

Rule Induction and Reasoning in Knowledge Graphs

Daria Stepanova

Bosch Center for Artificial Intelligence, Renningen, Germany

21.11.2019



Motivation

Preliminaries

Rule Learning

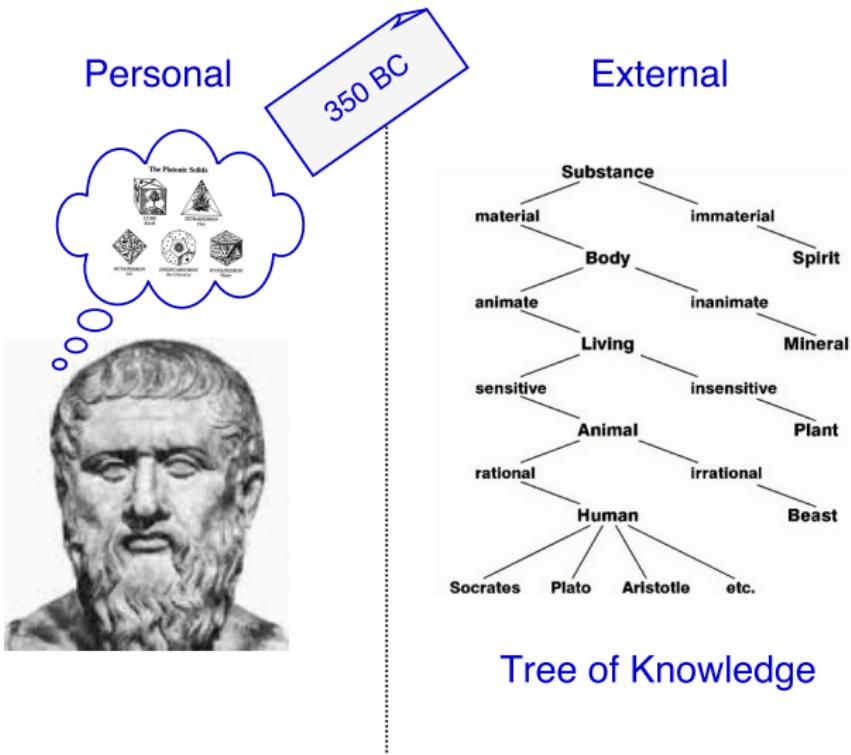
Exception-awareness

Incompleteness

Rules from Hybrid Sources

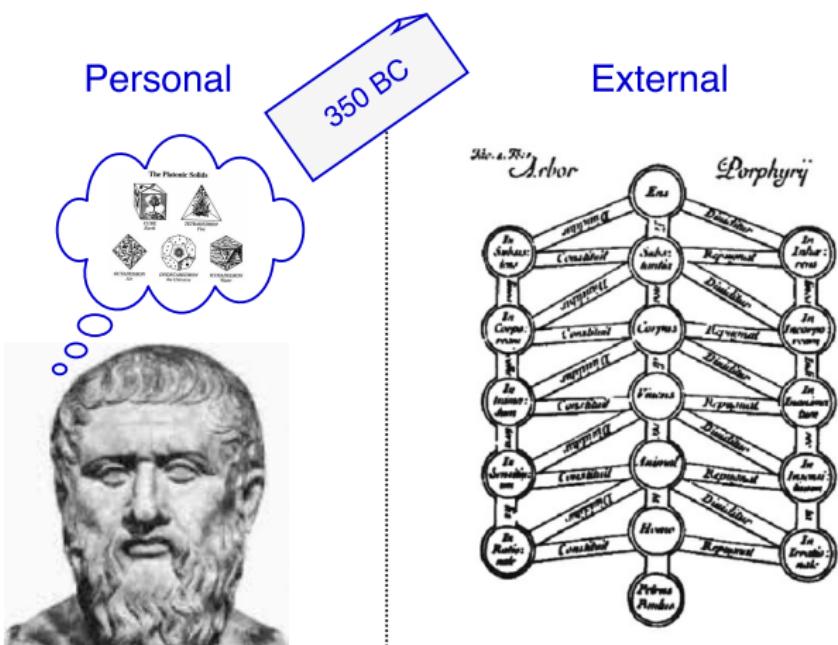
What is Knowledge?

Plato: “*Knowledge is justified true belief*”



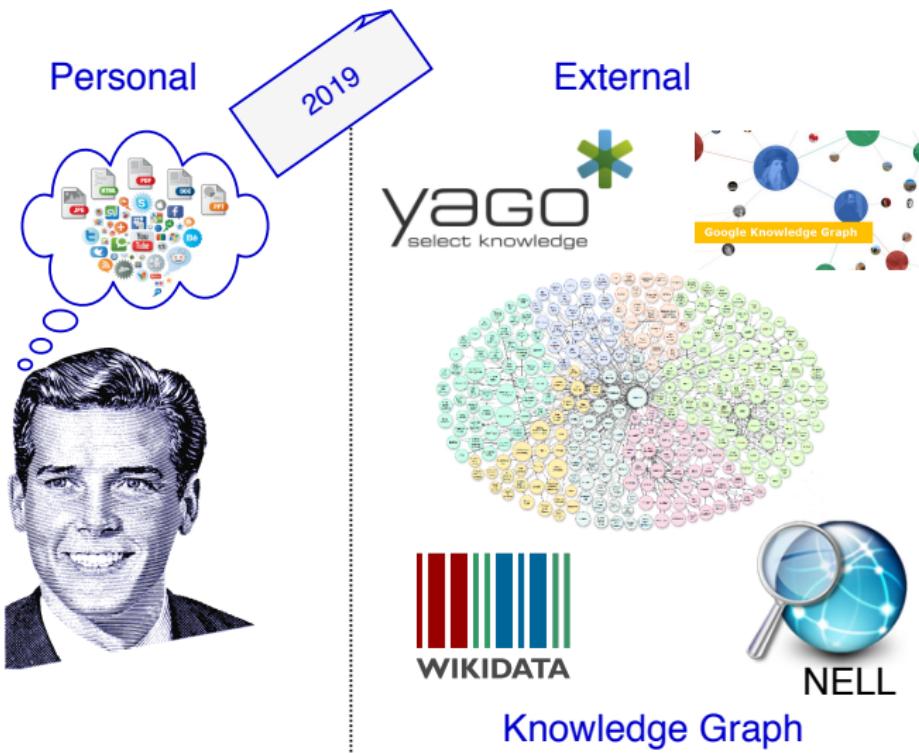
What is Knowledge?

Plato: “*Knowledge is justified true belief*”



Knowledge Graphs as Digital Knowledge

“Digital knowledge is semantically enriched machine processable data”



Semantic Web Search



winner of Australian Open 2018



Roger Federer

Tennis player



[rogerfederer.com](#)

Roger Federer is a Swiss professional tennis player who is currently ranked world No. 10 by the Association of Tennis Professionals. Many players and analysts have called him the greatest tennis player of all time. [Wikipedia](#)

Born: August 8, 1981 (age 35 years), Basel, Switzerland

Height: 1.85 m

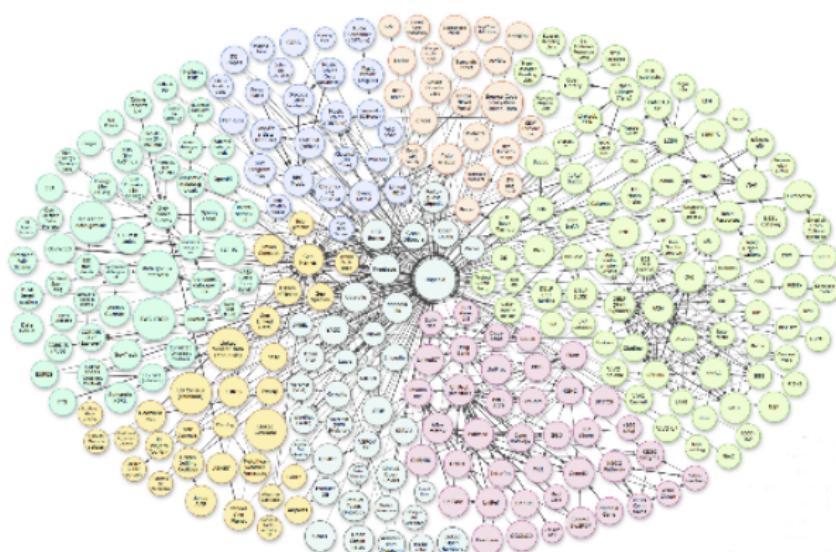
Weight: 85 kg

Spouse: Mirka Federer (m. 2009)

Children: Lenny Federer, Myla Rose Federer, Charlene Riva Federer, Leo Federer



Semantic Web Search

 $\exists X \text{ winnerOf}(X, \text{AustralianOpen2018})$ 

Roger Federer

Tennis player

rogerfederer.com

Roger Federer is a Swiss professional tennis player who is currently ranked world No. 10 by the Association of Tennis Professionals. Many players and analysts have called him the greatest tennis player of all time. [Wikipedia](#)

Born: August 8, 1981 (age 35 years), Basel, Switzerland

Height: 1.85 m

Weight: 85 kg

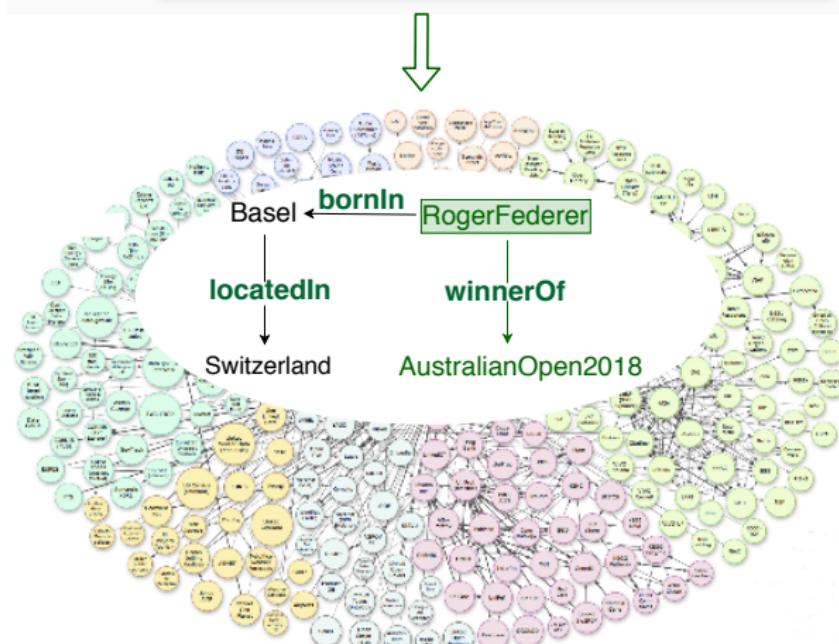
Spouse: Mirka Federer (m. 2009)

Children: Lenny Federer, Myla Rose Federer, Charlene Riva Federer, Leo Federer

Semantic Web Search



winner of Australian Open 2018



Roger Federer

Tennis player

rogerfederer.com

Roger Federer is a Swiss professional tennis player who is currently ranked world No. 10 by the Association of Tennis Professionals. Many players and analysts have called him the greatest tennis player of all time. [Wikipedia](#)

Born: August 8, 1981 (age 35 years), Basel, Switzerland

Height: 1.85 m

Weight: 85 kg

Spouse: Mirka Federer (m. 2009)

Children: Lenny Federer, Myla Rose Federer, Charlene Riva Federer, Leo Federer

Semantic Web Search

living place of the winner of australian open 2018



All News Images Videos Maps More Settings Tools

About 1,220,000,000 results (1.10 seconds)

2018 Australian Open - Wikipedia

https://en.wikipedia.org/wiki/2018_Australian_Open ▾

Roger Federer was the defending **champion** in the men's singles event and successfully retained his title (his sixth), defeating Marin Čilić in the final, while Caroline Wozniacki **won** the women's title, defeating Simona Halep in the final.

Venue: [Melbourne Park](#)

Prize money: A\$55,000,000

Location: [Melbourne, Victoria, Australia](#)

Draw: 128S / 64D /

Missing: [living](#) | Must include: [living](#)

Semantic Web Search

wife of Roger Federer



All

Images

News

Videos

Maps

More

Settings

Tools

About 42,200,000 results (0.50 seconds)

Roger Federer / Wife

Mirka Federer

m. 2009



Miroslava "Mirka" Federer is a Slovak-born Swiss former professional tennis player. She reached her career-high WTA singles ranking of world No. 76 on 10 September 2001 and a doubles ranking of No. 215 on 24 August 1998. She is the wife of tennis player Roger Federer, having first met him at the 2000 Summer Olympics. [Wikipedia](#)

Semantic Web Search

living place of Mirka Federer



All

Images

News

Shopping

Videos

More

Settings

Tools

About 1.910.000 results (0,92 seconds)

Mirka Federer / Residence



Map data ©2017 GeoBasis-DE/BKG (©2009), Google

Bottmingen, Switzerland

Human Reasoning

*livesIn(Y, Z) ← marriedTo(X, Y),
livesIn(X, Z)* *Married people live together*

marriedTo(mirka, roger) *Mirka is married to Roger*

livesIn(mirka, bottmingen) *Mirka lives in Bottmingen*

Human Reasoning

*livesIn(Y, Z) ← marriedTo(X, Y),
livesIn(X, Z)*

Married people live together

marriedTo(mirka, roger)

Mirka is married to Roger

livesIn(mirka, bottmingen)

Mirka lives in Bottmingen

livesIn(roger, bottmingen)

Roger lives in Bottmingen



livesIn →



Human Reasoning

*livesIn(Y, Z) ← marriedTo(X, Y),
livesIn(X, Z)*

Married people live together

marriedTo(mirka, roger)

Mirka is married to Roger

livesIn(mirka, bottmingen)

Mirka lives in Bottmingen

livesIn(roger, bottmingen)

Roger lives in Bottmingen



livesIn →



But where can a machine get such rules from?

Applications of Rule Learning

- Fact prediction
- Fact checking
- Data cleaning
- Domain description
- Finding trends in KGs ...

Motivation

Preliminaries

Rule Learning

Exception-awareness

Incompleteness

Rules from Hybrid Sources

Horn Rules

Rule: $\underbrace{a}_{\text{head}} \leftarrow \underbrace{b_1, \dots, b_m}_{\text{body}}$.

Informal semantics: If b_1, \dots, b_m are true, then a must be true.

Logic program: Set of rules

Example: ground rule

```
% If Mirka is married to Roger and lives in B., then Roger lives there too  
livesIn(roger, bottmingen) ← isMarried(mirka, roger), livesIn(mirka, bottmingen)
```

Horn Rules

Rule: $\underbrace{a}_{\text{head}} \leftarrow \underbrace{b_1, \dots, b_m}_{\text{body}}$.

Informal semantics: If b_1, \dots, b_m are true, then a must be true.

Logic program: Set of rules

Example: non-ground rule

```
% Married people live together  
livesIn(Y, Z) ← isMarried(X, Y), livesIn(X, Z)
```

Nonmonotonic Rules

Rule: $\underbrace{a}_{\text{head}} \leftarrow \underbrace{b_1, \dots, b_m, \text{not } b_{m+1}, \dots, \text{not } b_n}_{\text{body}}$

Informal semantics: If b_1, \dots, b_m are true and none of b_{m+1}, \dots, b_n is known, then a must be true.

