Rule Induction and Reasoning in Knowledge Graphs

Daria Stepanova

Bosch Center for Artificial Intelligence, Renningen, Germany

14.05.2020

Motivation

Rule Induction under Incompleteness

Numerical Rule Learning

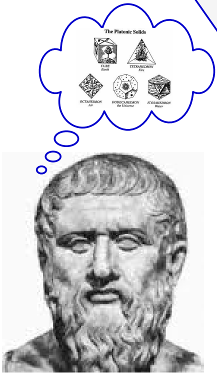
Rule-based Fact Checking

What is Knowledge?

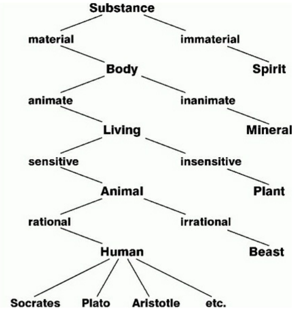
Plato: *"Knowledge is justified true belief"*

350 BC

Personal



External



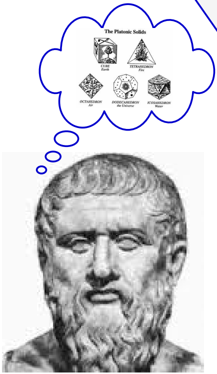
Tree of Knowledge

What is Knowledge?

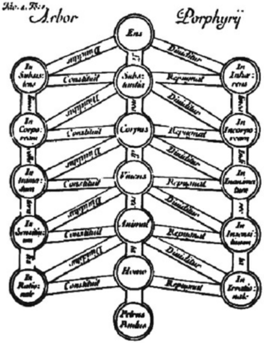
Plato: *“Knowledge is justified true belief”*

Personal

350 BC



External



Knowledge Graphs as Digital Knowledge

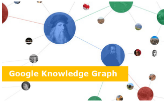
"Digital knowledge is semantically enriched machine processable data"

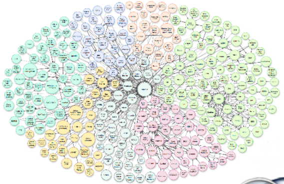
Personal

2020

External

yago
select knowledge



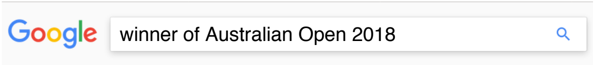


WIKIDATA

NELL

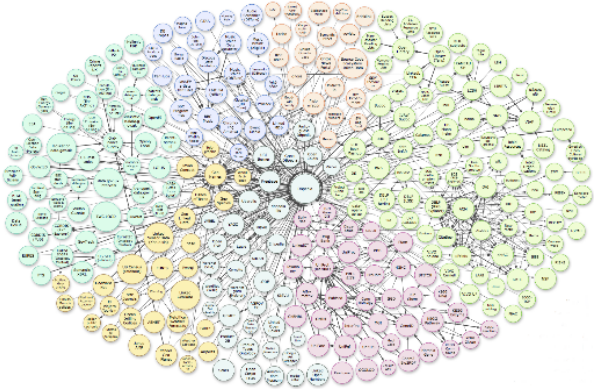
Knowledge Graph

Semantic Web Search



Semantic Web Search

$\exists X \; winnerOf(X, AustralianOpen2018)$



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

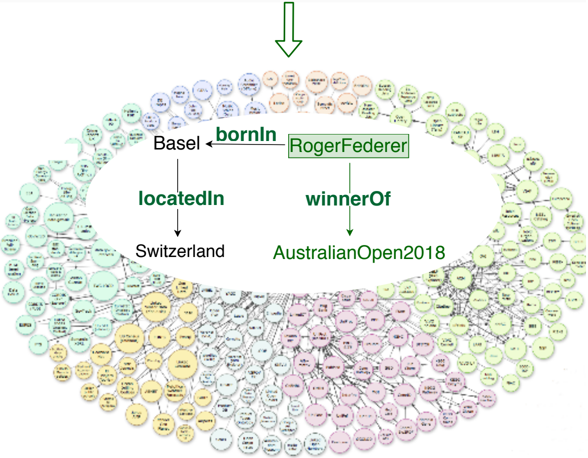
Weight: 85 kg

Spouse: Mirka Federer (m. 2009)

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

Semantic Web Search

Google $\exists X \; winnerOf(X, AustralianOpen2018)$



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

Industrial KGs

amazon

BOSCH

SIEMENS

Thousands of companies are developing their own KGs, not only for search and indexing but advanced reasoning tasks on top of machine learning

KG Incompleteness

Semantic Web Search

wife of Roger Federer

All | Images | News | Videos | Maps | More | Settings | Tools

About 42,200,000 results (0.50 seconds)

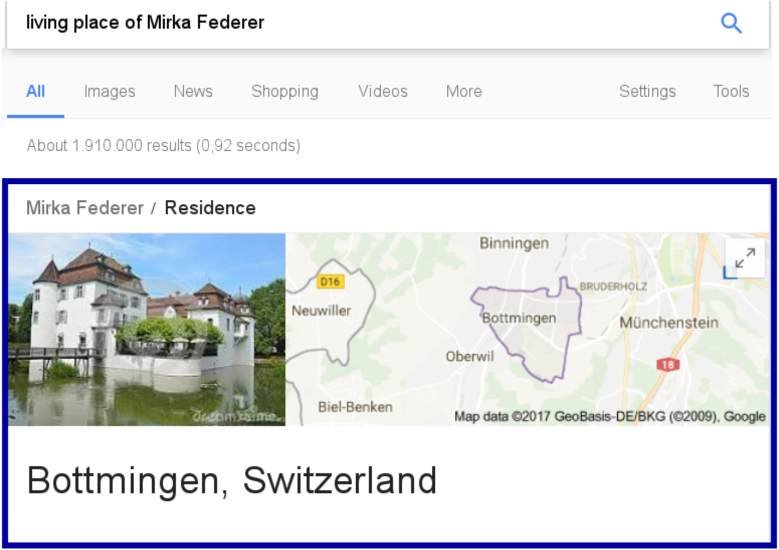
Roger Federer / Wife

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



Human Reasoning

$$livesIn(Y,Z) \leftarrow marriedTo(X,Y),\ livesIn(X,Z)$$

Married people live together

$marriedTo(mirka, roger)$

Mirka is married to Roger

$livesIn(mirka, bottmingen)$

Mirka lives in Bottmingen

Human Reasoning

$livesIn(Y, Z) \leftarrow marriedTo(X, Y), livesIn(X, Z)$	*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

$$livesIn(Y,Z) \leftarrow marriedTo(X,Y),\; livesIn(X,Z)$$

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

Rule Induction under Incompleteness

Numerical Rule Learning

Rule-based Fact Checking

Horn Rules

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

Informal semantics: If $b_1, \ldots, b_m$ 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

$livesIn(roger, bottmingen) \leftarrow isMarried(mirka, roger), livesIn(mirka, bottmingen)$

Horn Rules

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

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

Logic program: Set of rules

Example: non-ground rule

% Married people live together
$livesIn(Y, Z) \leftarrow isMarried(X, Y), livesIn(X, Z)$

Rules wih Negation

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

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

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

Example: rule with negation

% Two married live together unless one is a researcher

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

Rules wih Negation

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

Informal semantics: If $b_1, \ldots, b_m$ are true and none of $b_{m+1}, \ldots, b_n$ 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"
$walk \leftarrow at(L), crossing(L), not\ \ train\_approaches(L)$

% At a rail road crossing cross the road if **no train** approaches
$walk \leftarrow at(L), crossing(L), \neg train\_approaches(L)$

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 [Muggleton, 1991]

