor}(x,y)$$

$$P(Q) = 1 - \Pi_{A \in \text{Domain}} (1 - P(\text{Scientist}(A) \land \exists y\, \text{Coauthor}(A,y))$$

$$\begin{aligned}
= 1 - (1 - &P(\text{Scientist}(A) \land \exists y\, \text{Coauthor}(A,y)) \\
\times (1 - &P(\text{Scientist}(B) \land \exists y\, \text{Coauthor}(B,y)) \\
\times (1 - &P(\text{Scientist}(C) \land \exists y\, \text{Coauthor}(C,y)) \\
\times (1 - &P(\text{Scientist}(D) \land \exists y\, \text{Coauthor}(D,y)) \\
\times (1 - &P(\text{Scientist}(E) \land \exists y\, \text{Coauthor}(E,y)) \\
\times (1 - &P(\text{Scientist}(F) \land \exists y\, \text{Coauthor}(F,y)) \\
&\ldots
\end{aligned}$$

No supporting facts in database!

Probability p in closed world

All together, probability $(1-p)^k$

Exploit symmetry

Lifted inference

Open-World Lifted Query Eval

$Q = \exists x\, \exists y\, \text{Scientist}(x) \wedge \text{Coauthor}(x,y)$

$$P(Q) = 1 - \prod_{A \in \text{Domain}} (1 - P(\text{Scientist}(A) \wedge \exists y\, \text{Coauthor}(A,y))$$

$$\begin{aligned}
= 1 - &(1 - P(\text{Scientist}(A) \wedge \exists y\, \text{Coauthor}(A,y)) \\
\times &(1 - P(\text{Scientist}(B) \wedge \exists y\, \text{Coauthor}(B,y)) \\
\times &(1 - P(\text{Scientist}(C) \wedge \exists y\, \text{Coauthor}(C,y)) \\
\times &(1 - P(\text{Scientist}(D) \wedge \exists y\, \text{Coauthor}(D,y)) \\
\times &(1 - P(\text{Scientist}(E) \wedge \exists y\, \text{Coauthor}(E,y)) \\
\times &(1 - P(\text{Scientist}(F) \wedge \exists y\, \text{Coauthor}(F,y)) \\
&\ldots
\end{aligned}$$

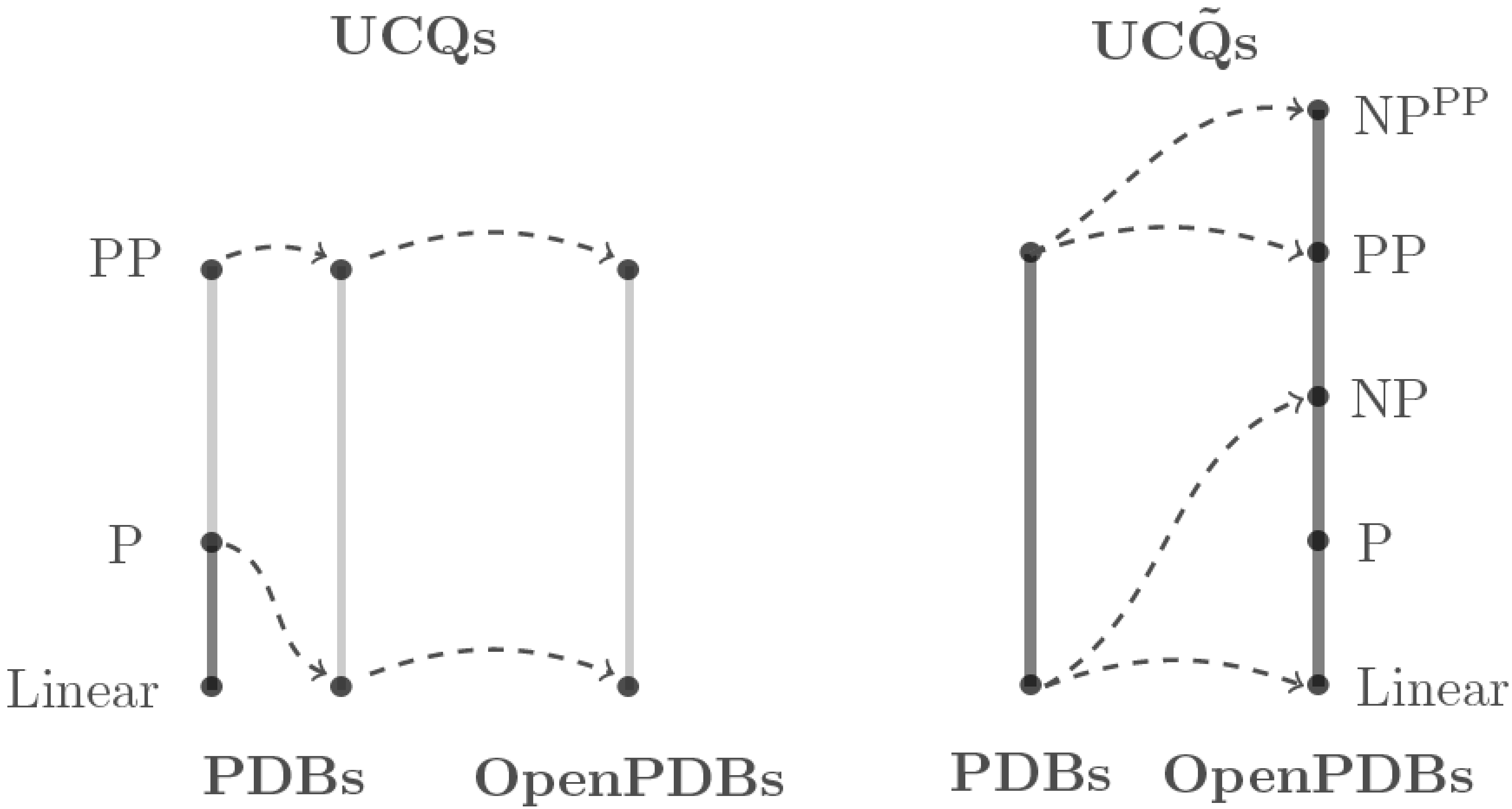
⇨ No supporting facts
in database!

⇨ Probability p in closed world

⇨ All together, probability $(1-p)^k$
Exploit symmetry
Lifted inference

Complexity linear time!

Complexity Results



$\mathrm{Linear} \subseteq \mathrm{P} \subseteq \mathrm{NP} \subseteq \mathrm{PP} \subseteq \mathrm{P}^{\mathrm{PP}} \subseteq \mathrm{NP}^{\mathrm{PP}} \subseteq \mathrm{PSpace} \subseteq \mathrm{ExpTime}$

[Ceylan'16]

Implement PDB Query in SQL

$Q = \exists x\, P(x) \wedge Q(x)$

- Convert to nested SQL recursively
- Open-world existential quantification

```
SELECT (1.0-(1.0-pUse)*power(1.0-0.0001,(4-ct))) AS pUse
FROM
  (SELECT ior(COALESCE(pUse,0)) AS pUse,
          count(*) AS ct
   FROM SQL(conjunction)
```

`0.0001` = open-world probability; `4` = # open-world query instances
`ior` = Independent OR aggregate function

- Conjunction

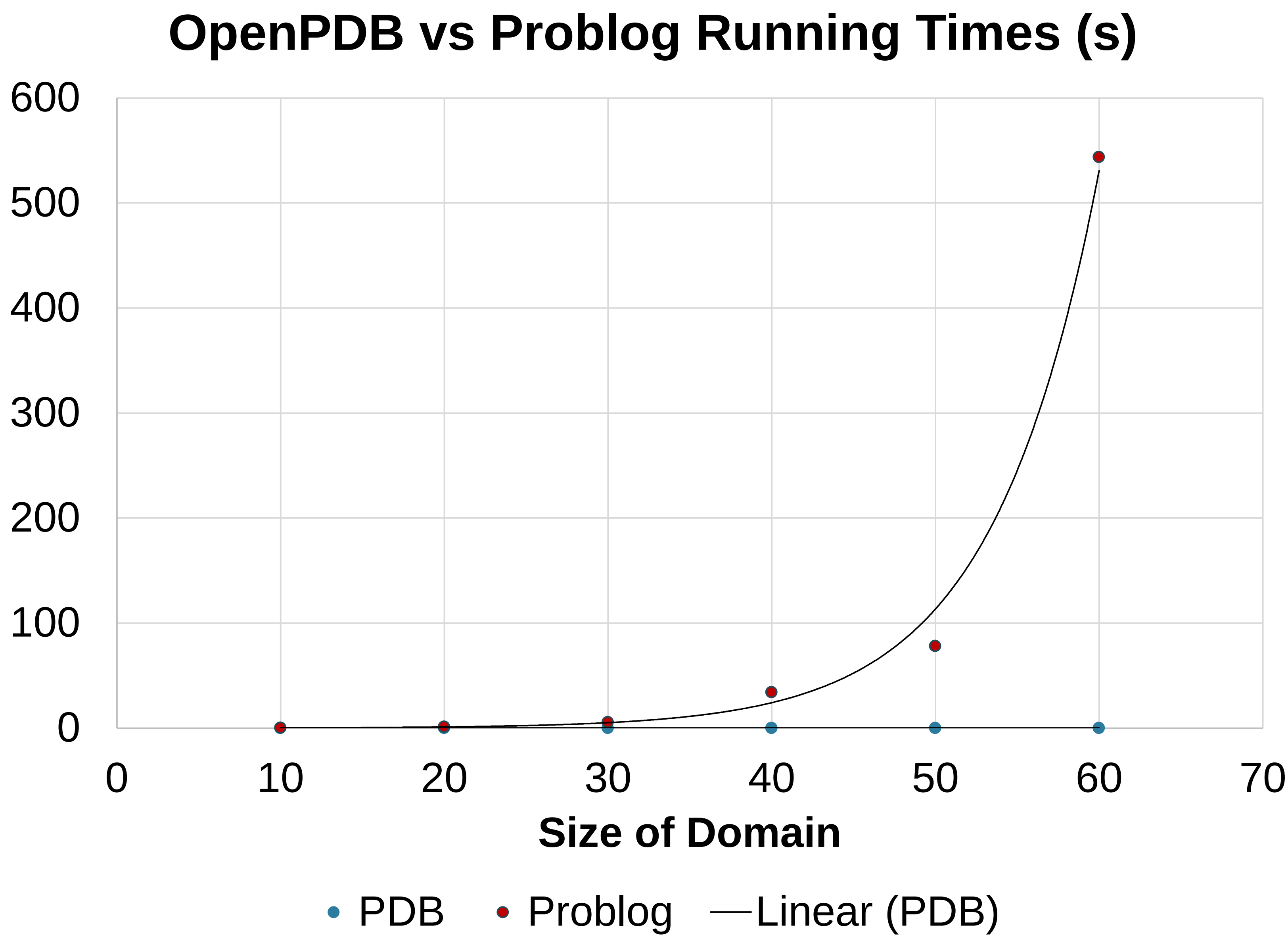
```
SELECT q9.c5,
  COALESCE(q9.pUse,λ)*COALESCE(q10.pUse,λ) AS pUse
FROM
  SQL(Q(X)) OUTER JOIN SQL(P(X))
```

```
SELECT Q.v0 AS c5,
       p AS pUse
        FROM Q
```

- Run as single PostgreSQL query!

