

salesforce

Reinforcement Learning for Knowledge Graph Reasoning

Knowledge Connexions Conference 2020

Victoria Lin

Senior Research Scientist, Salesforce AI Research

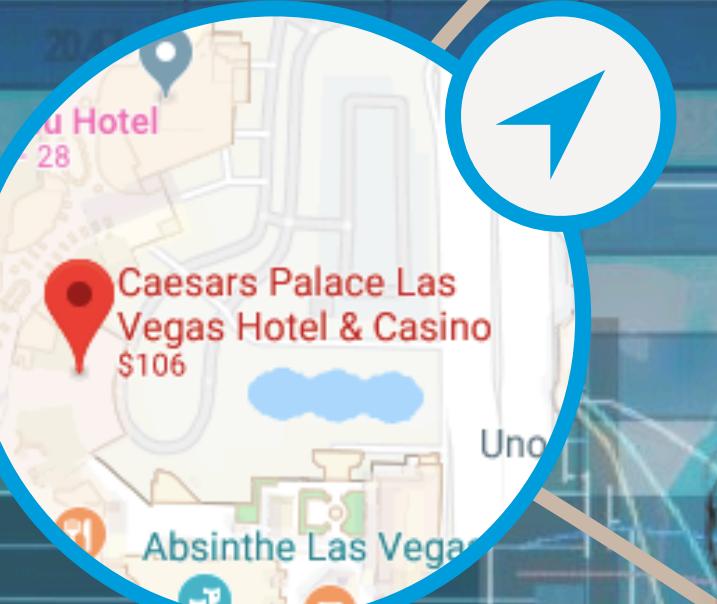
 @VictoriaLinML



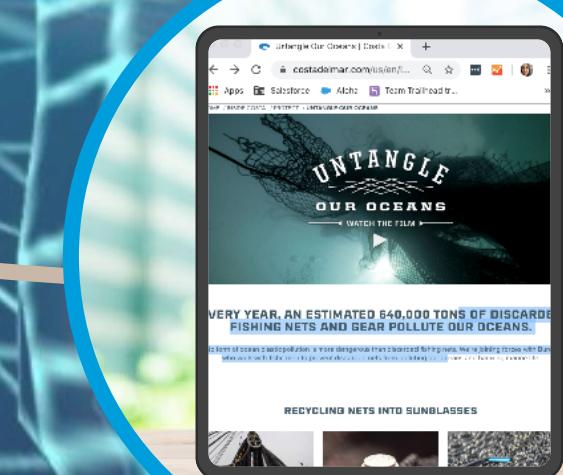
Personal



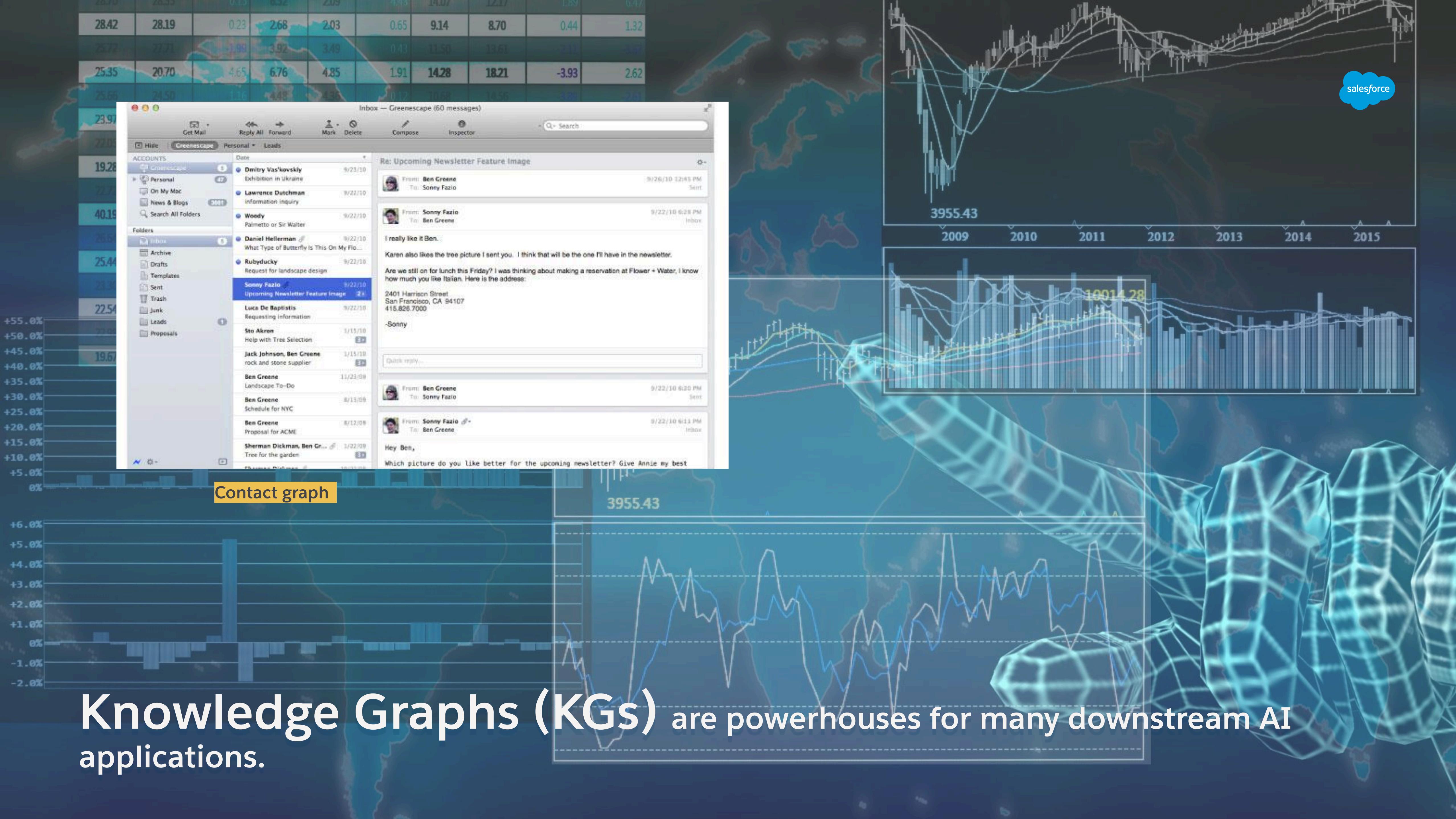
Intelligent

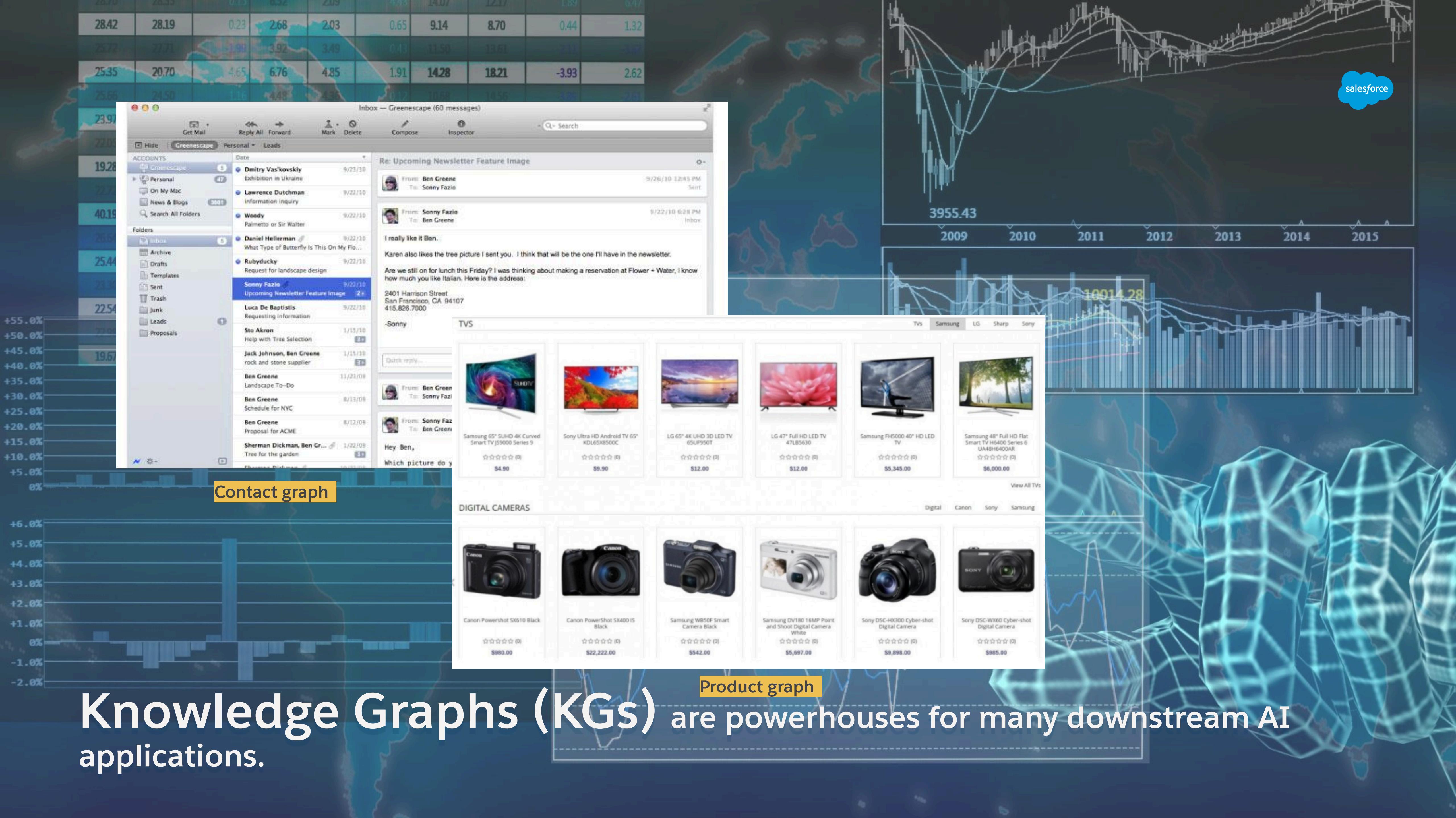


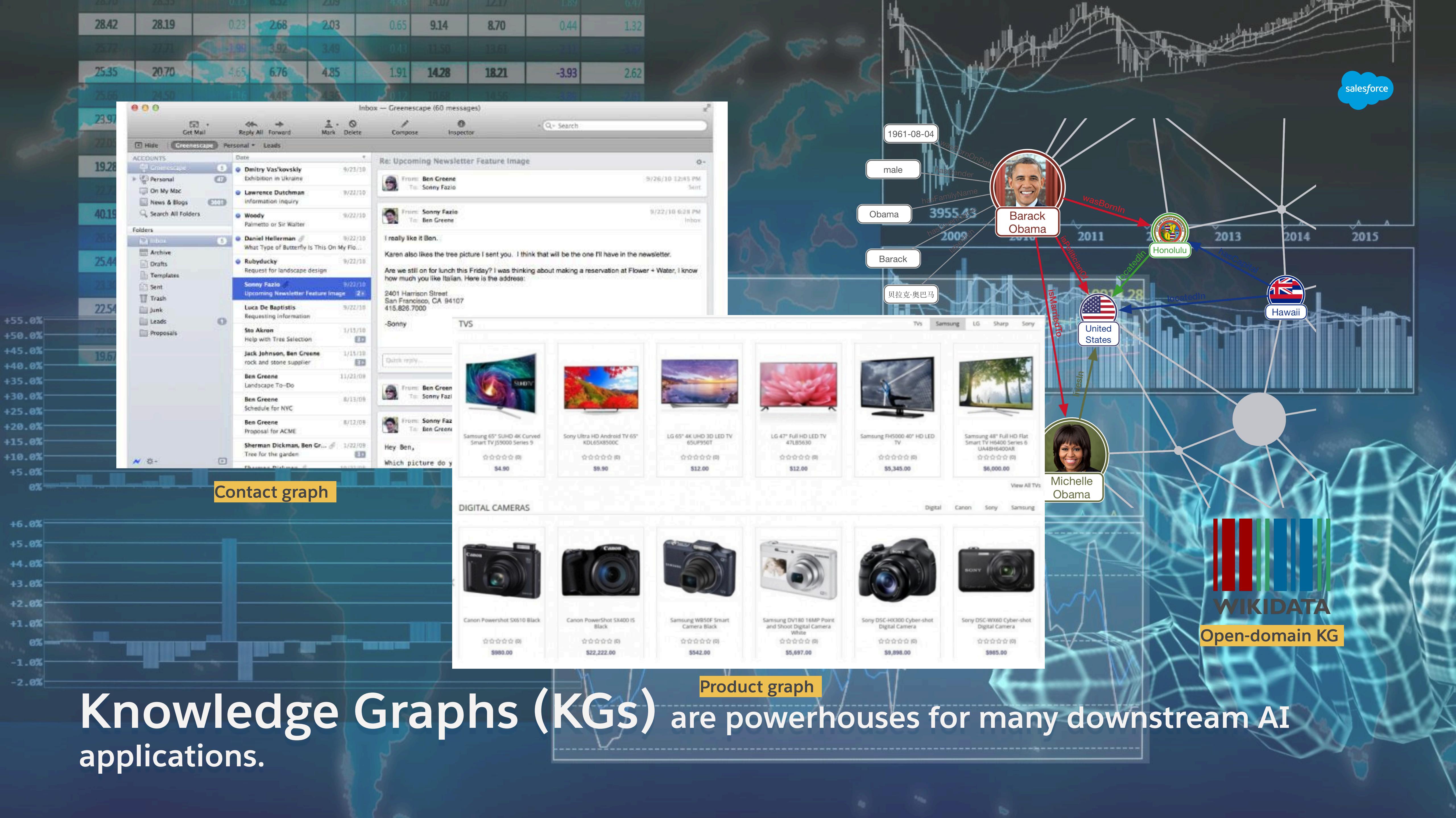
Conversational



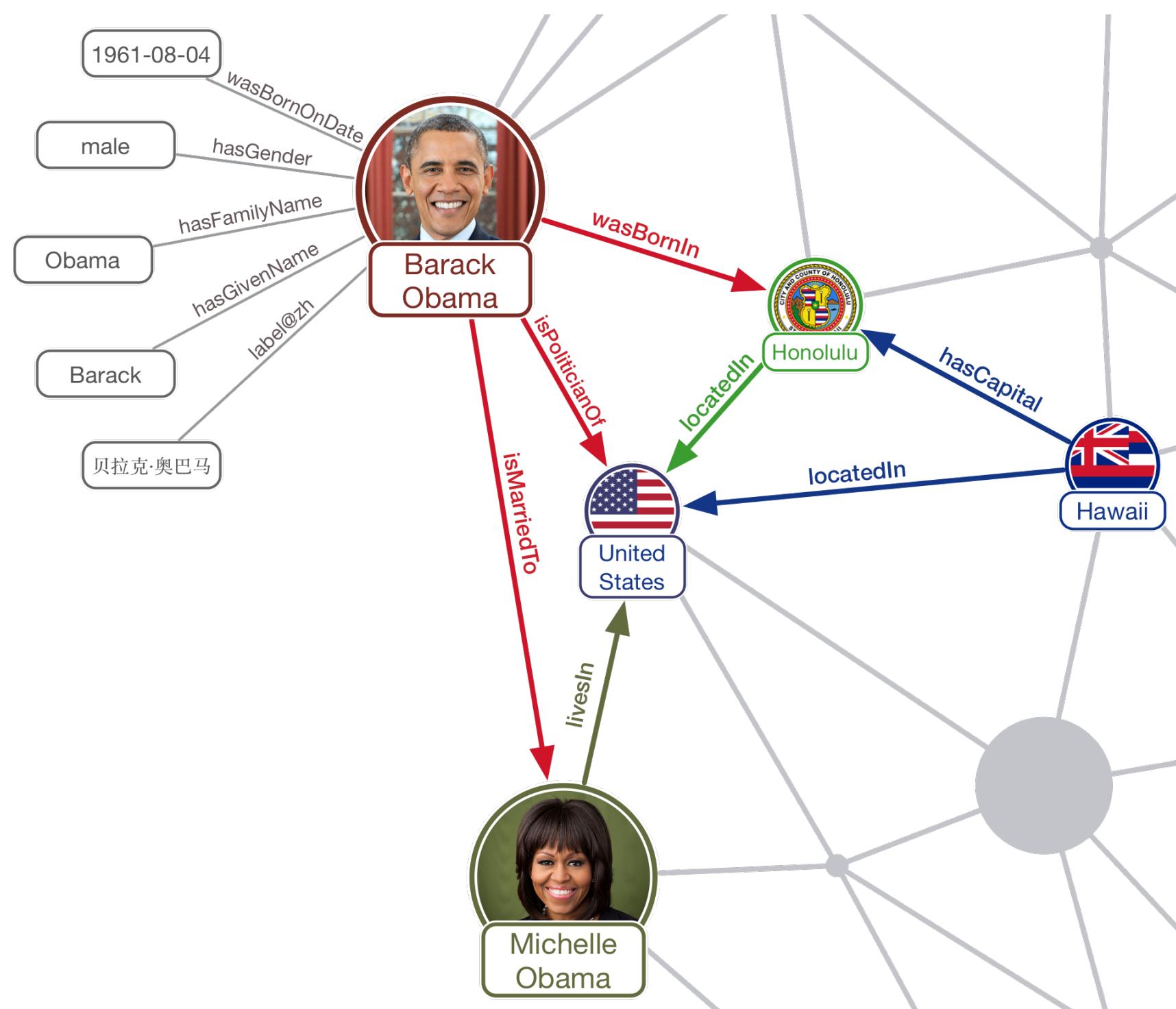
salesforce







Knowledge Graph Search

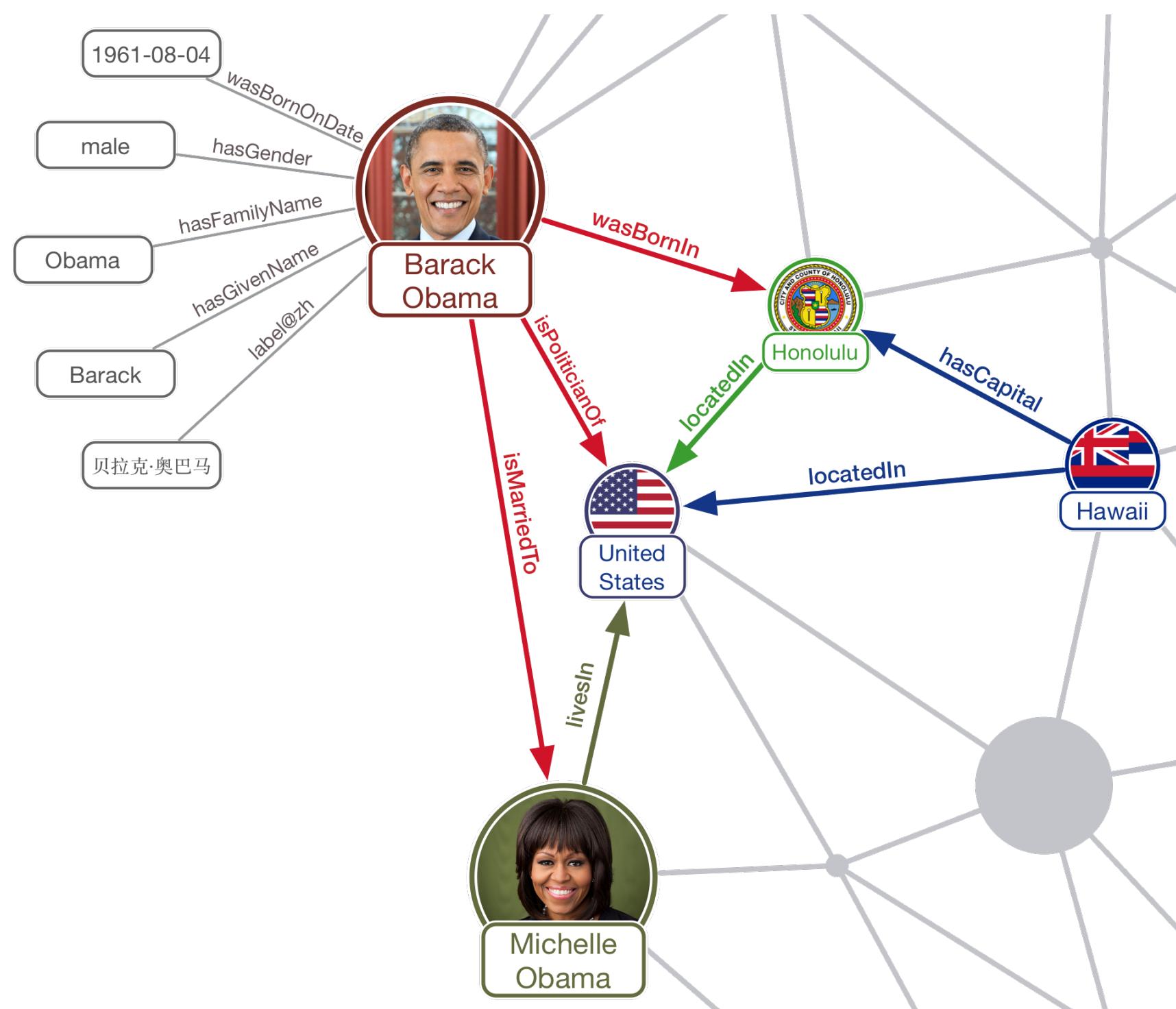


Search:

