Rule Induction and Reasoning in Knowledge Graphs

Daria Stepanova

Bosch Center for Artificial Intelligence, Renningen, Germany

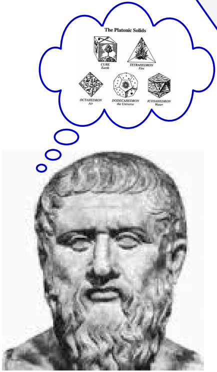
ODSC 2022

What is Knowledge?

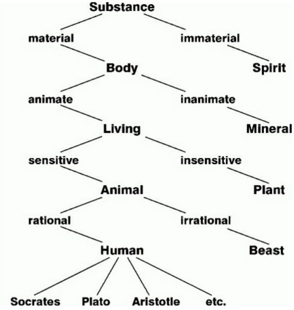
Plato: *"Knowledge is justified true belief"*

350 BC

Personal



External



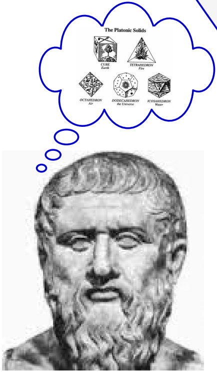
Tree of Knowledge

What is Knowledge?

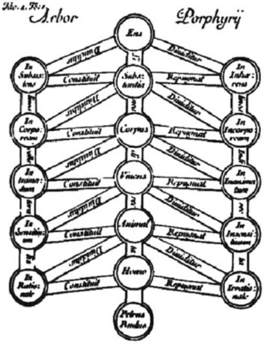
Plato: *“Knowledge is justified true belief”*

350 BC

Personal



External



Knowledge Graphs as Digital Knowledge

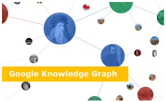
"Digital knowledge is semantically enriched machine processable data"

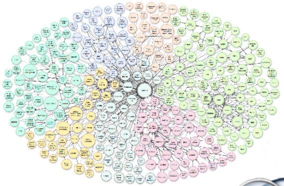
2022

Personal

External

yago
select knowledge





WIKIDATA

NELL

Knowledge Graph

Industrial KGs

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

Semantic Web Search

Google

winner of Australian Open 2022

Rafael Nadal

Spanish tennis player

Rafael Nadal Parera is a Spanish professional tennis player. He is ranked world No. 5 in singles by the Association of Tennis Professionals; he has been ranked world No. 1 for 209 weeks and finished as the year-end No. 1 five times. Wikipedia

Born: June 3, 1986 (age 35 years), Manacor, Spain

Grand slams won (singles): 21

Height: 1.85 m

Spouse: Maria Francisca Perello (m. 2019)

Books: Rafa

Parents: Sebastián Nadal, Ana María Parera

Nicknames: El Nino, King of clay, Rafa, Rafi, Spain's Raging Bull

Semantic Web Search

Google

$\exists X\ winnerOf(X, AustralianOpen)$

Rafael Nadal

Spanish tennis player

Rafael Nadal Parera is a Spanish professional tennis player. He is ranked world No. 5 in singles by the Association of Tennis Professionals; he has been ranked world No. 1 for 209 weeks and finished as the year-end No. 1 five times. Wikipedia

Born: June 3, 1986 (age 35 years), Manacor, Spain

Grand slams won (singles): 21

Height: 1.85 m

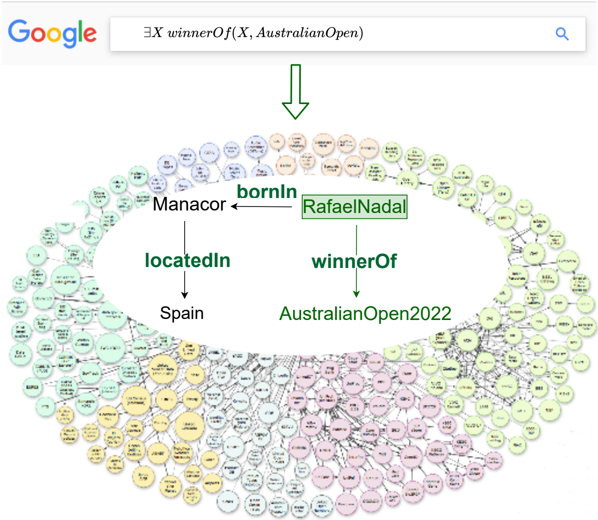
Spouse: Maria Francisca Perello (m. 2019)

Books: Rafa

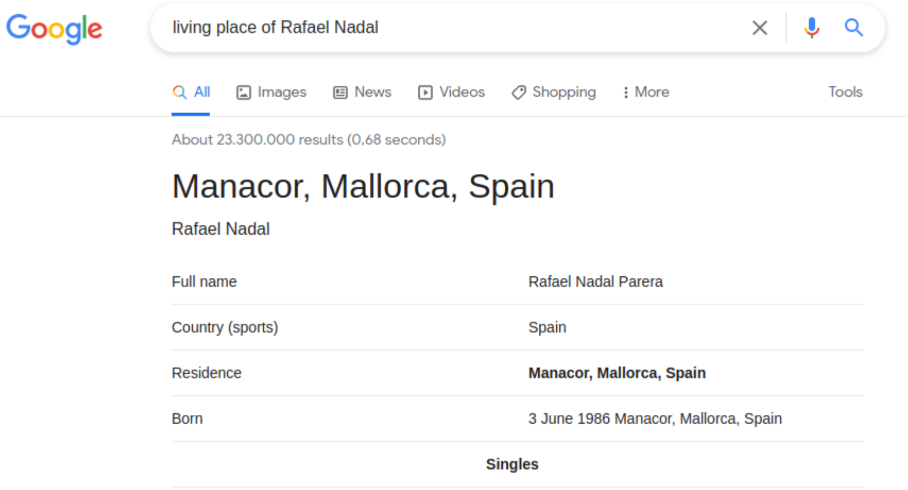
Parents: Sebastián Nadal, Ana María Parera