Given:

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

Learning from Examples

Inductive Learning from Examples [Muggleton, 1991]

Given:

- $E^+ = \{fatherOf(john, mary), fatherOf(david, steve)\}$
- $E^- = \{fatherOf(kathy, ellen), fatherOf(john, steve)\}$
- $T = \{parentOf(john, mary), male(john),$
  $parentOf(david, steeve), male(david),$
  $parentOf(kathy, ellen), female(kathy)\}$
- 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 [Raedt and Dzeroski, 1994]

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 [Raedt and Dzeroski, 1994]

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
- Genaralization 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
- Genaralization 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
- Genaralization as θ-subsumption
  - Atoms: $a \succeq b$ iff a substitution θ exists such that $a\theta = b$
    $person(X) \succeq person(roger), \;\; \theta = \{X/roger\}$
  - 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
- Genaralization as θ-subsumption
  - Atoms: $a \succeq b$ iff a substitution θ exists such that $a\theta = b$
    $person(X) \succeq person(roger), \;\; \theta = \{X/roger\}$
  - Clause: $C \succeq D$ iff θ exists, s.t. $C\theta \subseteq D$
    $\{worksAt(X, Y)\} \succeq \{worksAt(Z, bosch), researcher(Z)\},$

Common Techniques in ILP

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

Common Techniques in ILP

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

Common Techniques in ILP

- Generality ($\succeq$): essential component of symbolic learning systems
- Genaralization as θ-subsumption
  - Atoms: $a \succeq b$ iff a substitution θ exists such that $a\theta = b$
    $person(X) \succeq person(roger), \ \ \theta = \{X/roger\}$
  - Clause: $C \succeq D$ iff θ exists, s.t. $C\theta \subseteq D$
    $\{worksAt(X,Y)\} \succeq \{worksAt(Z,bosch), researcher(Z)\}$,
    $\theta = \{X/Z, Y/bosch\}$
- Generalization as entailment
  - Logic program: $Hyp1 \succeq Hyp2$ iff $Hyp1 \models Hyp2$
    $\underbrace{person(X) \leftarrow researcher(X)}_{Hyp1} \ \underbrace{person(mat) \leftarrow researcher(mat)}_{Hyp_2}$

Common Techniques in ILP

- Generality ($\succeq$): essential component of symbolic learning systems
- Genaralization as θ-subsumption
  - Atoms: $a \succeq b$ iff a substitution θ exists such that $a\theta = b$
    $person(X) \succeq person(roger), \;\; \theta = \{X/roger\}$
  - Clause: $C \succeq D$ iff θ exists, s.t. $C\theta \subseteq D$
    $\{worksAt(X, Y)\} \succeq \{worksAt(Z, bosch), researcher(Z)\},$
    $\theta = \{X/Z, Y/bosch\}$
- Generalization as entailment
  - Logic program: $Hyp1 \succeq Hyp2$ iff $Hyp1 \models Hyp2$
    $\underbrace{person(X) \leftarrow researcher(X)}_{Hyp1} \; \underbrace{person(mat) \leftarrow researcher(mat)}_{Hyp_2}$

$$Hyp1 \succeq Hyp2$$

Common Techniques in ILP

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

Common Techniques in ILP

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

$$Hyp1 \succeq Hyp2$$

Common Techniques in ILP

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

Common Techniques in ILP

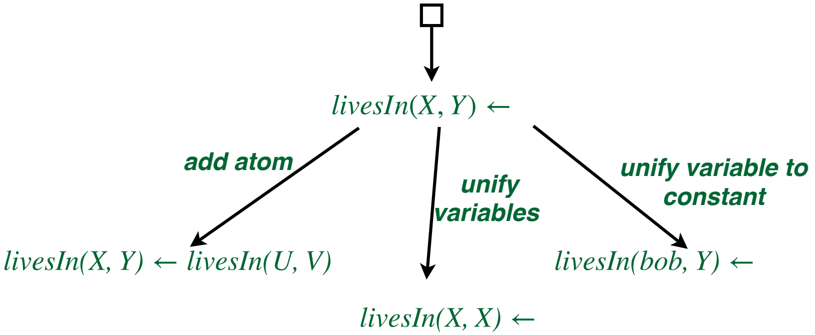
- Generality ($\succeq$): essential component of symbolic learning systems
- Genaralization as θ-subsumption
  - Atoms: $a \succeq b$ iff a substitution θ exists such that $a\theta = b$
    $person(X) \succeq person(roger), \quad \theta = \{X/roger\}$
  - Clause: $C \succeq D$ iff θ exists, s.t. $C\theta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

Common Techniques in ILP

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

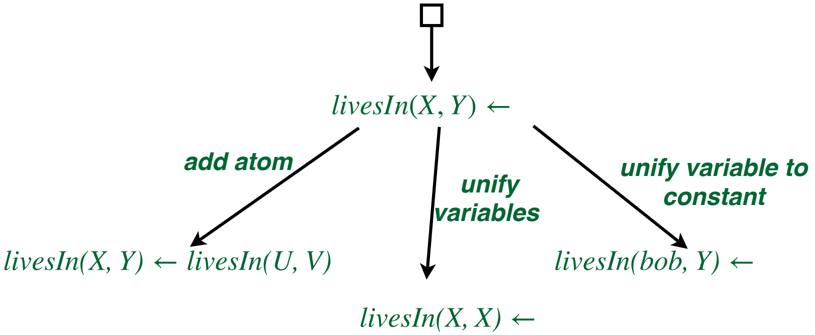
Common Techniques in ILP

- Clause refinement [Shapiro, 1991]: 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 [Shapiro, 1991]: 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 [Muggleton, 1995]: 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 [Law *et al.*, 2015]
- type of the output knowledge, e.g., rules, DL ontologies [Lehmann, 2009]
- the way the data is given as input, e.g., all at once, incrementally [Katzouris *et al.*, 2015]
- availability of an oracle, e.g., human in the loop
- quality of the data source, e.g., noisy [Evans and Grefenstette, 2018]
- data (in)completeness, e.g., OWA vs CWA...
- background knowledge, e.g., DL ontology [d'Amato *et al.*, 2016], hybrid theories [Lisi, 2010]

Challenges of Rule Induction from KGs

Open World Assumption: negative facts cannot be easily derived

Challenges of Rule Induction from KGs

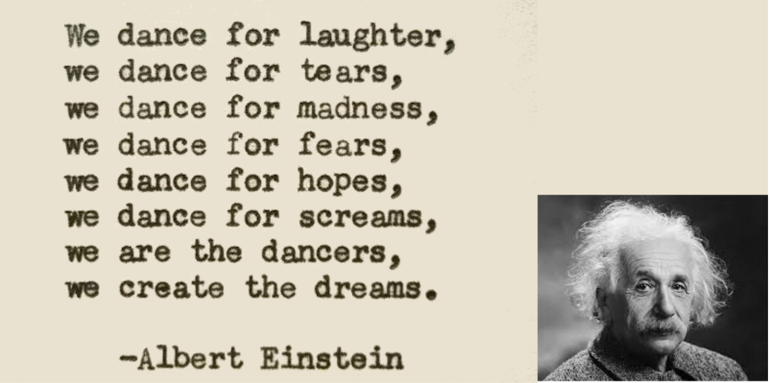
Open World Assumption: negative facts cannot be easily derived

