中國人民大學
RENMIN UNIVERSITY OF CHINA

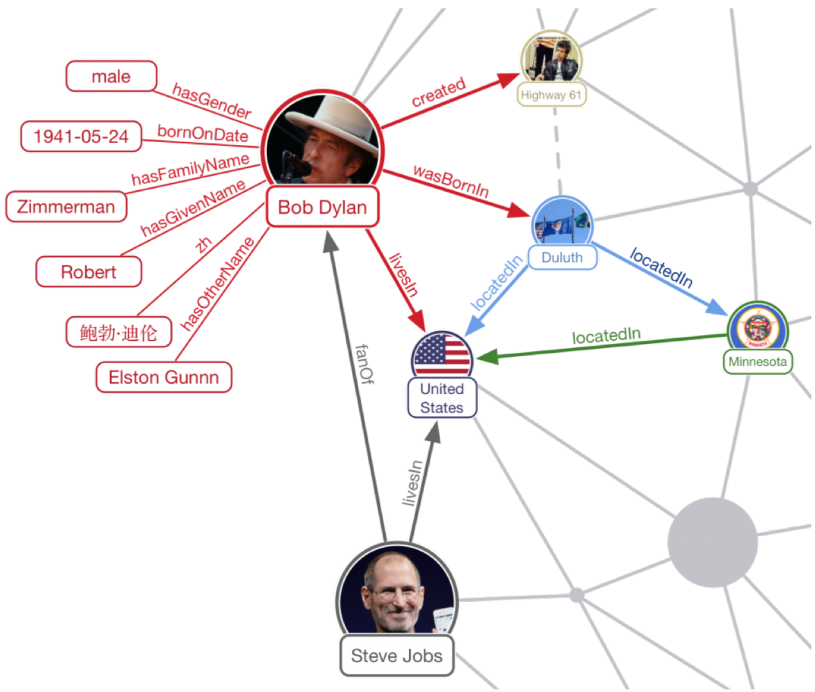
Neural, Symbolic and Neural-Symbolic Reasoning on Knowledge Graphs

Jing Zhang

School of Information, Renmin University of China

Knowledge Graphs

- A set of facts represented as triplets
  - (head entity, relation, tail entity)



yago select knowledge

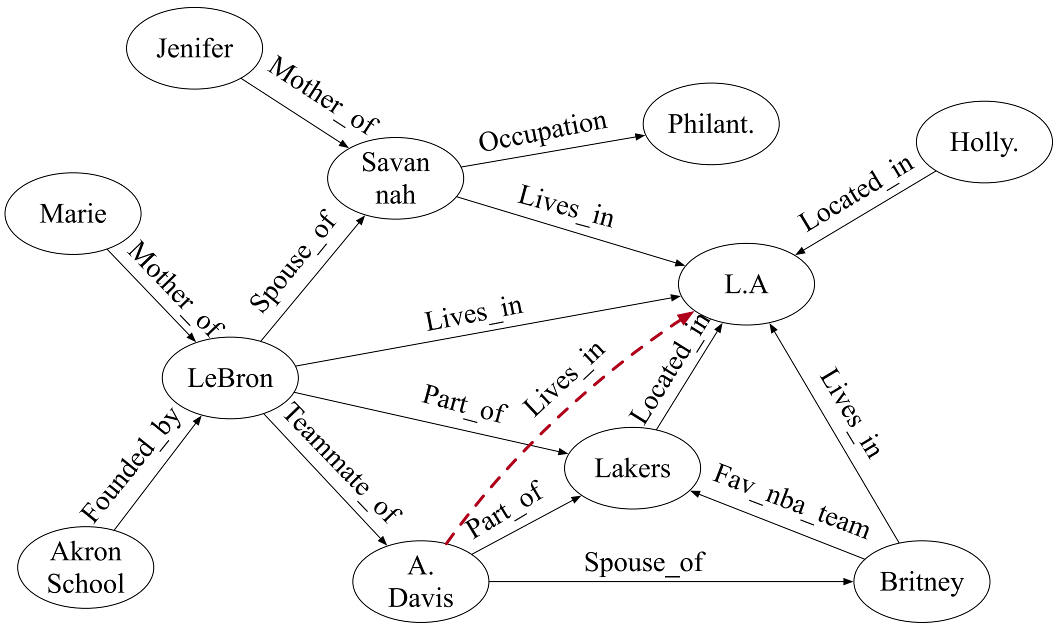
Freebase™

DBpedia

WIKI DATA CON

Knowledge Graph Reasoning

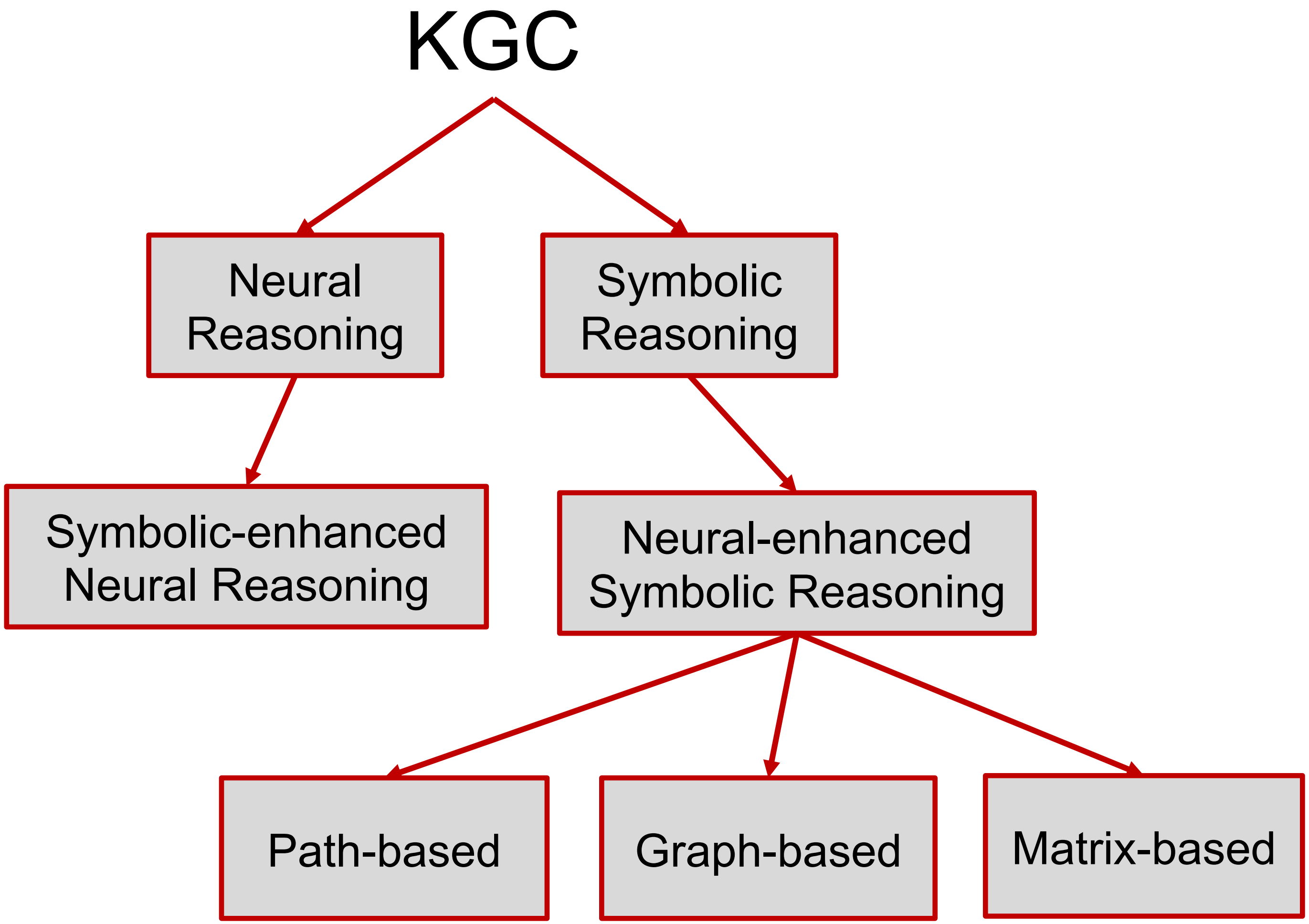
- Knowledge graph reasoning
  - Deduce tails entities over KGs as the answers to the given query
- A query can be
  - A head entity and a relation (KGC), A natural language question (KGQA)



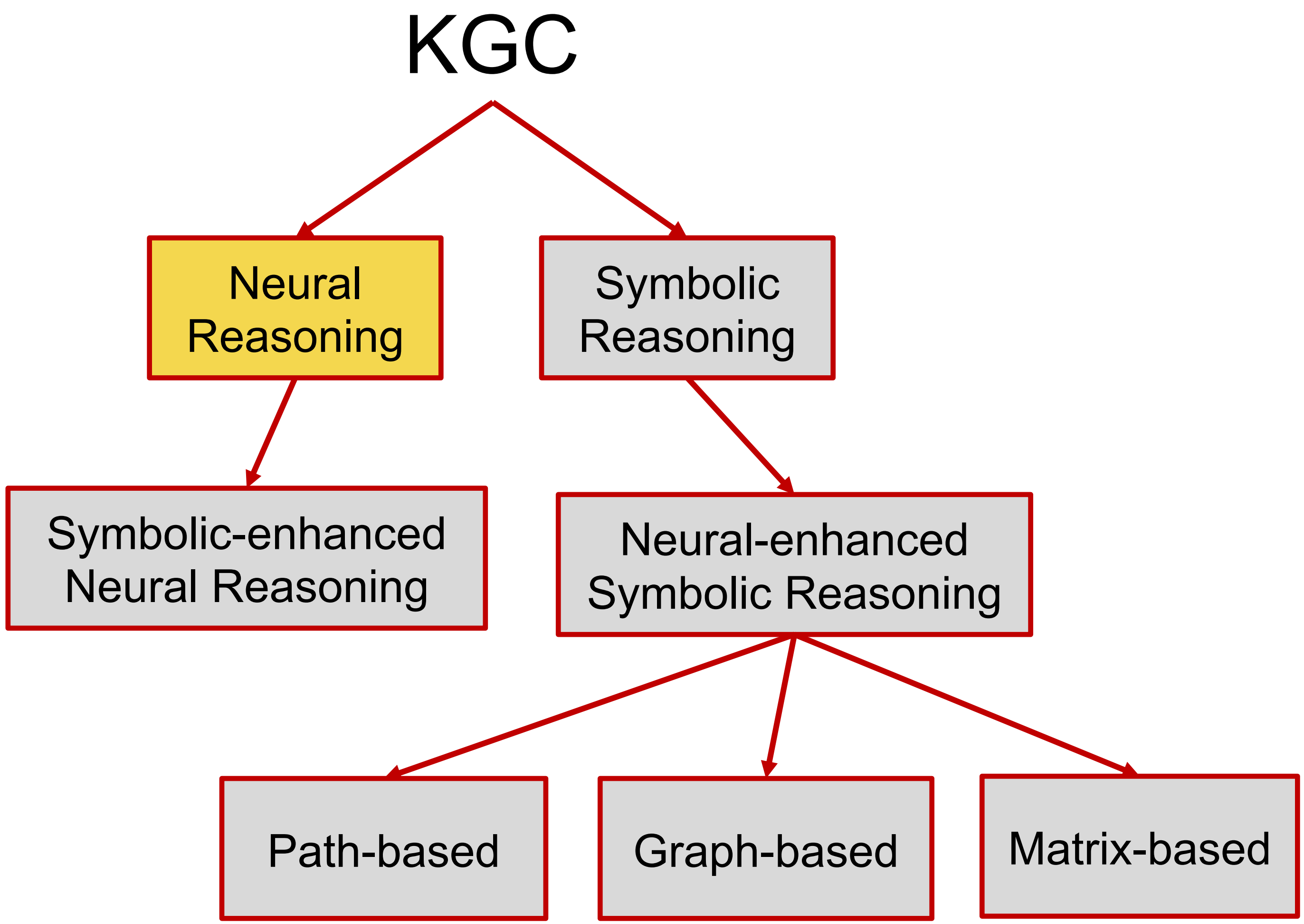
An example of knowledge graph completion:
Query relation: Lives_in, head entity: A. Davis,
Reasoning result: L.A

An example of knowledge graph question answering:
Question: Where do the spouses of the teammates of Lakers usually live?
Reasoning result: L.A