Look up for facts that exist in the knowledge graph via a formal query language (e.g. SPARQL)

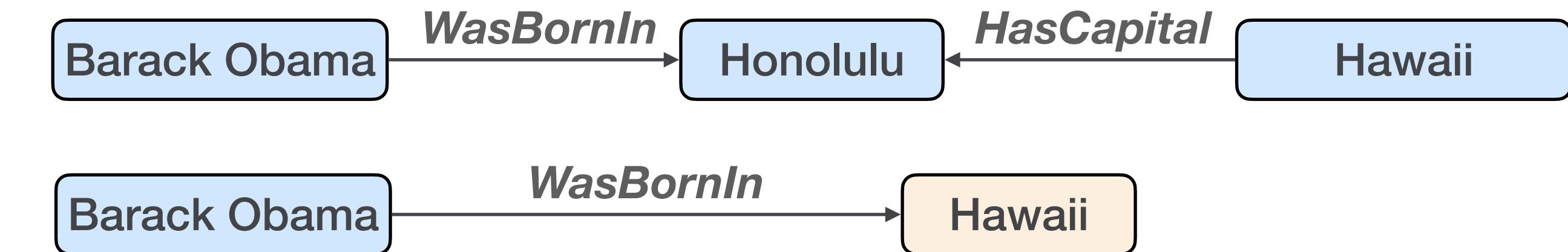


Knowledge Graph Reasoning

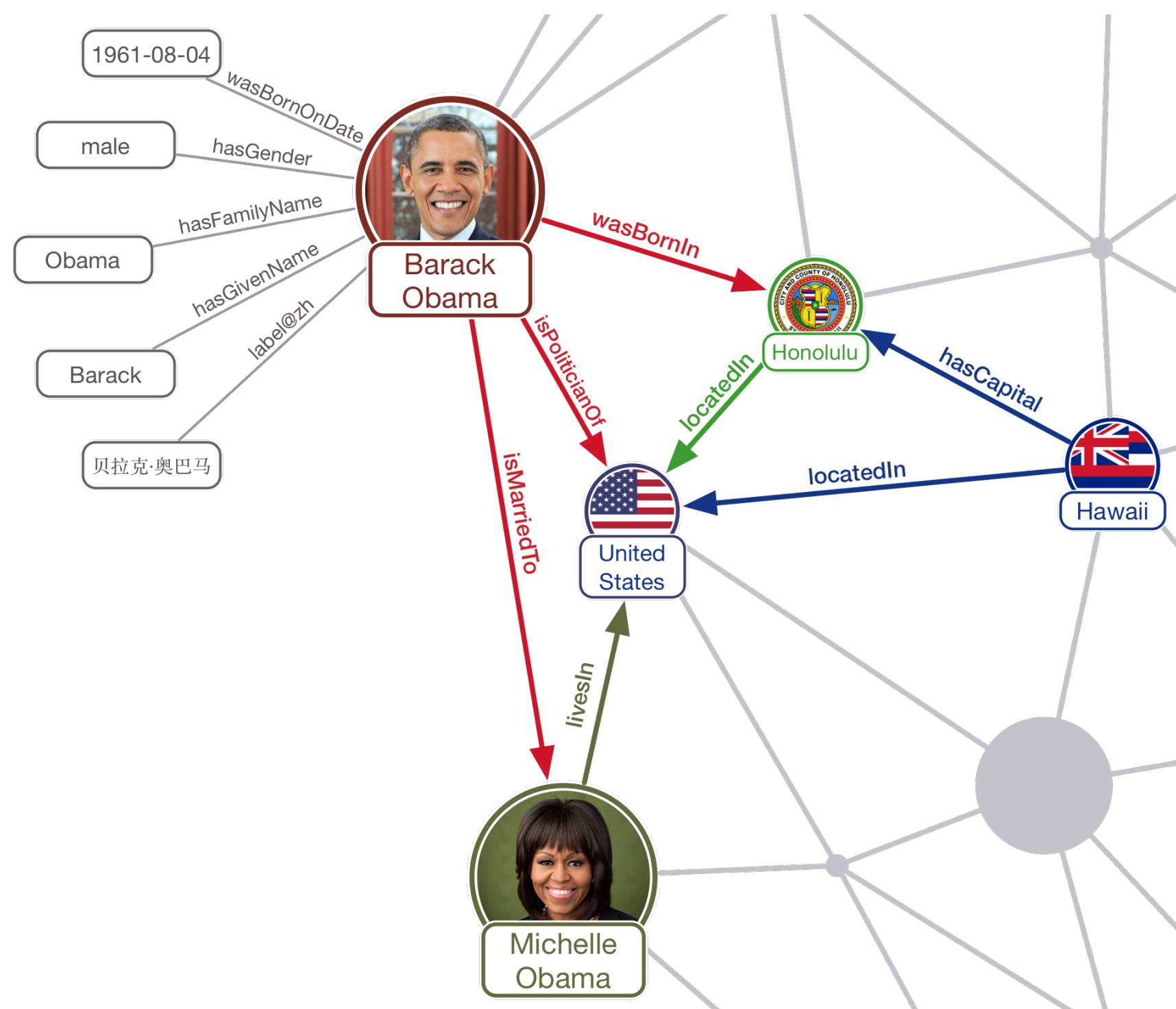


Inference/Reasoning:

Derive additional knowledge given existing facts

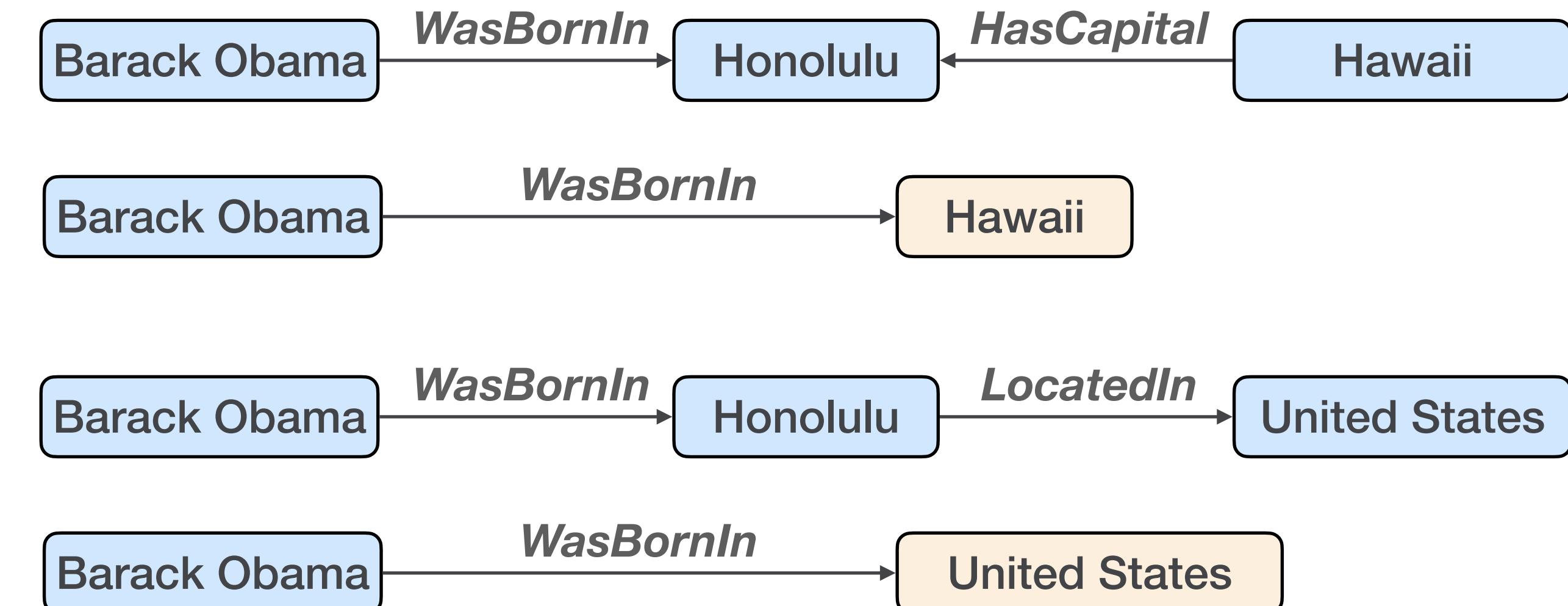


Knowledge Graph Reasoning

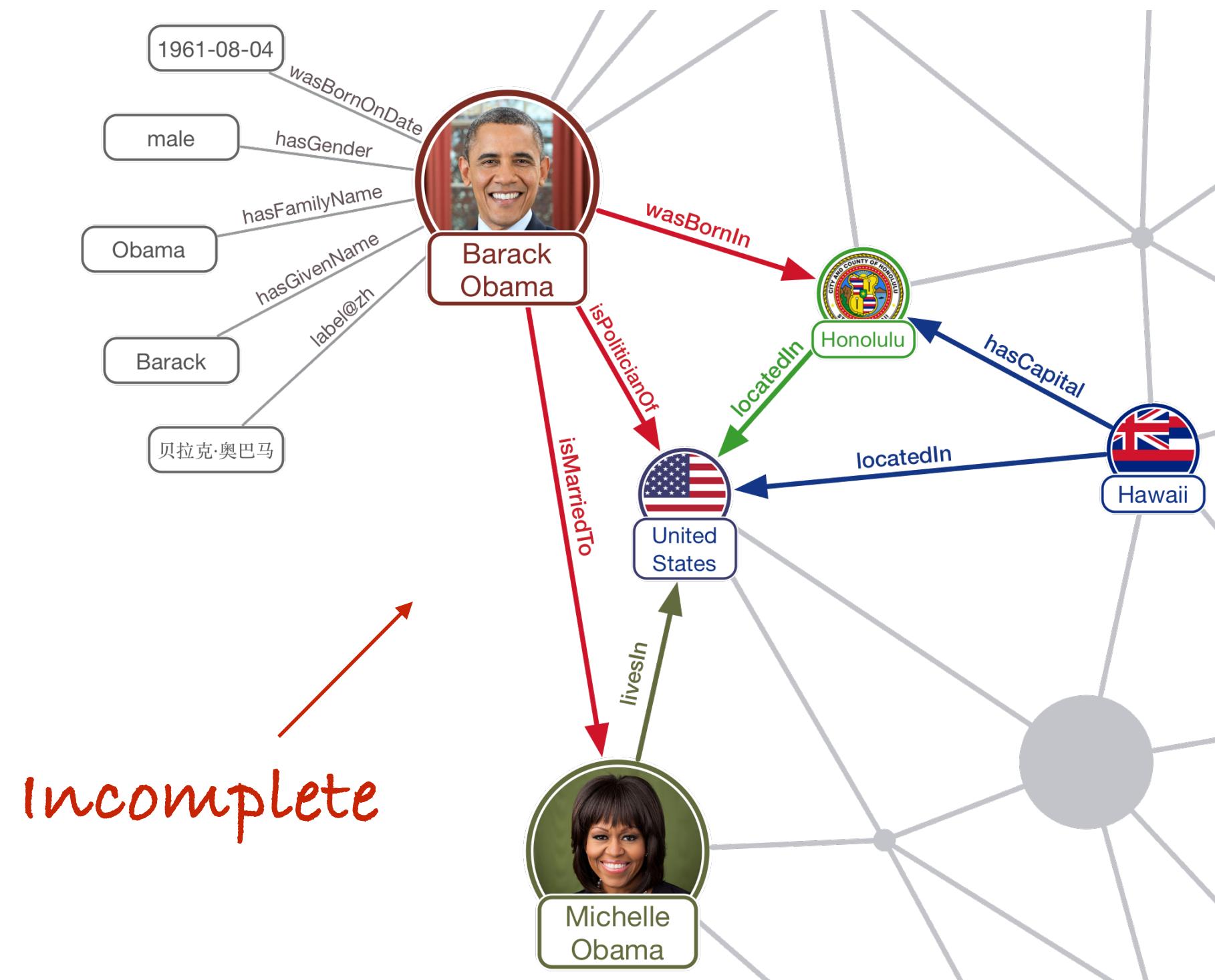


Inference/Reasoning:

Derive additional knowledge given existing facts

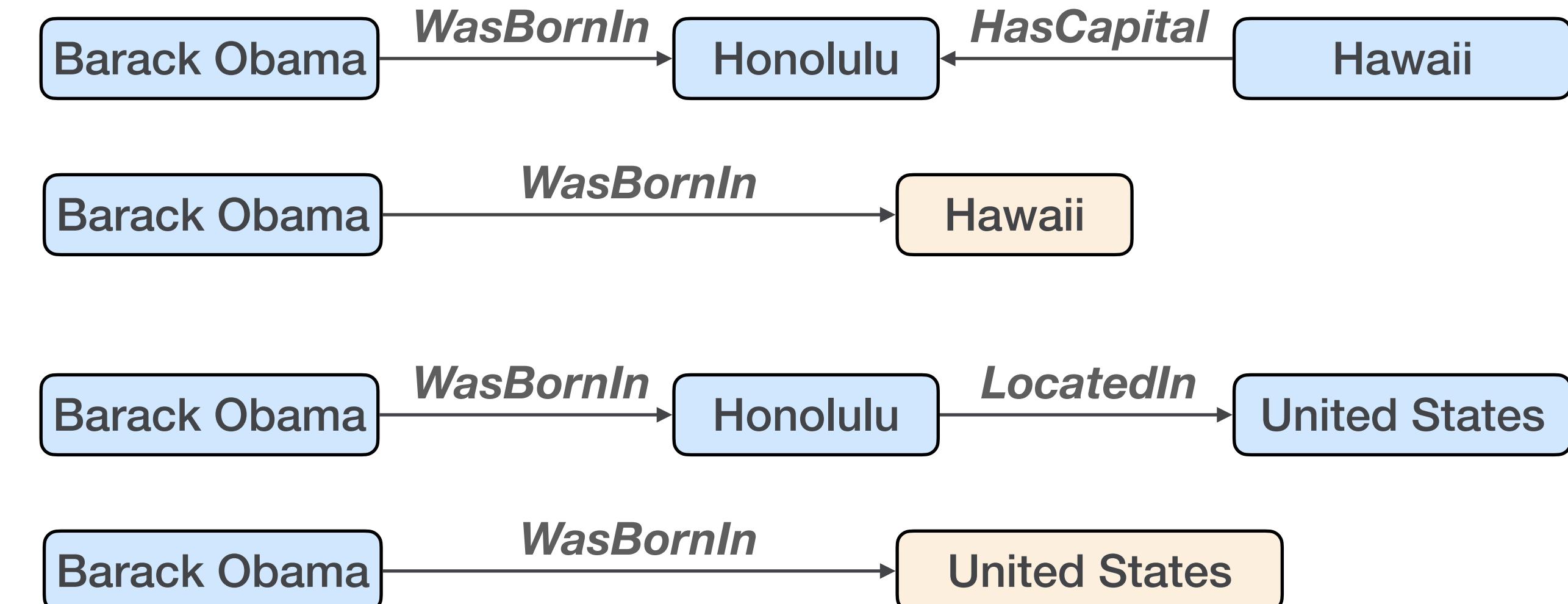


Knowledge Graph Reasoning



Inference/Reasoning:

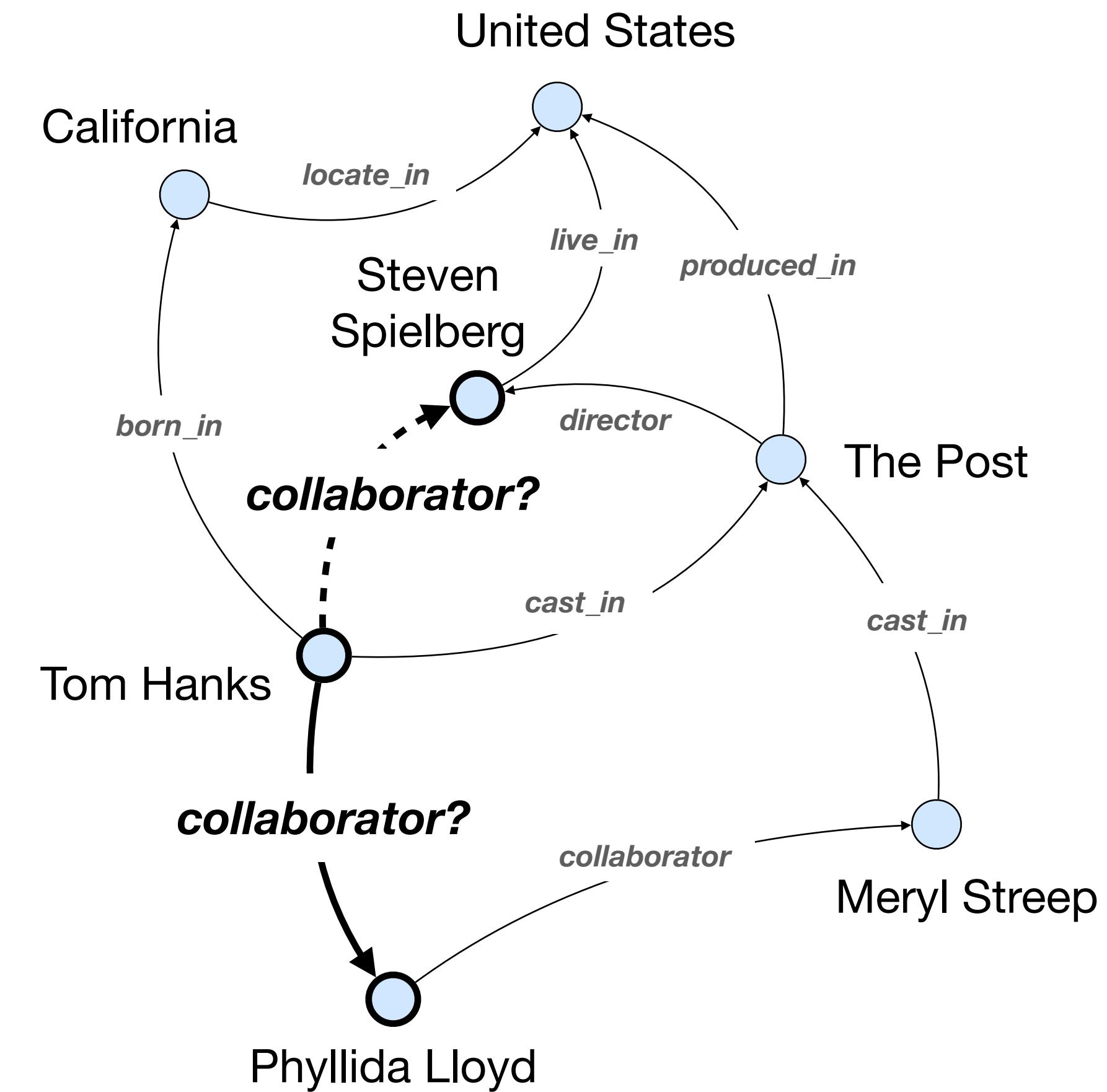
Derive additional knowledge given existing facts



Reasoning is a core problem for KGs as most often it is impossible to curate and store all facts in a KG.

