Rule Induction and Reasoning in Knowledge Graphs

Daria Stepanova

Bosch Center for Artificial Intelligence, Renningen, Germany

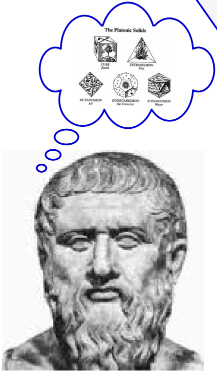
ODSC 2022

What is Knowledge?

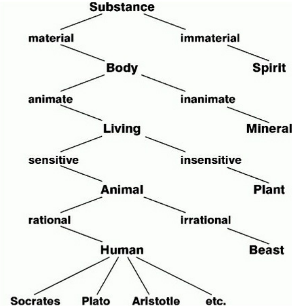
Plato: *“Knowledge is justified true belief”*

350 BC

Personal



External

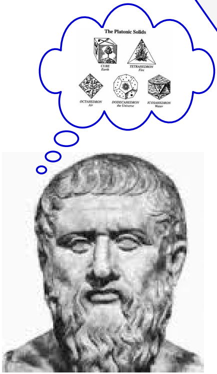


Tree of Knowledge

What is Knowledge?

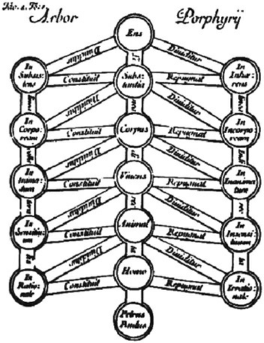
Plato: *"Knowledge is justified true belief"*

Personal



350 BC

External



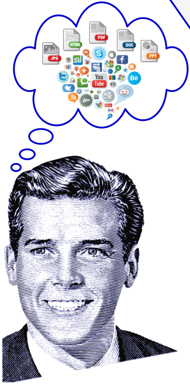
Knowledge Graphs as Digital Knowledge

“Digital knowledge is semantically enriched machine processable data”

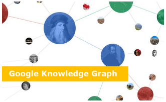
Personal

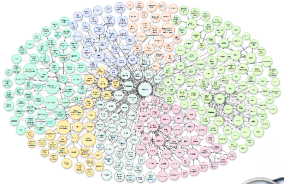
2021

External



yago
select knowledge





WIKIDATA

NELL

Knowledge Graph

Semantic Web Search

Google

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

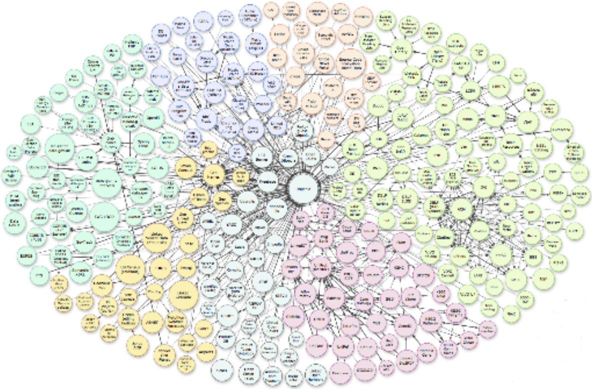
Weight: 85 kg

Spouse: Mirka Federer (m. 2009)

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

Semantic Web Search

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



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

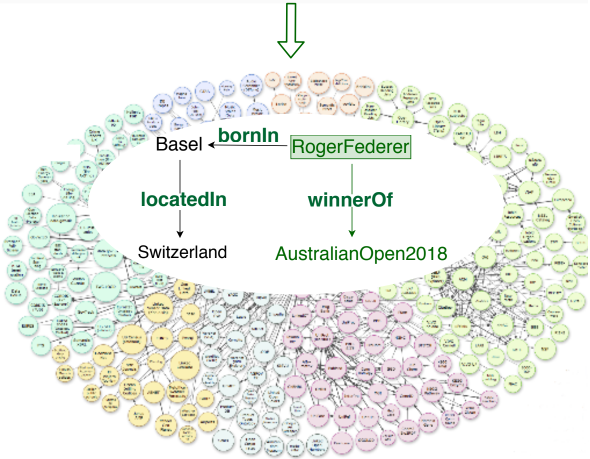
Weight: 85 kg

Spouse: Mirka Federer (m. 2009)

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

Semantic Web Search

Google $\exists X\ \mathit{winnerOf}(X, \mathit{AustralianOpen}2018)$



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

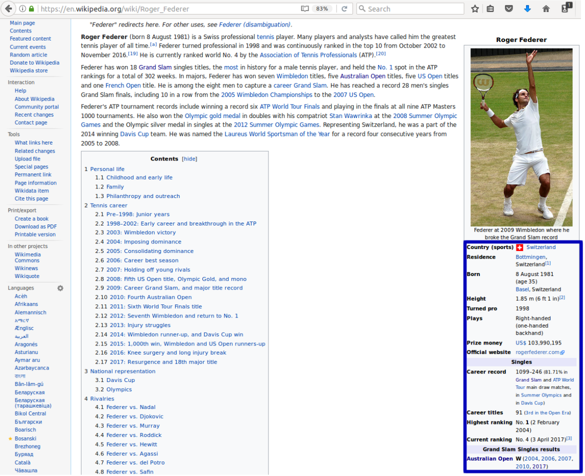
Height: 1.85 m

Weight: 85 kg

Spouse: Mirka Federer (m. 2009)

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

Knowledge Graphs



Industrial Knowledge Graphs

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)

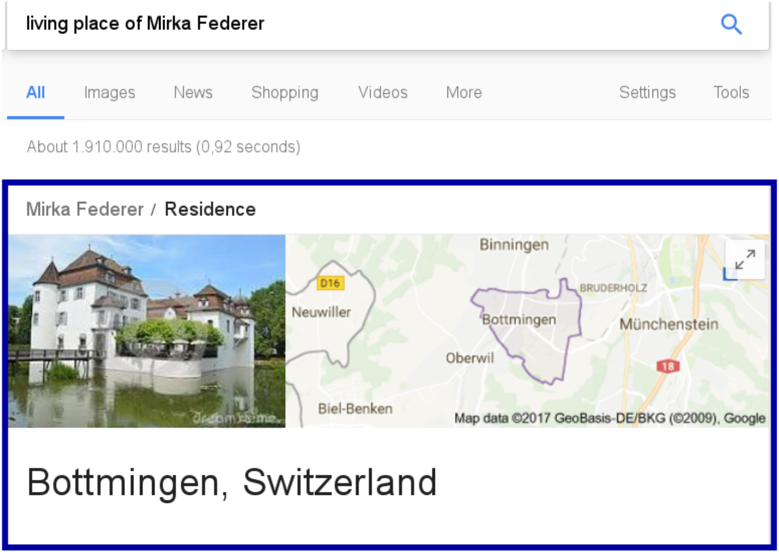
Roger Federer / Wife

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

$$\mathit{livesIn}(Y, Z) \leftarrow \mathit{marriedTo}(X, Y), \\ \mathit{livesIn}(X, Z)$$

Married people live together

$\mathit{marriedTo}(\mathit{mirka}, \mathit{roger})$

Mirka is married to Roger

$\mathit{livesIn}(\mathit{mirka}, \mathit{bottmingen})$

Mirka lives in Bottmingen

Human Reasoning

$$\mathit{livesIn}(Y, Z) \leftarrow \mathit{marriedTo}(X, Y), \mathit{livesIn}(X, Z)$$

Married people live together

$$\mathit{marriedTo}(\mathit{mirka}, \mathit{roger})$$

Mirka is married to Roger

$$\mathit{livesIn}(\mathit{mirka}, \mathit{bottmingen})$$

Mirka lives in Bottmingen

$$\mathit{livesIn}(\mathit{roger}, \mathit{bottmingen})$$

Roger 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 →

But where can a machine get such rules from?

Applications of Rule Learning

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

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.