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

Challenges of Rule Induction from KGs

Open World Assumption: negative facts cannot be easily derived

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



Challenges of Rule Induction from KGs

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

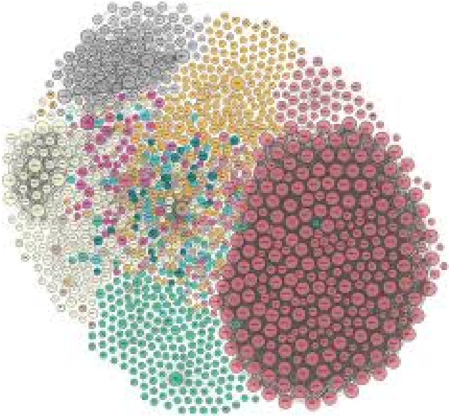
"Man bites dog phenomenon"[1]

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

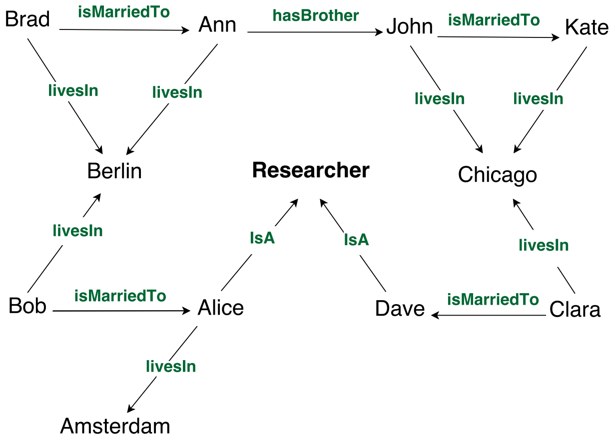
Challenges of Rule Induction from KGs

Huge size: Modern KGs contain billions of facts

E.g., Google KG stores 70 billion facts

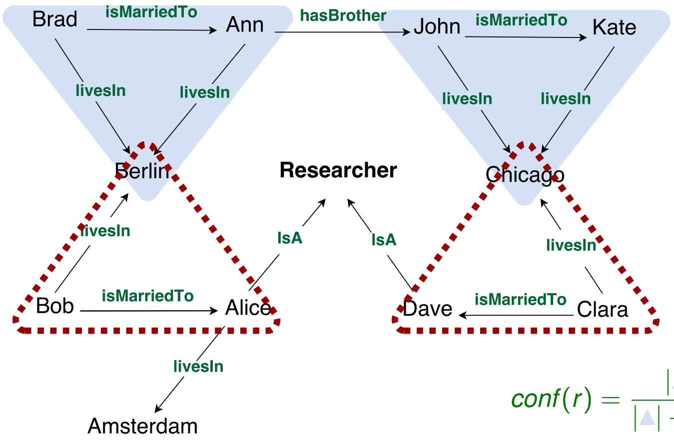


Rule Induction from KGs



Rule Induction from KGs

Confidence, e.g., WARMER [Goethals and den Bussche, 2002]
CWA: whatever is missing is false



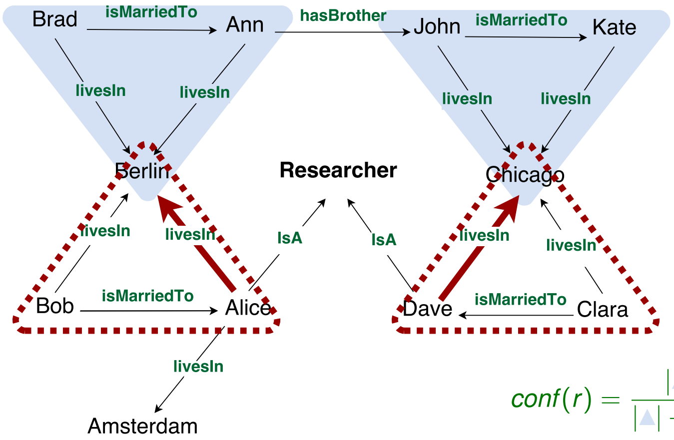
$conf(r) = \frac{|\blacktriangle|}{|\blacktriangle| + |\triangle|} = \frac{2}{4}$

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

Rule Induction from KGs

Confidence, e.g., WARMER [Goethals and den Bussche, 2002]

CWA: whatever is missing is false



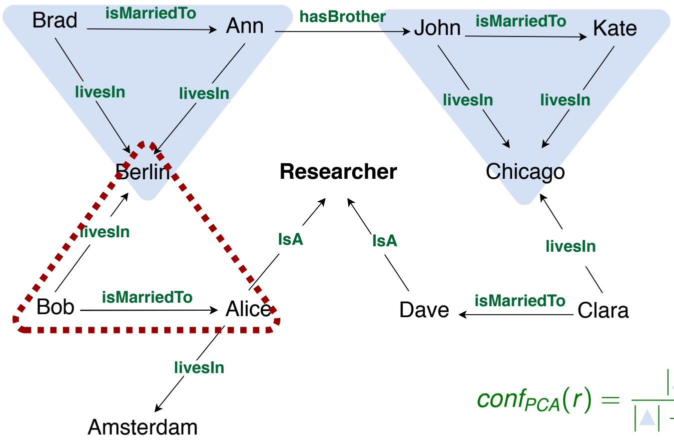
$$conf(r) = \frac{|\blacktriangle|}{|\blacktriangle| + |\triangle|} = \frac{2}{4}$$

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

Rule Induction from KGs

PCA confidence AMIE [Galarraga *et al.*, 2015]

PCA: Since Alice has a living place already, all others are incorrect.

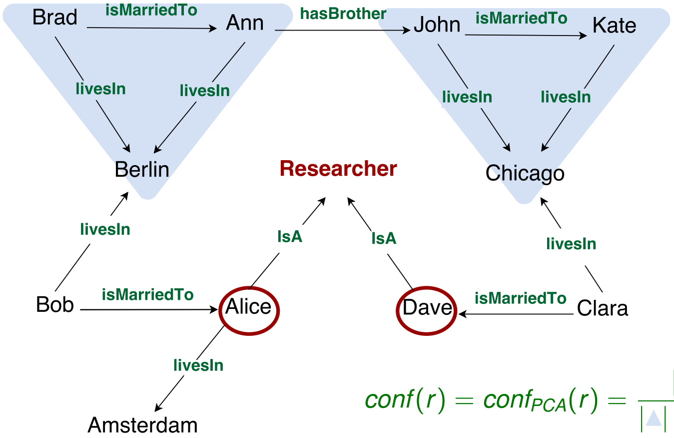


$$conf_{PCA}(r) = \frac{|\blacktriangle|}{|\blacktriangle| + |\triangle|} = \frac{2}{3}$$

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

Rule Induction from KGs

Exception-enriched rules: **OWA** is a challenge!

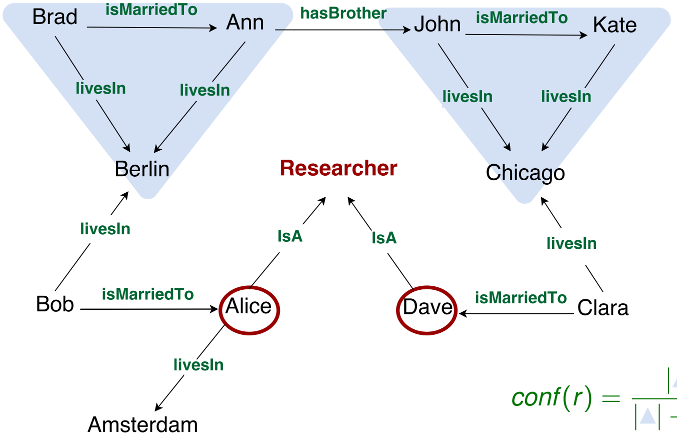


$$conf(r) = conf_{PCA}(r) = \frac{|\blacktriangle|}{|\blacktriangle| + |\triangle|} = 1$$

$$r: \ livesIn(X, Y) \leftarrow isMarriedTo(Z, X), livesIn(Z, Y), not \ isA(X, researcher)$$

Rule Induction from KGs

Exception-enriched rules: **OWA** is a challenge!