Closed World Assumption (CWA): facts not known to be true are false

Example: nonmonotonic rule

% Two married live together unless one is a researcher

$\text{livesIn}(Y, Z) \leftarrow \text{isMarried}(X, Y), \text{livesIn}(X, Z), \text{not researcher}(Y)$

Nonmonotonic Rules

Rule: $\underbrace{a}_{\text{head}} \leftarrow \underbrace{b_1, \dots, b_m, \text{not } b_{m+1}, \dots, \text{not } b_n}_{\text{body}}$

Informal semantics: If b_1, \dots, b_m are true and none of b_{m+1}, \dots, b_n is known, then a must be true.

Closed World Assumption (CWA): facts not known to be true are false

not is different from \neg !

% At a rail road crossing cross the road if no train is known to approach"
 $\text{walk} \leftarrow \text{at}(L), \text{crossing}(L), \text{not train_approaches}(L)$

% At a rail road crossing cross the road if no train approaches
 $\text{walk} \leftarrow \text{at}(L), \text{crossing}(L), \neg \text{train_approaches}(L)$

Answer Set Programs

Evaluation of ASP programs is model-based

Answer set program (ASP) is a set of nonmonotonic rules

- (1) $isMarriedTo(mary, john)$
- (2) $livesIn(mary, ulm)$
- (3) $livesIn(Y, Z) \leftarrow isMarriedTo(X, Y), livesIn(X, Z),$
not researcher(Y)

Answer Set Programs

Evaluation of ASP programs is model-based

1. Grounding: substitute all variables with constants in all possible ways

Answer set program (ASP) is a set of nonmonotonic rules

- (1) $isMarriedTo(mary, john)$
- (2) $livesIn(mary, ulm)$
- (3) $livesIn(Y, Z) \leftarrow isMarriedTo(X, Y), livesIn(X, Z),$
not researcher(Y)

Answer Set Programs

Evaluation of ASP programs is model-based

1. Grounding: substitute all variables with constants in all possible ways

Answer set program (ASP) is a set of nonmonotonic rules

- (1) *isMarriedTo(mary, john)*
- (2) *livesIn(mary, ulm)*
- (3) *livesIn(john, ulm) ← isMarriedTo(mary, john), livesIn(mary, ulm), not researcher(john)*

Answer Set Programs

Evaluation of ASP programs is model-based

1. Grounding: substitute all variables with constants in all possible ways
2. Solving: compute a minimal model (answer set) / satisfying all rules

Answer set program (ASP) is a set of nonmonotonic rules

- (1) *isMarriedTo(mary, john)* (2) *livesIn(mary, ulm)*
(3) *livesIn(john, ulm) ← isMarriedTo(mary, john), livesIn(mary, ulm),
not researcher(john)*

$I = \{isMarriedTo(mary, john), livesIn(mary, ulm), livesIn(john, ulm)\}$

CWA: *researcher(john)* can not be derived, thus it is false

Answer Set Programs

Evaluation of ASP programs is model-based

1. **Grounding:** substitute all variables with constants in all possible ways
2. **Solving:** compute a minimal model (answer set) I satisfying all rules

Answer set program (ASP) is a set of nonmonotonic rules

(1) $\text{isMarriedTo}(\text{mary}, \text{john})$ (2) $\text{livesIn}(\text{mary}, \text{ulm})$

(3) $\text{livesIn}(\text{john}, \text{ulm}) \leftarrow \text{isMarriedTo}(\text{mary}, \text{john}), \text{livesIn}(\text{mary}, \text{ulm}),$
not researcher(john)

(4) $\text{researcher}(\text{john})$

researcher(john)

$I = \{\text{isMarriedTo}(\text{mary}, \text{john}), \text{livesIn}(\text{mary}, \text{ulm}), \underline{\text{livesIn}(\text{john}, \text{ulm})}\}$

Answer Set Programs

Evaluation of ASP programs is model-based

1. **Grounding:** substitute all variables with constants in all possible ways
2. **Solving:** compute a minimal model (answer set) I satisfying all rules

Answer set program (ASP) is a set of nonmonotonic rules

- (1) *isMarriedTo(mary, john)* (2) *livesIn(mary, ulm)*
(3) *livesIn(john, ulm) ← isMarriedTo(mary, john), livesIn(mary, ulm),
not researcher(john)*
(4) *researcher(john)*

researcher(john)

$$I = \{ \textit{isMarriedTo(mary, john)}, \textit{livesIn(mary, ulm)}, \underline{\textit{livesIn(john, ulm)}} \}$$

Particularly suited for reasoning under incompleteness!

Motivation

Preliminaries

Rule Learning

Exception-awareness

Incompleteness

Rules from Hybrid Sources

Reasoning with Incomplete Information

Default Reasoning

Assume normal state of affairs, unless there is evidence to the contrary

By default married people live together.

Reasoning with Incomplete Information

Default Reasoning

Assume normal state of affairs, unless there is evidence to the contrary

By default married people live together.

Abduction

Choose between several explanations that explain an observation

John and Mary live together. They must be married.

Reasoning with Incomplete Information

Default Reasoning

Assume normal state of affairs, unless there is evidence to the contrary

By default married people live together.

Abduction

Choose between several explanations that explain an observation

John and Mary live together. They must be married.

Induction

Generalize a number of similar observations into a hypothesis

Given many examples of spouses living together generalize this knowledge.

Reasoning with Incomplete Information

Default Reasoning

Assume normal state of affairs, unless there is evidence to the contrary

By default married people live together.

Abduction

Choose between several explanations that explain an observation

John and Mary live together. They must be married.

Induction

Generalize a number of similar observations into a hypothesis

Given many examples of spouses living together generalize this knowledge.

History of Inductive Learning

- AI & Machine Learning 1960s-70s:
Banerji, Plotkin, Vere, Michalski, ...
- AI & Machine Learning 1980s:
Shapiro, Sammut, Muggleton, ...
- Inductive Logic Programming (ILP) 1990s:
Muggleton, Quinlan, De Raedt, ...
- Statistical Relational Learning 2000s:
Getoor, Koller, Domingos, Sato, ...

Learning from Examples

Inductive Learning from Examples [?]

Given:

- $E^+ = \{fatherOf(john, mary), fatherOf(david, steve)\}$
- $E^- = \{fatherOf(kathy, ellen), fatherOf(john, steve)\}$
- $T = \{parentOf(john, mary), male(john),
parentOf(david, steve), male(david),
parentOf(kathy, ellen), female(kathy)\}$
- Language bias: Horn rules with 2 body atoms

Learning from Examples

Inductive Learning from Examples [?]

Given:

- $E^+ = \{fatherOf(john, mary), fatherOf(david, steve)\}$
- $E^- = \{fatherOf(kathy, ellen), fatherOf(john, steve)\}$
- $T = \{parentOf(john, mary), male(john),
parentOf(david, steve), male(david),
parentOf(kathy, ellen), female(kathy)\}$
- Language bias: Horn rules with 2 body atoms

Possible hypothesis:

- $Hyp : fatherOf(X, Y) \leftarrow parentOf(X, Y), male(X)$

Learning from Interpretations

Inductive Learning from Interpretations [?]

Given:

- $I = \{isMarriedTo(mirka, roger), livesIn(mirka, b), livesIn(roger, b), bornIn(mirka, b)\}$
- $T = \{isMarriedTo(mirka, roger); bornIn(mirka, b); livesIn(X, Y) \leftarrow bornIn(X, Y)\}$
- Language bias: Horn rules with 2 body atoms

Learning from Interpretations

Inductive Learning from Interpretations [?]

Given:

- $I = \{isMarriedTo(mirka, roger), livesIn(mirka, b), livesIn(roger, b), bornIn(mirka, b)\}$
- $T = \{isMarriedTo(mirka, roger); bornIn(mirka, b); livesIn(X, Y) \leftarrow bornIn(X, Y)\}$
- Language bias: Horn rules with 2 body atoms

Possible Hypothesis:

- Hyp : $livesIn(Y, Z) \leftarrow isMarriedTo(X, Y), bornIn(X, Z)$

Common Techniques in ILP

- **Generality (\succeq)**: essential component of symbolic learning systems
- Generalization as θ -subsumption
 - Atoms: $a \succeq b$ iff a substitution θ exists such that $a\theta = b$

Common Techniques in ILP

- **Generality (\succeq)**: essential component of symbolic learning systems
- Generalization as θ -subsumption
 - Atoms: $a \succeq b$ iff a substitution θ exists such that $a\theta = b$
 $person(X) \succeq person(roger)$, $\theta = \{X/roger\}$

Common Techniques in ILP

- Generality (\succeq): essential component of symbolic learning systems
- Generalization as θ -subsumption
 - Atoms: $a \succeq b$ iff a substitution θ exists such that $a\theta = b$
 $\text{person}(X) \succeq \text{person(roger)}$, $\theta = \{X/\text{roger}\}$
 - Clause: $C \succeq D$ iff θ exists, s.t. $C\theta \subseteq D$

Common Techniques in ILP

- **Generality (\succeq)**: essential component of symbolic learning systems
- Generalization as θ -subsumption
 - Atoms: $a \succeq b$ iff a substitution θ exists such that $a\theta = b$
 $\text{person}(X) \succeq \text{person(roger)}, \theta = \{X/\text{roger}\}$
 - Clause: $C \succeq D$ iff θ exists, s.t. $C\theta \subseteq D$
 $\{\text{worksAt}(X, Y)\} \succeq \{\text{worksAt}(Z, \text{bosch}), \text{researcher}(Z)\},$

Common Techniques in ILP

- **Generality (\succeq)**: essential component of symbolic learning systems
- Generalization as θ -subsumption
 - Atoms: $a \succeq b$ iff a substitution θ exists such that $a\theta = b$
 $\text{person}(X) \succeq \text{person(roger)}$, $\theta = \{X/\text{roger}\}$
 - Clause: $C \succeq D$ iff θ exists, s.t. $C\theta \subseteq D$
 $\{\text{worksAt}(X, Y)\} \succeq \{\text{worksAt}(Z, \text{bosch}), \text{researcher}(Z)\}$,
 $\theta = \{X/Z, Y/\text{bosch}\}$

Common Techniques in ILP

- Generality (\succeq): essential component of symbolic learning systems
- Generalization as θ -subsumption
 - Atoms: $a \succeq b$ iff a substitution θ exists such that $a\theta = b$
 $\text{person}(X) \succeq \text{person(roger)}$, $\theta = \{X/\text{roger}\}$
 - Clause: $C \succeq D$ iff θ exists, s.t. $C\theta \subseteq D$
 $\{\text{worksAt}(X, Y)\} \succeq \{\text{worksAt}(Z, \text{bosch}), \text{researcher}(Z)\}$,
 $\theta = \{X/Z, Y/\text{bosch}\}$
- Generalization as entailment
 - Logic program: $Hyp1 \succeq Hyp2$ iff $Hyp1 \models Hyp2$

Common Techniques in ILP

- Generality (\succeq): essential component of symbolic learning systems
- Generalization as θ -subsumption
 - Atoms: $a \succeq b$ iff a substitution θ exists such that $a\theta = b$
 $\text{person}(X) \succeq \text{person(roger)}$, $\theta = \{X/\text{roger}\}$
 - Clause: $C \succeq D$ iff θ exists, s.t. $C\theta \subseteq D$
 $\{\text{worksAt}(X, Y)\} \succeq \{\text{worksAt}(Z, \text{bosch}), \text{researcher}(Z)\}$,
 $\theta = \{X/Z, Y/\text{bosch}\}$
- Generalization as entailment
 - Logic program: $Hyp1 \succeq Hyp2$ iff $Hyp1 \models Hyp2$
 $\underbrace{\text{person}(X) \leftarrow \text{researcher}(X)}_{Hyp1} \quad \underbrace{\text{person}(\text{mat}) \leftarrow \text{researcher}(\text{mat})}_{Hyp2}$

Common Techniques in ILP

- Generality (\succeq): essential component of symbolic learning systems
- Generalization as θ -subsumption
 - Atoms: $a \succeq b$ iff a substitution θ exists such that $a\theta = b$
 $\text{person}(X) \succeq \text{person(roger)}$, $\theta = \{X/\text{roger}\}$
 - Clause: $C \succeq D$ iff θ exists, s.t. $C\theta \subseteq D$
 $\{\text{worksAt}(X, Y)\} \succeq \{\text{worksAt}(Z, \text{bosch}), \text{researcher}(Z)\}$,
 $\theta = \{X/Z, Y/\text{bosch}\}$
- Generalization as entailment
 - Logic program: $Hyp1 \succeq Hyp2$ iff $Hyp1 \models Hyp2$
 $\underbrace{\text{person}(X) \leftarrow \text{researcher}(X)}_{Hyp1} \quad \underbrace{\text{person}(mat) \leftarrow \text{researcher}(mat)}_{Hyp2}$
 $Hyp1 \succeq Hyp2$

Common Techniques in ILP

- Generality (\succeq): essential component of symbolic learning systems
- Generalization as θ -subsumption
 - Atoms: $a \succeq b$ iff a substitution θ exists such that $a\theta = b$
 $\text{person}(X) \succeq \text{person(roger)}, \theta = \{X/\text{roger}\}$
 - Clause: $C \succeq D$ iff θ exists, s.t. $C\theta \subseteq D$
 $\{\text{worksAt}(X, Y)\} \succeq \{\text{worksAt}(Z, \text{bosch}), \text{researcher}(Z)\},$
 $\theta = \{X/Z, Y/\text{bosch}\}$
- Generalization as entailment

- Logic program: $Hyp1 \succeq Hyp2$ iff $Hyp1 \models Hyp2$
 $\underbrace{\text{person}(X) \leftarrow \text{researcher}(X)}_{Hyp1} \quad \underbrace{\text{person}(X) \leftarrow \text{researcher}(X), \text{alive}(X)}_{Hyp2}$

Common Techniques in ILP

- Generality (\succeq): essential component of symbolic learning systems
- Generalization as θ -subsumption
 - Atoms: $a \succeq b$ iff a substitution θ exists such that $a\theta = b$
 $\text{person}(X) \succeq \text{person(roger)}$, $\theta = \{X/\text{roger}\}$
 - Clause: $C \succeq D$ iff θ exists, s.t. $C\theta \subseteq D$
 $\{\text{worksAt}(X, Y)\} \succeq \{\text{worksAt}(Z, \text{bosch}), \text{researcher}(Z)\}$,
 $\theta = \{X/Z, Y/\text{bosch}\}$
- Generalization as entailment
 - Logic program: $Hyp1 \succeq Hyp2$ iff $Hyp1 \models Hyp2$
 $\underbrace{\text{person}(X) \leftarrow \text{researcher}(X)}_{Hyp1} \quad \underbrace{\text{person}(X) \leftarrow \text{researcher}(X), \text{alive}(X)}_{Hyp2}$
 $Hyp1 \succeq Hyp2$