Default reasoning: Facts not known to be true are assumed to be false

Example: rule with negation

% Two married live together unless one is a researcher

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

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:

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

Common Techniques in ILP

- Generality ($\succeq$): essential component of symbolic learning systems

Common Techniques in ILP

- Generality ($\succeq$): essential component of symbolic learning systems
- Genaralization as θ-subsumption
  - $\mathit{person}(X) \succeq \mathit{person}(\mathit{roger}), \;\; \theta = \{X/\mathit{roger}\}$

Common Techniques in ILP

- Generality ($\succeq$): essential component of symbolic learning systems
- Genaralization as θ-subsumption
  - $\mathit{person}(X) \succeq \mathit{person}(\mathit{roger}),\ \ \theta = \{X/\mathit{roger}\}$
  - $\{\mathit{worksAt}(X, Y)\} \succeq \{\mathit{worksAt}(Z, \mathit{bosch}), \mathit{researcher}(Z)\},$

Common Techniques in ILP

- Generality ($\succeq$): essential component of symbolic learning systems
- Genaralization as θ-subsumption
  - $\mathit{person}(X) \succeq \mathit{person}(\mathit{roger}), \;\; \theta = \{X/\mathit{roger}\}$
  - $\{\mathit{worksAt}(X, Y)\} \succeq \{\mathit{worksAt}(Z, \mathit{bosch}), \mathit{researcher}(Z)\}$,
    $\theta = \{X/Z, Y/\mathit{bosch}\}$

Common Techniques in ILP

- Generality ($\succeq$): essential component of symbolic learning systems

- Genaralization as θ-subsumption
  - $person(X) \succeq person(roger), \;\; \theta = \{X/roger\}$
  - $\{worksAt(X, Y)\} \succeq \{worksAt(Z, bosch), researcher(Z)\}$,
    $\theta = \{X/Z, Y/bosch\}$

- Generalization as entailment
  - $\underbrace{person(X) \leftarrow researcher(X)}_{Hyp1} \; \underbrace{person(mat) \leftarrow researcher(mat)}_{Hyp_2}$

Common Techniques in ILP

- Generality ($\succeq$): essential component of symbolic learning systems
- Genaralization as θ-subsumption
  - $\mathit{person}(X) \succeq \mathit{person}(\mathit{roger}), \;\; \theta = \{X/\mathit{roger}\}$
  - $\{\mathit{worksAt}(X, Y)\} \succeq \{\mathit{worksAt}(Z, \mathit{bosch}), \mathit{researcher}(Z)\}$, $\theta = \{X/Z, Y/\mathit{bosch}\}$
- Generalization as entailment
  - $\underbrace{\mathit{person}(X) \leftarrow \mathit{researcher}(X)}_{\mathit{Hyp1}} \; \underbrace{\mathit{person}(\mathit{mat}) \leftarrow \mathit{researcher}(\mathit{mat})}_{\mathit{Hyp}_2}$

$$\mathit{Hyp1} \succeq \mathit{Hyp2}$$

Common Techniques in ILP

- Generality ($\succeq$): essential component of symbolic learning systems
- Genaralization as θ-subsumption
  - $person(X) \succeq person(roger), \;\; \theta = \{X/roger\}$
  - $\{worksAt(X,Y)\} \succeq \{worksAt(Z,bosch), researcher(Z)\}$,
    $\theta = \{X/Z, Y/bosch\}$
- Generalization as entailment
  - $\underbrace{person(X) \leftarrow researcher(X)}_{Hyp1} \; \underbrace{person(X) \leftarrow researcher(X), alive(X)}_{Hyp2}$

Common Techniques in ILP

- Generality ($\succeq$): essential component of symbolic learning systems

- Genaralization as θ-subsumption
  - $person(X) \succeq person(roger), \;\; \theta = \{X/roger\}$
  - $\{worksAt(X, Y)\} \succeq \{worksAt(Z, bosch), researcher(Z)\},$
    $\theta = \{X/Z, Y/bosch\}$

- Generalization as entailment
  - $\underbrace{person(X) \leftarrow researcher(X)}_{Hyp1} \; \underbrace{person(X) \leftarrow researcher(X), alive(X)}_{Hyp2}$

$$Hyp1 \succeq Hyp2$$

Common Techniques in ILP

- Generality ($\succeq$): essential component of symbolic learning systems

- Genaralization as θ-subsumption
  - $person(X) \succeq person(roger), \;\; \theta = \{X/roger\}$
  - $\{worksAt(X,Y)\} \succeq \{worksAt(Z,bosch), researcher(Z)\}$,
    $\theta = \{X/Z, Y/bosch\}$

- Generalization as entailment
  - $\underbrace{person(X) \leftarrow researcher(X)}_{Hyp1}\ \underbrace{person(X) \leftarrow researcher(X), alive(X)}_{Hyp2}$

    $$Hyp1 \succeq Hyp2$$

  - $livesIn(roger, bottmingen)$? $livesIn(roger, switzerland)$

Common Techniques in ILP

- Generality ($\succeq$): essential component of symbolic learning systems

- Genaralization as θ-subsumption
  - $person(X) \succeq person(roger), \;\; \theta = \{X/roger\}$
  - $\{worksAt(X, Y)\} \succeq \{worksAt(Z, bosch), researcher(Z)\},$
    $\theta = \{X/Z, Y/bosch\}$

- Generalization as entailment
  - $\underbrace{person(X) \leftarrow researcher(X)}_{Hyp1}\ \underbrace{person(X) \leftarrow researcher(X), alive(X)}_{Hyp2}$

    $$Hyp1 \succeq Hyp2$$

  - $livesIn(roger, bottmingen)\ ?\ livesIn(roger, switzerland)$
    $T:\ livesIn(X, switzerland) \leftarrow livesIn(X, bottmingen)$

Common Techniques in ILP

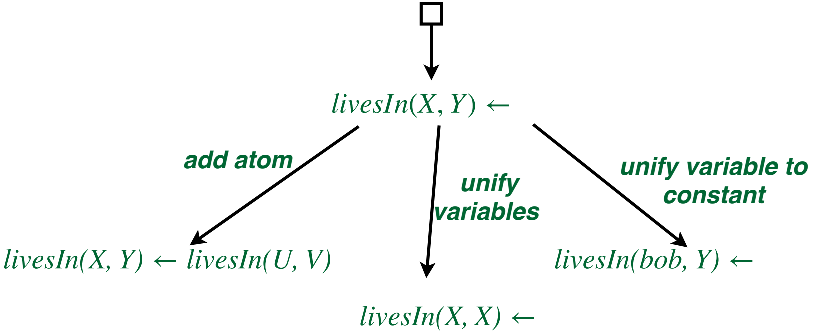
- Generality ($\succeq$): essential component of symbolic learning systems
- Genaralization as θ-subsumption
  - $person(X) \succeq person(roger), \;\; \theta = \{X/roger\}$
  - $\{worksAt(X, Y)\} \succeq \{worksAt(Z, bosch), researcher(Z)\}$, $\theta = \{X/Z, Y/bosch\}$
- Generalization as entailment
  - $\underbrace{person(X) \leftarrow researcher(X)}_{Hyp1} \; \underbrace{person(X) \leftarrow researcher(X), alive(X)}_{Hyp2}$

    $$Hyp1 \succeq Hyp2$$

  - $livesIn(roger, bottmingen) \succeq livesIn(roger, switzerland)$
    $T: \; livesIn(X, switzerland) \leftarrow livesIn(X, bottmingen)$

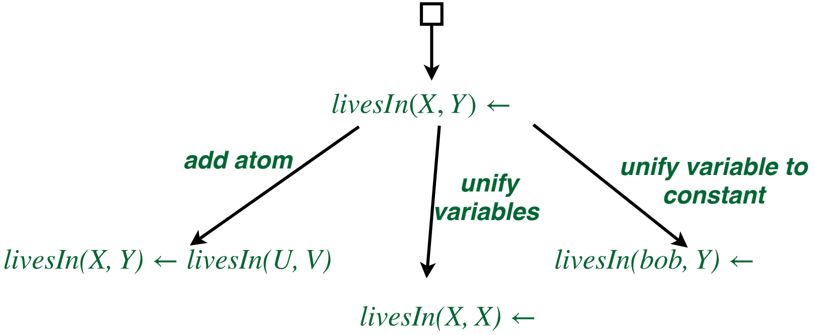
Common Techniques in ILP

- 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, 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., complete, incomplete, partially complete
- background knowledge, e.g., 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

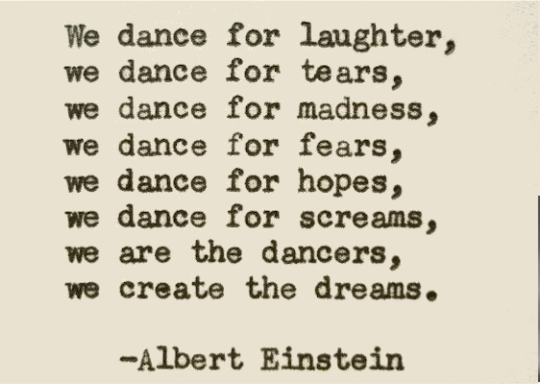
Open World Assumption: negative facts cannot be easily derived

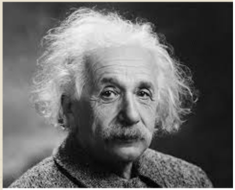
Maybe R. Federer is a researcher and A. Einstein was a dancer?

