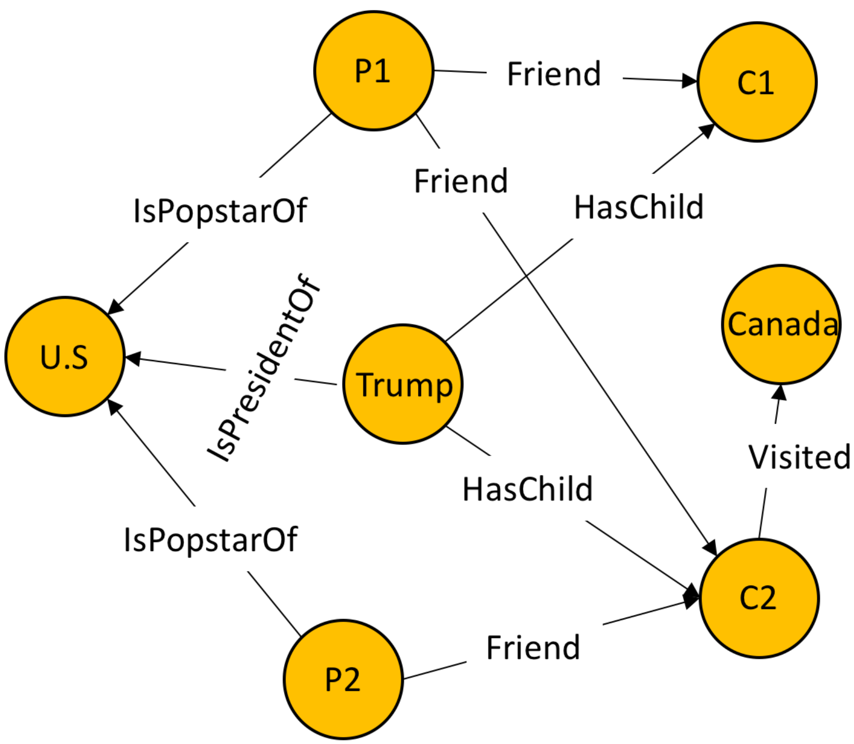
PAGE: Answering Pattern Queries via Knowledge Graph Embedding

S. Hong[°], N. Park[*], T. Chakraborty[^], H. Kang[·], S. Kwon[·]

University of Maryland College Park, USA[°]
University of North Carolina at Charlotte, USA[*]
Indraprastha Institute of Information Technology Delhi (IIIT-D), India[^]
Electronics and Telecommunications Research Institute, South Korea[·]

