DSC250: Advanced Data Mining

Knowledge Graphs

Zhiting Hu

Lecture 15, November 16, 2023

UC San Diego
HALICIOĞLU DATA SCIENCE INSTITUTE

Logistics

- Zhiting's Office Hour next week:
  - Wed, Nov.22, 10:30am
  - Friday Nov.24: Thanksgiving holidays

Outline

- Knowledge Graphs

- 6 paper presentations
    - Sarthak Doshi, Sarvesh Khire
    - Sourabh Prakash, Priyanshi Shah
    - Reventh Sharm
    - Aman Parikh, Kartikay Anand
    - Junke Ye, Yongqi Tong
    - Yunfei Luo, Tanjid Tonmoy

Knowledge Graphs (KGs)

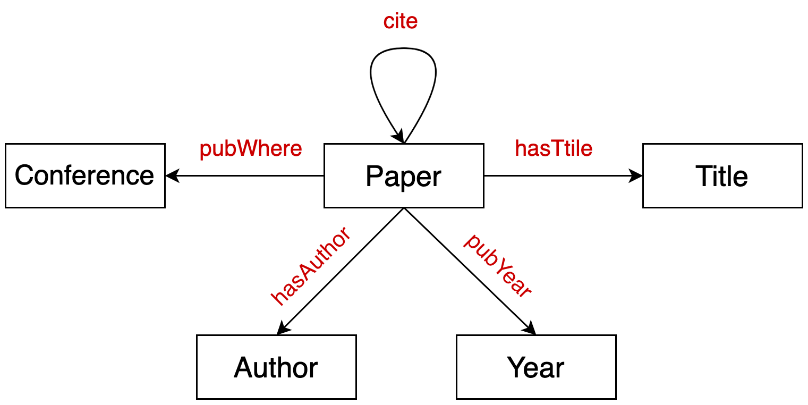
Slides adapted from:

- Jure Leskovec, Stanford CS224W: Machine Learning with Graphs

Recap: Example KGs

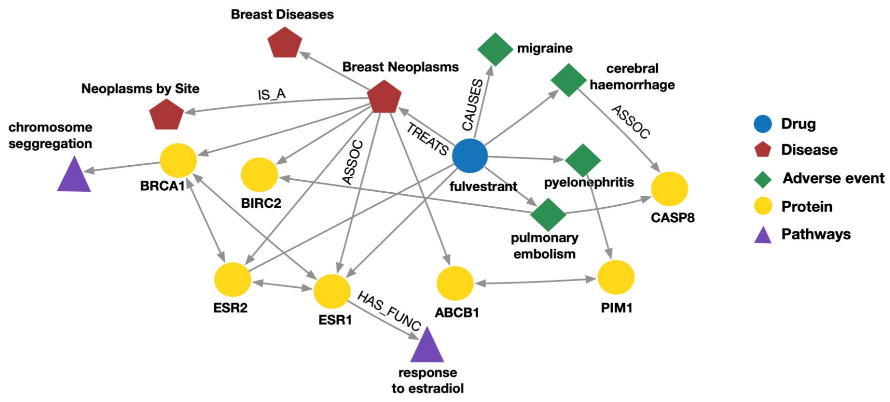
Bibliographic Networks

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



Bio Knowledge Graphs

- **Node types**: drug, disease, adverse event, protein, pathways
- **Relation types**: has_func, causes, assoc, treats, is_a



Outline

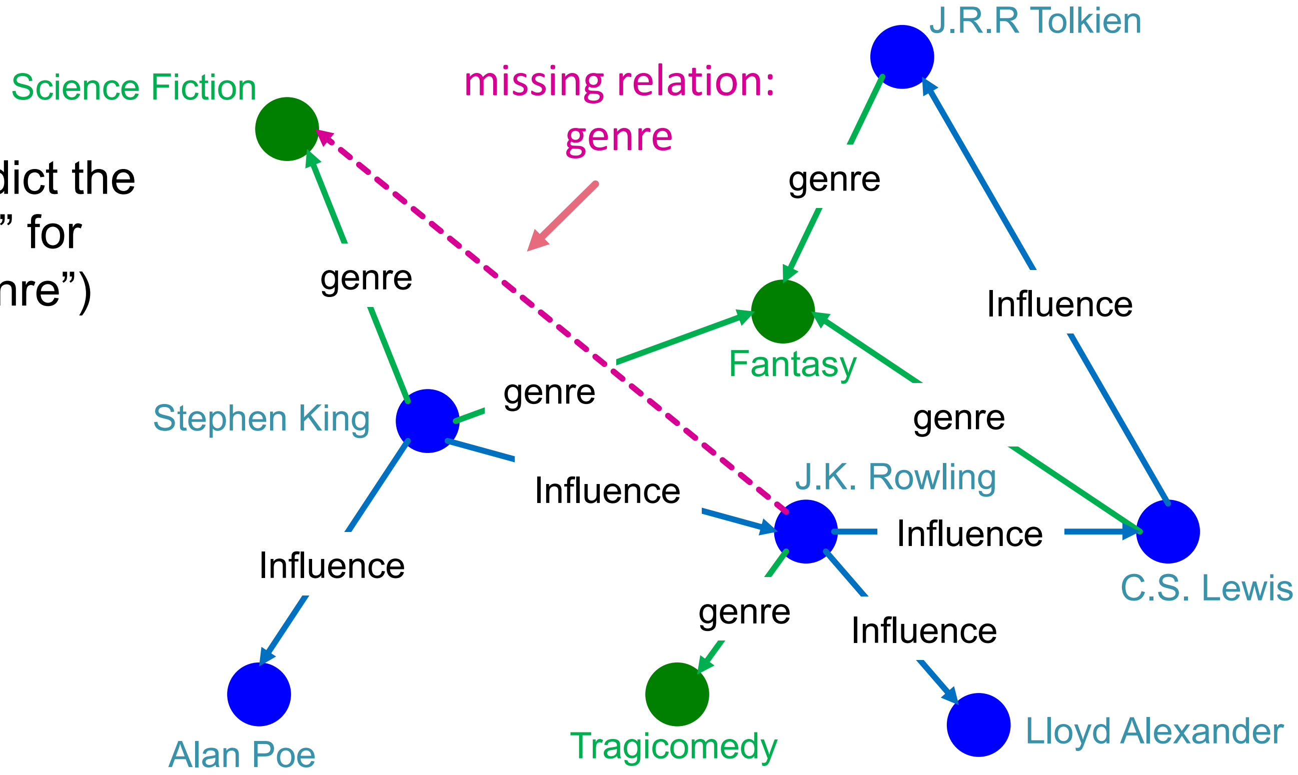
- Overview
- **Knowledge Graph Completion (Link Prediction)**
- Reasoning on Knowledge Graphs

Recap: KG Completion Task

Given an enormous KG, can we complete the KG?