Common Techniques in ILP

- Generality (\succeq): essential component of symbolic learning systems
- Generalization as θ -subsumption
 - Atoms: $a \succeq b$ iff a substitution θ exists such that $a\theta = b$
 $\text{person}(X) \succeq \text{person(roger)}$, $\theta = \{X/\text{roger}\}$
 - Clause: $C \succeq D$ iff θ exists, s.t. $C\theta \subseteq D$
 $\{\text{worksAt}(X, Y)\} \succeq \{\text{worksAt}(Z, \text{bosch}), \text{researcher}(Z)\}$,
 $\theta = \{X/Z, Y/\text{bosch}\}$
- Generalization as entailment
 - Logic program: $Hyp1 \succeq Hyp2$ iff $Hyp1 \models Hyp2$
 $\underbrace{\text{person}(X) \leftarrow \text{researcher}(X)}_{Hyp1} \quad \underbrace{\text{person}(X) \leftarrow \text{researcher}(X), \text{alive}(X)}_{Hyp2}$
 $Hyp1 \succeq Hyp2$
 - Relative entailment: $Hyp1 \succeq Hyp2$ wrt T iff $Hyp1 \cup T \models Hyp2$

Common Techniques in ILP

- Generality (\succeq): essential component of symbolic learning systems
- Generalization as θ -subsumption
 - Atoms: $a \succeq b$ iff a substitution θ exists such that $a\theta = b$
 $\text{person}(X) \succeq \text{person(roger)}, \theta = \{X/\text{roger}\}$
 - Clause: $C \succeq D$ iff θ exists, s.t. $C\theta \subseteq D$
 $\{\text{worksAt}(X, Y)\} \succeq \{\text{worksAt}(Z, \text{bosch}), \text{researcher}(Z)\},$
 $\theta = \{X/Z, Y/\text{bosch}\}$
- Generalization as entailment
 - Logic program: $Hyp1 \succeq Hyp2$ iff $Hyp1 \models Hyp2$
 $\underbrace{\text{person}(X) \leftarrow \text{researcher}(X)}_{Hyp1} \quad \underbrace{\text{person}(X) \leftarrow \text{researcher}(X), \text{alive}(X)}_{Hyp2}$
 $Hyp1 \succeq Hyp2$
 - Relative entailment: $Hyp1 \succeq Hyp2$ wrt T iff $Hyp1 \cup T \models Hyp2$
 $\text{livesIn(roger, bottmingen)} ? \text{livesIn(roger, switzerland)}$

Common Techniques in ILP

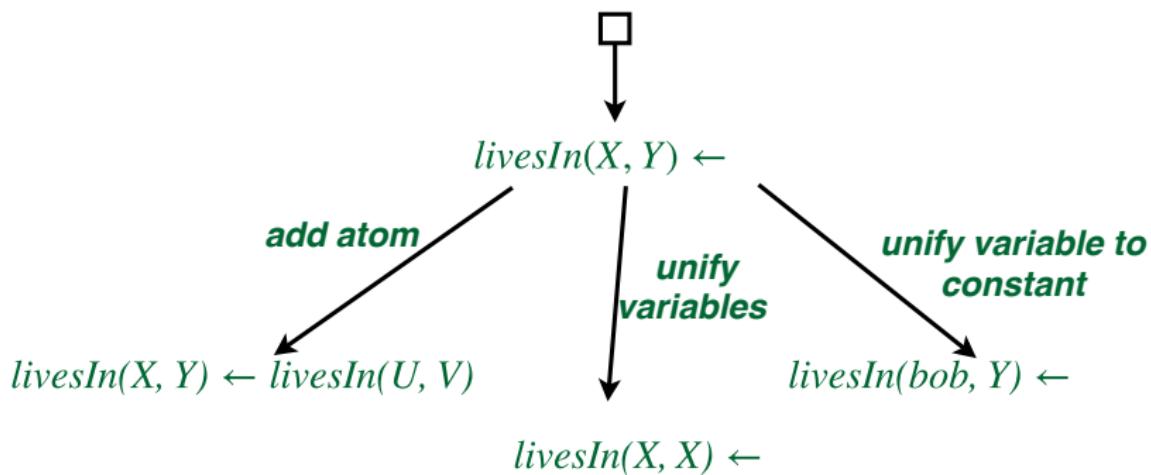
- Generality (\succeq): essential component of symbolic learning systems
- Generalization as θ -subsumption
 - Atoms: $a \succeq b$ iff a substitution θ exists such that $a\theta = b$
 $\text{person}(X) \succeq \text{person(roger)}, \theta = \{X/\text{roger}\}$
 - Clause: $C \succeq D$ iff θ exists, s.t. $C\theta \subseteq D$
 $\{\text{worksAt}(X, Y)\} \succeq \{\text{worksAt}(Z, \text{bosch}), \text{researcher}(Z)\},$
 $\theta = \{X/Z, Y/\text{bosch}\}$
- Generalization as entailment
 - Logic program: $Hyp1 \succeq Hyp2$ iff $Hyp1 \models Hyp2$
 $\underbrace{\text{person}(X) \leftarrow \text{researcher}(X)}_{Hyp1} \quad \underbrace{\text{person}(X) \leftarrow \text{researcher}(X), \text{alive}(X)}_{Hyp2}$
 $Hyp1 \succeq Hyp2$
 - Relative entailment: $Hyp1 \succeq Hyp2$ wrt T iff $Hyp1 \cup T \models Hyp2$
 $\text{livesIn(roger, bottmingen)} ? \text{livesIn(roger, switzerland)}$
 $T : \text{livesIn}(X, \text{switzerland}) \leftarrow \text{livesIn}(X, \text{bottmingen})$

Common Techniques in ILP

- Generality (\succeq): essential component of symbolic learning systems
- Generalization as θ -subsumption
 - Atoms: $a \succeq b$ iff a substitution θ exists such that $a\theta = b$
 $\text{person}(X) \succeq \text{person(roger)}$, $\theta = \{X/\text{roger}\}$
 - Clause: $C \succeq D$ iff θ exists, s.t. $C\theta \subseteq D$
 $\{\text{worksAt}(X, Y)\} \succeq \{\text{worksAt}(Z, \text{bosch}), \text{researcher}(Z)\}$,
 $\theta = \{X/Z, Y/\text{bosch}\}$
- Generalization as entailment
 - Logic program: $Hyp1 \succeq Hyp2$ iff $Hyp1 \models Hyp2$
 $\underbrace{\text{person}(X) \leftarrow \text{researcher}(X)}_{Hyp1} \quad \underbrace{\text{person}(X) \leftarrow \text{researcher}(X), \text{alive}(X)}_{Hyp2}$
 $Hyp1 \succeq Hyp2$
 - Relative entailment: $Hyp1 \succeq Hyp2$ wrt T iff $Hyp1 \cup T \models Hyp2$
 $\text{livesIn(roger, bottmingen)} \succeq \text{livesIn(roger, switzerland)}$
 $T : \text{livesIn}(X, \text{switzerland}) \leftarrow \text{livesIn}(X, \text{bottmingen})$

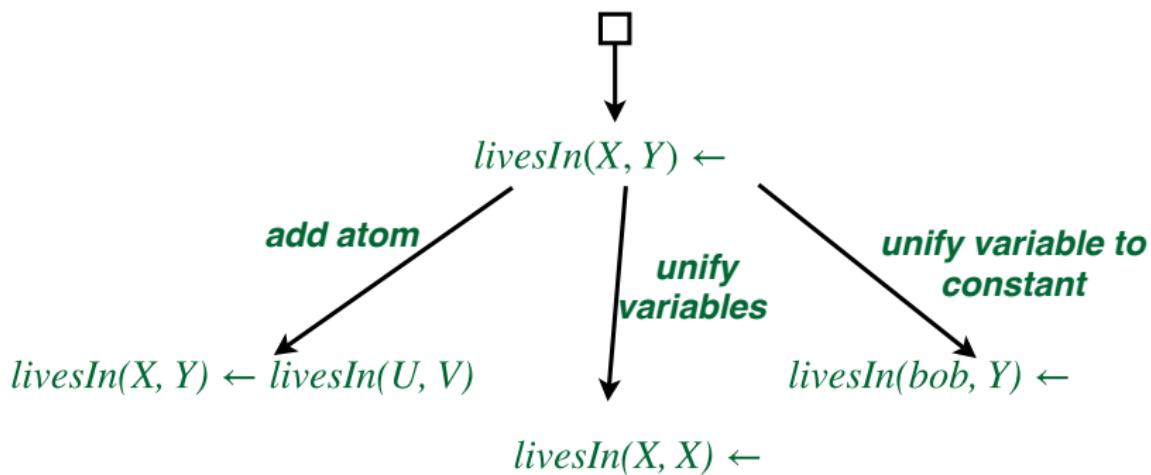
Common Techniques in ILP

- **Clause refinement [?]:** e.g., MIS, FOIL, etc.
 - Explore clause search space from general to specific or vice versa to find a hypothesis that covers all examples.



Common Techniques in ILP

- **Clause refinement** [?]: e.g., MIS, FOIL, etc.
 - Explore clause search space from general to specific or vice versa to find a hypothesis that covers all examples.



- **Inverse entailment** [?]: e.g., Progol, etc.
 - Properties of deduction to make hypothesis search space finite

Zoo of Other ILP Tasks

ILP tasks can be classified along several dimensions:

- type of the data source, e.g., positive/negative examples, interpretations, answer sets [?]
- type of the output knowledge, e.g., rules, DL ontologies [?]
- the way the data is given as input, e.g., all at once, incrementally [?]
- availability of an oracle, e.g., human in the loop
- quality of the data source, e.g., noisy [?]
- data (in)completeness, e.g., OWA vs CWA...
- background knowledge, e.g., DL ontology [?], hybrid theories [?]

Classical ILP for KGs

ILP Goal

"The goal of ILP is to develop a correct (and complete) algorithm which efficiently computes hypotheses." [?]

Knowledge Graphs

But the world knowledge is complex, and this might not always be possible in the context of KGs due to several issues...

Specialities of KGs

Open World Assumption: negative facts cannot be easily derived

Maybe Roger Federer is a researcher and Albert Einstein was a ballet dancer?

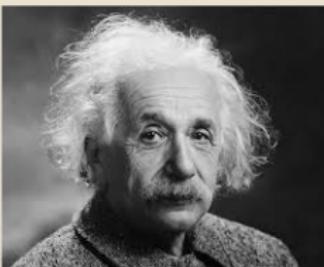
Specialities of KGs

Open World Assumption: negative facts cannot be easily derived

Maybe Roger Federer is a researcher and Albert Einstein was a ballet dancer?

We dance for laughter,
we dance for tears,
we dance for madness,
we dance for fears,
we dance for hopes,
we dance for screams,
we are the dancers,
we create the dreams.

-Albert Einstein



Challenges of Rule Induction from KGs

Data bias: KGs are extracted from text, which typically mentions only popular entities and interesting facts about them.

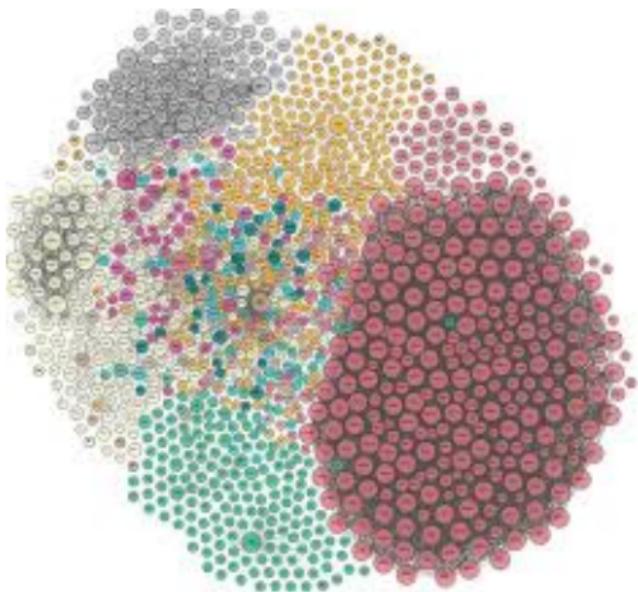
“Man bites dog phenomenon”¹



¹[https://en.wikipedia.org/wiki/Man_bites_dog_\(journalism\)](https://en.wikipedia.org/wiki/Man_bites_dog_(journalism))

Challenges of Rule Induction from KGs

Huge size: Modern KGs contain billions of facts
E.g., Google KG stores 70 billion facts



Challenges of Rule Induction from KGs

World knowledge is complex, none of its “models” is perfect



Exploratory Data Analysis

Question:

How can we still learn rules from KGs, which do not perfectly fit the data, but still reflect interesting correlations that can predict sufficiently many correct facts?

Answer:

Relational association rule mining! Roots in classical datamining.



Association Rules

- Classical data mining task: Given a transaction database, find out products (called itemsets) that are frequently bought together and form recommendation rules.

Transaction 1	🍎	🍺	⌚	🌯
Transaction 2	🍎	🍺	⌚	
Transaction 3	🍎	🍺		
Transaction 4	🍎	🍐		
Transaction 5	🍼	🍺	⌚	🌯
Transaction 6	🍼	🍺	⌚	
Transaction 7	🍼	🍺		
Transaction 8	🍼	🍐		

Out of 4 people who bought apples, 3 also bought beer.