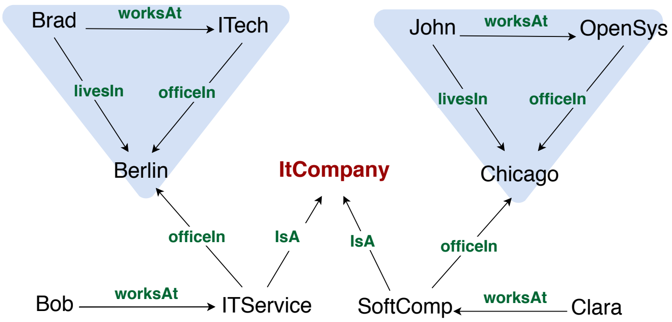
$$conf(r) = \frac{|\blacktriangle|}{|\blacktriangle| + |\triangle|} = \frac{2}{4}$$

$$r: \ livesIn(X, Y) \leftarrow isMarriedTo(Z, X), livesIn(Z, Y), not\ isA(X, researcher)$$

M. Gad-Elrab, D. Stepanova, J. Urbani, G. Weikum. Exception-enriched Rule Learning from KGs. *ISWC2016*
H. Dang Tran, D. Stepanova, M. Gad-Elrab, F. A. Lisi, G. Weikum. Towards Nonmonotonic Rule Learning from KGs. *ILP2016*

Absurd Rules due to Data Incompleteness

Problem: rules learned from highly incomplete KGs might be absurd..

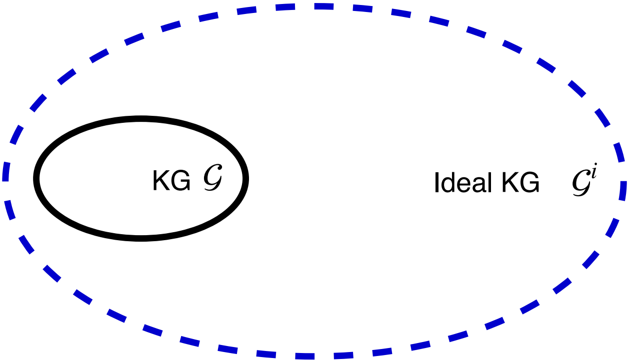


$$conf(r) = conf_{PCA}(r) = 1$$

$$livesIn(X, Y) \leftarrow worksAt(X, Z), officeIn(Z, Y), not\ isA(Z, itCompany)$$

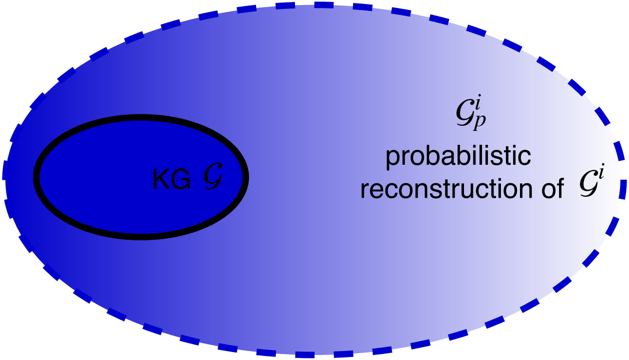
Ideal KG

$\mu(r, \mathcal{G}^i)$: 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 $\mathcal{G}_p^i$



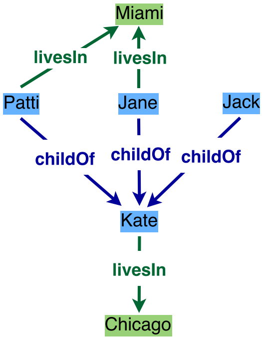
Hybrid Rule Measure

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

- **$\lambda \in [0..1]$: 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 $\mathcal{G}^i$)

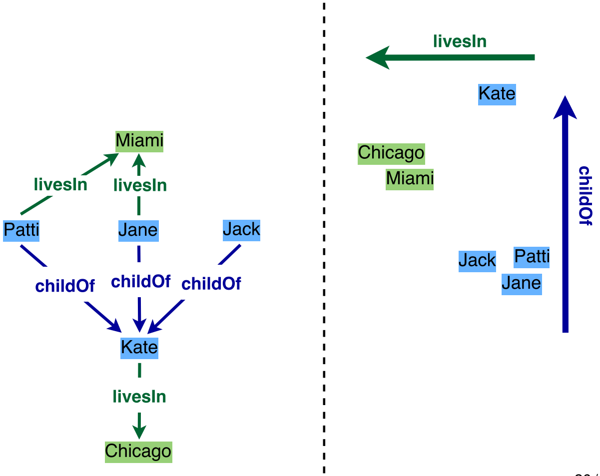
KG Embeddings

- **Intuition:** For $\langle s, p, o \rangle$ in KG, find $\mathbf{s}, \mathbf{p}, \mathbf{o}$ such that $\mathbf{s} + \mathbf{p} \approx \mathbf{o}$
- 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



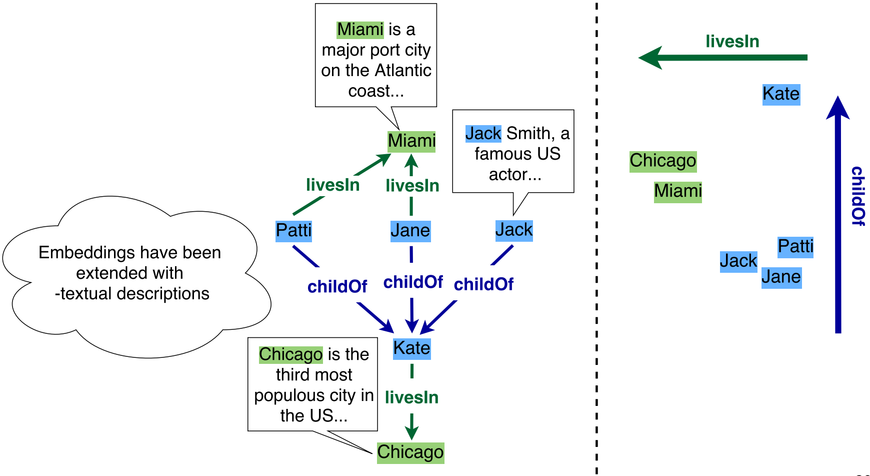
KG Embeddings

- **Intuition:** For $\langle s, p, o \rangle$ in KG, find $\mathbf{s}, \mathbf{p}, \mathbf{o}$ such that $\mathbf{s} + \mathbf{p} \approx \mathbf{o}$
- 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



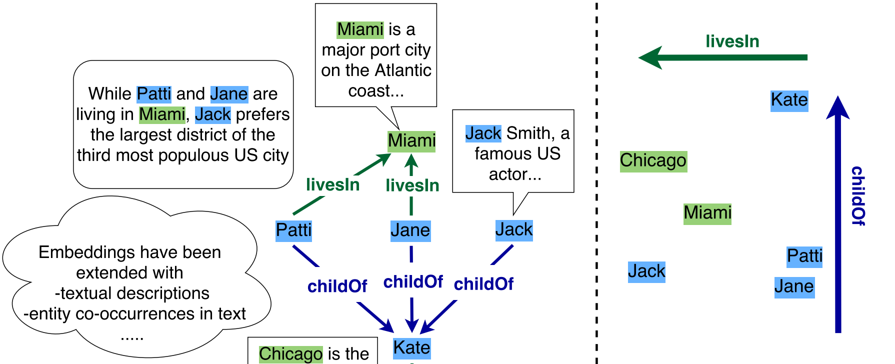
KG Embeddings

- **Intuition:** For $\langle s, p, o \rangle$ in KG, find $\mathbf{s}, \mathbf{p}, \mathbf{o}$ such that $\mathbf{s} + \mathbf{p} \approx \mathbf{o}$
- 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