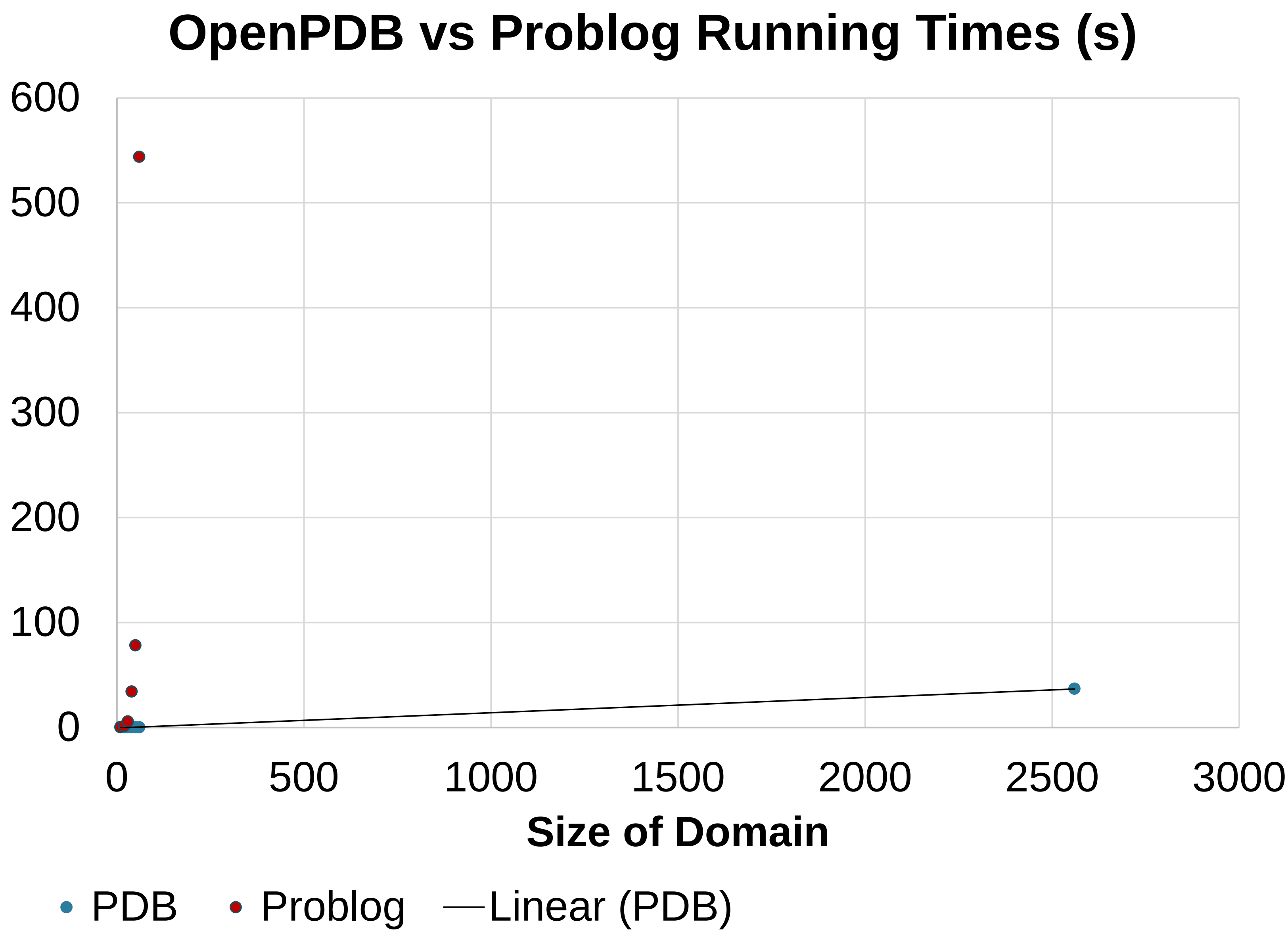
[Tal Friedman, Eric Gribkoff]

[Tal Friedman]

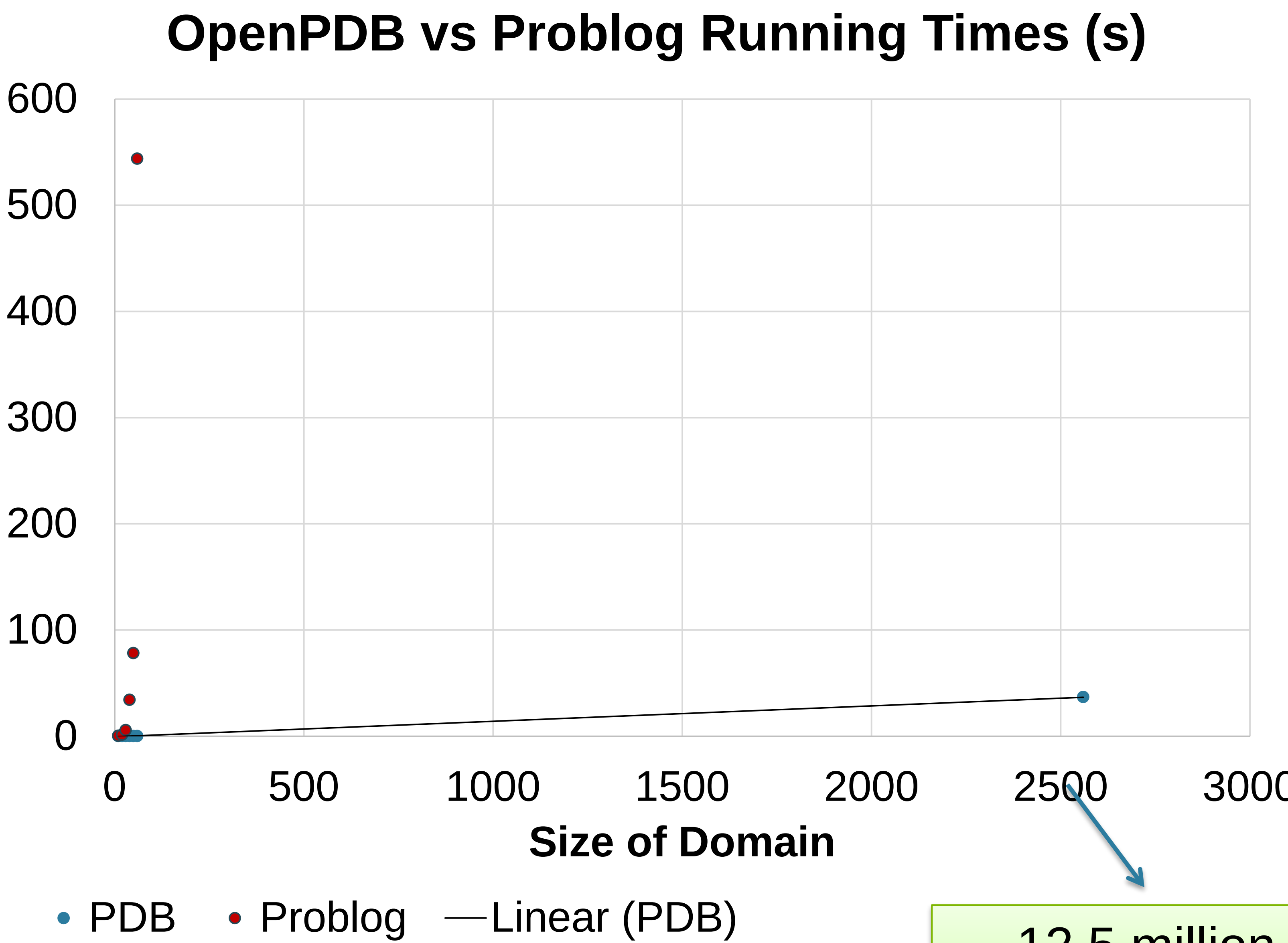


Out of memory trying to run the ProbLog query with 70 constants in domain

[Tal Friedman]



[Tal Friedman]



12.5 million random variables!

What *is the broader picture?*

First-Order Model Counting

Model Counting

- Model = solution to a propositional logic formula Δ
- Model counting = #SAT

Δ = (Rain ⇒ Cloudy)

Rain	Cloudy	Model?
T	T	Yes
T	F	No
F	T	Yes
F	F	Yes

+

#SAT = 3

Model Counting

- Model = solution to a propositional logic formula Δ
- Model counting = #SAT

Δ = (Rain ⇒ Cloudy)

Rain	Cloudy	Model?
T	T	Yes
T	F	No
F	T	Yes
F	F	Yes

+ ———

#SAT = 3

[Valiant] #P-hard, even for 2CNF

Weighted Model Count

- Weights for assignments to variables
- Model weight = product of variable weights

Δ = (Rain ⇒ Cloudy)

Rain	Cloudy	Model?
T	T	Yes
T	F	No
F	T	Yes
F	F	Yes

Weighted Model Count

- Weights for assignments to variables
- Model weight = product of variable weights

$\Delta = (\text{Rain} \Rightarrow \text{Cloudy})$

Rain

w(R)	w(¬R)
1	2

Cloudy

w(C)	w(¬C)
3	5

Rain	Cloudy	Model?
T	T	Yes
T	F	No
F	T	Yes
F	F	Yes

Weighted Model Count

- Weights for assignments to variables
- Model weight = product of variable weights

$\Delta = (\text{Rain} \Rightarrow \text{Cloudy})$

Rain

w(R)	w(¬R)
1	2

Cloudy

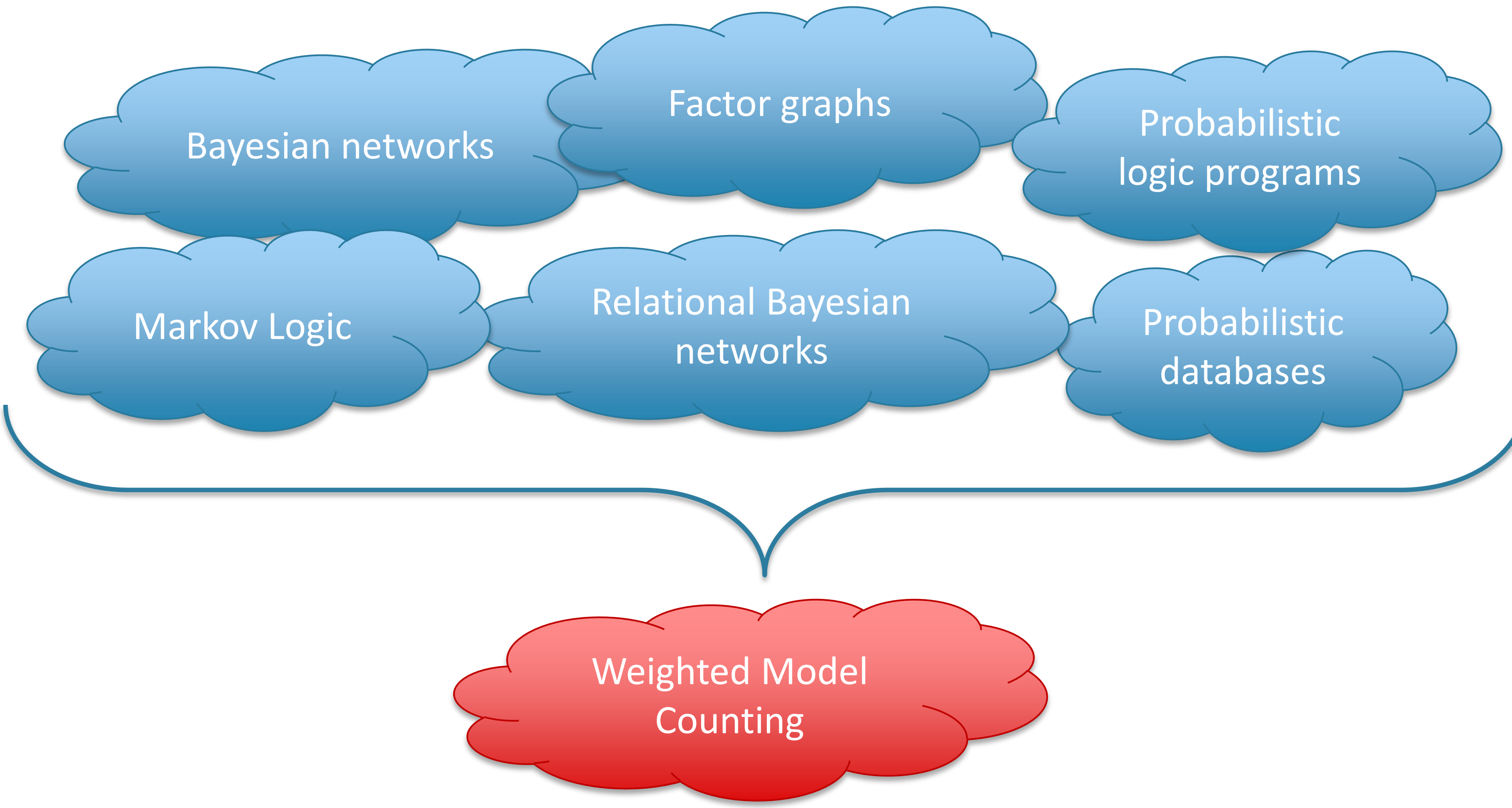
w(C)	w(¬C)
3	5

Rain	Cloudy	Model?	Weight
T	T	Yes	1 * 3 = 3
T	F	No	0
F	T	Yes	2 * 3 = 6
F	F	Yes	2 * 5 = 10