- For a given (**head**, **relation**), we predict missing **tails**.
  - (Note this is slightly different from link prediction task)

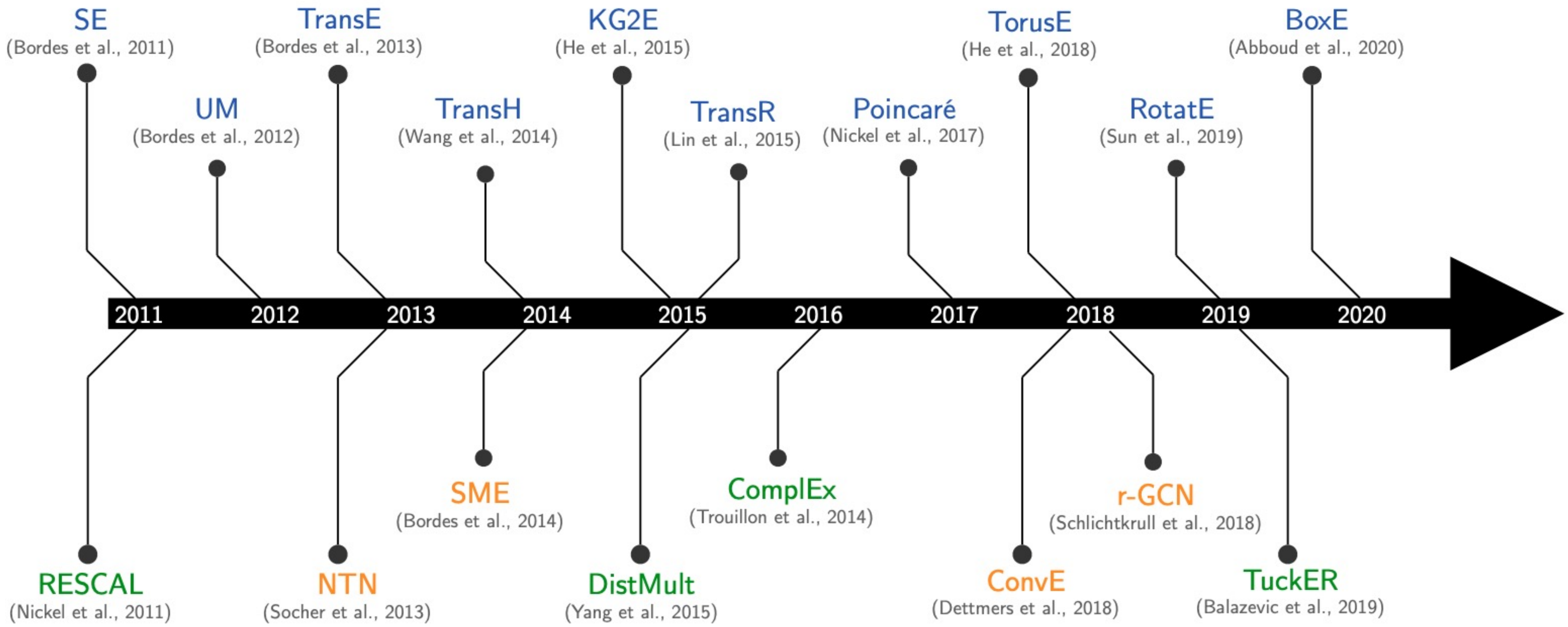
Example task: predict the tail "Science Fiction" for ("J.K. Rowling", "genre")



Recap: 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 embedding space $\mathbb{R}^d$
    - Associate entities and relations with **shallow embeddings**
      - **Note we do not learn a GNN here!**
  - Given a 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 score $f_r(h, t)$?
      - Score f_r is high if (h, r, t) exists, else f_r is low

Many KG Embedding Methods



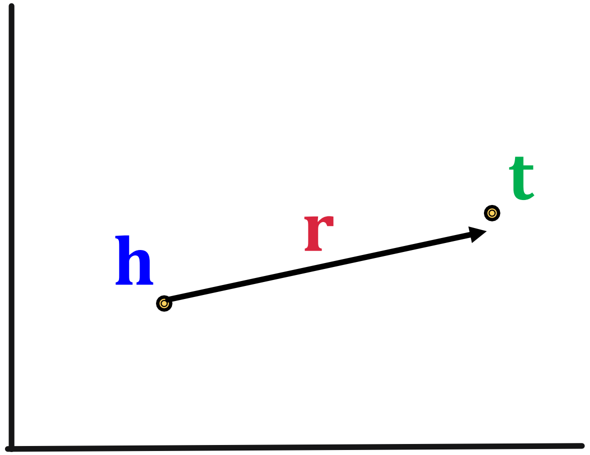
TransE for KG Completion

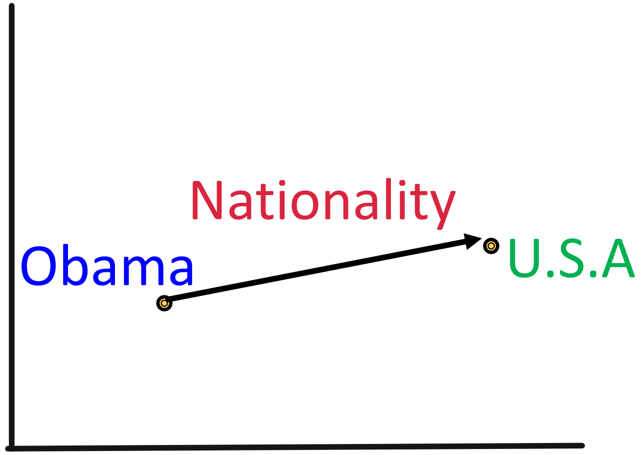
- **Intuition: Translation**
  For a triple (h, r, t), let $\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathbb{R}^d$ be embedding vectors.

embedding vectors will appear in boldface

- **TransE:** $\mathbf{h} + \mathbf{r} \approx \mathbf{t}$ if the given link exists else $\mathbf{h} + \mathbf{r} \neq \mathbf{t}$

Entity scoring function: $f_r(h, t) = -||\mathbf{h} + \mathbf{r} - \mathbf{t}||$





Bordes et al., Translating embeddings for modeling multi-relational data, NeurIPS 2013.