Challenges of Rule Induction from KGs

Open World Assumption: negative facts cannot be easily derived

Maybe R. Federer is a researcher and A. Einstein was a dancer?





Challenges of Rule Induction from KGs

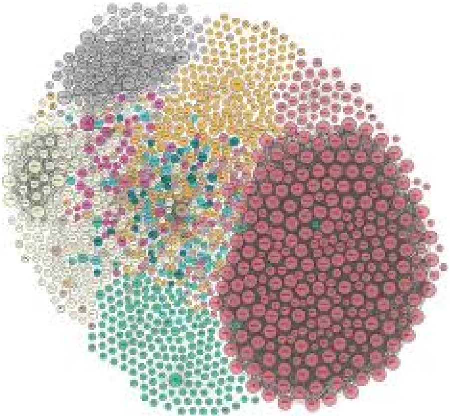
Data bias: KGs are extracted from text, which typically mentions only popular entities and interesting facts about them.

"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



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 reflect interesting correlations that can predict sufficiently many correct facts?

Answer:

Relational association rule mining! Roots in classical datamining.

Association Rules

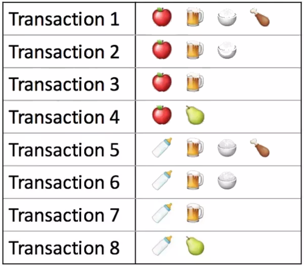
- Classical data mining task: Given a transaction database, find out products (called itemsets) that are frequently bought together and form recommendation rules.

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.

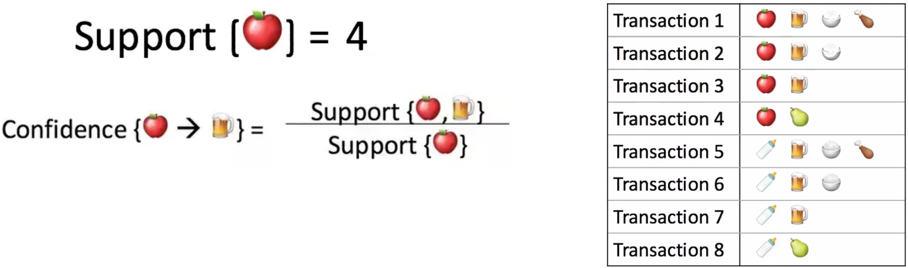
Some Rule Measures

Support, confidence, lift



Some Rule Measures

Support, confidence, lift



Some Rule Measures

Support, confidence, lift

Support {🍎} = 4

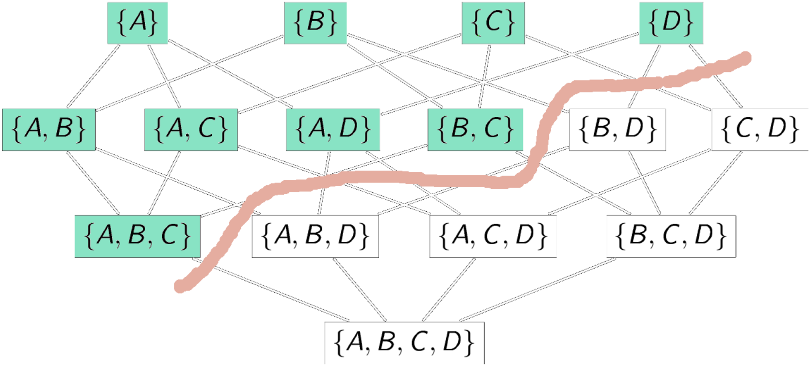
$$\text{Confidence } \{🍎 \rightarrow 🍺\} = \frac{\text{Support } \{🍎, 🍺\}}{\text{Support } \{🍎\}}$$

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

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 frequent itemset mining algorithm



How to Apply this to Relational Data?

- DOWNGRADING DATA: Can we change the representation from richer representations to simpler ones? (So we can use systems working with simpler representations)
- UPGRADING SYSTEMS: Can we develop systems that work with richer representations (starting from systems for simpler representations)?

Downgrading the Data

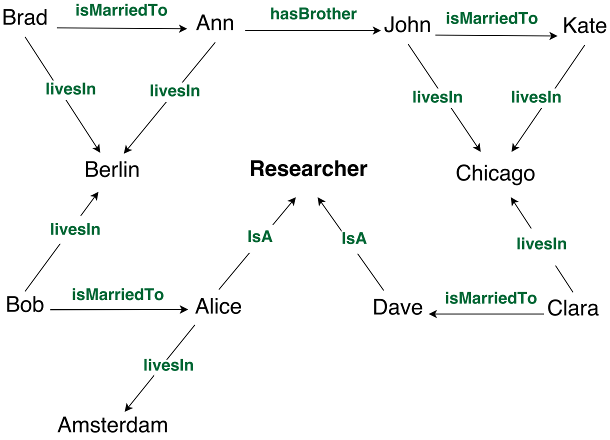
- **Propositionalization** [Krogel *et al.*, 2003]: transform a KG into a transaction database

	bornInUS	*livesInUS*	*isMarriedToSinger*	*researcher*	*sportsman*
p1	✓	✓			✓
p2	✓	✓		✓	
p3	✓	✓			
p4	✓	✓			
p5	✓		✓		
p6	✓		✓		✓
p7	✓			✓	
p8	✓	✓			

Upgrading the Systems

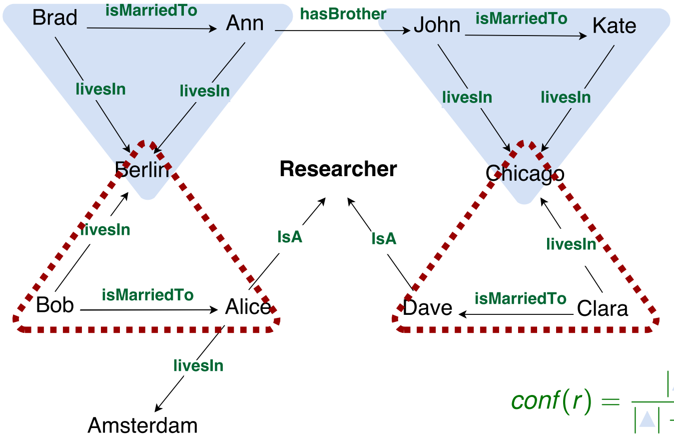
- Start from existing system for simpler representation
- Extend it for use with richer representation (while trying to keep the original system as a special case)

Relational Association Rule Learning



Relational Association Rule Learning

Confidence, e.g., WARMER [Goethals and den Bussche, 2002]
Closed World Assumption (CWA): Whatever is missing is false

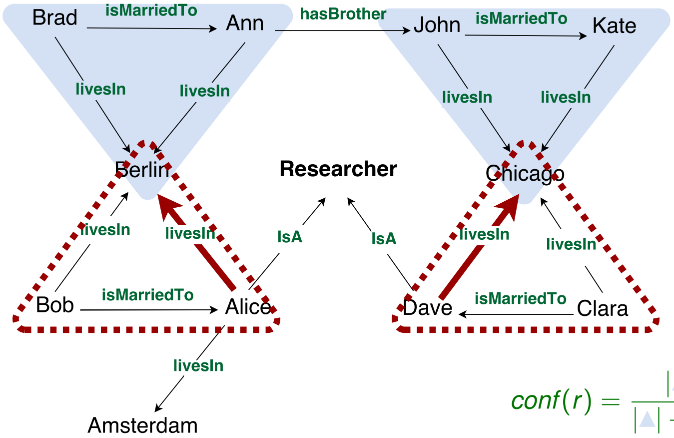


$$conf(r) = \frac{|\blacktriangle|}{|\blacktriangle| + |\triangle|} = \frac{2}{4}$$