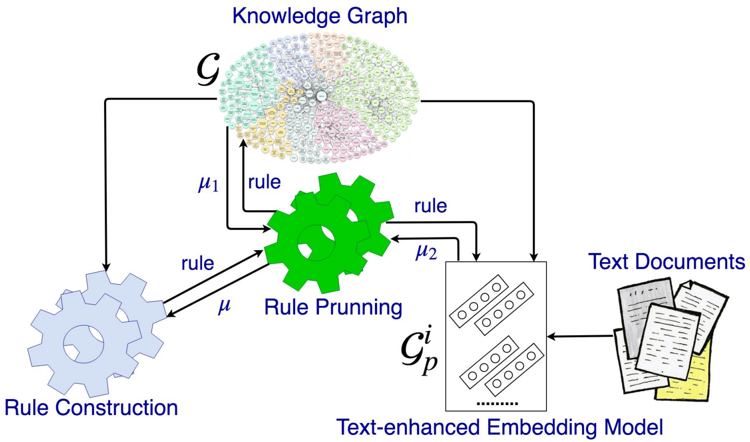
KG Embeddings

- **Intuition:** For $\langle s, p, o \rangle$ in KG, find $\mathbf{s}, \mathbf{p}, \mathbf{o}$ such that $\mathbf{s} + \mathbf{p} \approx \mathbf{o}$
- 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



rank(<Jack livesIn Chicago>) = 0.8
rank(<Jack livesIn Miami>) = 0.6
...
rank(<Jack livesIn NewYork>) = 0.2

Embedding-based Rule Learning



V. Thinh Ho, D. Stepanova, M. Gad-Elrab, E. Kharlamov, G. Weikum. Rule Learning from KGs Guided by Embeddings. *ISWC2018*

Rule Construction

- Clause exploration from general to specific
  - Our work: closed and safe rules with negation
    $livesIn(X,Y) \leftarrow marriedTo(X,Z), livesIn(Z,Y), not\ isA(X, researcher)$

$\square$

$livesIn(X, Y) \leftarrow$

add dangling atom

add instantiated atom

$livesIn(X, Y) \leftarrow marriedTo(X, Z)$

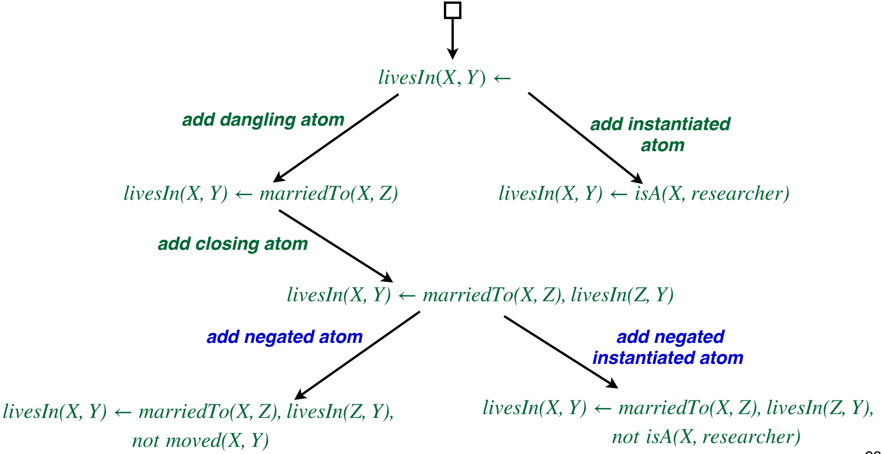
$livesIn(X, Y) \leftarrow isA(X, researcher)$

add closing atom

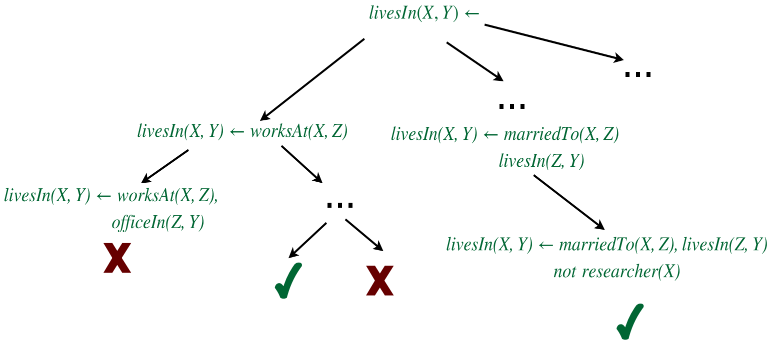
$livesIn(X, Y) \leftarrow marriedTo(X, Z), livesIn(Z, Y)$

Rule Construction

- Clause exploration from general to specific
  - Our work: closed and safe rules with negation
    $livesIn(X, Y) \leftarrow marriedTo(X, Z), livesIn(Z, Y), not\ isA(X, researcher)$



Rule Prunning



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

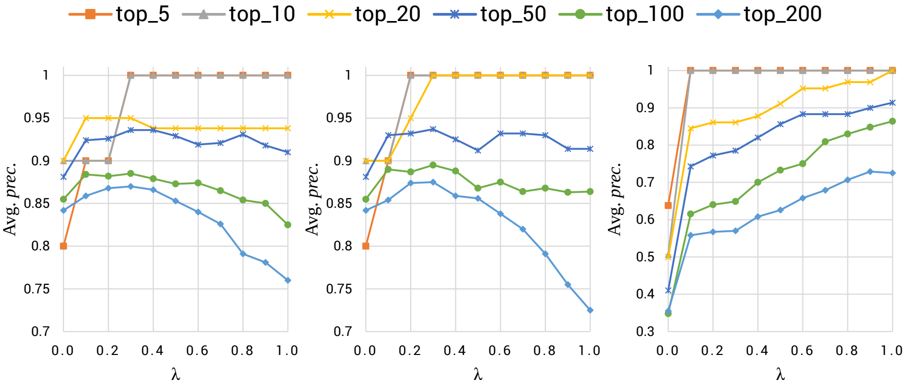
- Embedding models $\mathcal{G}_p^i$:
  - TransE [Bordes *et al.*, 2013], HolE [Nickel *et al.*, 2016]
  - With text: SSP [Xiao *et al.*, 2017]

- 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

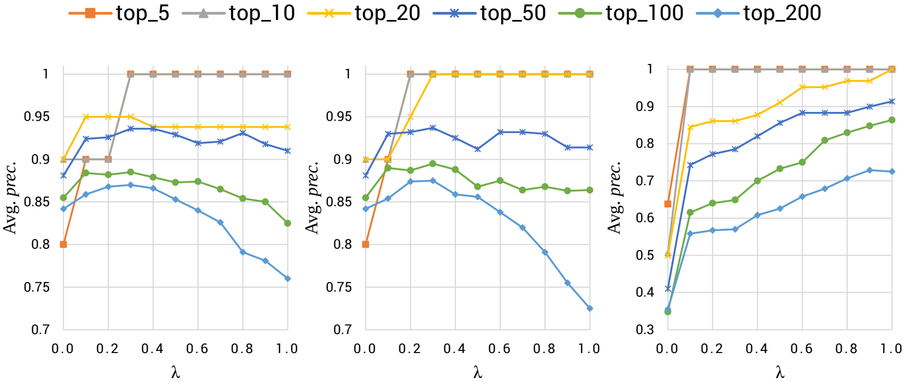
Evaluation of Hybrid Rule Measure



(a) Conf-HolE (b) Conf-SSP (c) PCA-SSP

Precision of *top-k* rules ranked using variations of μ on FB15K.