What to learn: entity and relation embeddings
Training data: observed triples

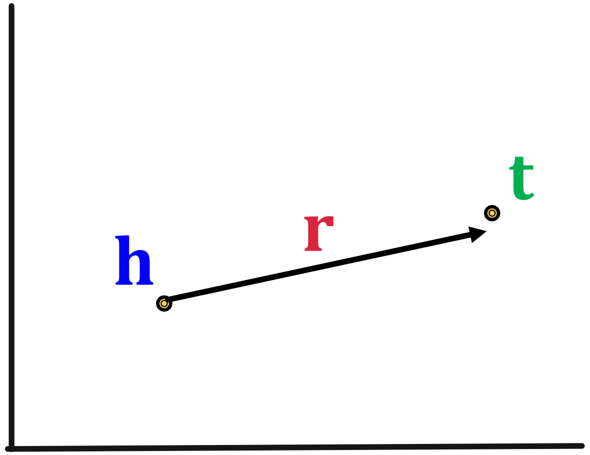
TransE for KG Completion

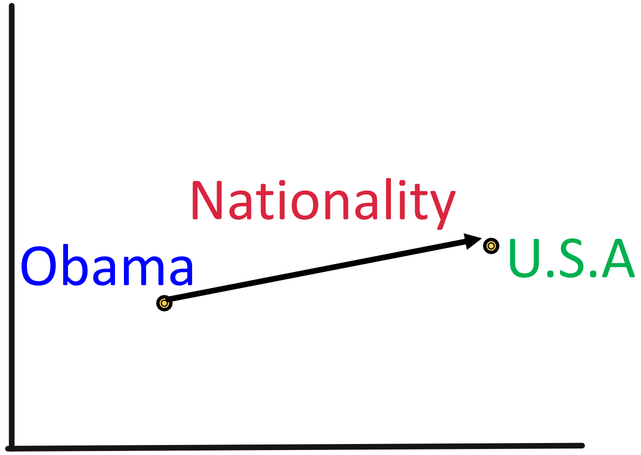
- **Intuition: Translation**
  For a triple (h, r, t), let $\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathbb{R}^d$ be embedding vectors.

embedding vectors will appear in boldface

- **TransE:** $\mathbf{h} + \mathbf{r} \approx \mathbf{t}$ if the given link exists else $\mathbf{h} + \mathbf{r} \neq \mathbf{t}$

Entity scoring function: $f_r(h, t) = -||\mathbf{h} + \mathbf{r} - \mathbf{t}||$





Bordes et al., *Translating embeddings for modeling multi-relational data*, NeurIPS 2013.

Connectivity Patterns in KG

- **Relations in a heterogeneous KG have different properties:**
  - Example:
    - **Symmetry:** If the edge $(h, \text{"Roommate"}, t)$ exists in KG, then the edge $(t, \text{"Roommate"}, h)$ should also exist.
    - **Inverse relation:** If the edge $(h, \text{"Advisor"}, t)$ exists in KG, then the edge $(t, \text{"Advisee"}, h)$ should also exist.
- Can we **categorize** these relation patterns?
- **Are KG embedding methods (e.g., TransE) expressive enough to model these patterns?**

Four Relationship Patterns

- **Symmetric (Antisymmetric) Relations:**

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

  - **Example**:
    - Symmetric: Family, Roommate
    - Antisymmetric: Hypernym (a word with a broader meaning: poodle vs. dog)
- **Inverse Relations:**

  $$r_2(h,t) \Rightarrow r_1(t,h)$$

  - **Example** : (Advisor, Advisee)
- **Composition (Transitive) 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 relations:**

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

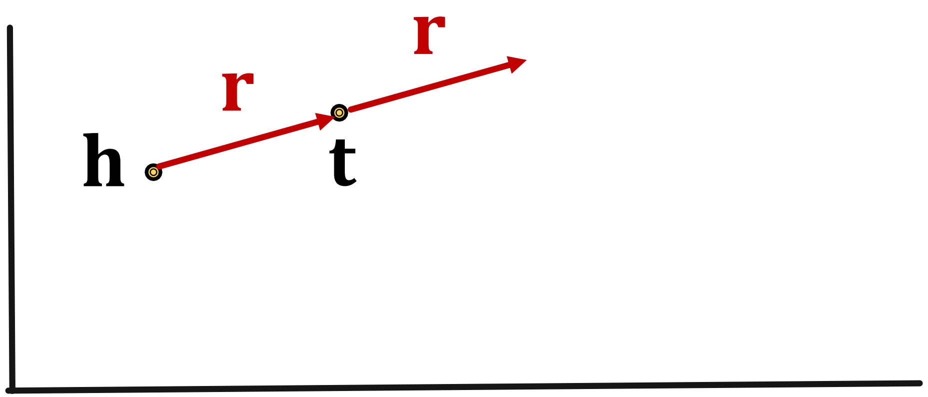
  - **Example**: r is "StudentsOf"

Antisymmetric Relations in TransE

- **Antisymmetric Relations:**

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

  - **Example**: Hypernym (a word with a broader meaning: poodle vs. dog)
- **TransE** can model antisymmetric relations ✓
  - $\mathbf{h}+\mathbf{r}=\mathbf{t}$, but $\mathbf{t}+\mathbf{r}\neq\mathbf{h}$

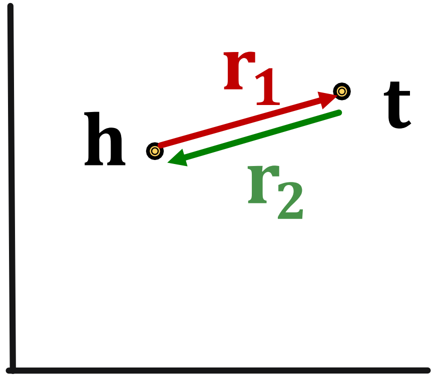


Inverse Relations in TransE

- **Inverse Relations:**

$$r_2(h,t) \Rightarrow r_1(t,h)$$

  - **Example** : (Advisor, Advisee)
- TransE can model inverse relations ✓
  - $\mathbf{h} + \mathbf{r}_2 = \mathbf{t}$, we can set $\mathbf{r}_1 = -\mathbf{r}_2$



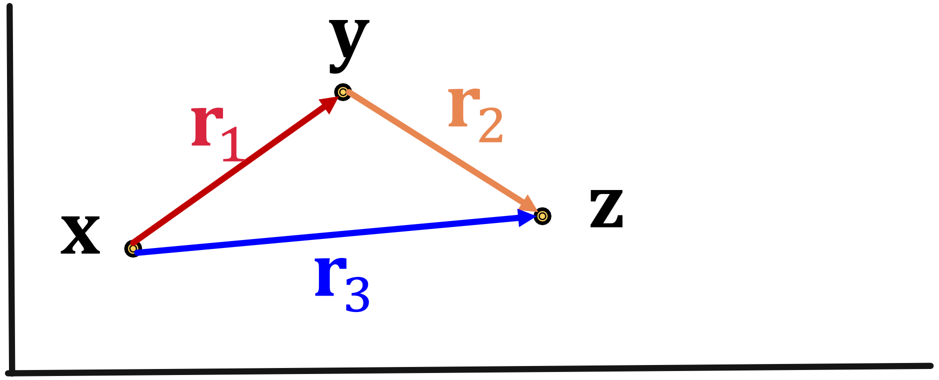
Composition in TransE

- **Composition (Transitive) 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.
- **TransE** can model composition relations ✓

