Reasoning over Knowledge Graphs

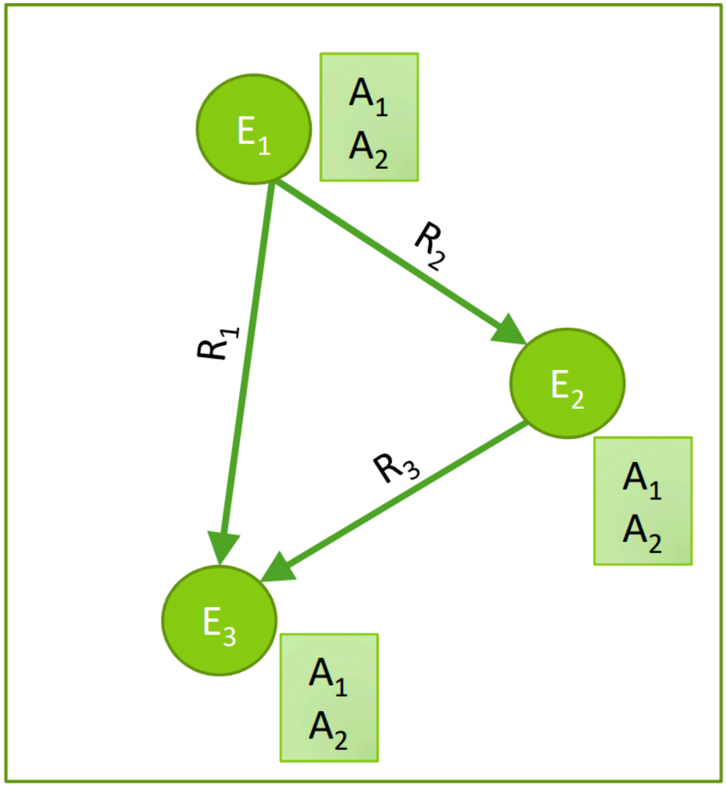
CS224W: Machine Learning with Graphs
Jure Leskovec, Hongyu Ren, Stanford University
http://cs224w.stanford.edu

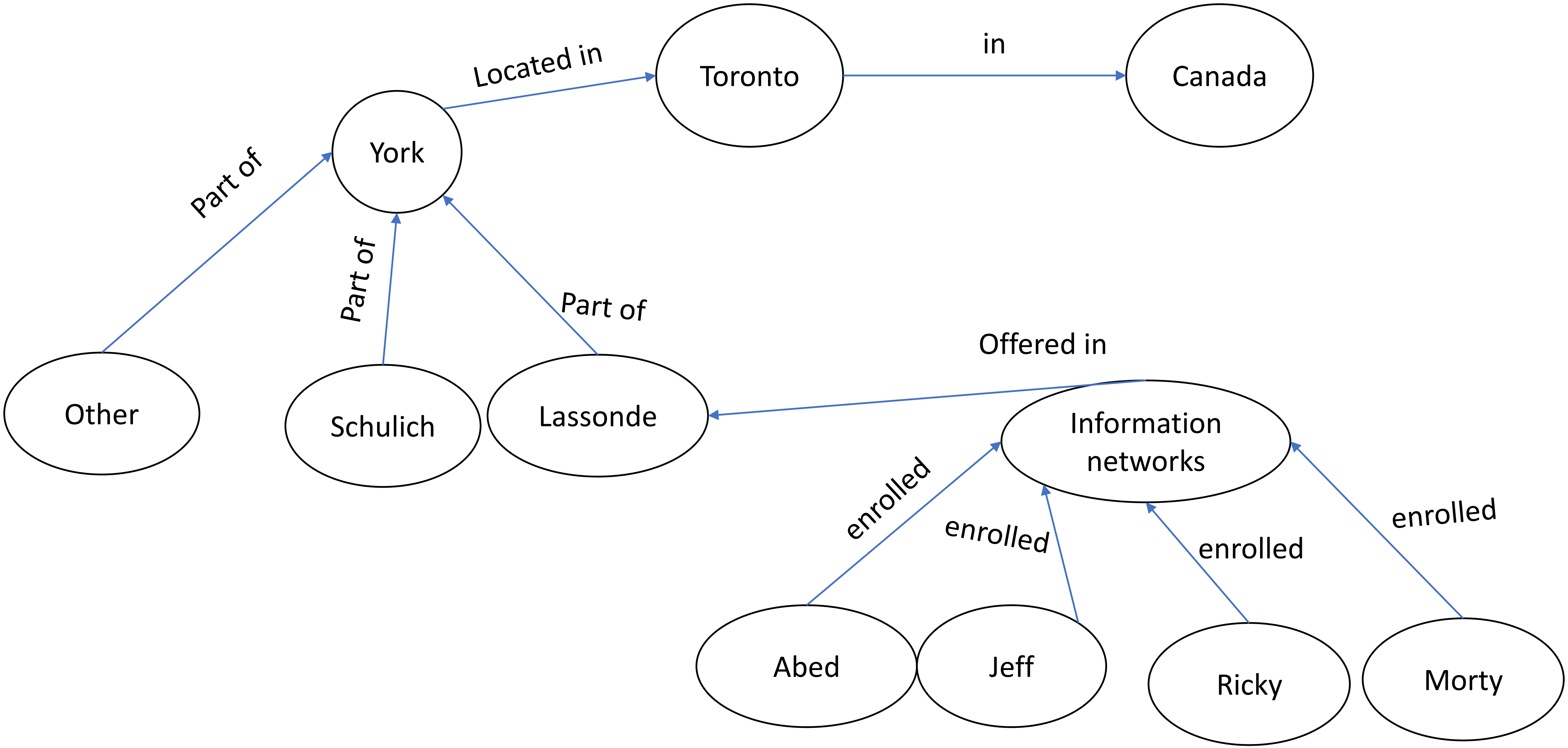
Outline of Today's Lecture

1. **Introduction to Knowledge Graphs**
2. **Knowledge Graph completion**
3. **Path Queries**
4. **Conjunctive Queries**
5. **Query2Box: Reasoning with Box Embeddings**

Knowledge Graphs

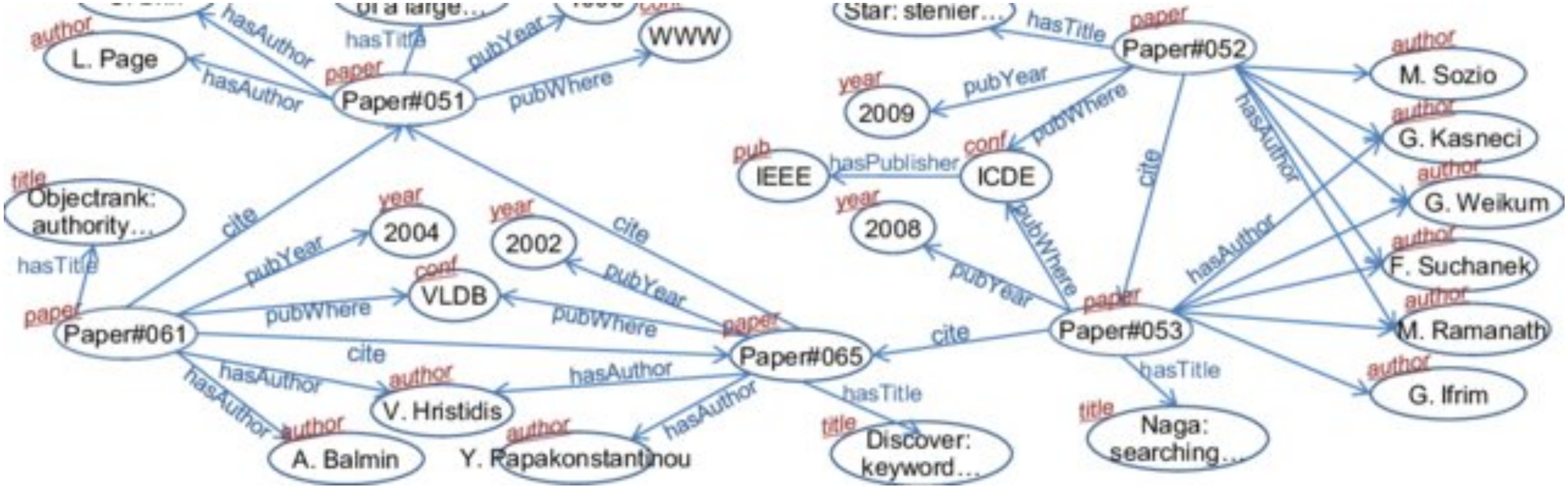
- Knowledge in graph form
  - Capture entities, types, and relationships
- Nodes are **entities**
- Nodes are labeled with their **types**
- Edges between two nodes capture **relationships** between entities