Graph Query Answering

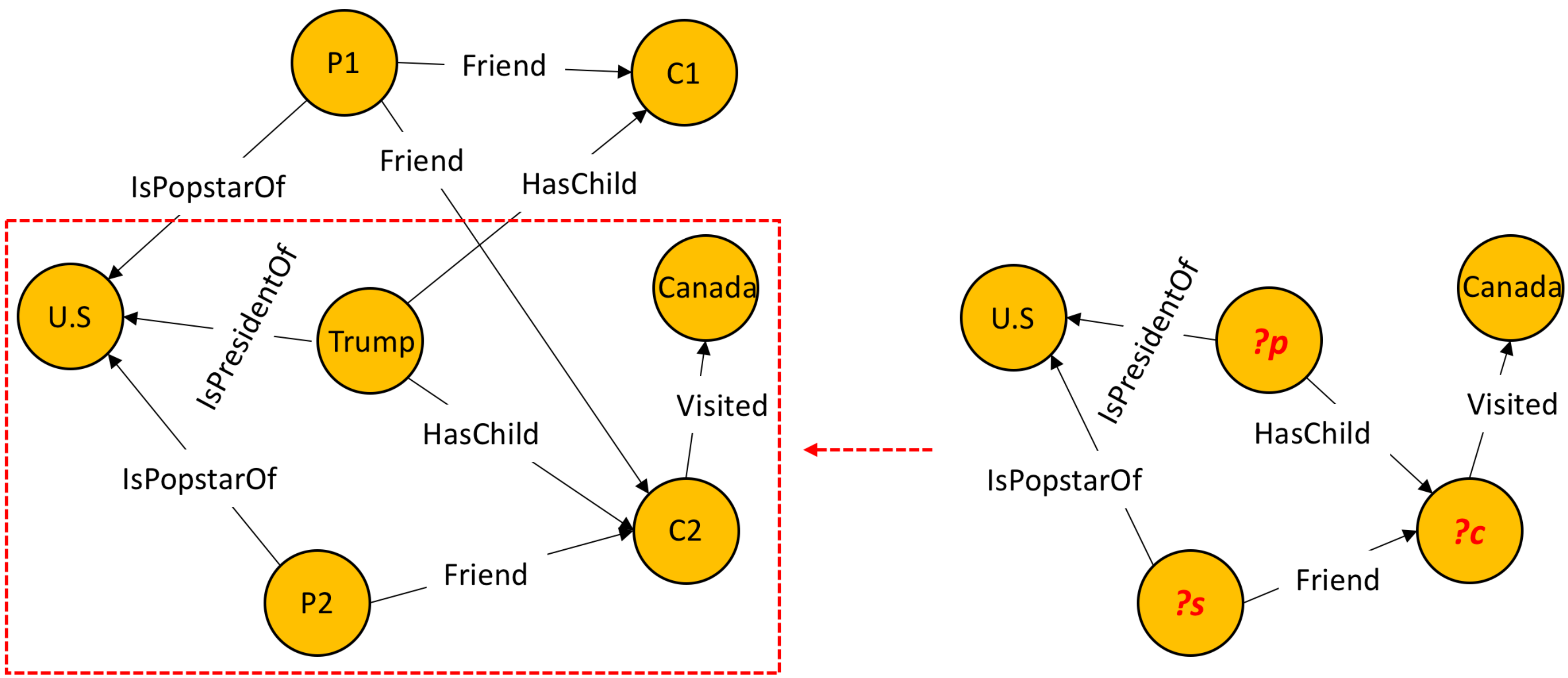


*a child **?c** of the president **?p** of the U.S,*
who had visited Canada before
*and is a friend of a pop star **?s***

A Knowledge Graph $\mathcal{G}$

A Graph Query Q

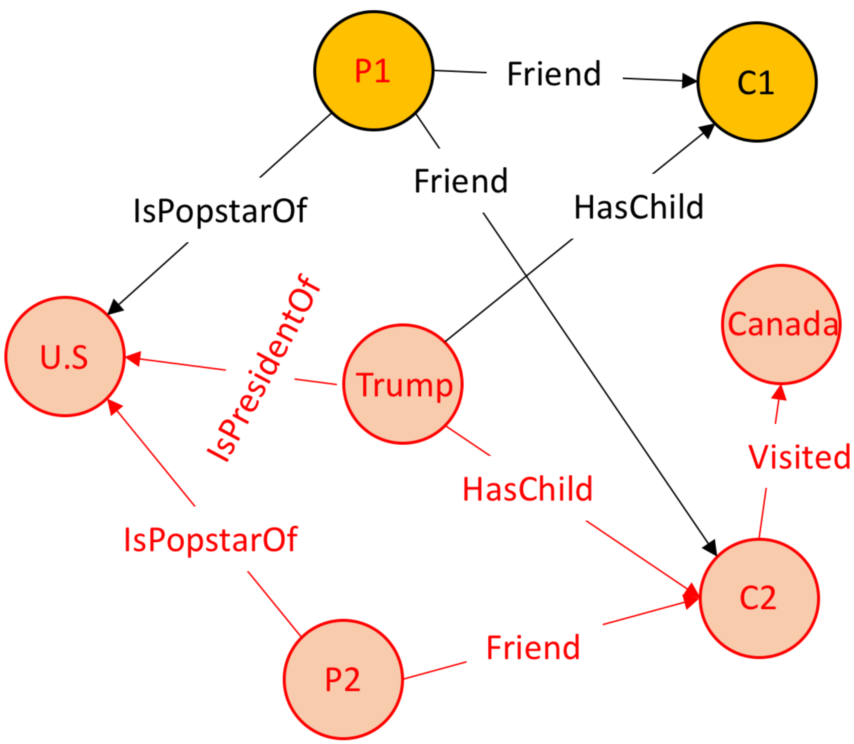
Graph Query Answering [cont]



A Knowledge Graph $\mathcal{G}$

A Graph Query Q

Graph Query Answering [cont]



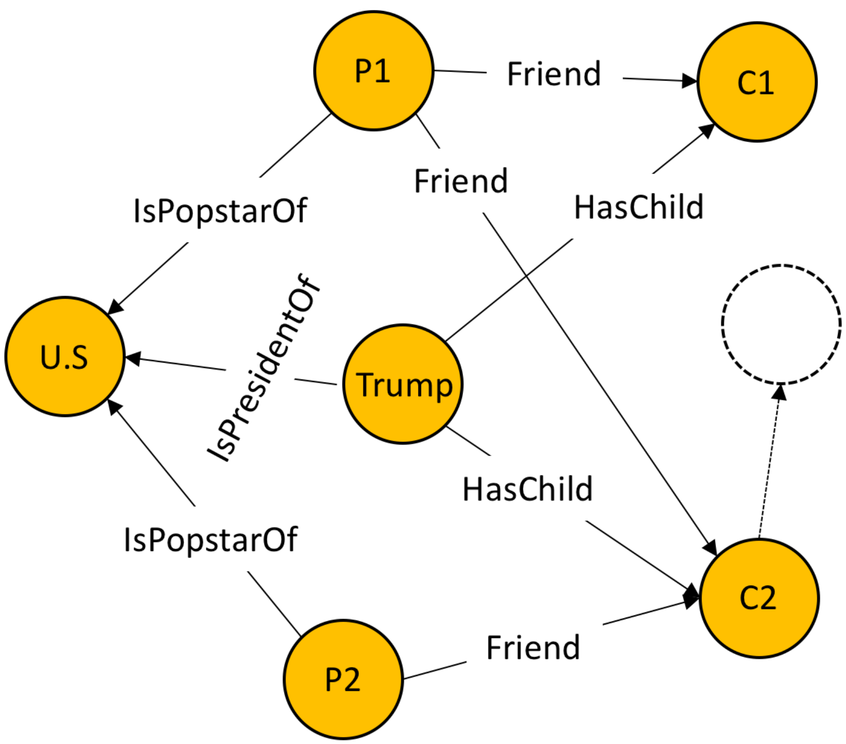
a child ***C2*** *of the president* ***Trump*** *of the U.S,*
who had visited Canada before
and is a friend of a pop star ***P2***