Knowledge Graph Completion



Knowledge Graph Completion



Neural Reasoning

- Learn distributed embeddings for entities/relations
  - Translation-based models

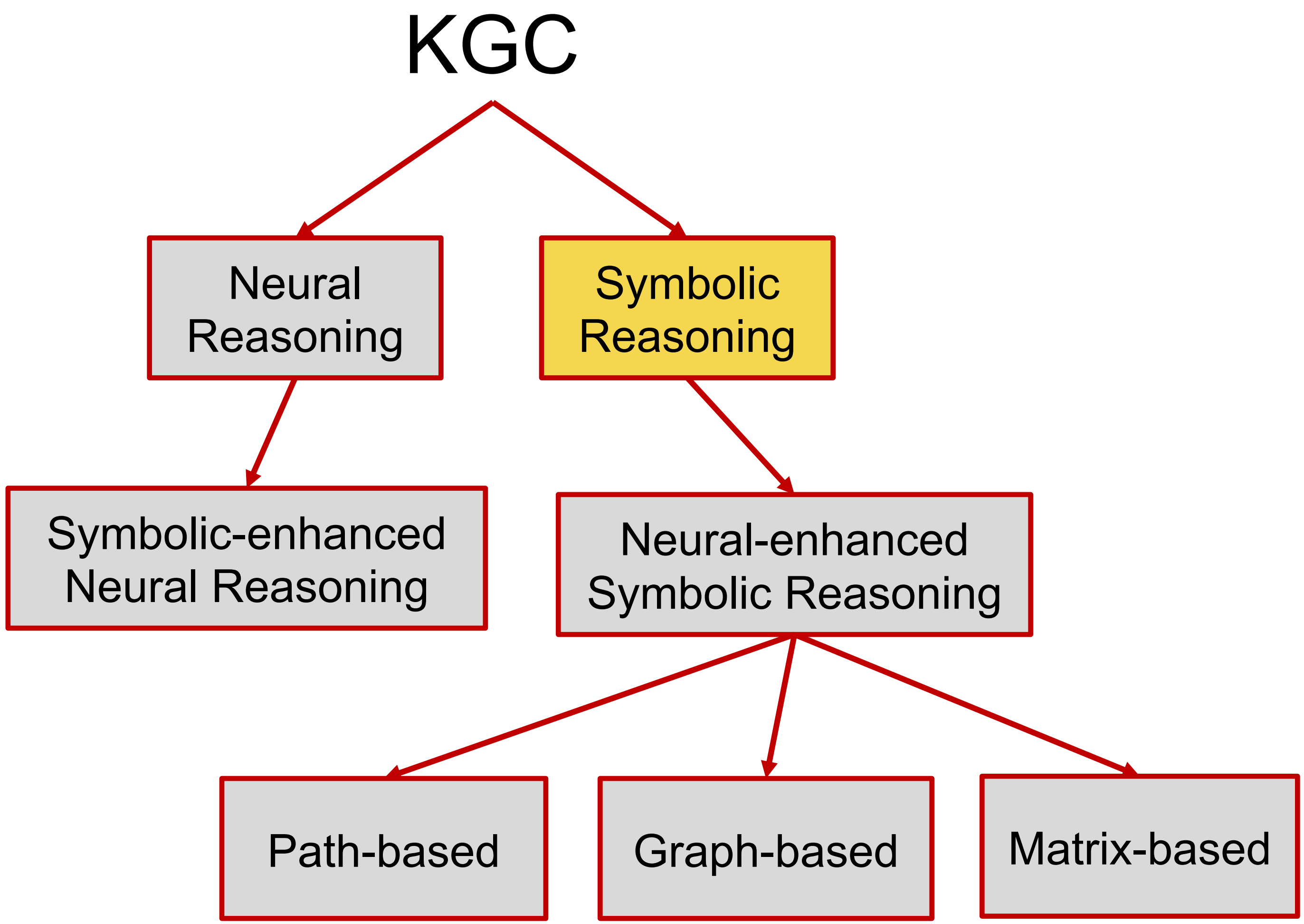
    $$s(h,r,t) = ||\mathbf{h}+\mathbf{r}-\mathbf{t}||_2^2$$

    - TransE, TransR, TransP….
  - Multiplicative models

    $$s(h,r,t) = \mathbf{h}^T\mathbf{M}_r\mathbf{t}$$

    - RESCAL, DisMult, ComplEx
  - Deep models
    - CNN:ConvE(h,r), ConvR(r-cnn), ConvKG(h,r,t)
    - RNN: RSN
    - GNN: R-GCN(r->W_r), CompGCN(r and W)
- Good generalization, but ineffective for complex logic relations, lack interpretation

Knowledge Graph Completion



Symbolic Reasoning

- Inductive logic programming (ILP)
  - Derive a set of if-then logic rules to describe the positive instances but not the negative instances

Rule: $\gamma : A(\alpha_1, \cdots, \alpha_m) \rightarrow \alpha$

Atom: $\alpha \equiv P_i(x_1, x_2, \cdots, x_n)$

Ground atom: all the variables are instantiated by constants

A triplet (h, r, t) can be viewed as a ground atom r(h,t)

Example:

Predicate set: $\mathcal{P} = \{\text{zero}, \text{succ}\}$

Ground atoms: $\mathcal{G} = \{\text{zero}(0), \text{succ}(0,1), \text{succ}(1,2), \cdots\}$

Positive/negative instances:

$$\begin{aligned} \mathcal{S} &= \{\text{even}(0), \text{even}(2), \text{even}(4), \cdots\} \\ \mathcal{N} &= \{\text{even}(1), \text{even}(3), \text{even}(5), \cdots\} \end{aligned}$$

Solution of rules for the even predicate:

$$\boxed{\begin{aligned} \text{even}(X) &\leftarrow \text{zero}(X), \\ \text{even}(X) &\leftarrow \text{even}(Y) \wedge \text{succ2}(Y, X), \\ \text{succ2}(X, Y) &\leftarrow \text{succ}(X, Z) \wedge \text{succ}(Z, Y) \end{aligned}}$$

AMIE (Galárraga et al., 2013)

- Rule Extending
  - Generate candidate rules by adding three kinds of new atoms into existing rules iteratively

Rule:$r_h(x, y) \leftarrow r_1(x, z_1) \wedge ... \wedge r_n(z_{n-1}, y)$

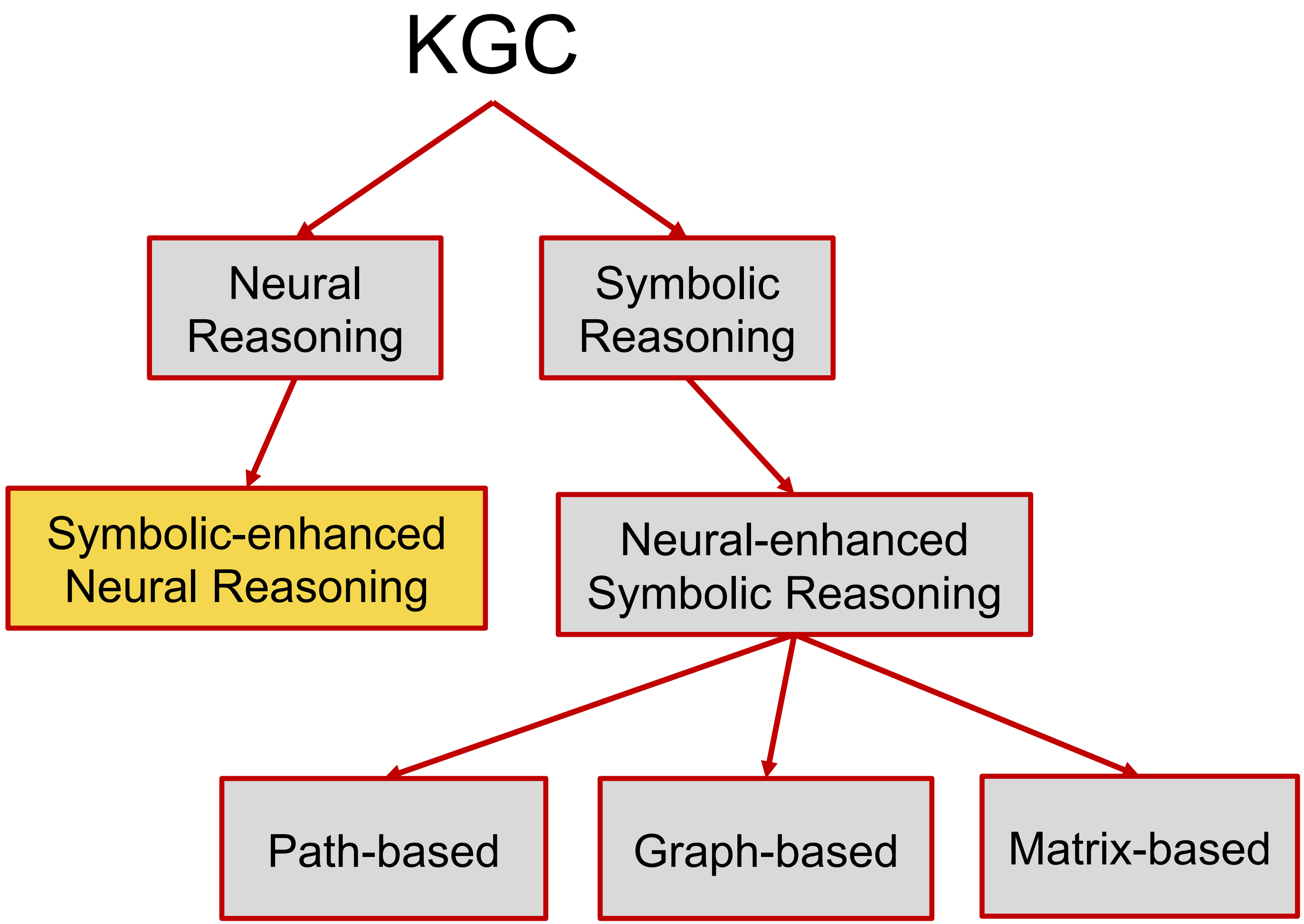
Dangling atom:$r^D(x, k), r^D(k, y), ...$

Instantiated atom:$r^I(x, K), r^I(K, y), ...$

Closing atom:$r^C(x, z), r^C(z, y), ...$

- Rule Pruning
  - Recall:
    - If a rule r<- B can cover more triplets with r, the head coverage of the rule will be high
  - Precision:
    - If more triplets derived by a rule r<-B satisfy r, the confidence of the rule will be high
- Good interpretation, but intolerant to the ambiguous and noisy data.

Knowledge Graph Completion



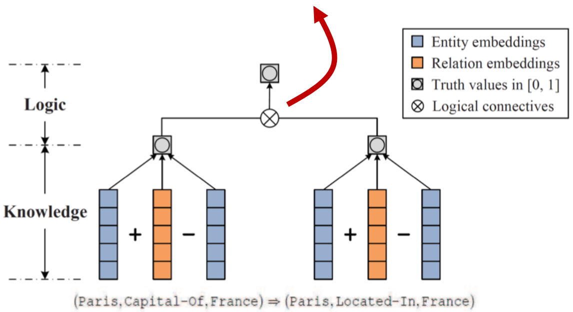
Symbolic-enhanced Neural Reasoning

- Extend the training set for embeddings
- KALE (Guo et al, 2016)
  - Deal with two types of rules
  - Score a ground rule

Inference and transitivity rules:

$$\forall x, y : (x, r_s, y) \Rightarrow (x, r_t, y)$$
$$\forall x, y, z : (x, r_{s_1}, y) \wedge (y, r_{s_2}, z) \Rightarrow (x, r_t, z)$$

$$s(f_1 \Rightarrow f_2) = s(f_1)s(f_2) - s(f_1) + 1$$



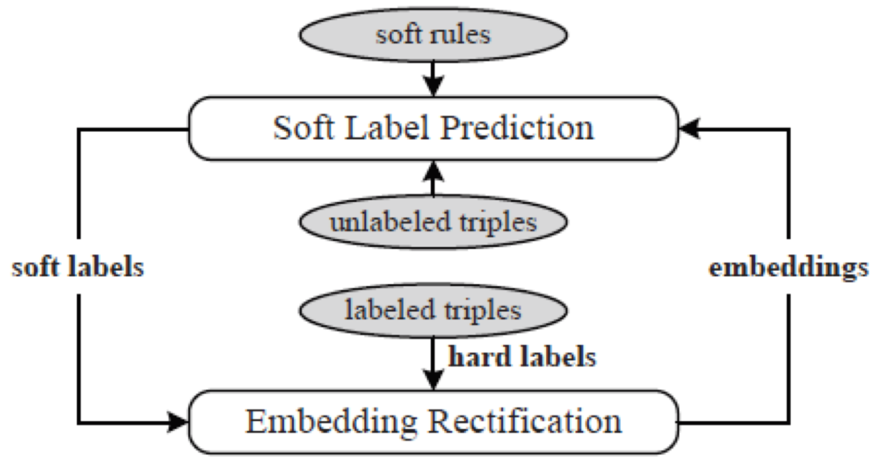
$$\begin{aligned} s(f_1 \wedge f_2) &= s(f_1) \cdot s(f_2), \\ s(f_1 \vee f_2) &= s(f_1) + s(f_2) - s(f_1) \cdot s(f_2). \\ s(\neg f_1) &= 1 - s(f_1). \end{aligned}$$