Nicknames: El Nino, King of clay, Rafa, Rafi, Spain's Raging Bull

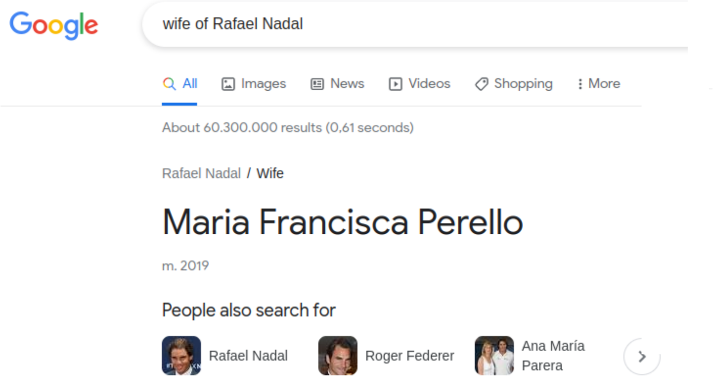
Semantic Web Search



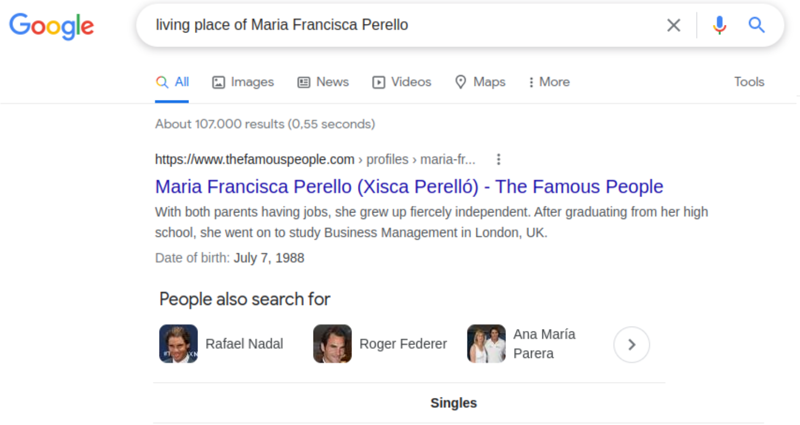
Semantic Web Search



Semantic Web Search



Incompleteness of KGs



Human Reasoning

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

Married people live together

$$marriedTo(rafael, mariaFrancisca)$$

Rafael is married to Maria Francisca

$$livesIn(rafael, manacor)$$

Rafael lives in Manacor

Human Reasoning

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

Married people live together

$$\mathit{marriedTo}(\mathit{rafael}, \mathit{mariaFrancisca})$$

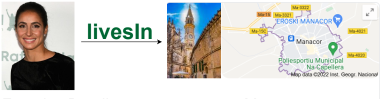
Rafael is married to Maria Francisca

$$\mathit{livesIn}(\mathit{rafael}, \mathit{manacor})$$

Rafael lives in Manacor

$$\mathit{livesIn}(\mathit{mariaFrancisca}, \mathit{manacor})$$

Maria Francisca lives in Manacor



Maria Franciso Borello

Manacor

Human Reasoning

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

Married people live together

$$marriedTo(rafael, mariaFrancisca)$$

Rafael is married to Maria Francisca

$$livesIn(rafael, manacor)$$

Rafael lives in Manacor

$$livesIn(mariaFrancisca, manacor)$$

Maria Francisca lives in Manacor

Maria Franciso Borello

Manacor

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 Nadal is married to Maria and lives in M., then Maria lives there too
$livesIn(maria, manacor) \leftarrow isMarried(nadal, maria), livesIn(nadal, manacor)$

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 with 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, ...
- Neuro-symbolic AI 2015 -...
  Hitzler, De Raedt, Leskovec, Tannenbaum, ...

Learning from Examples

Inductive Learning from Examples [Muggleton, 1991]

Given:

- $E^+ = \{\mathit{fatherOf}(\mathit{john}, \mathit{mary}), \mathit{fatherOf}(\mathit{david}, \mathit{steve})\}$
- $E^- = \{\mathit{fatherOf}(\mathit{kathy}, \mathit{ellen}), \mathit{fatherOf}(\mathit{john}, \mathit{steve})\}$
- $T = \{\mathit{parentOf}(\mathit{john}, \mathit{mary}), \mathit{male}(\mathit{john}),$
  $\mathit{parentOf}(\mathit{david}, \mathit{steeve}), \mathit{male}(\mathit{david}),$
  $\mathit{parentOf}(\mathit{kathy}, \mathit{ellen}), \mathit{female}(\mathit{kathy})\}$
- 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
  - $\mathit{person}(X) \succeq \mathit{person}(\mathit{nadal}), \;\; \theta = \{X/\mathit{nadal}\}$

Common Techniques in ILP

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

Common Techniques in ILP

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

Common Techniques in ILP

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

Common Techniques in ILP

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

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

- 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
  - $\mathit{person}(X) \succeq \mathit{person}(\mathit{nadal}), \;\; \theta = \{X/\mathit{nadal}\}$
  - $\{\mathit{worksAt}(X, Y)\} \succeq \{\mathit{worksAt}(Z, \mathit{bosch}), \mathit{researcher}(Z)\}$, $\theta = \{X/Z, Y/\mathit{bosch}\}$
- Generalization as entailment
  - $\underbrace{\mathit{person}(X) \leftarrow \mathit{researcher}(X)}_{\mathit{Hyp1}} \; \underbrace{\mathit{person}(X) \leftarrow \mathit{researcher}(X), \mathit{alive}(X)}_{\mathit{Hyp2}}$

Common Techniques in ILP

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

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

- 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
  - $\mathit{person}(X) \succeq \mathit{person}(\mathit{nadal}), \;\; \theta = \{X/\mathit{nadal}\}$
  - $\{\mathit{worksAt}(X, Y)\} \succeq \{\mathit{worksAt}(Z, \mathit{bosch}), \mathit{researcher}(Z)\}$,
    $\theta = \{X/Z, Y/\mathit{bosch}\}$
- Generalization as entailment
  - $\underbrace{\mathit{person}(X) \leftarrow \mathit{researcher}(X)}_{\mathit{Hyp1}} \; \underbrace{\mathit{person}(X) \leftarrow \mathit{researcher}(X), \mathit{alive}(X)}_{\mathit{Hyp2}}$

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

  - $\mathit{livesIn}(\mathit{nadal}, \mathit{manacor})$? $\mathit{livesIn}(\mathit{nadal}, \mathit{spain})$

Common Techniques in ILP

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

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

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

    $$\textit{Hyp1} \succeq \textit{Hyp2}$$

  - $\textit{livesIn}(\textit{nadal}, \textit{manacor}) \; ? \; \textit{livesIn}(\textit{nadal}, \textit{spain})$
    $T: \; \textit{livesIn}(X, \textit{spain}) \leftarrow \textit{livesIn}(X, \textit{manacor})$

Common Techniques in ILP

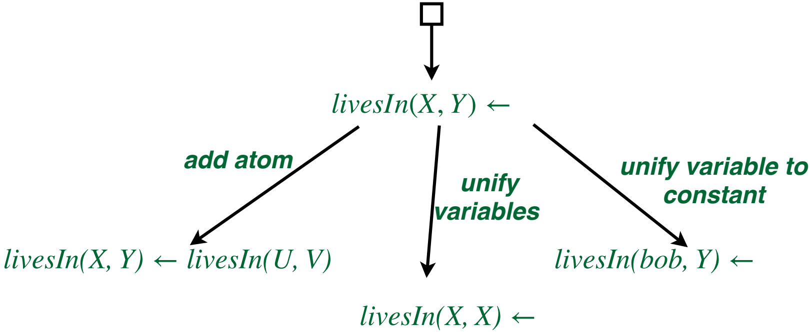
- Generality ($\succeq$): essential component of symbolic learning systems
- Genaralization as θ-subsumption
  - $\mathit{person}(X) \succeq \mathit{person}(\mathit{nadal}), \ \theta = \{X/\mathit{nadal}\}$
  - $\{\mathit{worksAt}(X, Y)\} \succeq \{\mathit{worksAt}(Z, \mathit{bosch}), \mathit{researcher}(Z)\}$,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

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

  - $\mathit{livesIn}(\mathit{nadal}, \mathit{manacor}) \succeq \mathit{livesIn}(\mathit{nadal}, \mathit{spain})$
    $T: \ \mathit{livesIn}(X, \mathit{spain}) \leftarrow \mathit{livesIn}(X, \mathit{manacor})$

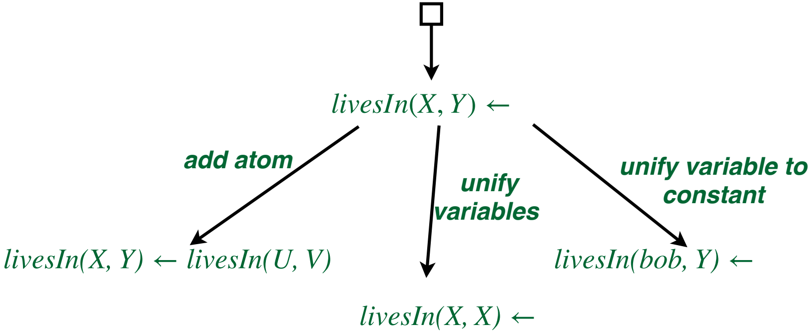
Common Techniques in ILP

- 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

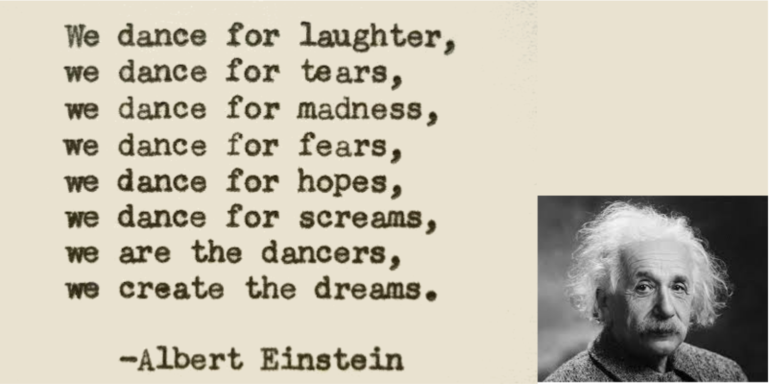
Open World Assumption: negative facts cannot be easily derived