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

Relational Association Rule Learning

Confidence, e.g., WARMER [Goethals and den Bussche, 2002]
Closed World Assumption (CWA): Whatever is missing is false



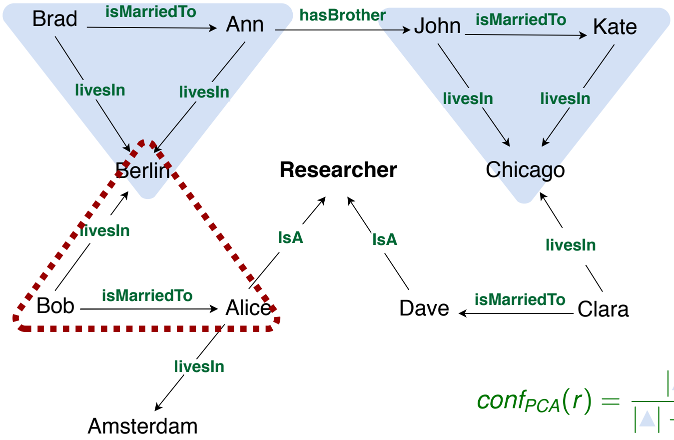
$$conf(r) = \frac{|\blacktriangle|}{|\blacktriangle| + |\triangle|} = \frac{2}{4}$$

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

Relational Association Rule Learning

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

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

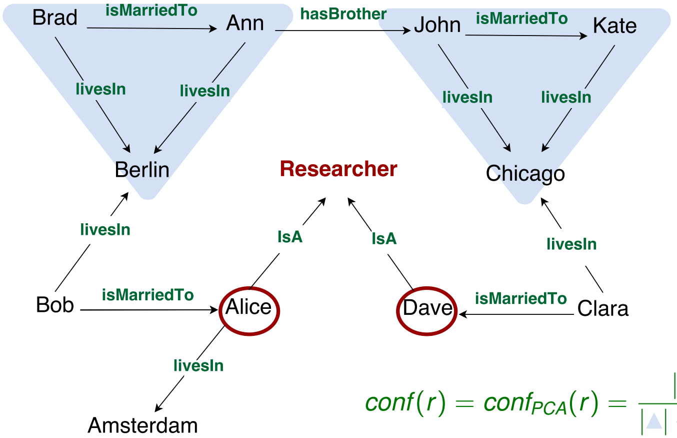


$$conf_{PCA}(r) = \frac{|\blacktriangle|}{|\blacktriangle| + |\triangle|} = \frac{2}{3}$$

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

Relational Association Rule Learning

Exception-enriched rules: **Open World Assumption** is a challenge!



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

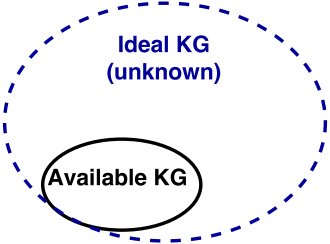
$$r: \ livesIn(X,Y) \leftarrow isMarriedTo(Z,X), livesIn(Z,Y), not\ isA(X, researcher)$$

Horn Theory Revision

Quality-based Horn Theory Revision

Given:

- Available KG

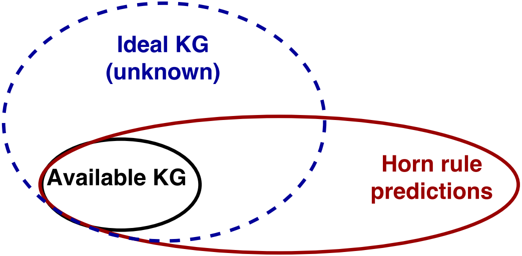


Horn Theory Revision

Quality-based Horn Theory Revision

Given:

- Available KG
- Horn rule set

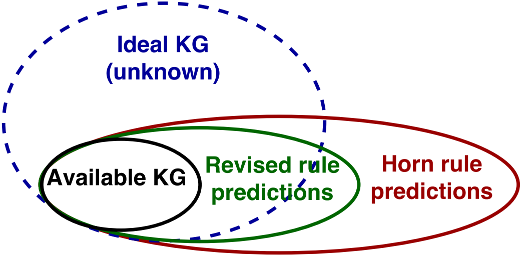


Horn Theory Revision

Quality-based Horn Theory Revision

Given:

- Available KG
- Horn rule set



Find:

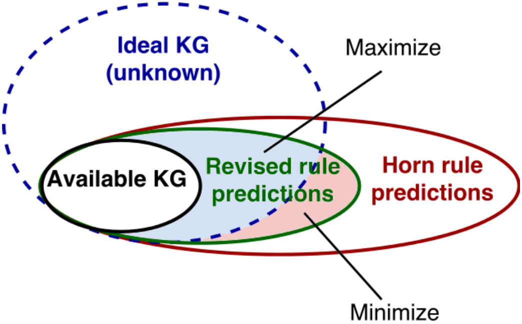
- Nonmonotonic revision of Horn rule set

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?

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

Avoid Data Overfitting

How to distinguish exceptions from noise?

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

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

Avoid Data Overfitting

How to distinguish exceptions from noise?

$$r1: \; livesIn(X,Z) \leftarrow isMarriedTo(Y,X), livesIn(Y,Z), not \;\; researcher(X)$$
$$not\_livesIn(X,Z) \leftarrow isMarriedTo(Y,X), livesIn(Y,Z), researcher(X)$$

$$r2: \; livesIn(X,Z) \leftarrow bornIn(X,Z), not \;\; moved(X)$$
$$not\_livesIn(X,Z) \leftarrow bornIn(X,Z), moved(X)$$

Avoid Data Overfitting

How to distinguish exceptions from noise?

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

$\quad not\_livesIn(X, Z) \leftarrow isMarriedTo(Y, X), livesIn(Y, Z), researcher(X)$

$r2: \ livesIn(X, Z) \leftarrow bornIn(X, Z), not\ \ moved(X)$

$\quad not\_livesIn(X, Z) \leftarrow bornIn(X, Z), moved(X)$

$\{livesIn(c, d), not\_livesIn(c, d)\}$ are conflicting predictions

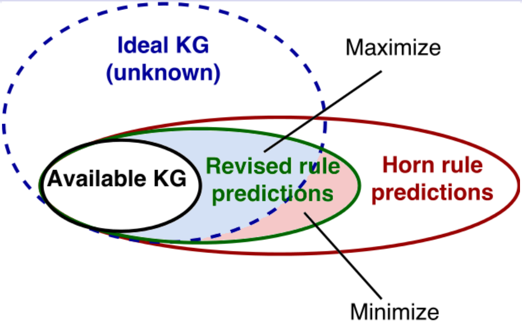
Intuition: Rules with good exceptions should make few conflicting predictions

Horn Theory Revision

Quality-based Horn Theory Revision

Given:

- Available KG
- Horn rule set

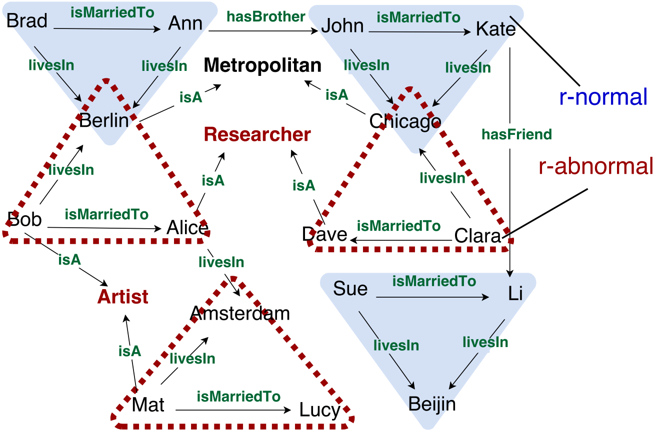


Find:

- Nonmonotonic revision of Horn rules, such that
  - number of conflicting predictions is **minimal**
  - average descriptive rule measure (e.g., confidence) is **maximal**

M. Gad-Elrab, D. Stepanova, J. Urbani, G. Weikum. Exception-enriched Rule Learning from Knowledge Graphs. ISWC2016
D. Tran, D. Stepanova, M. Gad-Elrab, F. Lisi, G. Weikum. Towards Nonmonotonic Relational Learning from KGs. ILP2016

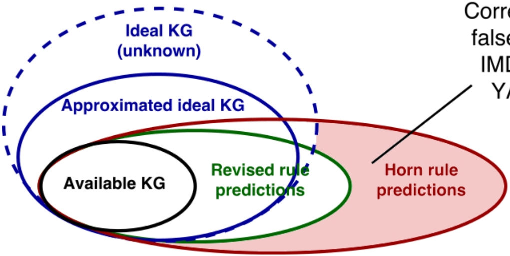
Exception Candidates



$$r\colon livesIn(X,Z) \leftarrow isMarriedTo(Y,X)\,,\, livesIn(Y,Z) \qquad \begin{Bmatrix} not\ \ researcher(X) \\ not\ \ artist(Y) \end{Bmatrix}$$

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 DLV reasoning system



Correctly removed false predictions:
IMDB: **57.75 %**
YAGO: **85 %**

https://github.com/htran010589/nonmonotonic-rule-mining.git

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 DLV reasoning system

Examples of revised rules:

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

$r_1 : \mathit{writtenBy}(X, Z) \leftarrow \mathit{hasPredecessor}(X, Y), \mathit{writtenBy}(Y, Z), \mathit{not\ american\_film}(X)$

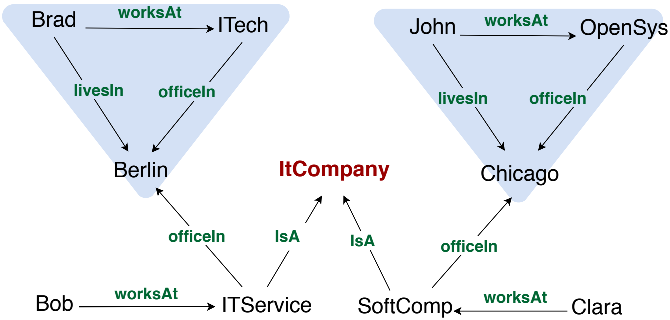
Spouses of film directors appear on the cast, unless they are silent film actors

$r_2 : \mathit{actedIn}(X, Z) \leftarrow \mathit{isMarriedTo}(X, Y), \mathit{directed}(Y, Z), \mathit{not\ silent\_film\_actor}(X)$

