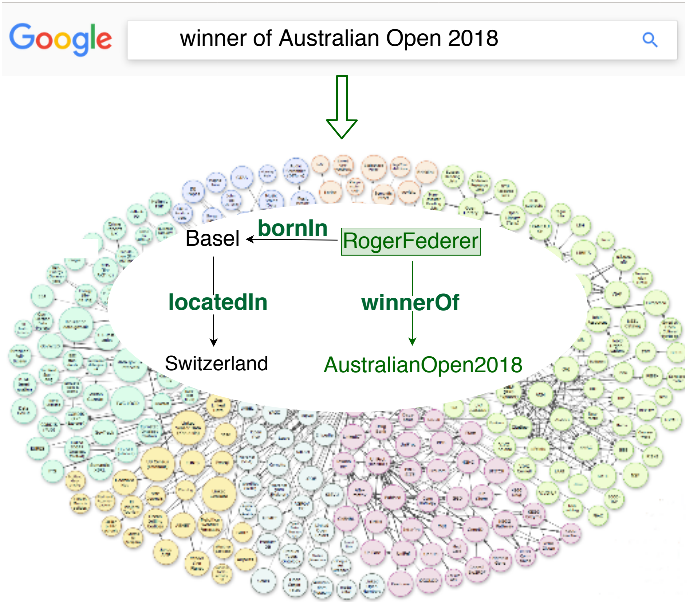
Differentiable learning of numerical rules in knowledge graphs

Po-Wei Wang[1,2], Daria Stepanova[1], Csaba Domokos[1] and Zico Kolter[1,2]

[1]Bosch Center for Artificial Intelligence
[2]Carnegie Mellon University

ICLR'20

BOSCH



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

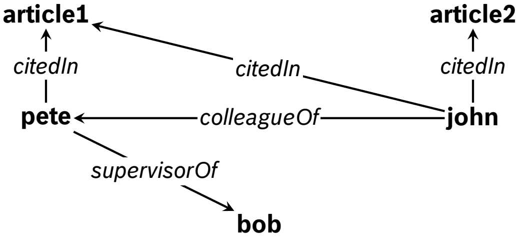
Spouse: Mirka Federer (m. 2009)

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

Knowledge graph = Multi-graph with typed edges

Entities (nodes): **article1**, **article2**, **pete**, **john**, **bob**

Facts (edges): citedIn(pete, article1), supervisorOf(pete, john)



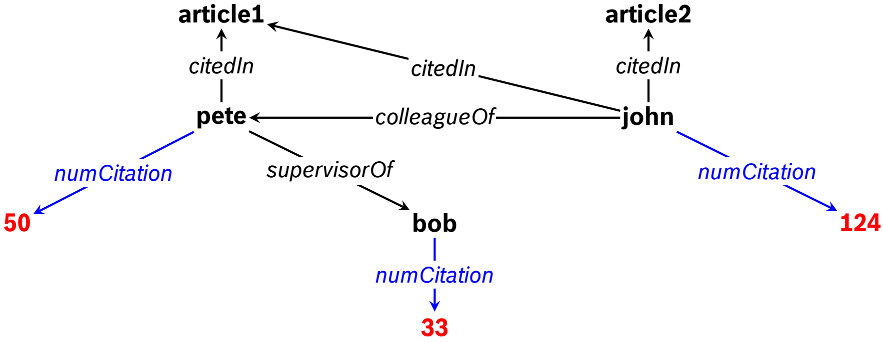
Public | Po-Wei Wang, Daria Stepanova, Csaba Domokos, Zico Kolter | 2020-04-08

Knowledge graph = Multi-graph with typed edges

Entities (nodes): **article1**, **article2**, **pete**, **john**, **bob**

Facts (edges): citedIn(pete, article1), supervisorOf(pete, john)

Numerical Facts: numCitation(pete, 50), numCitation(john, 124)