Compute answers via
subgraph isomorphism (matching)

A Knowledge Graph $\mathcal{G}$

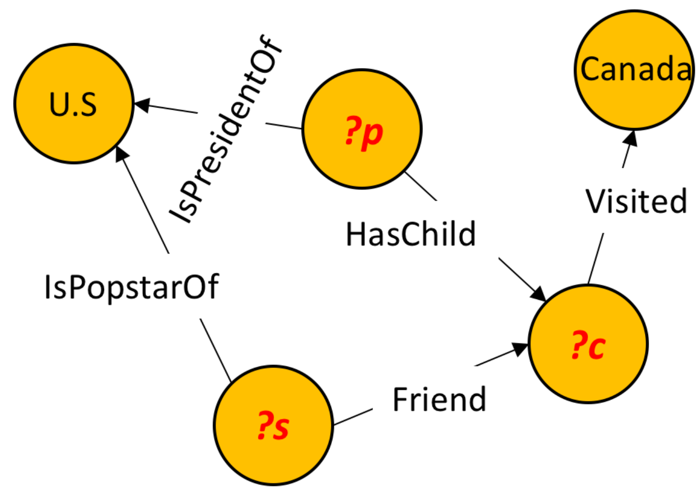
A Graph Query Q

Graph Query Answering: Problem



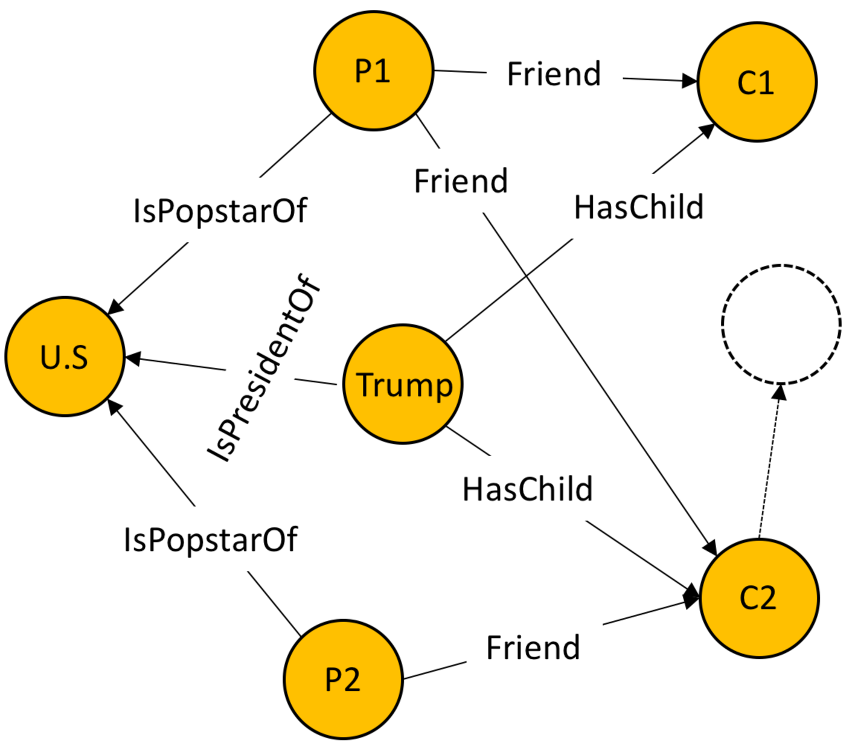
A Knowledge Graph $\mathcal{G}$

[What if ***Canada*** *is missing in $\mathcal{G}$]*



A Graph Query Q

Graph Query Answering: Problem [cont]



a child C1? or C2? of the president Trump of the U.S, who had visited Canada before and is a friend of a pop star P1? or P2?