+

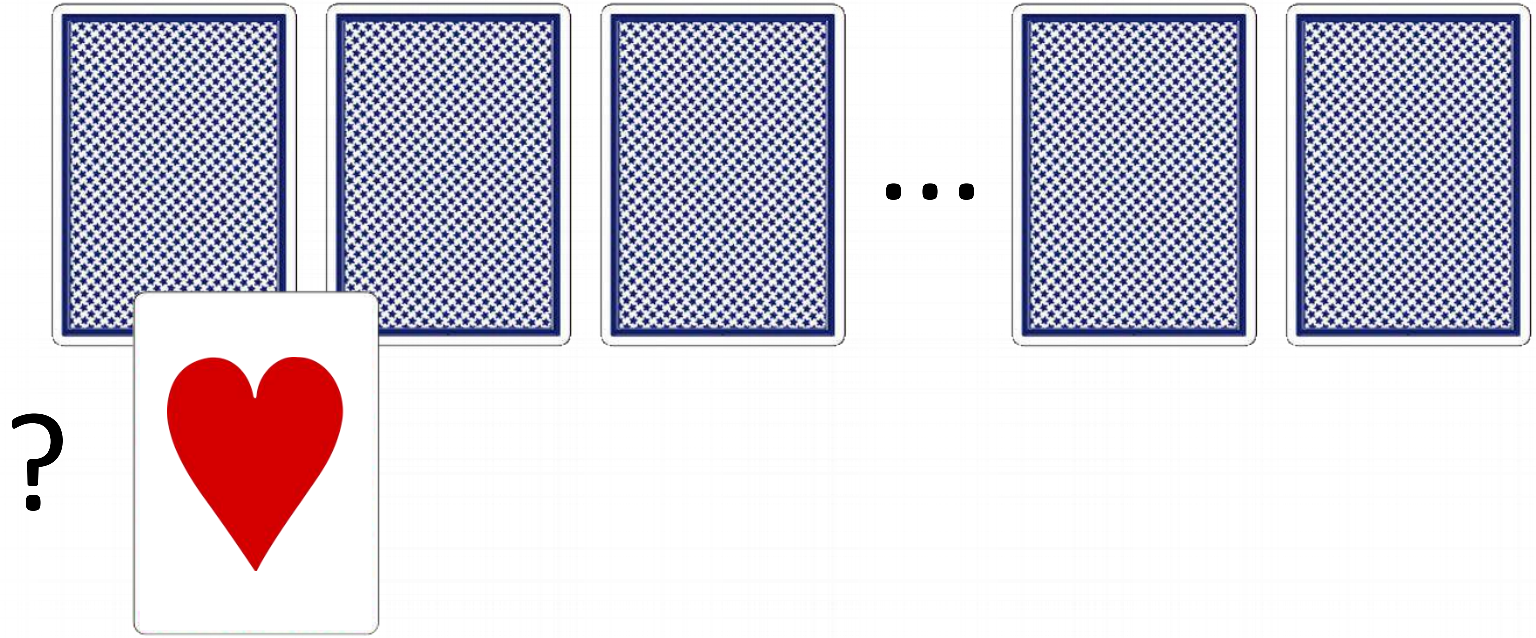
WMC = 19

Assembly language for probabilistic reasoning



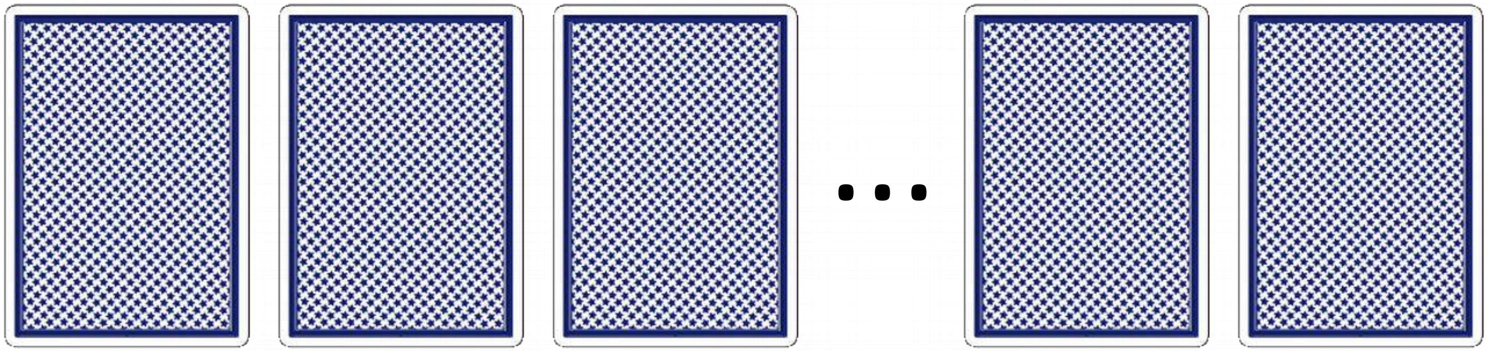
[Chavira 2006, Chavira 2008, Sang 2005, Fierens 2015]

Simple Reasoning Problem



Probability that Card1 is Hearts? 1/4

[Van den Broeck; AAAI-KRR’15]



Model distribution by FOMC:

$$\Delta = \quad \forall p, \exists c, \mathrm{Card}(p,c)$$
$$\forall c, \exists p, \mathrm{Card}(p,c)$$
$$\forall p, \forall c, \forall c', \mathrm{Card}(p,c) \wedge \mathrm{Card}(p,c') \Rightarrow c = c'$$

[Van den Broeck 2015]

Beyond NP Pipeline for #P

Reduce to propositional model counting:

[Van den Broeck 2015]

Beyond NP Pipeline for #P

Reduce to propositional model counting:

$$
\begin{aligned}
\Delta = \; & \text{Card}(A\heartsuit, p_1) \vee \ldots \vee \text{Card}(2\clubsuit, p_1) \\
& \text{Card}(A\heartsuit, p_2) \vee \ldots \vee \text{Card}(2\clubsuit, p_2) \\
& \ldots \\
& \text{Card}(A\heartsuit, p_1) \vee \ldots \vee \text{Card}(A\heartsuit, p_{52}) \\
& \text{Card}(K\heartsuit, p_1) \vee \ldots \vee \text{Card}(K\heartsuit, p_{52}) \\
& \ldots \\
& \neg\text{Card}(A\heartsuit, p_1) \vee \neg\text{Card}(A\heartsuit, p_2) \\
& \neg\text{Card}(A\heartsuit, p_1) \vee \neg\text{Card}(A\heartsuit, p_3) \\
& \ldots
\end{aligned}
$$

[Van den Broeck 2015]

Beyond NP Pipeline for #P

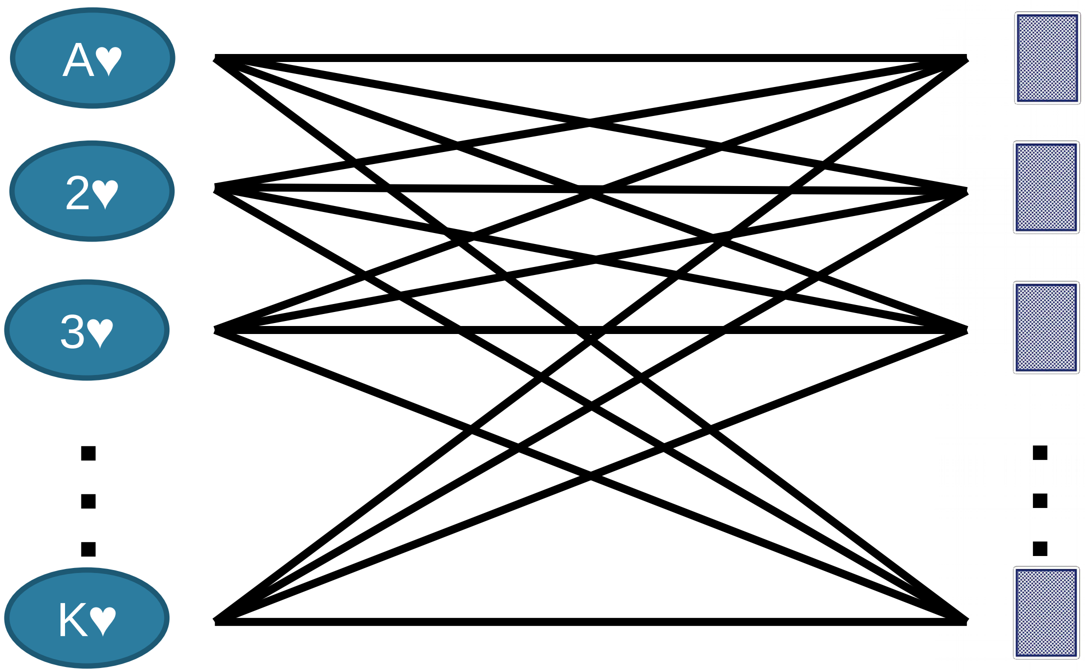
Reduce to propositional model counting:

$$
\begin{aligned}
\Delta = \; & \text{Card}(A\heartsuit,p_1) \lor \ldots \lor \text{Card}(2\clubsuit,p_1) \\
& \text{Card}(A\heartsuit,p_2) \lor \ldots \lor \text{Card}(2\clubsuit,p_2) \\
& \ldots \\
& \text{Card}(A\heartsuit,p_1) \lor \ldots \lor \text{Card}(A\heartsuit,p_{52}) \\
& \text{Card}(K\heartsuit,p_1) \lor \ldots \lor \text{Card}(K\heartsuit,p_{52}) \\
& \ldots \\
& \neg\text{Card}(A\heartsuit,p_1) \lor \neg\text{Card}(A\heartsuit,p_2) \\
& \neg\text{Card}(A\heartsuit,p_1) \lor \neg\text{Card}(A\heartsuit,p_3) \\
& \ldots
\end{aligned}
$$

What will happen?

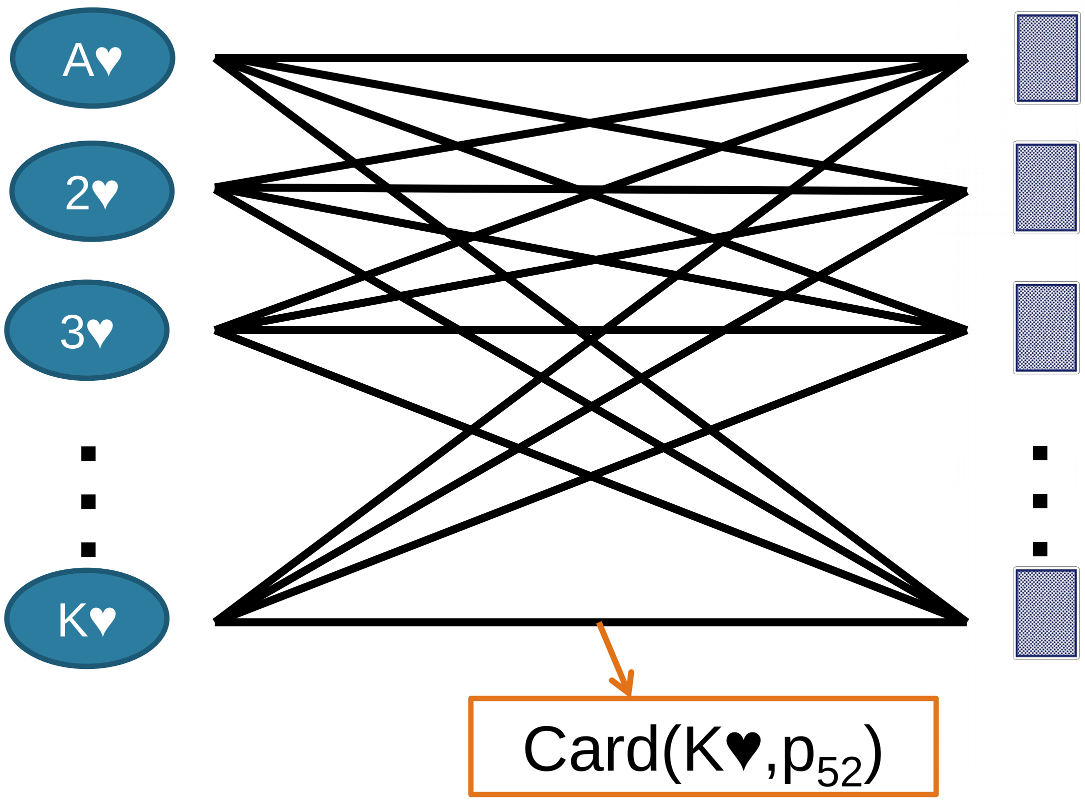
[Van den Broeck 2015]

Deck of Cards Graphically



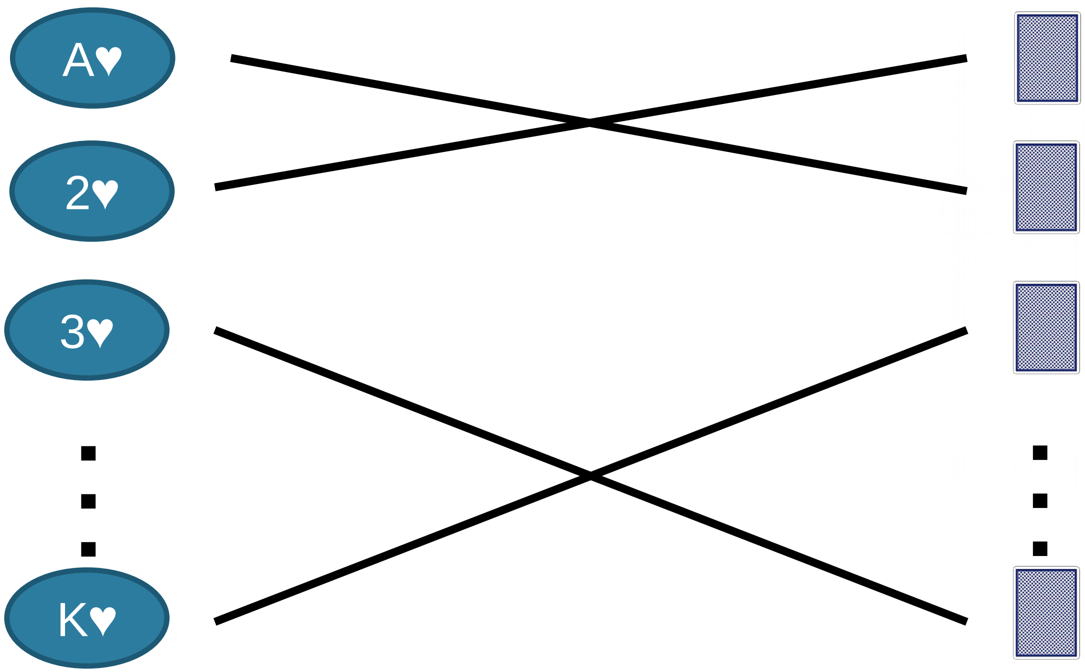
[VdB'15]

Deck of Cards Graphically



[VdB'15]