  $$\mathbf{r}_3 = \mathbf{r}_1 + \mathbf{r}_2$$



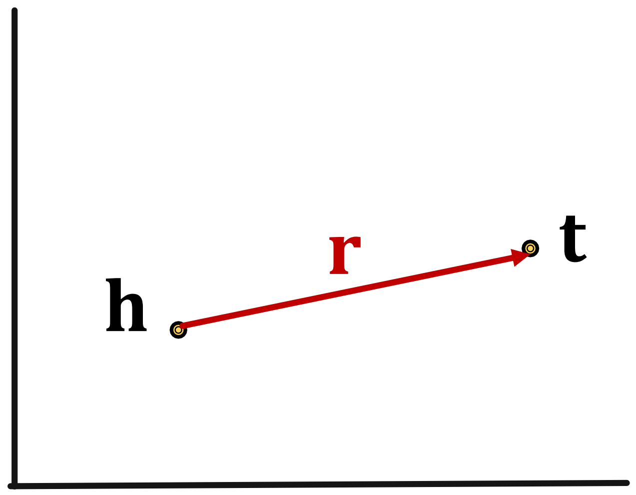
Limitations of TransE: Symmetric Relations

- **Symmetric Relations:**

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

  - **Example**: Family, Roommate
- **TransE cannot** model symmetric relations ✘

only if $\mathbf{r} = 0, \ \mathbf{h} = \mathbf{t}$

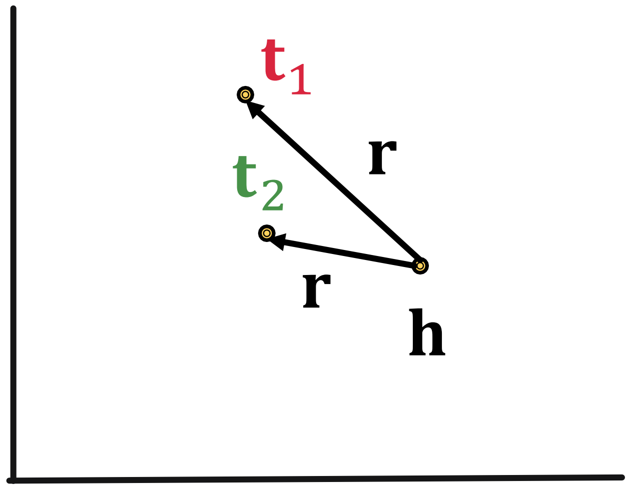


For all h, t that satisfy $r(h,t)$, $r(t,h)$ is also True, which means $||\mathbf{h} + \mathbf{r} - \mathbf{t}|| = 0$ and $||\mathbf{t} + \mathbf{r} - \mathbf{h}|| = 0$. Then $\mathbf{r} = 0$ and $\mathbf{h} = \mathbf{t}$, however h and t are two different entities and should be mapped to different locations.

Limitations of TransE: 1-to-N Relations

- **1-to-N Relations:**
  - **Example**: (h, r, t_1) and (h, r, t_2) both exist in the knowledge graph, e.g., r is "StudentsOf"
- **TransE cannot** model 1-to-N relations ✘
  - $\mathbf{t}_1$ and $\mathbf{t}_2$ 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!



KG Completion Methods

Model	Score	Embedding	Sym.	Antisym.	Inv.	Compos.	1-to-N
TransE	$-\|\mathbf{h}+\mathbf{r}-\mathbf{t}\|$	$\mathbf{h},\mathbf{t},\mathbf{r}\in\mathbb{R}^k$	✗	✓	✓	✓	✗
TransR	$-\|\boldsymbol{M}_r\mathbf{h}+\mathbf{r}-\boldsymbol{M}_r\mathbf{t}\|$	$\mathbf{h},\mathbf{t}\in\mathbb{R}^k$, $\mathbf{r}\in\mathbb{R}^d$, $\boldsymbol{M}_r\in\mathbb{R}^{d\times k}$	✓	✓	✓	✓	✓
DistMult	$<\mathbf{h},\mathbf{r},\mathbf{t}>$	$\mathbf{h},\mathbf{t},\mathbf{r}\in\mathbb{R}^k$	✓	✗	✗	✗	✓
ComplEx	$\mathrm{Re}(<\mathbf{h},\mathbf{r},\bar{\mathbf{t}}>)$	$\mathbf{h},\mathbf{t},\mathbf{r}\in\mathbb{C}^k$	✓	✓	✓	✗	✓
RotateE	$-\|\mathbf{h}\circ\mathbf{r}-t\|$	$\mathbf{h},\mathbf{t},\mathbf{r}\in\mathbb{C}^k$	✓	✓	✓	✓	✓