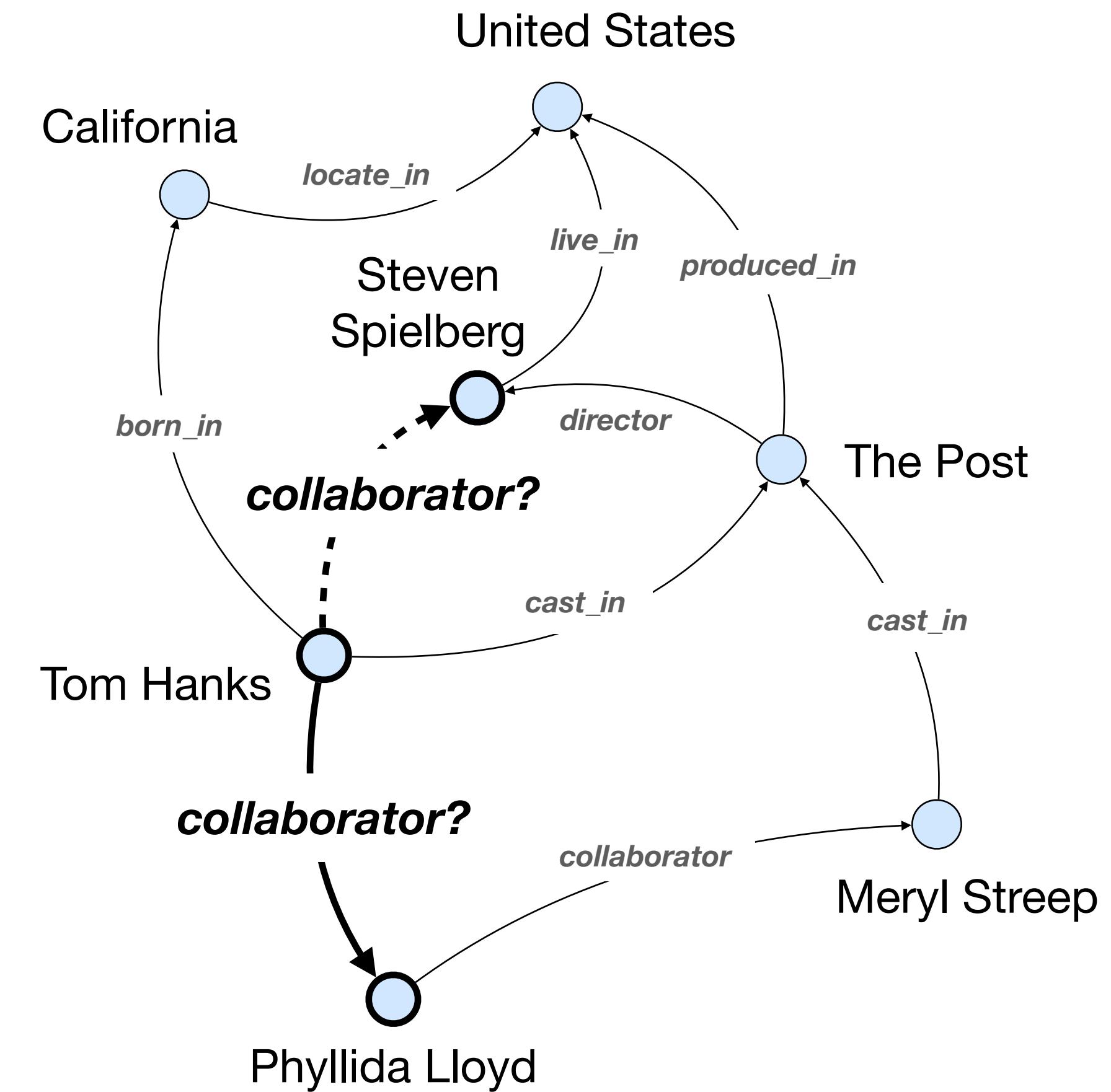
Knowledge Graph Reasoning

- Knowledge Graph Embeddings
- Path Ranking Algorithm (PRA) 
- Sequential Decision Making 

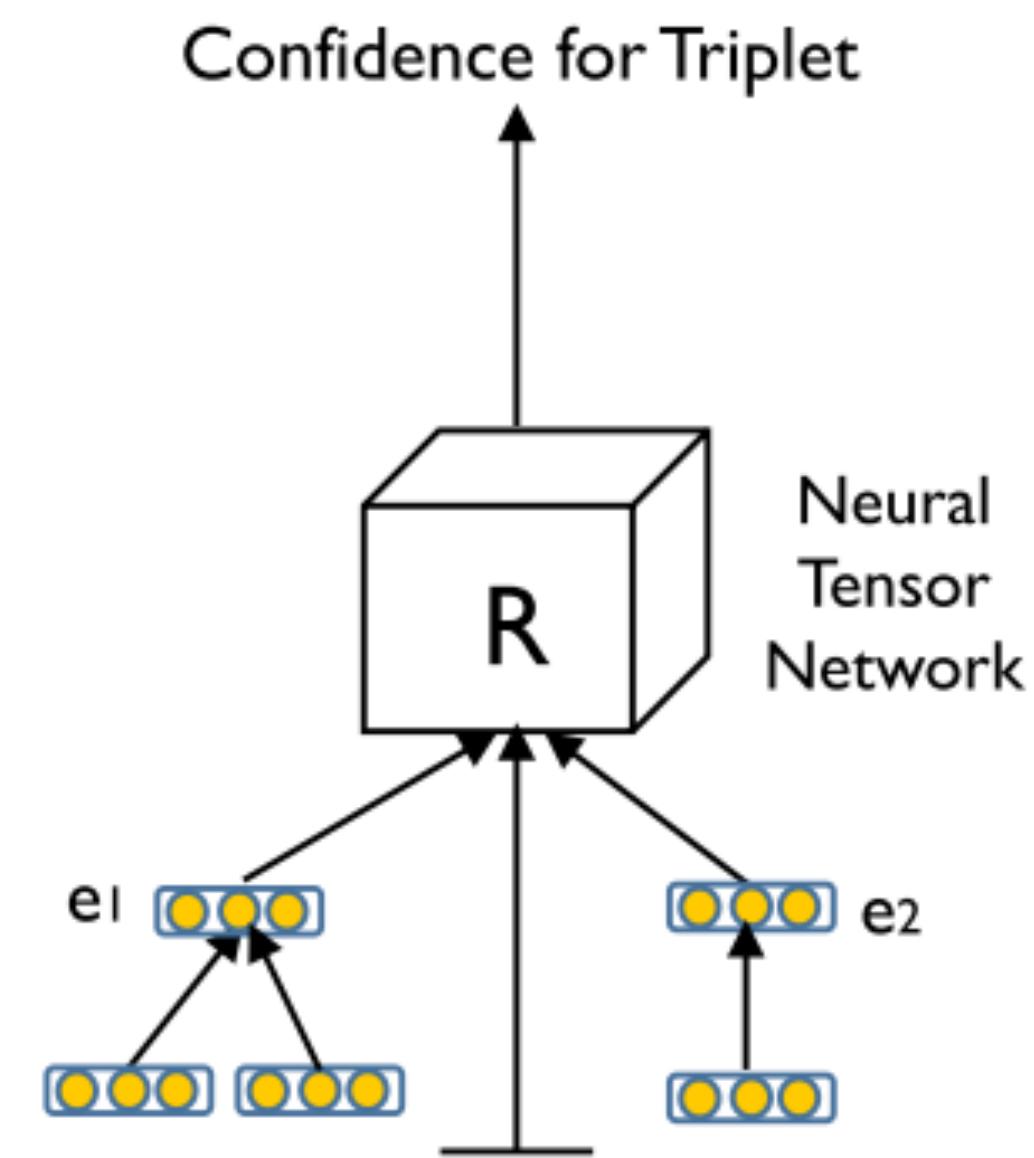
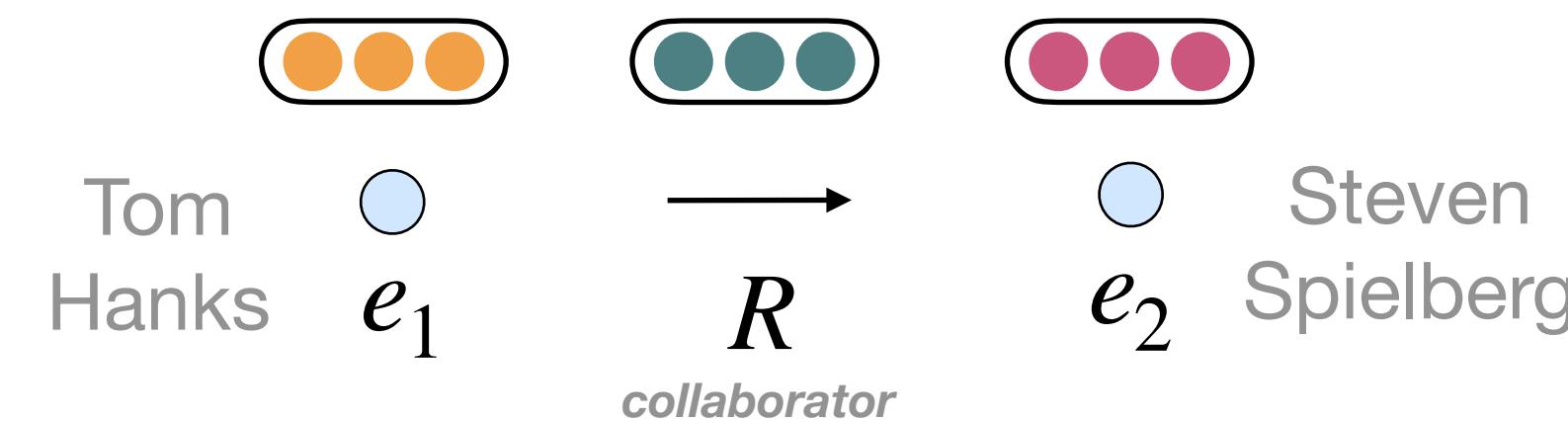


Knowledge Graph Reasoning

- Knowledge Graph Embeddings
- Path Ranking Algorithm (PRA)
- Reinforcement Learning

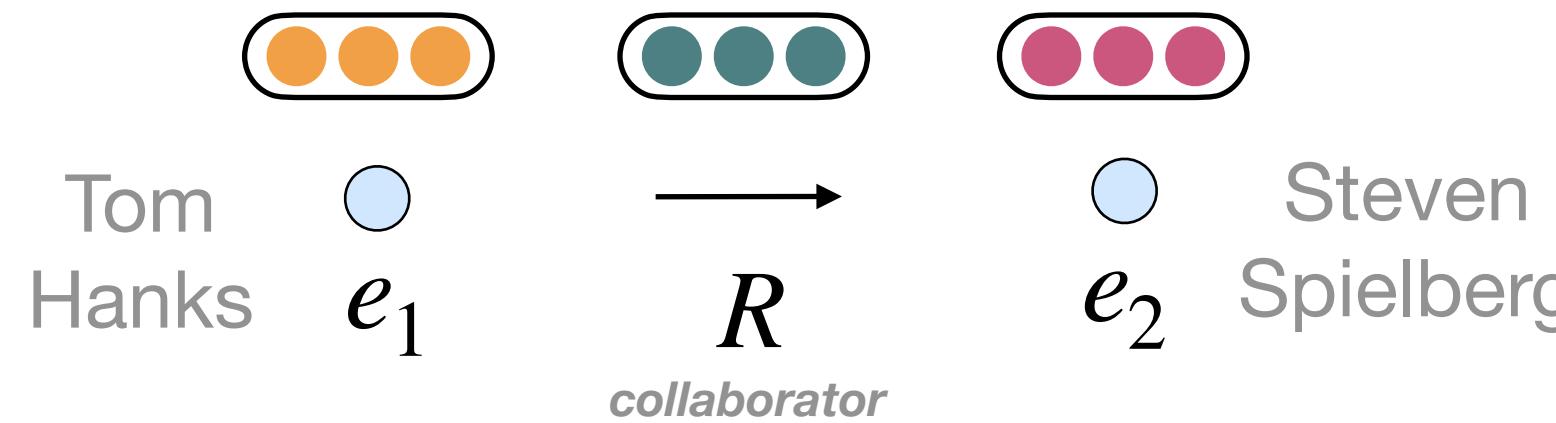


Knowledge Graph Embeddings



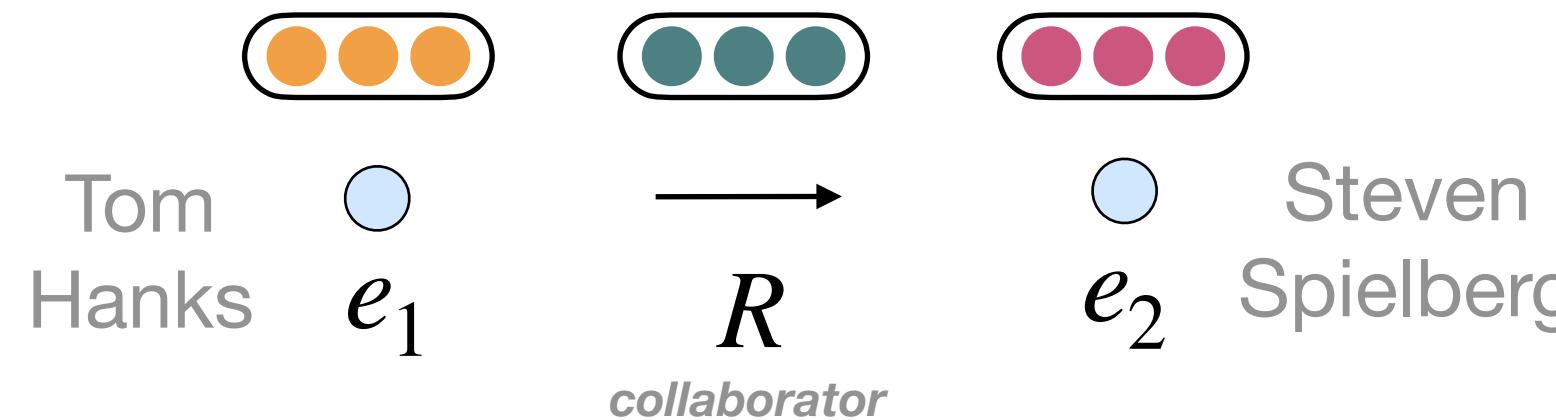
NTN (Socher et. al. 2013), DistMult (Yang et al. 2015), ComplEx (Trouillon et al. 2016), ConvE (Dettmers et al. 2018).

Knowledge Graph Embeddings



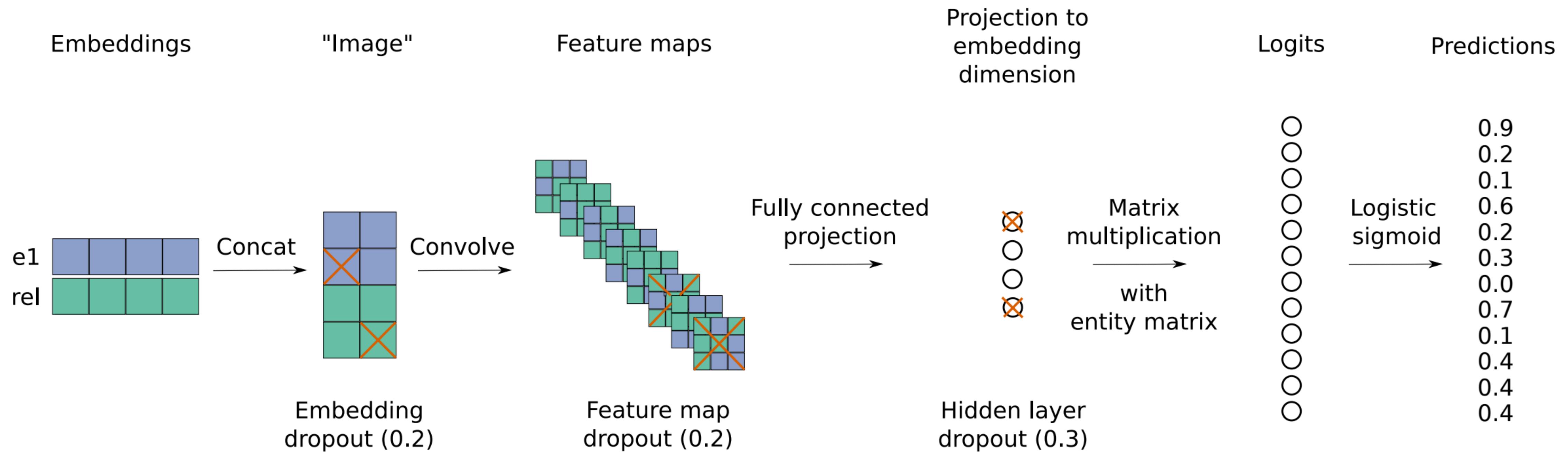
Model	Scoring Function $\psi_r(\mathbf{e}_s, \mathbf{e}_o)$	Relation Parameters	Space Complexity
SE (Bordes et al. 2014)	$\ \mathbf{W}_r^L \mathbf{e}_s - \mathbf{W}_r^R \mathbf{e}_o\ _p$	$\mathbf{W}_r^L, \mathbf{W}_r^R \in \mathbb{R}^{k \times k}$	$\mathcal{O}(n_e k + n_r k^2)$
TransE (Bordes et al. 2013a)	$\ \mathbf{e}_s + \mathbf{r}_r - \mathbf{e}_o\ _p$	$\mathbf{r}_r \in \mathbb{R}^k$	$\mathcal{O}(n_e k + n_r k)$
DistMult (Yang et al. 2015)	$\langle \mathbf{e}_s, \mathbf{r}_r, \mathbf{e}_o \rangle$	$\mathbf{r}_r \in \mathbb{R}^k$	$\mathcal{O}(n_e k + n_r k)$
ComplEx (Trouillon et al. 2016)	$\langle \mathbf{e}_s, \mathbf{r}_r, \mathbf{e}_o \rangle$	$\mathbf{r}_r \in \mathbb{C}^k$	$\mathcal{O}(n_e k + n_r k)$
ConvE	$f(\text{vec}(f([\overline{\mathbf{e}}_s; \overline{\mathbf{r}}_r] * \omega)) \mathbf{W}) \mathbf{e}_o$	$\mathbf{r}_r \in \mathbb{R}^{k'}$	$\mathcal{O}(n_e k + n_r k')$

Knowledge Graph Embeddings

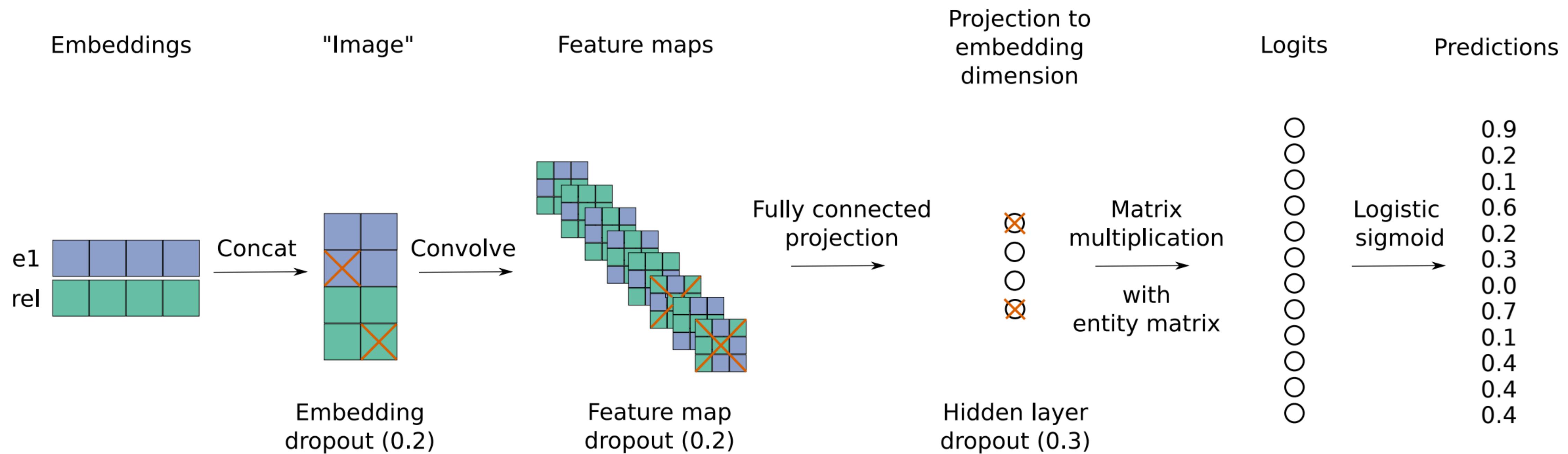


Model	Scoring Function $\psi_r(\mathbf{e}_s, \mathbf{e}_o)$	Relation Parameters	Space Complexity
SE (Bordes et al. 2014)	$\ \mathbf{W}_r^L \mathbf{e}_s - \mathbf{W}_r^R \mathbf{e}_o\ _p$	$\mathbf{W}_r^L, \mathbf{W}_r^R \in \mathbb{R}^{k \times k}$	$\mathcal{O}(n_e k + n_r k^2)$
TransE (Bordes et al. 2013a)	$\ \mathbf{e}_s + \mathbf{r}_r - \mathbf{e}_o\ _p$	$\mathbf{r}_r \in \mathbb{R}^k$	$\mathcal{O}(n_e k + n_r k)$
DistMult (Yang et al. 2015)	$\langle \mathbf{e}_s, \mathbf{r}_r, \mathbf{e}_o \rangle$	$\mathbf{r}_r \in \mathbb{R}^k$	$\mathcal{O}(n_e k + n_r k)$
ComplEx (Trouillon et al. 2016)	$\langle \mathbf{e}_s, \mathbf{r}_r, \mathbf{e}_o \rangle$	$\mathbf{r}_r \in \mathbb{C}^k$	$\mathcal{O}(n_e k + n_r k)$
ConvE	$f(\text{vec}(f([\overline{\mathbf{e}}_s; \overline{\mathbf{r}}_r] * \omega)) \mathbf{W}) \mathbf{e}_o$	$\mathbf{r}_r \in \mathbb{R}^{k'}$	$\mathcal{O}(n_e k + n_r k')$

ConvE: Convolutional 2D Knowledge Graph Embeddings

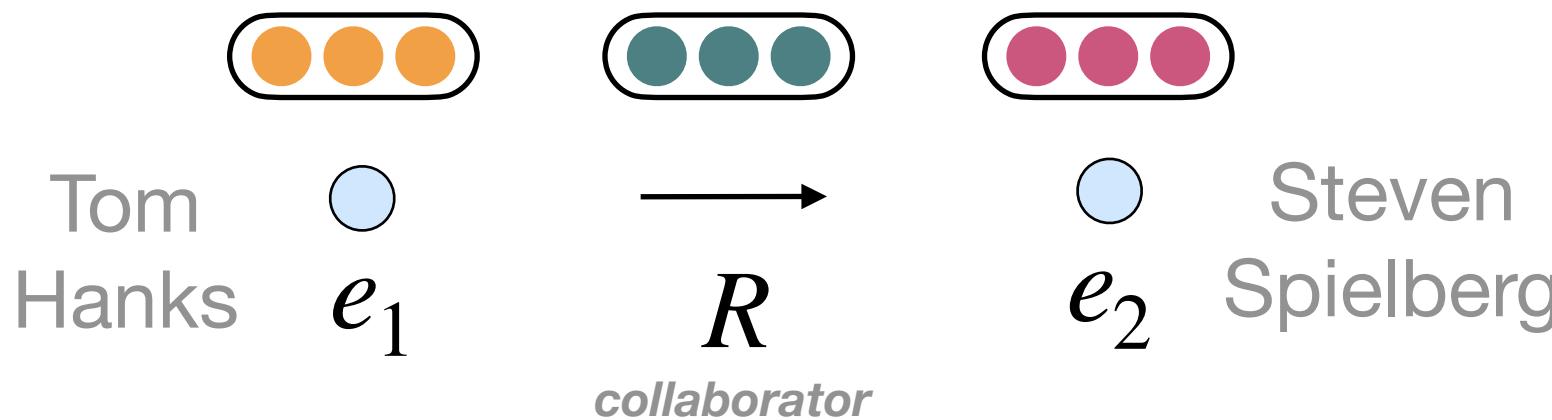


ConvE: Convolutional 2D Knowledge Graph Embeddings



 SOTA model proposed in 2018
Better scoring functions were proposed since. Please refer to the literature for more details.