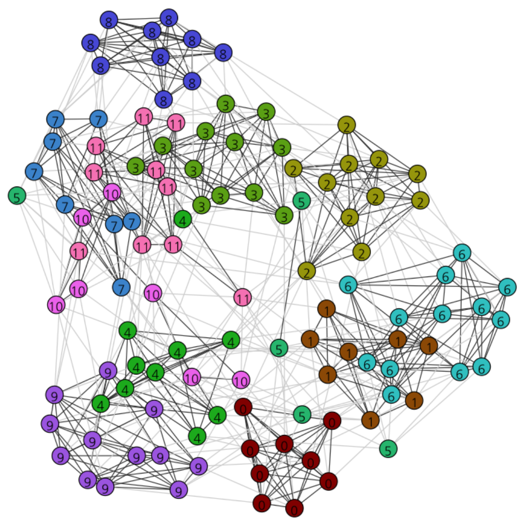
Hard to find exact answers

A Knowledge Graph $\mathcal{G}$

*[What if **Canada** is missing in $\mathcal{G}$]*

A Graph Query Q

Knowledge Graph Embedding



A Knowledge Graph $\mathcal{G}$

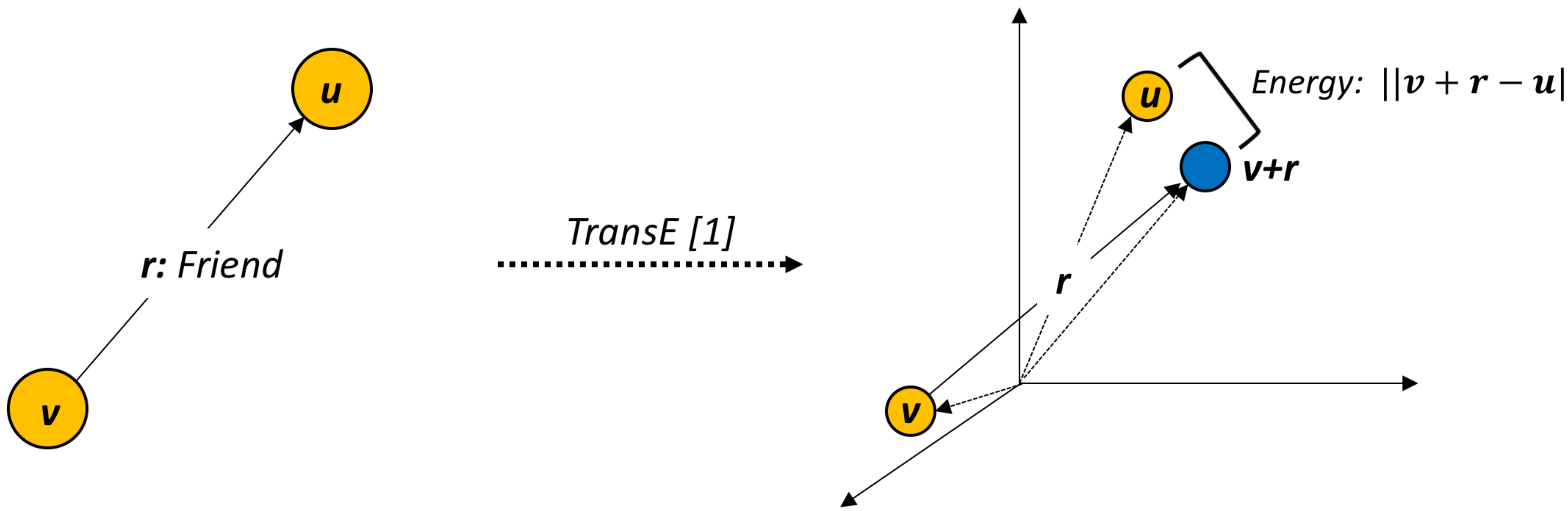
Learns

$$\begin{array}{cc} \begin{bmatrix} 0.11\ 0.5 & \cdots\cdots\cdots & 0.98 \\ & \vdots & \\ & \vdots & \\ & \vdots & \\ 0.42\ 0.7 & \cdots\cdots\cdots & -0.3 \end{bmatrix} & \textit{vertices \& relations} \\ \textit{features} & \end{array}$$

An Embedding (features) M

Knowledge Graph Embedding [cont]

The embedding learns vector representations of $\boldsymbol{v}, \boldsymbol{r}, \boldsymbol{u}$
such that $\boldsymbol{v}, \boldsymbol{r}, \boldsymbol{u}$ *minimizes the energy between* $\boldsymbol{v} + \boldsymbol{r}$ *and* $\boldsymbol{u}$.

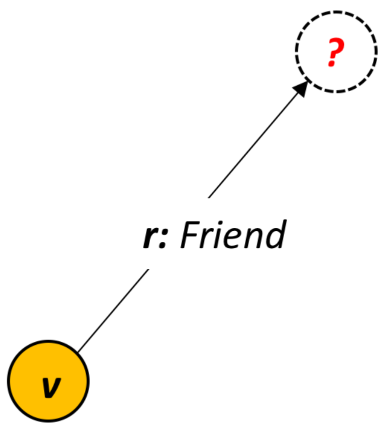


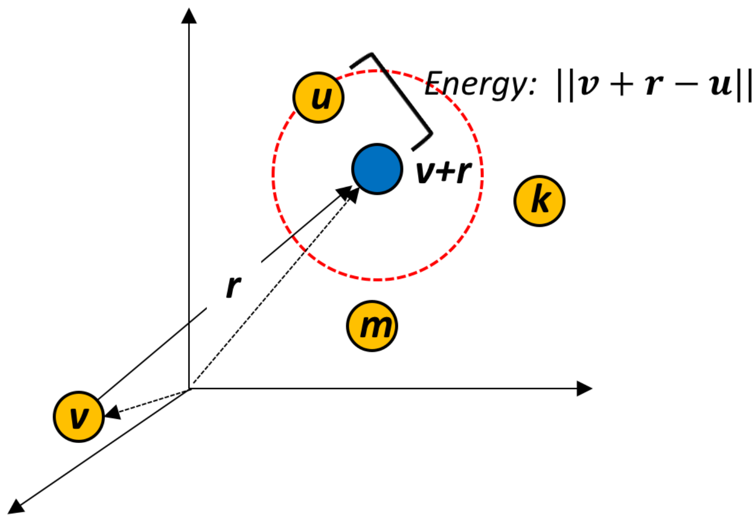
A Toy Knowledge Graph $\mathcal{G}$ *An Embedding in 3-D Space*

