Reasoning in Knowledge Graphs Using Embeddings

Hongyu Ren
Stanford University

Knowledge Graphs

Knowledge Graphs are heterogeneous graphs

- $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{R})$
- $\mathcal{V}$: node set; $\mathcal{E}$: edge set; $\mathcal{R}$: relation set;
- Each edge $e \in \mathcal{E}$ represents a fact $(v_s, r, v_o)$ over subject $v_s$ and object $v_o \in \mathcal{V}$ with relation $r \in \mathcal{R}$.
- KGs capture real world facts.

Examples:

- WIKIDATA
- Freebase

Hongyu Ren, Stanford University
Traditional Tasks

Knowledge Graph Completion
Given the existing edge set $\mathcal{E}$, predict the missing object for a given triplet $(v_s, r, v_t)$
Example: Is Louvre an object to (James, Visited, ?)
Traditional Tasks

Knowledge Graph Completion

Given the existing edge set $\mathcal{E}$, predict the missing object for a given triplet $(v_s, r, v_t)$

Example: Is Louvre an object to (James, Visited, ?)

Solution: TransE

TransE learns an entity embedding matrix and a relation embedding matrix, such that given a triplet $e = (v_s, r, v_t) \in \mathcal{E}$, we have $v_s + r = v_t$, where $v_s, r, v_t \in \mathbb{R}^d$.

Our Task: Multi-hop Reasoning

Answering Acyclic Conjunctive Queries

- A subset of **first-order logic** with existential quantifier (∃), conjunction (∧)

“Where did all Canadian citizens with Turing Award graduate?”

**Input:** \( q = V_? \cdot \exists V : \text{Win}(\text{Turing Award}, V) \land \text{Citizen}(\text{Canada}, V) \land \text{Graduate}(V, V_?) \)

**Output:** \{Edinburgh, Cambridge, McGill\}
Examples

Applications in biomedicine:

\[ C_?: \exists P : \text{ASSOC}(d_1, P) \land \text{ASSOC}(d_2, P) \land \text{TARGET}(P, C_?) \]

“Predict drugs \( C_? \) that might target proteins that are associated with the given disease nodes \( d_1 \) and \( d_2 \)”

Bio KG
Why is it hard?

Key challenges:
Big graphs and queries can involve noisy and unobserved data!

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

98.12% of movies recorded in 2020 have no information about their cost!

[Distant Supervision for Relation Extraction with an Incomplete Knowledge Base. Min et al., NAACL 2013]
[Enriching Wikidata with Linked Open Data. Zhang et al., 2022]
Our General Idea

- **Reason in the embedding space:** Map the query and entities into embedding space such that answer entity embeddings are close to the query embedding (contrastive learning).
  - Low-dimensional embedding allows generalization.

[Embedding Logical Queries on Knowledge Graphs, Hamilton, et al., NeurIPS 2018]
How to Embed the Query?

- Embed the query following the query plan!

“Where did all Canadian citizens with Turing Award graduate?”

**Input:** \( q = V \cdot \exists V : \text{Win}(\text{Turing Award}, V) \land \text{Citizen}(\text{Canada}, V) \land \text{Graduate}(V, V) \)

**Query Plan**

- Traversing query plan from leaves (anchors) until the root (answers)
Our Model: Query2box

Query2Box embedding:
Embed queries with hyper-rectangles (boxes): $q = (\text{Cen}(q), \text{Off}(q))$.

[Probabilistic Embedding of Knowledge Graphs with Box Lattice Measures, Vilnis, et al., ACL 2018]
Example: Query2Box

“Where did Canadian citizens with Turing Award graduate?”