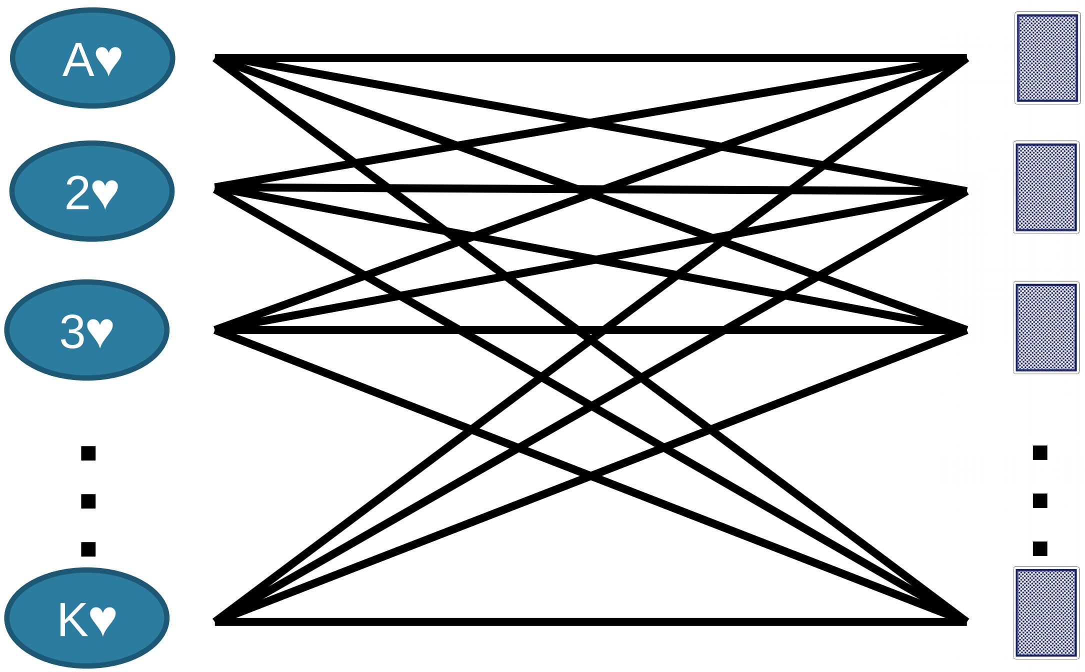
Deck of Cards Graphically



One model/*perfect matching*

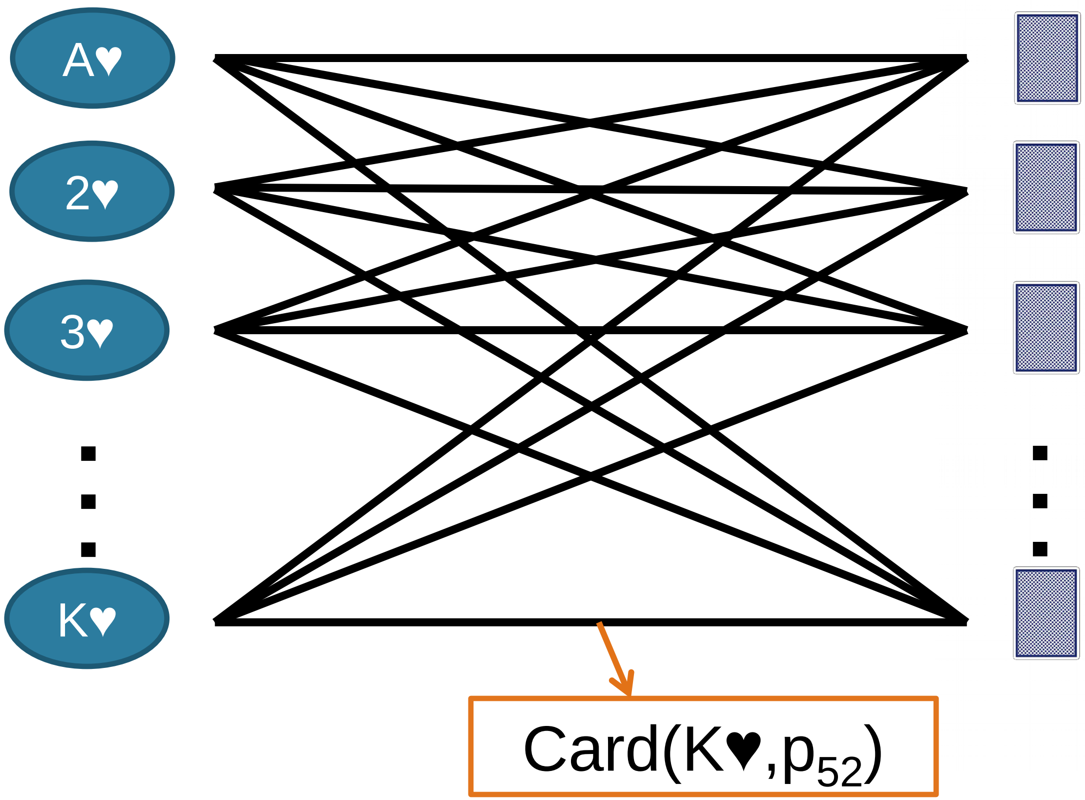
[VdB'15]

Deck of Cards Graphically



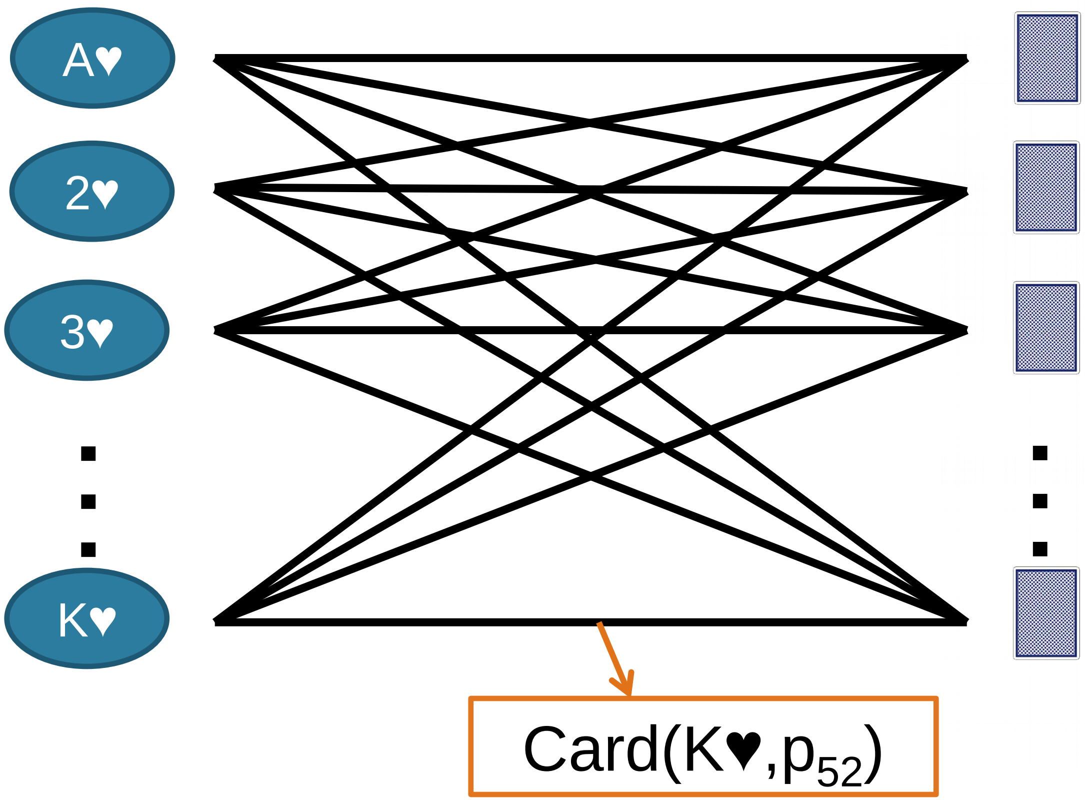
[VdB'15]

Deck of Cards Graphically



[VdB'15]

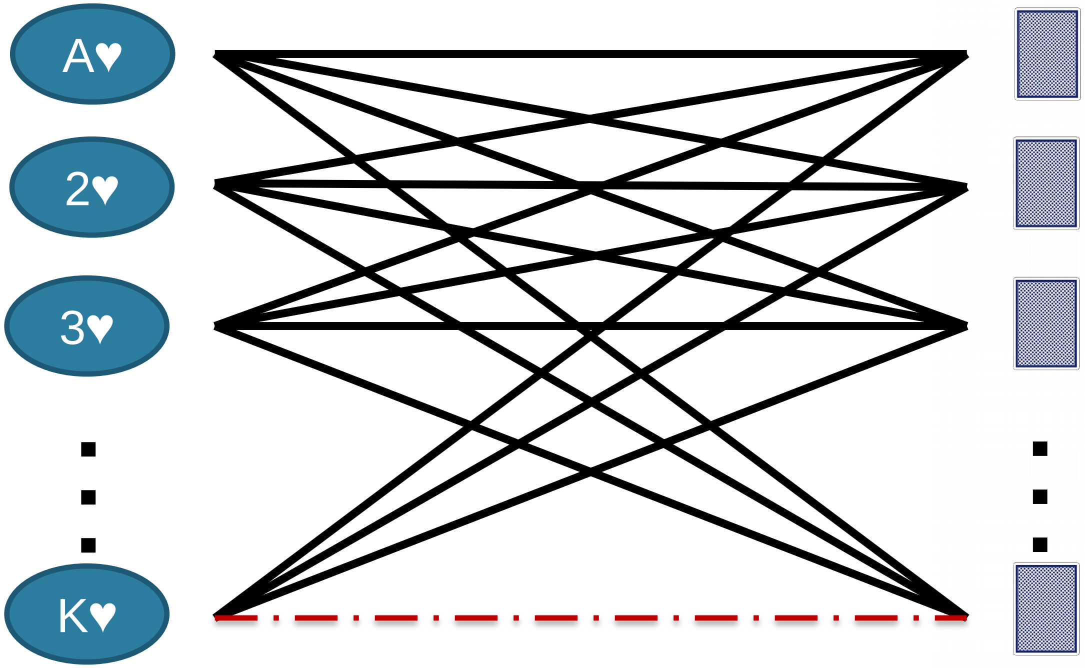
Deck of Cards Graphically



Model counting: How many *perfect matchings*?

[VdB’15]

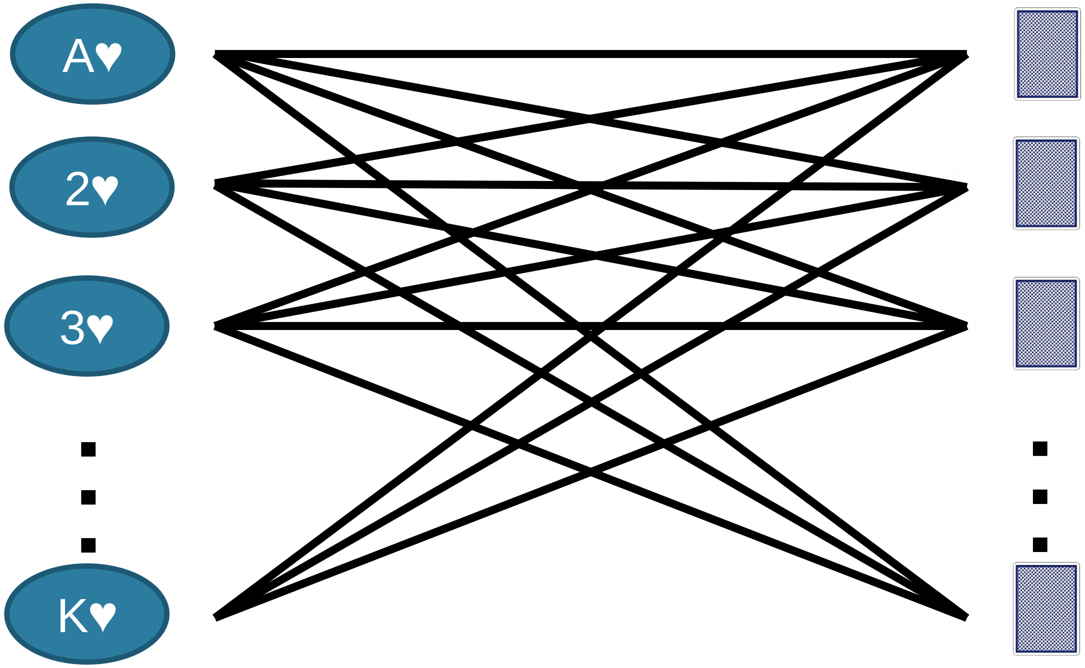
Deck of Cards Graphically



What if I set
w(Card(K♥,p_{52})) = 0?

[VdB'15]

Deck of Cards Graphically



What if I set

w(Card(K♥,p_{52})) = 0?

[VdB'15]

Observations

- Weight function = bipartite graph
- # models = # perfect matchings
- Problem is **#P**-complete! ☹

[VdB'15]

Observations

- Weight function = bipartite graph
- # models = # perfect matchings
- Problem is **#P**-complete! ☹

No propositional WMC solver can handle cards problem efficiently!