Knowledge Graph Embeddings



Highly accurate &
Efficient

	Acc
WordNet	86.2
Freebase	90.0

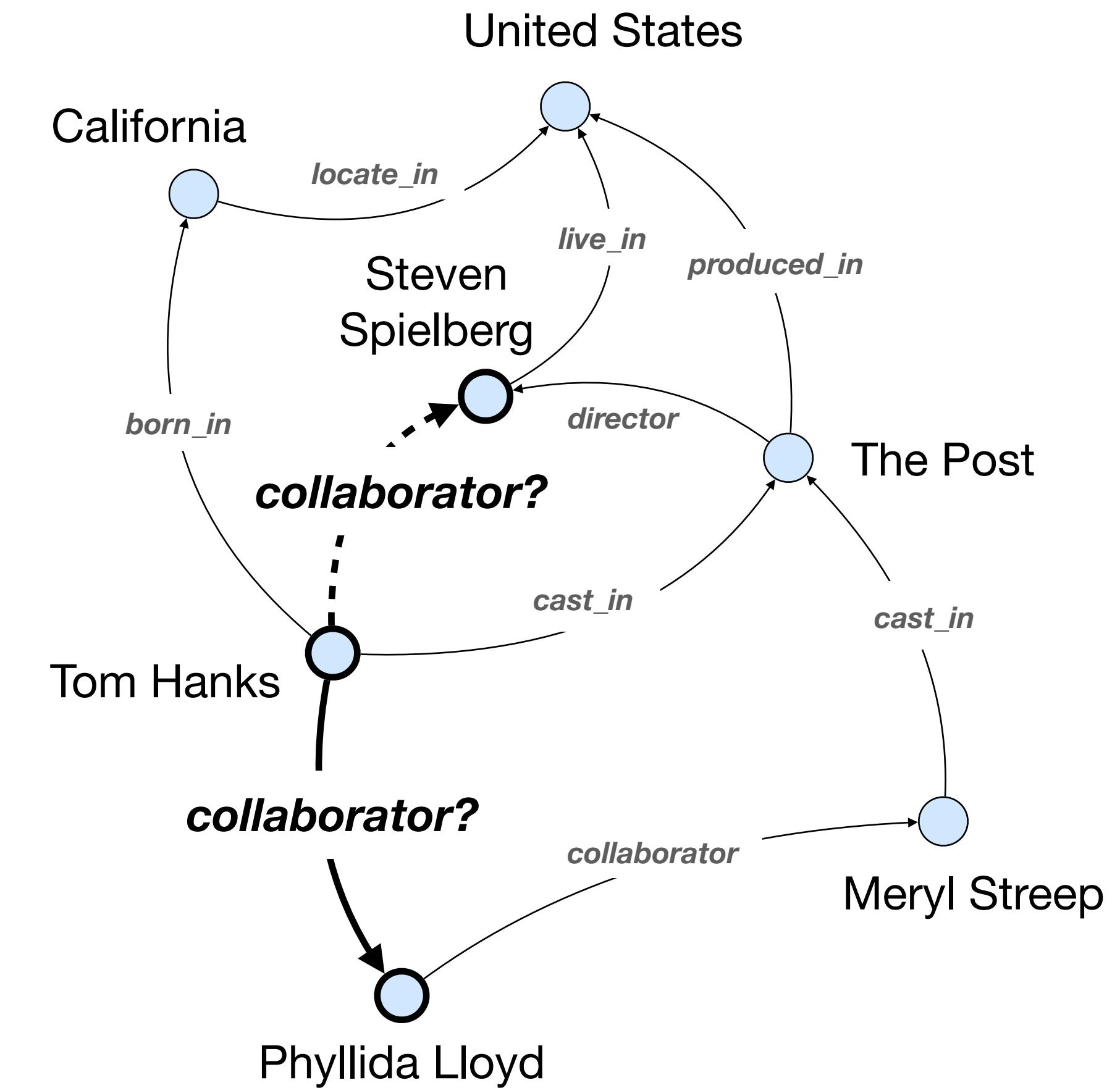
Tab 1. NTN KB fact inference performance on the WordNet and Freebase benchmarks (Socher et. al. 2013)

- Lack interpretability
- Does not perform well for rare/unseen entities

Knowledge Graph Reasoning



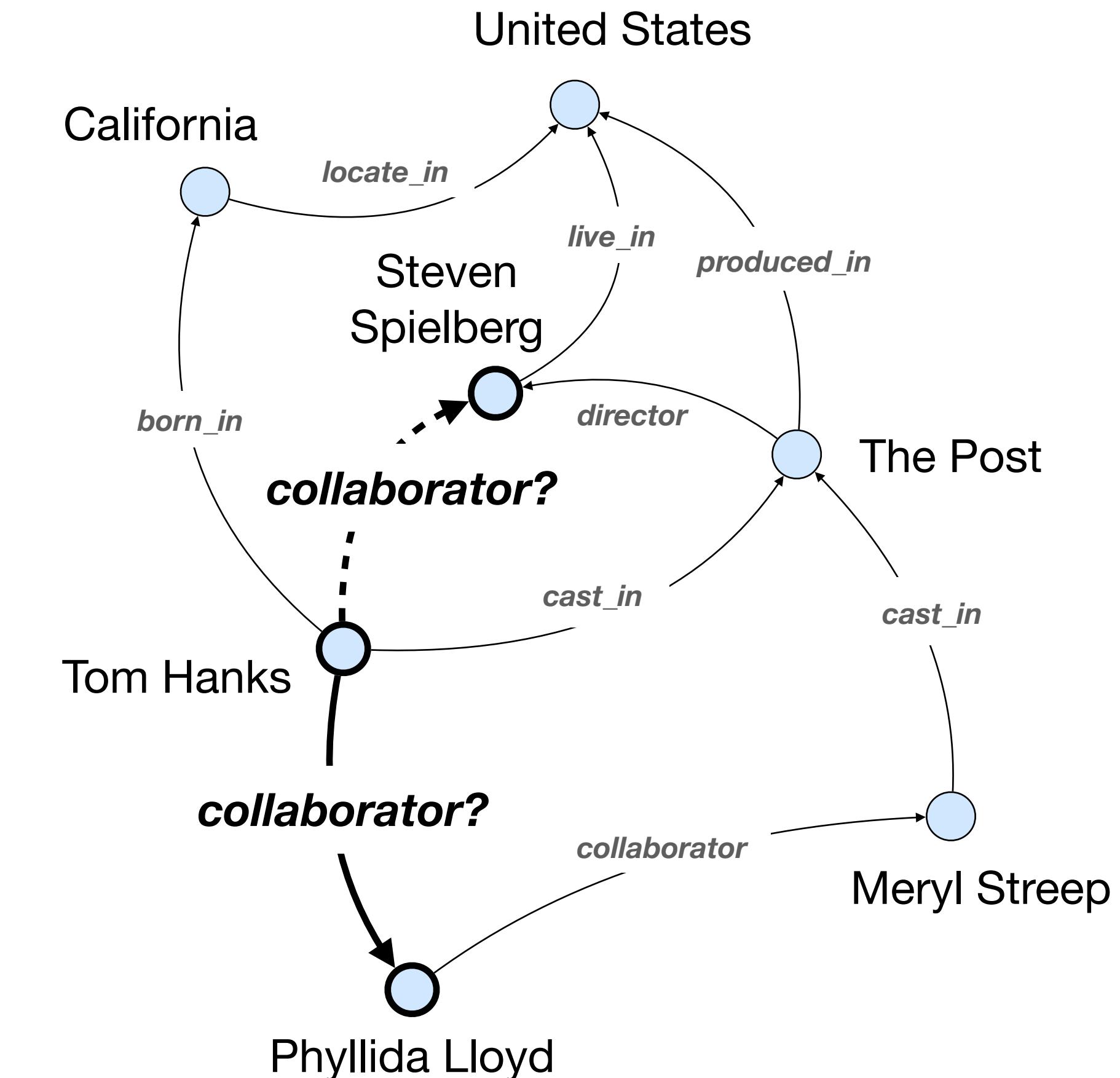
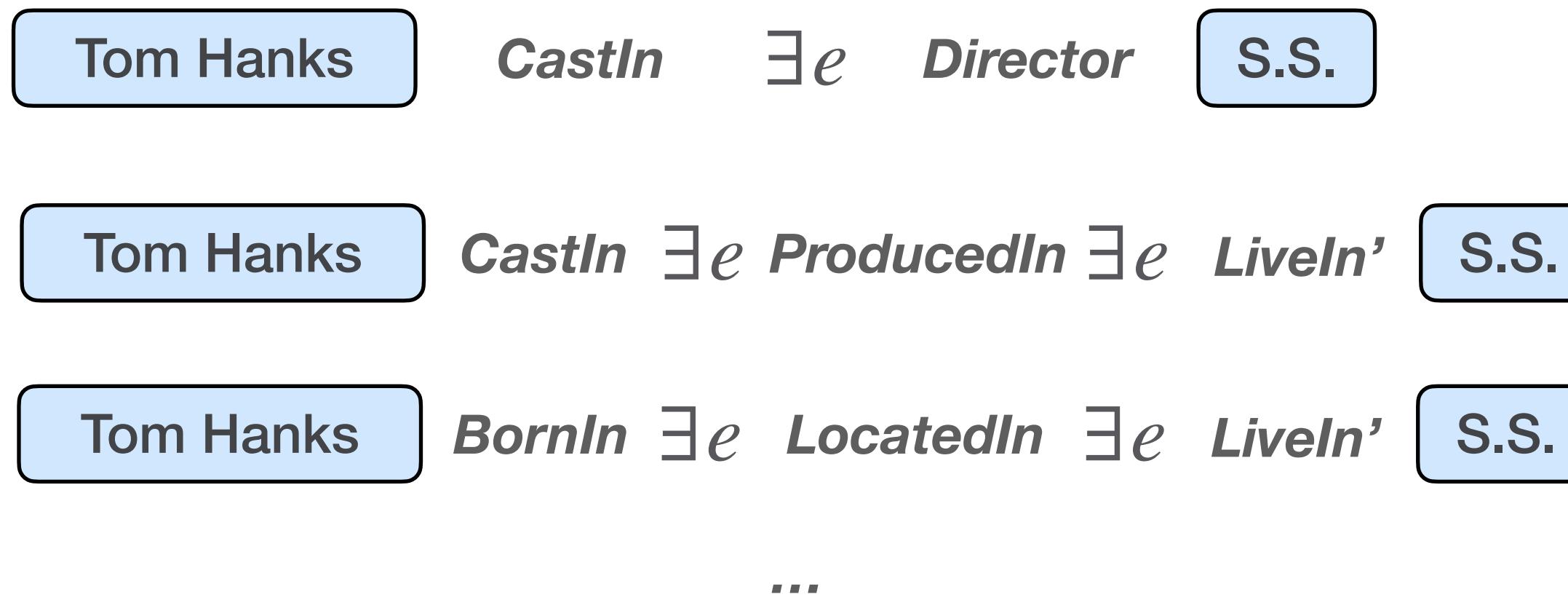
- Knowledge Graph Embeddings
- Path Ranking Algorithm (PRA) 
- Reinforcement Learning



Path Ranking Algorithm

- Identifying relation paths connecting two entities (e_1, e_2) in the KG and use the paths as features for predicting new relations.

$$P = R_1 \dots R_\ell$$



Relational Retrieval Using a Combination of Path-Constrained Random Walks. Lao and Cohen 2010.

Random Walk Inference and Learning in A Large-Scale Knowledge Base. Lao and Cohen 2011.

Finding Inference Paths

- Exhaustive (Lao and Cohen 2010)
 - Obtaining all paths connecting e_1 and e_2 (dynamic programming)
 -

Relational Retrieval Using a Combination of Path-Constrained Random Walks. Lao and Cohen 2010.

Random Walk Inference and Learning in A Large-Scale Knowledge Base. Lao and Cohen 2011.

Finding Inference Paths

- Exhaustive (Lao and Cohen 2010)
 - Obtaining all paths connecting e_1 and e_2 (dynamic programming)
- Data-driven (Lao and Cohen 2011)
 - Identifying only paths that are potentially useful for an inference task
 - Any node e visited during path search must be supported by at least a fraction α of seed entities s_i seen during training
 - Any path P must retrieve at least one target entity t_i on the training set

 s_i supports e iff. the random walk probability between s_i and e is greater than 0

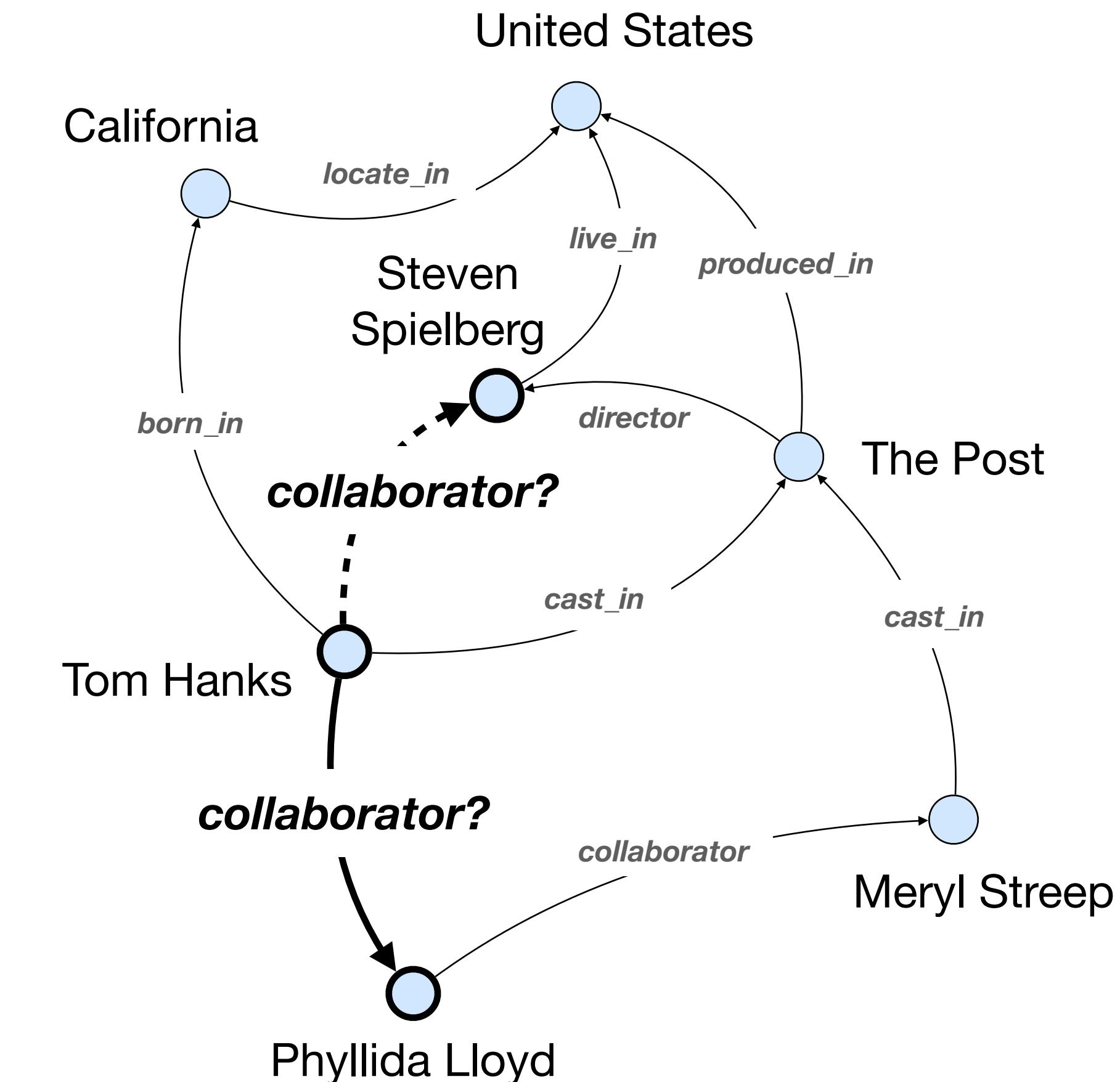
Finding Inference Paths

- Reinforcement Learning (Xiong et al. 2018)
 - Learn a policy based agent to sample the most informative paths between e_1 and e_2
 - Starting from e_1 , the agent uses a policy network to pick the most promising relation to extend its path at each step until it reaches the target entity e_2 , or has reached a maximum number of search steps.
 - Hybrid reward
$$r_{\text{GLOBAL}} = \begin{cases} +1, & \text{if the path reaches } e_{\text{target}} \\ -1, & \text{otherwise} \end{cases} \quad r_{\text{EFFICIENCY}} = \frac{1}{\text{length}(p)} \quad r_{\text{DIVERSITY}} = -\frac{1}{|F|} \sum_{i=1}^{|F|} \cos(\mathbf{p}, \mathbf{p}_i)$$
 - Supervised policy learning and retraining with reward

Path Ranking Algorithm

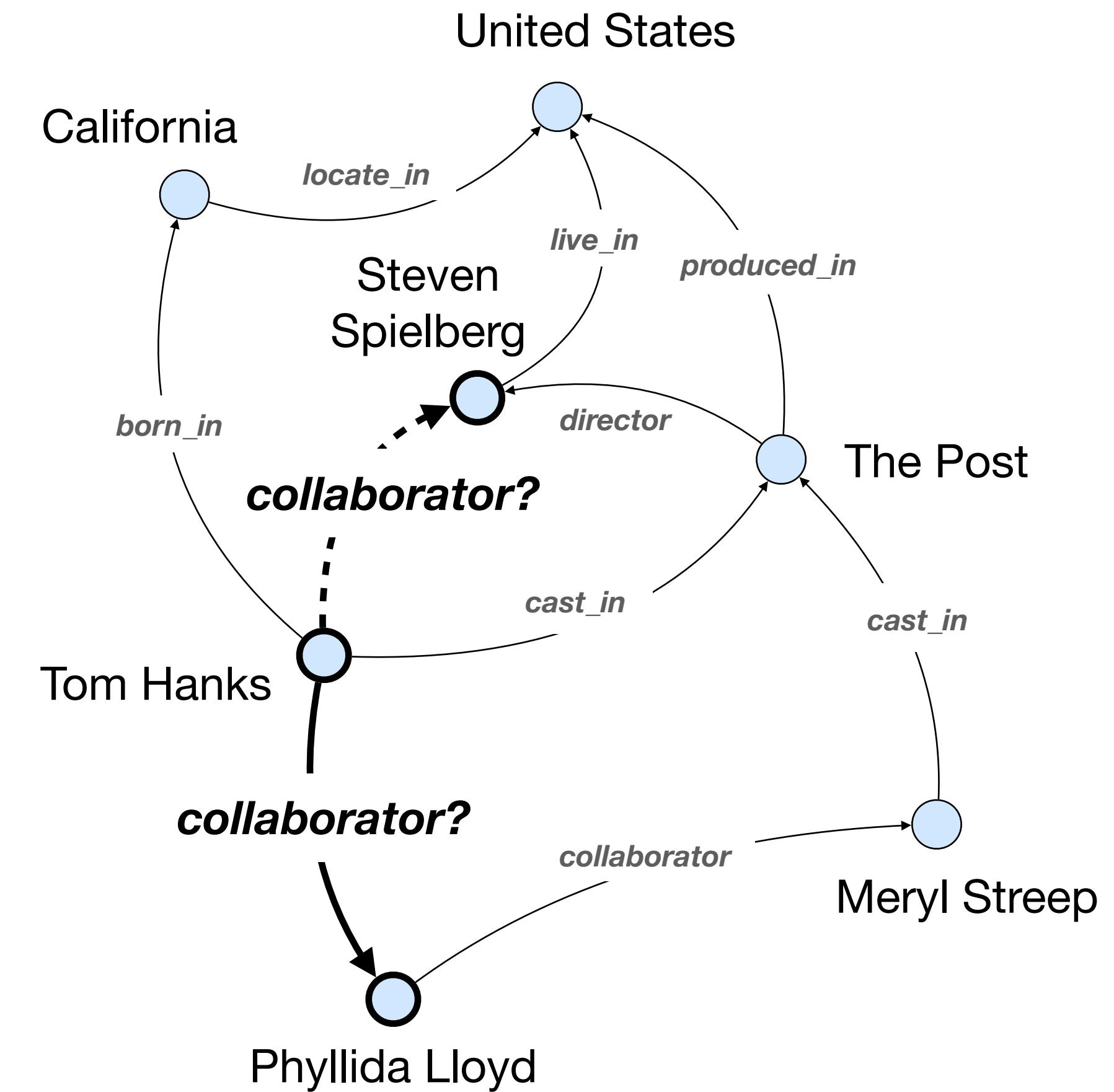


- Explainable
- Performs well for $e_s \xrightarrow{r_q} e_t$ queries
- can work for rare/unseen entities as reasoning is based on path features
- Inefficient for $e_s \xrightarrow{r_q} ?$ queries