https://github.com/htran010589/nonmonotonic-rule-mining.git

Absurd Rules due to Data Incompleteness

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

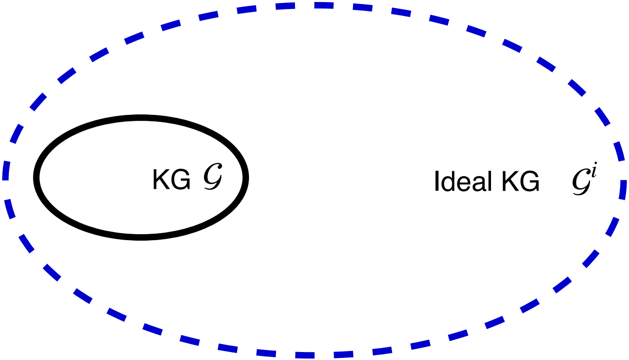


$$conf(r) = conf_{PCA}(r) = 1$$

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

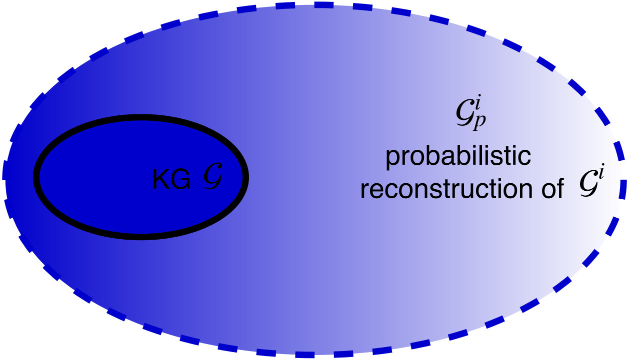
Ideal KG

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



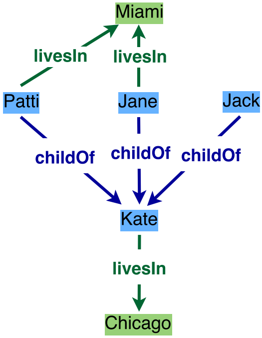
Hybrid Rule Measure

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

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

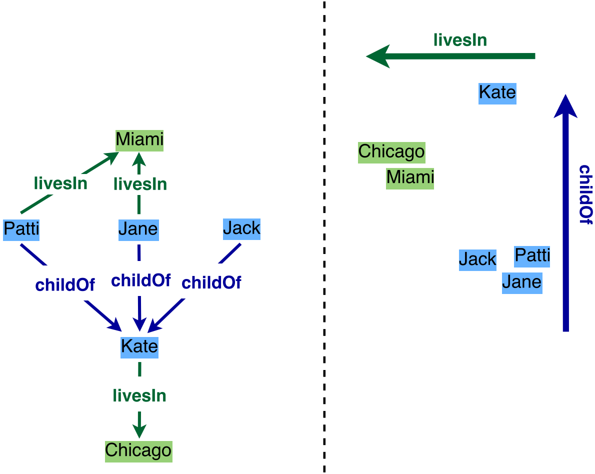
KG Embeddings

- **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



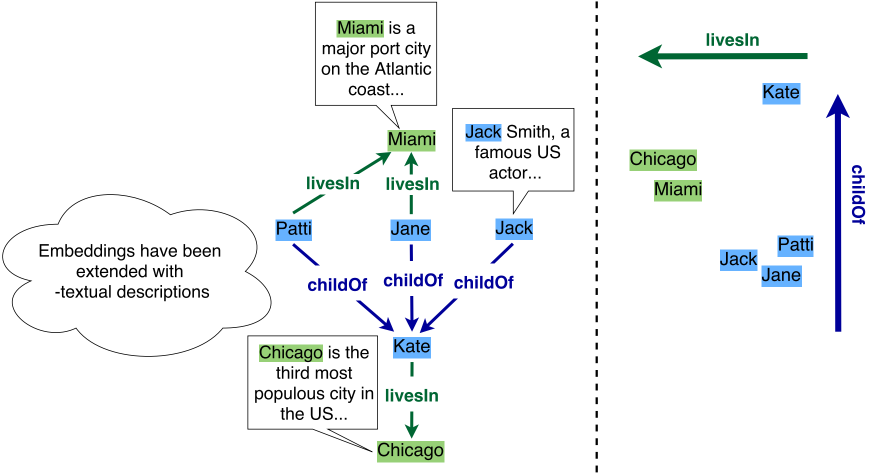
KG Embeddings

- **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



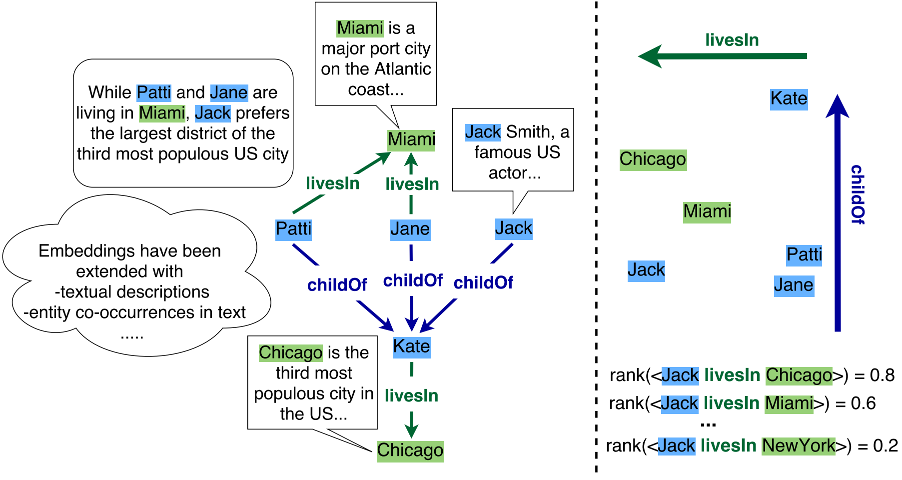
KG Embeddings

- **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

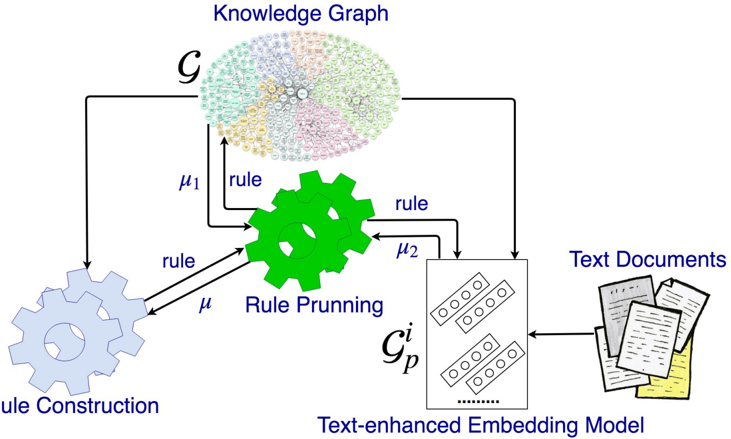


KG Embeddings

- **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



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

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

add closing atom

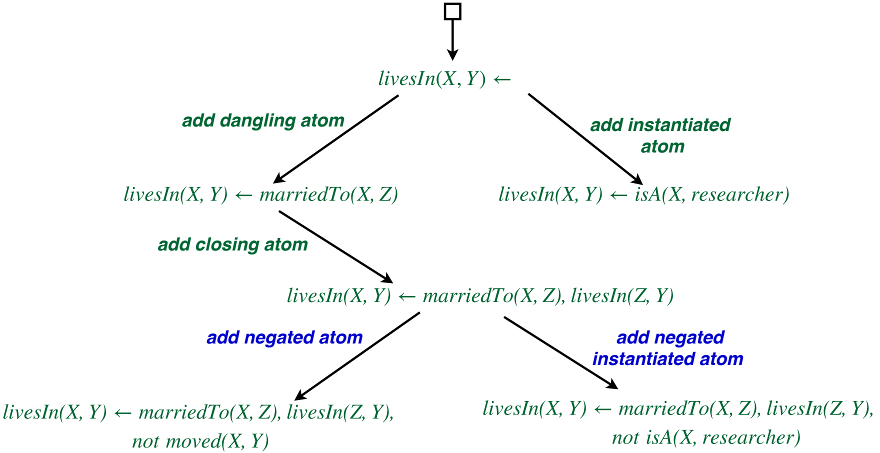
$livesIn(X, Y) \leftarrow marriedTo(X, Z), livesIn(Z, Y)$

add instantiated atom

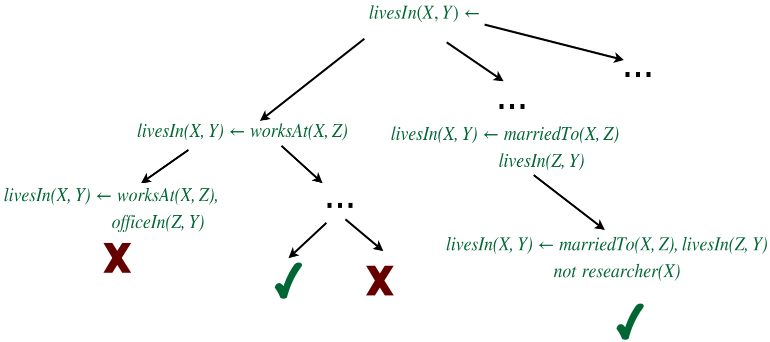
$livesIn(X, Y) \leftarrow isA(X, researcher)$

Rule Construction

- Clause exploration from general to specific
  - Our work: closed and safe rules with negation
    livesIn(X, Y) ← 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

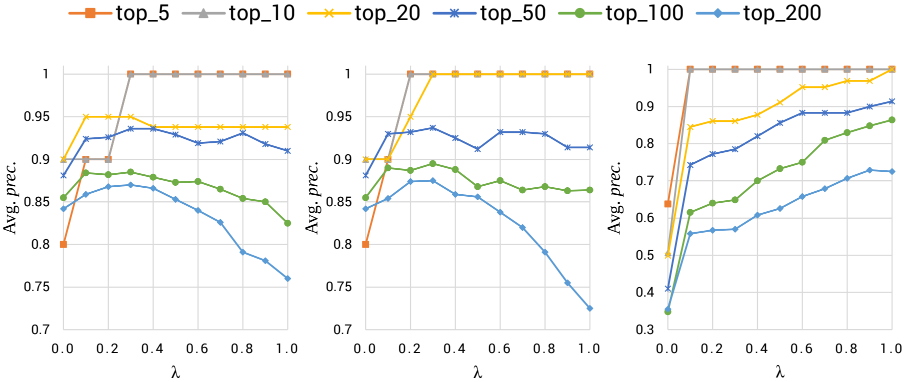
- Embedding models $\mathcal{G}_p^i$:
  - TransE [Bordes *et al.*, 2013], HolE [Nickel *et al.*, 2016]
  - With text: SSP [Xiao *et al.*, 2017]

- 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

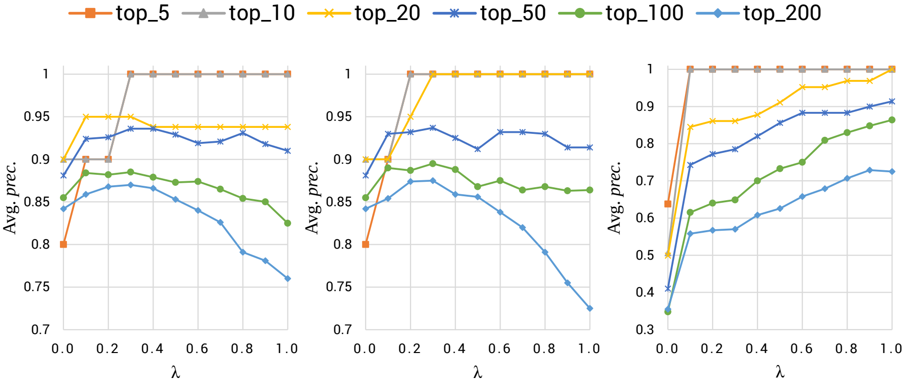
Evaluation of Hybrid Rule Measure



(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.

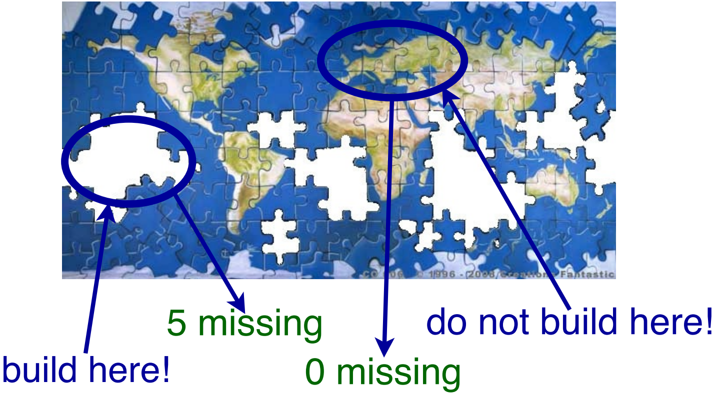
- Positive impact of embeddings in all cases for $\lambda = 0.3$
- **Note:** in (c) comparison to AMIE [Galarraga *et al.*, 2015] ($\lambda = 0$)

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 Cardinality Meta-data in Rule Learning

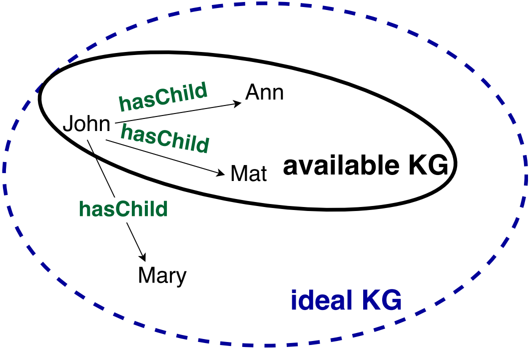
Goal: make use of cardinality constraints on edges of the KG to improve rule learning.