[VdB'15]

Observations

- Weight function = bipartite graph
- # models = # perfect matchings
- Problem is **#P**-complete! ☹

No propositional WMC solver can handle cards problem efficiently!

What is going on here?

[VdB'15]

Symmetric Weighted FOMC

No database! No literal-specific weights!

Def. A *weighted vocabulary* is ($\mathbf{R}$, $\mathbf{w}$), where

- $\mathbf{R} = (R_1, R_2, \ldots, R_k)$ = relational vocabulary
- $\mathbf{w} = (w_1, w_2, \ldots, w_k)$ = weights
- Implicit weights: $w(R_i(t)) = w_i$

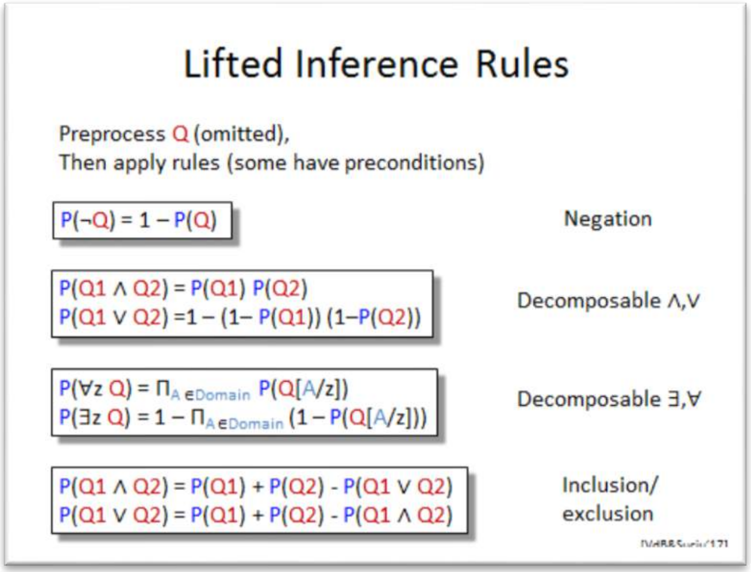
Special case: $w_i = 1$ is model counting

Complexity in terms of domain size n

FOMC Inference Rules

- Simplification to ∃,∀ rules:

For example:
$P(\forall z\ Q) = P(Q[C_1/z])^{|Domain|}$

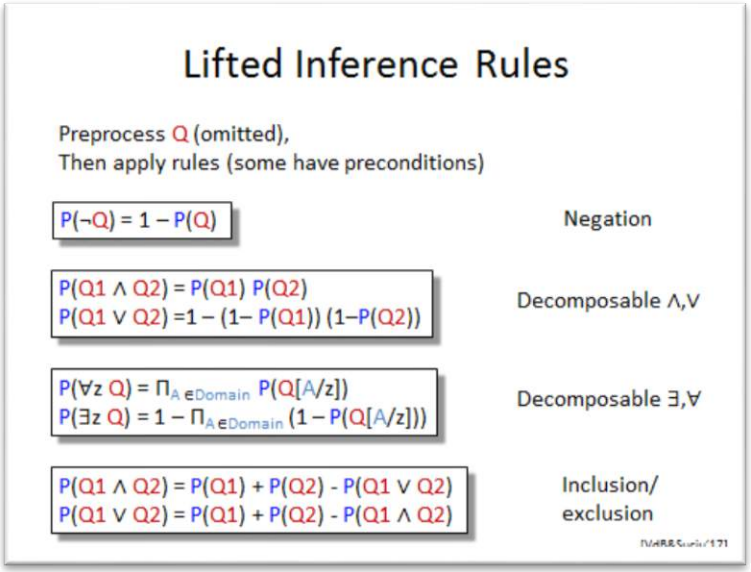


[VdB’11]

FOMC Inference Rules

- Simplification to ∃,∀ rules:

  For example:
  $P(\forall z\ Q) = P(Q[C_1/z])^{|\text{Domain}|}$

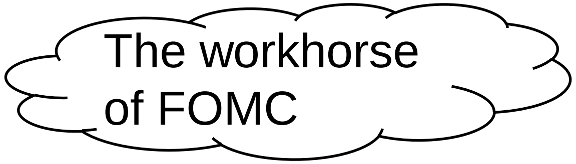


Lifted Inference Rules

Preprocess Q (omitted),
Then apply rules (some have preconditions)

$P(\neg Q) = 1 - P(Q)$	Negation
$P(Q1 \wedge Q2) = P(Q1)\,P(Q2)$ $P(Q1 \vee Q2) = 1 - (1 - P(Q1))\,(1 - P(Q2))$	Decomposable ∧,∨
$P(\forall z\ Q) = \Pi_{A \in Domain}\ P(Q[A/z])$ $P(\exists z\ Q) = 1 - \Pi_{A \in Domain}\ (1 - P(Q[A/z]))$	Decomposable ∃,∀
$P(Q1 \wedge Q2) = P(Q1) + P(Q2) - P(Q1 \vee Q2)$ $P(Q1 \vee Q2) = P(Q1) + P(Q2) - P(Q1 \wedge Q2)$	Inclusion/ exclusion

- A powerful new inference rule: *atom counting*
  Only possible with symmetric weights
  Intuition: **Remove unary relations**



[VdB’11]

First-Order Model Counting: Example

Δ = $\forall$x ,y $\in$ **People**: Smokes(x) $\wedge$ Friends(x,y) $\Rightarrow$ Smokes(y)

[Van den Broeck 2015]

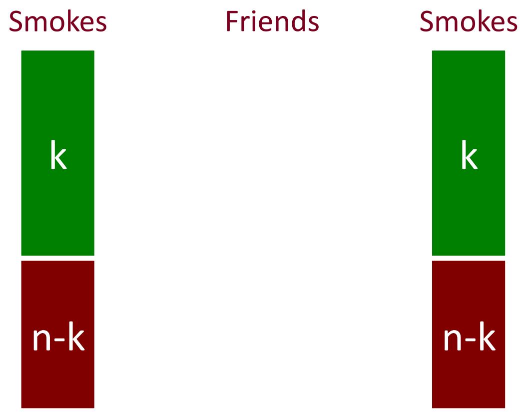
First-Order Model Counting: Example

Δ = ∀x ,y ∈ **People**: Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

- If we know **D** precisely: who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...



[Van den Broeck 2015]

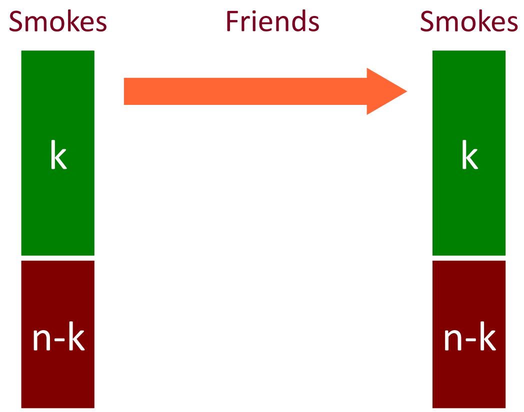
First-Order Model Counting: Example

$\color{red}{\Delta}$ = ∀x ,y ∈ **People**: Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

- If we know **D** precisely: who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...



[Van den Broeck 2015]

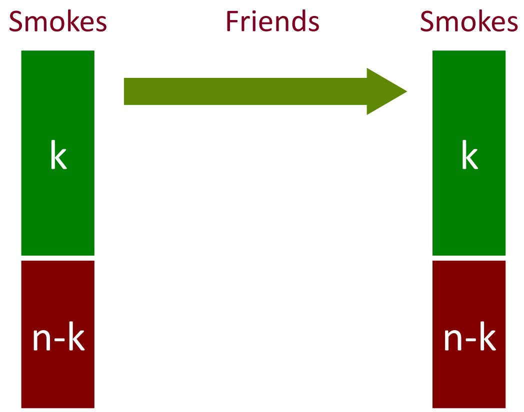
First-Order Model Counting: Example

Δ = ∀x ,y ∈ **People**: Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

- If we know **D** precisely: who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...



[Van den Broeck 2015]

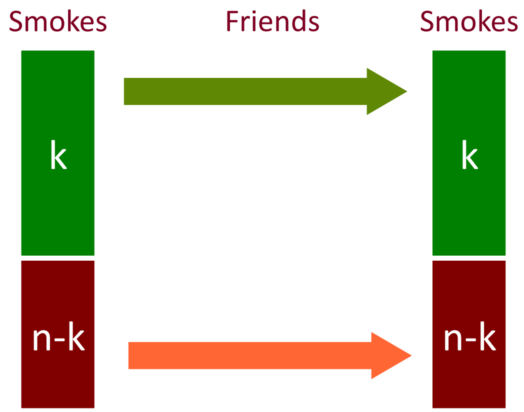
First-Order Model Counting: Example

Δ = ∀x ,y ∈ **People**: Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

- If we know **D** precisely: who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...



[Van den Broeck 2015]

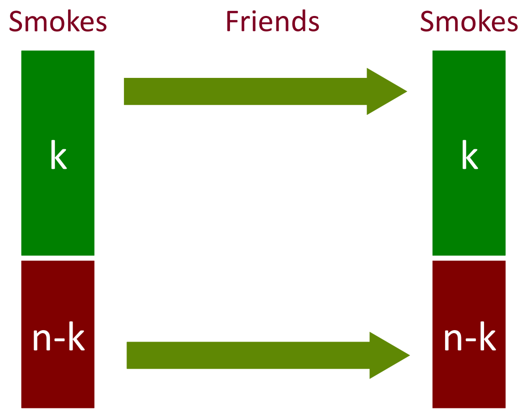
First-Order Model Counting: Example

Δ = ∀x ,y ∈ **People**: Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

- If we know **D** precisely: who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...



[Van den Broeck 2015]

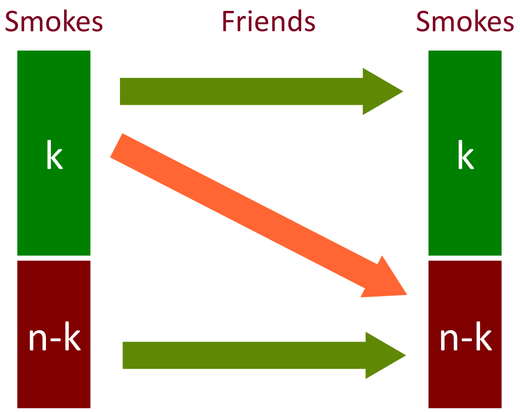
First-Order Model Counting: Example

Δ = ∀x ,y ∈ **People**: Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

- If we know **D** precisely: who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...



[Van den Broeck 2015]

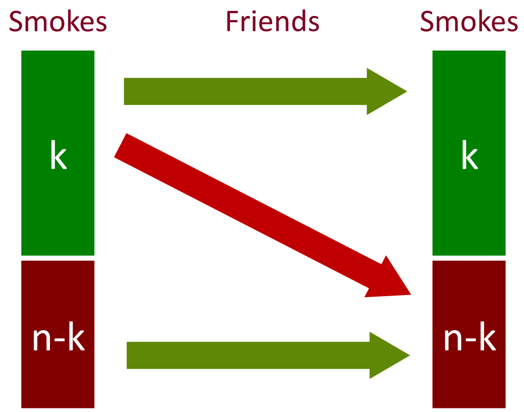
First-Order Model Counting: Example

Δ = ∀x ,y ∈ **People**: Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

- If we know **D** precisely: who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...



[Van den Broeck 2015]

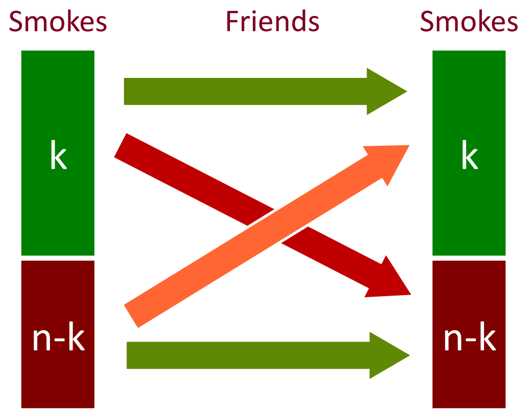
First-Order Model Counting: Example

Δ = ∀x ,y ∈ **People**: Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

- If we know **D** precisely: who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...



[Van den Broeck 2015]

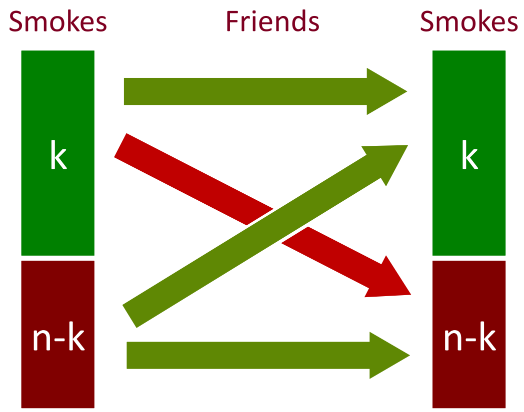
First-Order Model Counting: Example

Δ = ∀x ,y ∈ **People**: Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

- If we know **D** precisely: who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...