Some Rule Measures

Support, confidence, lift

Support [apple] = 4

Transaction 1	apple	beer	bowl	meat
Transaction 2	apple	beer	bowl	
Transaction 3	apple	beer		
Transaction 4	apple	pear		
Transaction 5	milk	beer	bowl	meat
Transaction 6	milk	beer	bowl	
Transaction 7	milk	beer		
Transaction 8	milk	pear		

Some Rule Measures

Support, confidence, lift

Support [🍎] = 4

$$\text{Confidence } \{ \text{🍎} \rightarrow \text{🍺} \} = \frac{\text{Support } \{ \text{🍎}, \text{🍺} \}}{\text{Support } \{ \text{🍎} \}}$$

Transaction 1	🍎	🍺	一碗	一块肉
Transaction 2	🍎	🍺	一碗	
Transaction 3	🍎	🍺		
Transaction 4	🍎	🍐		
Transaction 5	一瓶水	🍺	一碗	一块肉
Transaction 6	一瓶水	🍺	一碗	
Transaction 7	一瓶水	🍺		
Transaction 8	一瓶水	🍐		

Some Rule Measures

Support, confidence, lift

$$\text{Support } \{\text{apple}\} = 4$$

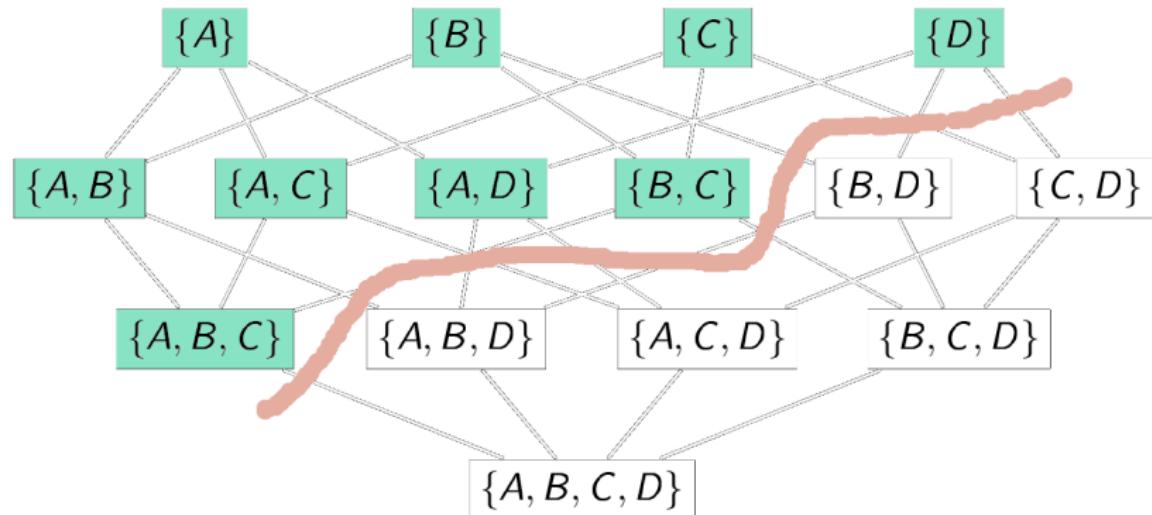
$$\text{Confidence } \{\text{apple} \rightarrow \text{beer}\} = \frac{\text{Support } \{\text{apple}, \text{beer}\}}{\text{Support } \{\text{apple}\}}$$

$$\text{Lift } \{\text{apple} \rightarrow \text{beer}\} = \frac{\text{Support } \{\text{apple}, \text{beer}\}}{\text{Support } \{\text{apple}\} \times \text{Support } \{\text{beer}\}}$$

Transaction 1	
Transaction 2	
Transaction 3	
Transaction 4	
Transaction 5	
Transaction 6	
Transaction 7	
Transaction 8	

Frequent Itemset Mining

- A=apple, B=beer... Frequent patterns are in green.
- Monotonicity: any superset of an infrequent pattern is infrequent
At the heart of Apriori algorithm



Relational Association Rule Learning

- **WARMER [?]**
- Upgrade frequent itemsets to frequent conjunctive queries

CQ: return all people with their spouses and living places

$$q_1(X, Y, Z) : \neg \text{isMarriedTo}(X, Y) \wedge \text{livesIn}(X, Z)$$

Output: 6 tuples, i.e., $\text{supp}(q_1) = 6$

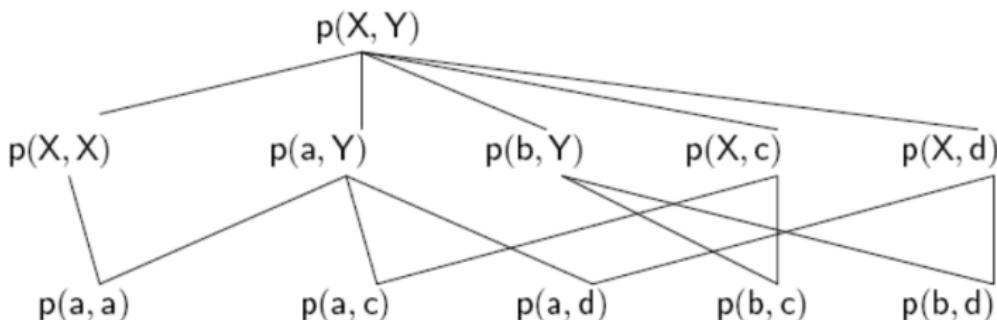
CQ: return all people with their spouses and living places

$$q_2(X, Y, Z) : \neg \text{isMarriedTo}(X, Y) \wedge \text{livesIn}(X, Z) \wedge \text{livesIn}(Y, Z)$$

Output: 3 tuples, i.e., $\text{supp}(q_2) = 3$

Relational Association Rule Learning

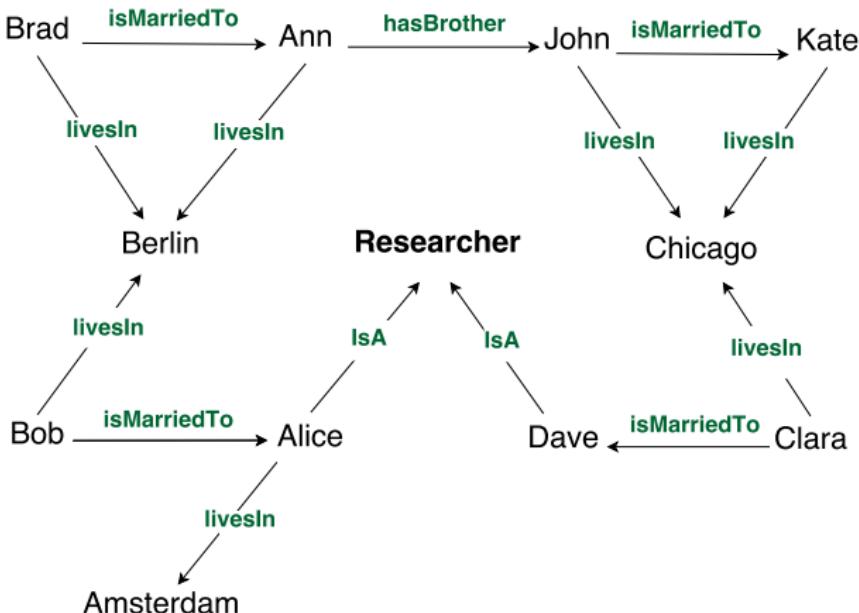
- **WARMER [?]**
- Upgrade frequent itemsets to frequent conjunctive queries
 - traverse the lattice
 - get frequent CQs based on user-specified value
 - split into body and head
 - rank based on a rule measure, e.g., confidence



Horn Rule Learning from KGs

WARMER: confidence

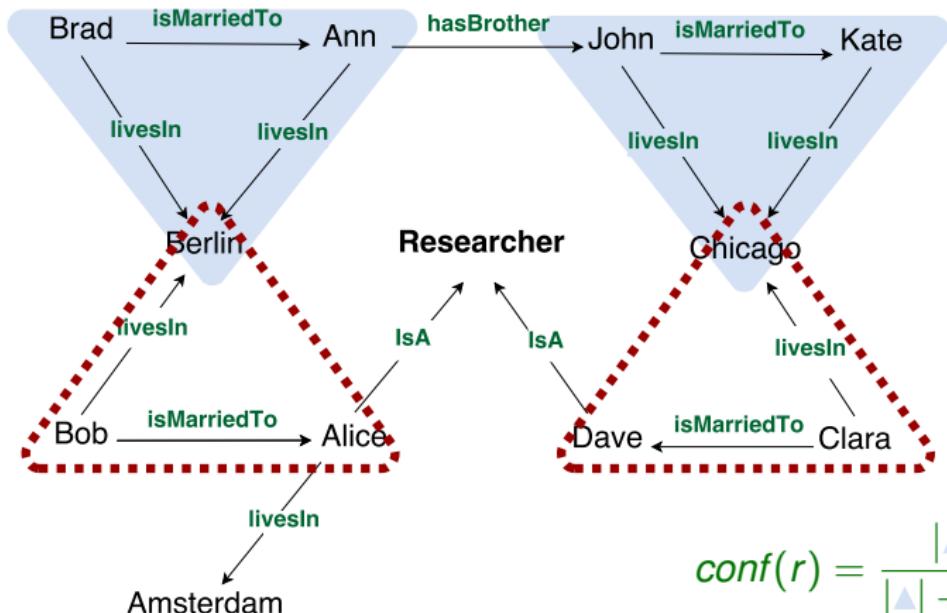
CWA: Whatever is not known is false.



Horn Rule Learning from KGs

WARMER: confidence

CWA: Whatever is not known is false.



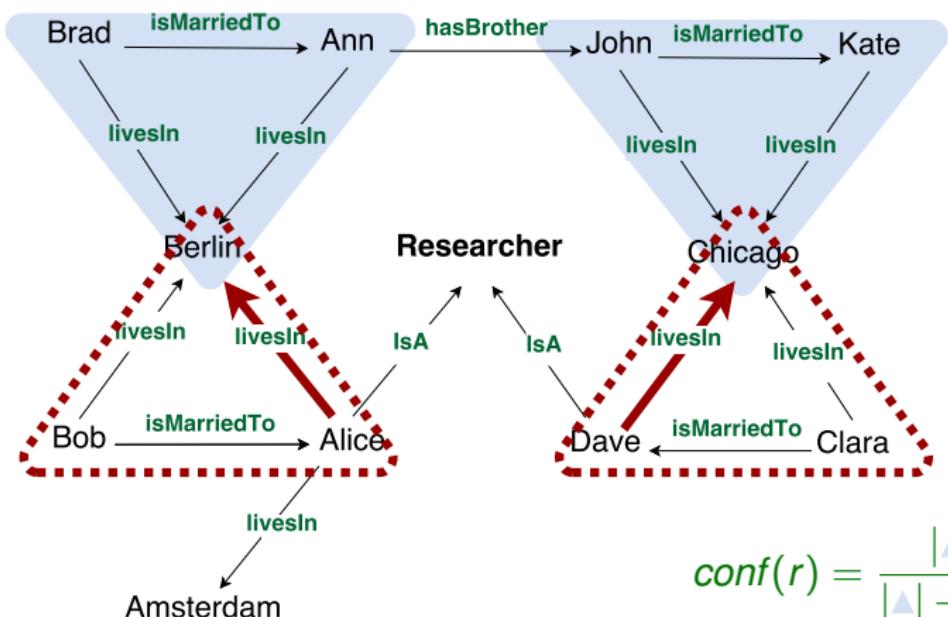
$$conf(r) = \frac{|\text{blue} \cap \text{red}|}{|\text{blue}| + |\text{red}|} = \frac{2}{4}$$

$r : \text{livesIn}(X, Z) \leftarrow \text{isMarriedTo}(Y, X), \text{livesIn}(Y, Z)$

Horn Rule Learning from KGs

WARMER: confidence

CWA: Whatever is not known is false.



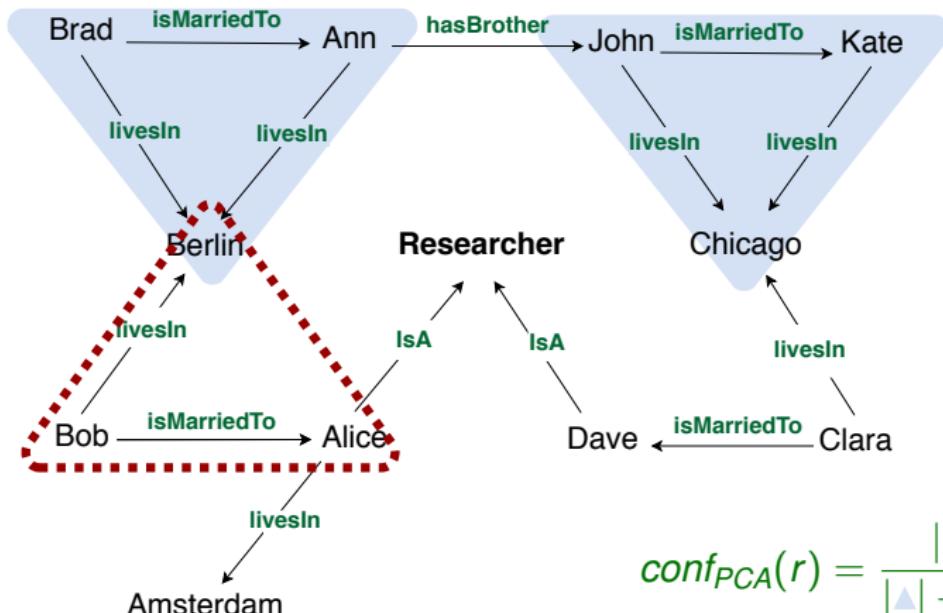
$$\text{conf}(r) = \frac{|\text{blue triangles}|}{|\text{blue triangles}| + |\text{red triangles}|} = \frac{2}{4}$$

$$r : \text{livesIn}(X, Z) \leftarrow \text{isMarriedTo}(Y, X), \text{livesIn}(Y, Z)$$

Horn Rule Learning from KGs

AMIE [?]: PCA confidence

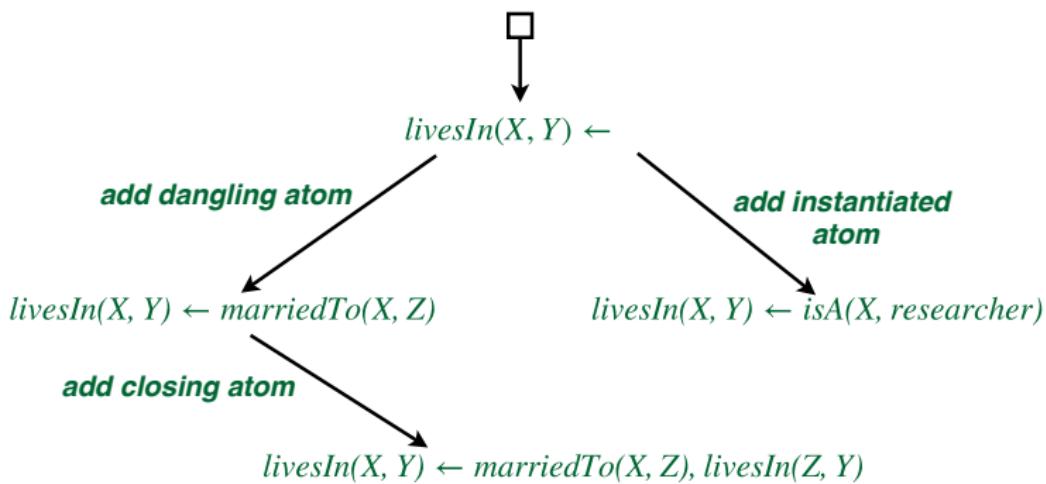
PCA: If at least 1 living place of Alice is known, then all are known.



$$conf_{PCA}(r) = \frac{|\text{blue} \cap \text{green}|}{|\text{blue}| + |\text{green}|} = \frac{2}{3}$$

$r : livesIn(X, Z) \leftarrow isMarriedTo(Y, X), livesIn(Y, Z)$

AMIE Refinement Operators



Motivation

Preliminaries

Rule Learning

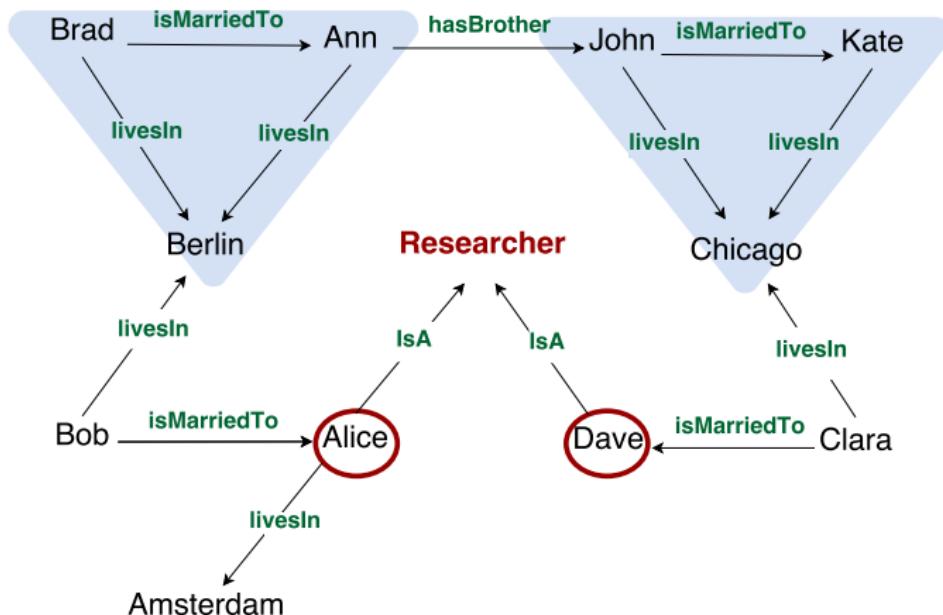
Exception-awareness

Incompleteness

Rules from Hybrid Sources

Nonmonotonic Rule Learning

Nonmonotonic rule mining from KGs: OWA is a challenge!