Public | Po-Wei Wang, Daria Stepanova, Csaba Domokos, Zico Kolter | 2020-04-08

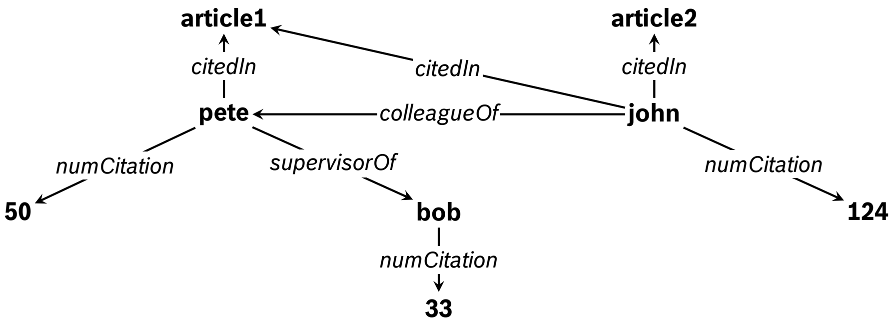
Goal: Learn (numerical) rules from KG and complete missing edges

Rule: pattern matching along a certain path

Goal: Learn (numerical) rules from KG and complete missing edges

Rule: pattern matching along a certain path

influences(**X**, **Z**) ← colleagueOf(**X**, **Y**) ∧ supervisorOf(**Y**, **Z**)

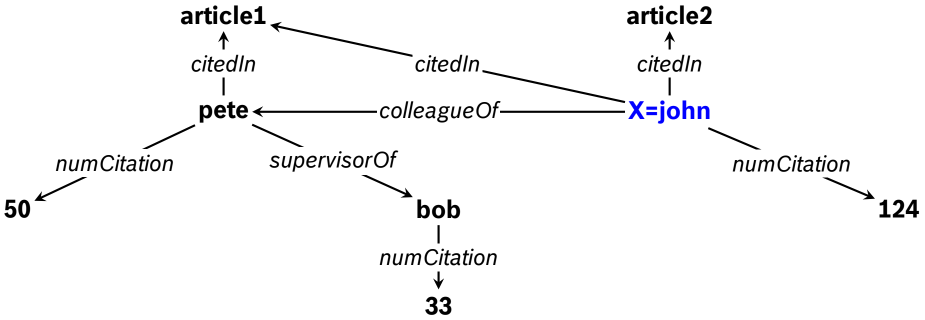


Public | Po-Wei Wang, Daria Stepanova, Csaba Domokos, Zico Kolter | 2020-04-08

Goal: Learn (numerical) rules from KG and complete missing edges

Rule: pattern matching along a certain path

influences(**X**, **Z**) ← colleagueOf(**X**, **Y**) ∧ supervisorOf(**Y**, **Z**)

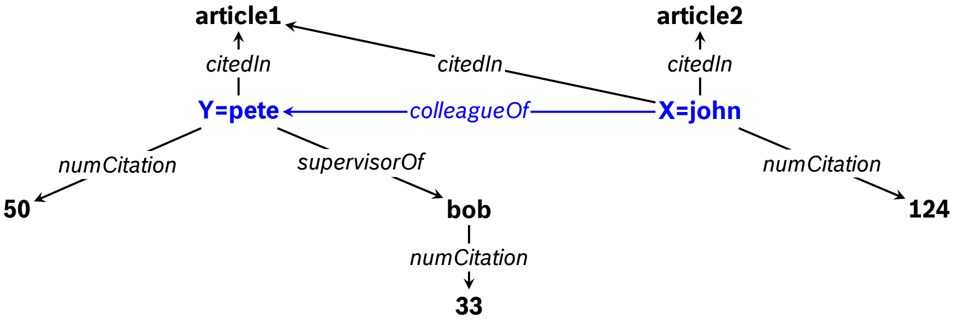


Public | Po-Wei Wang, Daria Stepanova, Csaba Domokos, Zico Kolter | 2020-04-08

Goal: Learn (numerical) rules from KG and complete missing edges

Rule: pattern matching along a certain path

influences(**X**, **Z**) ← colleagueOf(**X**, **Y**) ∧ supervisorOf(**Y**, **Z**)