[Van den Broeck 2015]

First-Order Model Counting: Example

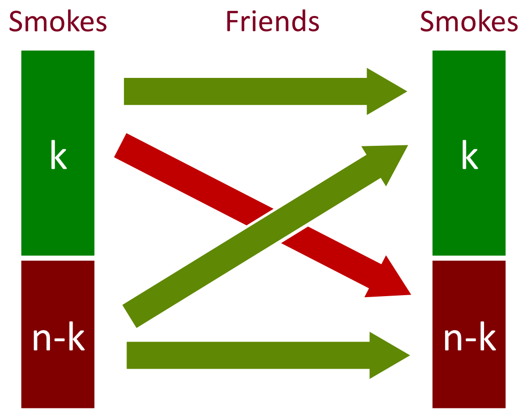
Δ = ∀x ,y ∈ **People**: Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

- If we know **D** precisely: who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...

→ $2^{n^2-k(n-k)}$ models



[Van den Broeck 2015]

First-Order Model Counting: Example

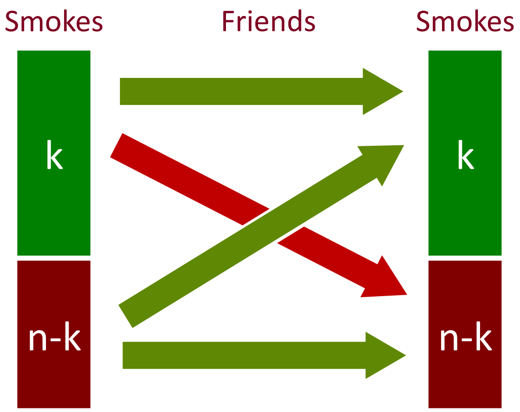
Δ = ∀x ,y ∈ **People**: Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

- If we know **D** precisely: who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...

→ $2^{n^2-k(n-k)}$ models



- If we know that there are *k* smokers?

[Van den Broeck 2015]

First-Order Model Counting: Example

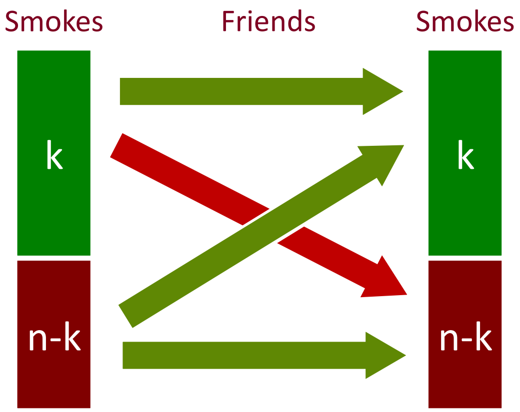
Δ = ∀x ,y ∈ **People**: Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

- If we know **D** precisely: who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...

→ $2^{n^2-k(n-k)}$ models



- If we know that there are *k* smokers?

→ $\binom{n}{k} 2^{n^2-k(n-k)}$ models

[Van den Broeck 2015]

First-Order Model Counting: Example

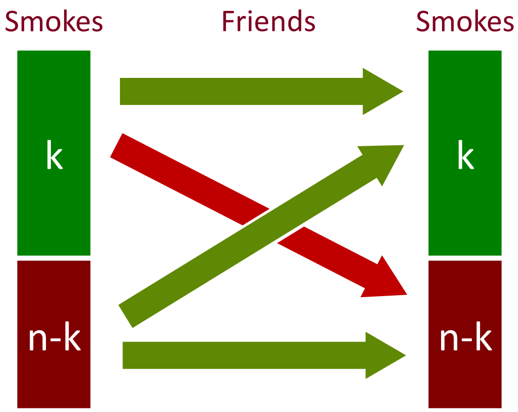
Δ = ∀x ,y ∈ **People**: Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

- If we know **D** precisely: who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...

→ $2^{n^2-k(n-k)}$ models



- If we know that there are *k* smokers? → $\binom{n}{k} 2^{n^2-k(n-k)}$ models

- In total…

[Van den Broeck 2015]

First-Order Model Counting: Example

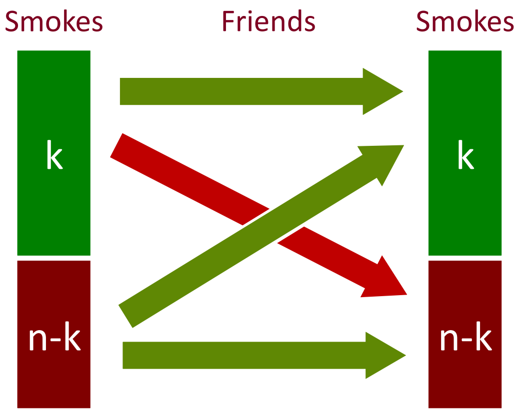
Δ = ∀x ,y ∈ **People**: Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

- If we know **D** precisely: who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...

→ $2^{n^2 - k(n-k)}$ models

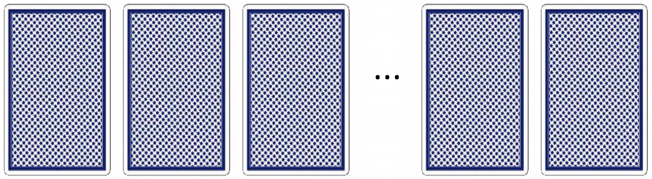


- If we know that there are *k* smokers? → $\binom{n}{k} 2^{n^2 - k(n-k)}$ models

- In total... → $\sum_{k=0}^{n} \binom{n}{k} 2^{n^2 - k(n-k)}$ models

[Van den Broeck 2015]

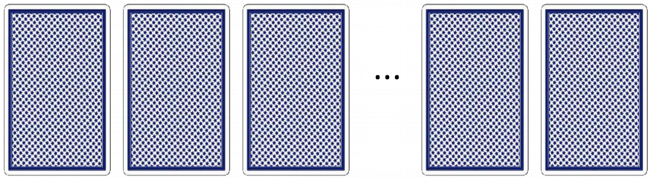
Playing Cards Revisited



$$\forall p, \exists c, \text{Card}(p,c)$$
$$\forall c, \exists p, \text{Card}(p,c)$$
$$\forall p, \forall c, \forall c', \text{Card}(p,c) \wedge \text{Card}(p,c') \Rightarrow c = c'$$

[Van den Broeck.; AAAI-KR’15]

Playing Cards Revisited



∀p, ∃c, Card(p,c)

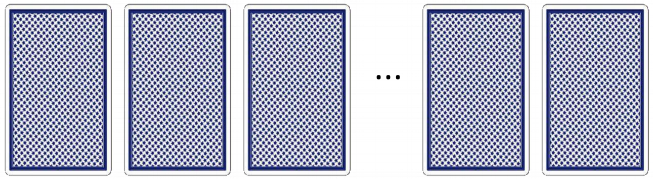
∀c, ∃p, Card(p,c)

∀p, ∀c, ∀c', Card(p,c) ∧ Card(p,c') ⇒ c = c'

$$\#SAT = \sum_{k=0}^{n} \binom{n}{k} \sum_{l=0}^{n} \binom{n}{l} (l+1)^k (-1)^{2n-k-l} = n!$$

[Van den Broeck.; AAAI-KR'15]

Playing Cards Revisited



∀p, ∃c, Card(p,c)
∀c, ∃p, Card(p,c)
∀p, ∀c, ∀c’, Card(p,c) ∧ Card(p,c’) ⇒ c = c’

$$\#SAT = \sum_{k=0}^{n} \binom{n}{k} \sum_{l=0}^{n} \binom{n}{l} (l+1)^k \, (-1)^{2n-k-l} = n!$$

Computed in time polynomial in n

[Van den Broeck.; AAAI-KR’15]

Open-World Lifted Query Eval

$Q = \exists x\ \exists y\ Smoker(x) \wedge Friend(x,y)$

$$P(Q) = 1 - \Pi_{A \in Domain} (1 - P(Scientist(A) \wedge \exists y\ Coauthor(A,y))$$

$= 1 - (1 - P(Scientist(A) \wedge \exists y\ Coauthor(A,y))$
$\quad \times (1 - P(Scientist(B) \wedge \exists y\ Coauthor(B,y))$
$\quad \times (1 - P(Scientist(C) \wedge \exists y\ Coauthor(C,y))$
$\quad \times (1 - P(Scientist(D) \wedge \exists y\ Coauthor(D,y))$
$\quad \times (1 - P(Scientist(E) \wedge \exists y\ Coauthor(E,y))$
$\quad \times (1 - P(Scientist(F) \wedge \exists y\ Coauthor(F,y))$
$\quad \dots$

$\Rightarrow$ All together, probability $(1-p)^k$

Open-World Lifted Query Eval

$Q = \exists x\ \exists y\ \text{Smoker}(x) \wedge \text{Friend}(x,y)$

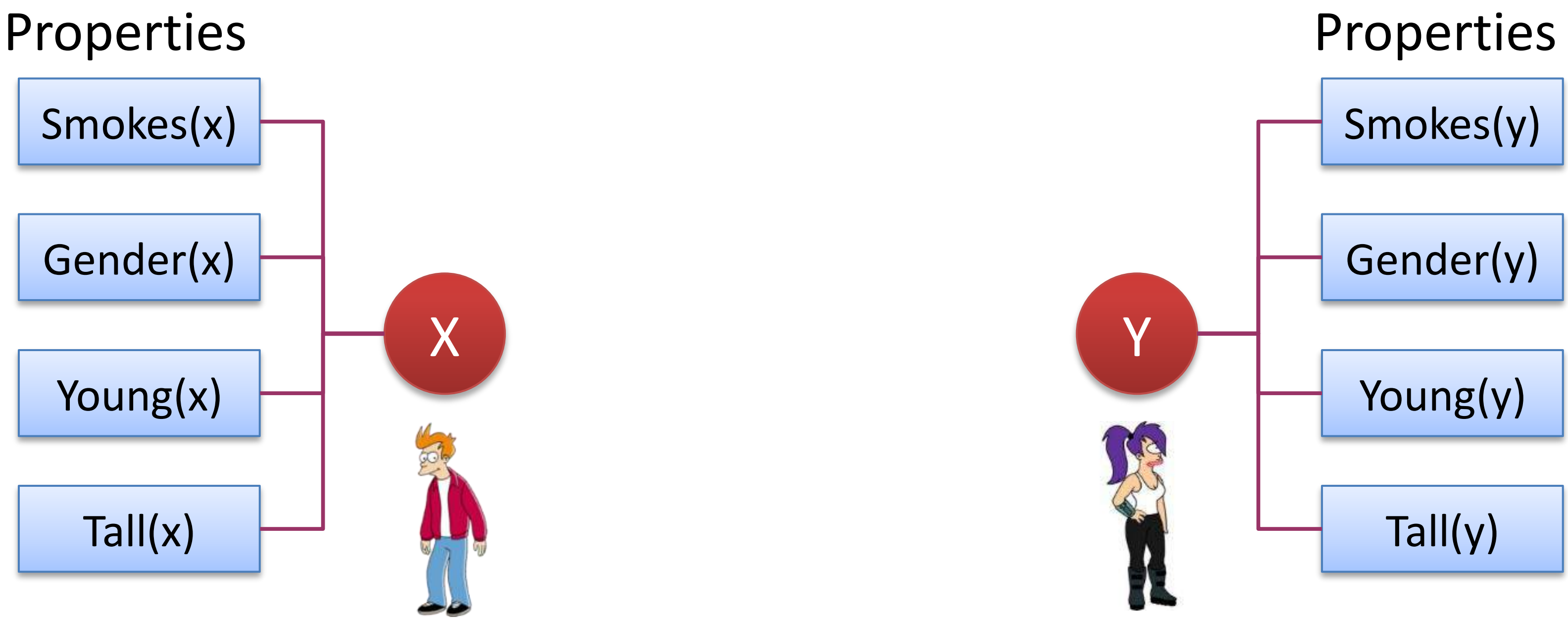
$P(Q) = 1 - \Pi_{A \in \text{Domain}} (1 - P(\text{Scientist}(A) \wedge \exists y\ \text{Coauthor}(A,y))$

$= 1 - (1 - P(\text{Scientist}(A) \wedge \exists y\ \text{Coauthor}(A,y))$
$\quad \times (1 - P(\text{Scientist}(B) \wedge \exists y\ \text{Coauthor}(B,y))$
$\quad \times (1 - P(\text{Scientist}(C) \wedge \exists y\ \text{Coauthor}(C,y))$
$\quad \times (1 - P(\text{Scientist}(D) \wedge \exists y\ \text{Coauthor}(D,y))$
$\quad \times (1 - P(\text{Scientist}(E) \wedge \exists y\ \text{Coauthor}(E,y))$
$\quad \times (1 - P(\text{Scientist}(F) \wedge \exists y\ \text{Coauthor}(F,y))$
$\quad \ldots$