$r : \text{livesIn}(X, Z) \leftarrow \text{isMarriedTo}(Y, X), \text{livesIn}(Y, Z), \text{not researcher}(X)$

Horn Theory Revision

Quality-based Horn Theory Revision

Given:

- Available KG

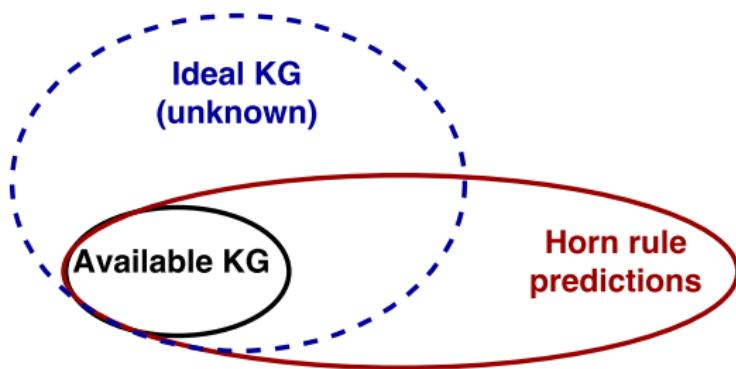


Horn Theory Revision

Quality-based Horn Theory Revision

Given:

- Available KG
- Horn rule set

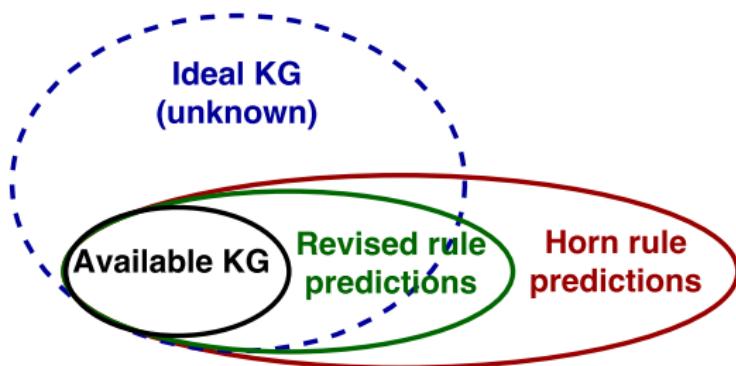


Horn Theory Revision

Quality-based Horn Theory Revision

Given:

- Available KG
- Horn rule set



Find:

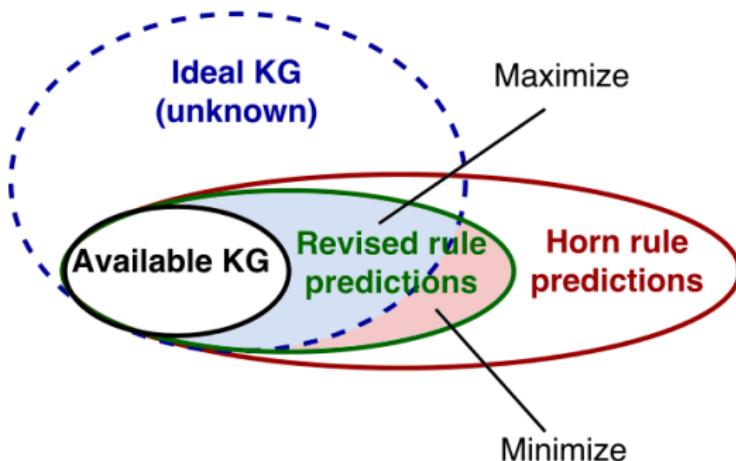
- Nonmonotonic revision of Horn rule set

Horn Theory Revision

Quality-based Horn Theory Revision

Given:

- Available KG
- Horn rule set



Find:

- Nonmonotonic revision of Horn rule set with better predictive quality

Avoid Data Overfitting

How to distinguish exceptions from noise?

$r1 : \text{livesIn}(X, Z) \leftarrow \text{isMarriedTo}(Y, X), \text{livesIn}(Y, Z), \text{not researcher}(X)$

Avoid Data Overfitting

How to distinguish exceptions from noise?

$r1 : livesIn(X, Z) \leftarrow isMarriedTo(Y, X), livesIn(Y, Z), \text{not researcher}(X)$
 $\text{not_livesIn}(X, Z) \leftarrow isMarriedTo(Y, X), livesIn(Y, Z), researcher(X)$

Avoid Data Overfitting

How to distinguish exceptions from noise?

$r1 : livesIn(X, Z) \leftarrow isMarriedTo(Y, X), livesIn(Y, Z), \text{not researcher}(X)$
 $\text{not_livesIn}(X, Z) \leftarrow isMarriedTo(Y, X), livesIn(Y, Z), researcher(X)$

$r2 : livesIn(X, Z) \leftarrow bornIn(X, Z), \text{not moved}(X)$
 $\text{not_livesIn}(X, Z) \leftarrow bornIn(X, Z), moved(X)$

Avoid Data Overfitting

How to distinguish exceptions from noise?

$r1 : \text{livesIn}(X, Z) \leftarrow \text{isMarriedTo}(Y, X), \text{livesIn}(Y, Z), \text{not researcher}(X)$
 $\text{not_livesIn}(X, Z) \leftarrow \text{isMarriedTo}(Y, X), \text{livesIn}(Y, Z), \text{researcher}(X)$

$r2 : \text{livesIn}(X, Z) \leftarrow \text{bornIn}(X, Z), \text{not moved}(X)$
 $\text{not_livesIn}(X, Z) \leftarrow \text{bornIn}(X, Z), \text{moved}(X)$

$\{\text{livesIn}(c, d), \text{not_livesIn}(c, d)\}$ are conflicting predictions

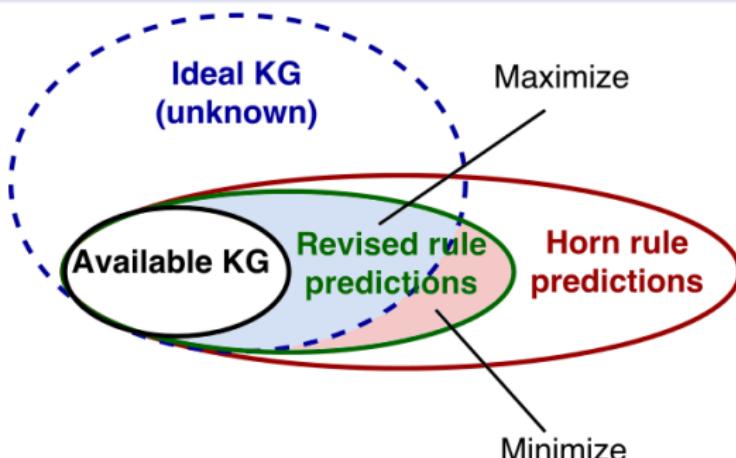
Intuition: Rules with good exceptions should make few conflicting predictions

Horn Theory Revision

Quality-based Horn Theory Revision

Given:

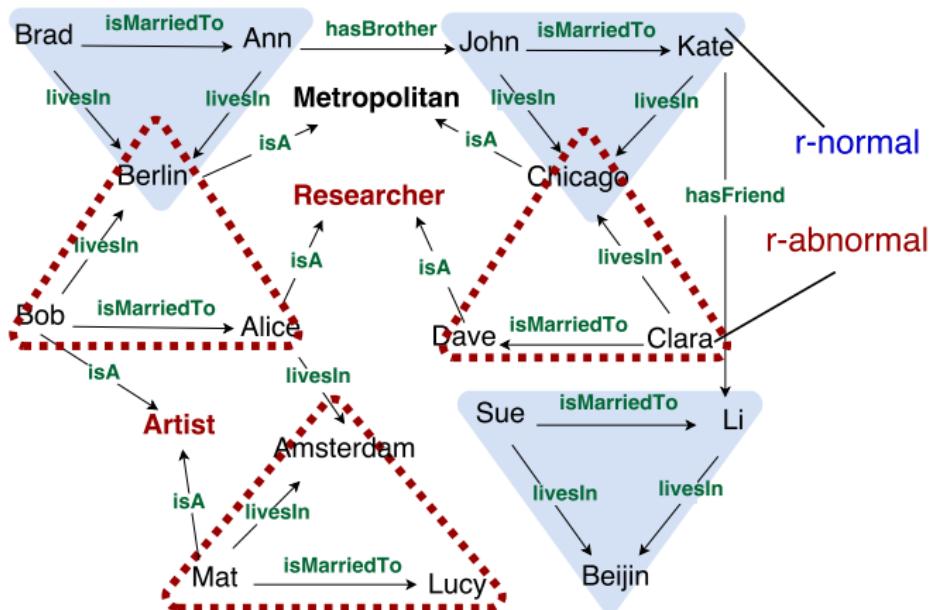
- Available KG
- Horn rule set



Find:

- Nonmonotonic revision of Horn rules, such that
 - number of **conflicting predictions** is **minimal**
 - average **conviction** is **maximal**

Exception Candidates

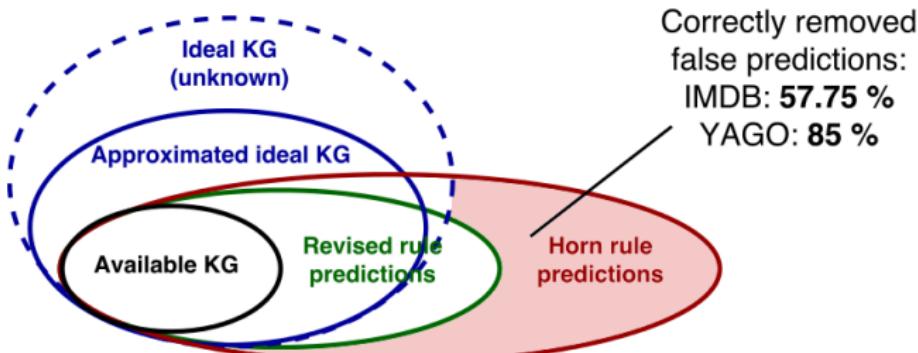


r: livesIn(X, Z) \leftarrow isMarriedTo(Y, X), livesIn(Y, Z)

$\{ \text{not researcher}(X) \}$
 $\{ \text{not artist}(Y) \}$

Experiments

- Approximated ideal KG: original KG
- Available KG: for every relation randomly remove 20% of facts from approximated ideal KG
- Horn rules: $h(X, Y) \leftarrow p(X, Z), q(Z, Y)$
- Exceptions: $e_1(X), e_2(Y), e_3(X, Y)$
- Predictions are computed using answer set solver DLV



Experiments

- Approximated ideal KG: original KG
- Available KG: for every relation randomly remove 20% of facts from approximated ideal KG
- Horn rules: $h(X, Y) \leftarrow p(X, Z), q(Z, Y)$
- Exceptions: $e_1(X), e_2(Y), e_3(X, Y)$
- Predictions are computed using answer set solver DLV

Examples of revised rules:

Plots of films in a sequel are written by the same writer, unless a film is American

$r_1 : \text{writtenBy}(X, Z) \leftarrow \text{hasPredecessor}(X, Y), \text{writtenBy}(Y, Z), \text{not american_film}(X)$

Spouses of film directors appear on the cast, unless they are silent film actors

$r_2 : \text{actedIn}(X, Z) \leftarrow \text{isMarriedTo}(X, Y), \text{directed}(Y, Z), \text{not silent_film_actor}(X)$

Motivation

Preliminaries

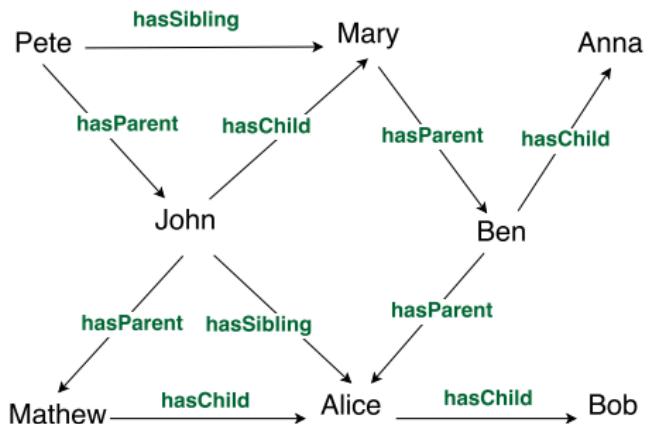
Rule Learning

Exception-awareness

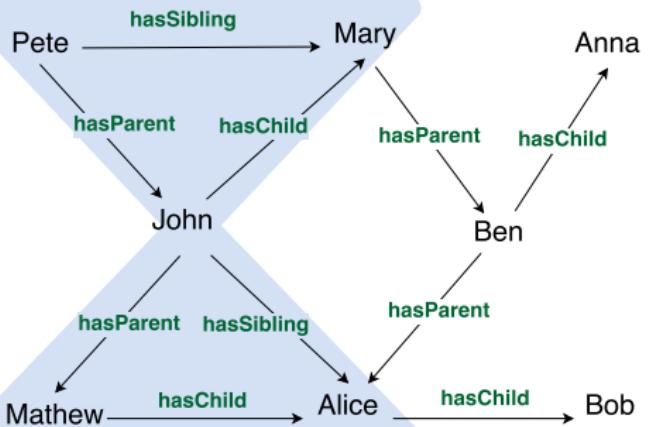
Incompleteness

Rules from Hybrid Sources

Reasonable Rules

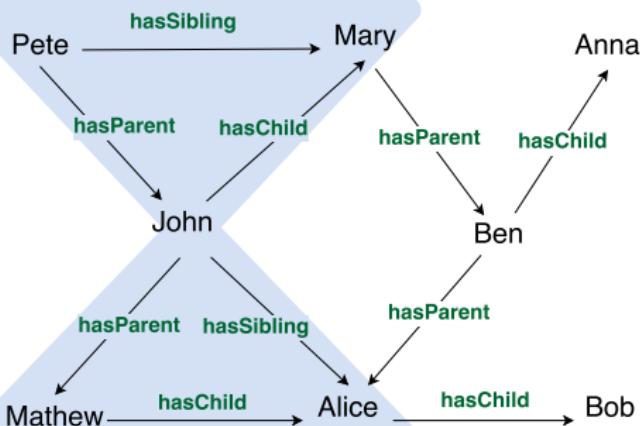


Reasonable Rules



Reasonable Rules

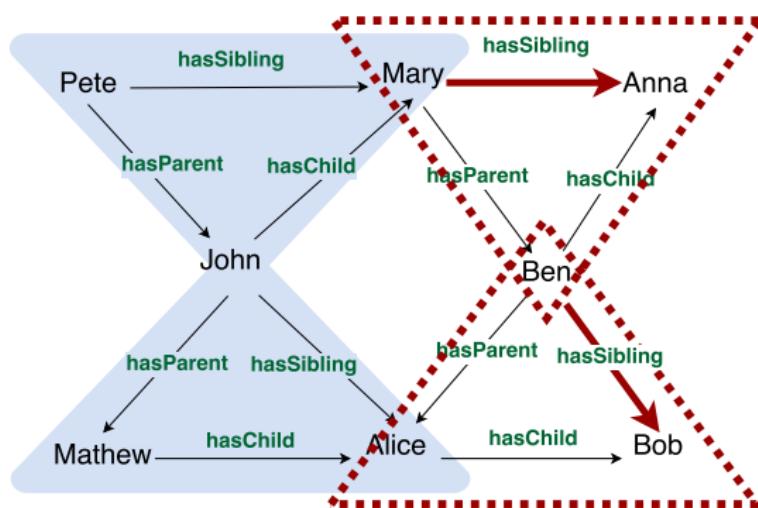
✓ *People with the same parents are likely siblings*