T. Pellissier-Tanon, D. Stepanova, S. Razniewski, P. Mirza, G. Weikum. Completeness-aware rule learning from KGs. ISWC2017.
`https://github.com/Tpt/CARL.git`

Cardinality Statements

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



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

Completeness Confidence

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

$$conf_{comp}(r) = \frac{|\blacktriangle|}{|\blacktriangle| + |\triangle| - npi(r)}$$

- $npi(r)$: 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

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

$recall_{comp}$: 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

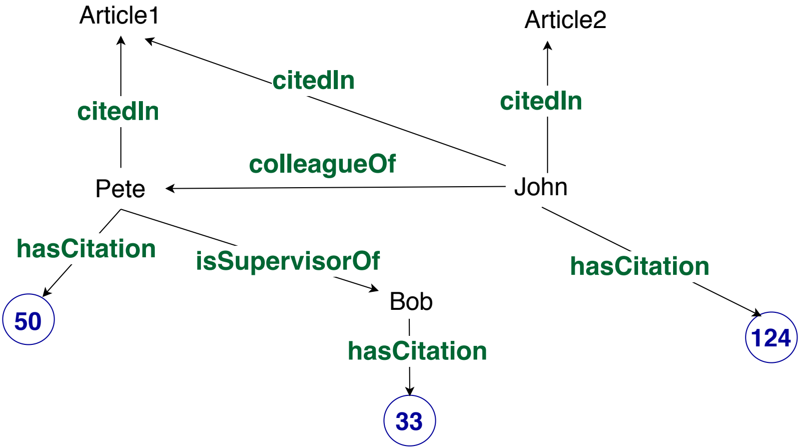
$$dir\_metric(r) = \frac{npi(r) - npc(r)}{2 \cdot (npi(r) + npc(r))} + 0.5$$

wdm : weighted combination of confidence and directional metric

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

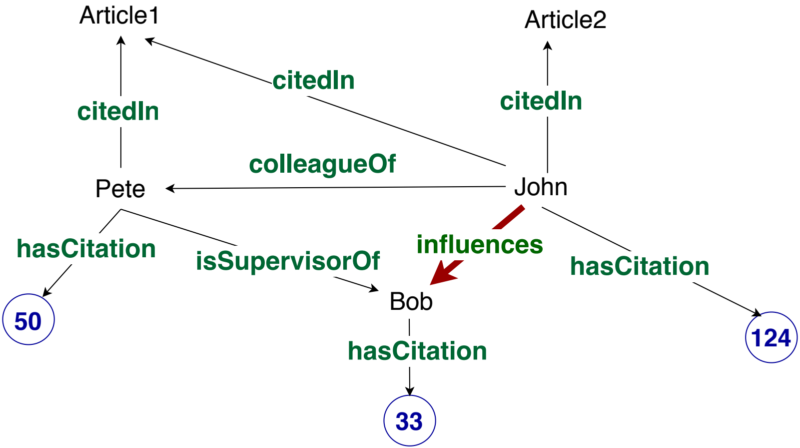
https://github.com/Tpt/CARL.git

Numerical Rules



$$\begin{aligned} \mathit{influences}(X, Y) \leftarrow\ & \mathit{colleagueOf}(X, Z), \mathit{supervisorOf}(Z, Y), \\ & X.\mathit{hasCitation} > Z.\mathit{hasCitation} \end{aligned}$$

Numerical Rules

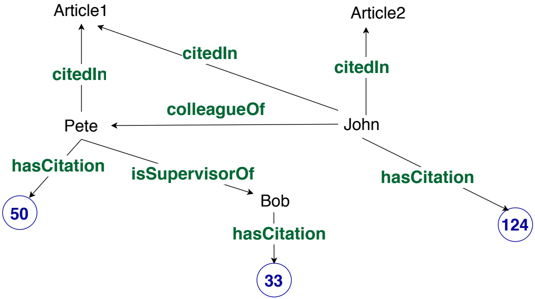


$$influences(X, Y) \leftarrow colleagueOf(X, Z), supervisorOf(Z, Y), \\ X.hasCitation > Z.hasCitation$$

Rule Learning via Boolean Matrix Multiplication

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

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

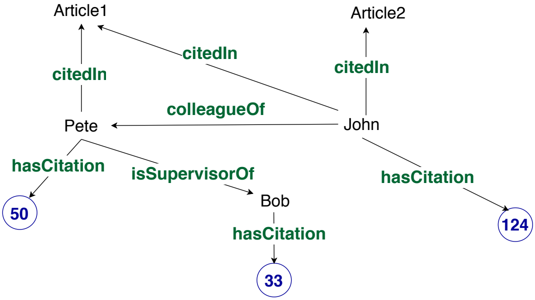


Rule Learning via Boolean Matrix Multiplication

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

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

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



Rule Learning via Boolean Matrix Multiplication

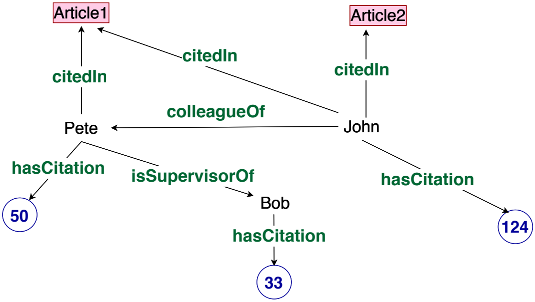
NeuralLP [Yang *et al.*, 2017]: Differentiable rule learning

$$
M_{\text{citedIn}} = \begin{array}{c} \\ \\ \\ \\ \\ \\ \end{array}
\begin{array}{ccccc}
\text{john} & \text{pete} & \text{bob} & \text{article1} & \text{article2} \\
\end{array}
$$

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

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

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



Rule Learning via Boolean Matrix Multiplication

Differentiable learning framework via (sparse) matrix-vector multiplication

Rule Learning via Boolean Matrix Multiplication

Differentiable learning framework via (sparse) matrix-vector multiplication

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

Rule Learning via Boolean Matrix Multiplication

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

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

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

Rule Learning via Boolean Matrix Multiplication

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

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

Rule Learning via Boolean Matrix Multiplication

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.

$$
\tilde{M}_{r^{\leq}_{\overline{pq}}} =
\begin{array}{c}
\begin{array}{cccccc} \mathtt{NaN} \ldots \mathtt{NaN} & \tilde{g}_1 & \leq & \ldots & \leq & \tilde{g}_n \end{array} \\
\left[\begin{array}{cccccccc}
0 & \cdots & 0 & \cdots & & & & 0 \\
\vdots & & \vdots & & & & & \vdots \\
 & & 0 & \cdots & & & & 0 \\
 & & \vdots & 1 & \cdots & & & 1 \\
 & & & 0 & 1 & \cdots & & \\
 & & & \vdots & 0 & 1 & \cdots & \\
 & & & & 0 & 1 & \cdots & \\
0 & \cdots & 0 & & \cdots & 0 & 1 & 1
\end{array}\right]
\end{array}
\begin{array}{c}
\\ \mathtt{NaN} \\ \vdots \\ \mathtt{NaN} \\ \tilde{f}_1 \\ \rotatebox{90}{$\leq$} \\ \vdots \\ \rotatebox{90}{$\leq$} \\ \tilde{f}_m
\end{array}
$$

Monotonic borderline:

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

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

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

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

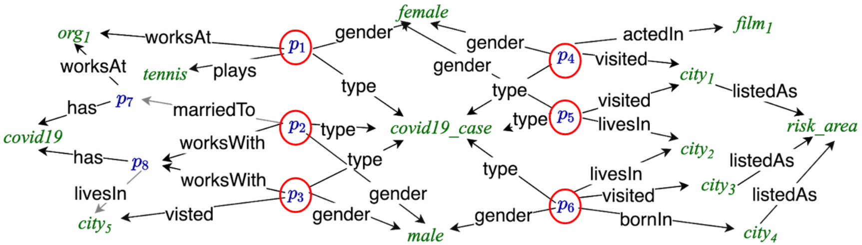
$$\mathit{symptomHasRiskFactors}(X, Y) \leftarrow f(X), \mathit{symtompOfDisease}(X, Z),$$
$$\mathit{diseaseHasRiskFactors}(Z, Y)$$

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

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

Explainable Clustering

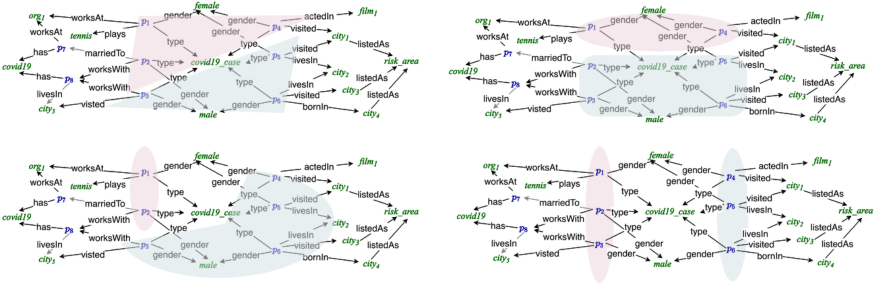
Huge Knowledge Graphs ⇨ Hard to Explore ⇨ Requires Summarization *E.g. Clustering*



Which is the best division for T={p_1...p_6}?

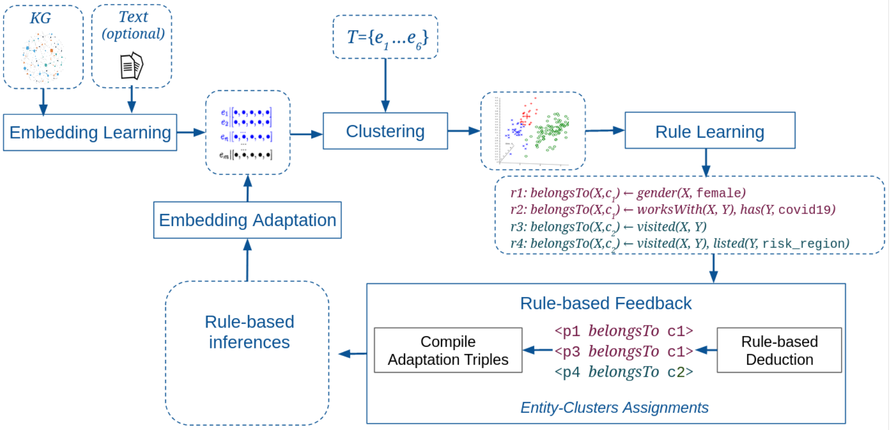
Explainable Clustering

Huge Knowledge Graphs → Hard to Explore → Requires Summarization *E.g. Clustering*



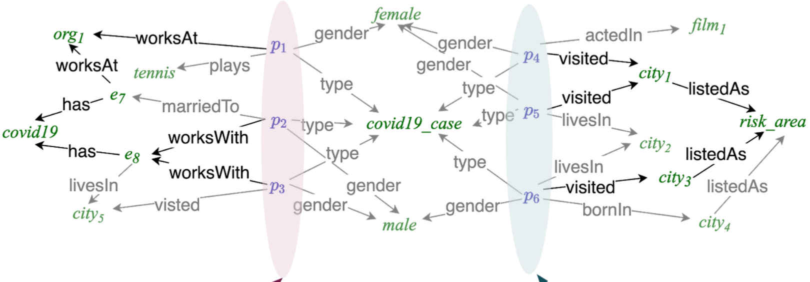
Which is the best division for $T=\{p_1 \ldots p_6\}$?

Explainable Clustering



M. Gad-Elrab, D. Stepanova, T. Kien Trung, H. Adel, G. Weikum: Explainable Embedding-based Clustering in KGs. ISWC 2020.
`https://github.com/mhmgad/ExCut`

Explainable Clustering



"People got infected from coworkers"

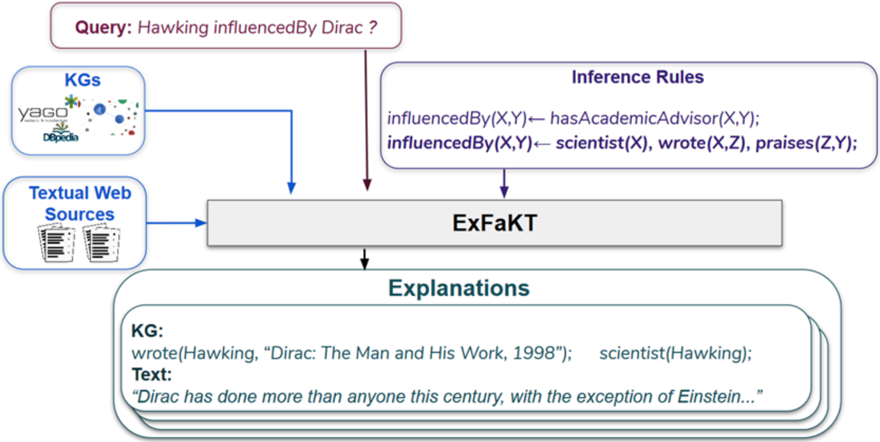
r1: belongsTo(X,C$_1$) ← worksWith(X, Y), has(Y, `covid19`*)*

"People who visited risk area"

r3: belongsTo(X,C$_2$) ← visited(X, Y), listed(Y, `risk_region`*)*

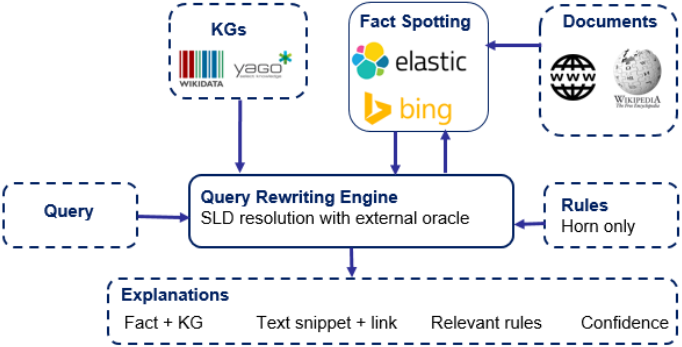
M. Gad-Elrab, D. Stepanova, T. Kien Trung, H. Adel, G. Weikum: Explainable Embedding-based Clustering in KGs. ISWC 2020.
`https://github.com/mhmgad/ExCut`

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.

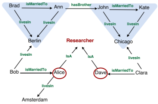
Rule-based Fact Checking



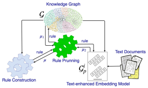
M. Gad-Elrab, D. Stepanova, J. Urbani, G. Weikum. ExFakt: A Framework for Explaining Facts over KGs and Text. WSDM 2019.
M. Gad-Elrab, D. Stepanova, J. Urbani, G. Weikum. Tracy: Tracing Facts over Knowledge Graphs and Text. WWW 2019.

Summary

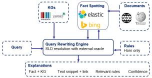
- Horn rule learning
- Exploiting embeddings to guide rule learning



- Numerical rule learning



- Applications:
  - Explainable clustering
  - Rule-based fact checking



Outlook

- Learning rules from text, images, etc.
- Make use of rules for explaining ML models
- ...

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, Zico Kolter, Csaba Domokos, Po-Wei Wang, Tran Kien Trung, Heike Adel
- For fruitful discussions and/or making slides available online:
  - Thomas Eiter, Stephen Muggleton, Luc De Raedt, Fabian Suchanek
- For providing amazing working atmosphere:
  - Bosch Center for AI

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-A. Krogel, Simon Alan Rawles, Filip Zelezný, Peter A. Flach, Nada Lavrac, and Stefan Wrobel.
Comparative evaluation of approaches to propositionalization.
In *ILP*, pages 197–214, 2003.

References II

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

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.

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.