  - Combine the triplets and the ground rules as the training set

$$\min_{\{\mathbf{e}\},\{\mathbf{r}\}} \sum_{f^+ \in \mathcal{F}} \sum_{f^- \in \mathcal{N}_{f^+}} \left[\gamma - I(f^+) + I(f^-)\right]_+$$

Symbolic-enhanced Neural Reasoning

- RUGE (Guo et al., 2018)
  - Inject the new triplets derived by some rules instead of the ground rules into the training set
  - Iteratively update entity/relation embeddings and label the new triplets derived by the rules



Symbolic-enhanced Neural Reasoning

- Wang et al., 2019
  - Avoid calculating the scores of triplets independently
  - First transform a ground rule into first-order logic, and then perform matrix operations

TABLE 1
The format of first-order logic [127]. For example, the third line defines the transitivity rule $(r_1 + r_2) \Rightarrow r_3$, following which we can infer a new triple (e_1, r_3, e_3) from two existing triplets (e_1, r_1, e_2) and (e_2, r_2, e_3).

Triple and ground rule	**The format of first-order logic**
(h, r, t)	$r(h) \Rightarrow t$
$(h, r_1, t) \Rightarrow (h, r_2, t)$	$[(h \in C) \wedge [r_1(h) \Rightarrow t]] \Rightarrow [r_2(h) \Rightarrow t]$
$(e_1, r_1, e_2) + (e_2, r_2, e_3) \Rightarrow (e_1, r_3, e_3)$	$[[r_1(e_1) \Rightarrow e_2] \wedge [r_2(e_2) \Rightarrow e_3]] \Rightarrow [r_3(e_1) \Rightarrow e_3]$
$(h, r_1, t) \Leftrightarrow (t, r_2, h)$	$[[r_1(h) \Rightarrow t] \Rightarrow [r_2(t) \Rightarrow h]] \wedge [[r_2(t) \Rightarrow h] \Rightarrow [r_1(h) \Rightarrow t]]$

- Lack interpretation

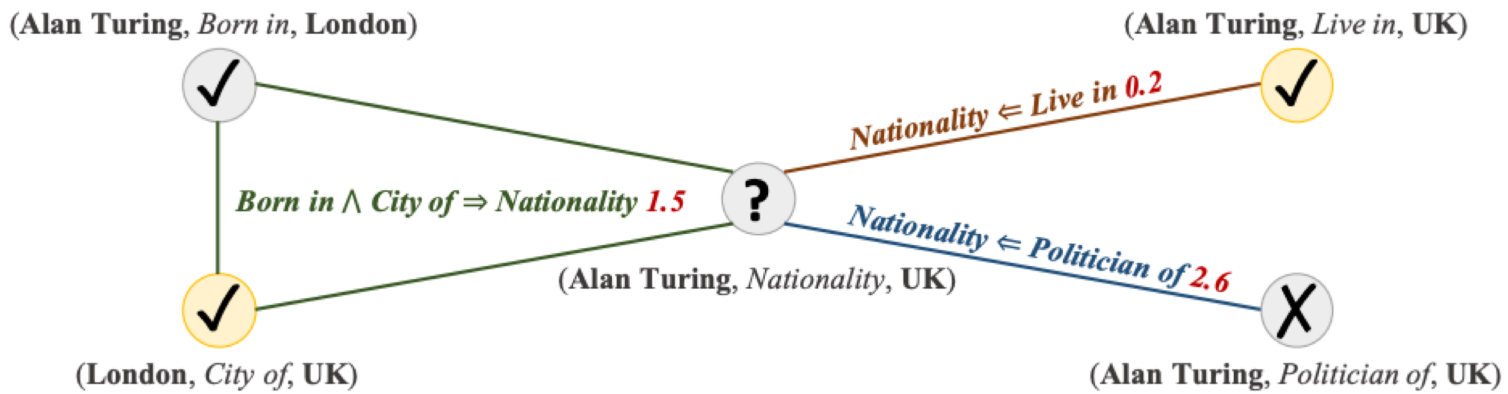
TABLE 2
Mathematical expression of first-order logic [127].

First-order logic	**Mathematical expression**
$r(h)$	$\mathbf{r} + \mathbf{h}$
$a \Rightarrow b$	$\mathbf{a} - \mathbf{b}$
$h \in C$	$\mathbf{h} \cdot \mathbf{C}$ ($\mathbf{C}$ is a matrix)
$a \wedge b$	$\mathbf{a} \otimes \mathbf{b}$
$a \Leftrightarrow b$	$(\mathbf{a} - \mathbf{b}) \otimes (\mathbf{a} - \mathbf{b})$

Symbolic-enhanced Neural Reasoning

Multiple rule inference together. pLogicNet (Qu et al, 2019)

Markov logic network



A node is built for each grounding atom

An edge is built between two nodes if they are in the same rule

All the nodes in a ground rule form a clique

$$p(\mathbf{v}_O, \mathbf{v}_H) = \frac{1}{Z} \exp\left(\sum_{l \in L} w_l \sum_{g \in G_l} \mathbb{1}\{g \text{ is true}\}\right) = \frac{1}{Z} \exp\left(\sum_{l \in L} w_l \boxed{n_l(\mathbf{v}_O, \mathbf{v}_H)}\right)$$

#true groundings of rule l

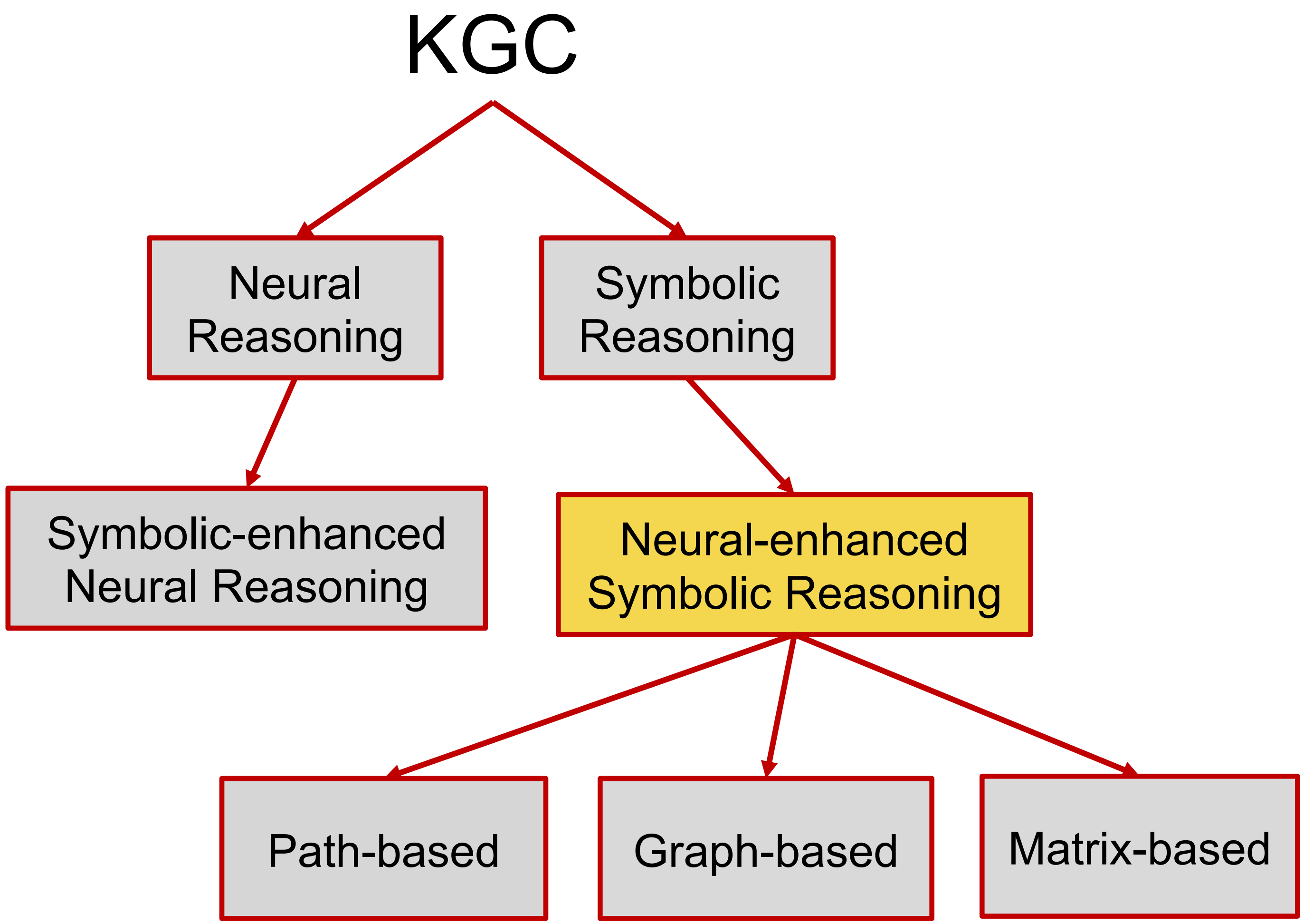
Learn the corresponding weights

Infer the label of a ground atom

Symbolic-enhanced Probabilistic Reasoning

- pLogicNet (Qu et al, 2019)
  - Combine MLN and graph embeddings
    - Use logic rules to predict the label of the ground atom, treat it as extra training data from KGE model.
    - Annotate all the hidden labels with the KGE model, and then update the weights of rules.
- The logic network is large, making the inference inefficient; can not learn new rules.

Knowledge Graph Completion



Neural-driven Symbolic Reasoning

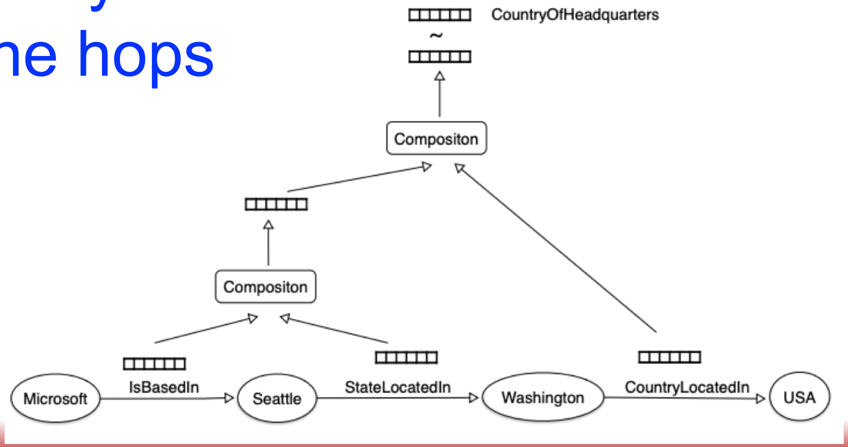
- To derive the logic rules
  - Extend multi-hop neighbors around the head entity, and then predict the answers in these neighbors
  - NN is to deal with the uncertainty and ambiguity, and also reduce the search space.

Path-based Reasoning

- Extend only one neighbor at each step
- PRA (Lao et al, 2011)
  - Given h and t, enumerate all the paths
  - Calculate $S_p(h,t)$ of different paths as features to train a classifier for each relation
  - Poor generalization, cannot deal with unobserved relations

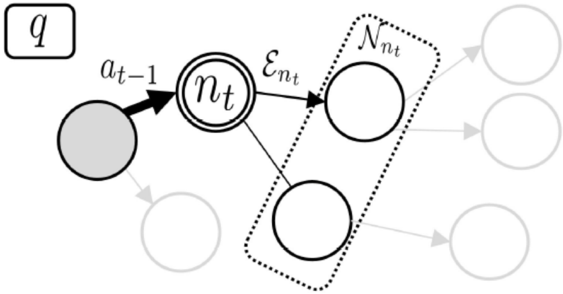
Path-based Reasoning (Cont.)