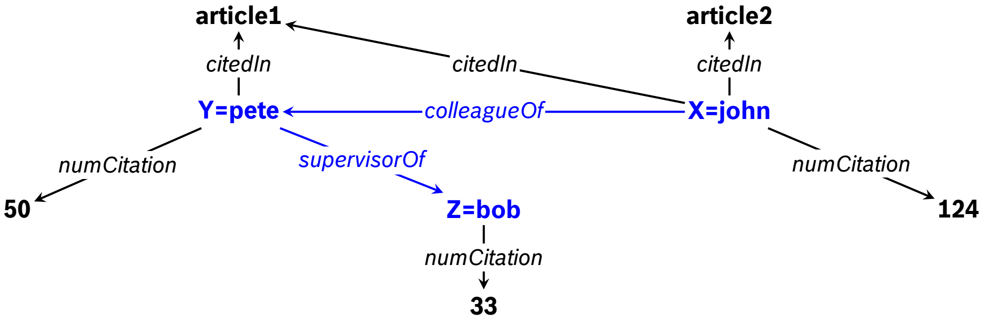


Public | Po-Wei Wang, Daria Stepanova, Csaba Domokos, Zico Kolter | 2020-04-08

Goal: Learn (numerical) rules from KG and complete missing edges

Rule: pattern matching along a certain path

influences(**X**, **Z**) $\leftarrow$ colleagueOf(**X**, **Y**) $\wedge$ supervisorOf(**Y**, **Z**)

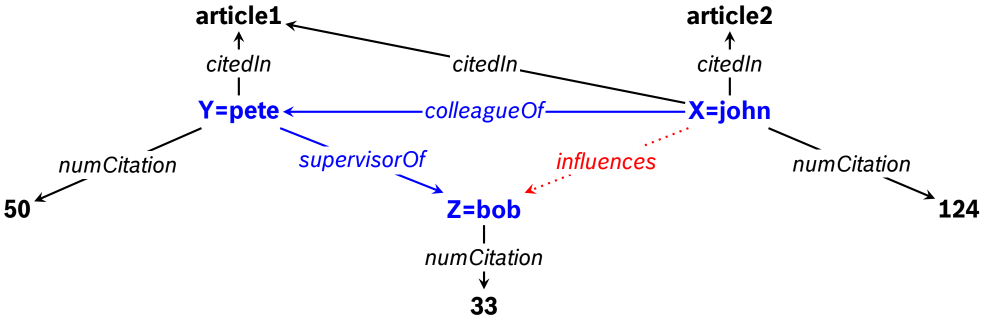


Public | Po-Wei Wang, Daria Stepanova, Csaba Domokos, Zico Kolter | 2020-04-08

Goal: Learn (numerical) rules from KG and complete missing edges

Rule: pattern matching along a certain path

influences(**X**, **Z**) ← colleagueOf(**X**, **Y**) ∧ supervisorOf(**Y**, **Z**)



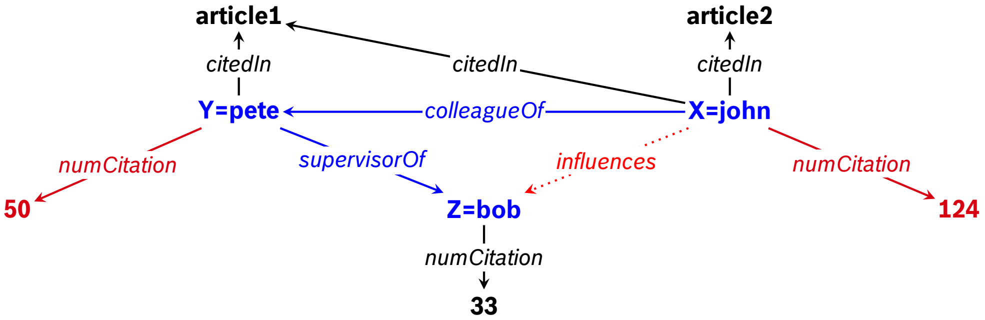
Public | Po-Wei Wang, Daria Stepanova, Csaba Domokos, Zico Kolter | 2020-04-08