[Query2box: Reasoning over Knowledge Graphs in Vector Space Using Box Embeddings. Ren et al., ICLR 2020]
Benefits of Query2Box

Scalability and efficiency:
- Any query can be reduced to a couple of matrix operations and a single k-nearest neighbor search

Generality:
- We can answer any query (even those we have never seen before)

Robustness to noise:
- Graph can contain missing and noisy relationships
Distributed Paradigm

Pipelined Asynchronous Design

First framework that scales reasoning algorithms to extremely large KGs (86m nodes, 338m edges)

[SMORE: Knowledge Graph Completion and Multi-hop Reasoning in Massive Knowledge Graphs. Ren et al., KDD 2022]
SMORE on Massive KGs

- SMORE supports 6 different link prediction + multi-hop reasoning methods
- Prior implementation runs out of GPU memory and time limit on large KGs
- SMORE enables (almost) graph-size agnostic speed and GPU memory usage

- SMORE enables almost linear multi-GPU speedup

---

Hongyu Ren, Stanford University
Application (1): KG Question Answering

KGQA: Answering natural language questions over massive KGs

Question: Who are Canadian Turing Award winners?

Query Plan:
- Turing Award
- Citizen
- Canada

Box Embedding:
- Win
- LeCun
- Hinton
- Bengio
- Trudeau
- Bieber
- Citizen
- Canada

Our method: Latent Execution-Guided Reasoning

Given a question, we iteratively do:

(start from an empty tree with mentioned entities as root)

- **Execution-guided synthesis**: synthesize one step (grow the query plan) based on the current query embedding.
- **Latent query execution**: execute the new step in the embedding space
Application (1): KG Question Answering

Our method: Latent Execution-Guided Reasoning (LEGO)

<table>
<thead>
<tr>
<th>Questions</th>
<th>MetaQA</th>
<th>WebQuestion</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train</td>
<td>96,106</td>
<td>2,848</td>
</tr>
<tr>
<td>Dev</td>
<td>9,992</td>
<td>250</td>
</tr>
<tr>
<td>Test</td>
<td>9,947</td>
<td>1,639</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>KG</th>
<th>Entities</th>
<th>Relations</th>
<th>Observed Edges</th>
<th>Missing Edges</th>
</tr>
</thead>
<tbody>
<tr>
<td>MetaQA</td>
<td>43k</td>
<td>18</td>
<td>134k</td>
<td>133k</td>
</tr>
<tr>
<td>WQSP</td>
<td>409k</td>
<td>1,836</td>
<td>2m</td>
<td>2m</td>
</tr>
</tbody>
</table>

MetaQA

WebQuestion (WQSP)

[Hongyu Ren, Stanford University]
Application (2): Fact Ranking

- Facts/Answers to queries are of different importance/uncertainty to users.

What's the occupation of Selena Gomez?

Synthesize Query

get(Q83287, P106)

Graph Traversal

Answers: {singer, songwriter, record producer, child actor, producer…}

Query Embeddings

Achieve a ranking of the answers
Conclusion

- Box embeddings for answering logical queries on knowledge graphs
- Scalable query embeddings on massive KGs
- Applications towards neural query engine

Knowledge graph

Embedding space

Query $q$

Reason in the embedding space

encode nodes

$\text{ENC}(u)$

$\text{ENC}(v)$

$z_u$

$z_v$

$z_q$
Query2box: Reasoning over Knowledge Graphs in Vector Space Using Box Embeddings
Hongyu Ren*, Weihua Hu*, Jure Leskovec
ICLR 2020

Beta Embeddings for Multi-Hop Logical Reasoning in Knowledge Graphs
Hongyu Ren, Jure Leskovec
NeurIPS 2020

Latent Execution-Guided Reasoning for Multi-Hop Question Answering on Knowledge Graphs
Hongyu Ren, Hanjun Dai, Bo Dai, Xinyun Chen, Michihiro Yasunaga, Haitian Sun, Dale Schuurmans, Jure Leskovec, Denny Zhou
ICML 2021

SMORE: Knowledge Graph Completion and Multi-hop Reasoning in Massive Knowledge Graphs
Hongyu Ren*, Hanjun Dai*, Bo Dai, Xinyun Chen, Denny Zhou, Jure Leskovec, Dale Schuurmans
KDD 2022