Located in
in
Toronto
Canada
York
Part of
Part of
Part of
Other
Schulich
Lassonde
Offered in
Information networks
enrolled
enrolled
enrolled
enrolled
Abed
Jeff
Ricky
Morty

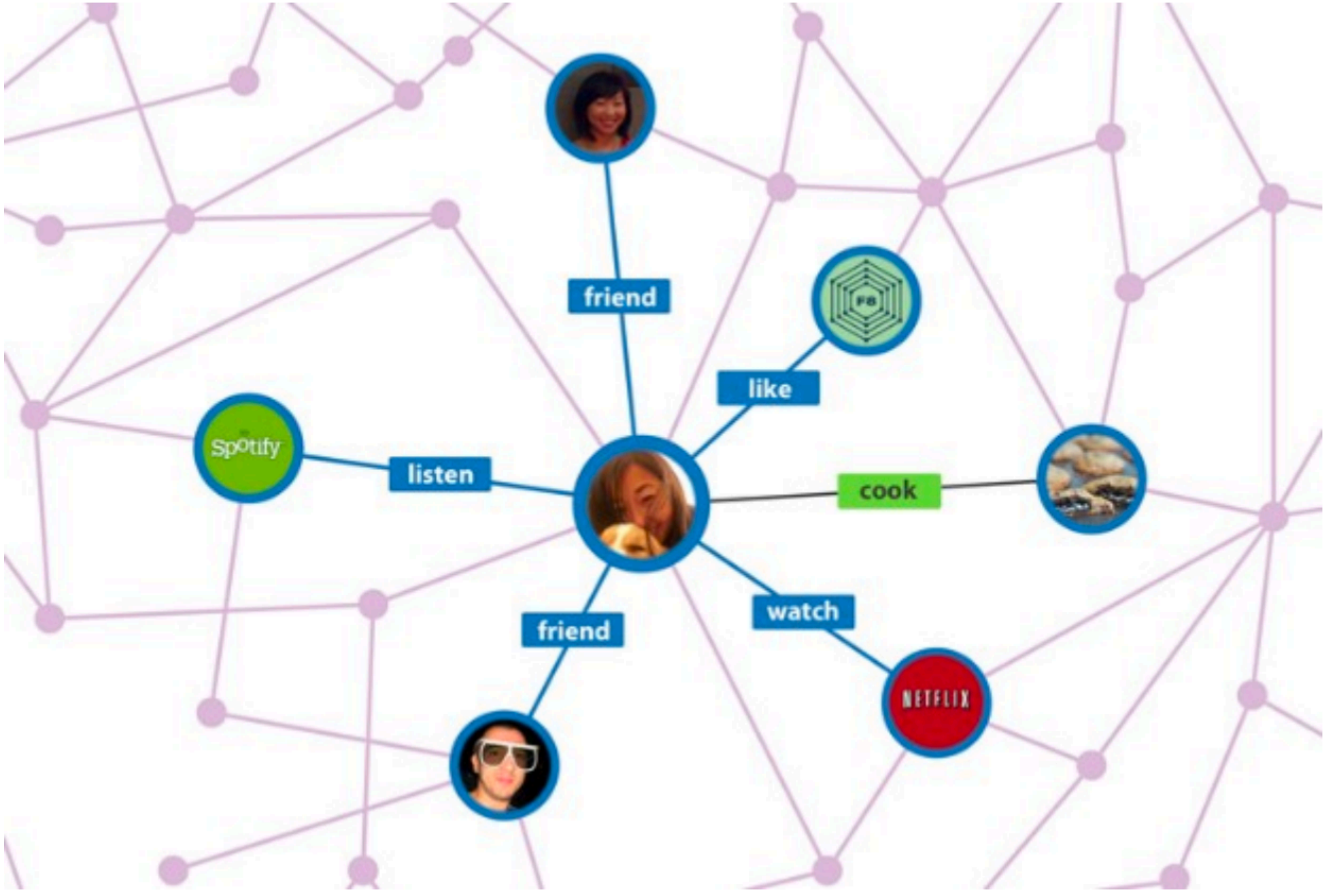
Example: Bibliographic networks

- **Node types**: paper, title, author, conference, year
- **Relation types**: pubWhere, pubYear, hasTitle, hasAuthor, cite

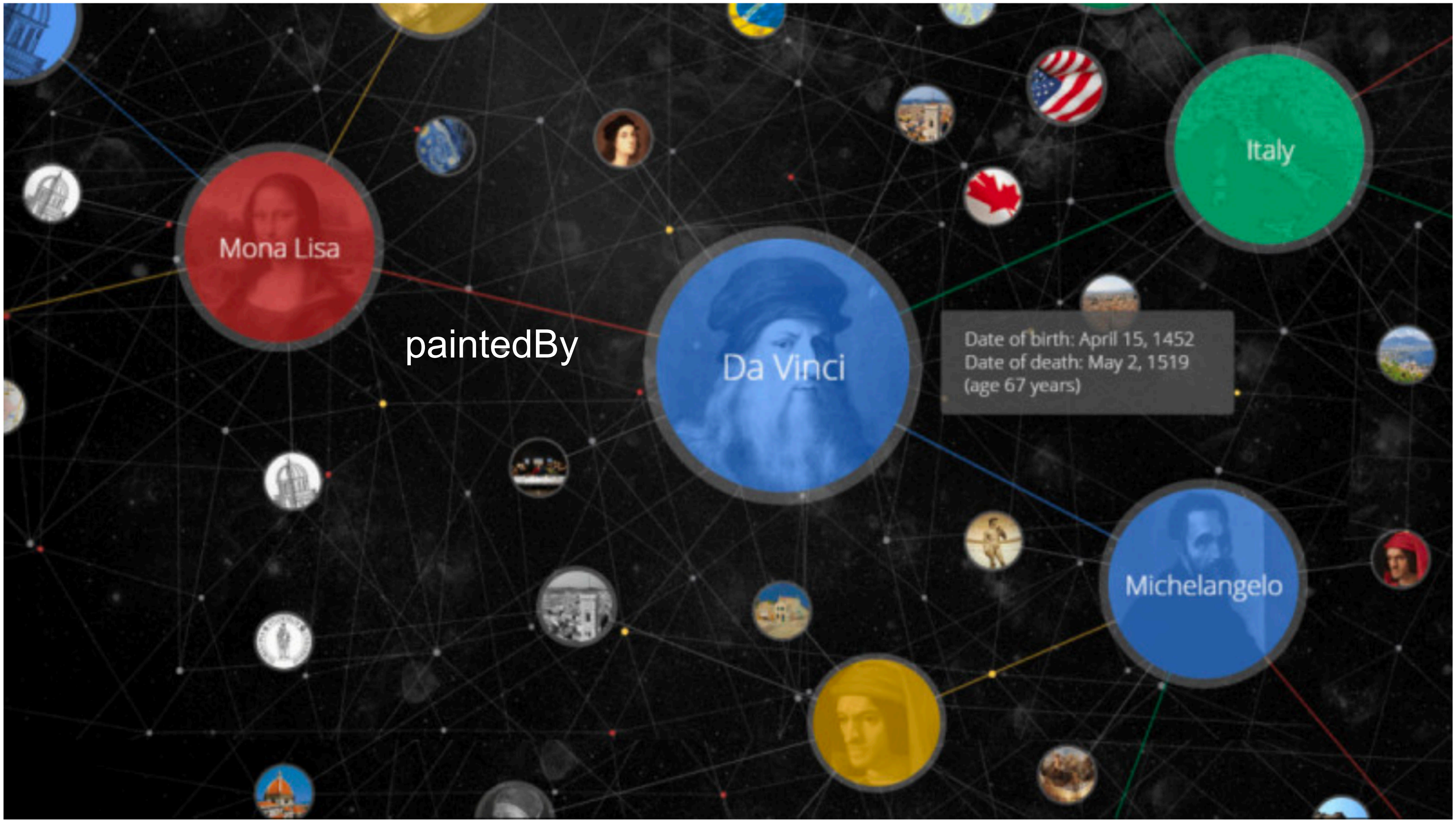


Example: Social networks

- **Node types**: account, song, post, food, channel
- **Relation types**: friend, like, cook, watch, listen