Knowledge Graph Reasoning

- Knowledge Graph Embeddings
- Path Ranking Algorithm (PRA)
- Sequential Decision Making

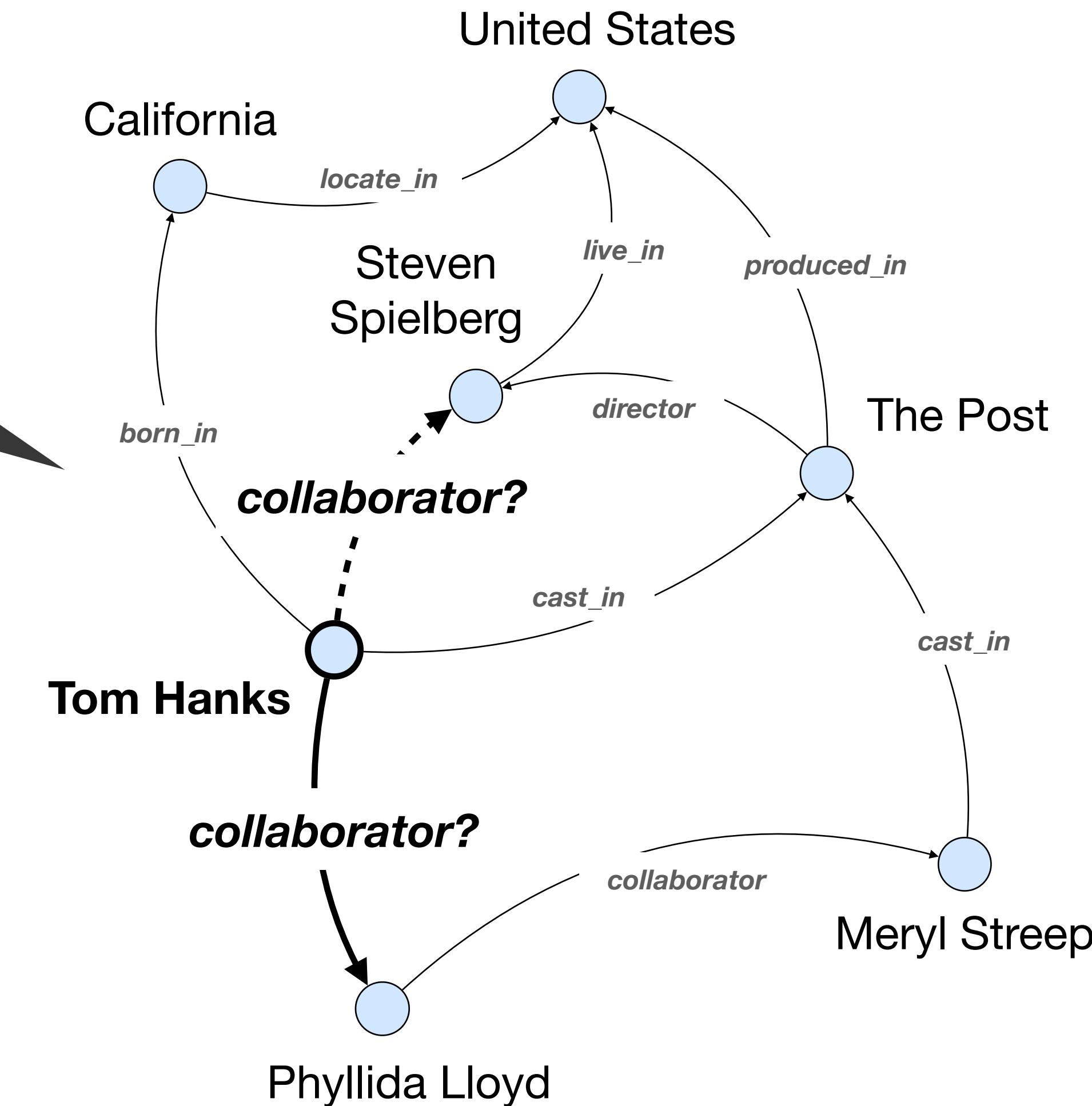


Sequential Multi-Hop Reasoning

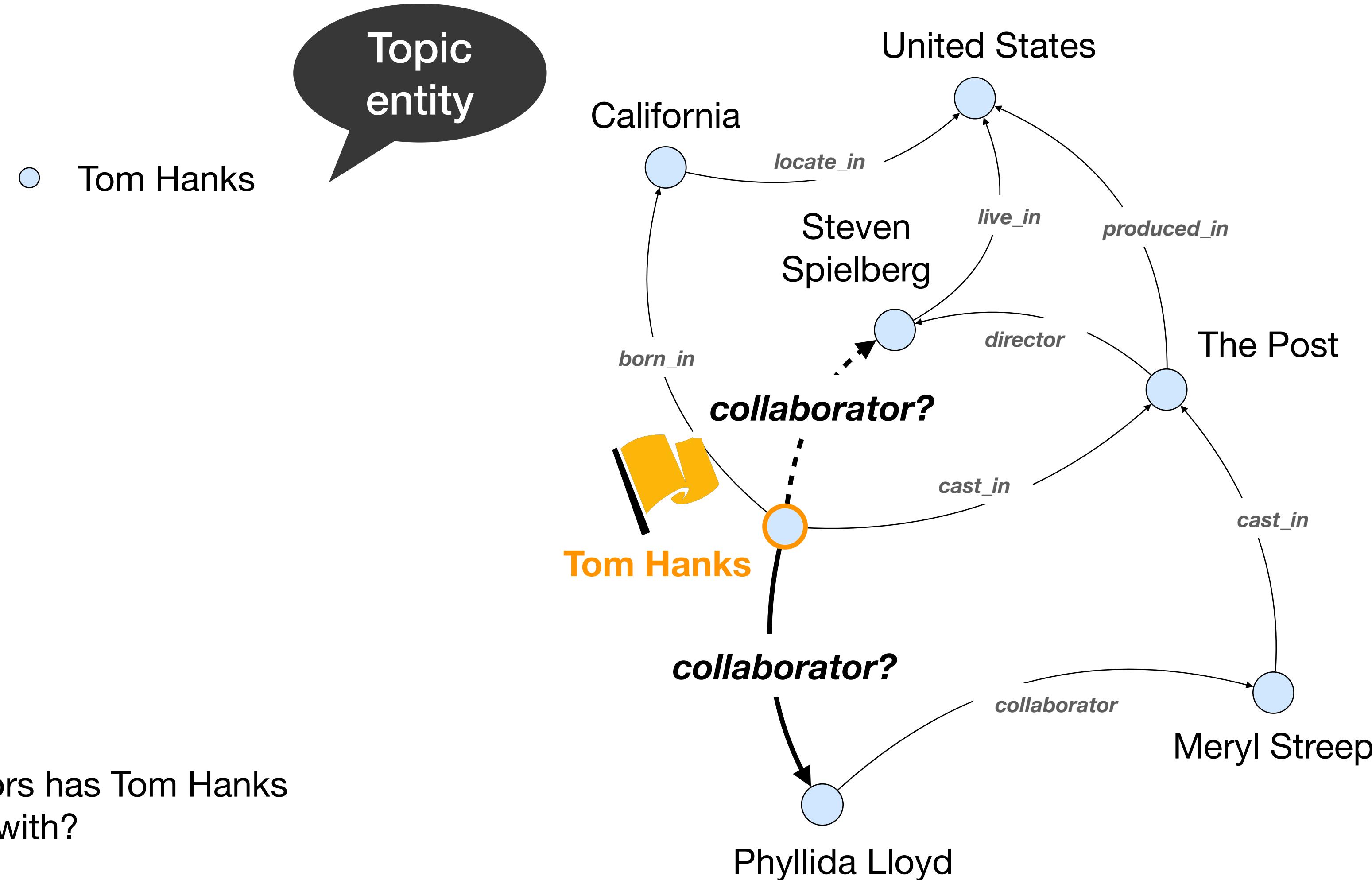
Sequential
decision making



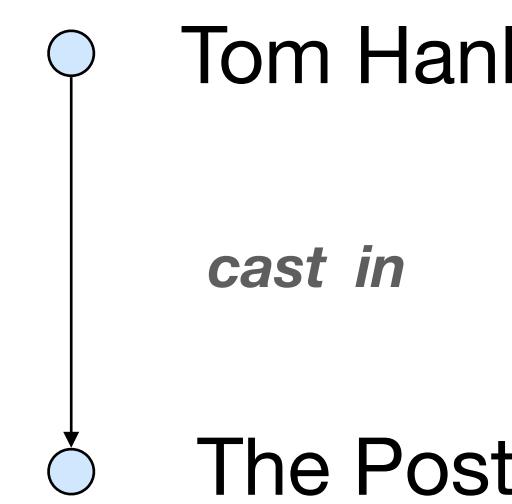
Which directors has Tom Hanks
collaborated with?



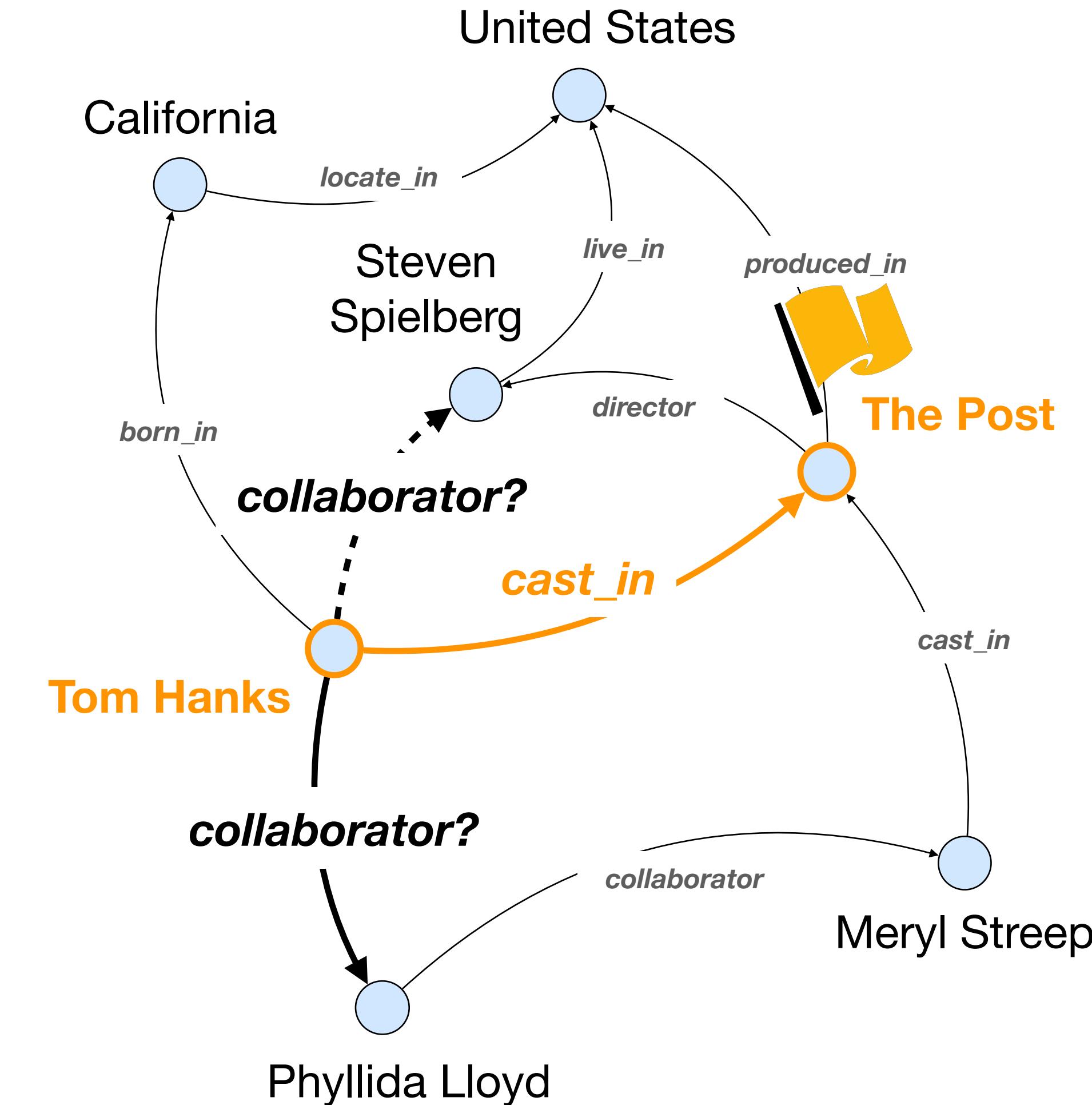
Sequential Multi-Hop Reasoning



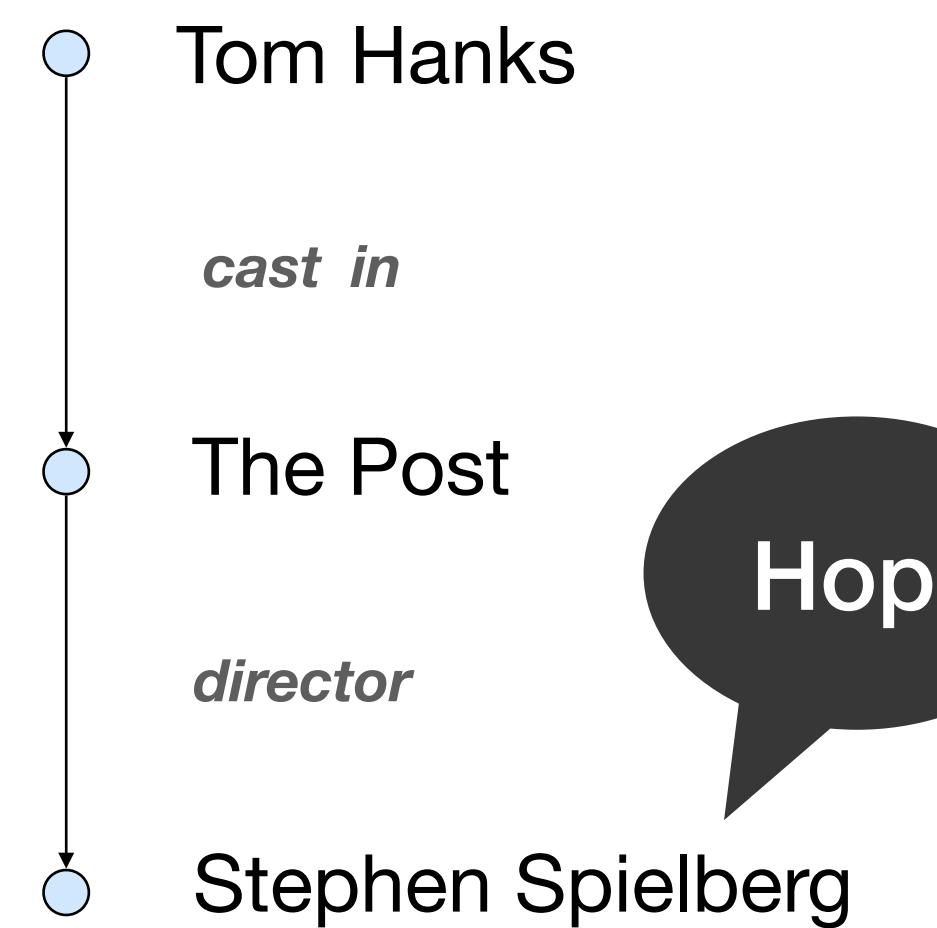
Sequential Multi-Hop Reasoning



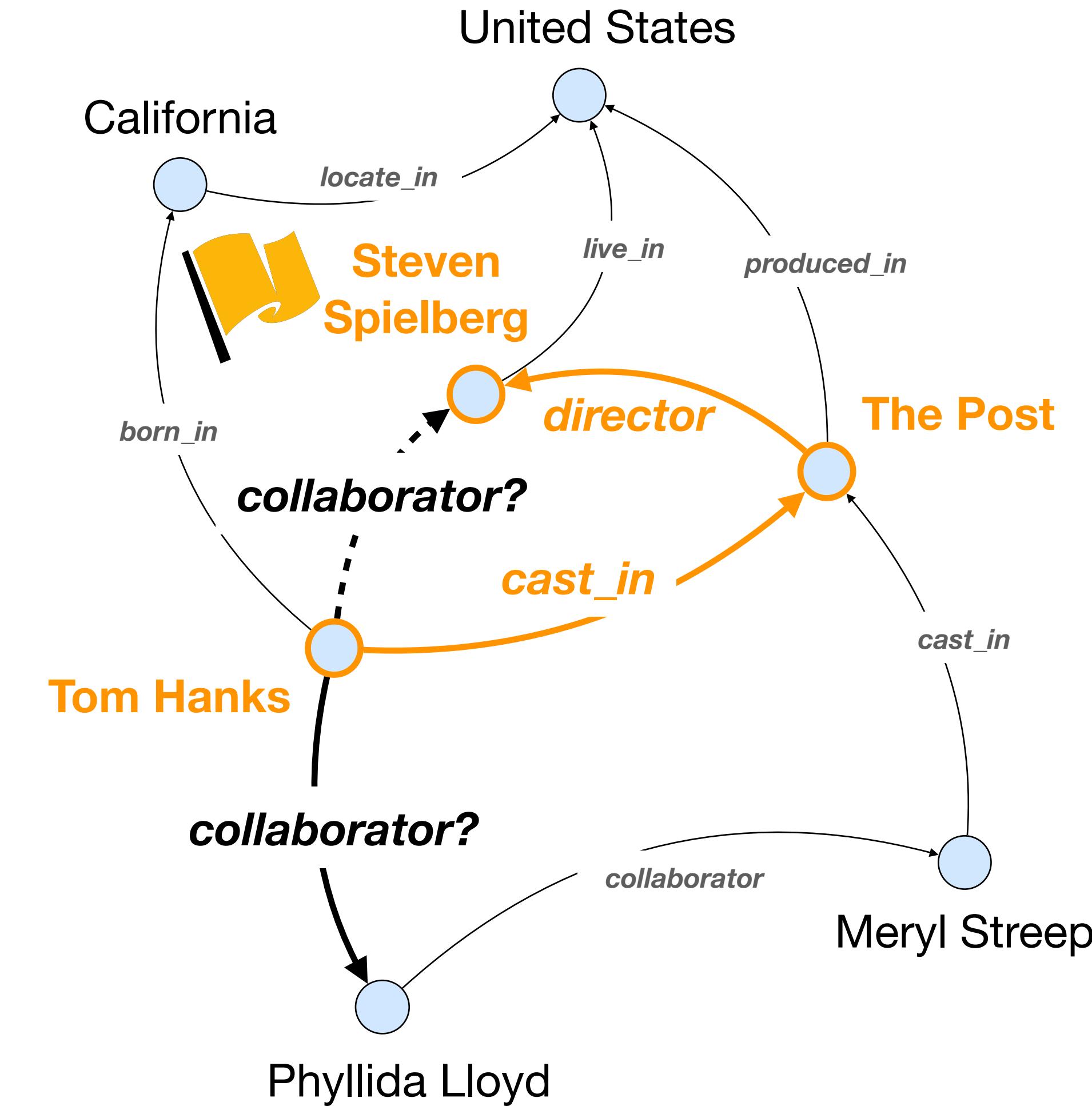
Which directors has Tom Hanks collaborated with?



