Differentiable Symbolic Reasoning with Graph Neural Networks

Ziniu Hu
4-th year Ph.D at UCLA

Advisor: Prof. Yizhou Sun
From Perception to **Reasoning**

- **System 1 (Perception)**
  - Intuitive and unconscious that human can’t explain verbally.
  - Most e2e DL system handles
  - React FAST by human

- **System 2 (Reasoning)**
  - The prediction is based on high-level semantic concepts.
  - Require logical, algorithmic planning and reasoning over symbolic space.
  - React Slow by human

---

From Bengio’s NeurIPS’2019 Keynote: From System 1 Deep Learning to System 2 Deep Learning,
Challenges of Neural-Symbolic Reasoning

- Most Symbolic Systems are non-differentiable, e.g., SAT Solver, Logic Inference, etc.
- The system cannot learn end-to-end, and still require an intermediate discrete parsing step, which requires lots of annotation.
My Solution: Differentiable Reason with Graph

• Graph is an expressive abstraction to model symbolic knowledge.

• Graph Neural Network (GNN) can learn representation from structured data.
• GNN supports differentiable reasoning
Outline of my Previous Works

• Symbolic Knowledge Modelling via Relational GNN

• Symbolic Knowledge Reasoning over Relational Graph
Relational Graph encodes World Knowledge

- Relational graphs can model complex systems
  - Nodes are labeled with multiple types
  - Edges between nodes have multiple relationships

Question: what episode was mike kelley the writer of

Question: what tv show did grant show play on in 2008
Our Solution: **Heterogeneous Graph Transformer**

- **Meta-Relation-based Parametrization**
  - Leverage meta relation `<source node type, edge type, target node type>` to parameterize attention and message passing weight.
  - Can automatically learn and extract meaningful “meta paths”

*Heterogeneous Graph Transformer. Ziniu Hu, Yuxiao Dong, Kuansan Wang, Yizhou Sun. WWW 2020.*
Label-Scarcity Challenge

• Though the HGT model incorporates the symbolic knowledge into model design, there lack sufficient labelled data for important reasoning tasks, such as QA and drug prediction:
  • The largest QA dataset only has around 10k labelled QA pairs, and only cover a small portion of relational facts

Labeled Data
Expensive, Scarce

Unlabeled Data
Accessible, Abundant (1000 X more)
Graph as self-supervised signal?

GPT-GNN: Generative Pre-training of Graph Neural Networks. Ziniu Hu, Yuxiao Dong, Kuansan Wang, Kai-Wei Chang, Yizhou Sun. KDD 2020.
Pre-Training GNN via Generation

- Model the graph distribution $p(G; \theta)$ by learning to reconstruct the attributed graph.

GPT-GNN: Generative Pre-training of Graph Neural Networks. Ziniu Hu, Yuxiao Dong, Kuansan Wang, Kai-Wei Chang, Yizhou Sun. KDD 2020.
Model the graph distribution $p(G; \theta)$ earning to reconstruct the attributed graph.

We factorize the graph likelihood into two terms:

1. **Attribute ($X$) Generation**

2. **Edge ($E$) Generation**

$$
\log p_{\theta}(X, E) = \sum_{i=1}^{|V|} \log p_{\theta}(X_i, E_i \mid X_{<i}, E_{<i}).
$$

$$
p_{\theta}(X_i, E_i \mid X_{<i}, E_{<i}) = \sum_{o} p_{\theta}(X_i, E_i, o \mid E_{i,o}, X_{<i}, E_{<i}) \cdot p_{\theta}(E_{i,o} \mid X_{<i}, E_{<i})
= \mathbb{E}_o \left[ p_{\theta}(X_i, E_i, o \mid E_{i,o}, X_{<i}, E_{<i}) \right]
= \mathbb{E}_o \left[ p_{\theta}(X_i \mid E_{i,o}, X_{<i}, E_{<i}) \cdot p_{\theta}(E_{i,o} \mid E_{i,o}, X_{<i}, E_{<i}) \right].
$$

1) generate attributes 2) generate edges
Outline of my Previous Works

• Symbolic Knowledge Modelling via Relational GNN

• Symbolic Knowledge Reasoning over Relational Graph
Complex question requires relation reasoning

- Traditional Knowledge Base Question Answering (KBQA) systems seek to parse the query into a logical form.

- The logical query can then be directly executed on the KB to get answers.

- **Limitation**: Existing KB suffer from low coverage of entities and relations.
Our Solution: Teach LM to reason without KG

We extract all mentioning of target entity $t$ from the Wiki-page of $s$, and use the context passage as $\text{desc.}(s, t)$. 
Relational QA Pair Generation

**Question Template:**

\[ q(s, r, t) = [\text{MASK}(r)] \text{ of } [s] \text{ which } [\text{desc.}(t, s)]? \]

**True Passage:**

\[ p^+ = \text{desc.}(s, t) \]

**Question:** \([<\text{MASK}(r)>]\) of [Stephen Curry] which [the Splash Brothers are a duo of American basketball players consisting of Stephen Curry and Klay Thompson].