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



Challenges of Rule Induction from KGs

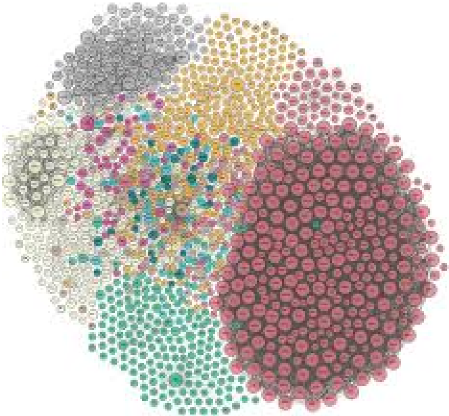
Data bias: KGs are extracted from text, which typically mentions only popular entities and interesting facts about them.

"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.

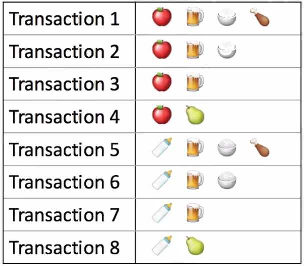
Transaction 1	🍎 🍺 🍚 🍗
Transaction 2	🍎 🍺 🍚
Transaction 3	🍎 🍺
Transaction 4	🍎 🍐
Transaction 5	🍼 🍺 🍚 🍗
Transaction 6	🍼 🍺 🍚
Transaction 7	🍼 🍺
Transaction 8	🍼 🍐

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

Some Rule Measures

Support, confidence, lift

Support [] = 4



Some Rule Measures

Support, confidence, lift

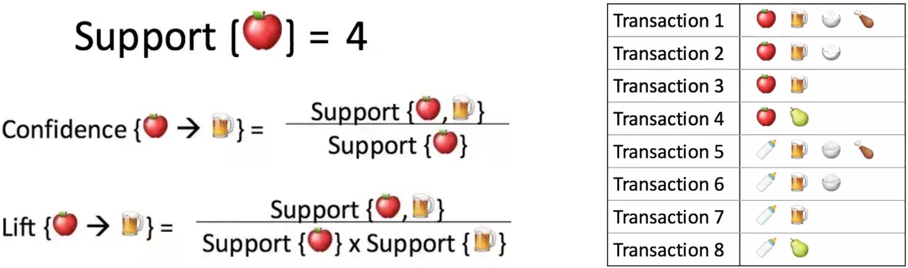
Support [🍎] = 4

Confidence {🍎 → 🍺} = Support {🍎, 🍺} / Support {🍎}

Transaction 1	🍎 🍺 🍚 🍗
Transaction 2	🍎 🍺 🍚
Transaction 3	🍎 🍺
Transaction 4	🍎 🍐
Transaction 5	🍼 🍺 🍚 🍗
Transaction 6	🍼 🍺 🍚
Transaction 7	🍼 🍺
Transaction 8	🍼 🍐

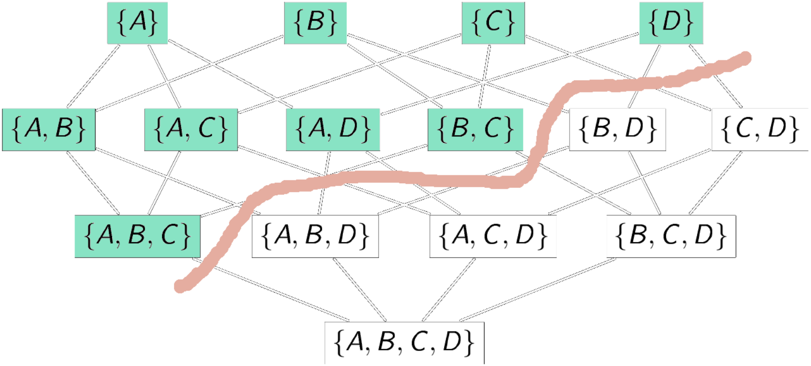
Some Rule Measures

Support, confidence, lift



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

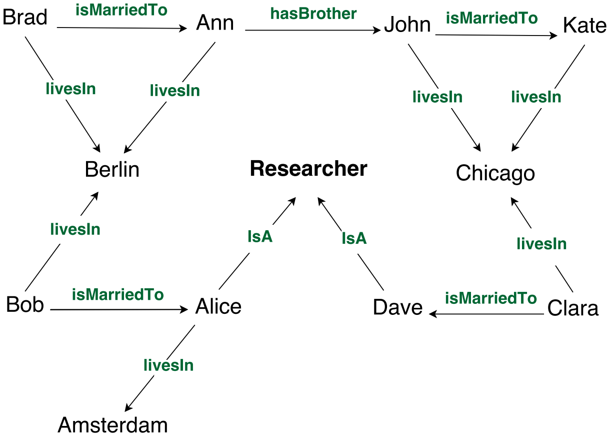
- **Propositionalization** [Krogel *et al.*, 2003]: transform a KG into a transaction database

	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)

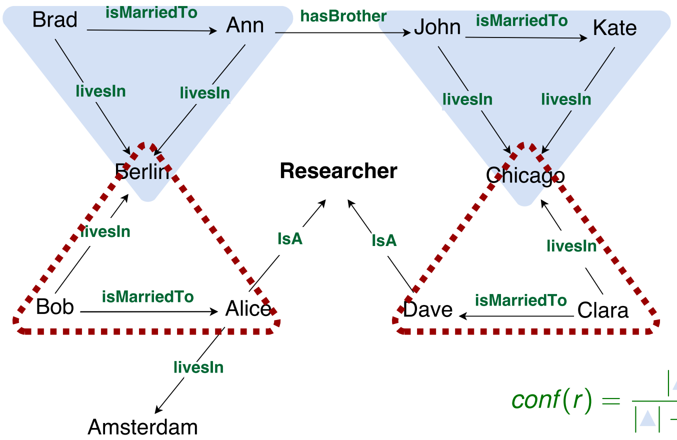
Relational Association Rule Learning



Relational Association Rule Learning

Confidence, e.g., WARMER [Goethals and den Bussche, 2002], AnyBurl [Meilicke *et al.*, 2019]

Closed World Assumption (CWA): Whatever is missing is false



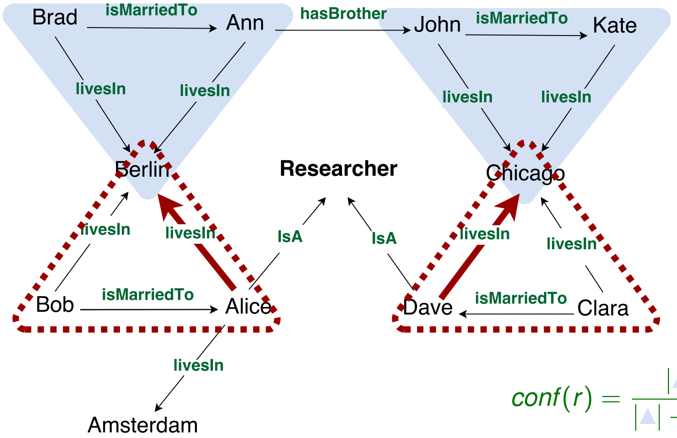
$$conf(r) = \frac{|\blacktriangle|}{|\blacktriangle| + |\triangle|} = \frac{2}{4}$$