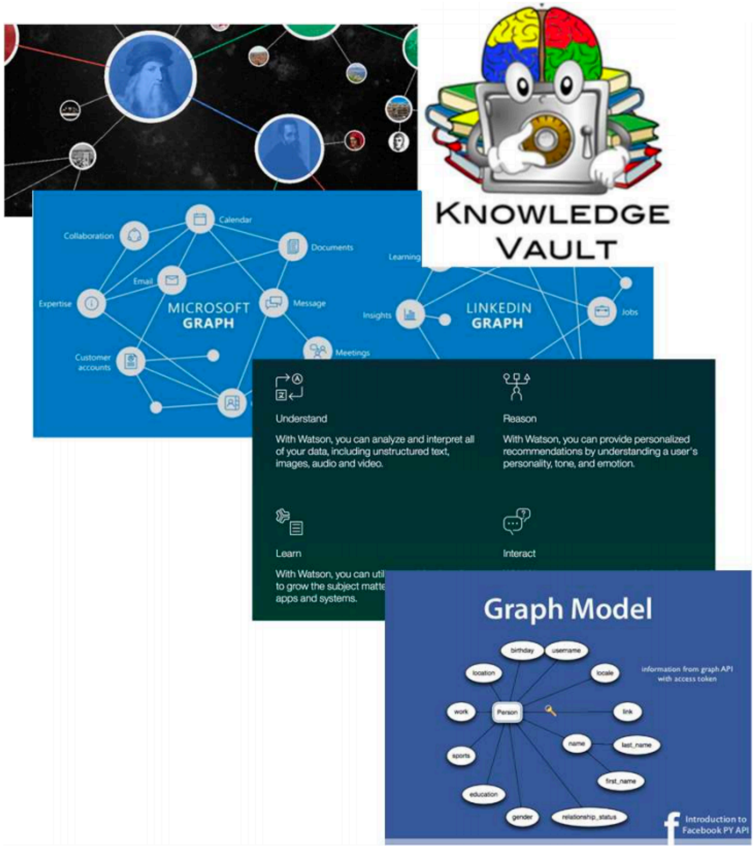


Example: Google Knowledge Graph



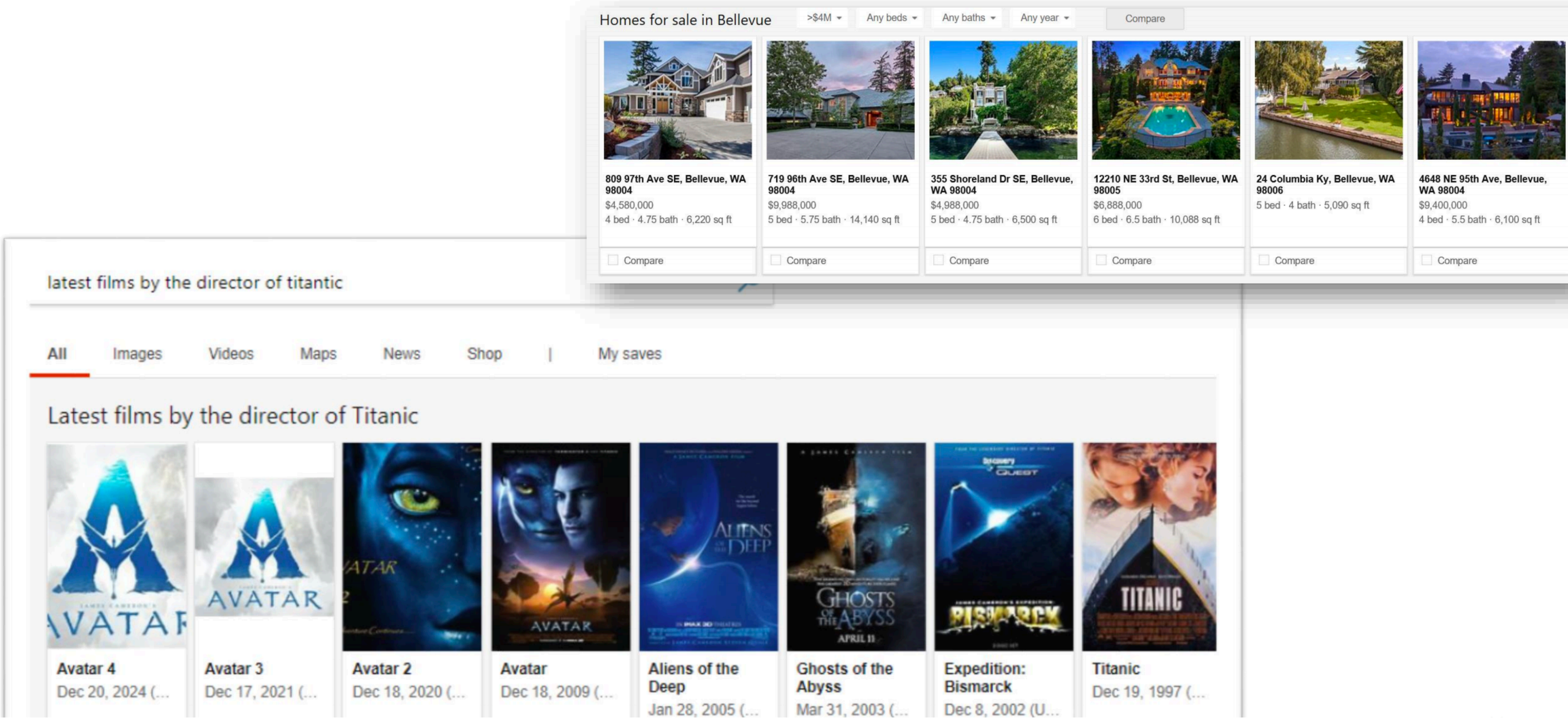
Knowledge Graphs in Practice

- Google Knowledge Graph
- Amazon Product Graph
- Facebook Graph API
- IBM Watson
- Microsoft Satori
- Project Hanover/Literome
- LinkedIn Knowledge Graph
- Yandex Object Answer