- Neelakantan et al., 2015
  - Use RNN to compose the semantics of relations in an arbitrary-length path
  - Compare the embeddings between a path and the query relation
  - Improve the generalization, can deal with unobserved relations
  - Paths are traversed heuristically without evaluation. Path space increases with the hops



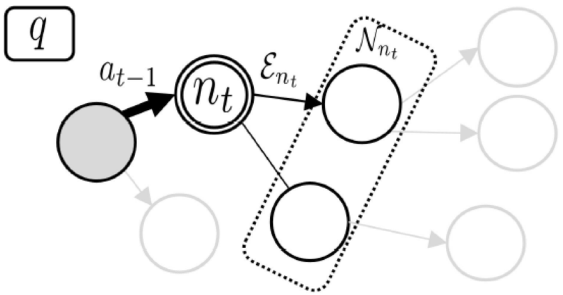
Path-based Reasoning (Cont.)

- DeepPath (Xiong et al., 2017)
  - Reinforcement learning
  - To evaluate a path
  - MDP
    - Agent: sample a relation at each hop
    - State: current entity, target entity
    - Reward: accuracy, length and diversity
  - Rules can be abstracted from the sampled paths (AnyBURL)
  - Tail entity should be given



Path-based Reasoning (Cont.)

- MINERVA(Das et al.,, 2018)
  - Reinforcement learning
  - To find the answer
  - MDP
    - State: query relation, historical path
    - Reward: accuracy
  - Soft reward, dropout actions (Multi-Hop)
  - Value-based RL (M-walk)
  - Model path as hidden variables (DIVA, RNNLogic)



Graph-based Reasoning

- Extend multiple neighbors at each step
  - FeedForward GNN
    - CogGraph (Du et al, 2020)
      - Limit neighbors at each step by a policy function
  - Source-specific GNN
    - NBFNet (Zhu et al., 2021)
      - Initialize the target relation and then perform GNN
  - Subgraph-specific GNN
    - GralL(Teru et al., 2020)
      - Given h and t, extract a subgraph (k-hop neighbors), use R-GCN to represent the subgraph

Matrix-based Reasoning

- Avoid selecting neighbors, but calculate a score to each neighbor.
- Express the logic relationships between the head and the tail entities by matrix operations.

Matrix-based Reasoning

- TensorLog (Cohen et al., 2016)
  - Given a head entity x, the score of each retrieved answer is:

$$\mathbf{s} = \underbrace{\sum_{\gamma} (\alpha_{\gamma} (\prod_{k \in \beta_{\gamma}} \mathbf{M}_{R_k} \mathbf{v}_x))}_{\text{The score of the query relation following different rules}}.$$

$$\max_{\{\alpha_\gamma, \beta_\gamma\}} \sum_{x,y} \mathrm{score}(y|x) =$$

$$\max_{\{\alpha_\gamma, \beta_\gamma\}} \sum_{x,y} \mathbf{v}_y^T \left(\sum_{\gamma} \alpha_\gamma (\prod_{k \in \beta_\gamma} \mathbf{M}_{R_k} \mathbf{v}_x) \right)$$

  - learning parameters is difficult as each rule is associated with a parameter. Enumerating rules is an discrete task

Matrix-based Reasoning (Cont.)

- Neural LP (Yang et al, 2017)
  - Interchanges the summation and the product
  - Change the weight of each rule into the weights of the predicates in the rule

$$\prod_{t=1}^{T} \sum_{k}^{|R|} a_t^k \mathbf{M}_{R_k}$$

Model the length dynamically

$$\begin{aligned} \mathbf{u}_0 &= \mathbf{v}_x \\ \mathbf{u}_t &= \underbrace{\sum_{k}^{|R|} a_t^k \mathbf{M}_{R_k}}_{\text{Softly combine next-hop relation}} \underbrace{\left(\sum_{\tau=0}^{t-1} b_t^{\tau} \mathbf{u}_{\tau}\right)}_{\text{Historical path}} \quad \text{for } 1 \le t \le T \\ \mathbf{u}_{T+1} &= \underbrace{\sum_{\tau=0}^{T} b_{T+1}^{\tau} \mathbf{u}_{\tau}}_{\text{Weighted average of the paths with different lengths}} \end{aligned}$$

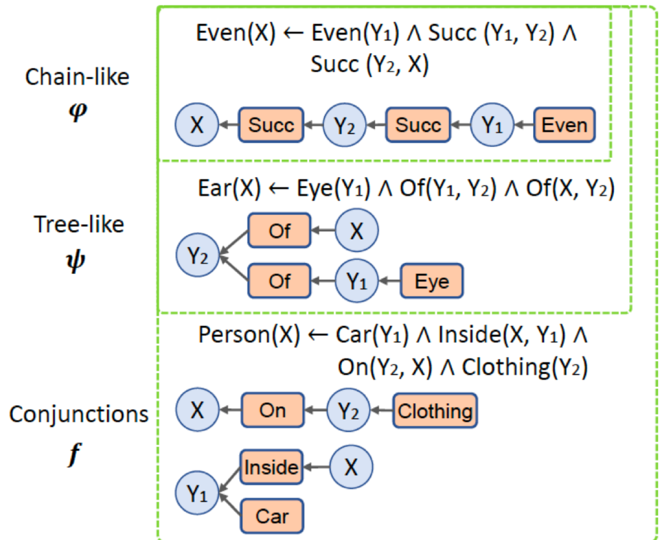
$$\begin{aligned} \mathbf{h_t} &= \text{update}\left(\mathbf{h_{t-1}}, \text{input}\right) \\ \mathbf{a_t} &= \text{softmax}\left(W\mathbf{h_t} + b\right) \\ \mathbf{b_t} &= \text{softmax}\left([\mathbf{h_0}, \ldots, \mathbf{h_{t-1}}]^T \mathbf{h_t}\right) \end{aligned}$$

Learn attentions by RNN

Fail to infer tree-like, conjunctions of rules

Matrix-based Reasoning (Cont.)

- Neural Logic Inductive Learning (Yang et al., 2020)



$$\psi_k(x,y) = \begin{cases} \sigma((\mathbf{M}_k\mathbf{1})^{\mathrm{T}}(\prod_{t'=1}^{T'}\mathbf{M}^{(t')}\mathbf{v}_y)) & \text{if } k \in \mathcal{U}, \\ \sigma((\prod_{t=1}^{T}\mathbf{M}^{(t)}\mathbf{v}_x)^{\mathrm{T}}\boxed{(\prod_{t'=1}^{T'}\mathbf{M}^{(t')}\mathbf{v}_y)}) & \text{if } k \in \mathcal{B}, \end{cases}$$

Replace $\mathbf{v}_y$ with another relation path. Thus it can represent the tree-like rules

$$\sum_{x,y} \boxed{\mathbf{v}_y^T} \left(\sum_{\gamma} \alpha_\gamma \left(\prod_{k \in \beta_\gamma} \mathbf{M}_{R_k} \mathbf{v}_x \right) \right)$$

Logic combination of primitive statements via $\{\wedge, \vee, \neg\}$

$$\mathcal{F}_0 = \Psi,$$
$$\hat{\mathcal{F}}_{l-1} = \mathcal{F}_{l-1} \cup \{1 - f(\mathrm{x}, \mathrm{x}') : f \in \mathcal{F}_{l-1}\},$$
$$\mathcal{F}_l = \{f_i(\mathrm{x}, \mathrm{x}') * f_i'(\mathrm{x}, \mathrm{x}') : f_i, f_i' \in \hat{\mathcal{F}}_{l-1}\}_{i=1}^{C}$$

Three stacked transformers are to learn attentions

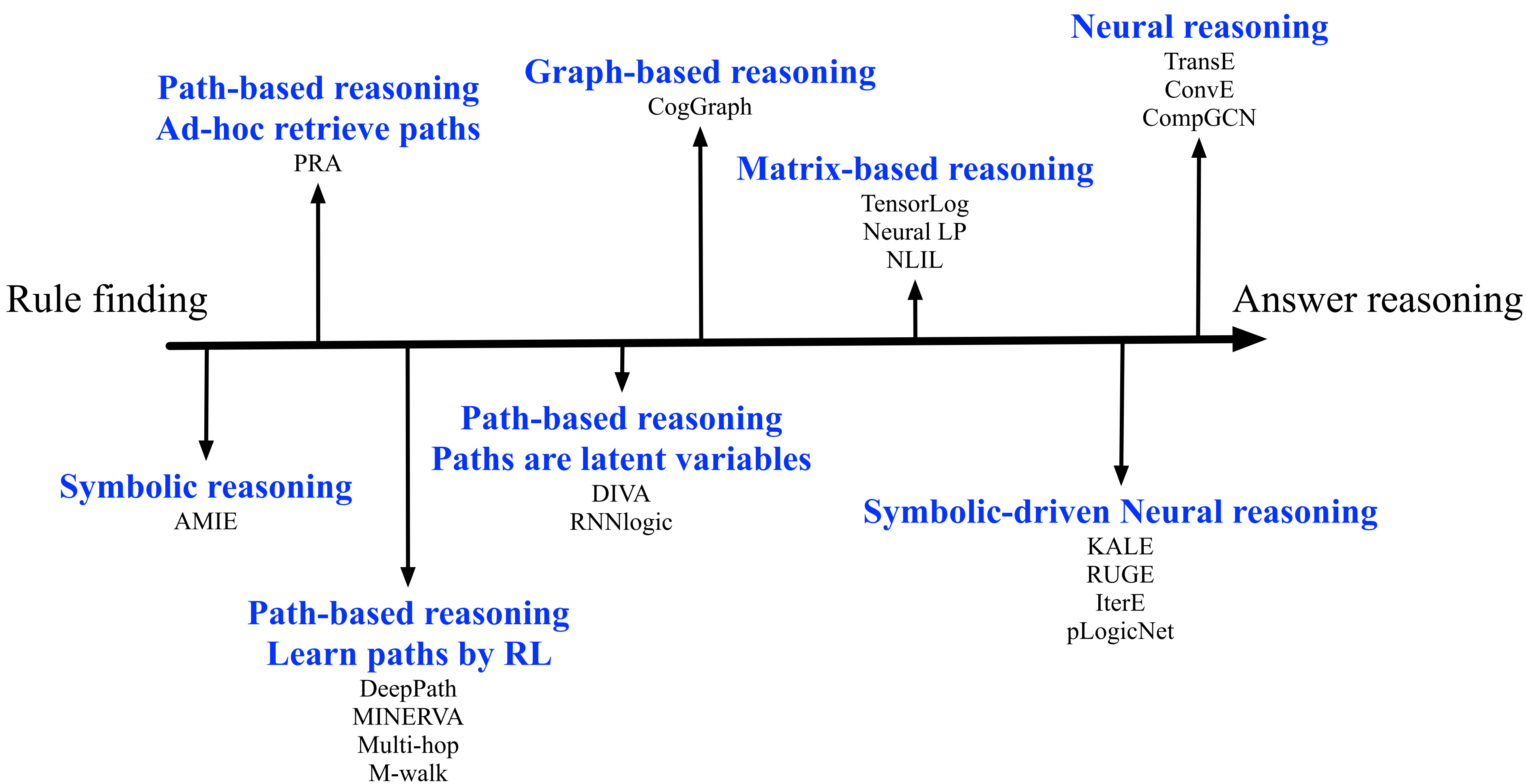
Matrix-based Reasoning (Cont.)