[1] Bordes, Antoine, et al. "Translating Embeddings for Modeling Multi-relational Data." NIPS'13

Knowledge Graph Embedding [cont]

*The embedding enables to **infer** the unknown vertex $\boldsymbol{u}$,*
connected to $\boldsymbol{v}$ via $\boldsymbol{r}$, by selecting the closest vertex around the vector $\boldsymbol{v}+\boldsymbol{r}$.

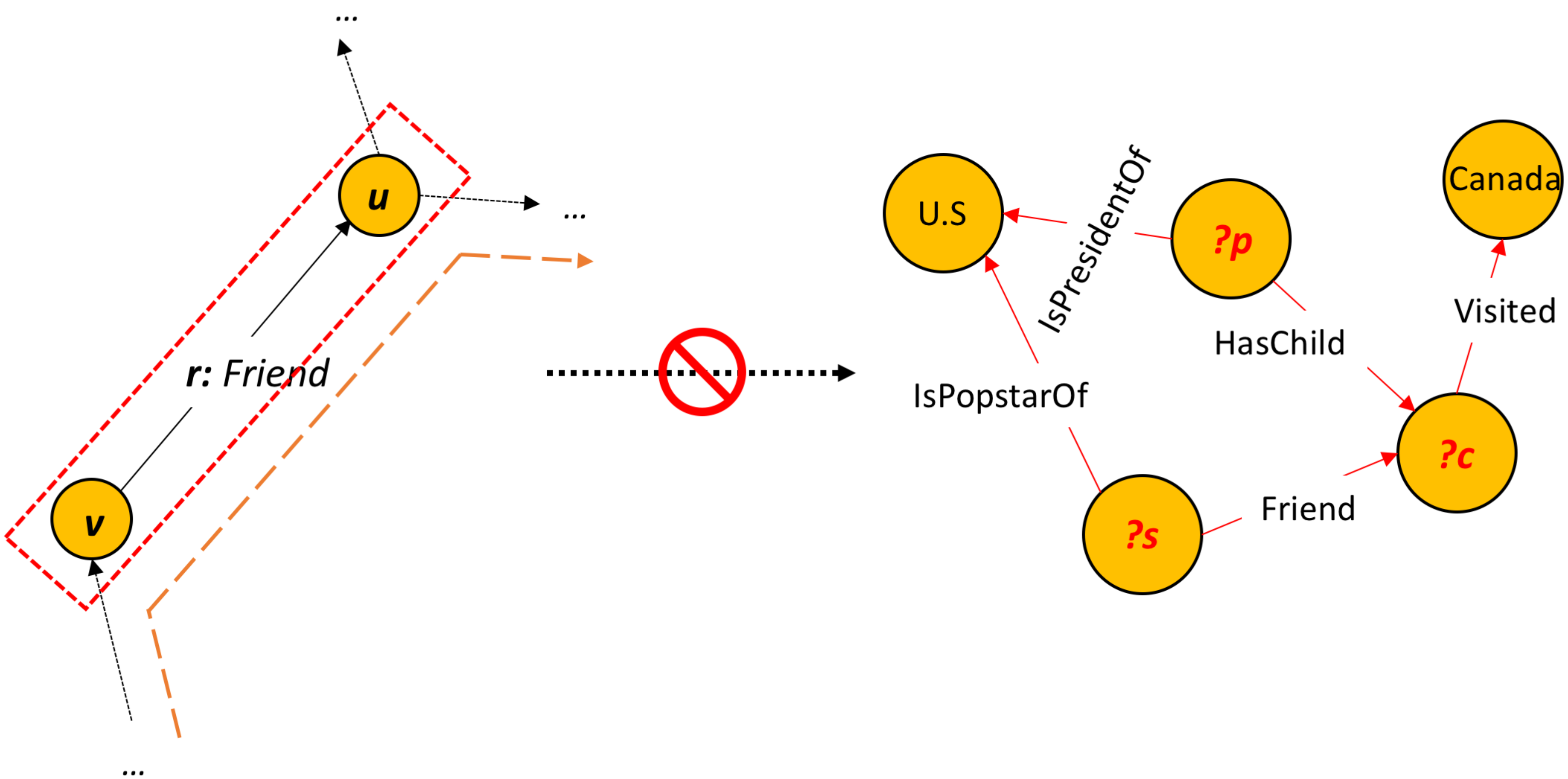




A Toy Knowledge Graph $\mathcal{G}$

An Embedding in 3-D Space

Knowledge Graph Embedding: Problem



A Toy Knowledge Graph $\mathcal{G}$

*[Consider **single edge** or **unidirectional** paths]*

A Graph Query Q

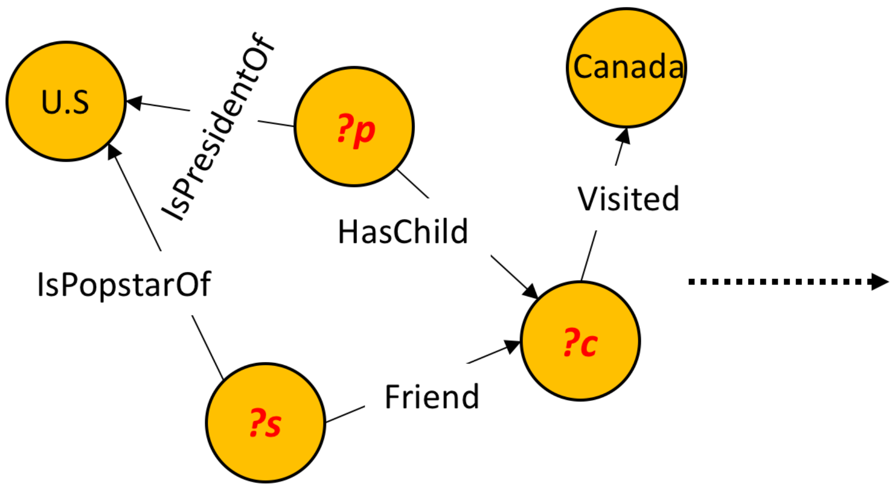
*[Composed of **bidirectional** paths]*

PAGE: Graph **P**attern Query **A**nswering via Knowledge **G**raph **E**mbedding

- **Contributions:**
  - We propose a novel method that answers a graph pattern query via knowledge graph embedding
  - Our method finds latent answers from a knowledge graph that carries *incorrect* or *incomplete information*
  - Our method learns vector representations, considering the error of *bi-directional paths* in a knowledge graph

PAGE: Energy of a Bi-directional Paths Query

A graph query Q can be seen as ***a series of bi-directional path queries***



p_1: (US) <- [:IsPresidentOf]-(?p) - [:HasChild] -> (?c)
p_2: (US) <- [:IsPopstarOf]-(?s) - [:Friend] -> (?c)
p_3: (?c) <- [:Visited] - (CANADA)

A Graph Query Q

Bi-directional Path Queries