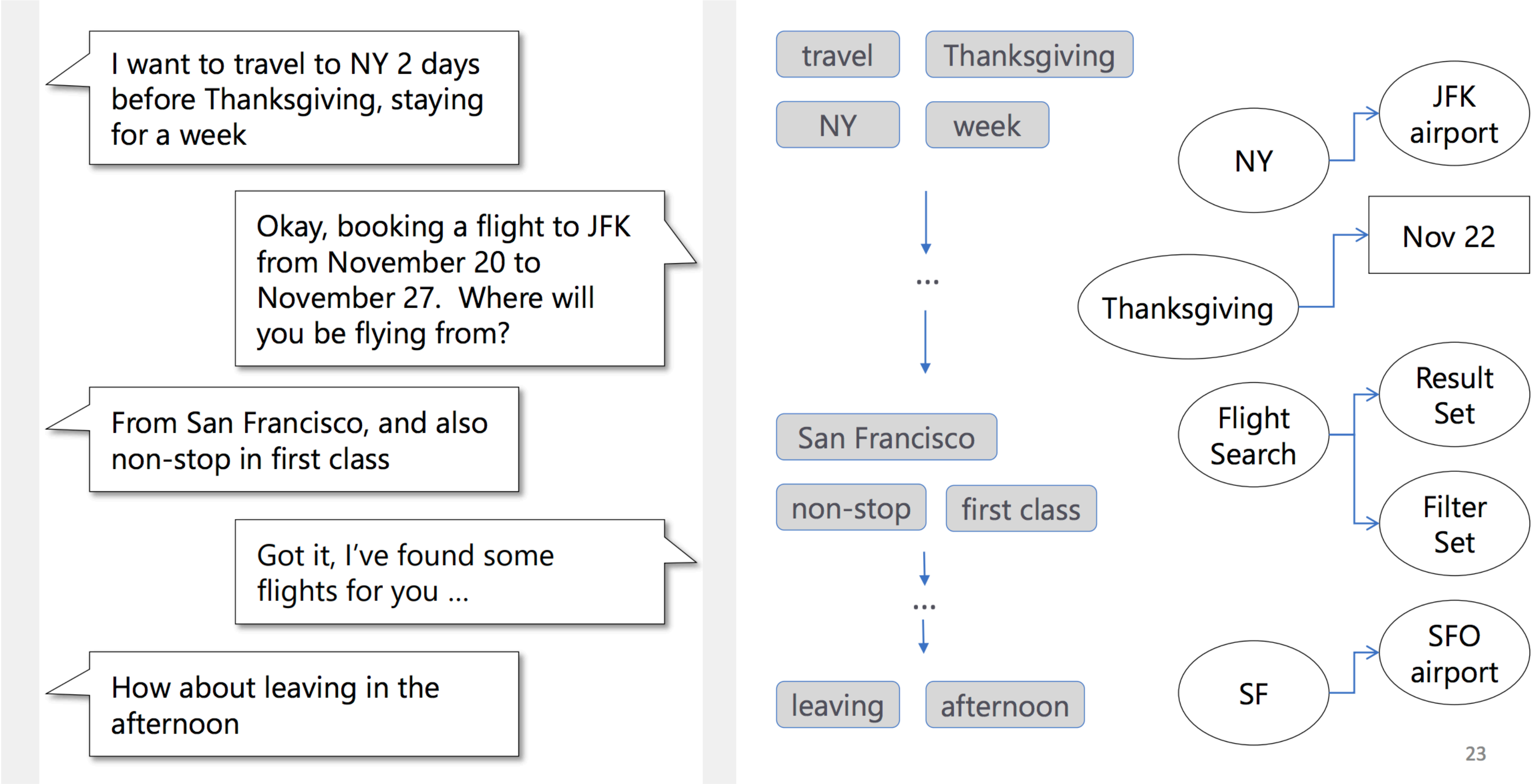
Applications of Knowledge Graphs

- **Serving information**

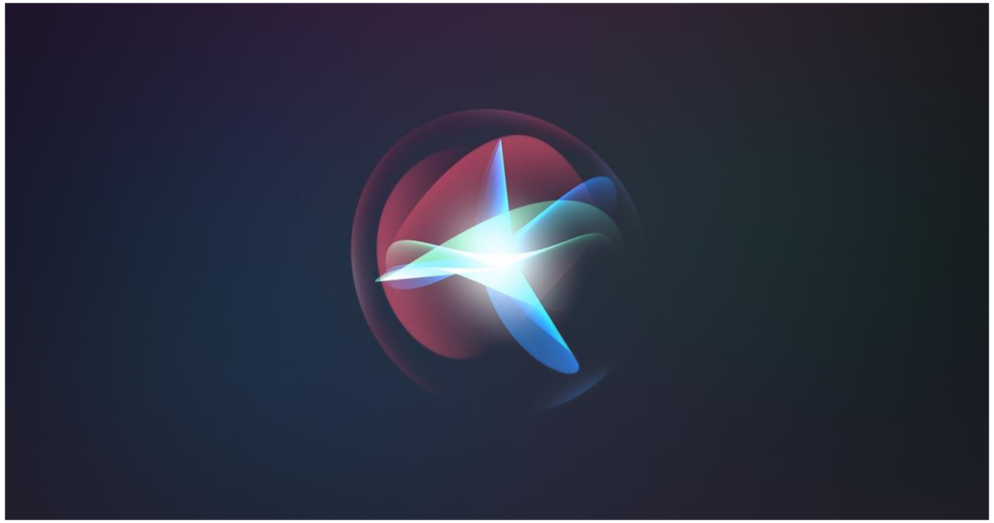


Applications of Knowledge Graphs

- **Question answering and conversation agents**




amazon



Outline

1. **Introduction to Knowledge Graphs**

2. **Knowledge Graph completion**

3. **Path Queries**

4. **Conjunctive Queries**

5. **Query2Box: Reasoning with Box Embeddings**

Knowledge Graph Datasets

- Publicly available KGs:
  - FreeBase, Wikidata, Dbpedia, YAGO, NELL, etc.
- Common characteristics:
  - **Massive**: millions of nodes and edges
  - **Incomplete**: many true edges are missing

Given a massive KG, enumerating all the possible facts is intractable! ⇨ **Can we predict plausible BUT missing links?**

Example: Freebase

- Freebase
  - ~50 million **entities**
  - ~38K **relation types**
  - ~3 billion **facts/triples**

93.8% of persons from Freebase have no place of birth and 78.5% have no nationality!

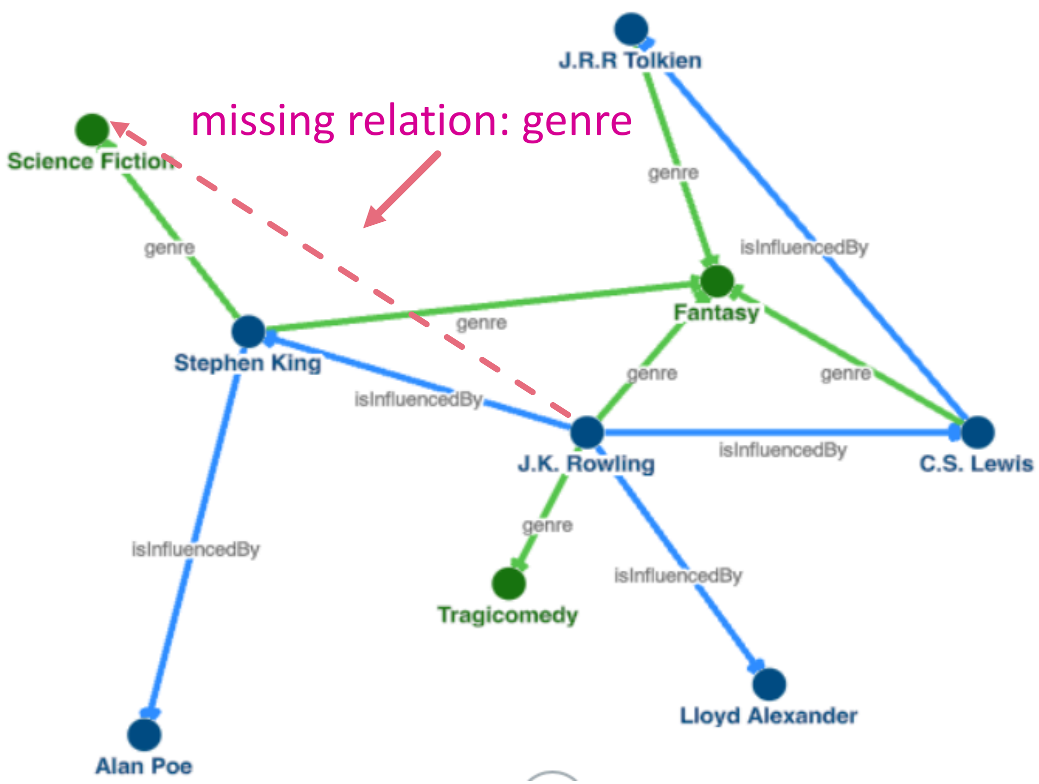
- FB15k/FB15k-237
  - A **complete** subset of Freebase, used by researchers to learn KG models

Dataset	Entities	Relations	Total Edges
FB15k	14,951	1,345	592,213
FB15k-237	14,505	237	310,079

[1] Paulheim, Heiko. "Knowledge graph refinement: A survey of approaches and evaluation methods." *Semantic web* 8.3 (2017): 489-508.
[2] Min, Bonan, et al. "Distant supervision for relation extraction with an incomplete knowledge base." *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 2013.

KG Completion

- Given an enormous KG, can we complete the KG / predict missing relations?
  - links + type



KG Representation

- Edges in KG are represented as **triples** (h, r, t)
  - head (h) has relation (r) with tail (t).
- Key Idea:
  - Model entities and relations in the embedding/vector space $\mathbb{R}^d$.
  - Given a true triple (h, r, t), the goal is that the embedding of (h, r) **should be close** to the embedding of t.
    - How to embed (h, r)?
    - How to define closeness?

Relation Patterns

- Symmetric Relations:

$$r(h,t) \Rightarrow r(t,h) \quad \forall h,t$$

  - **Example**: Family, Roommate
- Composition Relations:

$$r_1(x,y) \wedge r_2(y,z) \Rightarrow r_3(x,z) \quad \forall x,y,z$$

  - **Example**: My mother's husband is my father.
- 1-to-N, N-to-1 relations:

$r(h,t_1), r(h,t_2), \ldots, r(h,t_n)$ are all True.