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

Relational Association Rule Learning

Confidence, e.g., WARMER [Goethals and den Bussche, 2002], AnyBurl [Meilicke *et al.*, 2019]

Closed World Assumption (CWA): Whatever is missing is false



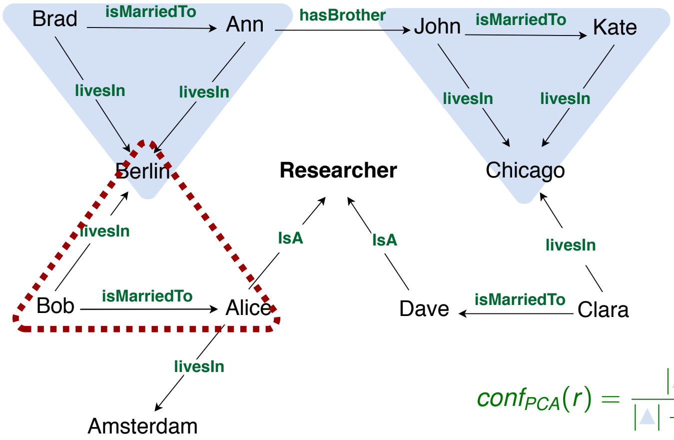
$$conf(r) = \frac{|\triangle|}{|\triangle| + |\triangle|} = \frac{2}{4}$$

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

Relational Association Rule Learning

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

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

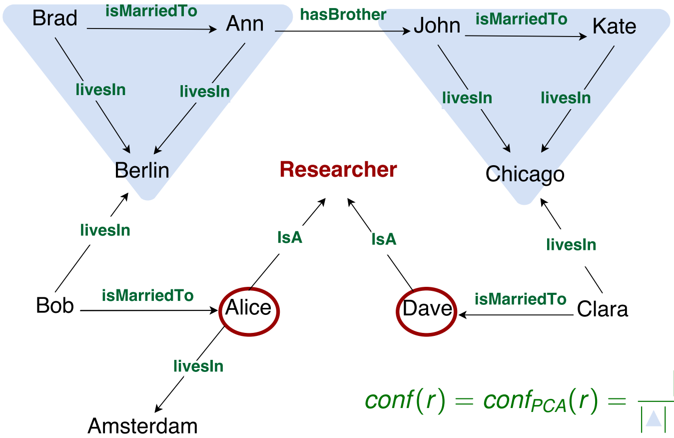


$$conf_{PCA}(r) = \frac{|\blacktriangle|}{|\blacktriangle| + |\triangle|} = \frac{2}{3}$$

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

Relational Association Rule Learning

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



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

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

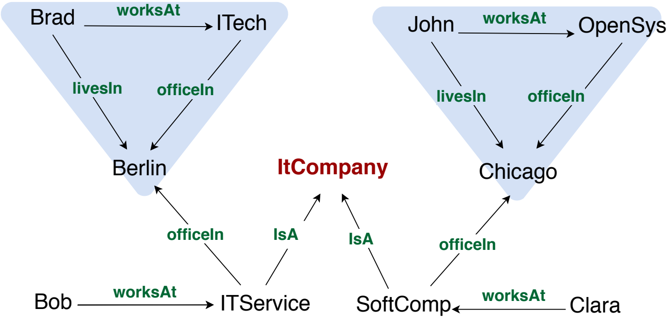
M. Gad-Elrab, D. Stepanova, J. Urbani, G. Weikum. Exception-enriched Rule Learning from Knowledge Graphs. ISWC2016

D. Tran, D. Stepanova, M. Gad-Elrab, F. Lisi, G. Weikum. Towards Nonmonotonic Relational Learning from KGs. ILP2016

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

Absurd Rules due to Data Incompleteness

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

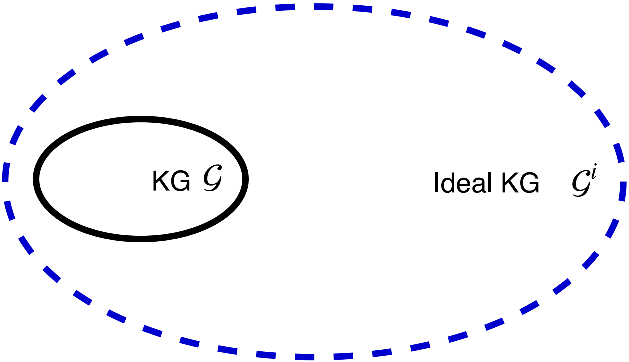


$$conf(r) = conf_{PCA}(r) = 1$$

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

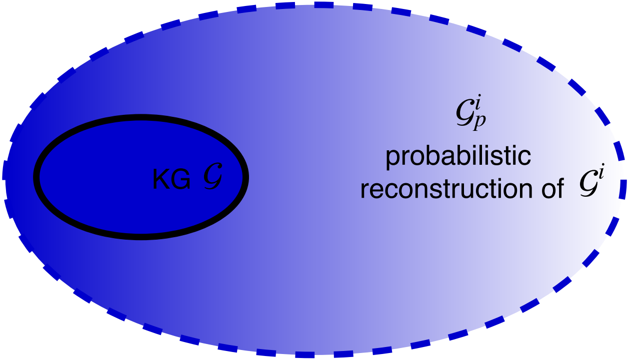
Ideal KG

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

Knowledge Graph Embeddings

- Map entities $\mathcal{E}$ and relations $\mathcal{R}$ to some vector space
  - $emb_E : \mathcal{E} \rightarrow \mathbb{R}^{d_E}$ and $emb_R : \mathcal{R} \rightarrow \mathbb{R}^{d_R}$

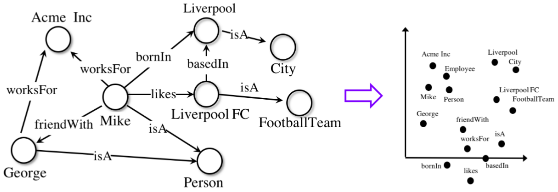


Figure from Ampligraph documentation

Knowledge Graph Embeddings

- Map entities $\mathcal{E}$ and relations $\mathcal{R}$ to some vector space
  - $emb_E : \mathcal{E} \rightarrow \mathbb{R}^{d_E}$ and $emb_R : \mathcal{R} \rightarrow \mathbb{R}^{d_R}$
- KG facts are scored using a dedicated scoring function:
  $score(h, r, t) = s(emb_E(h), emb_R(r), emb_E(t))$

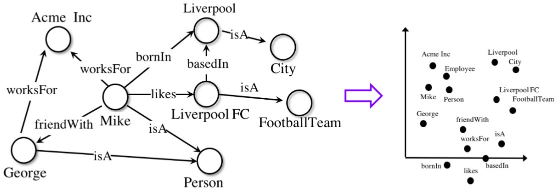


Figure from Ampligraph documentation

Knowledge Graph Embeddings

- Map entities $\mathcal{E}$ and relations $\mathcal{R}$ to some vector space
  - $emb_E : \mathcal{E} \rightarrow \mathbb{R}^{d_E}$ and $emb_R : \mathcal{R} \rightarrow \mathbb{R}^{d_R}$
- KG facts are scored using a dedicated scoring function: $score(h, r, t) = s(emb_E(h), emb_R(r), emb_E(t))$
- E.g., TransE [Bordes *et al.*, 2013]: $score(h, r, t) = -\|\mathbf{h} + \mathbf{r} - \mathbf{t}\|$ Text-enhanced variations exist, e.g., SSP [Xiao *et al.*, 2017]