PAGE: Energy of a Bi-directional Paths Query [cont]

- **Energy of a bi-directional path:**

  - **Inverse operation:**
    Given a query $?x \overset{r}{\rightarrow} u$, the inverse operation is to find x such that $energy(\boldsymbol{x}, \boldsymbol{r}, \boldsymbol{u}) = 0$
    (ex. [in TransE] $\boldsymbol{x} = \boldsymbol{u} - \boldsymbol{r}$)

  - **Energy of a bidirectional path:**
    Given a h-hop bi-directional path p, whose left and right ends are $\boldsymbol{u}$ and $\boldsymbol{v}$ with a series of intermediate relations $\boldsymbol{r_1} \dots \boldsymbol{r_{h-1}}$,
    $$energy(p) = \begin{cases} ||\boldsymbol{x} + \boldsymbol{r_h} - \boldsymbol{v}|| & \text{, if the last edge is } \overset{\boldsymbol{r_h}}{\rightarrow} \boldsymbol{v} \\ ||\boldsymbol{v} + \boldsymbol{r_h} - \boldsymbol{x}|| & \text{, if the last edge is } \overset{\boldsymbol{r_h}}{\leftarrow} \boldsymbol{v} \end{cases}$$
    where $\boldsymbol{x}$ is a vector calculated from $\boldsymbol{u}$ up to $\boldsymbol{r_{h-1}}$[in TransE].

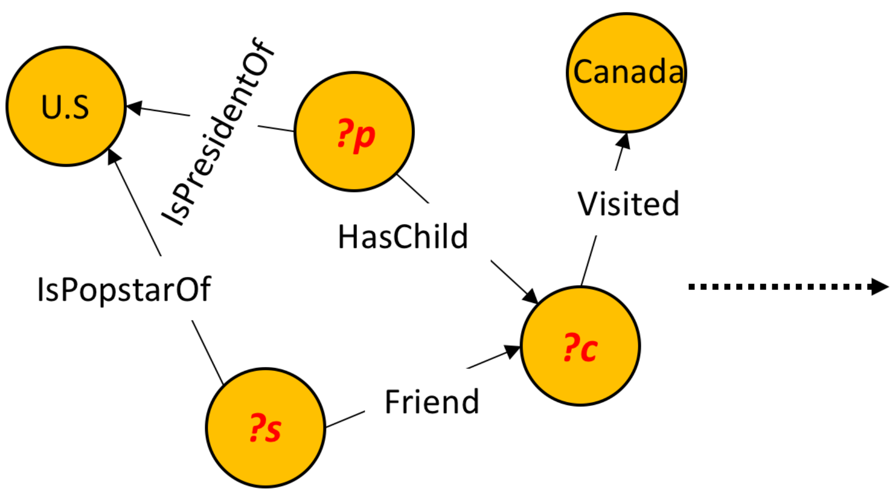
PAGE: Energy of a Bi-directional Paths Query [cont]

- **Energy of a graph query:**
  - Let Q be a graph pattern query and q be an candidate answer to Q, the energy of the graph query is defined as:

$$energy(q) = \sum_{p \in path(q)} energy(p)$$

PAGE: Energy of a Bi-directional Paths Query [cont]

- **Energy of a graph query [cont]:**
  - ex.