  - **Example**: r is "StudentsOf"

TransE

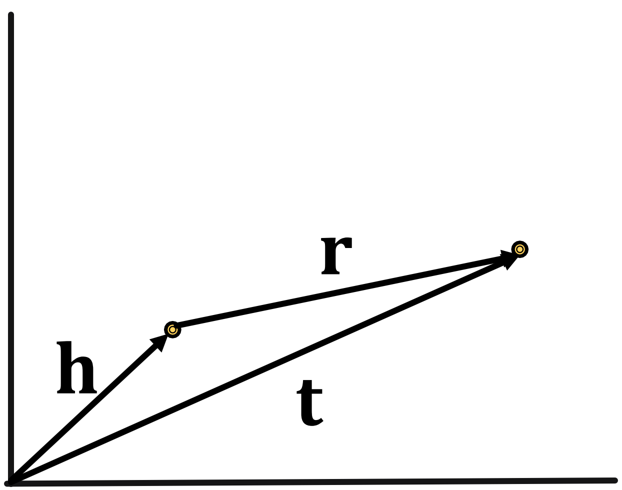
- **Translation Intuition**:

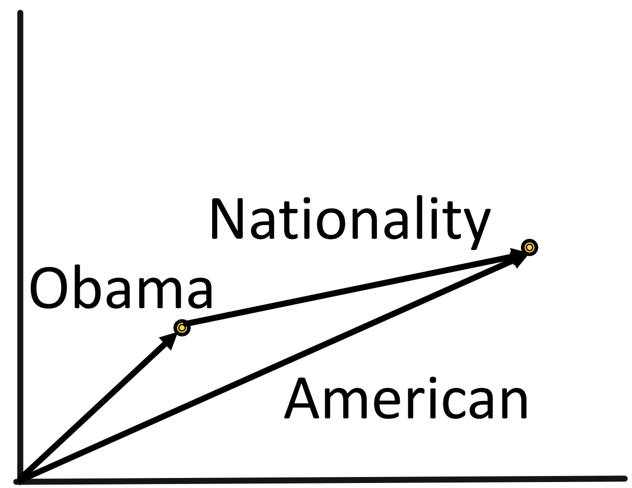
For a triple (h, r, t), $\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathbb{R}^d$,

$$\mathbf{h} + \mathbf{r} = \mathbf{t}$$

NOTATION: embedding vectors will appear in boldface

Score function: $f_r(h, t) = ||h + r - t||$





Bordes, Antoine, et al. "Translating embeddings for modeling multi-relational data." *Advances in neural information processing systems*. 2013.

TransE Training

- Translation Intuition: for a triple (h, r, t),

$$\mathbf{h} + \mathbf{r} = \mathbf{t}$$

Max margin loss:

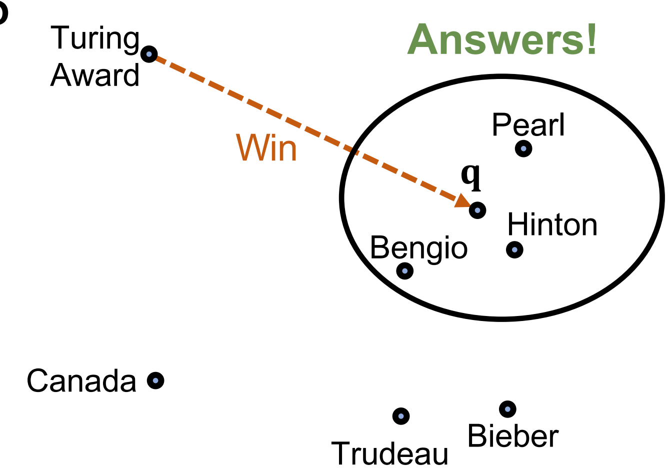
$$\mathcal{L} = \sum_{(h,r,t)\in G,(h,r,t')\notin G} [\gamma + \underbrace{f_r(h,t)}_{\text{Valid triple}} - \underbrace{f_r(h,t')}_{\text{Corrupted triple}}]_+$$

where γ is the margin, i.e., the smallest distance tolerated by the model between a valid triple and a corrupted one.

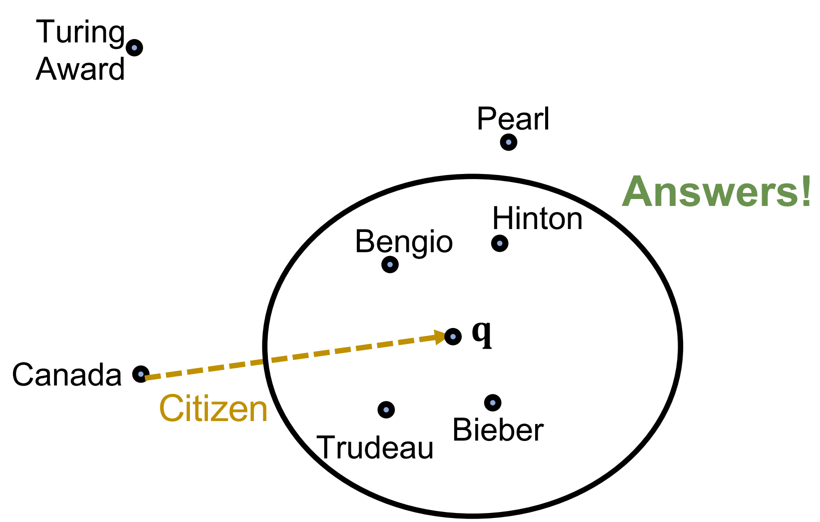
NOTE: check lecture 7 for a more in-depth discussion of TransE!

Link Prediction in a KG using TransE

- Who has won the Turing award?



- Who is a Canadian citizen?



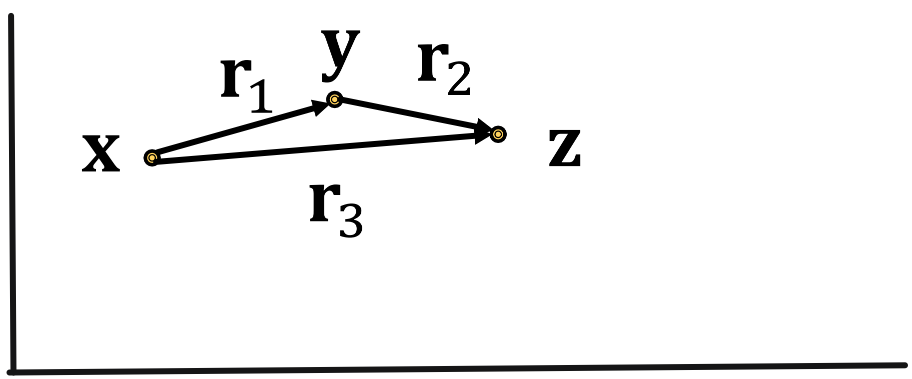
Composition in TransE

- Composition Relations:

  $$r_1(x,y) \wedge r_2(y,z) \Rightarrow r_3(x,z) \quad \forall x,y,z$$

- **Example**: My mother's husband is my father.
- In TransE:

  $$r_3 = r_1 + r_2 \checkmark$$



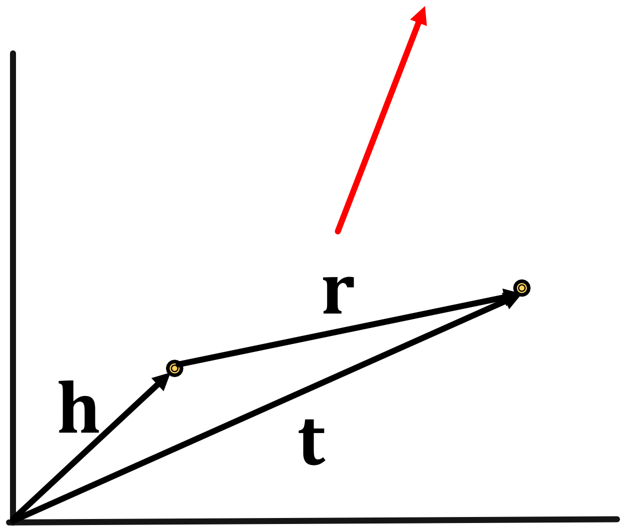
Limitation: Symmetric Relations

- Symmetric Relations:

$$r(h,t) \Rightarrow r(t,h) \quad \forall h,t$$

- **Example**: Family, Roommate
- In TransE:

$$r = 0, \; h = t \; ✘$$



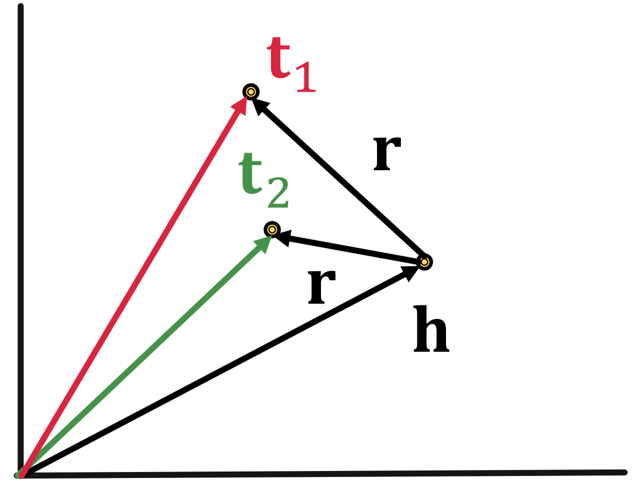
If we want TransE to handle symmetric relations r, for all h, t that satisfy $r(h,t)$, $r(t,h)$ is also True, which means $\|h + r - t\| = 0$ and $\|t + r - h\| = 0$. Then $r = 0$ and $h = t$, however h and t are two different entities and should be mapped to different locations.