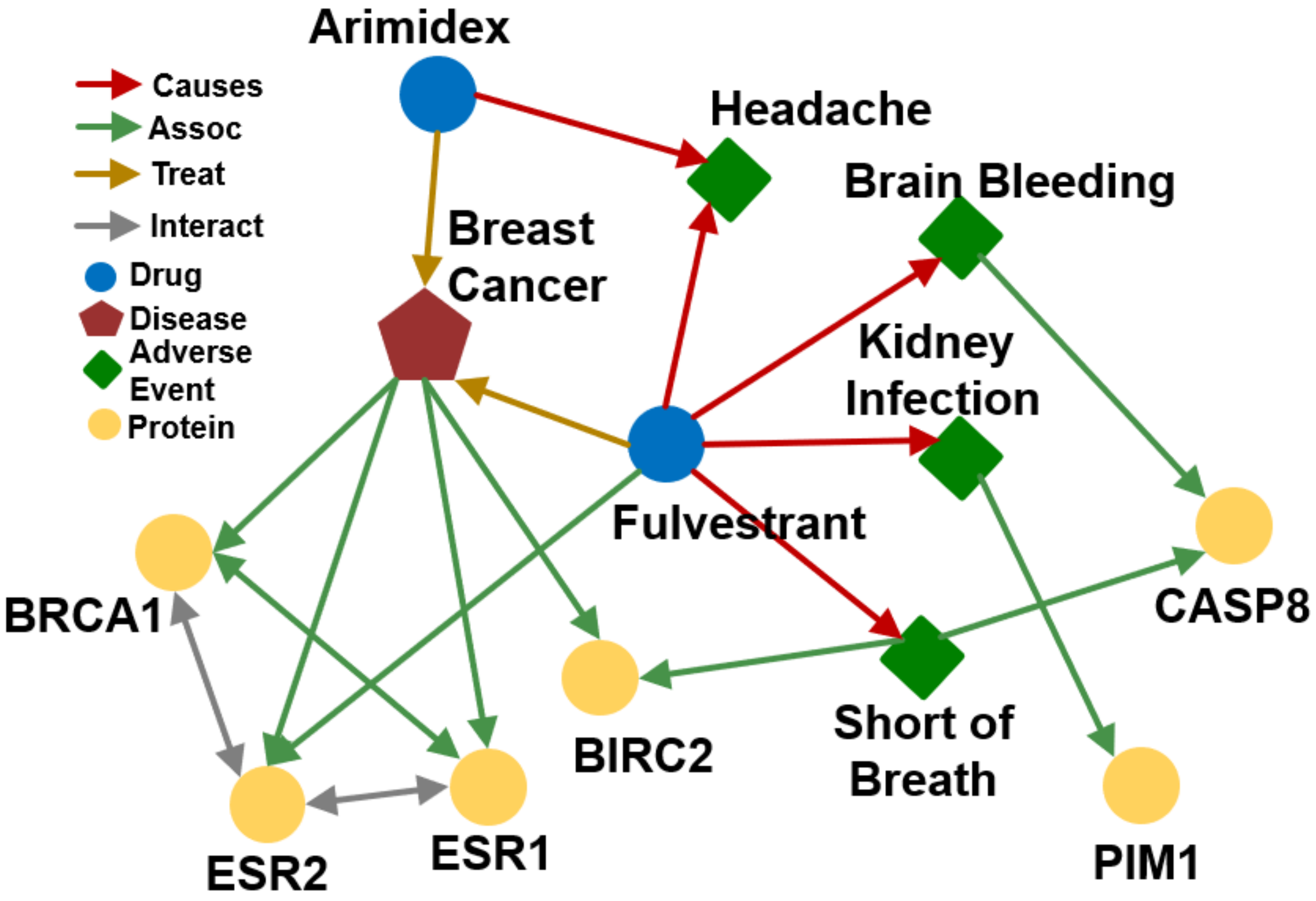
Outline

- Overview
- Knowledge Graph Completion (Link Prediction)
- **Reasoning on Knowledge Graphs**

Reasoning over KGs

- **Goal**:
  - How to perform multi-hop reasoning over KGs?
- **Reasoning over Knowledge Graphs**
  - Answering multi-hop queries
    - Path Queries
    - Conjunctive Queries
  - Query2Box

Example KG: Biomedicine

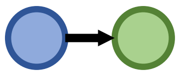


Predictive Queries on KG

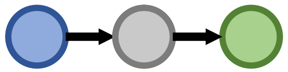
Can we do multi-hop reasoning, i.e., answer complex queries on an incomplete, massive KG?

Query Types	Examples: Natural Language Question, Query
One-hop Queries	What adverse event is caused by Fulvestrant? (e:Fulvestrant, (r:Causes))
Path Queries	What protein is associated with the adverse event caused by Fulvestrant? (e:Fulvestrant, (r:Causes, r:Assoc))
Conjunctive Queries	What is the drug that treats breast cancer and caused headache? ((e:BreastCancer, (r:TreatedBy)), (e:Migraine, (r:CausedBy))

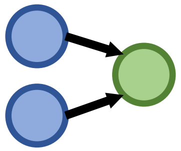
In this lecture, we only focus on answering queries on a KG!
The notation will be detailed next.



One-hop Queries



Path Queries



Conjunctive Queries

Predictive One-hop Queries

- **We can formulate knowledge graph completion problems as answering one-hop queries.**

- **KG completion:** Is link (h, r, t) in the KG?

$$\Updownarrow$$

- **One-hop query:** Is t an answer to query (h, r)?
  - **For example:** What side effects are caused by drug Fulvestrant?

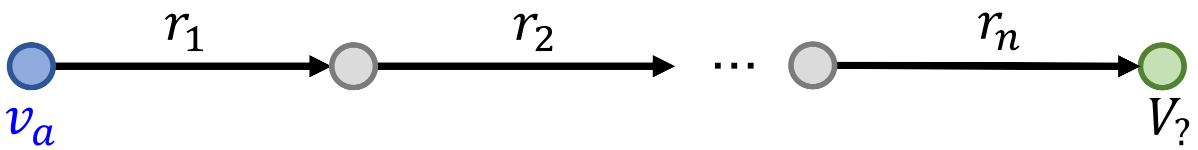
Path Queries

- Generalize one-hop queries to path queries by **adding more relations on the path**.
- An n-hop path query q can be represented by

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

  - v_a is an "anchor" entity,
  - Let answers to q in graph G be denoted by $[\![q]\!]_G$.

Query Plan of q:

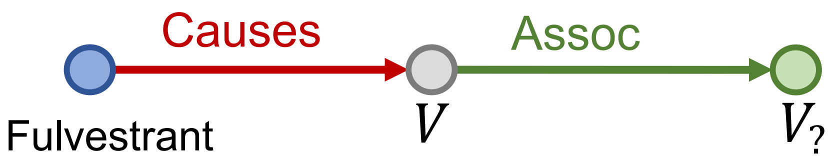


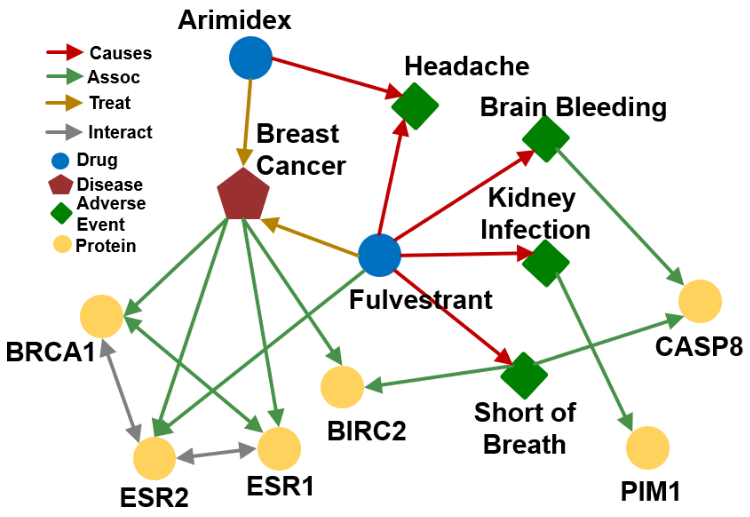
Query plan of path queries is a chain.