$r_1 : \text{hasSibling}(X, Z) \leftarrow \text{hasParent}(X, Y), \text{hasChild}(Y, Z)$

Reasonable Rules

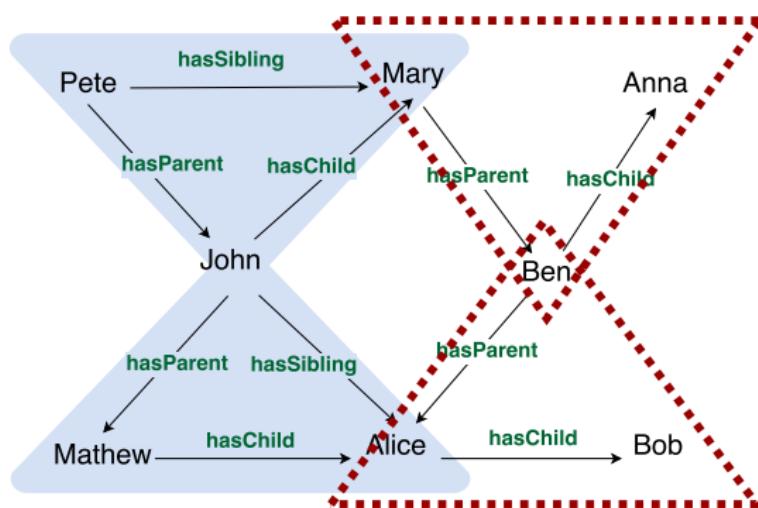
- ✓ People with the same parents are likely siblings



$r_1 : \text{hasSibling}(X, Z) \leftarrow \text{hasParent}(X, Y), \text{hasChild}(Y, Z)$

Reasonable Rules

- ✓ People with the same parents are likely siblings

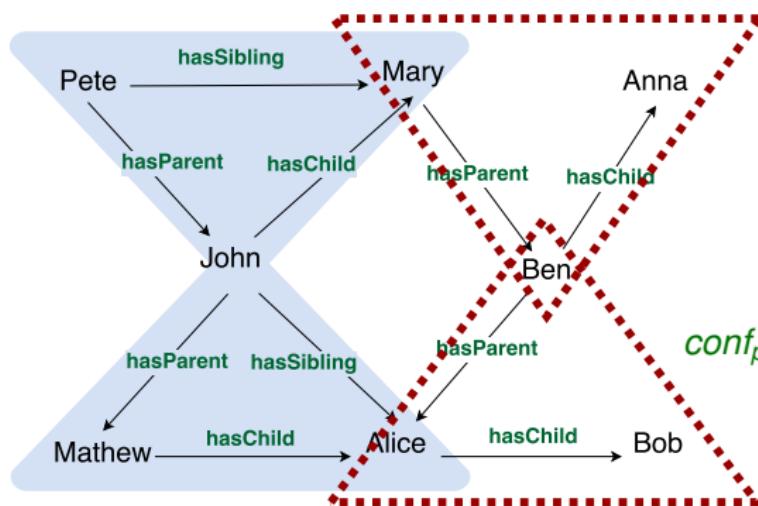


$$conf(r_1) = \frac{|\Delta|}{|\Delta| + |\Delta|} = \frac{2}{4}$$

$r_1 : hasSibling(X, Z) \leftarrow hasParent(X, Y), hasChild(Y, Z)$

Reasonable Rules

✓ People with the same parents are likely siblings

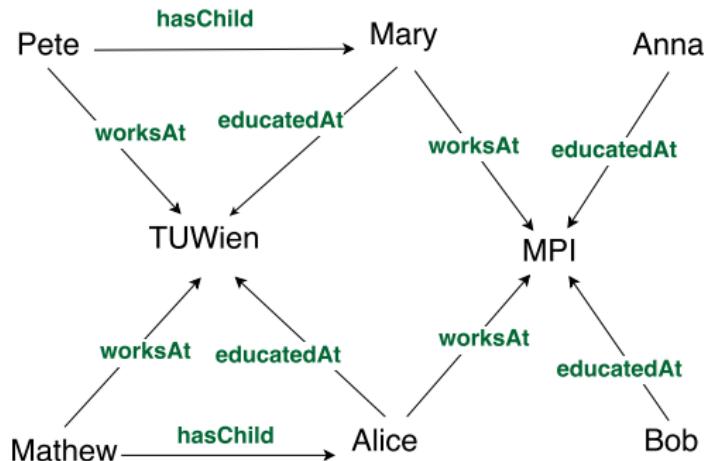


$$conf(r_1) = \frac{|\Delta|}{|\Delta| + |\Delta|} = \frac{2}{4}$$

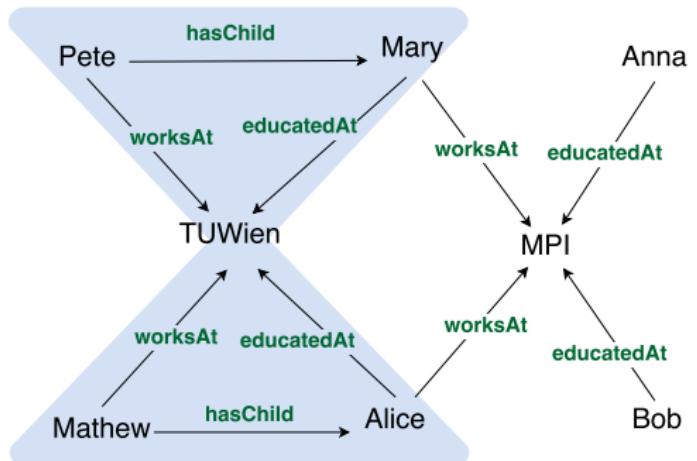
$$conf_{pca}(r_1) = \frac{|\Delta|}{|\{\Delta | hasSibling(X, -) \in \mathcal{G}\}|} = \frac{2}{2}$$

$r_1 : hasSibling(X, Z) \leftarrow hasParent(X, Y), hasChild(Y, Z)$

Erroneous Rules due to Data Bias

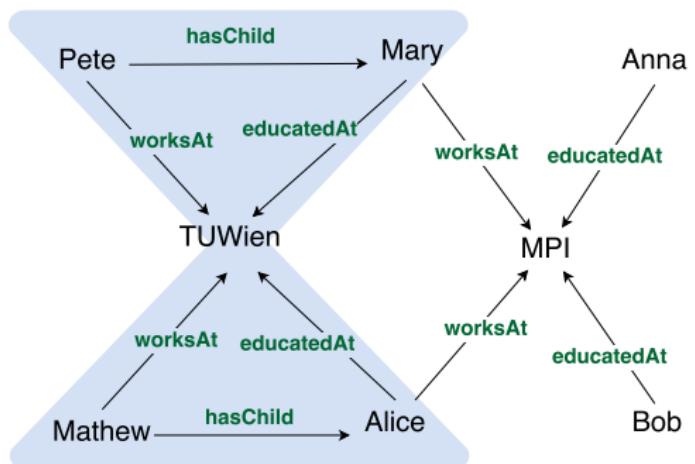


Erroneous Rules due to Data Bias



Erroneous Rules due to Data Bias

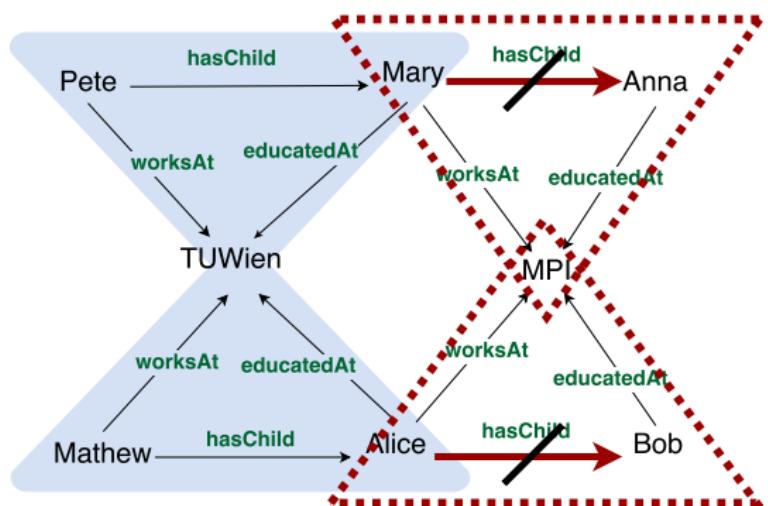
✗ If one is studying in a university where you teach, he/she is your child



$r_2 : \text{hasChild}(X, Z) \leftarrow \text{worksAt}(X, Y), \text{educatedAt}(Z, Y)$

Erroneous Rules due to Data Bias

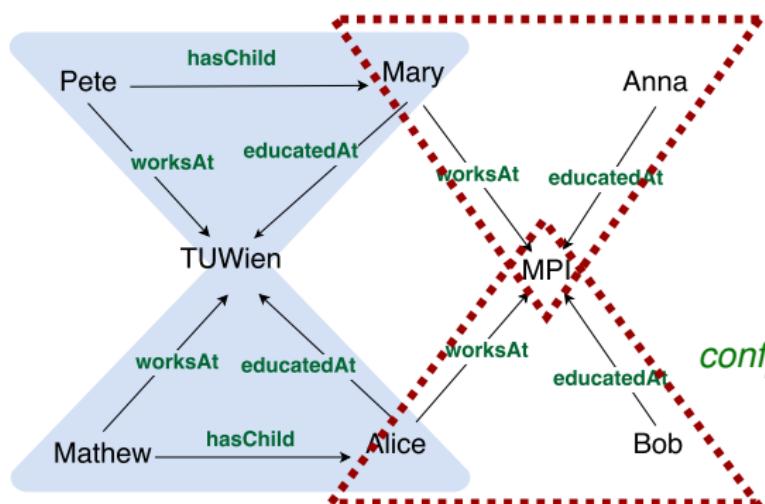
✗ If one is studying in a university where you teach, he/she is your child



$$r_2 : \text{hasChild}(X, Z) \leftarrow \text{worksAt}(X, Y), \text{educatedAt}(Z, Y)$$

Erroneous Rules due to Data Bias

✗ If one is studying in a university where you teach, he/she is your child



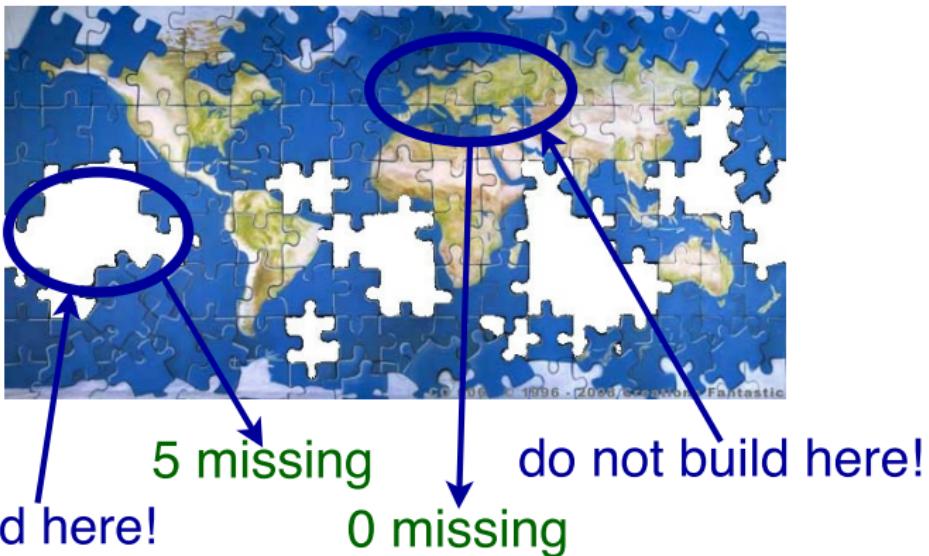
$$conf(r_2) = \frac{|\Delta|}{|\Delta| + |\Delta^c|} = \frac{2}{4}$$

$$conf_{pca}(r_2) = \frac{|\Delta|}{|\{\Delta | hasChild(X, -) \in \mathcal{G}\}|} = \frac{2}{2}$$

$r_2 : hasChild(X, Z) \leftarrow worksAt(X, Y), educatedAt(Z, Y)$

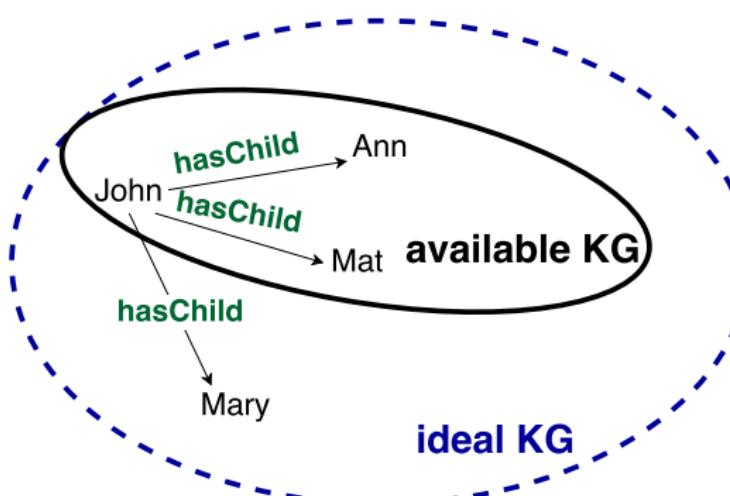
Exploiting Meta-data in Rule Learning

Goal: make use of cardinality constraints on edges of the KG to improve rule learning.