Limitation: N-ary Relations

- 1-to-N, N-to-1, N-to-N relations.
- **Example**: (h, r, t_1) and (h, r, t_2) both exist in the knowledge graph, e.g., r is "StudentsOf"

With TransE, t_1 and t_2 will map to the same vector, although they are different entities.

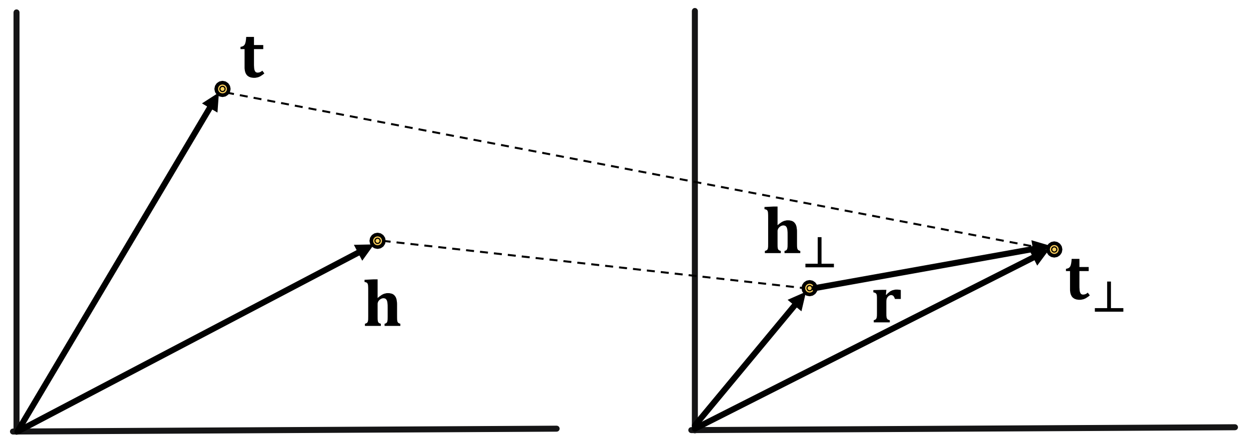
- $\mathbf{t}_1 = \mathbf{h} + \mathbf{r} = \mathbf{t}_2$
- $\mathbf{t}_1 \neq \mathbf{t}_2$ contradictory!



TransR

- TransR: model entities as vectors in the entity space $\mathbb{R}^d$ and **model each relation as vector r in relation space** $\mathbb{R}^k$ with $\mathbf{M}_r \in \mathbb{R}^{k\times d}$ as the projection matrix.

- $h_\perp = M_r h,\ t_\perp = M_r t$
- $f_r(h,t) = ||h_\perp + r - t_\perp||$



Lin, Yankai, et al. "Learning entity and relation embeddings for knowledge graph completion." *AAAI*. 2015.

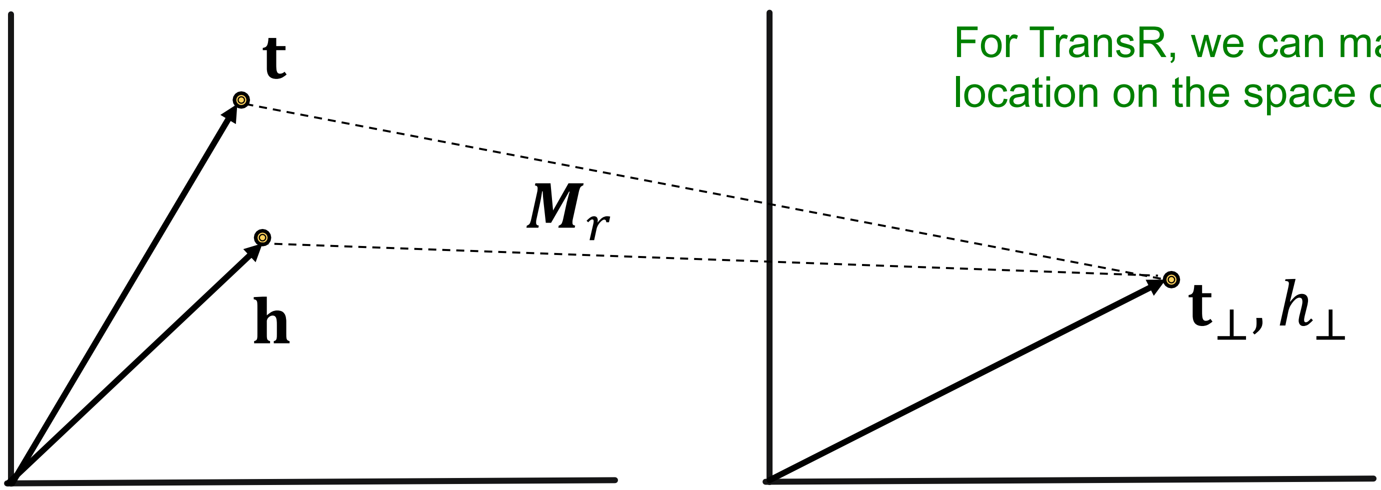
Symmetric Relations in TransR

- Symmetric Relations:

$$r(h, t) \Rightarrow r(t, h) \quad \forall h, t$$

- **Example**: Family, Roommate

$$r = 0,\ h_\perp = M_r h = M_r t = t_\perp \checkmark$$

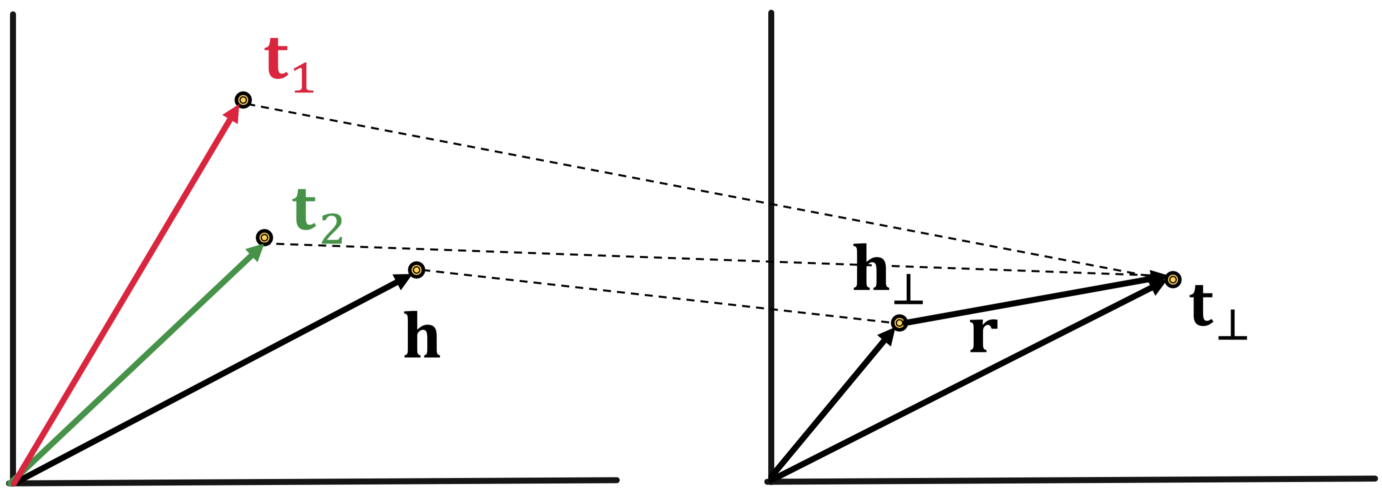


For TransR, we can map h and t to the same location on the space of relation r.