- Neural-Num-LP (Wang et al, 2020)
  - Extends Neural LP to learn the numerical rules
  - Support the comparison operator

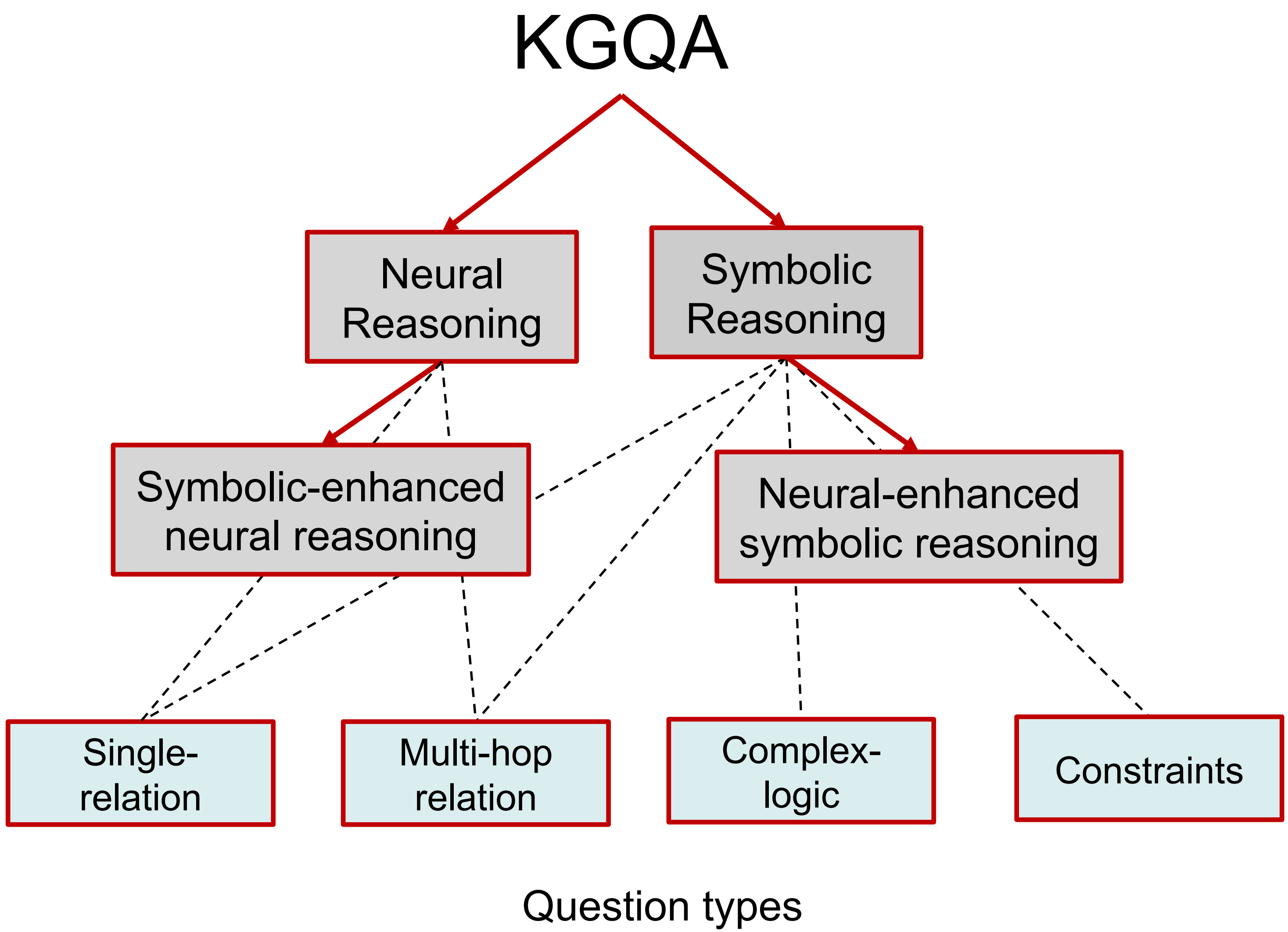
$$(M_{r_{pq}^{\leq}})_{ij} = \begin{cases} 1 & \text{if } p_i \leq q_j, \\ 0 & \text{otherwise,} \end{cases}$$

$$\sum_{x,y} \mathbf{v}_y^T \left(\sum_{\gamma} \alpha_\gamma (\prod_{k \in \beta_\gamma} \boxed{\mathbf{M}_{R_k}} \mathbf{v}_x) \right)$$

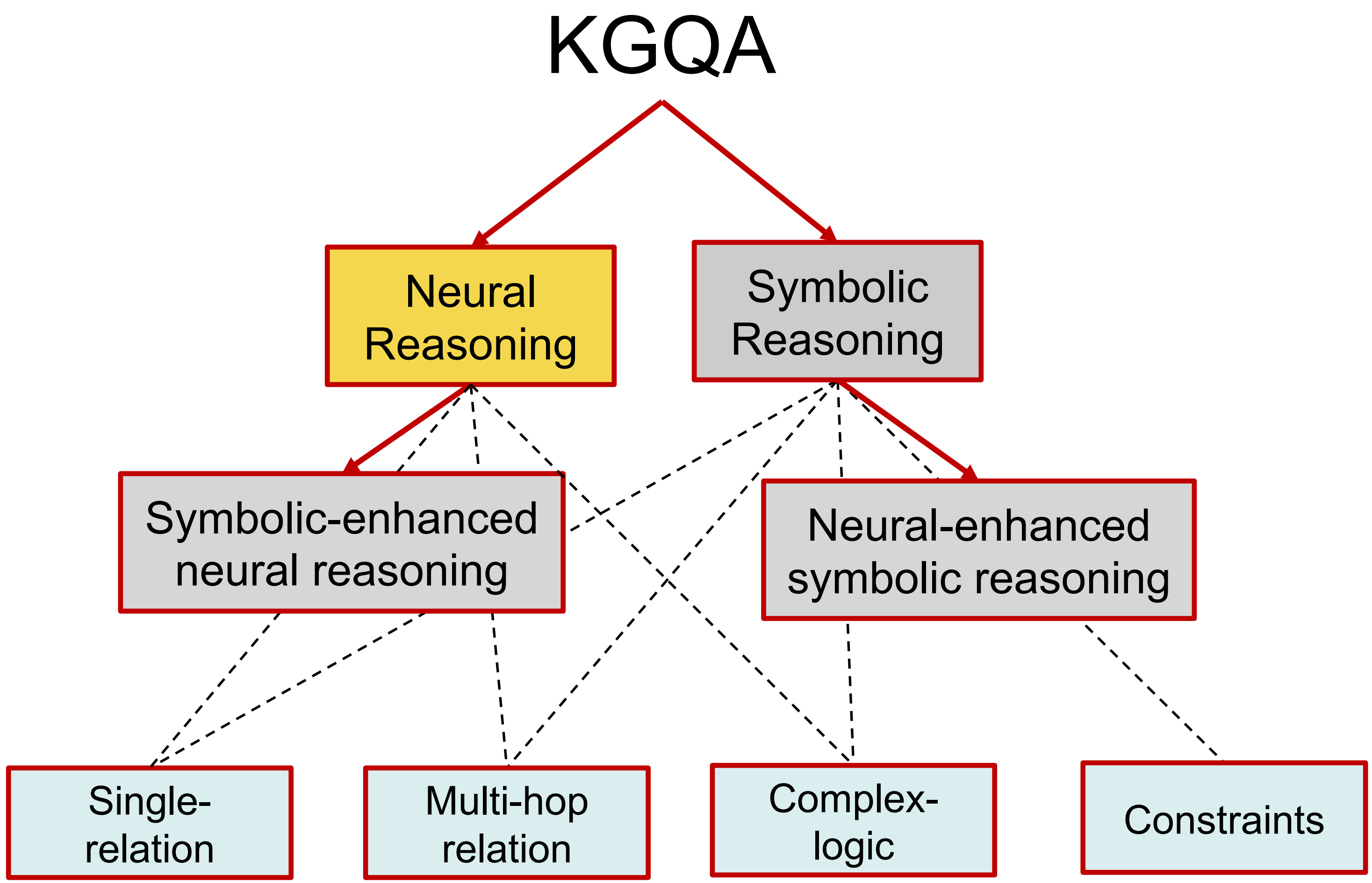
Summary of KGC



Knowledge Graph Question Answering



Knowledge Graph Question Answering

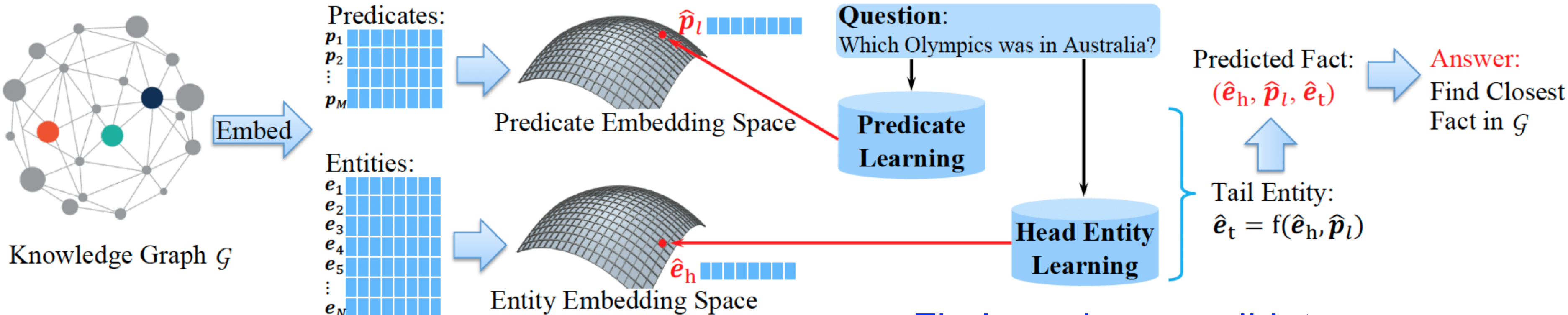


Question types

Neural Reasoning for Single-Relation

- KEQA（Huang et al. 2019）
  - Embed both the triplets and the question

Infer the predicate in the question

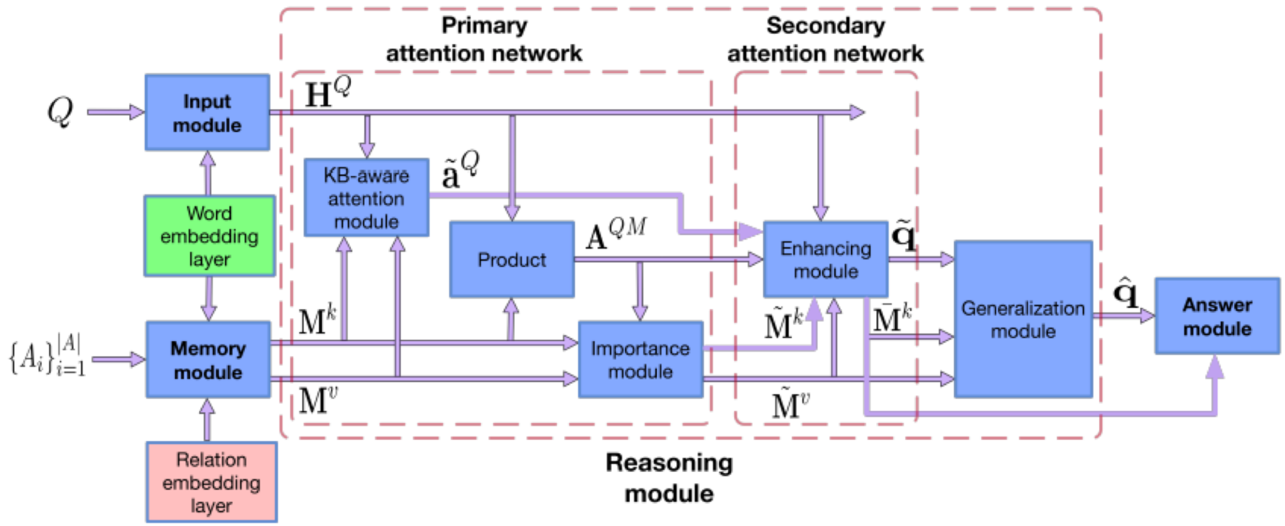


Find one-hop candidate answers from the head entity

Key idea: bridge the gap between the natural language expressions and the KB's predicates.