**True Passage:** ...Curry and teammate Klay Thompson have earned the nickname of the **Splash Brothers**...

**Answer (Target Entity):** Splash Brothers
Can we do more than relational Reasoning?

Query: List the presidents of European countries that have never held the World Cup

FOL Query: $q = \exists V : \text{Located}(\text{Europe}, V) \land \neg \text{Held}(\text{World Cup}, V) \land \text{President}(V, V)$

Computation Graph:

Fuzzy Logic based Logical Query Answering on Knowledge Graph. Xuelu Chen, Ziniu Hu, Yizhou Sun
Our Approach FuzzQE

- **Objective**
  - To implement logical operators in a more principled and learning-free manner
- **Our solution: Fuzzy Logic based Query Embedding (FuzzQE)**
  - Borrow the idea of fuzzy logic
  - Use fuzzy conjunction, disjunction, and negation operations to define the logical operators in vector space.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>GQE</td>
<td>✓ (✓)</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✓ (✗)</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>N/A</td>
</tr>
<tr>
<td>Query2Box</td>
<td>✓ (✓)</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓ (✗)</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>N/A</td>
</tr>
<tr>
<td>BetaE</td>
<td>✓ (✓)</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✓ (✗)</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓ (✓)</td>
</tr>
<tr>
<td>FuzzQE</td>
<td>✓ (✓)</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓ (✓)</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>
Experimental Results

- **Train with only KG Edges (No Complex Queries)**

Table 6.8: MRR results (%) of logical query embedding models that are trained with only link prediction. This task tests the ability of the model to generalize to arbitrary complex logical queries, when no complex logical query data is available for training. Avg\textsubscript{EPFO} and Avg\textsubscript{Neg} denote the average MRR on EPFO ($\exists$, $\land$, $\lor$) queries and queries containing negation respectively.

<table>
<thead>
<tr>
<th>Model</th>
<th>Avg\textsubscript{EPFO}</th>
<th>Avg\textsubscript{Neg}</th>
<th>1p</th>
<th>2p</th>
<th>3p</th>
<th>2i</th>
<th>3i</th>
<th>pi</th>
<th>ip</th>
<th>2u</th>
<th>up</th>
<th>2in</th>
<th>3in</th>
<th>inp</th>
<th>pin</th>
<th>pni</th>
</tr>
</thead>
<tbody>
<tr>
<td>FB15k-237</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GQE</td>
<td>17.7</td>
<td>N/A</td>
<td>41.6</td>
<td>7.9</td>
<td>5.4</td>
<td>25.0</td>
<td>33.6</td>
<td>16.3</td>
<td>10.9</td>
<td>11.9</td>
<td>6.2</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Query2Box</td>
<td>18.2</td>
<td>N/A</td>
<td>42.6</td>
<td>6.9</td>
<td>4.7</td>
<td>27.3</td>
<td>36.8</td>
<td>17.5</td>
<td>11.1</td>
<td>11.7</td>
<td>5.5</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>BetaE</td>
<td>15.8</td>
<td>0.5</td>
<td>37.7</td>
<td>5.6</td>
<td>4.4</td>
<td>23.3</td>
<td>34.5</td>
<td>15.1</td>
<td>7.8</td>
<td>9.5</td>
<td>4.5</td>
<td>0.1</td>
<td>1.1</td>
<td>0.8</td>
<td>0.1</td>
<td>0.2</td>
</tr>
<tr>
<td>FuzzQIE</td>
<td><strong>21.8</strong></td>
<td><strong>6.6</strong></td>
<td><strong>44.0</strong></td>
<td><strong>10.8</strong></td>
<td><strong>8.6</strong></td>
<td><strong>32.3</strong></td>
<td><strong>41.4</strong></td>
<td><strong>22.7</strong></td>
<td><strong>15.1</strong></td>
<td><strong>13.5</strong></td>
<td><strong>8.7</strong></td>
<td><strong>7.7</strong></td>
<td><strong>9.5</strong></td>
<td><strong>7.0</strong></td>
<td><strong>4.1</strong></td>
<td><strong>4.7</strong></td>
</tr>
<tr>
<td>NELL995</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GQE</td>
<td>21.7</td>
<td>N/A</td>
<td>47.2</td>
<td>12.7</td>
<td>9.3</td>
<td>30.6</td>
<td>37.0</td>
<td>20.6</td>
<td>16.1</td>
<td>12.6</td>
<td>9.6</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Query2Box</td>
<td>21.6</td>
<td>N/A</td>
<td>47.6</td>
<td>12.5</td>
<td>8.7</td>
<td>30.7</td>
<td>36.5</td>
<td>20.5</td>
<td>16.0</td>
<td>12.7</td>
<td>9.6</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>BetaE</td>
<td>19.0</td>
<td>0.4</td>
<td>53.1</td>
<td>6.0</td>
<td>3.9</td>
<td>32.0</td>
<td>37.7</td>
<td>15.8</td>
<td>8.5</td>
<td>10.1</td>
<td>3.5</td>
<td>0.1</td>
<td>1.4</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td>FuzzQIE</td>
<td><strong>27.1</strong></td>
<td><strong>7.3</strong></td>
<td><strong>57.6</strong></td>
<td><strong>17.2</strong></td>
<td><strong>13.3</strong></td>
<td><strong>38.2</strong></td>
<td><strong>41.5</strong></td>
<td><strong>27.0</strong></td>
<td><strong>19.4</strong></td>
<td><strong>16.9</strong></td>
<td><strong>12.7</strong></td>
<td><strong>9.1</strong></td>
<td><strong>8.3</strong></td>
<td><strong>8.9</strong></td>
<td><strong>4.4</strong></td>
<td><strong>5.6</strong></td>
</tr>
</tbody>
</table>
Thanks for Listening~