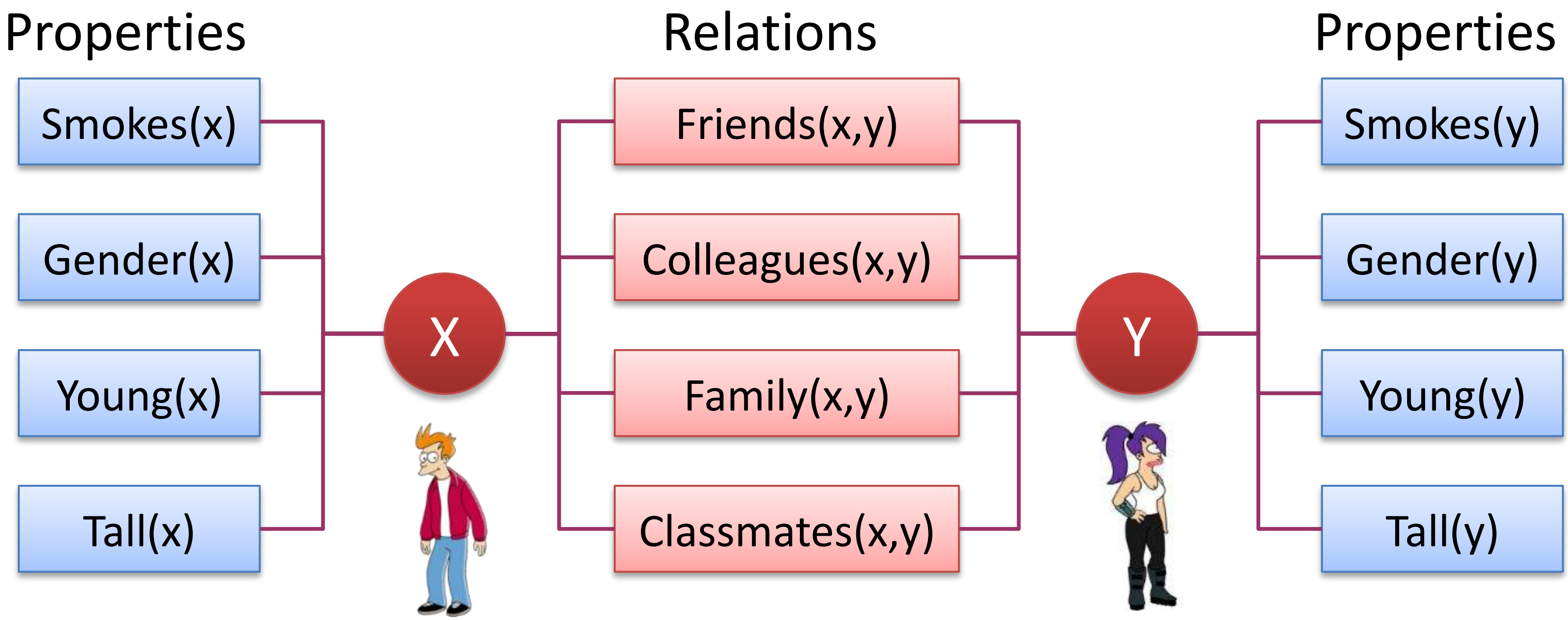
$\Rightarrow$ All together, probability $(1-p)^k$

Open-world query evaluation on empty db
= Symmetric First-Order Model Counting

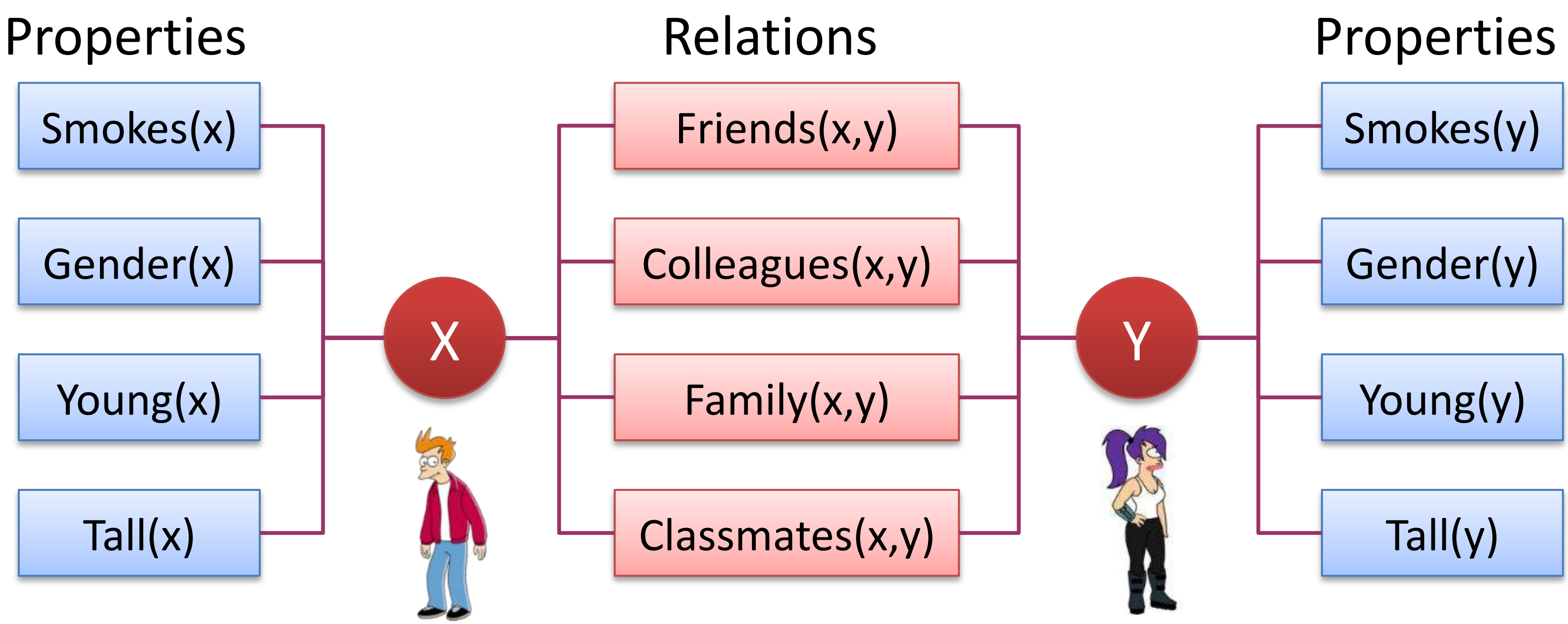
FO^2 is liftable!



FO^2 is liftable!

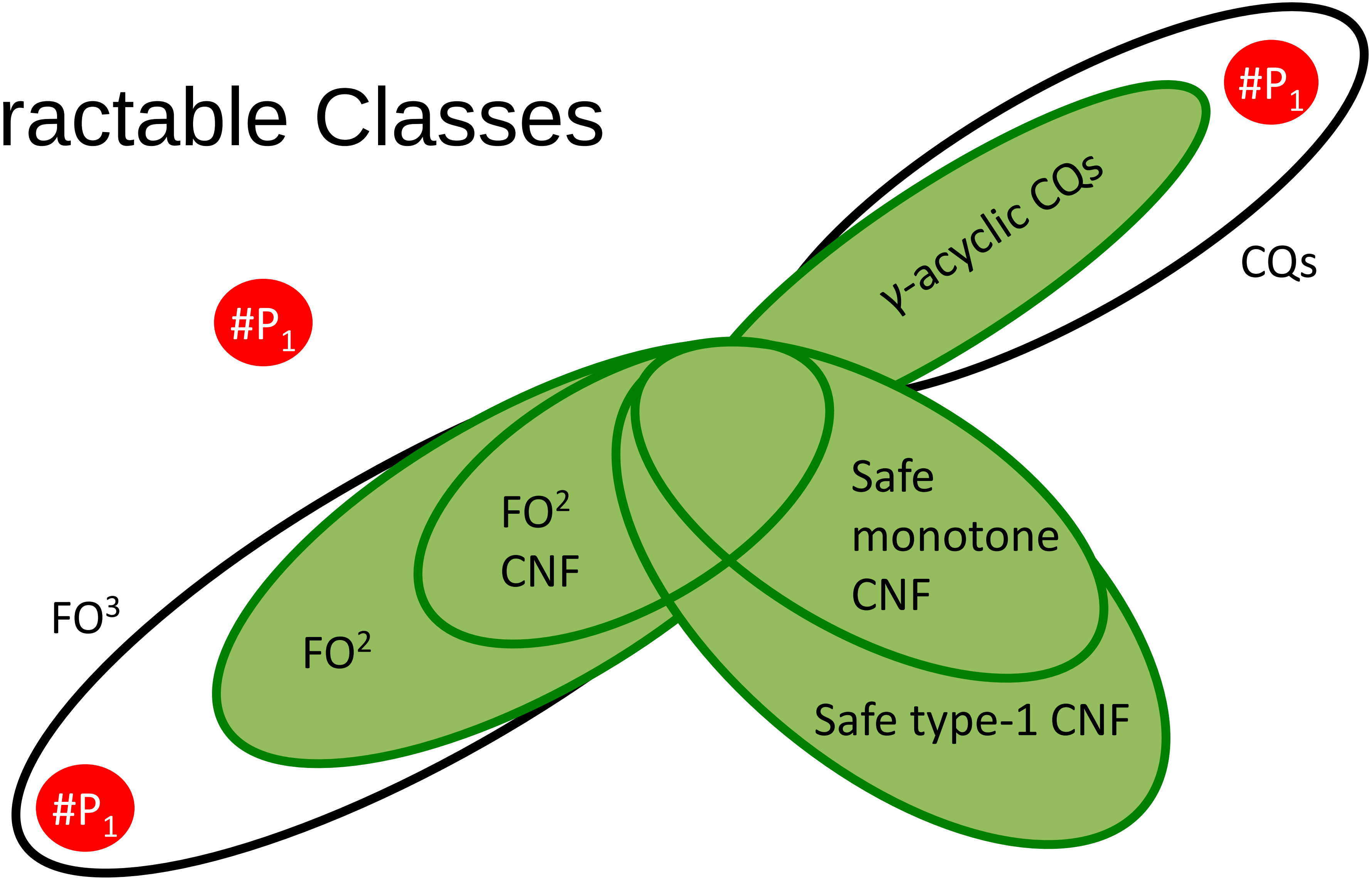


FO^2 is liftable!



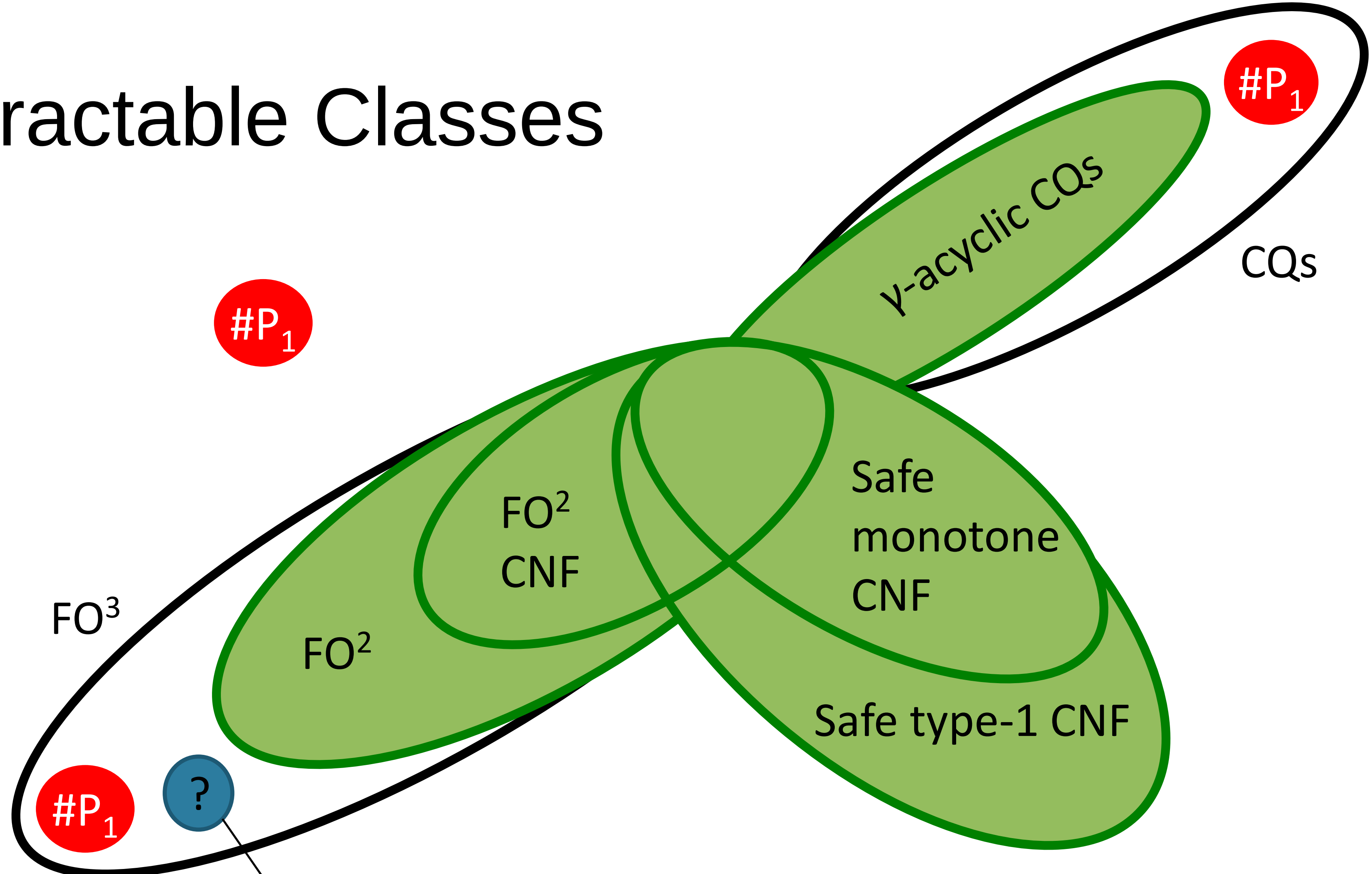
"Smokers are more likely to be friends with other smokers."
"Colleagues of the same age are more likely to be friends."
"People are either family or friends, but never both."
"If X is family of Y, then Y is also family of X."
"If X is a parent of Y, then Y cannot be a parent of X."