Path Queries

Question: *“What proteins are **associated** with adverse events **caused** by **Fulvestrant**?”*

- v_a is **e:Fulvestrant**
- (r_1, r_2) is (**r:Causes**, **r:Assoc**)
- **Query:** (**e:Fulvestrant**, (**r:Causes**, **r:Assoc**))



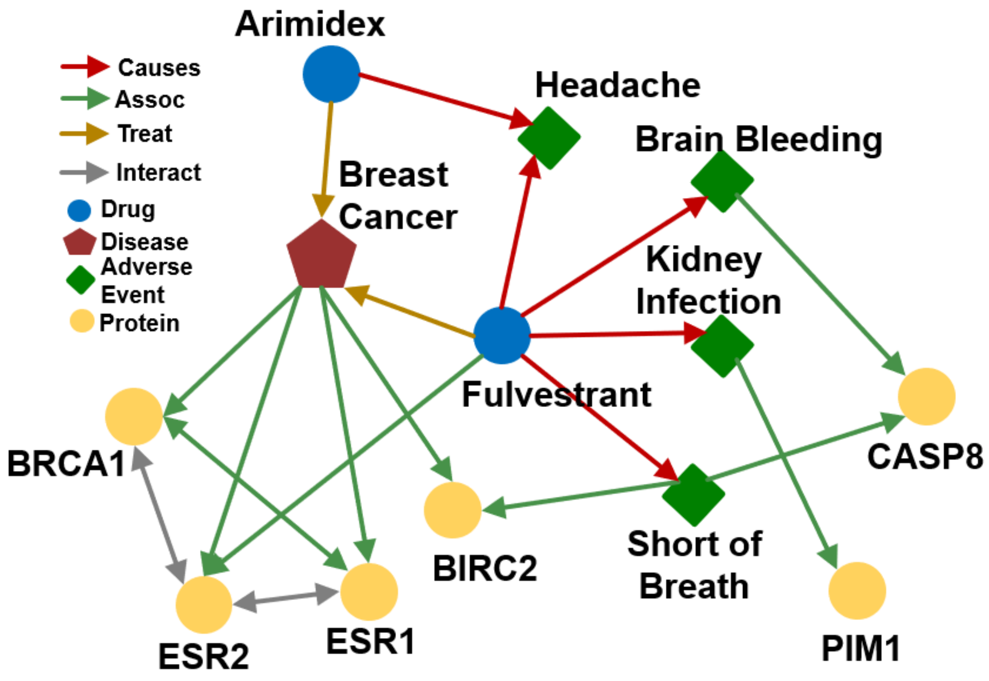


Path Queries

Question: *"What proteins are **associated** with adverse events **caused** by **Fulvestrant**?"*

- **Query: (e:Fulvestrant, (r:Causes, r:Assoc))**

Given a KG, how to answer a path query?



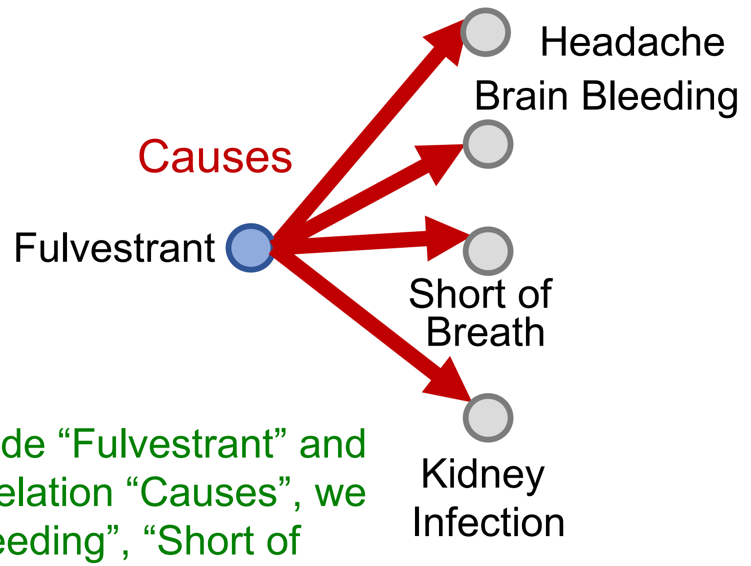
Traversing Knowledge Graphs

- **We answer path queries by traversing the KG:** *"What proteins are **associated** with adverse events **caused** by **Fulvestrant**?"*
- **Query: (e:Fulvestrant, (r:Causes, r:Assoc))**

Start from the **anchor node** (Fulvestrant).

Traversing Knowledge Graphs

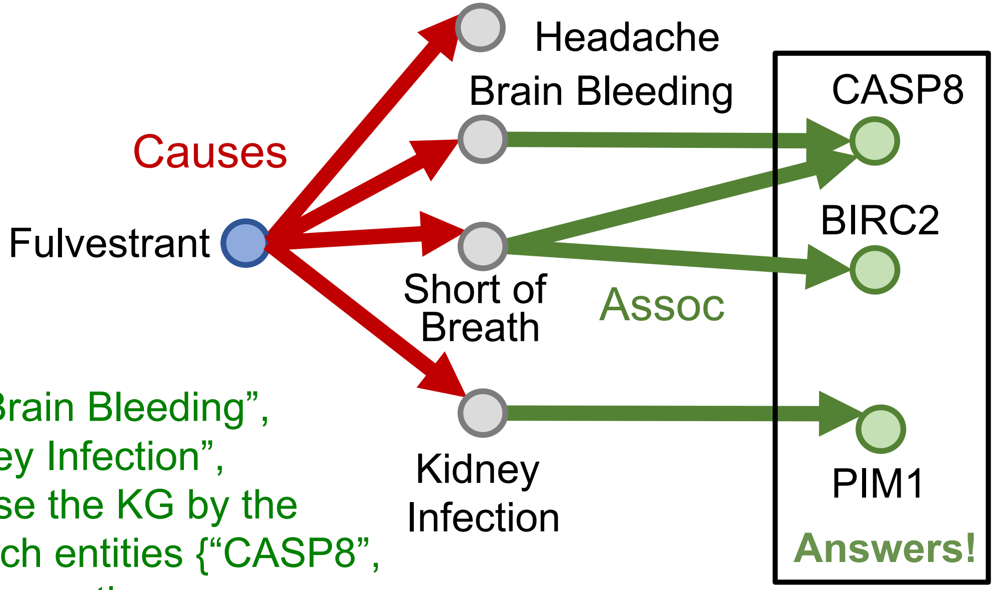
- **We answer path queries by traversing the KG:** *"What proteins are **associated** with adverse events **caused** by **Fulvestrant**?"*
- **Query: (e:Fulvestrant, (r:Causes, r:Assoc))**



Start from the anchor node "Fulvestrant" and traverse the KG by the relation "Causes", we reach entities {"Brain Bleeding", "Short of Breath", "Kidney Infection", "Headache"}.

Traversing Knowledge Graphs

- **We answer path queries by traversing the KG:** *"What proteins are **associated** with adverse events **caused** by **Fulvestrant**?"*
- **Query: (e:Fulvestrant, (r:Causes, r:Assoc))**



Start from the nodes {"Brain Bleeding", "Short of Breath", "Kidney Infection", "Headache"} and traverse the KG by the relation "Assoc", we reach entities {"CASP8", "BIRC2", "PIM1"}. These are the answers.