N-ary Relations in TransR

- 1-to-N, N-to-1, N-to-N relations
- **Example**: If (h, r, t_1) and (h, r, t_2) exist in the knowledge graph.

We can learn M_r so that $t_\perp = M_r t_1 = M_r t_2$, note that t_1 does not need to be equal to t_2!



Limitation: Composition in TransR

- Composition Relations:
  $$r_1(x, y) \wedge r_2(y, z) \Rightarrow r_3(x, z) \quad \forall x, y, z$$
- **Example**: My mother's husband is my father.

Each relation has different space.
It is **not naturally compositional** for multiple relations! ✘

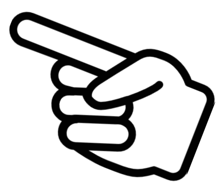
Translation-Based Embedding

Embedding	Entity	Relation	$f_r(h,t)$
TransE	$h, t \in \mathbb{R}^d$	$r \in \mathbb{R}^d$	$\|\|h + r - t\|\|$
TransR	$h, t \in \mathbb{R}^d$	$r \in \mathbb{R}^k, M_r \in \mathbb{R}^{k\times d}$	$\|\|M_r h + r - M_r t\|\|$

Embedding	Symmetry	Composition	One-to-many
TransE	✘	✓	✘
TransR	✓	✘	✓

Outline

1. **Introduction to Knowledge Graphs**
2. **Knowledge Graph completion**
3. **Path Queries**



4. **Conjunctive Queries**
5. **Query2Box: Reasoning with Box Embeddings**

Query Types on KG

- Can we do multi-hop reasoning, i.e., answer complex queries **efficiently** on an incomplete, massive KG?

Query Types	Examples
One-hop Queries	Where did Hinton graduate?
Path Queries	Where did Turing Award winners graduate?
Conjunctive Queries	Where did Canadians with Turing Award graduate?
EPFO Queries	Where did Canadians with Turing Award or Nobel graduate?

One-hop Queries

- We can formulate link prediction problems as answering one-hop queries.

- Link prediction: Is link (h, r, t) True?

$$\Updownarrow$$

- One-hop query: Is t an answer to query (h, r)?

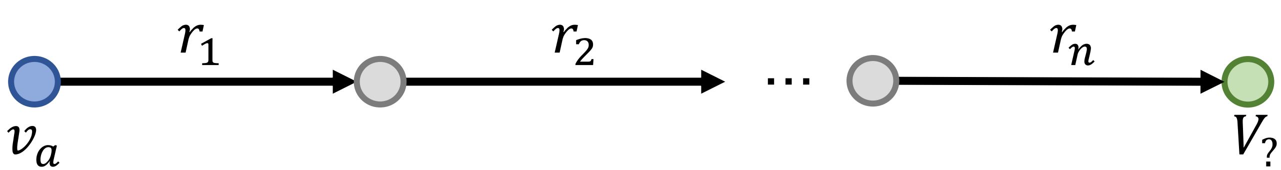
Path Queries

- Generalize one-hop queries to path queries by adding more relations on the path.
- Path queries can be represented by

$$q = (v_a, r_1, \ldots, r_n)$$

v_a is a constant node, answers are denoted by $[\![q]\!]$.

Computation graph of q:

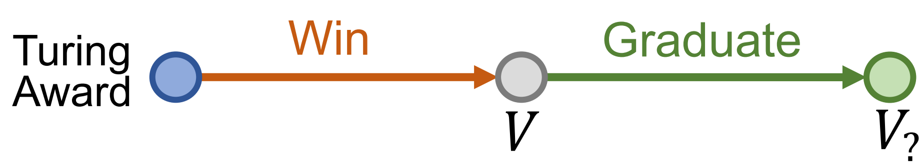


Computation graph of path queries is a chain.

Path Queries

"Where did Turing Award winners graduate?"

- v_a is "Turing Award".
- (r_1, r_2) is ("win", "graduate").



Given a KG, how to answer the query?

Traversing Knowledge Graphs

- Answer path queries by traversing the KG.

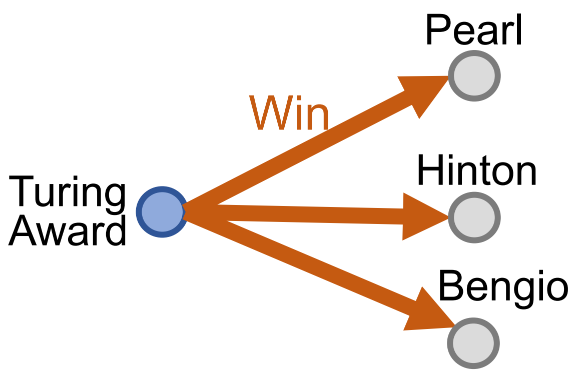
"Where did Turing Award winners graduate?"

The anchor node is Turing Award.

Traversing Knowledge Graphs

- Answer path queries by traversing the KG.

"Where did Turing Award winners graduate?"

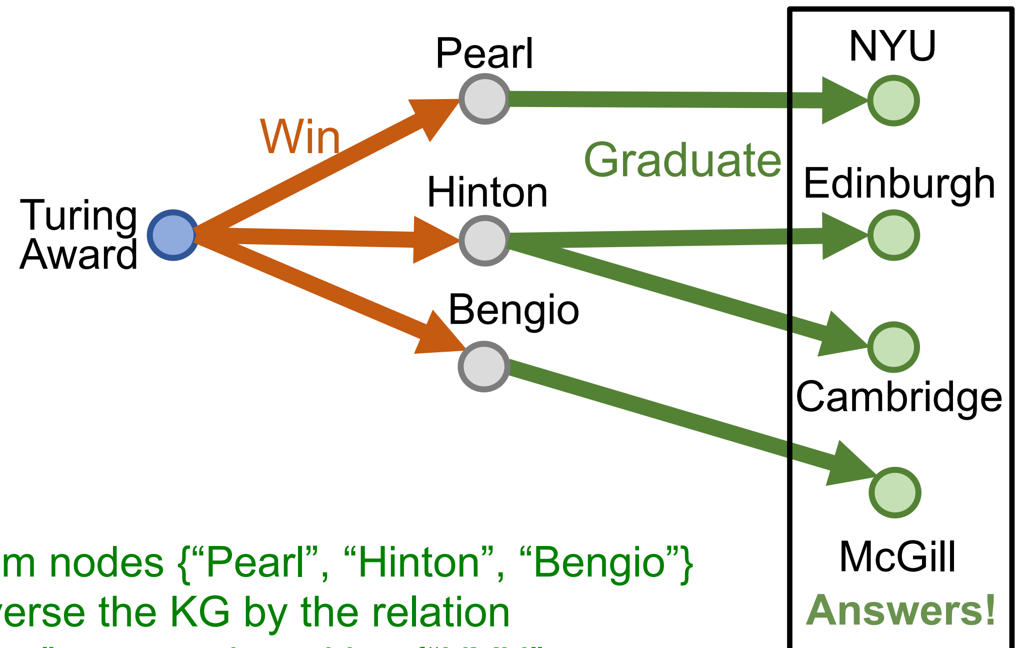


Start from the anchor node "Turing Award" and traverse the KG by the relation "Win", we reach entities {"Pearl", "Hinton", "Bengio"}.

Traversing Knowledge Graphs

- Answer path queries by traversing the KG.

"Where did Turing Award winners graduate?"