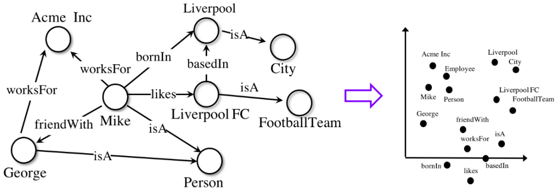


Figure from Ampligraph documentation

Knowledge Graph Embeddings

- Map entities $\mathcal{E}$ and relations $\mathcal{R}$ to some vector space
  - $emb_E : \mathcal{E} \to \mathbb{R}^{d_E}$ and $emb_R : \mathcal{R} \to \mathbb{R}^{d_R}$
- KG facts are scored using a dedicated scoring function: $score(h, r, t) = s(emb_E(h), emb_R(r), emb_E(t))$
- E.g., TransE [Bordes *et al.*, 2013]: $score(h, r, t) = -\|\mathbf{h} + \mathbf{r} - \mathbf{t}\|$ Text-enhanced variations exist, e.g., SSP [Xiao *et al.*, 2017]
- Ranking interpretation: If $score(h_1, r_1, t_1) > score(h_2, r_2, t_2)$, then (h_1, r_1, t_1) is more likely to hold than (h_2, r_2, t_2)

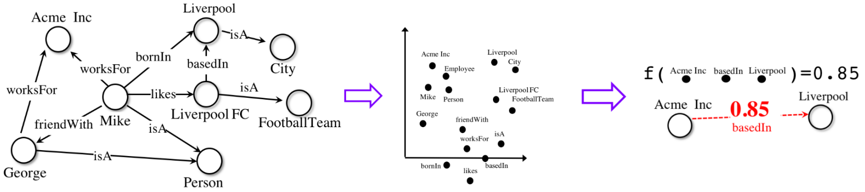
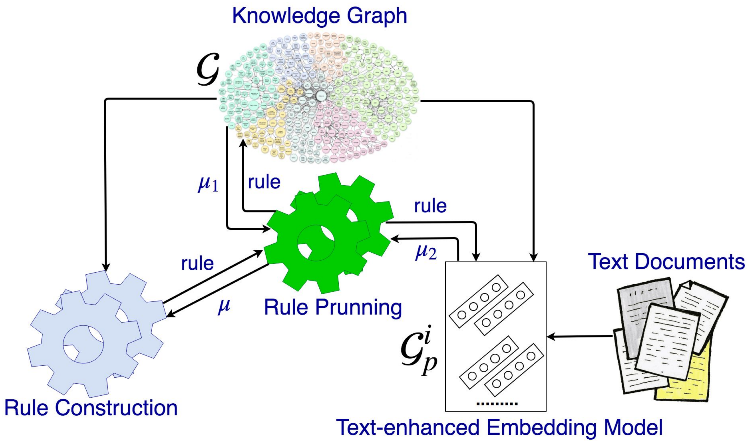


Figure from Ampligraph documentation

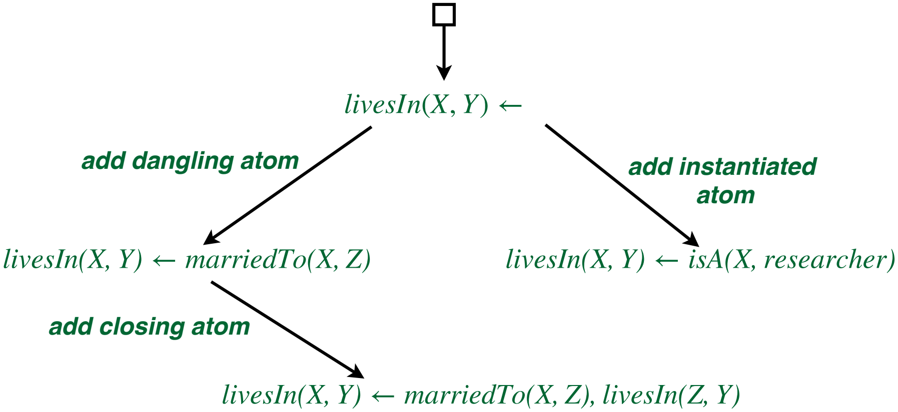
Embedding-based Rule Learning



V. Thinh Ho, D. Stepanova, M. Gad-Elrab, E. Kharlamov, G. Weikum. Rule Learning from KGs Guided by Embeddings. ISWC2018
`https://github.com/hovinhthinh/RuLES`

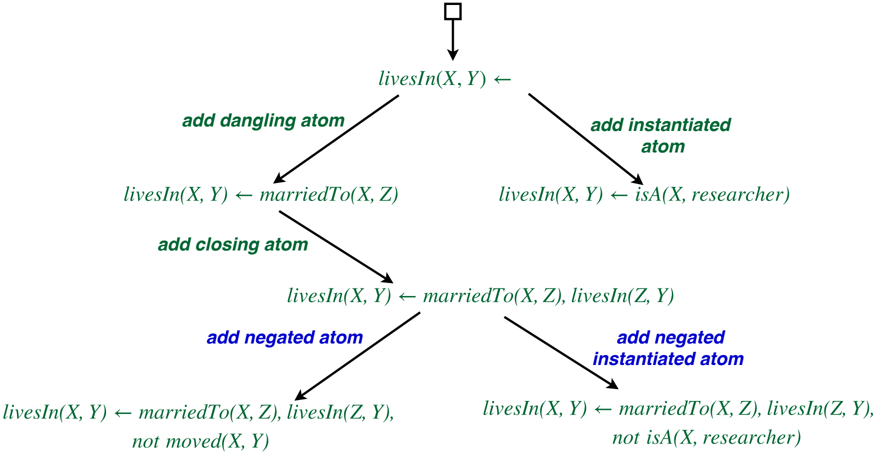
Rule Construction

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

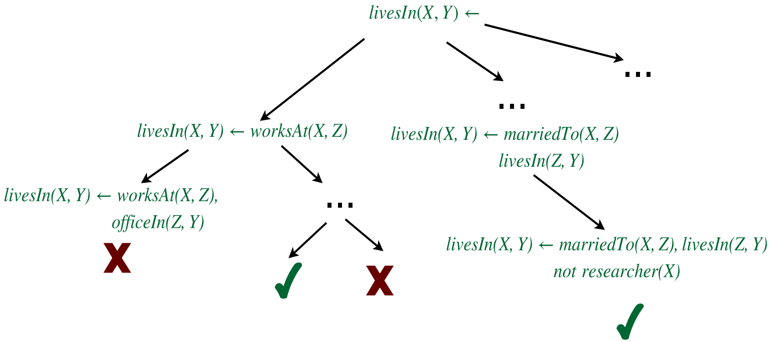


Rule Construction

- Clause exploration from general to specific
  - 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