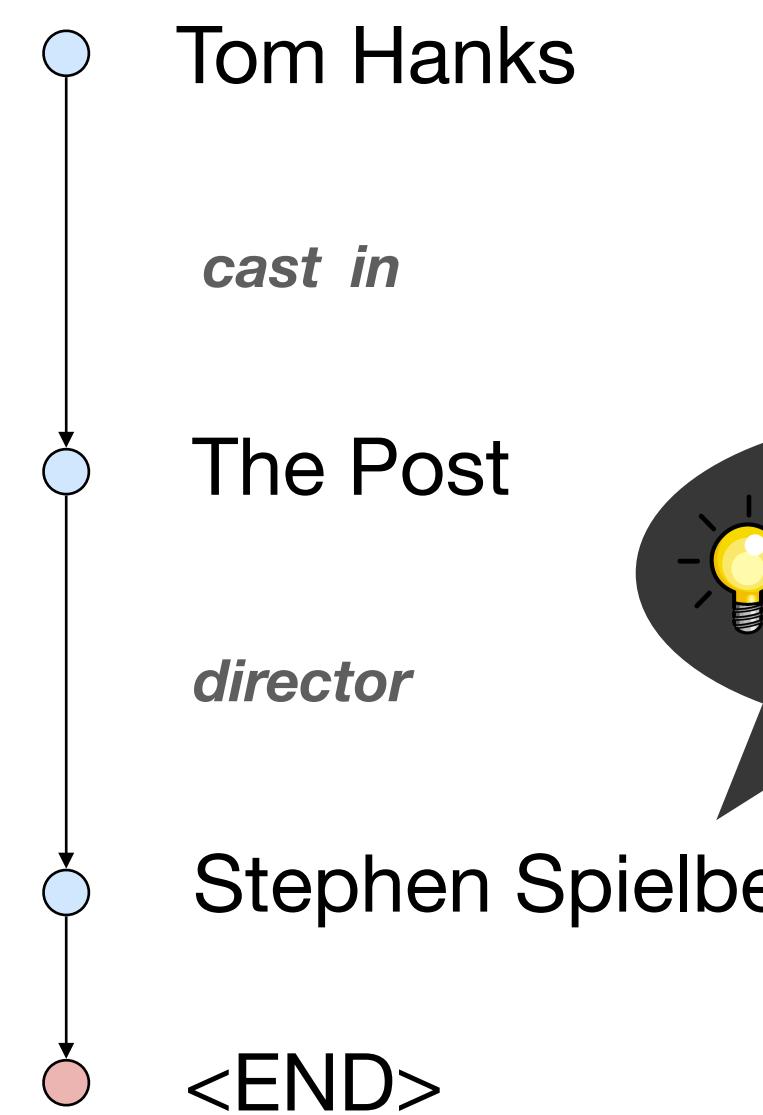
Sequential Multi-Hop Reasoning



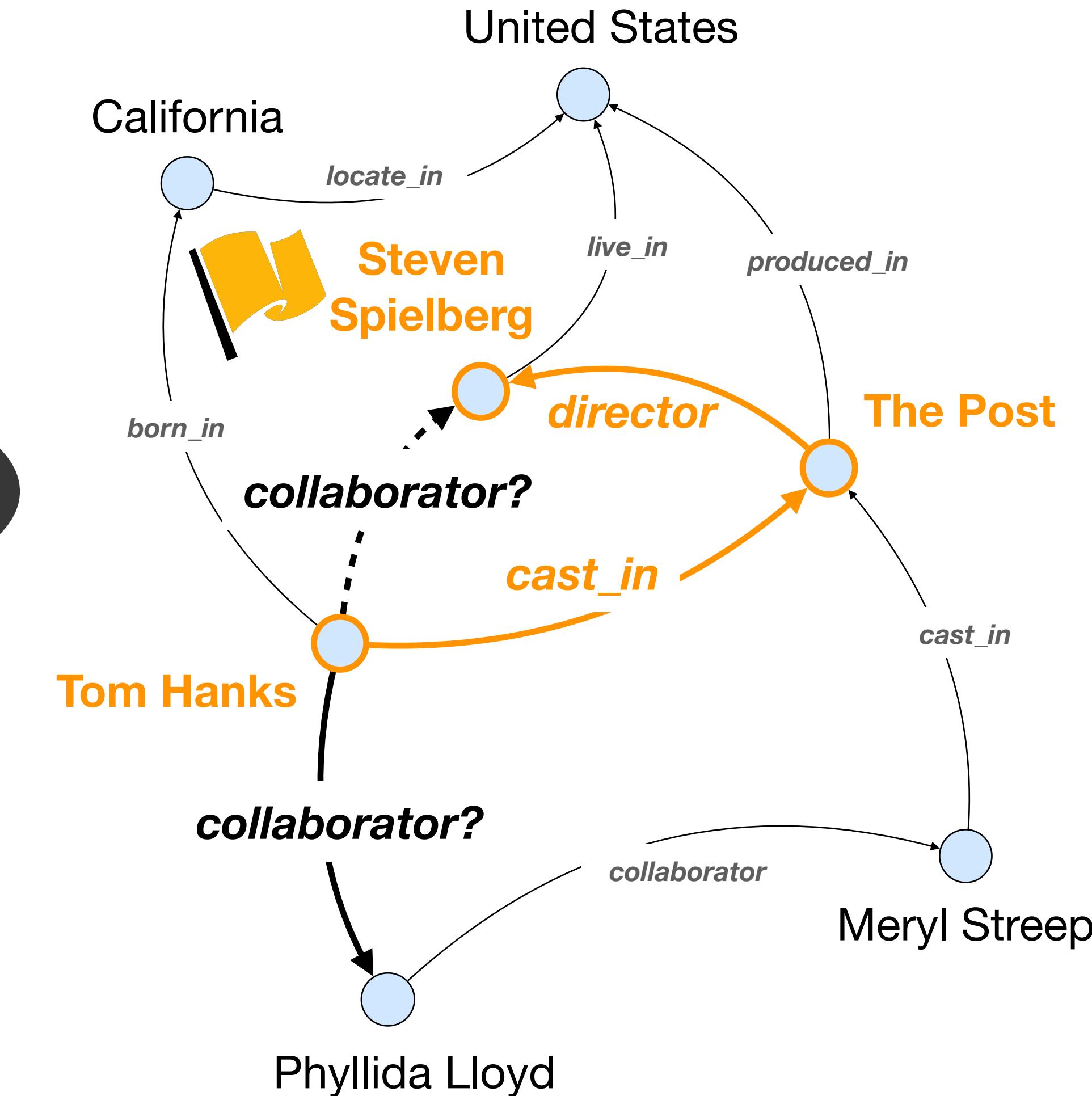
Which directors has Tom Hanks collaborated with?



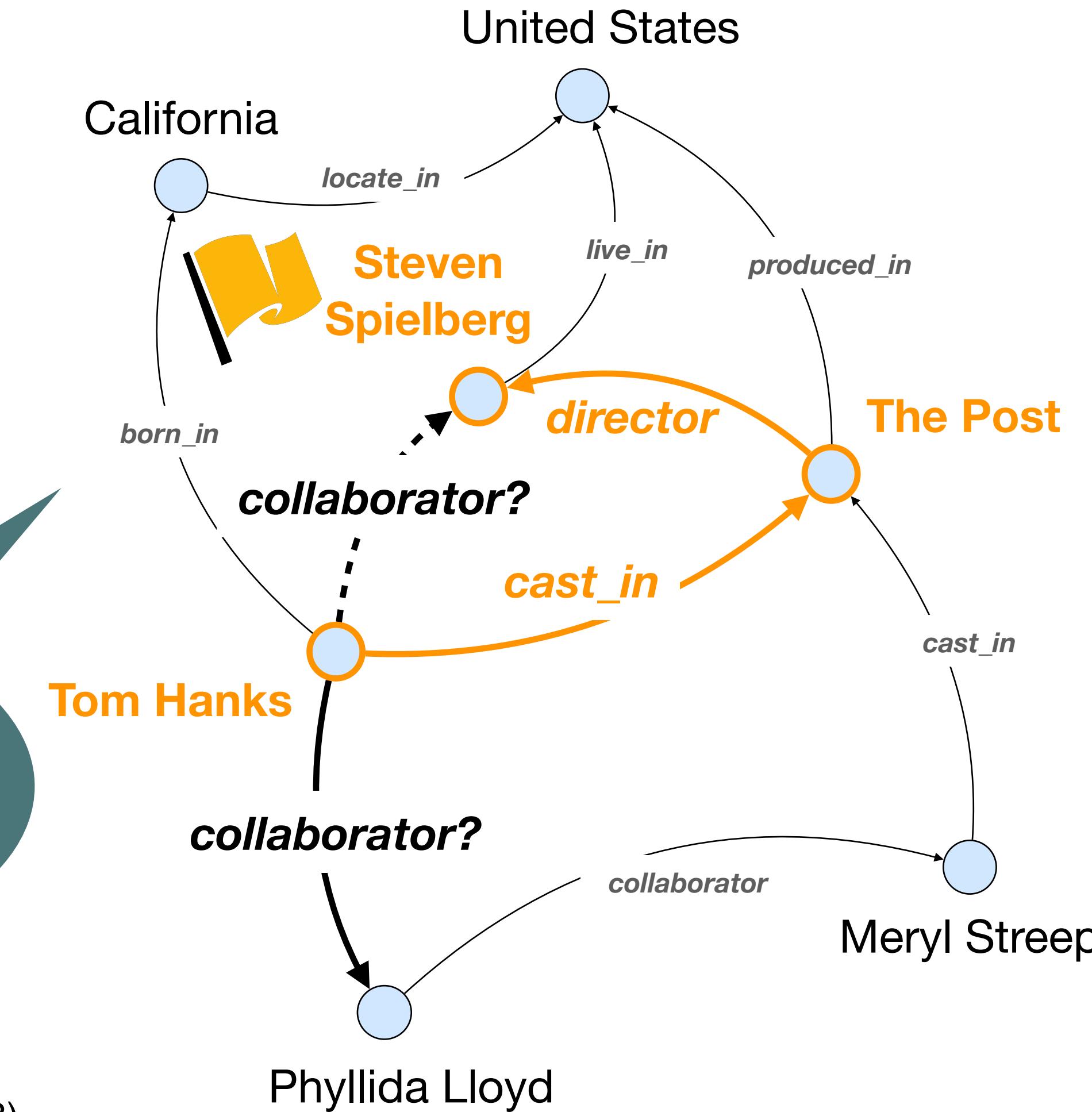
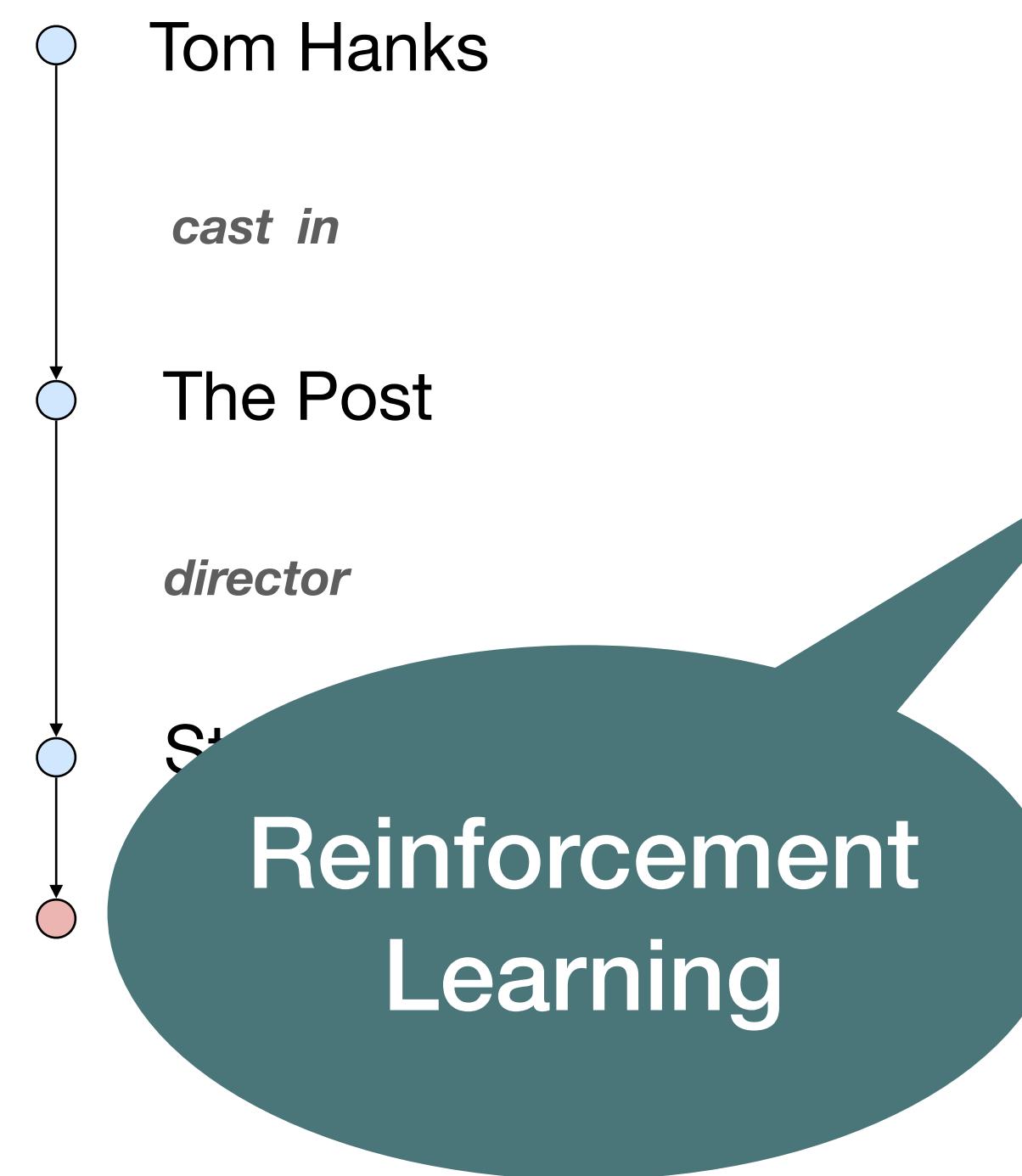
Sequential Multi-Hop Reasoning



Which directors has Tom Hanks collaborated with?



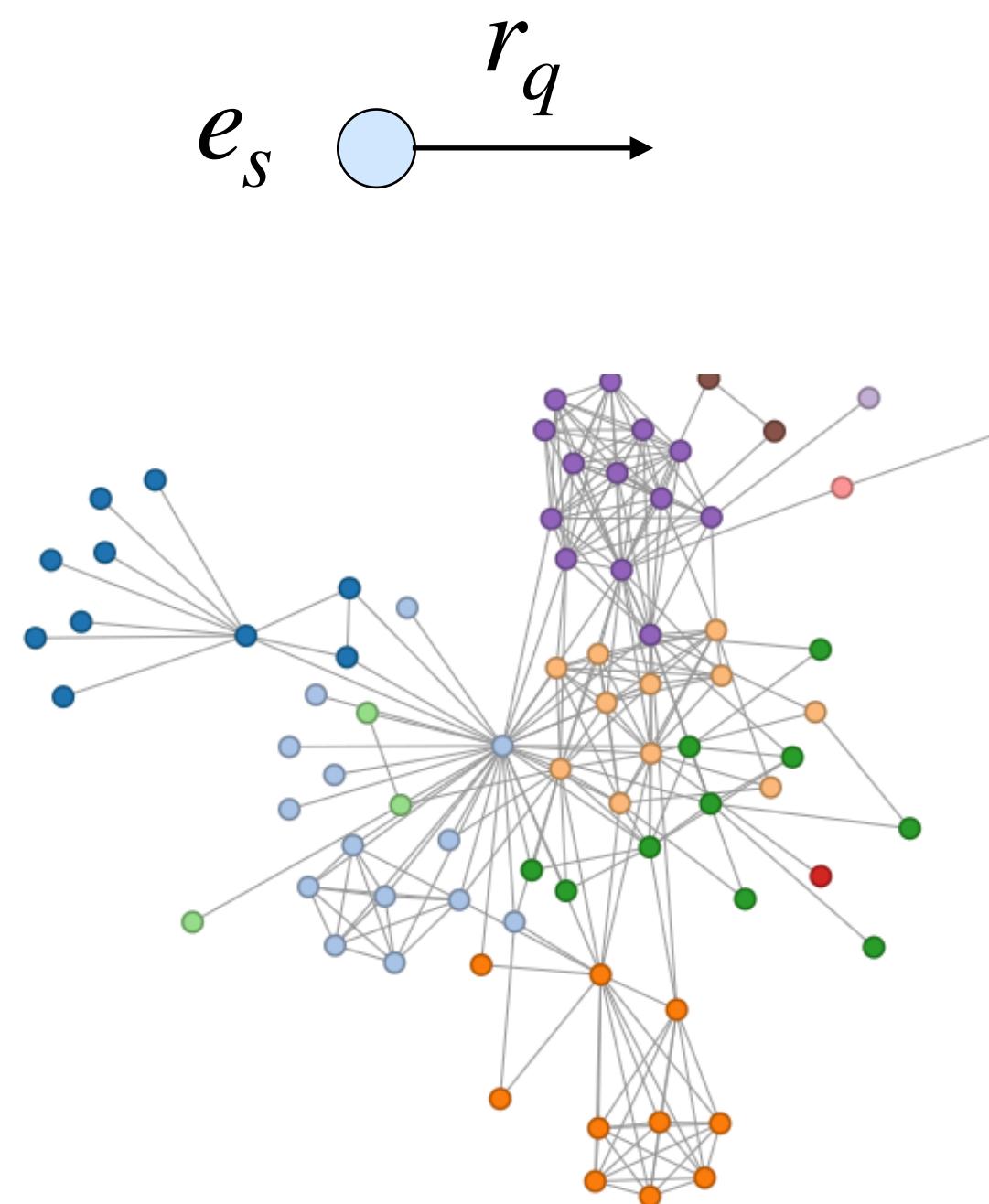
Sequential Multi-Hop Reasoning



MINERVA (Das et al. 2018); MINERVA + Reward Shaping (Lin et al. 2018)