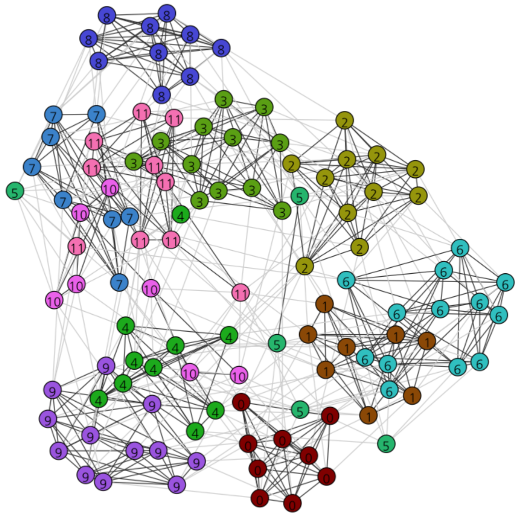
A Graph Query Q

$\boldsymbol{p_1}$: (US) <- [:IsPresidentOf]-(?p) - [:HasChild] -> (?c)
$\boldsymbol{p_2}$: (US) <- [:IsPopstarOf]-(?s) - [:Friend] -> (?c)
$\boldsymbol{p_3}$: (?c) <- [:Visited] - (CANADA)

$$energy(q) = e(p_1) + e(p_2) + e(p_3)$$

PAGE: Improve Training of KGE Methods

The training datasets used in most knowledge graph embedding methods only consist of ***single edge (factoid) queries***
but PAGE need to learn the latent representation for ***graph queries***



?

A Knowledge Graph $\mathcal{G}$

A Training Dataset for PAGE

PAGE: Improve Training of KGE Methods [cont]

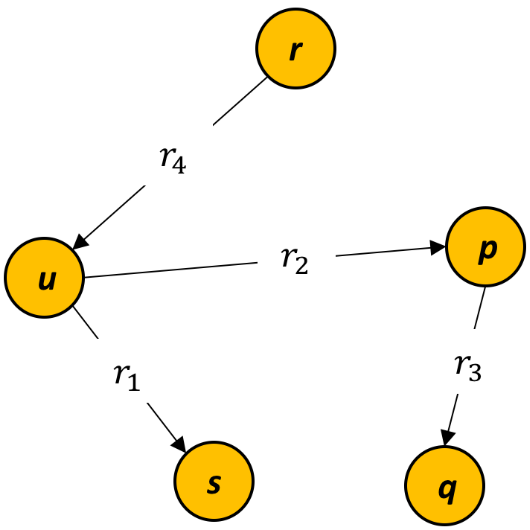
- **Sampling spanning trees:**

  1. Randomly choose a terminal vertex from a knowledge graph G
  2. Perform the Join 3(b) of the FFSM [2] e times
     so that a spanning tree that has e edges can be sampled
  3. Repeat the step 1 and 2 until all vertices and edges in G
     are covered by at least c different sampled spanning trees

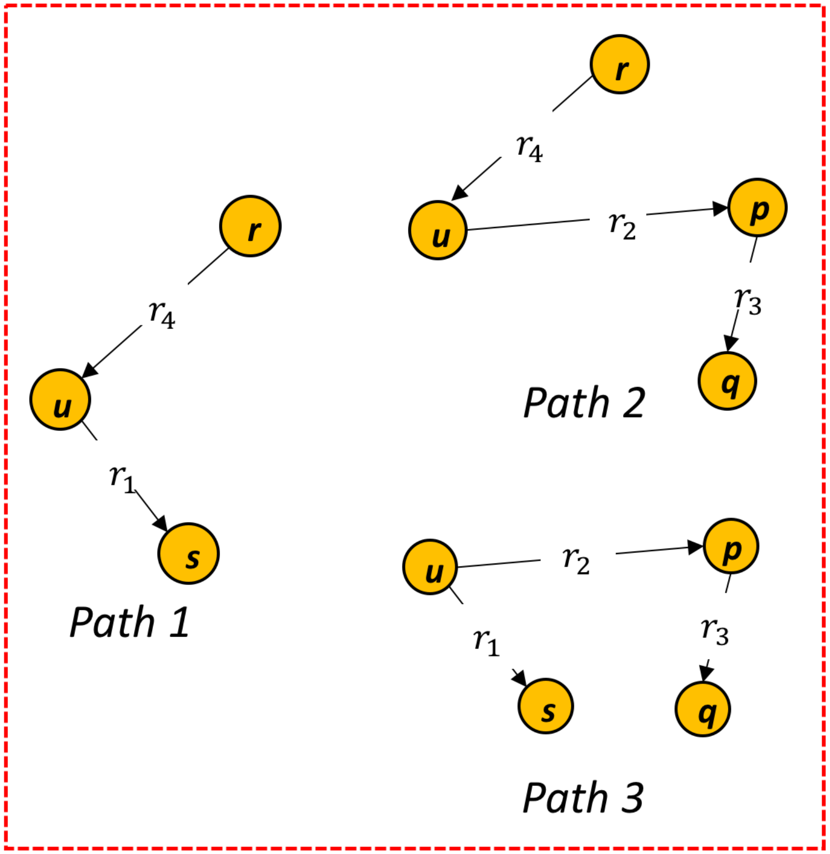
[2] Huan, Jun et al. "Efficient mining of frequent subgraphs in the presence of isomorphism." ICDM'03.