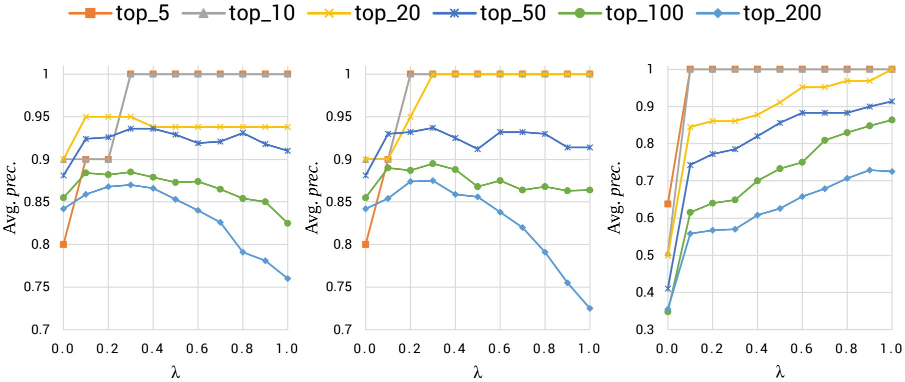
Evaluation Setup

- 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

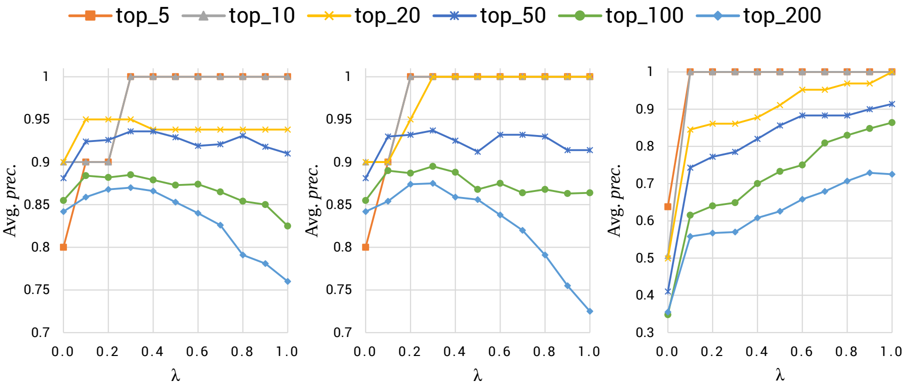
Evaluation of Hybrid Rule Measure



(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] ($\lambda = 0$)

Examples of Learned Rules

Rules learned from Wikidata and IMDB

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

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

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

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

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

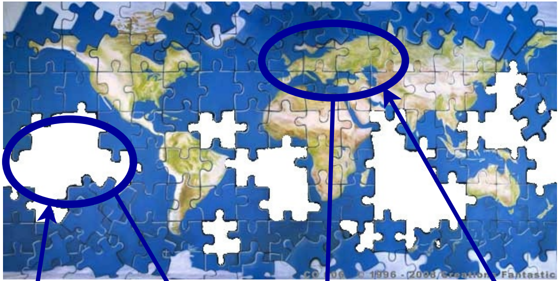
r_3 : $\mathit{actedIn}(X, Z) \leftarrow \mathit{isMarriedTo}(X, Y), \mathit{directed}(Y, Z), \mathit{not}\ \mathit{silent\_film\_actor}(X)$

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]
  - $\mathit{complete}(X, \mathit{hasChild}) \leftarrow \mathit{child}(X)$

Exploiting Cardinality Meta-data in Rule Learning

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



5 missing

do not build here!

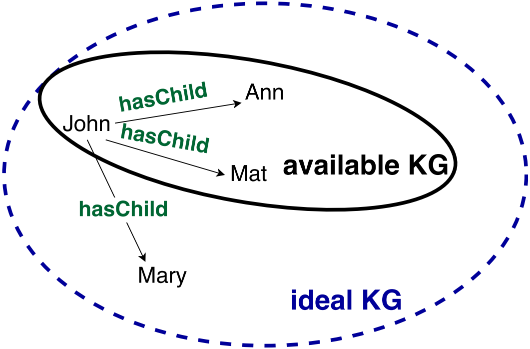
build here!

0 missing

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

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



$num(hasChild, john) = 3$
$miss(hasChild, john) = 1$
$incomplete(hasChild, john)$

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

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

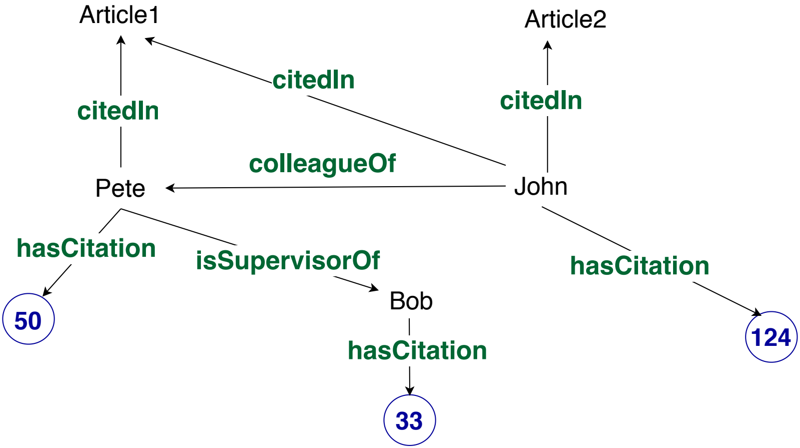
dir_metric : proportion of predictions in complete and incomplete parts

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

wdm : weighted combination of confidence and directional metric

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

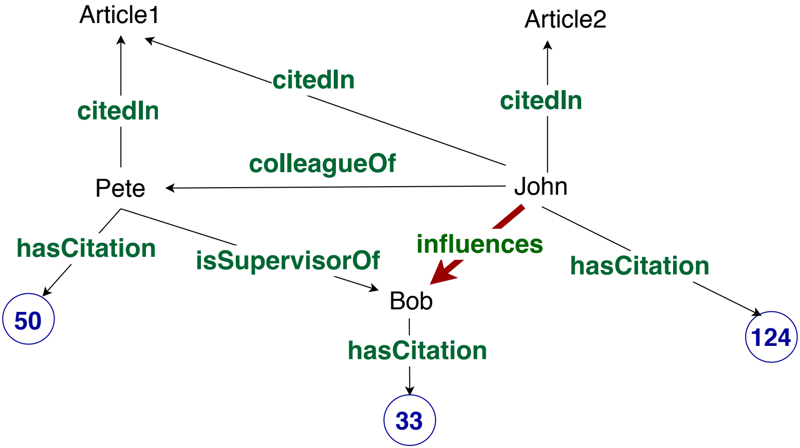
Numerical Rules



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

Numerical KG construction: *V. Thinh Ho, D. Stepanova, D. Milchevski, J. Strötgen, G. Weikum: Enhancing Knowledge Bases with Quantity Facts. WWW 2022*

Numerical Rules



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

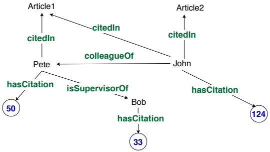
Numerical KG construction: *V. Thinh Ho, D. Stepanova, D. Milchevski, J. Strötgen, G. Weikum: Enhancing Knowledge Bases with Quantity Facts. WWW 2022*

NeuralLP

NeuralLP [Yang *et al.*, 2017]: Differentiable rule learning via (sparse) matrix-vector multiplication

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

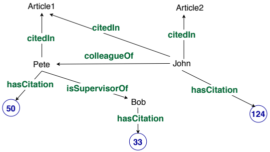
Columns: john, pete, bob, article1, article2



NeuralLP

NeuralLP [Yang *et al.*, 2017]: Differentiable rule learning via (sparse) matrix-vector multiplication

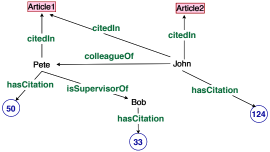
$$M_{\text{citedIn}} = \begin{array}{c} \begin{array}{ccccc} \text{john} & \text{pete} & \text{bob} & \text{article1} & \text{article2} \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array} \begin{array}{l}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



NeuralLP

NeuralLP [Yang *et al.*, 2017]: Differentiable rule learning via (sparse) matrix-vector multiplication

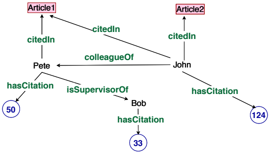
$$M_{\mathrm{citedIn}} = \begin{array}{c} \begin{array}{ccccc} \text{john} & \text{pete} & \text{bob} & \text{article1} & \text{article2} \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array} \begin{array}{l}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



NeuralLP

NeuralLP [Yang *et al.*, 2017]: Differentiable rule learning via (sparse) matrix-vector multiplication