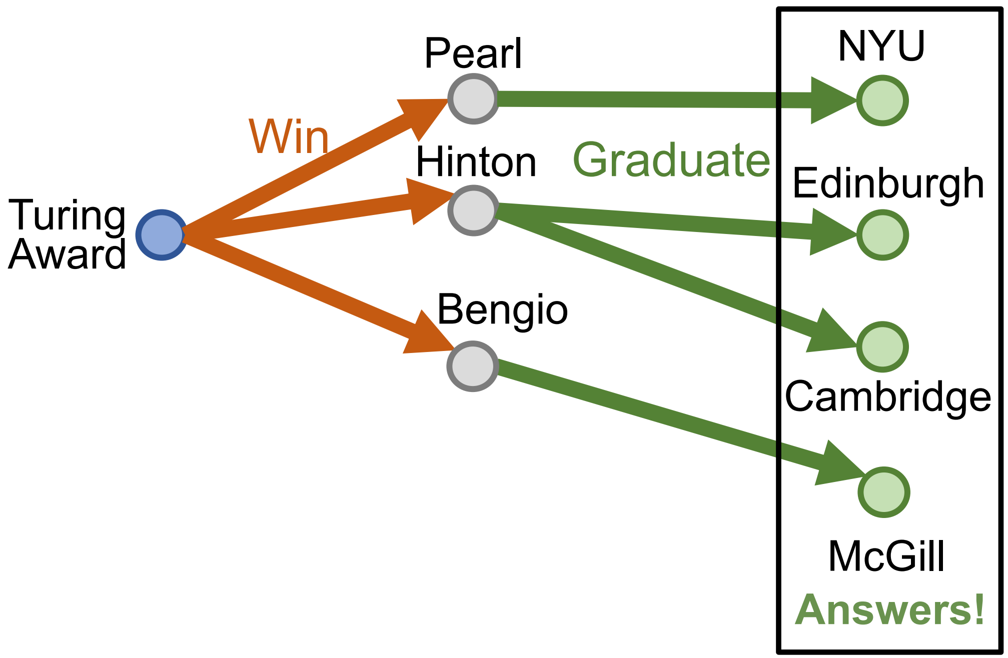


Start from nodes {"Pearl", "Hinton", "Bengio"} and traverse the KG by the relation "Graduate", we reach entities {"NYU", "Edinburgh", "Cambridge", "McGill"}. These are the answers to the query!

Traversing Knowledge Graphs

- Answer path queries by traversing the KG.

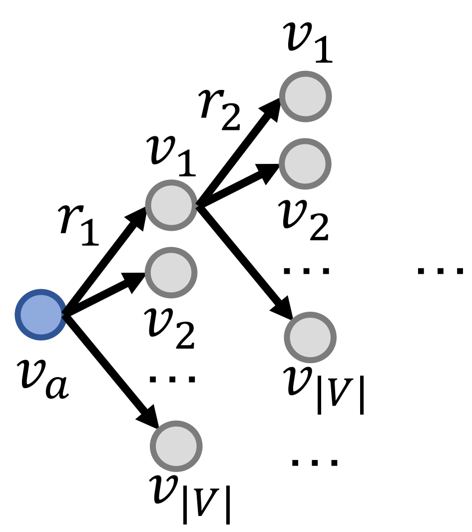
"Where did Turing Award winners graduate?"



What if KG is incomplete?

Answering Path Queries

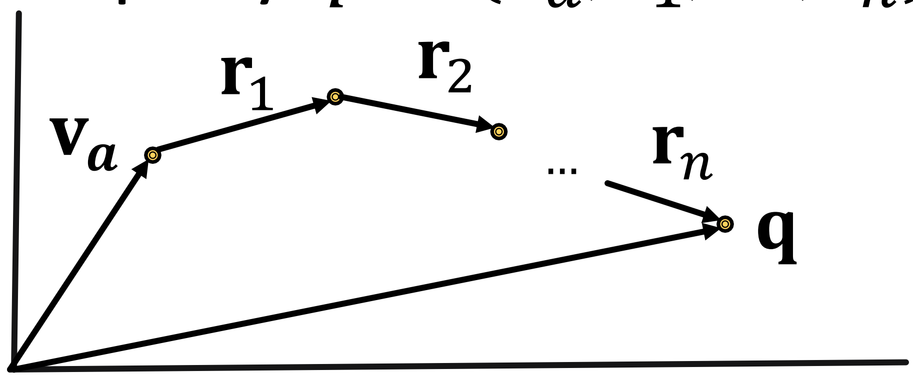
- Can we first do link prediction and then traverse the completed (probabilistic) KG?
- **No!** The completed KG is a **dense graph**!
- Time complexity of traversing a dense KG with $|V|$ entities to answer $(v_a, r_1, \dots, r_n)$ of length n is $\mathcal{O}(|V|^n)$.



Traversing KG in Vector Space

- **Key idea: embed queries!**
  - Generalize TransE to multi-hop reasoning.

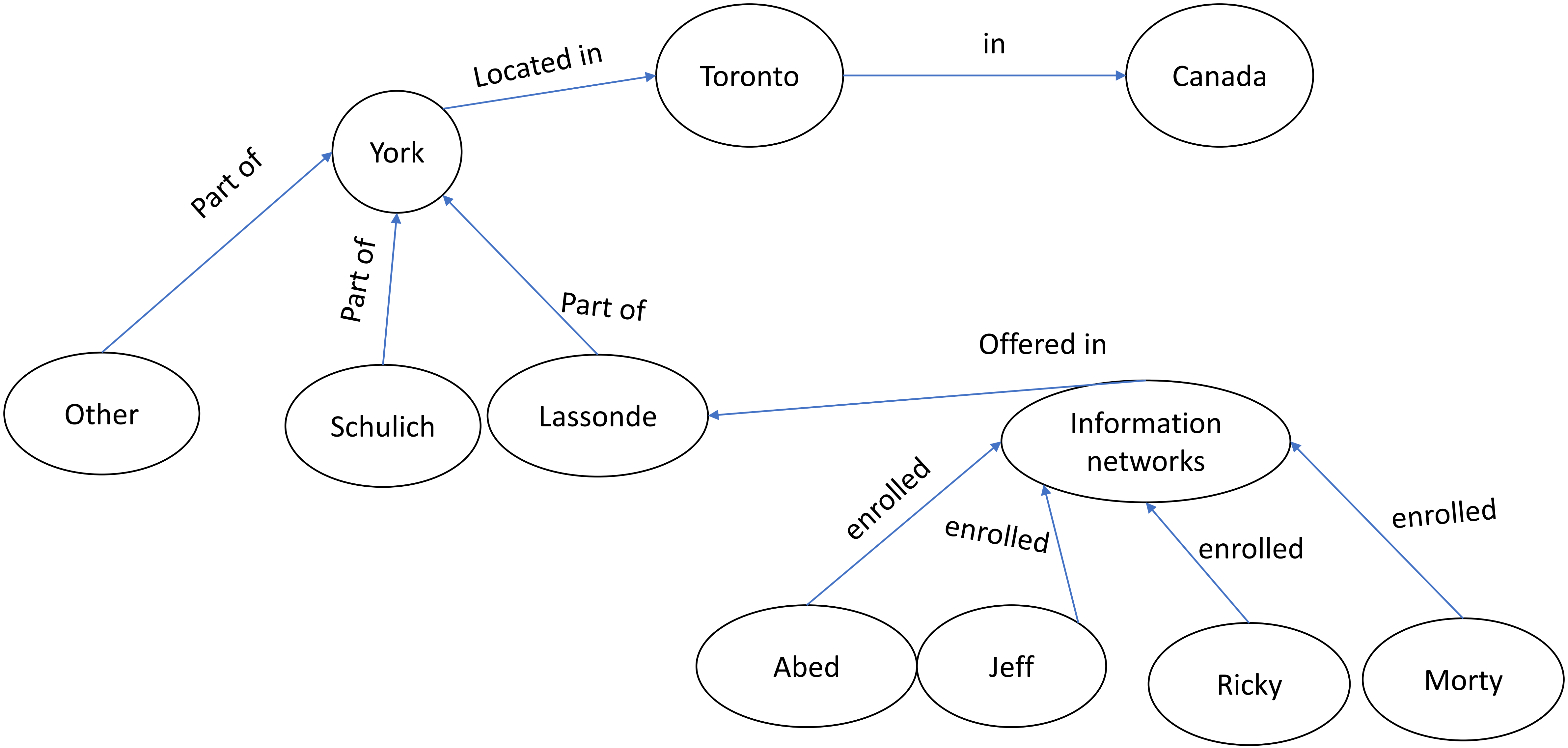
Given a path query $q = (v_a, r_1, \dots, r_n)$,



$$\mathbf{q} = \mathbf{v}_a + \mathbf{r}_1 + \cdots + \mathbf{r}_n$$

- Is v an answer to q?
  - Do a nearest neighbor search for all v based on $f_q(v) = ||\mathbf{q} - \mathbf{v}||$, time complexity is $\mathcal{O}(V)$.

Guu, Kelvin, John Miller, and Percy Liang. "Traversing knowledge graphs in vector space." arXiv preprint arXiv:1506.01094 (2015).


Located in
Toronto
in
Canada
York
Part of
Part of
Part of
Other
Schulich
Lassonde
Offered in
Information networks
enrolled
enrolled
enrolled
enrolled
Abed
Jeff
Ricky
Morty