Evaluation of Hybrid Rule Measure



(a) Conf-HolE (b) Conf-SSP (c) PCA-SSP

Precision of *top-k* rules ranked using variations of μ on FB15K.

- Positive impact of embeddings in all cases for $\lambda = 0.3$
- **Note:** in (c) comparison to AMIE [Galarraga *et al.*, 2015] ($\lambda = 0$)

Example Rules Learned by Our Methods

Rules learned from Wikidata and IMDB

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

r_1 : $nobleFamily(X, Y) \leftarrow spouse(X, Z), nobleFamily(Z, Y), not\ isA(Y, chineseDynasty)$

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

r_2 : $writtenBy(X, Z) \leftarrow hasPredecessor(X, Y), writtenBy(Y, Z), not\ american\_film(X)$

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

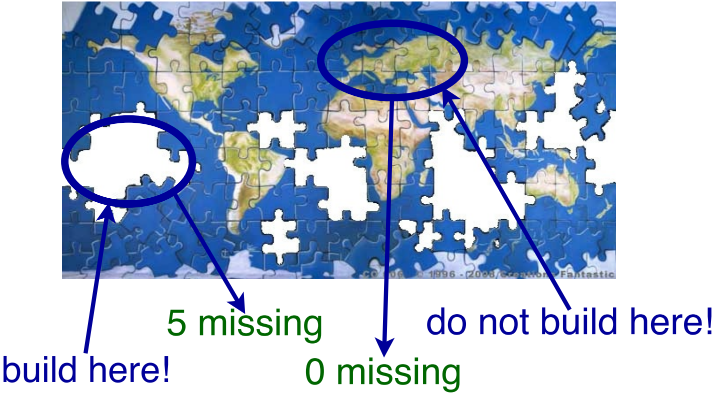
r_3 : $actedIn(X, Z) \leftarrow isMarriedTo(X, Y), directed(Y, Z), not\ silent\_film\_actor(X)$

Meta-data about Missing Facts in the KG

- Mining cardinality assertions from the Web [Mirza *et al.*, 2016]
  - *"... Albert Einstein had **3** children ..."*
- Estimating recall of KGs by crowd sourcing [Razniewski *et al.*, 2016]
  - *20 % of Nobel laureates in physics are missing*
- Predicting completeness in KGs [Galárraga *et al.*, 2017]
  - $complete(X, hasChild) \leftarrow child(X)$

Exploiting Meta-data in Rule Learning

Goal: make use of topological constraints on edge counts in the KG to improve rule learning.



T. Tanon, D. Stepanova, S. Razniewski, P. Mirza, G. Weikum. Completeness-aware Rule Learning from KGs. *ISWC2017*

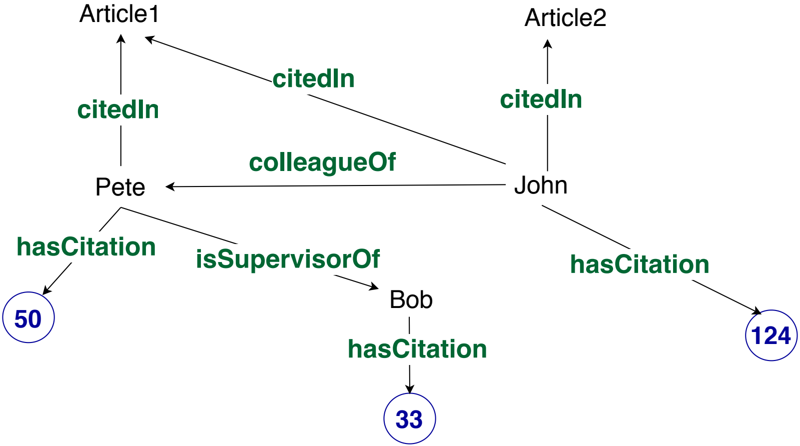
Motivation

Rule Induction under Incompleteness

Numerical Rule Learning

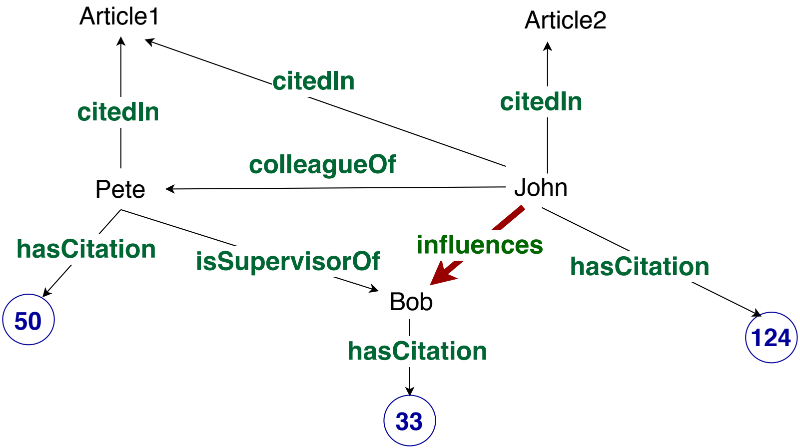
Rule-based Fact Checking

Numerical Rules



$$influences(X, Y) \leftarrow colleague(X, Z), supervisorOf(Z, Y),$$
$$X.hasCitation > Z.hasCitation$$

Numerical Rules

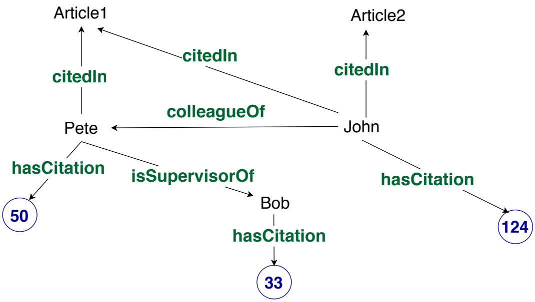


$$influences(X, Y) \leftarrow colleague(X, Z), supervisorOf(Z, Y), \\ X.hasCitation > Z.hasCitation$$

Rule Learning via Boolean Matrix Multiplication

NeuralLP [Yang *et al.*, 2017]: Differentiable rule learning

$$M_{\text{citedIn}} = \begin{array}{c} \\ \\ \end{array}\begin{array}{cc} \begin{array}{ccccc} \text{john} & \text{pete} & \text{bob} & \text{article1} & \text{article2} \end{array} & \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} & \begin{array}{l} \text{john} \\ \text{pete} \\ \text{bob} \\ \text{article1} \\ \text{article2} \end{array} \end{array}$$



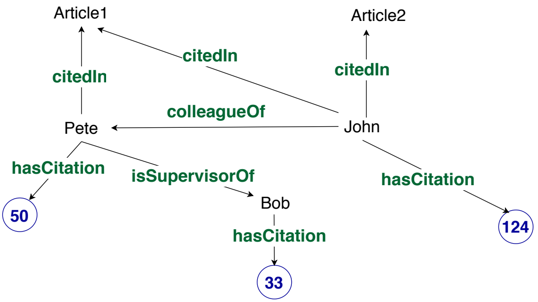
Rule Learning via Boolean Matrix Multiplication

NeuralLP [Yang *et al.*, 2017]: Differentiable rule learning

$$M_{\text{citedIn}} = \begin{array}{c} \\ \left[\begin{array}{ccccc} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{array}\right] \end{array} \quad \begin{array}{l} \text{john} \\ \text{pete} \\ \text{bob} \\ \text{article1} \\ \text{article2} \end{array}$$