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



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

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

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

Numerical Rule Learning

$$p = \mathit{hasCitation} \qquad f = \begin{array}{ccccc} \mathit{john} & \mathit{pete} & \mathit{bob} & \mathit{article}1 & \mathit{article}2 \\ \end{array}$$
$$\begin{bmatrix} 124 & 50 & 33 & \mathtt{NaN} & \mathtt{NaN} \end{bmatrix}$$

Numerical Rule Learning

$$p = hasCitation \qquad f = \begin{array}{ccccc} john & pete & bob & article1 & article2 \\ [124 & 50 & 33 & \mathtt{NaN} & \mathtt{NaN}] \end{array}$$

Comparison matrix:

$$M_{r_p^{\leq}} = \begin{array}{c} \begin{array}{ccccc} 124 & 50 & 33 & \mathtt{NaN} & \mathtt{NaN} \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array} \begin{array}{l} \\ 124 \\ 50 \\ 33 \\ \mathtt{NaN} \\ \mathtt{NaN} \end{array}$$

Numerical Rule Learning

$$p = hasCitation \qquad f = \begin{array}{ccccc} john & pete & bob & article1 & article2 \\ [124 & 50 & 33 & \texttt{NaN} & \texttt{NaN}] \end{array}$$

Comparison matrix:

$$M_{r_p^{\leq}} = \begin{array}{c|ccccc|c} & 124 & 50 & 33 & \texttt{NaN} & \texttt{NaN} & \\ \hline & 1 & 0 & 0 & 0 & 0 & 124 \\ & 1 & 1 & 0 & 0 & 0 & 50 \\ & 1 & 1 & 1 & 0 & 0 & 33 \\ & 0 & 0 & 0 & 0 & 0 & \texttt{NaN} \\ & 0 & 0 & 0 & 0 & 0 & \texttt{NaN} \end{array}$$

Problem: may be a dense matrix $\Rightarrow$ cannot be treated efficiently

Numerical Rule Learning

$$p = hasCitation \qquad f = \begin{array}{ccccc} john & pete & bob & article1 & article2 \\ [124 & 50 & 33 & \mathtt{NaN} & \mathtt{NaN}\ ] \end{array}$$

Comparison matrix:

$$M_{r_p^{\leq}} = \begin{array}{c} \begin{array}{ccccc} 124 & 50 & 33 & \mathtt{NaN} & \mathtt{NaN} \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array} \begin{array}{l} 124 \\ 50 \\ 33 \\ \mathtt{NaN} \\ \mathtt{NaN} \end{array}$$

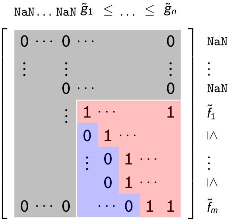
Problem: may be a dense matrix $\Rightarrow$ cannot be treated efficiently
Trick: Sort values by permutation matrices to allow for efficient computation

Numerical Rule Learning

$$p = \mathit{hasCitation} \qquad f = \begin{array}{cccccc} & \mathit{john} & \mathit{pete} & \mathit{bob} & \mathit{article}1 & \mathit{article}2 \\ & [124 & 50 & 33 & \mathtt{NaN} & \mathtt{NaN}\ ] \end{array}$$

Comparison matrix:

$$M_{r_p^{\leq}} = \begin{array}{c} \begin{array}{ccccc} 124 & 50 & 33 & \mathtt{NaN} & \mathtt{NaN} \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array} \begin{array}{l} \\ 124 \\ 50 \\ 33 \\ \mathtt{NaN} \\ \mathtt{NaN} \end{array}$$



Problem: may be a dense matrix ⇒ cannot be treated efficiently
Trick: Sort values by permutation matrices to allow for efficient computation

P. Wang, D. Stepanova, C. Domokos, Z. Kolter. Differentiable Learning of Numerical Rules from KGs. ICLR 2020

Explainable Entity Clustering in KGs

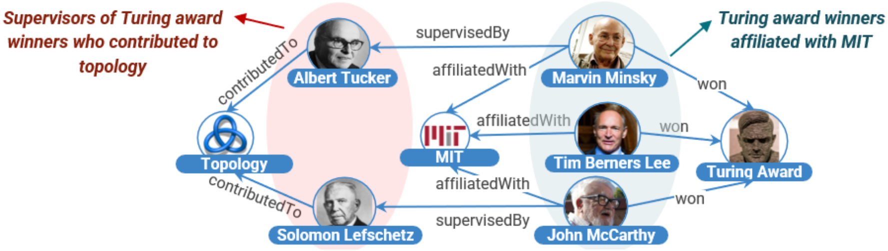
- Knowledge graphs are large and hard to explore

Explainable Entity Clustering in KGs

- Knowledge graphs are large and hard to explore
- Explainable clustering is a useful tool for KG summarization

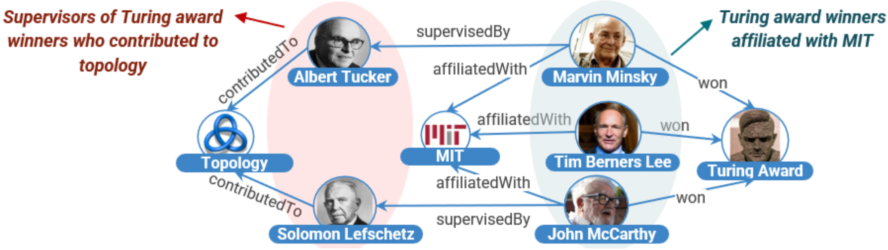
Explainable Entity Clustering in KGs

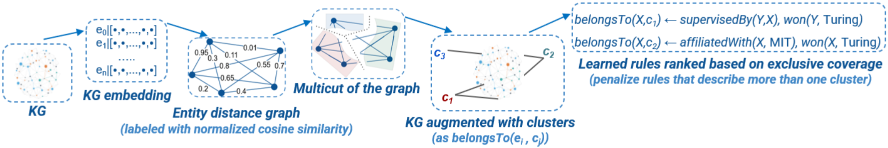
- Knowledge graphs are large and hard to explore
- Explainable clustering is a useful tool for KG summarization



Explainable Entity Clustering in KGs

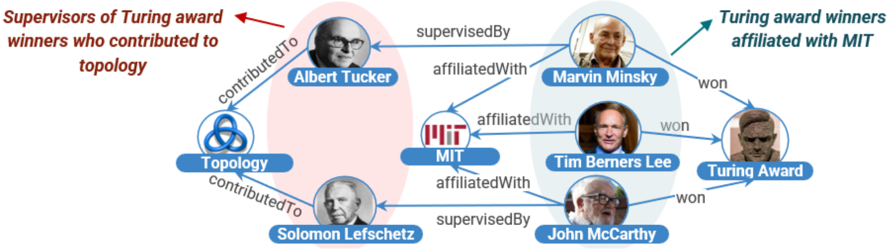
- Knowledge graphs are large and hard to explore
- Explainable clustering is a useful tool for KG summarization

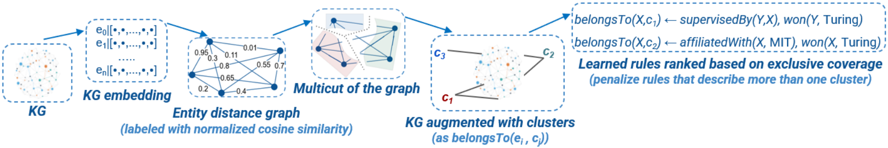




Explainable Entity Clustering in KGs

- Knowledge graphs are large and hard to explore
- Explainable clustering is a useful tool for KG summarization