Goal: Learn (numerical) rules from KG and complete missing edges

Rule: pattern matching along a certain path

Numerical rule: Comparison / classification operator using features along the path

influences(**X**, **Z**) ← colleagueOf(**X**, **Y**) ∧ supervisorOf(**Y**, **Z**) ∧ (**X**.numCitation > **Y**.numCitation)



Public | Po-Wei Wang, Daria Stepanova, Csaba Domokos, Zico Kolter | 2020-04-08

Problem: Implementing the (numerical) rule matching

`NeuralLP`: differentiable learning framework via (sparse) matrix-vector multiplication

Public | Po-Wei Wang, Daria Stepanova, Csaba Domokos, Zico Kolter | 2020-04-08

Problem: Implementing the (numerical) rule matching

`NeuralLP`: differentiable learning framework via (sparse) matrix-vector multiplication

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

Public | Po-Wei Wang, Daria Stepanova, Csaba Domokos, Zico Kolter | 2020-04-08

Problem: Implementing the (numerical) rule matching

`NeuralLP`: 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

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

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

Public | Po-Wei Wang, Daria Stepanova, Csaba Domokos, Zico Kolter | 2020-04-08

Problem: Implementing the (numerical) rule matching

`NeuralLP`: differentiable learning framework via (sparse) matrix-vector multiplication

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

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

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

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

For numerical rules, we can similarly create the comparison matrix

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

Public | Po-Wei Wang, Daria Stepanova, Csaba Domokos, Zico Kolter | 2020-04-08

Problem: Implementing the (numerical) rule matching

`NeuralLP`: 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

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

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

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

Public | Po-Wei Wang, Daria Stepanova, Csaba Domokos, Zico Kolter | 2020-04-08

Contribution: Efficient matrix-vector mult for numerical operators

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

Public | Po-Wei Wang, Daria Stepanova, Csaba Domokos, Zico Kolter | 2020-04-08

Contribution: Efficient matrix-vector mult for numerical operators

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

$$\texttt{NaN} \ldots \texttt{NaN}\ \tilde{g}_1 \le \ldots \le \tilde{g}_n$$

$$\tilde{M}_{r^{\le}_{pq}} = \begin{bmatrix} 0 & \cdots & 0 & \cdots & & & & 0 \\ \vdots & & \vdots & & & & & \vdots \\ & & 0 & \cdots & & & & 0 \\ & & \vdots & 1 & \cdots & & & 1 \\ & & & 0 & 1 & \cdots & & \\ & & & \vdots & 0 & 1 & \cdots & \\ & & & & 0 & 1 & \cdots & \\ 0 & \cdots & 0 & & \cdots & 0 & 1 & 1 \end{bmatrix} \begin{matrix} \texttt{NaN} \\ \vdots \\ \texttt{NaN} \\ \tilde{f}_1 \\ \le \\ \vdots \\ \le \\ \tilde{f}_m \end{matrix}$$

Monotonic borderline:

Public | Po-Wei Wang, Daria Stepanova, Csaba Domokos, Zico Kolter | 2020-04-08

Contribution: Efficient matrix-vector mult for numerical operators

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

$$\begin{array}{c} \\ \tilde{M}_{r^{\leq}_{pq}} = \end{array} \begin{array}{c} \begin{array}{ccccccc} \mathtt{NaN} \ldots \mathtt{NaN} & \tilde{g}_1 & \leq & \ldots & \leq & \tilde{g}_n \end{array} \\ \begin{bmatrix} 0 & \cdots & 0 & \cdots & & & 0 \\ \vdots & & \vdots & & & & \vdots \\ & & 0 & \cdots & & & 0 \\ & & \vdots & 1 & \cdots & & 1 \\ & & & 0 & 1 & \cdots & \\ & & & \vdots & 0 & 1 \cdots & \\ & & & & 0 & 1 \cdots & \\ 0 & \cdots & 0 & \cdots & 0 & 1 & 1 \end{bmatrix} \end{array} \begin{array}{c} \\ \mathtt{NaN} \\ \vdots \\ \mathtt{NaN} \\ \tilde{f}_1 \\ \leq \\ \vdots \\ \leq \\ \tilde{f}_m \end{array}$$