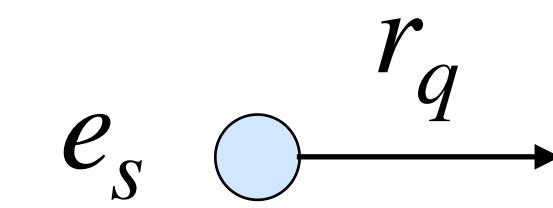
Reinforcement Learning Framework

Environment State Action Transition Reward



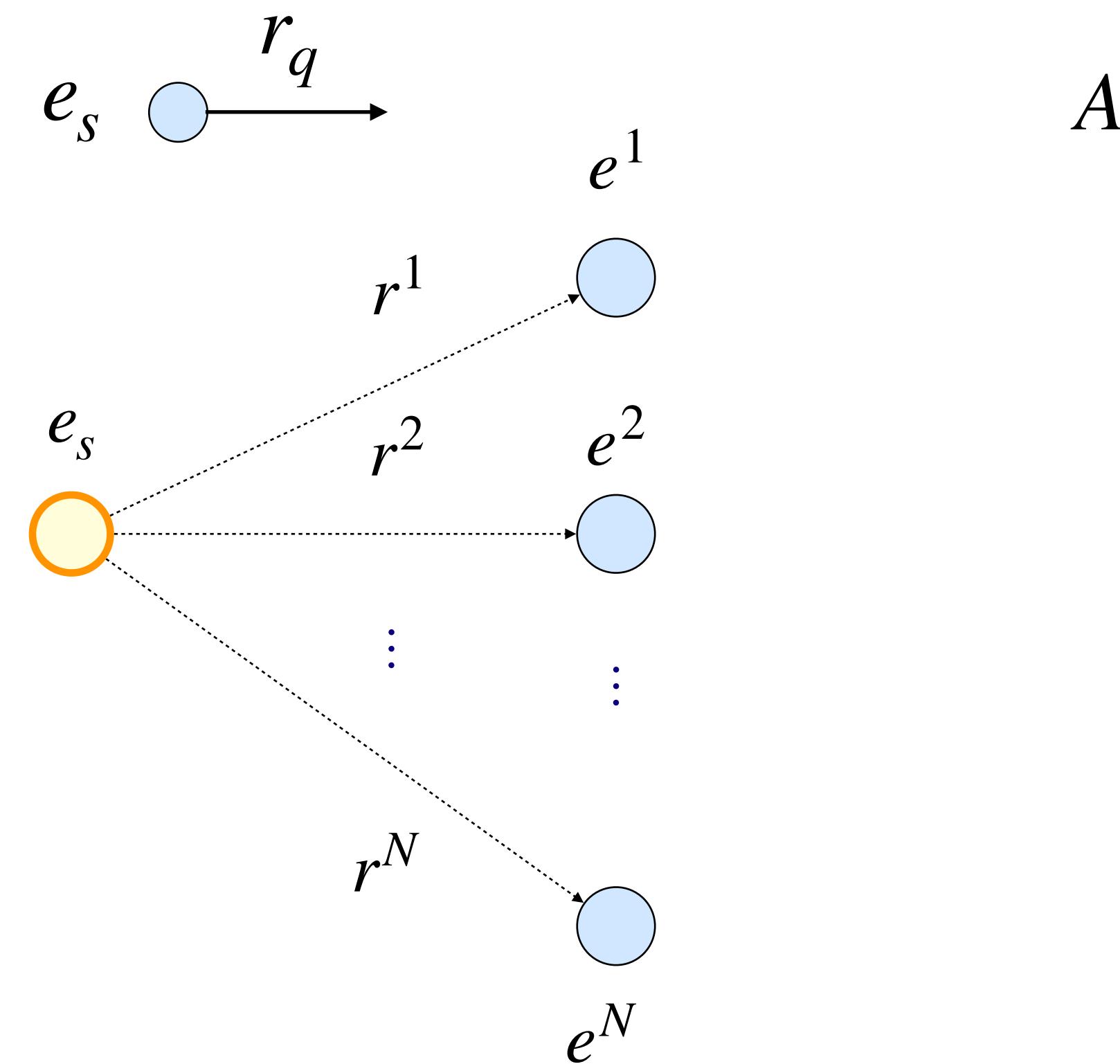
Reinforcement Learning Framework

Environment State Action Transition Reward



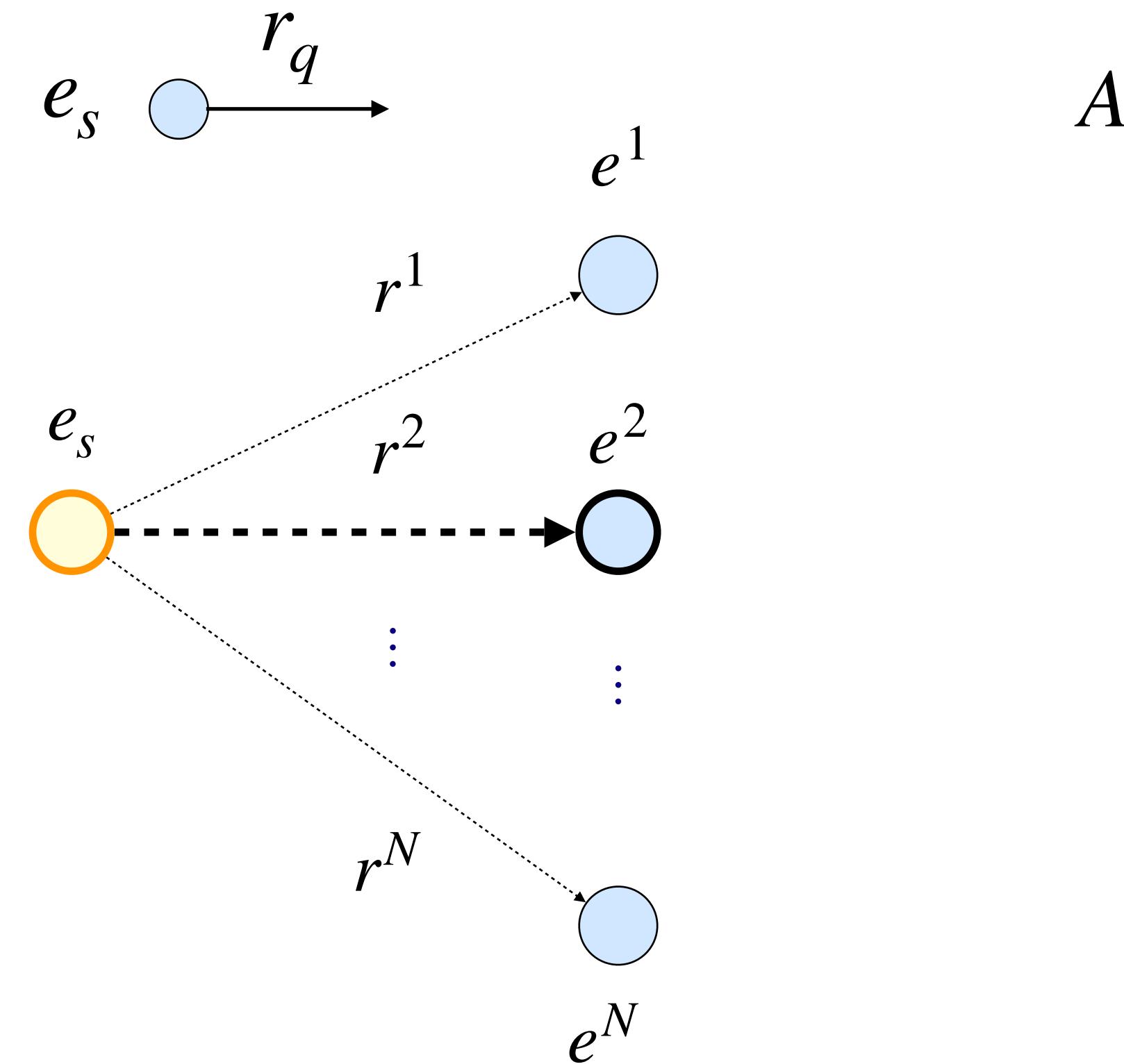
Reinforcement Learning Framework

Environment State **Action** Transition Reward

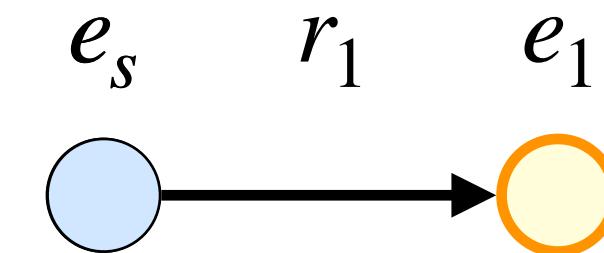
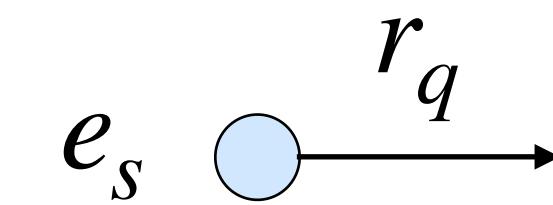


Reinforcement Learning Framework

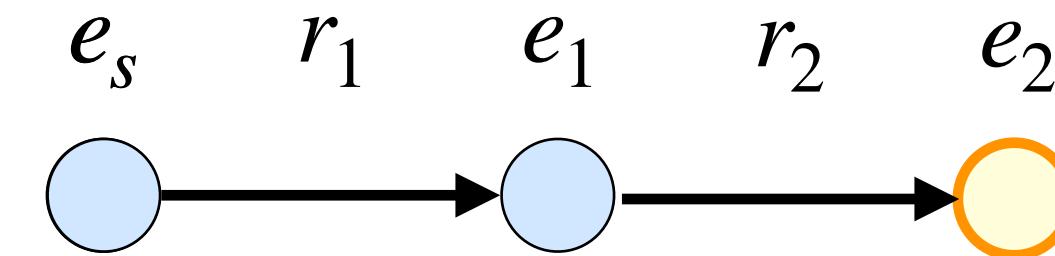
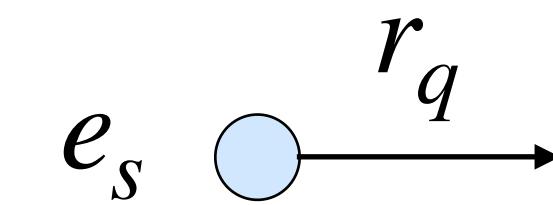
Environment State **Action** Transition Reward



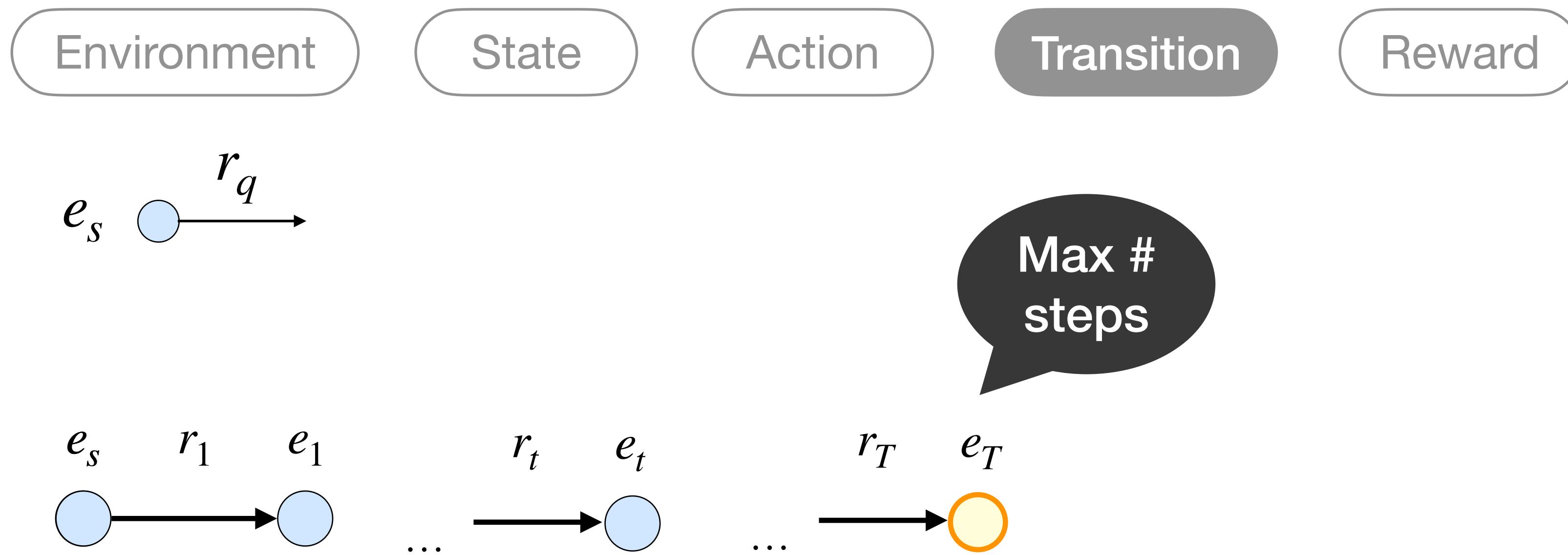
Reinforcement Learning Framework



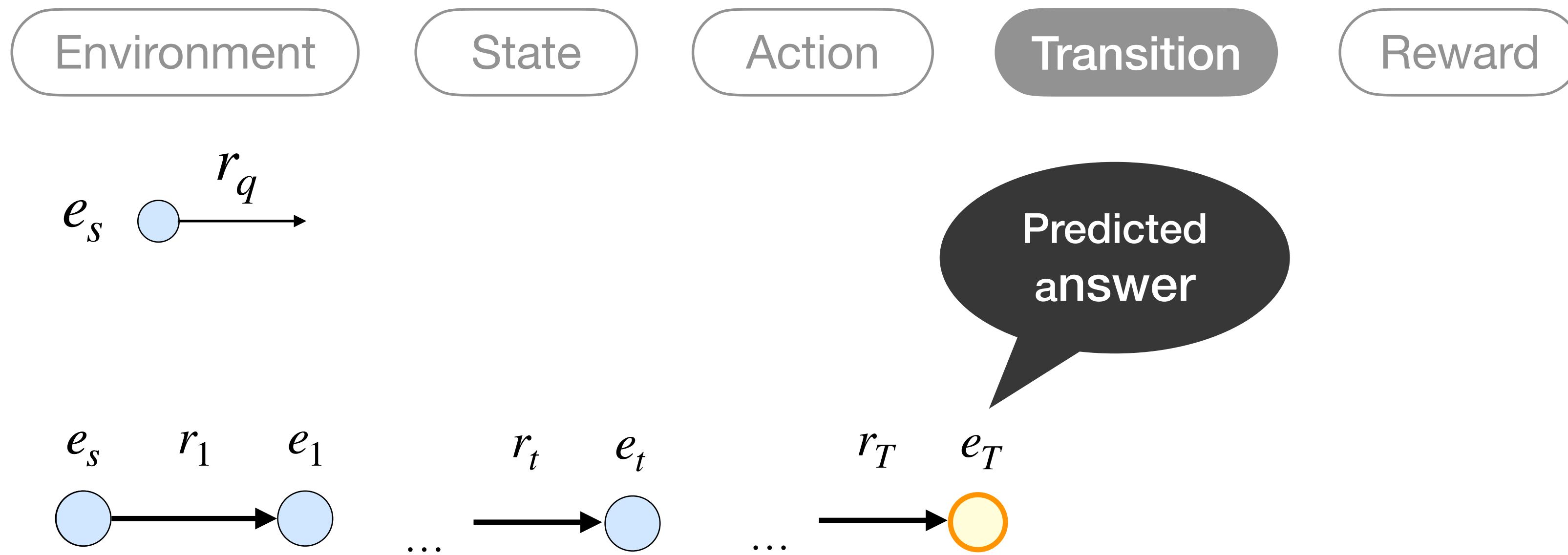
Reinforcement Learning Framework



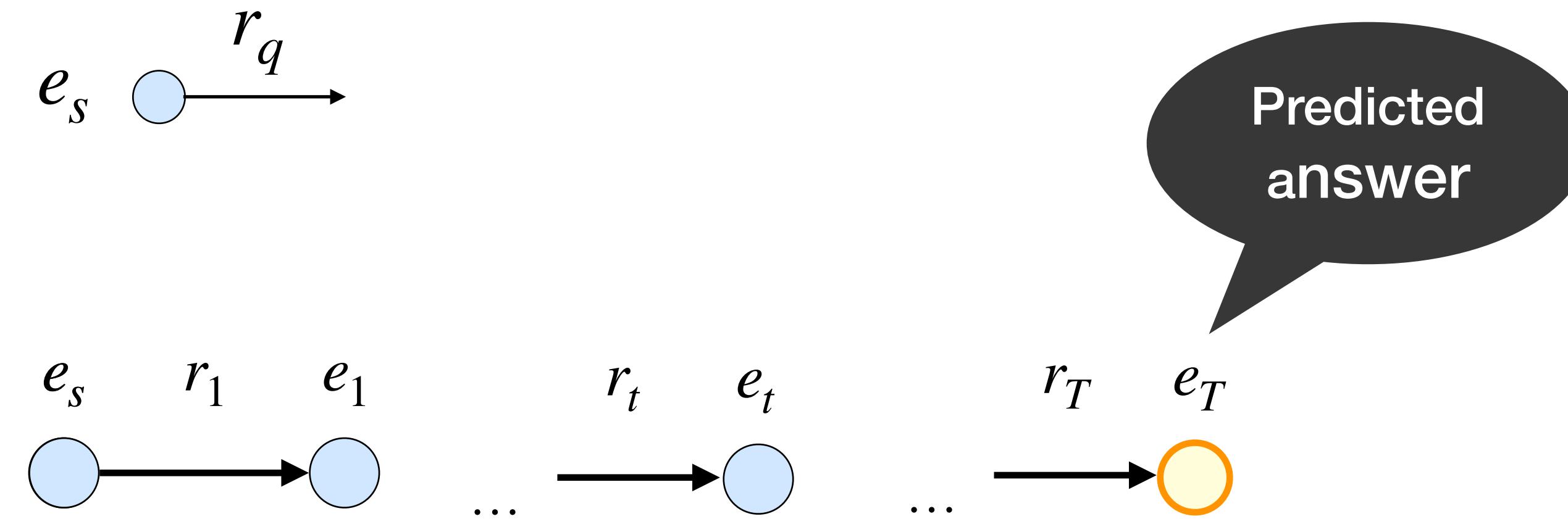
Reinforcement Learning Framework



Reinforcement Learning Framework



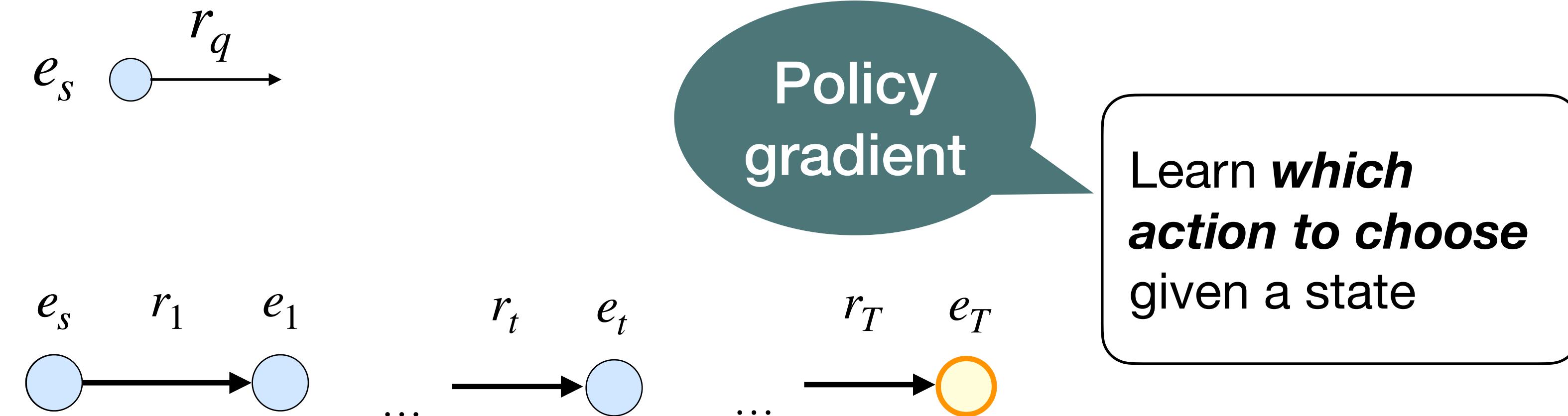
Reinforcement Learning Framework



$$R_b(s_T) = \mathbf{1}\{(e_s, r_q, e_T) \in G\}$$

Reinforcement Learning Framework

Environment State Action Transition Reward



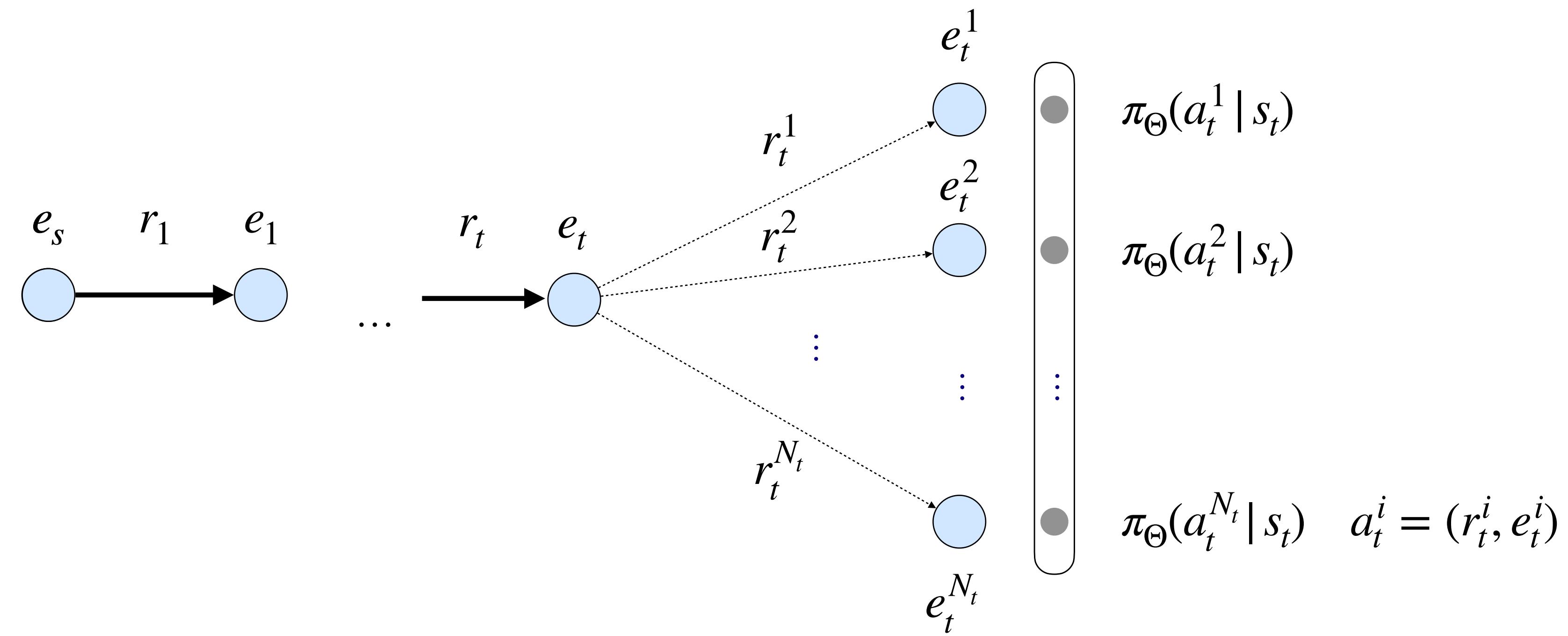
$$R_b(s_T) = \mathbf{1}\{(e_s, r_q, e_T) \in G\}$$