Neural Reasoning for Multi-hop Relation

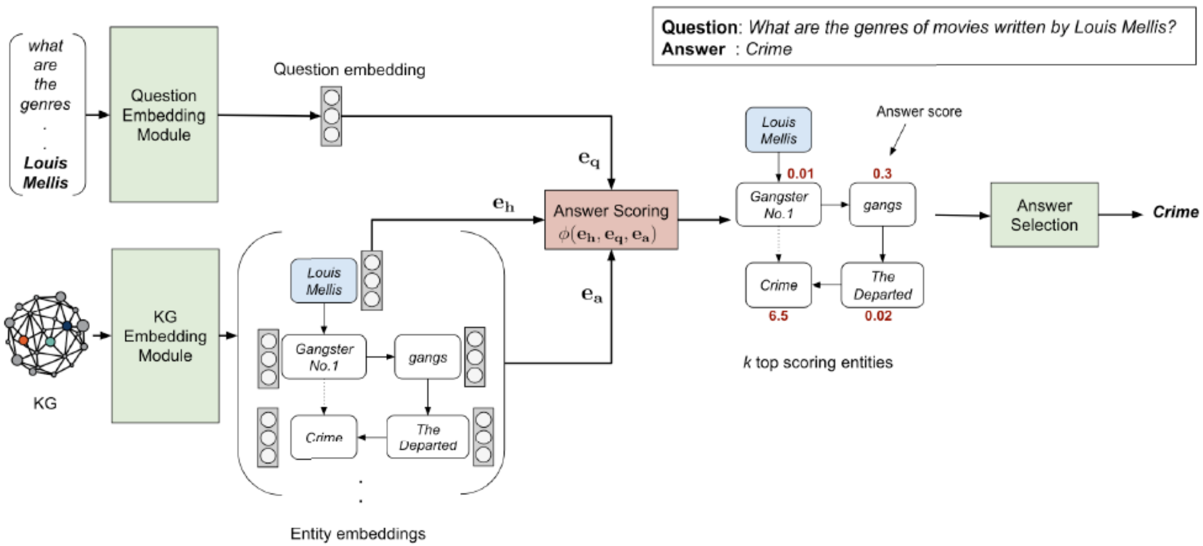
- BAMnet (Chen et al. 2019)
  - Capture interactions between question and KB
  - Bidirectional Attentive Memory Network



Candidate answers are the entities within h hops of topic entity.

Neural Reasoning for Multi-hop Relation

- EmbedKGQA (Saxena et al., 2020)
  - relaxes the requirement of answer selection from a pre-specified local neighborhood



Candidate answers are all the entities in KGs.

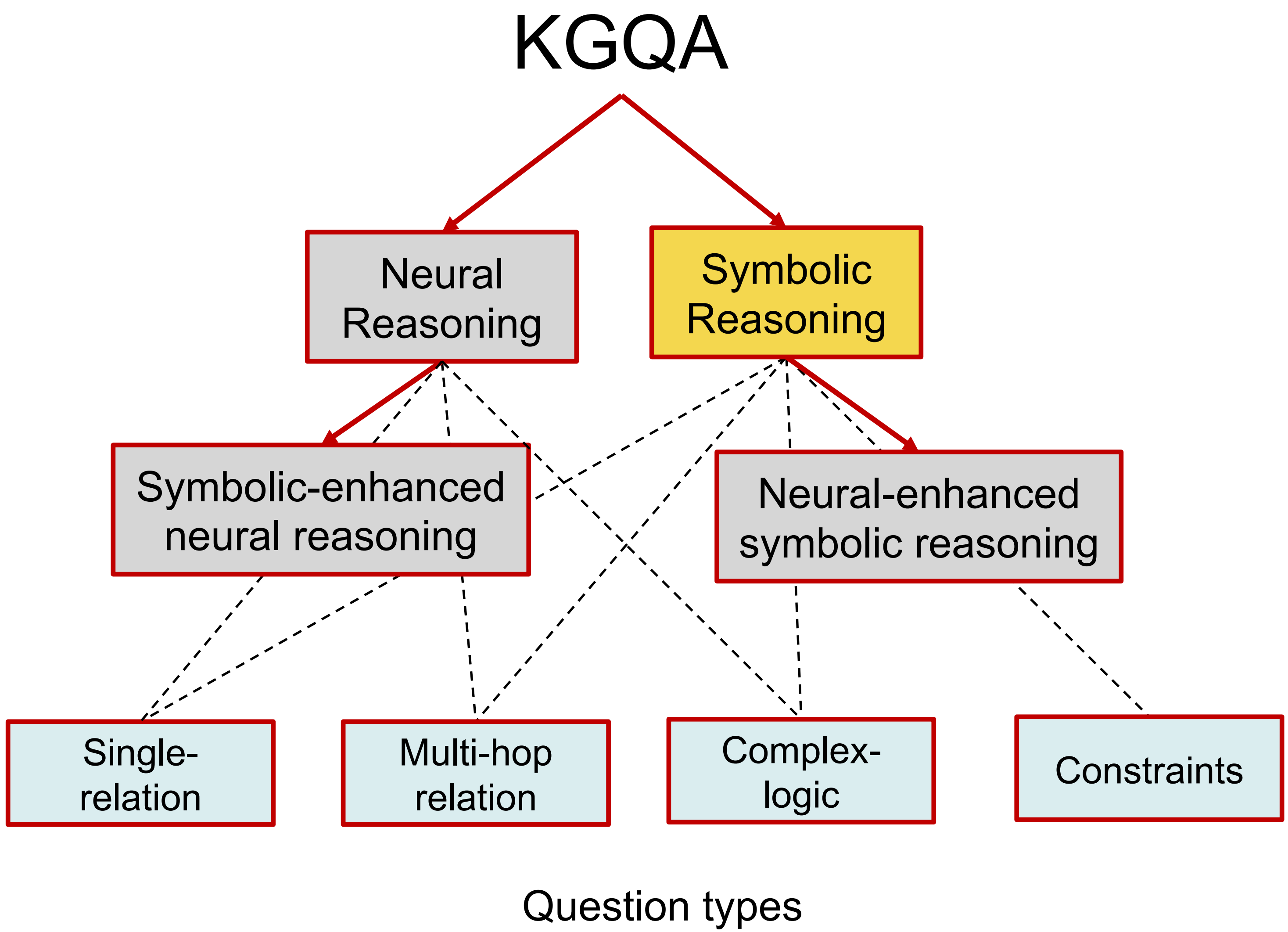
Neural Reasoning for Multi-hop Relation

- KV-MemNNs (Xu et al., 2021)
  - Use question to match key (head + relation)
  - Read value (tail)
  - Update query

Repeat

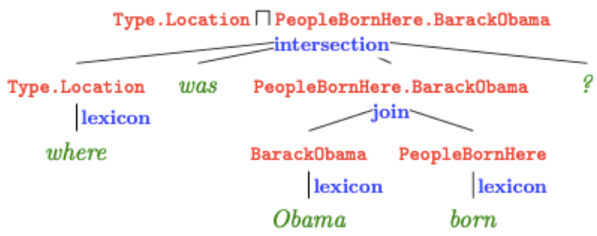
Repeated KV match-and-retrieval simulates the multi-hop reasoning process.

Knowledge Graph Question Answering



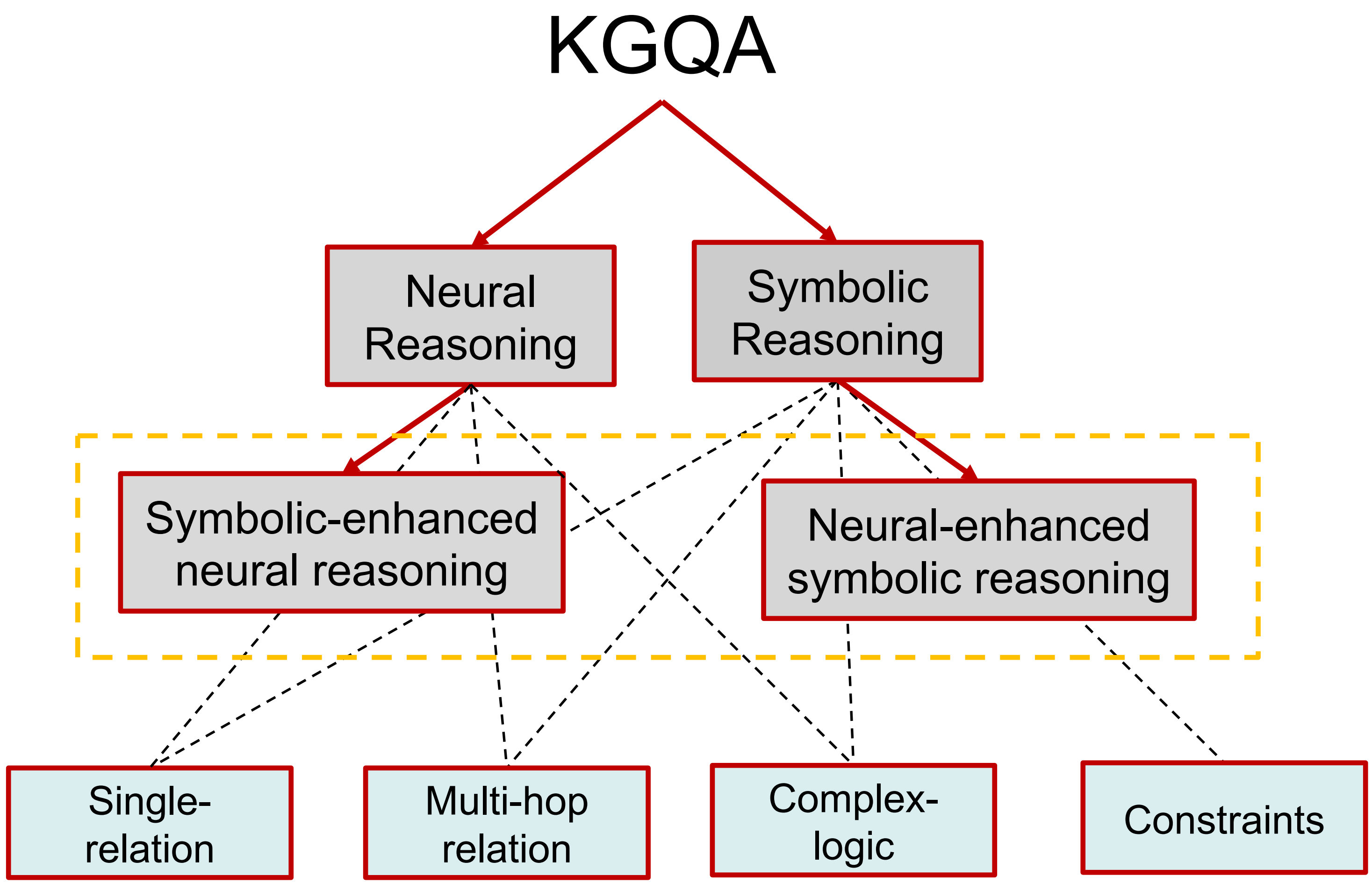
Semantic Parsing

- Parse questions into logic expression, and then execute the logic expression to get the answer
- Kwiatkowksi et al. 2010
  - Follow CCG to convert questions
  - E.g., x="New York borders Vermont", z="next_to(ny,vt)"
  - Learn a function f from the training data {(x,z)} to map x to z.
- Berant et al. 2013
  - Follow **λ**-DCS to convert questions