Cardinality Statements

- $\text{num}(p, s)$: Number of outgoing p -edges from s in the ideal KG
- $\text{miss}(p, s)$: Number of missing p -edges from s in the available KG
- If $\text{miss}(p, s) = 0$, then $\text{complete}(p, s)$, otherwise $\text{incomplete}(p, s)$



$\text{num}(\text{hasChild}, \text{john}) = 3$
 $\text{miss}(\text{hasChild}, \text{john}) = 1$
 $\text{incomplete}(\text{hasChild}, \text{john})$

Cardinality Constraints on Edges

- Mining cardinality assertions from the Web [?]
 - *... John has 2 children ...*
- Estimating recall of KGs by crowd sourcing [?]
 - *20 % of Nobel laureates in physics are missing*
- Predicting completeness in KGs [?]
 - Add *complete(john, hasChild)* to KG and mine rules
complete(X, hasChild) ← child(X)

Completeness Confidence

$conf_{comp}$: do not penalize rules that predict new facts in incomplete areas

$$conf_{comp}(r) = \frac{|\Delta|}{|\Delta| + |\Delta^c| - npi(r)}$$

- $npi(r)$: number of facts added to incomplete areas by r
- Generalizes standard confidence ($miss(r) = 0$)
- Generalizes PCA confidence ($miss(r) \in \{0, +\infty\}$)

Other Completeness-aware Measures

$precision_{comp}$: penalize r that predict facts in complete areas

$$precision_{comp}(r) = 1 - \frac{npc(r)}{|\Delta| + |\triangle|}$$

$recall_{comp}$: ratio of missing facts filled by r

$$recall_{comp}(r) = \frac{npi(r)}{\sum_s miss(h, s)}$$

dir_metric : proportion of predictions in complete and incomplete parts

$$dir_metric(r) = \frac{npi(r) - npc(r)}{2 \cdot (npi(r) + npc(r))} + 0.5$$

wdm : weighted combination of confidence and directional metric

$$wdm(r) = \beta \cdot conf(r) + (1 - \beta) \cdot dir_metric(r)$$

Motivation

Preliminaries

Rule Learning

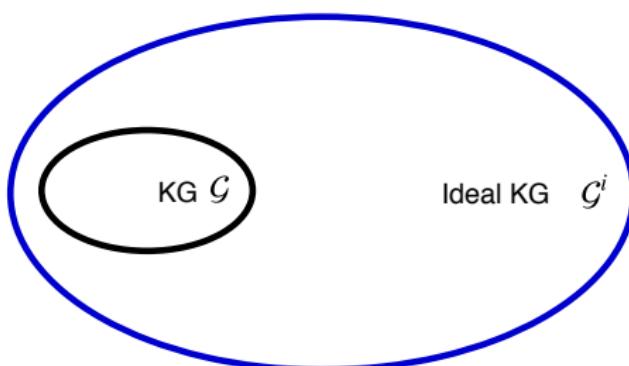
Exception-awareness

Incompleteness

Rules from Hybrid Sources

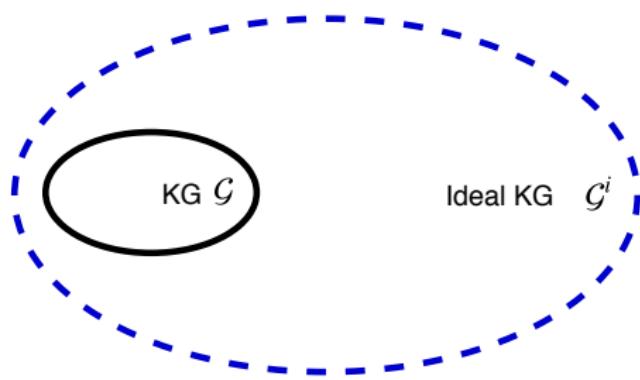
Ideal KG

$\mu(r, \mathcal{G}^i)$: measure quality of the rule r on \mathcal{G}^i



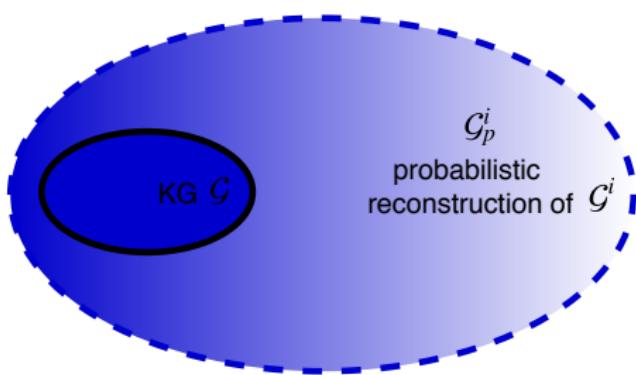
Ideal KG

$\mu(r, \mathcal{G}^i)$: measure quality of the rule r on \mathcal{G}^i , but \mathcal{G}^i is unknown



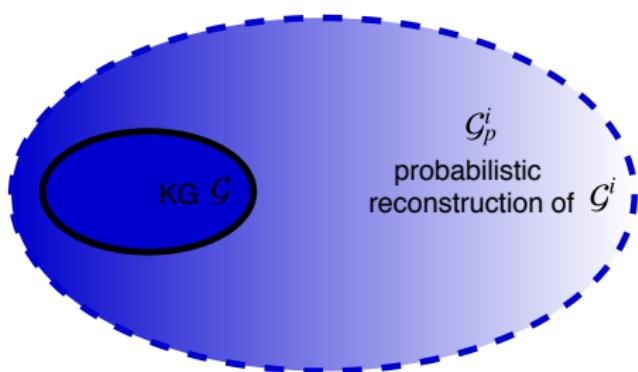
Probabilistic Reconstruction of Ideal KG

$\mu(r, \mathcal{G}_p^i)$: measure quality of r on \mathcal{G}_p^i



Hybrid Rule Measure

$$\mu(r, \mathcal{G}_p^i) = (1 - \lambda) \times \mu_1(r, \mathcal{G}) + \lambda \times \mu_2(r, \mathcal{G}_p^i)$$



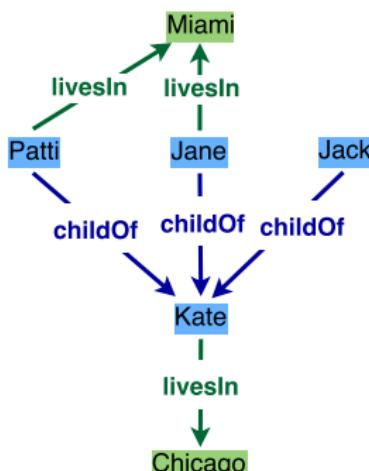
Hybrid Rule Measure

$$\mu(r, \mathcal{G}_p^i) = (1 - \lambda) \times \mu_1(r, \mathcal{G}) + \lambda \times \mu_2(r, \mathcal{G}_p^i)$$

- $\lambda \in [0..1]$: **weighting factor**
- μ_1 : **descriptive quality** of rule r over the available KG \mathcal{G}
 - confidence
 - PCA confidence
- μ_2 : **predictive quality** of r relying on \mathcal{G}_p^i (probabilistic reconstruction of the ideal KG \mathcal{G}^i)

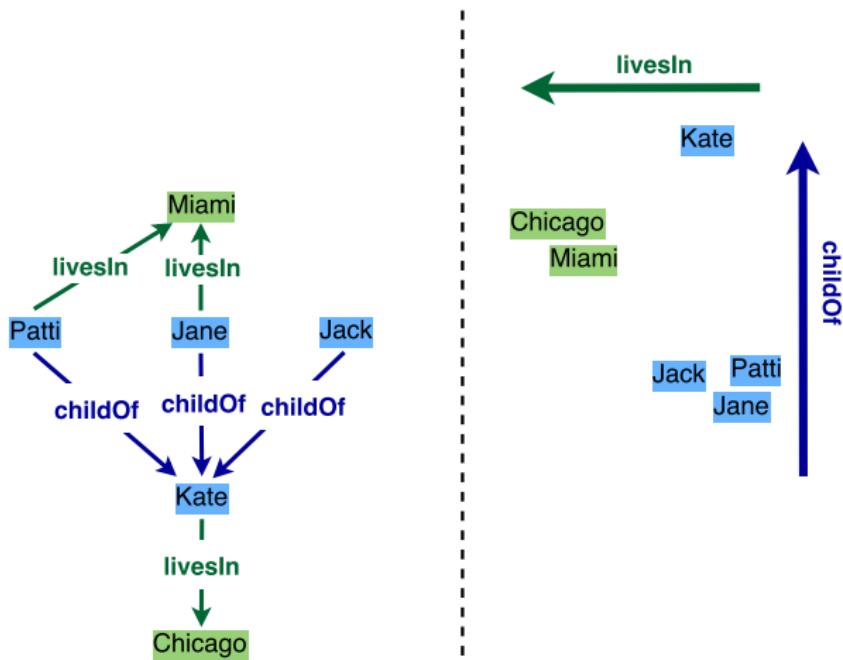
KG Embeddings

- **Intuition:** For $\langle s, p, o \rangle$ in KG, find s, p, o such that $s + p \approx o$
- The “error of translation” of a true KG fact should be smaller by a certain margin than the “error of translation” of an out-of-KG one



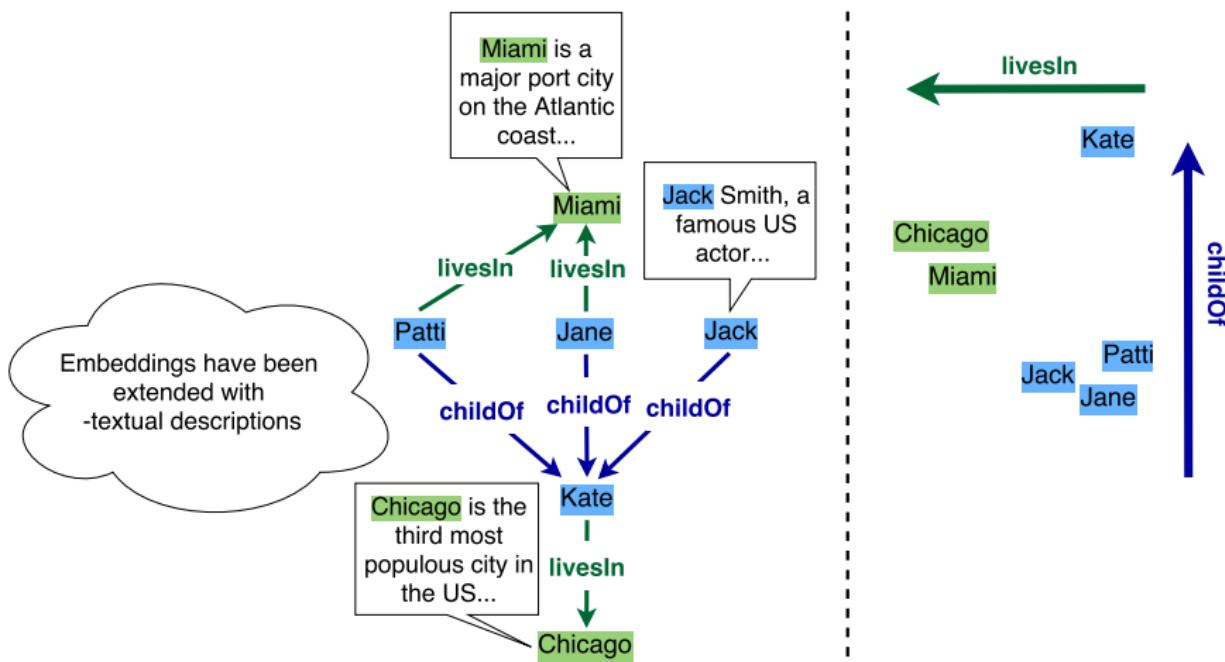
KG Embeddings

- **Intuition:** For $\langle s, p, o \rangle$ in KG, find s, p, o such that $s + p \approx o$
- The “error of translation” of a true KG fact should be smaller by a certain margin than the “error of translation” of an out-of-KG one



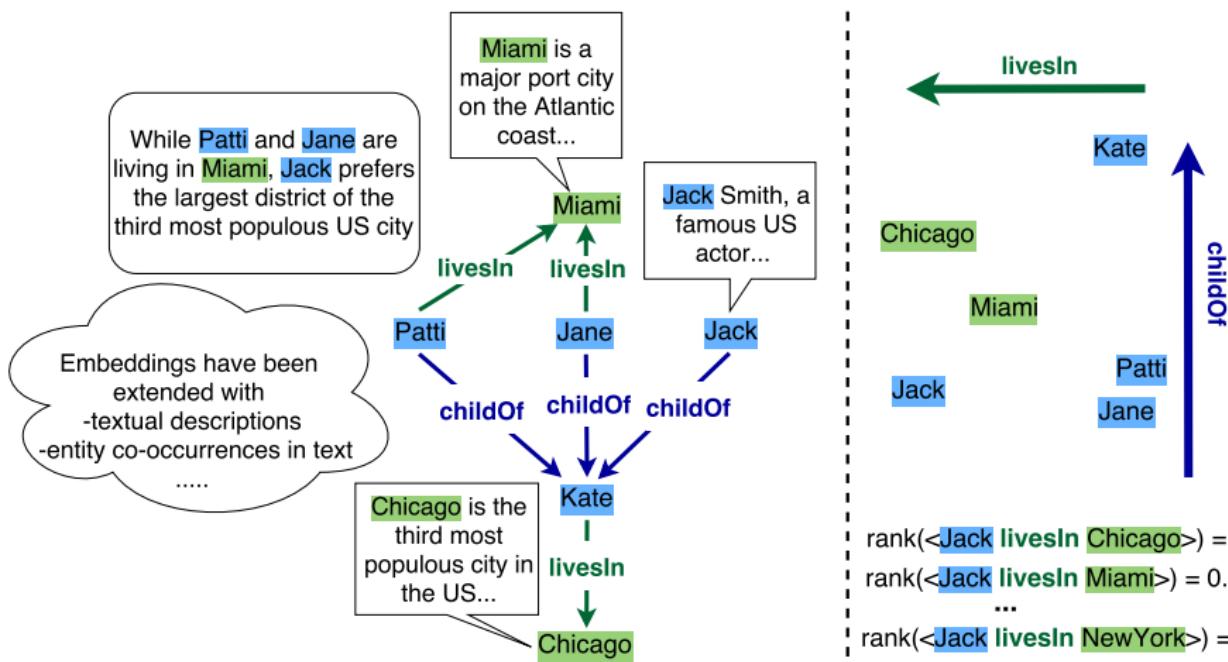
KG Embeddings

- **Intuition:** For $\langle s, p, o \rangle$ in KG, find s, p, o such that $s + p \approx o$
- The “error of translation” of a true KG fact should be smaller by a certain margin than the “error of translation” of an out-of-KG one