PAGE: Improve Training of KGE Methods [cont]

- **Decompose spanning trees into bi-directional paths:**



A Sampled Spanning Tree



A Set of Bi-directional Paths

PAGE: Improve Training of KGE Methods [cont]

- **Margin-based (Hinge) loss function**

$$\mathcal{L} = \sum_{p^+ \in path(p)} \sum_{p^-} \max(0, \gamma + e(p^+) - e(p^-))$$

⇢ Maximize the error between the true and false bi-directional paths
(ex., true path $p^+ = (\boldsymbol{x}, \boldsymbol{r_1} \ldots \boldsymbol{r_{h-1}}, \boldsymbol{u})$
false path $p^- = (\boldsymbol{x}, \boldsymbol{r_1} \ldots \boldsymbol{r_{h-1}}, \boldsymbol{v})$ or $(\boldsymbol{w}, \boldsymbol{r_1} \ldots \boldsymbol{r_{h-1}}, \boldsymbol{u})$

- **The optimization[1] [training]**

$$\underset{\mathrm{M}}{argmin}\ \mathcal{L}(\mathrm{M}) + \lambda \mathcal{R}(\mathrm{M})$$

1. Note: we use the stochastic gradient descent (SGD) method. The details are in our paper.

Evaluation

- **Experimental Setups**

  - **Tasks:** 1) Factoid query answering
    2) Graph query answering
  - **Databases:** *FB15K* and *Nell186*
    - Datasets: sampled spanning trees from these databases (training, testing, validating)
  - **Baseline methods:** *TransE* and *SE*[1]
  - **Metrics:** Mean rank and Hits@10/100/1000

1. Note: as the inverse operation of the SME method cannot be defined, we use TransE and SE.

Evaluation [cont]

- **Factoid Query Answering**

at most 13% better

Database	Metric	Type	TransE	PAGE-TransE	SE	PAGE-SE
FB15K	Mean Rank	Micro	181.76	**178.98**	408.69	375.48
		Macro	109.02	**106.09**	412.04	364.24
	Hits@10/100	Micro	43.4% / **76.6%**	**44.2%** / 76.2%	21.9% / 59.2%	22.3% / 59.3%
		Macro	49.2% / **81.3%**	**49.4%** / 80.9%	27.6% / 62.8%	28.9% / 62.8%
Nell186	Mean Rank	Micro	885.54	**784.02**	3412.0	3752.5
		Macro	885.54	**784.02**	4492.2	4736.8
	Hits@10/100	Micro	**41.5% / 74.6%**	38.6% / 72.4%	10.1% / 15.75%	9.2% / 14.5%
		Macro	**41.5% / 74.6%**	38.6% / 72.4%	3.3% / 8.0%	3.0% / 7.1%

similar

Evaluation [cont]

- **Graph Query Answering**

  - **Graph query generation from the databases:**
    1. Merge training and testing datasets into a knowledge graph
    2. Randomly choose a vertex v from the testing set
    3. Create z paths by iterating the following steps z times
       a) Choose a path length in between 2 and 4
       b) Randomly select a path of the chosen length starting from v, whose path should have at least one edge in the testing set.
    4. Convert v and all intermediate vertices of the paths into variables and create a graph query q from them
    5. The correct answer to the query q is v; we are interested in finding a vertex mapped to the variable converted from v

Evaluation [cont]

- **Graph Query Answering**

slightly improved

Database	Metric	Type	TransE	PAGE-TransE	SE	PAGE-SE
FB15K	Mean Rank	Micro	1150.5	**1088**	7493.5	7514.0
		Macro	2509.9	**2362.8**	7571.4	7933.7
	Hits@100/1000	Micro	18.3% / 56.7%	**25% / 60%**	1.7% / 5.0%	1.7% / 10%
		Macro	19% / 58.7%	**24.3% / 61.7%**	2.0% / 6.0%	2.0% / 7.7%
Nell186	Mean Rank	Micro	38.5	**38**	4803.6	4960.1
		Macro	769.4	**491.1**	5240.3	5431.8
	Hits@100/1000	Micro	64.8% / 89.4%	**66.5% / 94.6%**	15.75% / 31.4%	14.5% / 28.9%
		Macro	60.2% / 80.7%	**65.4% / 87.2%**	7.9% / 23.4%	7.1% / 21.3%

9% to 28% improvements

Conclusions

- We propose PAGE, a novel method that answers a graph pattern query via knowledge graph embedding
- PAGE is able to find latent answers from a knowledge graph that carries *incorrect* or *incomplete* information
- PAGE improves the performances in both the factoid query (*at most 13%*) and graph query answering (*9 to 28%*) tasks

Thanks!

Sanghyun Hong

shhong@cs.umd.edu