- Define hand-engineered templates or require ground truth query for supervision

Knowledge Graph Question Answering

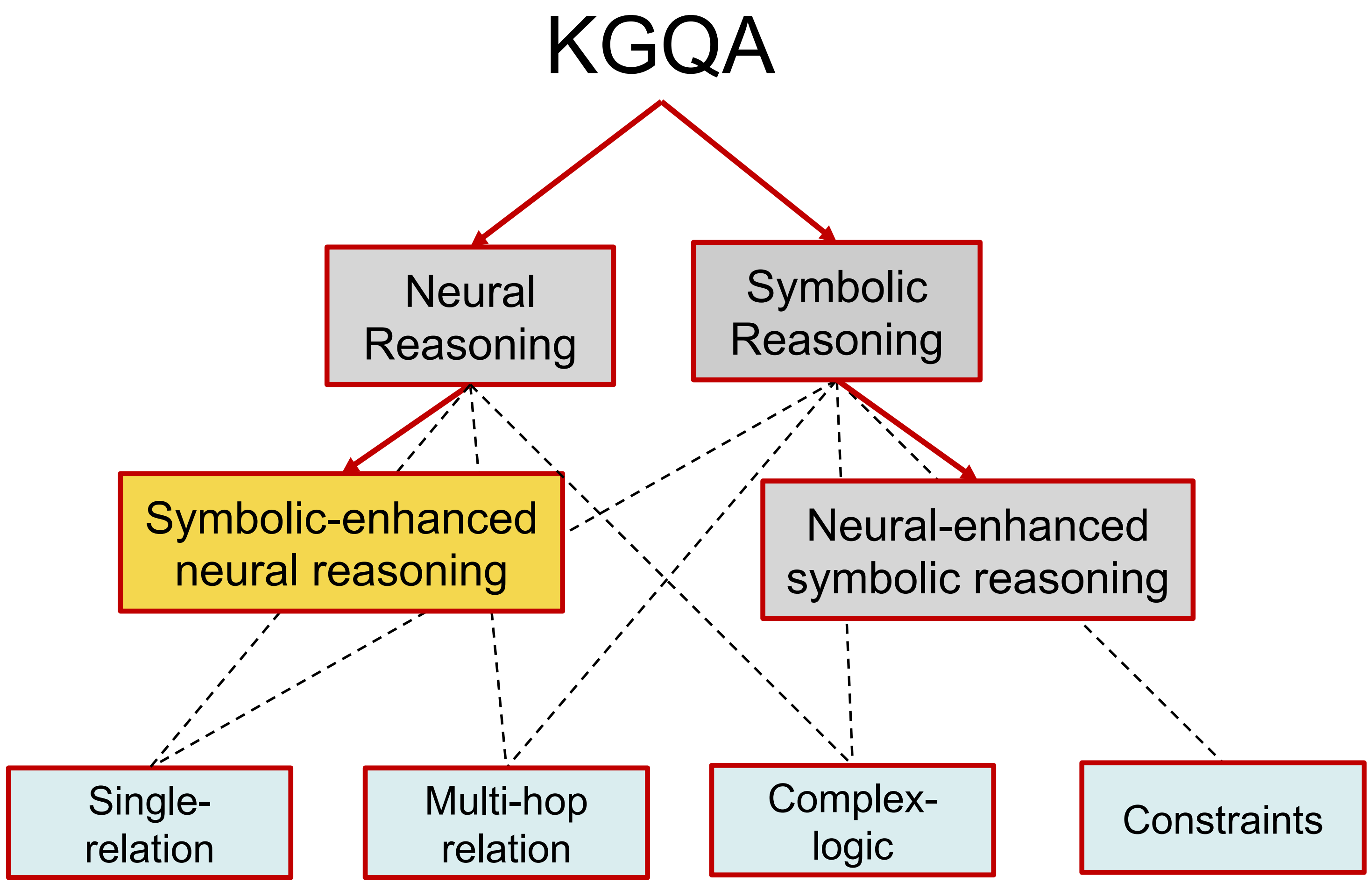


Question types

Neural Symbolic Reasoning

- Symbolic-enhanced neural reasoning
  - Use NN to define the complex logic operation
- Neural-enhanced symbolic reasoning
  - Parse and execute: target at parsing the questions, NN is to measure the similarity between the questions and the parsed graphs.
  - End-to-end: Parse and reason the answer simultaneously, NN is to measure the similarity and also embed the inferred paths or graphs, based on which the answer can be determined.

Knowledge Graph Question Answering



Symbolic-enhanced Neural Reasoning

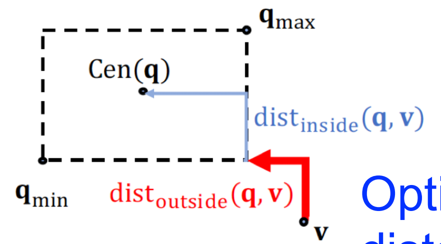
- Complex-logic question
  - Intersection
- GQE (Hamilton et al., 2018)
  - Start with the embeddings of the topic entities
  - Iteratively apply geometric operations to generate the query embedding.
  - Projection operator P
    - Forward $\mathcal{P}(\mathbf{q}, \tau) = \mathbf{R}_\tau \mathbf{q}$
  - Intersection operator
    - Intersection $\mathcal{I}(\{\mathbf{q}_1, ..., \mathbf{q}_n\}) = \mathbf{W}_\gamma \Psi\left(\mathrm{NN}_k(\mathbf{q}_i), \forall i = 1, ...n\}\right)$
  - Embed a question (a set) into a single point

Neural Reasoning for Complex-logic Question

- Complex-logic question
  - Intersection, Union
- Query2Box (Ren et al., 2020)
  - Embed a query as a box

$$\mathrm{Box}_{\mathbf{p}} \equiv \{\mathbf{v} \in \mathbb{R}^d : \mathrm{Cen}(\mathbf{p}) - \mathrm{Off}(\mathbf{p}) \preceq \mathbf{v} \preceq \mathrm{Cen}(\mathbf{p}) + \mathrm{Off}(\mathbf{p})\}$$

  - An entity embedding **v** is represented as (**v**,0)
  - A relation embedding **r** is represented as (cen(**r**), off(**p**))
  - Projection operator
  - Intersection operator $\mathbf{p} + \mathbf{r}$



Optimize the distance between v and the answer box.

$$\mathrm{Cen}(\mathbf{p}_{\mathrm{inter}}) = \sum_i \mathbf{a}_i \odot \mathrm{Cen}(\mathbf{p_i}), \quad \mathbf{a}_i = \frac{\exp(\mathrm{MLP}(\mathbf{p_i}))}{\sum_j \exp(\mathrm{MLP}(\mathbf{p_j}))},$$

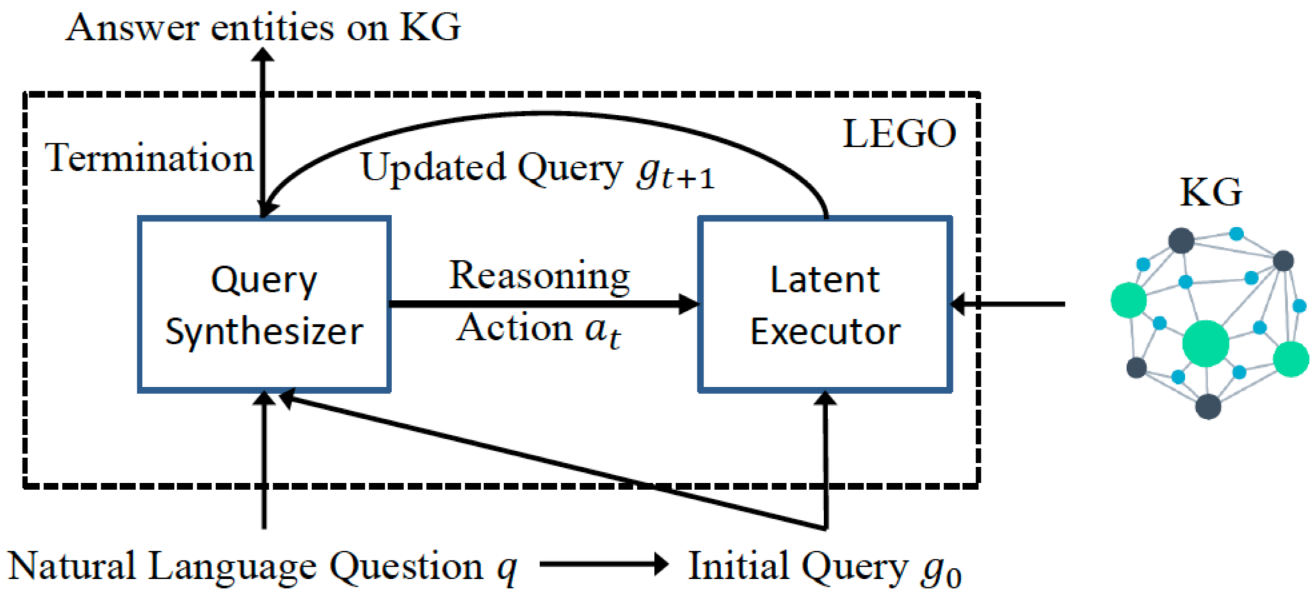
$$\mathrm{Off}(\mathbf{p}_{\mathrm{inter}}) = \mathrm{Min}(\{\mathrm{Off}(\mathbf{p_1}), \dots, \mathrm{Off}(\mathbf{p_n})\}) \odot \sigma(\mathrm{DeepSets}(\{\mathbf{p_1}, \dots, \mathbf{p_n}\}))$$

Symbolic-enhanced Neural Reasoning

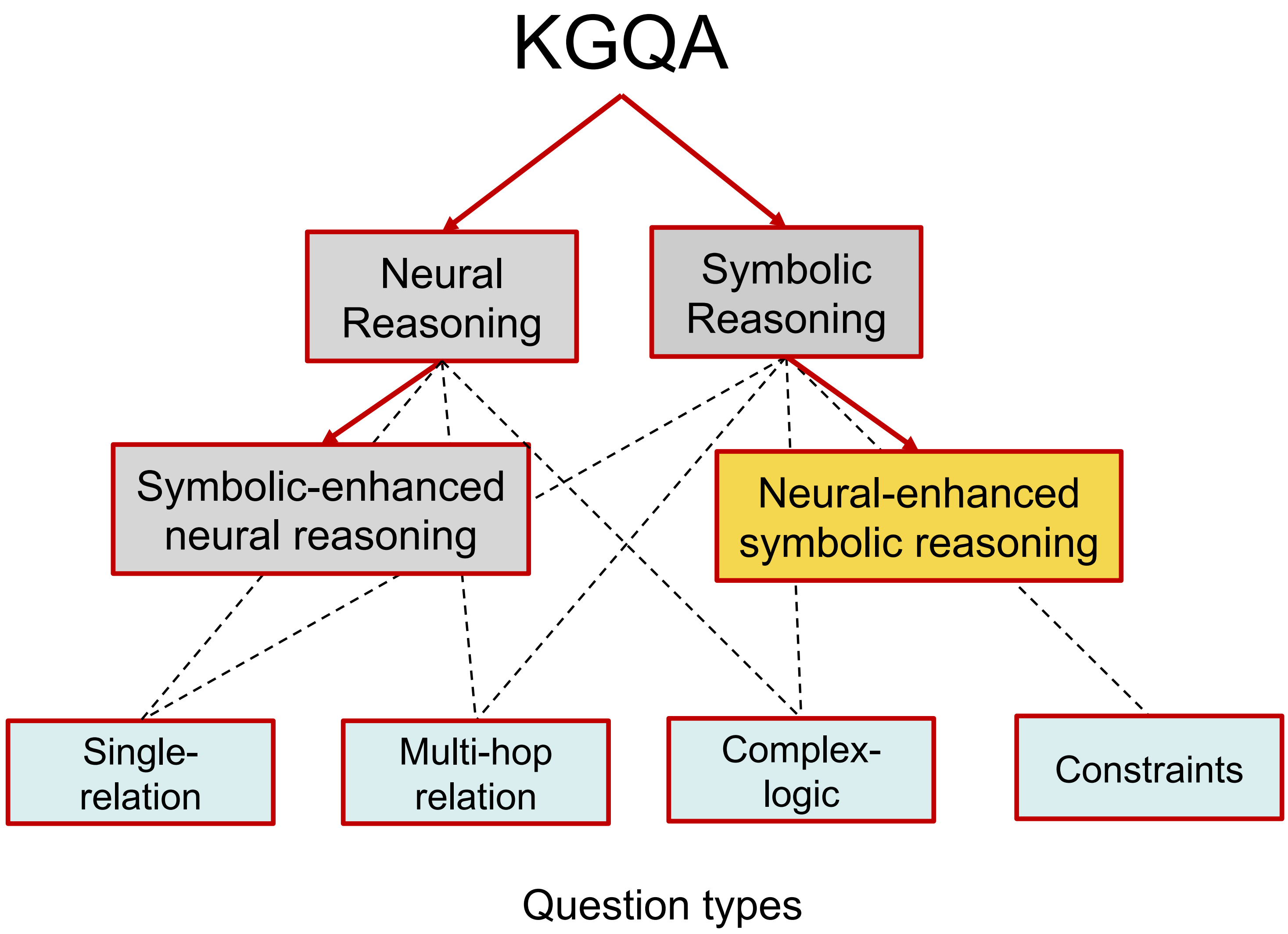
- Complex-logic question
  - Intersection，Union
- EMQL (Sun et al., 2020)
  - Faithful reasoning and generalization: represent entity set that support generalization and precise encoding.
  - MIPS: generalization
  - Count-min sketch: precise encoding
  - Support set intersection and union

Symbolic-enhanced Neural Reasoning

- Complex-logic question
  - Intersection，Union
- LEGO (Ren et al., 2021)
  - Parse query tree and embedding update simultaneously