Tractable Classes



[VdB; NIPS'11], [VdB et al.; KR'14], [Gribkoff, VdB, Suciu; UAI'15], [Beame, VdB, Gribkoff, Suciu; PODS'15], etc.

Tractable Classes



Δ = ∀x,y,z, Friends(x,y) ∧ Friends(y,z) ⇒ Friends(x,z)

[VdB; NIPS'11], [VdB et al.; KR'14], [Gribkoff, VdB, Suciu; UAI'15], [Beame, VdB, Gribkoff, Suciu; PODS'15], etc.

Statistical Relational Learning

Markov Logic

3.14	Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

[Van den Broeck,PhD'13]

Statistical Relational Learning

Markov Logic

3.14 Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

Weight Function

w(Smokes)=1
w(¬Smokes)=1
w(Friends)=1
w(¬Friends)=1
w(F)=3.14
w(¬F)=1

FOL Sentence

∀x,y, F(x,y) ⇔ [Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)]

[Van den Broeck,PhD'13]

Statistical Relational Learning

Markov Logic

3.14 Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

Weight Function

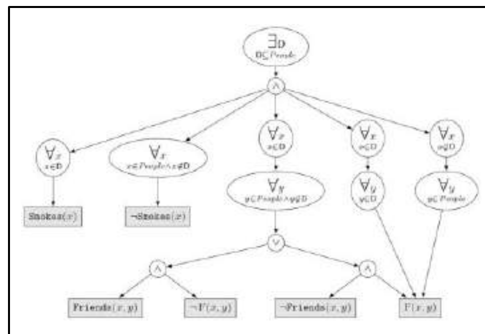
w(Smokes)=1
w(¬Smokes)=1
w(Friends)=1
w(¬Friends)=1
w(F)=3.14
w(¬F)=1

FOL Sentence

∀x,y, F(x,y) ⇔ [Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)]

Compile?

First-Order d-DNNF Circuit



[Van den Broeck,PhD'13]

Statistical Relational Learning

Markov Logic

3.14 Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

Weight Function

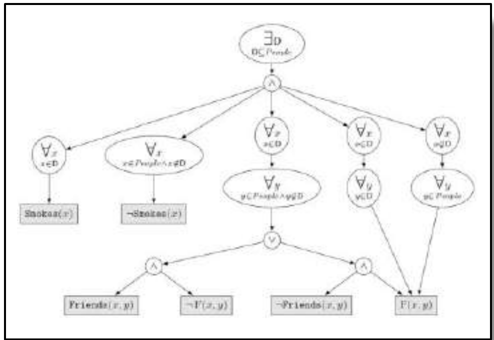
w(Smokes)=1
w(¬Smokes)=1
w(Friends)=1
w(¬Friends)=1
w(F)=3.14
w(¬F)=1

FOL Sentence

∀x,y, F(x,y) ⇔ [Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)]

Compile?

First-Order d-DNNF Circuit



Domain

Alice
Bob
Charlie

[Van den Broeck,PhD'13]

Statistical Relational Learning

Markov Logic

3.14 Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

Weight Function

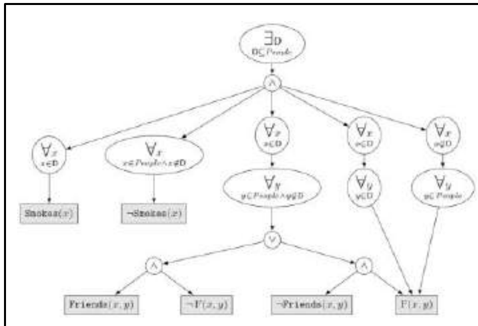
w(Smokes)=1
w(¬Smokes)=1
w(Friends)=1
w(¬Friends)=1
w(F)=3.14
w(¬F)=1

FOL Sentence

∀x,y, F(x,y) ⇔ [Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)]

Compile?

First-Order d-DNNF Circuit



Domain

Alice
Bob
Charlie

Z = WFOMC = 1479.85

[Van den Broeck,PhD'13]

Statistical Relational Learning

Markov Logic

3.14 Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

Weight Function

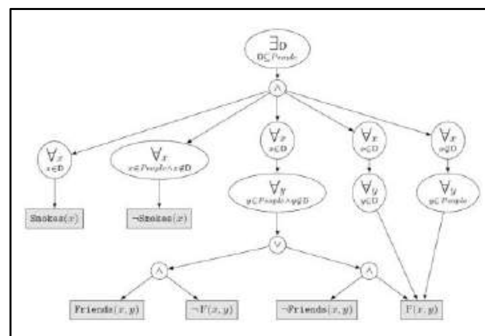
w(Smokes)=1
w(¬Smokes)=1
w(Friends)=1
w(¬Friends)=1
w(F)=3.14
w(¬F)=1

FOL Sentence

∀x,y, F(x,y) ⇔ [Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)]

Compile?

First-Order d-DNNF Circuit



Domain

Alice
Bob
Charlie

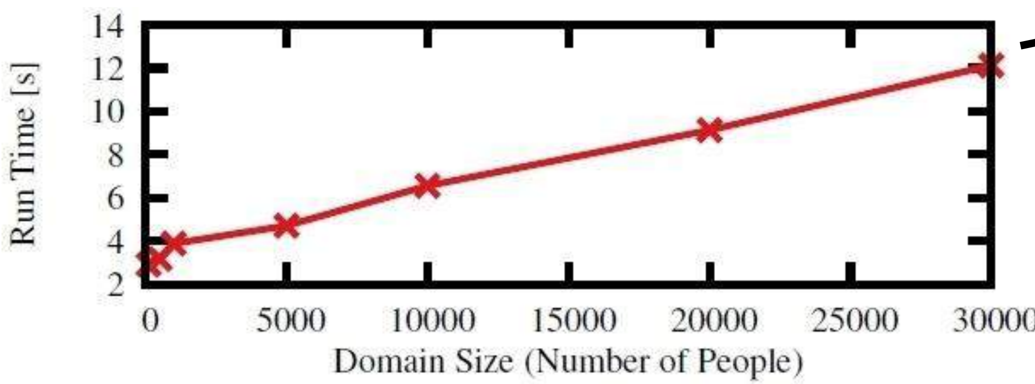
Z = WFOMC = 1479.85

Evaluation in time polynomial in domain size!

[Van den Broeck,PhD'13]

Lifted Machine Learning

- **Given:** A set of first-order logic **formulas**
  A set of training **databases**
- **Learn:** Maximum-likelihood **weights**



900,030,000 random variables

- Also structure learning!

	IMDb			UWCSE		
	Baseline	Lifted Weight Learning	Lifted Structure Learning	Baseline	Lifted Weight Learning	Lifted Structure Learning
Fold 1	-548	-378	**-306**	-1,860	-1,524	**-1,477**
Fold 2	-689	-390	**-309**	-594	-535	**-511**
Fold 3	-1,157	-851	**-733**	-1,462	-1,245	**-1,167**
Fold 4	-415	-285	**-224**	-2,820	-2,510	**-2,442**
Fold 5	-413	-267	**-216**	-2,763	-2,357	**-2,227**

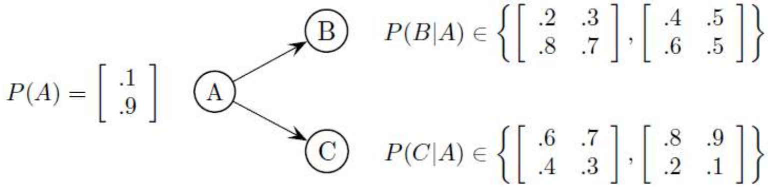
[Van Haaren et al.; MLJ'15]

The Even Broader Picture

- Statistical relational learning (e.g., Markov logic)

  Open-domain models (BLOG)
- Probabilistic description logics
- Certain query answers in databases
- Open information extraction
- Learning from positive-only examples
- Imprecise probabilities

  Credal sets, interval probability, qualitative uncertainty
- Credal Bayesian networks



Conclusions

- Relational probabilistic reasoning is **frontier** and **integration** of AI, KR, ML, DB, TH, etc.
- We need
  - relational models and logic
  - probabilistic models and statistical learning
  - algorithms that scale
- Open-world data model
  - semantics makes sense
  - FREE for UCQs, expensive otherwise
  - deep connection to model counting

QUESTIONS?

References

- Ceylan, Ismail Ilkan, Adnan Darwiche, and Guy Van den Broeck. "Open-world probabilistic databases." Proceedings of KR (2016).

- Suciu, Dan, Dan Olteanu, Christopher Ré, and Christoph Koch. "Probabilistic databases." Synthesis Lectures on Data Management 3, no. 2 (2011): 1-180.

- Dong, Xin, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. "Knowledge vault: A web-scale approach to probabilistic knowledge fusion." In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 601-610. ACM, 2014.

- Carlson, Andrew, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R. Hruschka Jr, and Tom M. Mitchell. "Toward an Architecture for Never-Ending Language Learning." In AAAI, vol. 5, p. 3. 2010.

- Niu, Feng, Ce Zhang, Christopher Ré, and Jude W. Shavlik. "DeepDive: Web-scale Knowledge-base Construction using Statistical Learning and Inference." VLDS 12 (2012): 25-28.

References

- Chen, Brian X. "Siri, Alexa and Other Virtual Assistants Put to the Test" The New York Times (2016).
- Dalvi, Nilesh, and Dan Suciu. "The dichotomy of probabilistic inference for unions of conjunctive queries." Journal of the ACM (JACM) 59, no. 6 (2012): 30.
- De Raedt, Luc, Anton Dries, Ingo Thon, Guy Van den Broeck, and Mathias Verbeke. "Inducing probabilistic relational rules from probabilistic examples." In Proceedings of the 24th International Conference on Artificial Intelligence, pp. 1835-1843. AAAI Press, 2015.
- Van den Broeck, Guy. "Towards high-level probabilistic reasoning with lifted inference." AAAI Spring Symposium on KRR (2015).
- Niepert, Mathias, and Guy Van den Broeck. "Tractability through exchangeability: A new perspective on efficient probabilistic inference." AAAI (2014).
- Van den Broeck, Guy. "On the completeness of first-order knowledge compilation for lifted probabilistic inference." In Advances in Neural Information Processing Systems, pp. 1386-1394. 2011.

References

- Van den Broeck, Guy, Wannes Meert, and Adnan Darwiche. "Skolemization for weighted first-order model counting." In Proceedings of the 14th International Conference on Principles of Knowledge Representation and Reasoning (KR). 2014.
- Gribkoff, Eric, Guy Van den Broeck, and Dan Suciu. "Understanding the complexity of lifted inference and asymmetric weighted model counting." UAI, 2014.
- Beame, Paul, Guy Van den Broeck, Eric Gribkoff, and Dan Suciu. "Symmetric weighted first-order model counting." In Proceedings of the 34th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, pp. 313-328. ACM, 2015.
- Chavira, Mark, and Adnan Darwiche. "On probabilistic inference by weighted model counting." Artificial Intelligence 172.6 (2008): 772-799.
- Sang, Tian, Paul Beame, and Henry A. Kautz. "Performing Bayesian inference by weighted model counting." AAAI. Vol. 5. 2005.

References

- Van den Broeck, Guy, Nima Taghipour, Wannes Meert, Jesse Davis, and Luc De Raedt. "Lifted probabilistic inference by first-order knowledge compilation." In Proceedings of the Twenty-Second international joint conference on Artificial Intelligence, pp. 2178-2185. AAAI Press/International Joint Conferences on Artificial Intelligence, 2011.

- Van den Broeck, Guy. Lifted inference and learning in statistical relational models. Diss. Ph. D. Dissertation, KU Leuven, 2013.

- Gogate, Vibhav, and Pedro Domingos. "Probabilistic theorem proving." UAI (2011).

- Guy Van den Broeck and Dan Suciu. Query Processing on Probabilistic Data: A Survey, Foundations and Trends in Databases, Now Publishers, 2017

References

- Belle, Vaishak, Andrea Passerini, and Guy Van den Broeck. "Probabilistic inference in hybrid domains by weighted model integration." Proceedings of 24th International Joint Conference on Artificial Intelligence (IJCAI). 2015.

- Belle, Vaishak, Guy Van den Broeck, and Andrea Passerini. "Hashing-based approximate probabilistic inference in hybrid domains." In Proceedings of the 31st Conference on Uncertainty in Artificial Intelligence (UAI). 2015.

- Fierens, Daan, Guy Van den Broeck, Joris Renkens, Dimitar Shterionov, Bernd Gutmann, Ingo Thon, Gerda Janssens, and Luc De Raedt. "Inference and learning in probabilistic logic programs using weighted boolean formulas." Theory and Practice of Logic Programming 15, no. 03 (2015): 358-401.