Monotonic borderline:

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

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

Public | Po-Wei Wang, Daria Stepanova, Csaba Domokos, Zico Kolter | 2020-04-08

BOSCH

Contribution: Efficient matrix-vector mult for numerical operators

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

$$\begin{array}{cccccccc} \texttt{NaN} & \ldots & \texttt{NaN} & \tilde{g}_1 & \le & \ldots & \le & \tilde{g}_n \end{array}$$

$$\tilde{M}_{r_{pq}^{\le}} = \left[\begin{array}{cccccccc} 0 & \cdots & 0 & \cdots & & & & 0 \\ \vdots & & \vdots & & & & & \vdots \\ & & 0 & \cdots & & & & 0 \\ & & \vdots & 1 & \cdots & & & 1 \\ & & & 0 & 1 & \cdots & & \\ & & & \vdots & 0 & 1 & \cdots & \\ & & & & 0 & 1 & \cdots & \\ 0 & \cdots & 0 & & \cdots & 0 & 1 & 1 \end{array}\right] \begin{array}{c} \texttt{NaN} \\ \vdots \\ \texttt{NaN} \\ \tilde{f}_1 \\ \rotatebox{90}{$\le$} \\ \vdots \\ \rotatebox{90}{$\le$} \\ \tilde{f}_m \end{array}$$

Monotonic borderline:

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

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

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

Public | Po-Wei Wang, Daria Stepanova, Csaba Domokos, Zico Kolter | 2020-04-08

Contribution: Efficient matrix-vector mult for numerical operators

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

$$\begin{array}{c} \texttt{NaN} \ldots \texttt{NaN}\ \tilde{g}_1 \leq \ldots \leq \tilde{g}_n \end{array}$$

$$\tilde{M}_{r_{pq}^{\leq}} = \begin{bmatrix} 0 & \cdots & 0 & \cdots & & & & 0 \\ \vdots & & \vdots & & & & & \vdots \\ & & 0 & \cdots & & & & 0 \\ & & \vdots & 1 & \cdots & & & 1 \\ & & & 0 & 1 & \cdots & & \\ & & & \vdots & 0 & 1 & \cdots & \\ & & & & 0 & 1 & \cdots & \\ 0 & \cdots & 0 & & \cdots & 0 & 1 & 1 \end{bmatrix} \begin{matrix} \texttt{NaN} \\ \vdots \\ \texttt{NaN} \\ \tilde{f}_1 \\ \leq \\ \vdots \\ \leq \\ \tilde{f}_m \end{matrix}$$

Monotonic borderline:

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

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

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

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

Public | Po-Wei Wang, Daria Stepanova, Csaba Domokos, Zico Kolter | 2020-04-08

Comparison to state-of-the-art rule learning methods

`Hit@10`: the number of correct head terms 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**

Public | Po-Wei Wang, Daria Stepanova, Csaba Domokos, Zico Kolter | 2020-04-08

Summary

Learning (numerical) rules in KGs by *path matching* with *matrix multiplications*

Summary

Learning (numerical) rules in KGs by *path matching* with *matrix multiplications*

Extension of the `NeuralLP` framework with

- numerical comparison
- classification (in paper)
- negation (in paper)

BOSCH

Summary

Learning (numerical) rules in KGs by *path matching* with *matrix multiplications*

Extension of the `NeuralLP` framework with

- numerical comparison
- classification (in paper)
- negation (in paper)

Improvement over the state-of-the-art rule learning methods

Summary

Learning (numerical) rules in KGs by *path matching* with *matrix multiplications*

Extension of the `NeuralLP` framework with

- numerical comparison
- classification (in paper)
- negation (in paper)

Improvement over the state-of-the-art rule learning methods

Thank you for your attention!