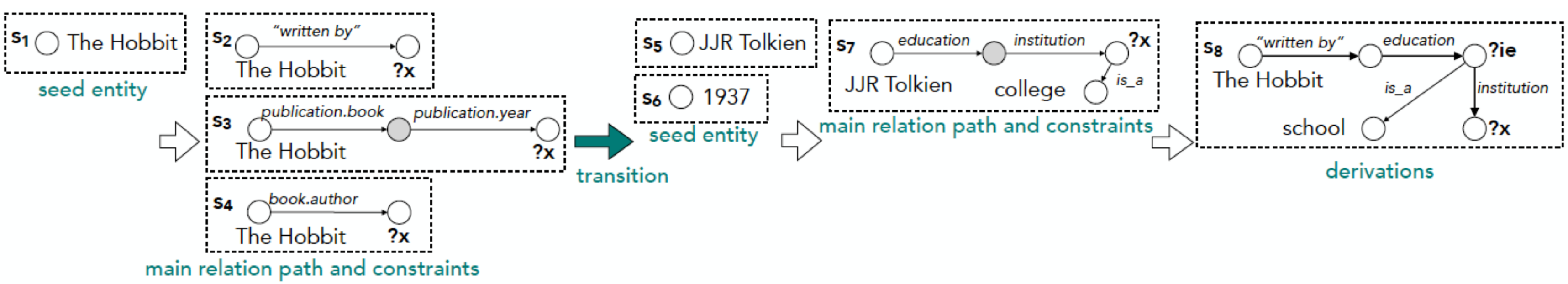


Knowledge Graph Question Answering



Parse and Execute

- Single-relation questions
  - Yih et al. 2014
    - Determine (mention, entity) , (nlp pattern, relation)
    - Add a CNN model to determine the mapping
- Multi-hop questions
  - MULTIQUE (Bhutani et al., 2020)
    - Add an LSTM to encode and measure the similarity between the question and each current sub-query graph



Parse and Execute

- Multi-constraint questions

Constraint Category	Example	Percentage
Multi-Entity	which films star by **Forest Whitaker** and are directed by **Mark Rydell**?	30.6%
Type	which **city** did Bill Clinton born?	38.8%
Explicit Temporal	who is the governor of Kentucky **2012** ?	10.4%
Implicit Temporal	who is the us president **when the Civil War started**?	3.5%
Ordinal	what is the **second longest** river in China?	5.1%
Aggregation	**how many** children does bill gates have?	1.2%

- Query Graph
  - Node: constant nodes such as entities or attribute values, variable nodes representing unknown entity/attribute value.
  - Edge: relation or function, e.g., “<, Max, Min, Limit”

Parse and Execute

- First construct multi-hop query graph, then add constraints
  - Bao et al., 2016, encode similarities by CNN
- Incorporate constraints and extend relation simultaneously
  - Lan et al. , 2020, encode similarities by BERT
  - Qiu et al., 2020, encode similarities by LSTM and transformer
  - Chen et al, 2020, encode similarities by graph transformer

Parse and Execute

- Train an encoder-decoder model
- Natural language question => sparql

> What colors do the school where **Donald Stanley Marshall** is grad student use?

```
SELECT DISTINCT ?x WHERE
?c educational_institution.students graduates ?k
?c education.student Donald Stanley Marshall
?c educational_institution.colors ?x
```

- Shi et al., 2020, BART; Das et al., 2021, BIGBIRD
- High accuracy, but depend on the large annotated (natural language question, sparql) pairs.

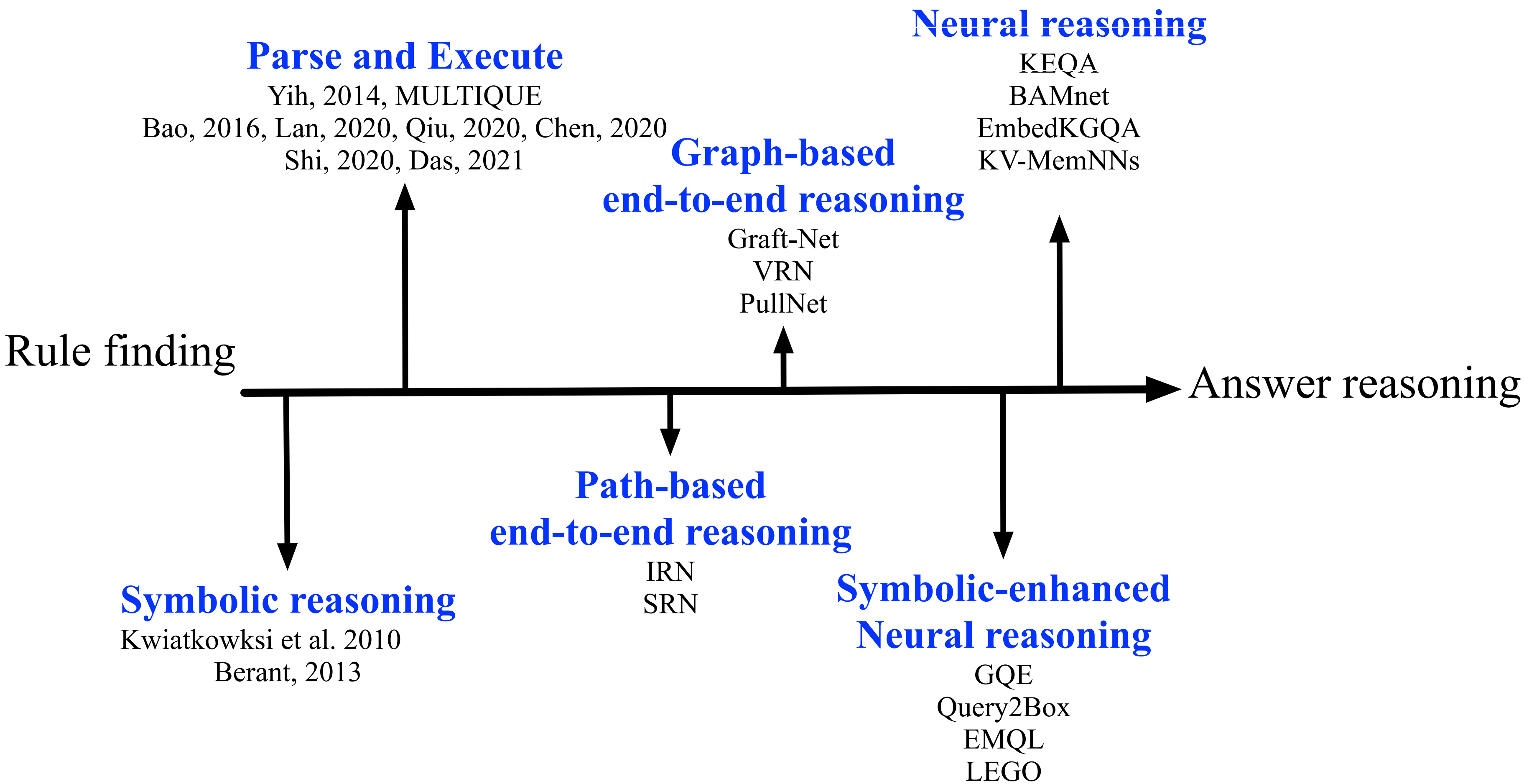
End-to-End Reasoning

- Multi-hop questions
- Path-based reasoning
- IRN (Zhou et al., 2018)
  - Input module: update the query embedding
  - Reasoning module: based on the question embedding and the historical path
  - The paths are observed
- SRN (Qiu et al., 2020)
  - Paths are unobserved. RL

End-to-End Reasoning

- Multi-hop questions
- Graph-based reasoning
  - Graft-Net (Sun et al., 2018)
    - Extract subgraphs around the topic entity in the question by PPR (Ad-hoc)
    - Perform GNN to represent nodes
  - PullNet (Sun et al., 2019)
    - Weak supervision by RL (shortest paths between topic entities and answer entities)
  - NSM (He et al.,2021)
    - Teacher-student, student finds the correct answer, teacher learns intermediate supervision signals by bidirectional reasoning

Summary of KGQA



Benchmark of KGC

- **FB15K**: a subset of Freebase. The main relation types are **symmetry/antisymmetry** and **inversion** patterns.
- **WN18**: a subset of WordNet. The main relation types are **symmetry/antisymmetry** and **inversion** patterns.
- **FB15K-237**: a subset of FB15K, where inversion relations are deleted. The main relation types are **symmetry/antisymmetry** and **composition** patterns.
- **WN18RR**: a subset of WN18, where inversion relations are deleted. The main relation types are **symmetry/antisymmetry** and **composition** patterns.

Dataset	#entity	#relation	#training	#validation	#test
FB15k	14,951	1,345	483,142	50,000	59,071
WN18	40,943	18	141,442	5,000	5,000
FB15k-237	14,541	237	272,115	17,535	20,466
WN18RR	40,943	11	86,835	3,034	3,134

Refer to “Neural and Symbolic Logical Reasoning on Knowledge Graphs. Jian Tang”

Benchmark of KGQA

Datasets	KB	Size	LF	NL
WebQuestions [Berant *et al.*, 2013]	Freebase	5,810	No	No
ComplexQuestions [Bao *et al.*, 2016]	Freebase	2,100	No	No
Multi-hop: $\boxed{\text{WebQuestionsSP [Yih et al., 2016]}}$	Freebase	4,737	Yes	Yes
composition, conjunction, comparative, superlative: $\boxed{\text{ComplexWebQuestions}}$ [Talmor and Berant, 2018]	Freebase	34,689	Yes	Yes
QALD series [Lopez *et al.*, 2013]	DBpedia	-	Yes	Yes
LC-QuAD [Trivedi *et al.*, 2017]	DBpedia	5,000	Yes	Yes
LC-QuAD 2.0 [Dubey *et al.*, 2019]	DBpedia, Wikidata	30,000	Yes	Yes
MetaQA Vanilla [Zhang *et al.*, 2018]	WikiMovies	400k	No	No
CFQ [Keysers *et al.*, 2020]	Freebase	239,357	Yes	No
Zero-shot: $\boxed{\text{GrailQA [Gu et al., 2020]}}$	Freebase	64,331	Yes	Yes
More constraints: $\boxed{\text{KQA Pro [Shi et al., 2020]}}$	Wikidata	117,970	Yes	Yes

Table 1: Several complex KBQA benchmark datasets. "LF" denotes whether the dataset provides Logic Forms, and "NL" denotes whether the dataset incorporates crowd workers to rewrite questions in Natural Language.

Refer to "A Survey on Complex Knowledge Base Question Answering:

Future Directions

- Complex questions
  - Symbolic reasoning
    - Can easily handle complex questions
    - Depend on large annotated question-sparql pairs.
    - How to automatically generate training data?
  - Neural reasoning
    - Only question-answer pairs are required.
    - Difficult to address various constraints
    - How to identify and express logic operations by NN?

Future Directions

- Pipeline
  - Topic entity identification
  - Entity linking
  - Relation detection
  - Answer reasoning

- Multi-task learning

  (Srivastava et al. 2021, Wang et al.)
  - Share BERT encoders across tasks

Future Directions

- Few-shot Reasoning

Reference

(Petersburg, SubPartOf, Virginia)
(Vacaville, SubPartOf, California)
(Prague, SubPartOf, Czech)
(Cavaliers, SubPartOf, NBA)
(Los Angeles Lakers, SubPartOf, NBA)

Query

(Chicago Bulls, SubPartOf, NBA)

  - Few-shot KGC (Sheng et al. 2020)
  - Zero-shot KGC (Teru et al, 2020)
  - Few-shot KGQA (Hua et al. 2020)
  - Zero-shot Cross-lingual KGQA (Zhou et al. 2021)
  - Dataset: I.I.D, Compositional Generalization, Zero-shot Generalization, Gu et al., 2021

Future Directions

- Temporal knowledge graph
  - (Barack Obama, held position, President of USA, 2008, 2016)

Reasoning	Example Template	Example Question
Simple time	When did {head} hold the position of {tail}	*When did Obama hold the position of President of USA*
Simple entity	Which award did {head} receive in {time}	*Which award did Brad Pitt receive in 2001*
Before/After	Who was the {tail} {type} {head}	*Who was the President of USA before Obama*
First/Last	When did {head} play their {adj} game	*When did Messi play their first game*
Time join	Who held the position of {tail} during {event}	*Who held the position of President of USA during WWII*

  - Saxena et al. (ACL 2021)
  - A temporal KBQA dataset
  - Revised EmbedKGQA (temporal KG embeding)

Future Directions

- Fuse Text and KG
  - Build entity-relation-entity from text, Fu 2019, Lu, 2019)
  - Build entity-text from text, Sun et al., 2018, Sun et al., 2019, Han et al., 2020
  - Without building the new edges from text, directly encode text, Xiong et al., 2019
  - Virtual KB, Dhingra et al, 2020, Sun et al., 2021
  - Unitedly encode text and KG by pre-trained LMs?

Thank you!

Neural-Symbolic Reasoning on Knowledge Graphs

Jing Zhang, School of Information, Renmin University of China