(columns: john, pete, bob, article1, article2)

$$v_{\text{john}} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \begin{array}{l} \text{john} \\ \text{pete} \\ \text{bob} \\ \text{article1} \\ \text{article2} \end{array}$$



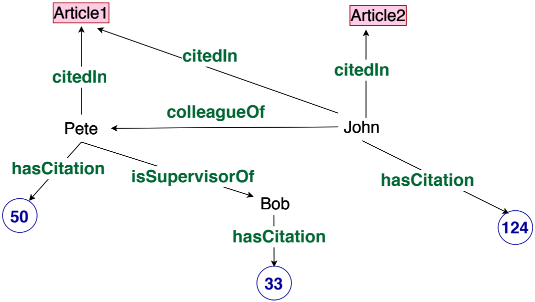
Rule Learning via Boolean Matrix Multiplication

NeuralLP [Yang *et al.*, 2017]: Differentiable rule learning

$$M_{\text{citedIn}} = \begin{array}{c} \begin{array}{ccccc} \text{john} & \text{pete} & \text{bob} & \text{article1} & \text{article2} \end{array} \\ \left[\begin{array}{ccccc} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{array}\right] \end{array} \begin{array}{l} \\ \text{john} \\ \text{pete} \\ \text{bob} \\ \text{article1} \\ \text{article2} \end{array}$$

$$v_{\text{john}} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \begin{array}{l} \text{john} \\ \text{pete} \\ \text{bob} \\ \text{article1} \\ \text{article2} \end{array}$$

$$M_{\text{citedIn}} v_{\text{john}} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$



Problem: Implementing Numerical Rule Matching

Differentiable learning framework via (sparse) matrix-vector multiplication

Problem: Implementing Numerical Rule Matching

Differentiable learning framework via (sparse) matrix-vector multiplication

$$\textit{Adj matrix}\ (M_{colleagueOf})_{y,x} = \begin{cases} 1 & \text{if colleagueOf(} \mathbf{x}\text{, } \mathbf{y} \text{)} \\ 0 & \text{otherwise} \end{cases}$$

Problem: Implementing Numerical Rule Matching

Differentiable learning framework via (sparse) matrix-vector multiplication

$$\textit{Adj matrix}\ (M_{colleagueOf})_{y,x} = \begin{cases} 1 & \text{if colleagueOf(} \mathbf{x}, \mathbf{y} \text{)} \\ 0 & \text{otherwise} \end{cases}$$

Apply rules (*path counting*) by sparse matrix-vector multiplication

$$\text{influences(} \mathbf{X}, \mathbf{Z} \text{)} \quad \leftarrow \quad \text{colleagueOf(} \mathbf{X}, \mathbf{Y} \text{), supervisorOf(} \mathbf{Y}, \mathbf{Z} \text{)}$$

$$\text{influences(} \mathbf{john}, \mathbf{Z} \text{)} = \text{one\_hot(} \mathbf{john} \text{)} \quad M^T_{colleagueOf} \quad M^T_{supervisorOf}$$

Problem: Implementing Numerical Rule Matching

Differentiable learning framework via (sparse) matrix-vector multiplication

$$\textit{Adj matrix } (M_{colleagueOf})_{y,x} = \begin{cases} 1 & \text{if colleagueOf(} \mathbf{x}, \mathbf{y} \text{)} \\ 0 & \text{otherwise} \end{cases}$$

Apply rules (*path counting*) by sparse matrix-vector multiplication

$$\text{influences(} \mathbf{X}, \mathbf{Z} \text{)} \quad \leftarrow \quad \text{colleagueOf(} \mathbf{X}, \mathbf{Y} \text{), supervisorOf(} \mathbf{Y}, \mathbf{Z} \text{)}$$

$$\text{influences(} \mathbf{john}, \mathbf{Z} \text{)} = \text{one\_hot(} \mathbf{john} \text{)} \quad M^T_{colleagueOf} \quad M^T_{supervisorOf}$$

For numerical rules, we can similarly create the comparison matrix

$$\textit{Adj matrix } (M_{cmp})_{y,x} = \begin{cases} 1 & \text{if } \mathbf{x}\text{.numCitation} < \mathbf{y}\text{.numCitation} \\ 0 & \text{otherwise} \end{cases}$$

Problem: Implementing Numerical Rule Matching

Differentiable learning framework via (sparse) matrix-vector multiplication

$$\textit{Adj matrix}\ (M_{colleagueOf})_{y,x} = \begin{cases} 1 & \text{if colleagueOf(} \mathbf{x}, \mathbf{y} \text{)} \\ 0 & \text{otherwise} \end{cases}$$

Apply rules (*path counting*) by sparse matrix-vector multiplication

$$\text{influences(} \mathbf{X}, \mathbf{Z} \text{)} \quad \leftarrow \quad \text{colleagueOf(} \mathbf{X}, \mathbf{Y} \text{), supervisorOf(} \mathbf{Y}, \mathbf{Z} \text{)}$$

$$\text{influences(} \mathbf{john}, \mathbf{Z} \text{)} = \text{one\_hot(} \mathbf{john} \text{)} \quad M^T_{colleagueOf} \quad M^T_{supervisorOf}$$

For numerical rules, we can similarly create the comparison matrix

$$\textit{Adj matrix}\ (M_{cmp})_{y,x} = \begin{cases} 1 & \text{if } \mathbf{x}\text{.numCitation} < \mathbf{y}\text{.numCitation} \\ 0 & \text{otherwise} \end{cases}$$

Problem: may be a dense matrix $\Rightarrow$ cannot be materialized on GPU

Efficient Matrix Vector Multiplication for Numerical Operators

Trick: assume values are sorted by the permutation matrices P_p and P_q, resp.

$$\tilde{M}_{r_{pq}^{\leq}} = \begin{array}{c} \begin{array}{ccccccc} \texttt{NaN} \ldots \texttt{NaN} & \tilde{g}_1 & \leq & \ldots & \leq & \tilde{g}_n & \end{array} \\ \left[\begin{array}{ccccccc} 0 & \cdots & 0 & \cdots & & & 0 \\ \vdots & & \vdots & & & & \vdots \\ & & 0 & \cdots & & & 0 \\ & & \vdots & 1 & \cdots & & 1 \\ & & & 0 & 1 & \cdots & \\ & & & \vdots & 0 & 1 & \cdots \\ & & & & 0 & 1 & \cdots \\ 0 & \cdots & 0 & & \cdots & 0 & 1 \quad 1 \end{array}\right] \end{array} \begin{array}{c} \\ \texttt{NaN} \\ \vdots \\ \texttt{NaN} \\ \tilde{f}_1 \\ \leq \\ \vdots \\ \leq \\ \tilde{f}_m \end{array}$$

Monotonic borderline:

γ_i: position of the first non-zero element in the i^{th} row

$$(\tilde{M}_{r_{pq}^{\leq}} v)_i = \sum_{\gamma_i \leq j \leq |\mathcal{C}|} v_j = \texttt{cumsum}(v)_{\gamma_i}$$

$$Mv = P_q^{\top} \texttt{cumsum}(P_p v)_{\gamma}$$

Complexity: $O(n^2) \Rightarrow O(n \log n)$

Po-Wei Wang, Daria Stepanova, Csaba Domokos, Zico Kolter. Differentiable Learning of Numerical Rules from KGs. *ICLR 2020*

Evaluation of Numerical Rule Learning

`Hit@10`: number of correct head atoms predicted out of the top 10 predictions

Dataset	Synthetic1	Synthetic2	FB15K-237-num	DBP15K-num
AnyBurl	0.031	0.685	**0.426**	0.522
`NeuralLP`	0.240	0.295	0.362	0.436
ours	**1.000**	**1.000**	0.415	**0.682**