However, KGs are incomplete

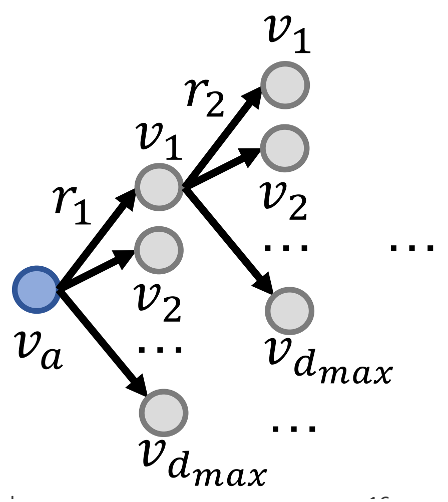
- **Answering queries seems easy: Just traverse the graph.**
- **But KGs are incomplete and unknown:**
  - Many relations between entities are missing or are incomplete
    - For example, we lack all the biomedical knowledge
    - Enumerating all the facts takes non-trivial time and cost, we cannot hope that KGs will ever be fully complete
- **Due to KG incompleteness, one is not able to identify all the answer entities**

Can KG Completion Help?

Can we first do KG completion and then traverse the completed (probabilistic) KG?

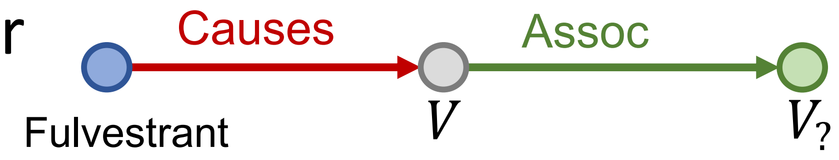
- **No!** The "completed" KG is a **dense graph**!
  - Most (h, r, t) triples (edge on KG) will have some non-zero probability.

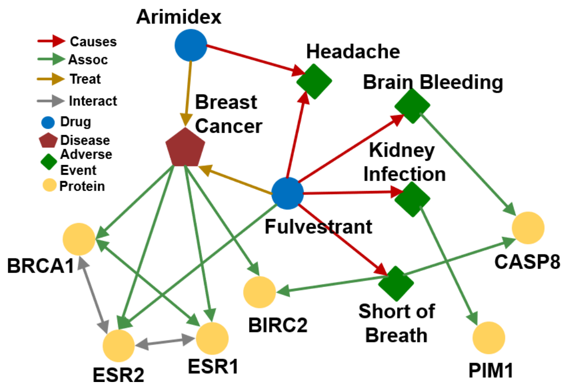
- Time complexity of traversing a dense KG is exponential as a function of the path length L: $O(d_{max}^{L})$



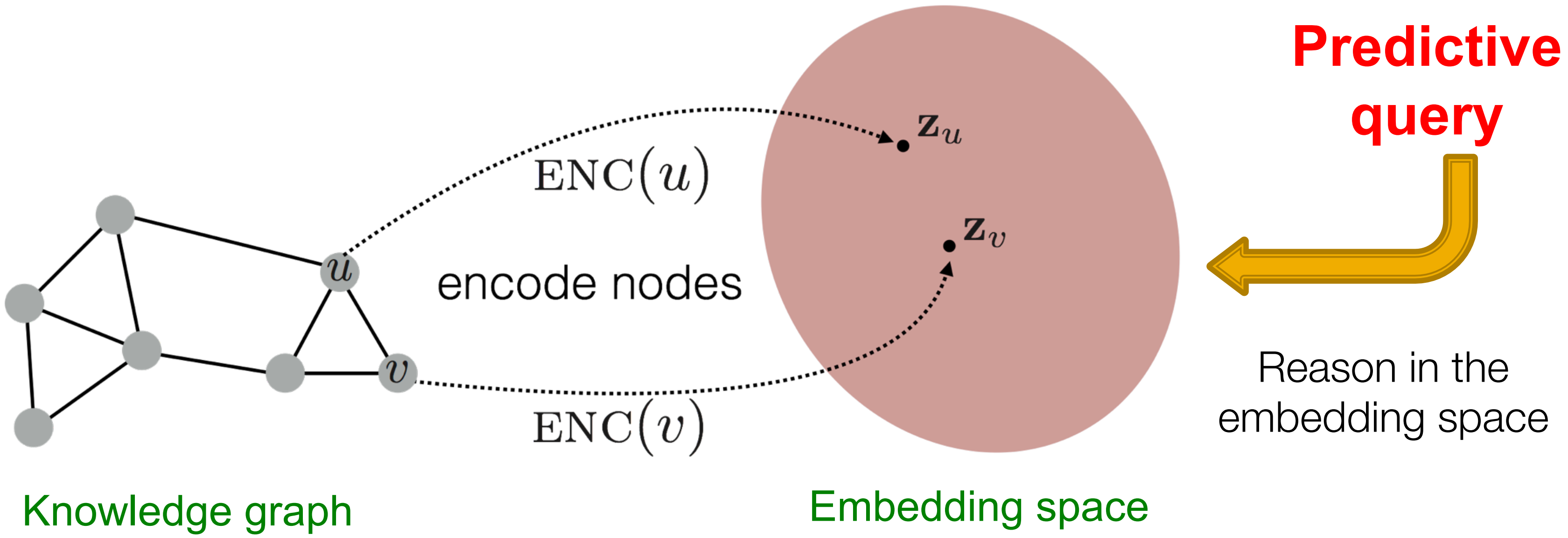
Task: Predictive Queries

- We need a way to answer path-based queries over an incomplete knowledge graph.
- We want our approach to implicitly impute and account for the incomplete KG.
- **Task: Predictive queries**
  - Want to be able to answer arbitrary queries while implicitly imputing for the missing information
  - **Generalization of the link prediction task**





A General Idea



Map queries into embedding space. **Learn to reason in that space**

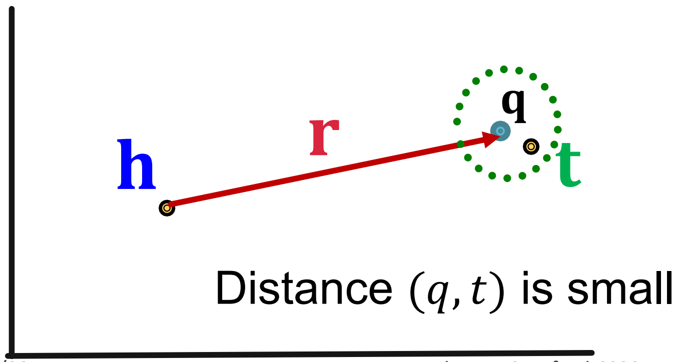
- Embed query into a single **point** in the Euclidean space: answer nodes are close to the query.
- **Query2Box:** Embed query into a hyper-rectangle (**box**) in the Euclidean space: answer nodes are enclosed in the box.