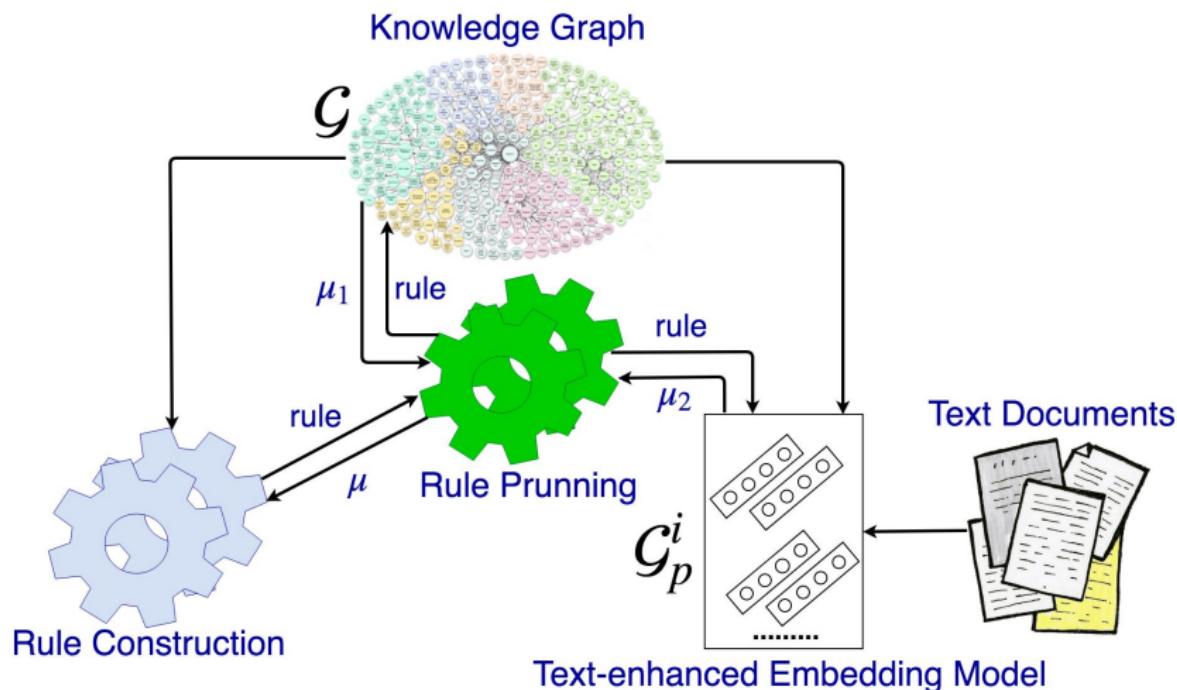


KG Embeddings

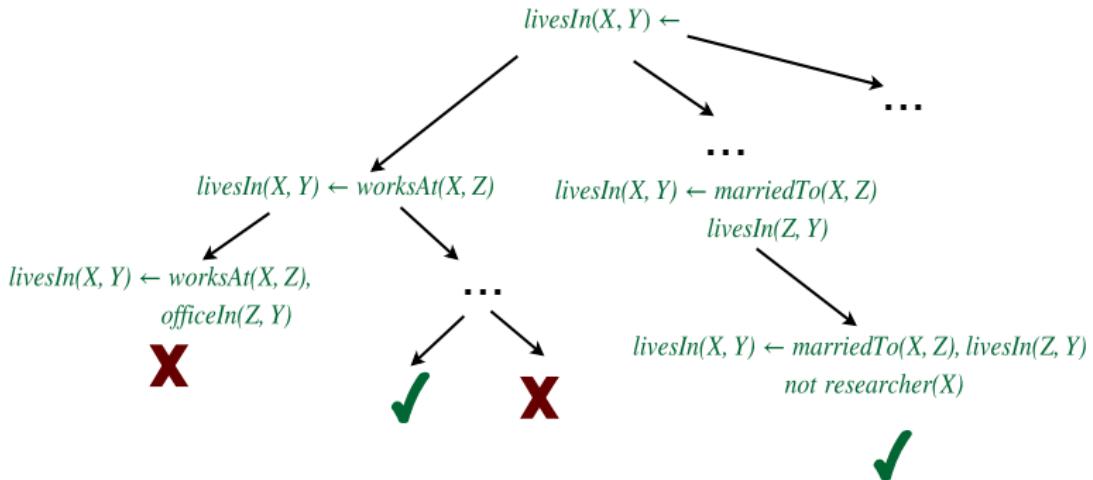
- **Intuition:** For $\langle s, p, o \rangle$ in KG, find s, p, o such that $s + p \approx o$
- The “error of translation” of a true KG fact should be smaller by a certain margin than the “error of translation” of an out-of-KG one



Embedding-based Rule Learning



Rule Pruning



Prune rule search space relying on

- novel hybrid embedding-based rule measure

Evaluation Setup

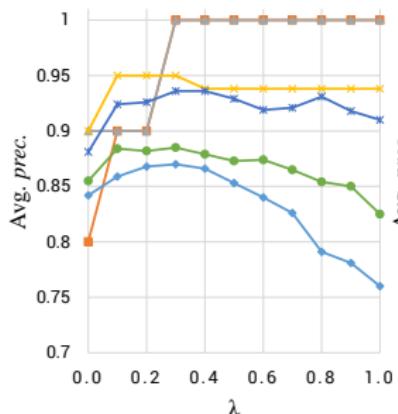
- Datasets:
 - FB15K: 592K facts, 15K entities and 1345 relations
 - Wiki44K: 250K facts, 44K entities and 100 relations
- Training graph \mathcal{G} : remove 20% from the available KG
- Embedding models \mathcal{G}_p^i :
 - TransE [?], HoIE [?]
 - With text: SSP [?]
- Goals:
 - Evaluate effectiveness of our hybrid rule measure

$$\mu(r, \mathcal{G}_p^i) = (1 - \lambda) \times \mu_1(r, \mathcal{G}) + \lambda \times \mu_2(r, \mathcal{G}_p^i)$$

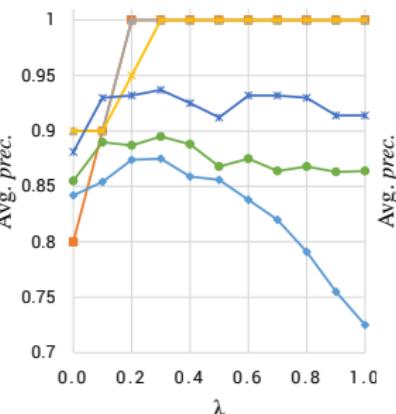
- Compare against state-of-the-art rule learning systems

Evaluation of Hybrid Rule Measure

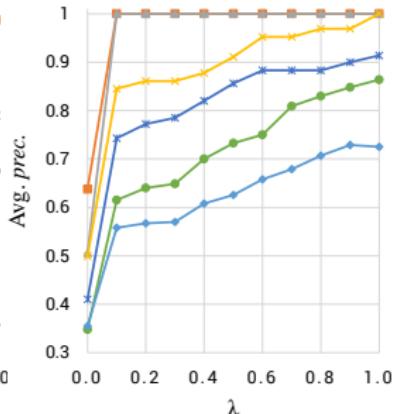
— top_5 — top_10 — top_20 — top_50 — top_100 — top_200



(a) Conf-HoIE

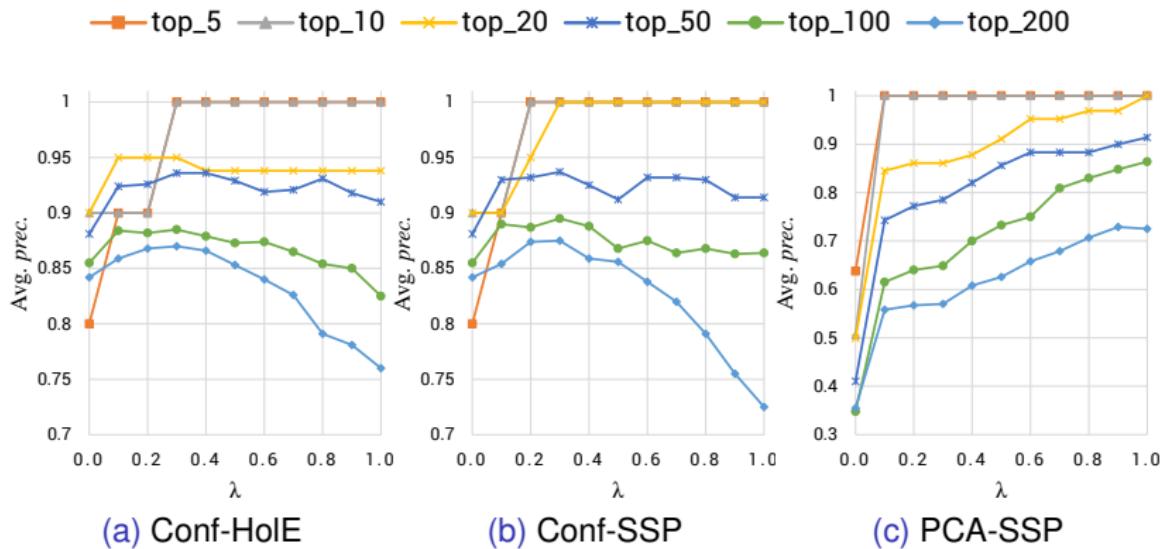


(b) Conf-SSP



(c) PCA-SSP

Evaluation of Hybrid Rule Measure



- Positive impact of embeddings in all cases for $\lambda = 0.3$
- Note:** in (c) comparison to AMIE [?] ($\lambda = 0$)

Example Rules

Examples of rules learned from Wikidata

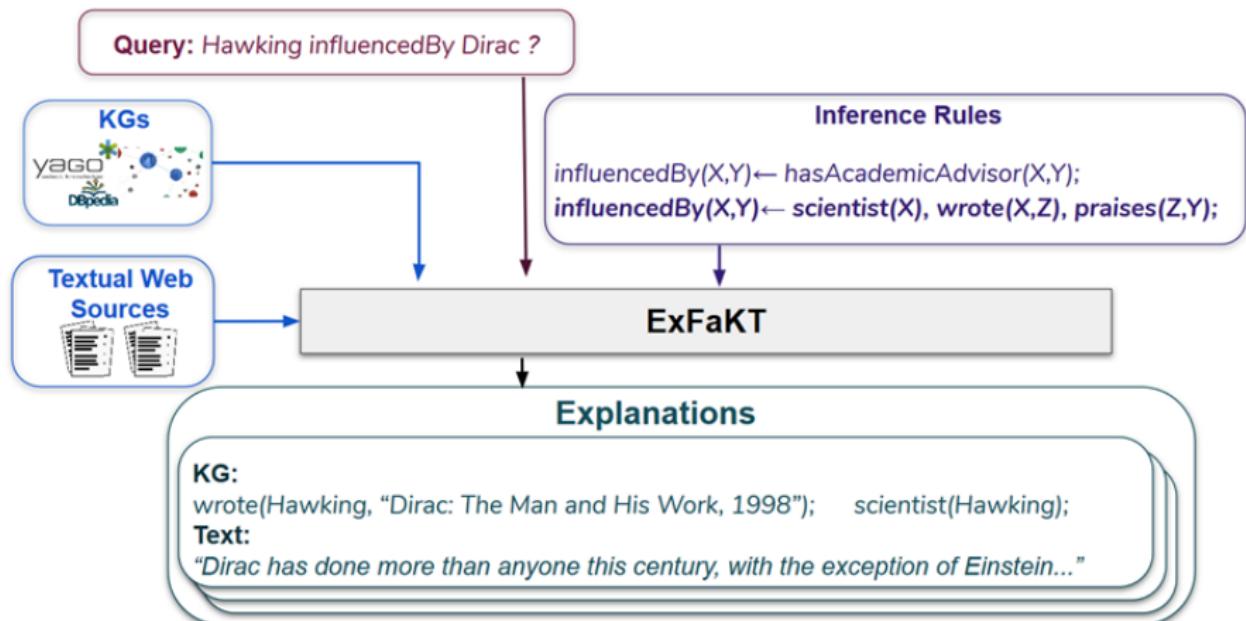
Script writers stay the same throughout a sequel, but not for TV series

$r_1 : \text{scriptwriterOf}(X, Y) \leftarrow \text{precededBy}(Y, Z), \text{scriptwriterOf}(X, Z), \text{not } \text{isA}(Z, \text{tvSeries})$

Nobles are typically married to nobles, but not in the case of Chinese dynasties

$r_2 : \text{nobleFamily}(X, Y) \leftarrow \text{spouse}(X, Z), \text{nobleFamily}(Z, Y), \text{not } \text{isA}(Y, \text{chineseDynasty})$

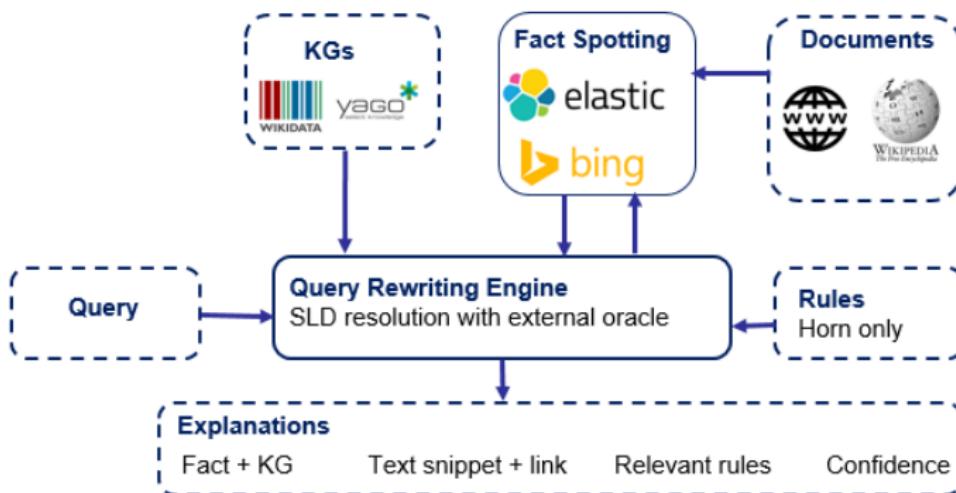
Rule-based Fact Checking



M. Gad-Elrab, D. Stepanova, J. Urbani, G. Weikum. ExFakt: A Framework for Explaining Facts over KGs and Text. WSDM 2019.

M. Gad-Elrab, D. Stepanova, J. Urbani, G. Weikum. Tracy: Tracing Facts over Knowledge Graphs and Text. WWW 2019.

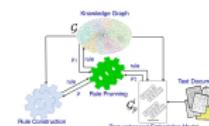
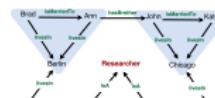
Rule-based Fact Checking



M. Gad-Elrab, D. Stepanova, J. Urbani, G. Weikum. *ExFakt: A Framework for Explaining Facts over KGs and Text*. WSDM 2019.
M. Gad-Elrab, D. Stepanova, J. Urbani, G. Weikum. *Tracy: Tracing Facts over Knowledge Graphs and Text*. WWW 2019.

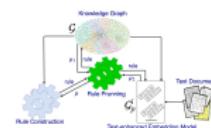
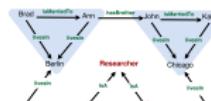
Summary

- Classical rule learning methods from ILP
- Rule learning from Knowledge Graphs
- Exploiting embeddings to guide rule learning
- Rule-based fact checking



Summary

- Classical rule learning methods from ILP
- Rule learning from Knowledge Graphs
- Exploiting embeddings to guide rule learning
- Rule-based fact checking



Interested in PhD/internship?
At BCAI we are hiring!

<https://jobs.smartrecruiters.com/BoschGroup/743999698936956-phd-combined-reasoning-and-learning-approaches>



Huge Thanks!

- For collaborations on the presented work:
 - Mohamed Gad-elrab, Thinh Vinh Ho, Hai Dang Tran, Thomas Pellissier-Tanon, Gerhard Weikum, Jacopo Urbani, Evgeny Kharlamov, Francesca A. Lisi, Simon Razniewski, Paramita Mirza
- For fruitful discussions and/or making slides available online:
 - Thomas Eiter, Stephen Muggleton, Luc De Raedt, Fabian Suchanek

References I