Evaluation of Numerical Rule Learning

`Hit@10`: number of correct head atoms predicted out of the top 10 predictions

Dataset	Synthetic1	Synthetic2	FB15K-237-num	DBP15K-num
AnyBurl	0.031	0.685	**0.426**	0.522
`NeuralLP`	0.240	0.295	0.362	0.436
ours	**1.000**	**1.000**	0.415	**0.682**

Rules learned from Freebase and DBPedia:

Some symptoms provoke risk factors inherited from diseases with these symptoms

$$symptomHasRiskFactors(X, Y) \leftarrow f(X), symtompOfDisease(X, Z), diseaseHasRiskFactors(Z, Y)$$

Minister of defense with certain properties is the general of military of the given country

$$general(X, Y) \leftarrow ministerOfDefense(X, Z), f(Z), militaryBranchOfCountry(Z, Y)$$

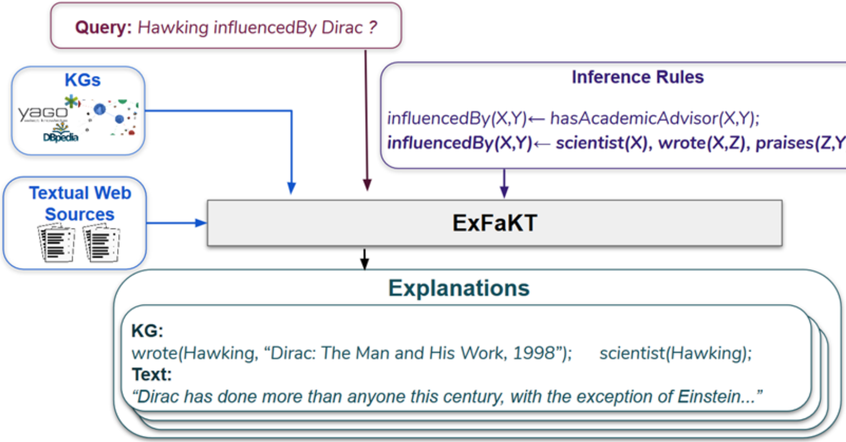
Motivation

Rule Induction under Incompleteness

Numerical Rule Learning

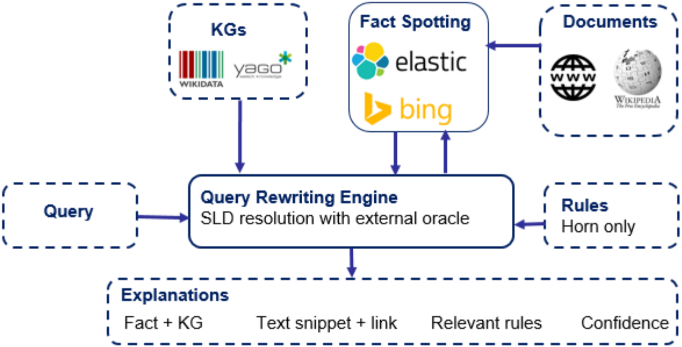
Rule-based Fact Checking

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*.

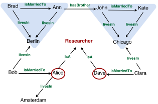
Rule-based Fact Checking



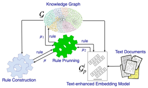
M. Gad-Elrab, D. Stepanova, J. Urbani, G. Weikum. ExFakt: A Framework for Explaining Facts over KGs and Text. *WSDM 2019*.
M. Gad-Elrab, D. Stepanova, J. Urbani, G. Weikum. Tracy: Tracing Facts over Knowledge Graphs and Text. *WWW 2019*.

Summary

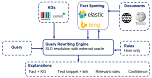
- Horn rule learning



- Exploiting embeddings to guide rule learning

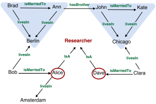


- Numerical rule learning
- Rule-based fact checking

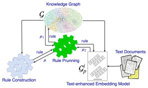


Summary

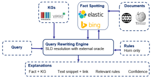
- Horn rule learning



- Exploiting embeddings to guide rule learning



- Numerical rule learning
- Rule-based fact checking



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

BOSCH
Invented for life

jobs.smartrecruiters.com/BoschGroup/743999698936956-phd-combined-reasoning-and-learning-approaches

References I

Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko.
Translating Embeddings for Modeling Multi-relational Data.
In *Proceedings of NIPS*, pages 2787–2795, 2013.

Claudia d'Amato, Steffen Staab, Andrea GB Tettamanzi, Tran Duc Minh, and Fabien Gandon.
Ontology enrichment by discovering multi-relational association rules from ontological knowledge bases.
In *SAC*, pages 333–338, 2016.

Richard Evans and Edward Grefenstette.
Learning explanatory rules from noisy data.
J. Artif. Intell. Res., 61:1–64, 2018.

Luis Galarraga, Christina Teflioudi, Katja Hose, and Fabian M. Suchanek.
Fast rule mining in ontological knowledge bases with AMIE+.
In *VLDB*, volume 24, pages 707–730, 2015.

Luis Galárraga, Simon Razniewski, Antoine Amarilli, and Fabian M Suchanek.
Predicting completeness in knowledge bases.
WSDM, 2017.

Bart Goethals and Jan Van den Bussche.
Relational association rules: Getting warmer.
In *PDD*, 2002.

Nikos Katzouris, Alexander Artikis, and Georgios Paliouras.
Incremental learning of event definitions with inductive logic programming.
Machine Learning, 100(2-3):555–585, 2015.

Mark Law, Alessandra Russo, and Krysia Broda.
The ILASP system for learning answer set programs.
https://www.doc.ic.ac.uk/~ml1909/ILASP, 2015.

References II

Jens Lehmann.
DL-Learner: Learning concepts in description logics.
Journal of Machine Learning Research, pages 2639–2642, 2009.

Francesca A. Lisi.
Inductive Logic Programming in Databases: From Datalog to DL+log.
TPLP, 10(3):331–359, 2010.

Paramita Mirza, Simon Razniewski, and Werner Nutt.
Expanding wikidata's parenthood information by 178%, or how to mine relation cardinality information.
In *ISWC 2016 Posters & Demos*, 2016.

Stephen Muggleton.
Inductive logic programming.
New Generation Comput., 8(4):295–318, 1991.

Stephen Muggleton.
Inverse entailment and progol.
New Generation Comput., 13(3&4):245–286, 1995.

Maximilian Nickel, Lorenzo Rosasco, and Tomaso A. Poggio.
Holographic embeddings of knowledge graphs.
In *AAAI*, 2016.

Luc De Raedt and Saso Dzeroski.
First-order jk-clausal theories are pac-learnable.
Artif. Intell., 70(1-2):375–392, 1994.

Simon Razniewski, Fabian M. Suchanek, and Werner Nutt.
But what do we actually know?
In *Proceedings of the 5th Workshop on Automated Knowledge Base Construction, AKBC@NAACL-HLT 2016, San Diego, CA, USA, June 17, 2016*, pages 40–44, 2016.

References III

Ehud Y. Shapiro.
Inductive inference of theories from facts.
In *Computational Logic - Essays in Honor of Alan Robinson*, pages 199–254, 1991.

Han Xiao, Minlie Huang, Lian Meng, and Xiaoyan Zhu.
SSP: semantic space projection for knowledge graph embedding with text descriptions.
In *AAAI*, 2017.

Fan Yang, Zhilin Yang, and William W. Cohen.
Differentiable learning of logical rules for knowledge base reasoning.
In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 2319–2328, 2017.