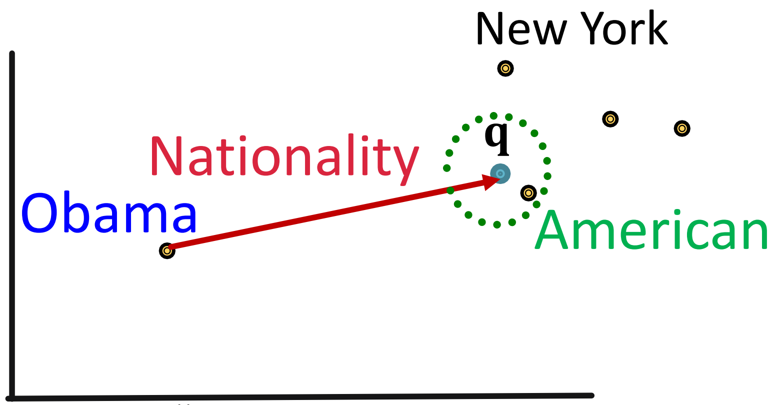
[Embedding Logical Queries on Knowledge Graphs. Hamilton, et al., NeurIPS 2018]
[Query2box: Reasoning over Knowledge Graphs in Vector Space Using Box Embeddings. Ren, et al., ICLR 2020]

Traversing KG in Vector Space

- **Key idea: Embed queries!**
  - Generalize **TransE** to multi-hop reasoning.
  - **Recap: TransE:** Translate $\mathbf{h}$ to $\mathbf{t}$ using $\mathbf{r}$ with score function $f_r(h,t) = -||\mathbf{h} + \mathbf{r} - \mathbf{t}||$.
  - Another way to interpret this is that:
    - **Query embedding**: $\mathbf{q} = \mathbf{h} + \mathbf{r}$
    - Goal: **query embedding** $\mathbf{q}$ is **close** to the **answer embedding** $\mathbf{t}$

$$f_q(t) = -\|\mathbf{q} - \mathbf{t}\|$$



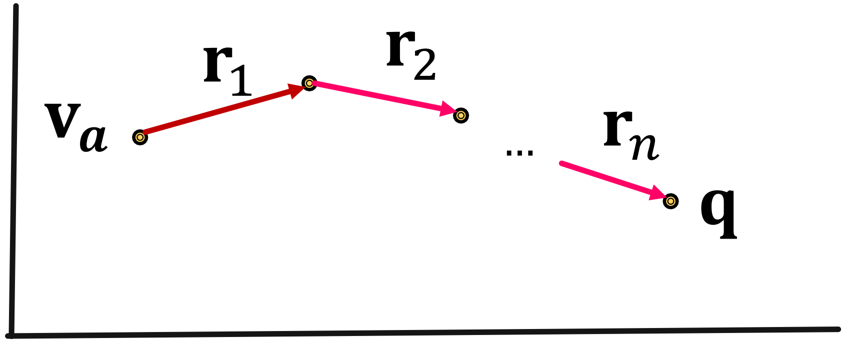


Guu, et al., *Traversing knowledge graphs in vector space, EMNLP 2015.*

Traversing KG in Vector Space

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

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



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

- The embedding process **only involves vector addition**, **independent of # entities** in the KG!

Guu, et al., Traversing knowledge graphs in vector space, EMNLP 2015.

Traversing KG in Vector Space

Embed path queries in vector space.

- **Question:** *"What proteins are **associated** with adverse events **caused** by **Fulvestrant**?"*
- **Query:** (**e:Fulvestrant**, (**r:Causes** , **r:Assoc**))

Follow the query plan:

Query Plan | **Embedding Process**

Guu, et al., Traversing knowledge graphs in vector space, EMNLP 2015.

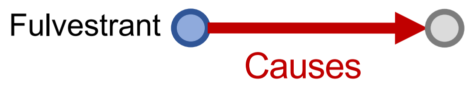
Traversing KG in Vector Space

Embed path queries in vector space.

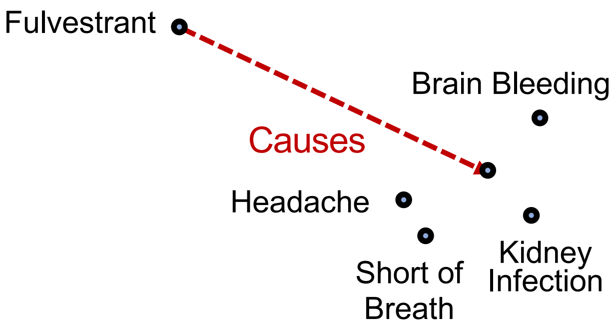
- **Question:** "*What proteins are **associated** with adverse events **caused** by **Fulvestrant**?*"
- **Query:** (e:Fulvestrant, (r:Causes , r:Assoc))

Follow the query plan:

Query Plan



Embedding Process



Guu, et al., Traversing knowledge graphs in vector space, EMNLP 2015.

Traversing KG in Vector Space

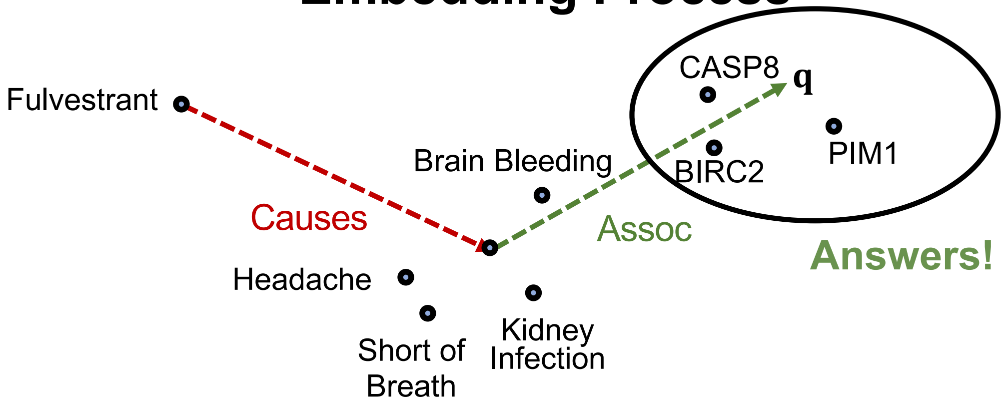
Embed path queries in vector space.

- **Question:** "*What proteins are **associated** with adverse events **caused** by **Fulvestrant**?*"
- **Query:** (e:Fulvestrant, (r:Causes , r:Assoc))

Follow the query plan:

Query Plan

Embedding Process



Guu, et al., Traversing knowledge graphs in vector space, EMNLP 2015.

Traversing KG in Vector Space

Insights:

- We can train **TransE** to optimize knowledge graph completion objective (Lecture 11)

- Since **TransE** can naturally handle **compositional relations**, it can handle path queries by translating in the latent space **for multiple hops using addition of relation embeddings**.

Questions?