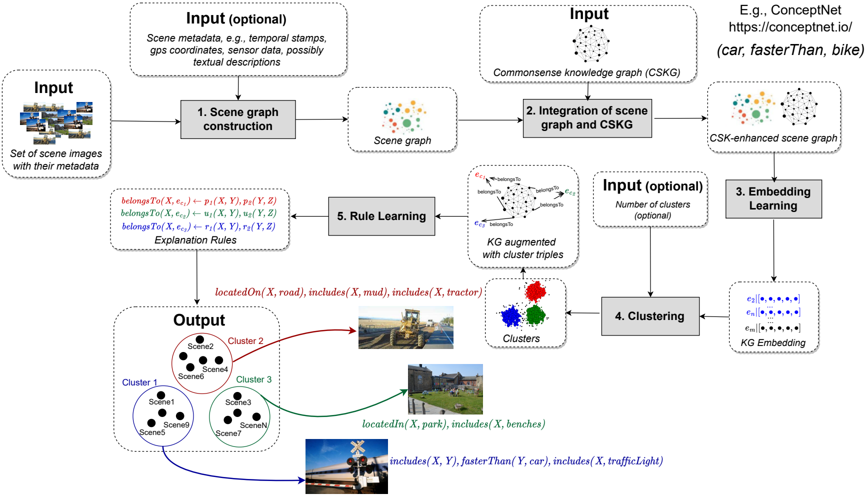
M. Gad-Elrab, D. Stepanova, T. Kien Trung, H. Adel, G. Weikum: Explainable Embedding-based Clustering in KGs. ISWC 2020.
M. H. Gad-Elrab, V. Thinh Ho, E. Levinkov, T. Kien Tran, D. Stepanova: Towards Utilizing Knowledge Graph Embedding Models for Conceptual Clustering. ISWC (Demos/Industry) 2020
`https://github.com/mhmgad/ExCut`

Explainable Scene Clustering

- Scene graphs, e.g., in autonomous driving domain [Wickramarachchi *et al.*, 2020]

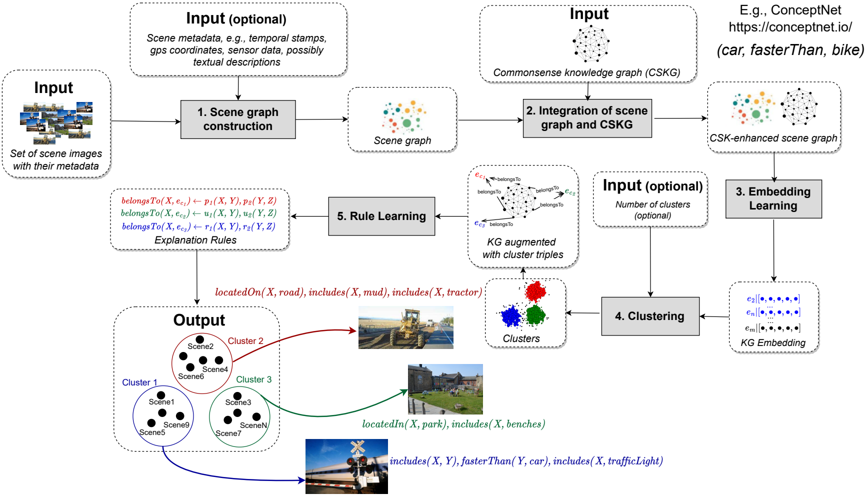
Explainable Scene Clustering

- Scene graphs, e.g., in autonomous driving domain [Wickramarachchi *et al.*, 2020]



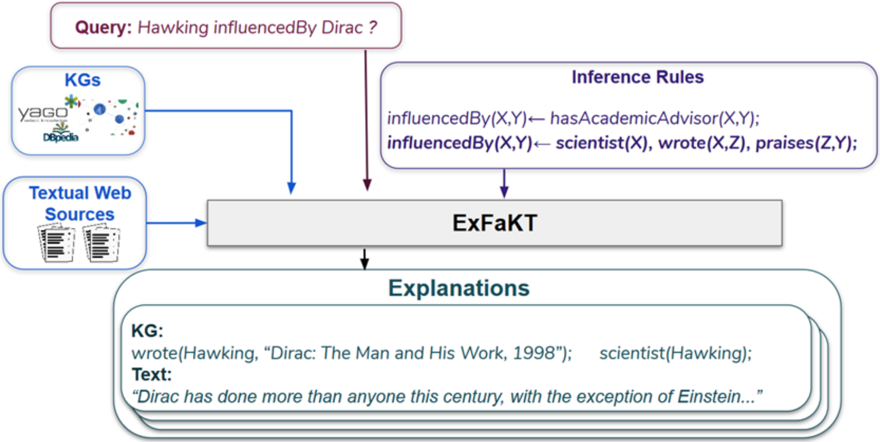
Explainable Scene Clustering

- Scene graphs, e.g., in autonomous driving domain [Wickramarachchi *et al.*, 2020]



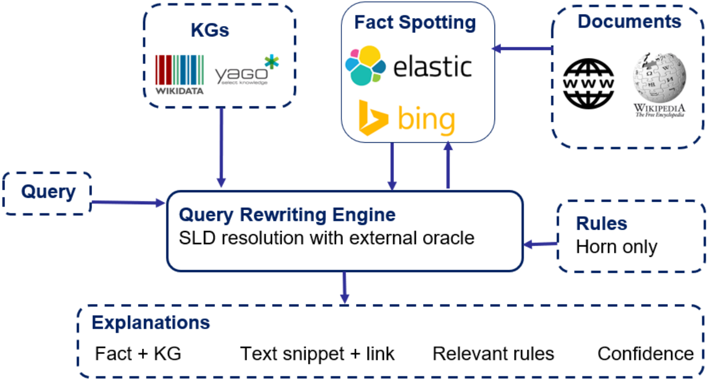
S. Nag Chowdhury, R. Wickramarachchi, M. H. Gad-Elrab, D. Stepanova, C. A. Henson: Towards Leveraging Commonsense Knowledge for Autonomous Driving. ISWC (Posters/Demos/Industry) 2021

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.
`https://github.com/mhmgad/ExFaKT`

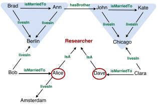
Rule-based Fact Checking

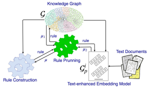


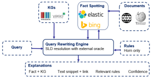
M. Gad-Elrab, D. Stepanova, J. Urbani, G. Weikum. ExFakt: A Framework for Explaining Facts over KGs and Text. WSDM 2019.
M. Gad-Elrab, D. Stepanova, J. Urbani, G. Weikum. Tracy: Tracing Facts over Knowledge Graphs and Text. WWW 2019.
`https://github.com/mhmgad/ExFaKT`

Summary

- Rule learning is useful for KG completion, reasoning, summarization, analytics
- Rule learning can be combined with KG embeddings
- Rules can be learned from KGs that contain numerical values
- Applications:
    - Explainable clustering
    - Rule-based fact checking







Outlook

- Learning rules from text or labeled images
- Make use of rules for explaining ML models
- Rules for KG cleaning...

For internship opportunities please contact me at daria.stepanova@de.bosch.com

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 sharing the slides:
  - Thomas Eiter, Stephen Muggleton, Luc De Raedt, Fabian Suchanek, Rainer Gemulla
- For providing great 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.

Christian Meilicke, Melisachew Wudage Chekol, Daniel Ruffinelli, and Heiner Stuckenschmidt.
Anytime bottom-up rule learning for knowledge graph completion.
In Sarit Kraus, editor, *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 3137–3143. ijcai.org, 2019.

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.

References III

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.

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

Ruwan Wickramarachchi, Cory A. Henson, and Amit P. Sheth.
An evaluation of knowledge graph embeddings for autonomous driving data: Experience and practice.
In *AAAI 2020 Spring Symposium on Combining Machine Learning and Knowledge Engineering in Practice*, CEUR Workshop Proceedings, 2020.

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.