Policy Gradient

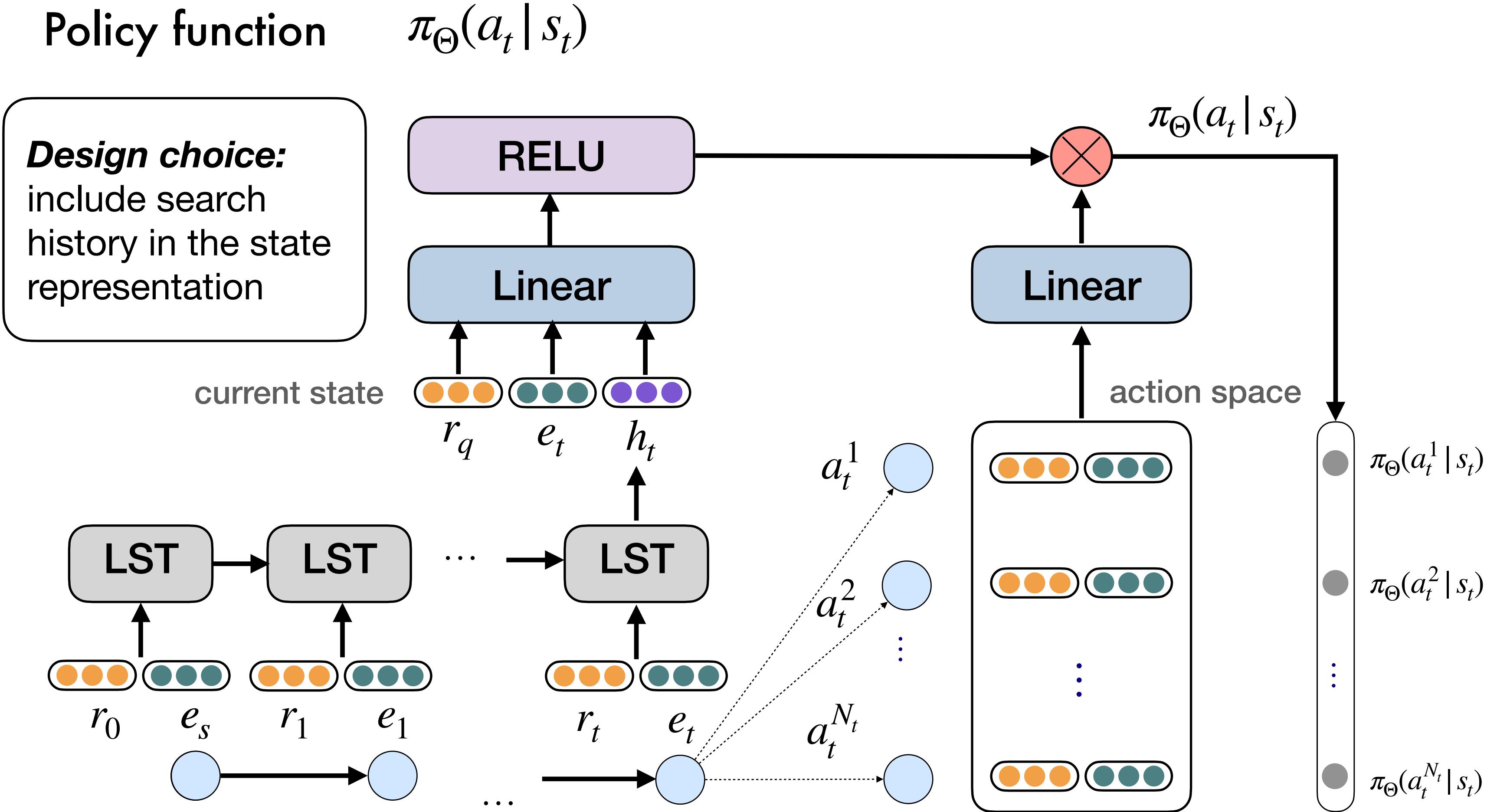
Policy function

$$\pi_{\Theta}(a_t | s_t)$$

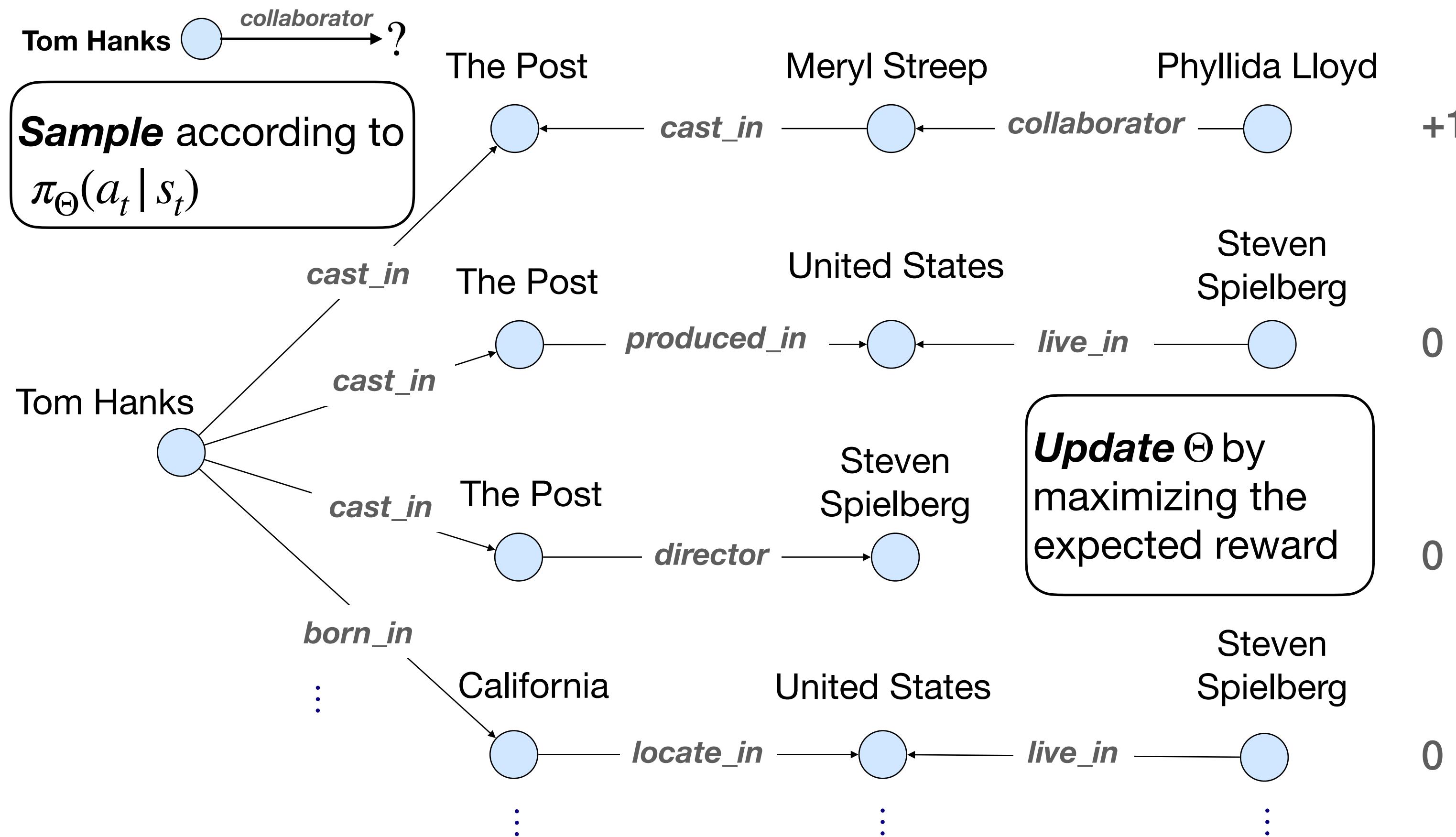
Probability of choosing
an action given the
current state



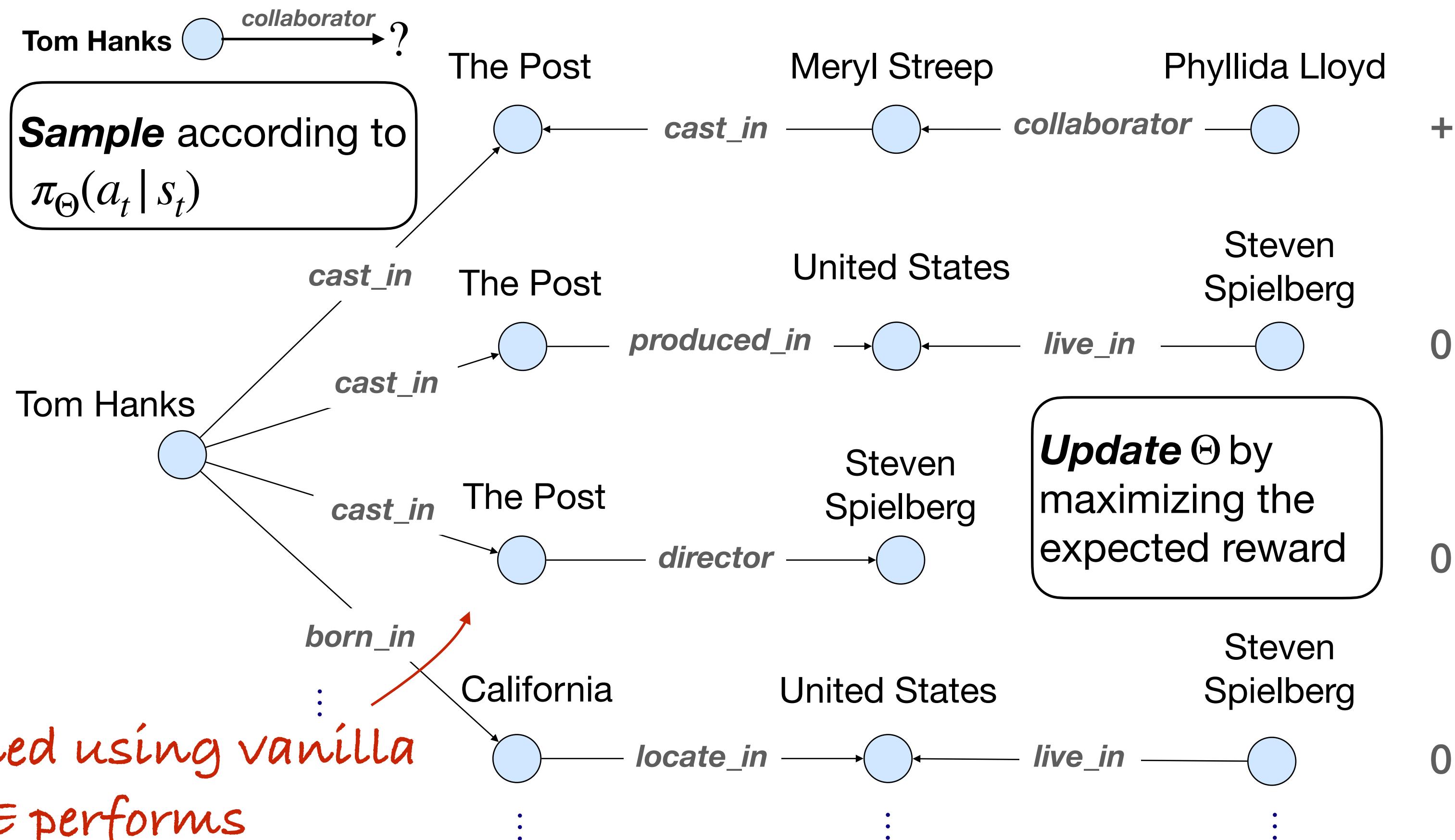
Policy Gradient



REINFORCE Training

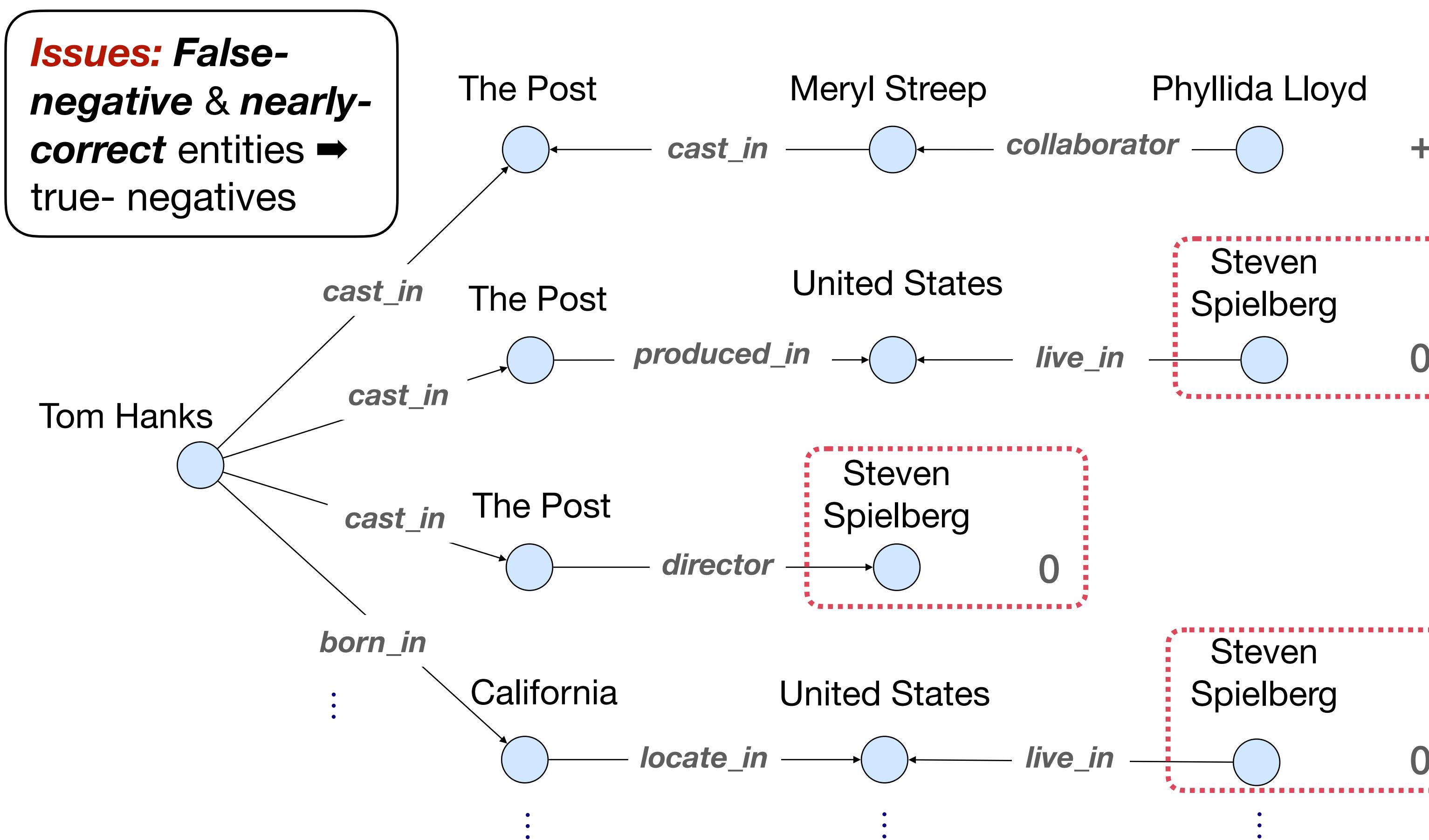


REINFORCE Training



Models trained using vanilla
REINFORCE performs
significantly worse compared
to KG embedding baselines

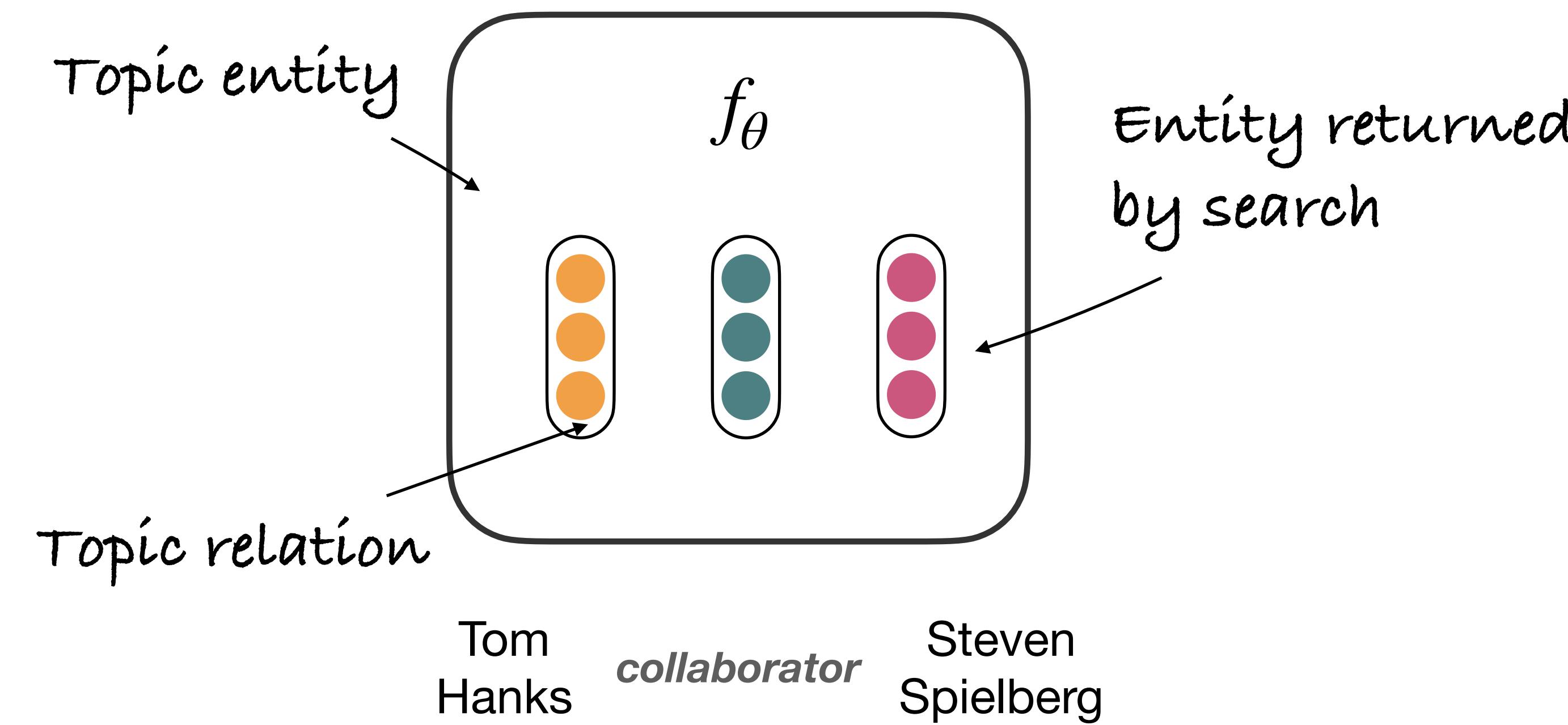
Sparse Reward



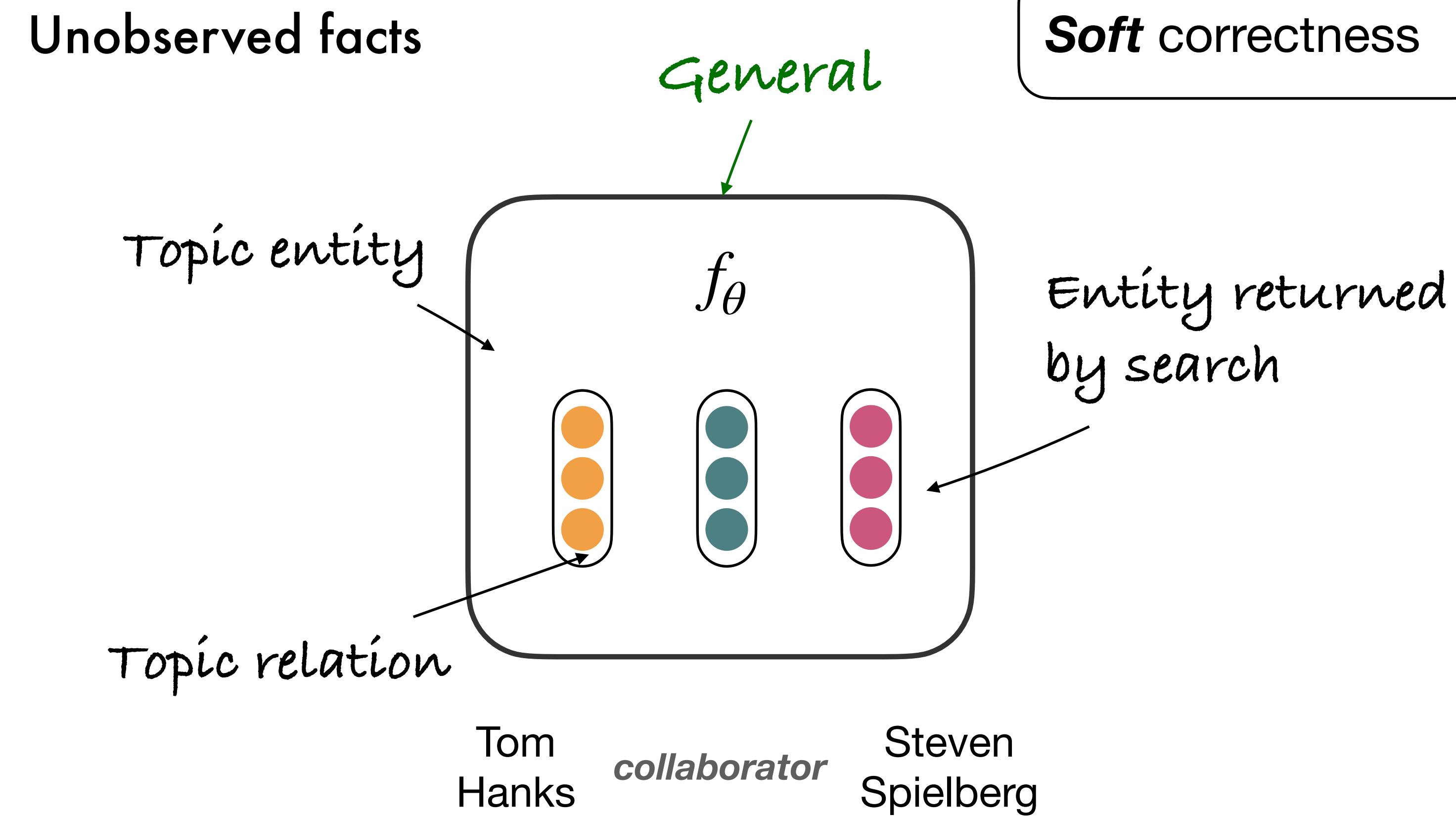
Reward Shaping

Unobserved facts

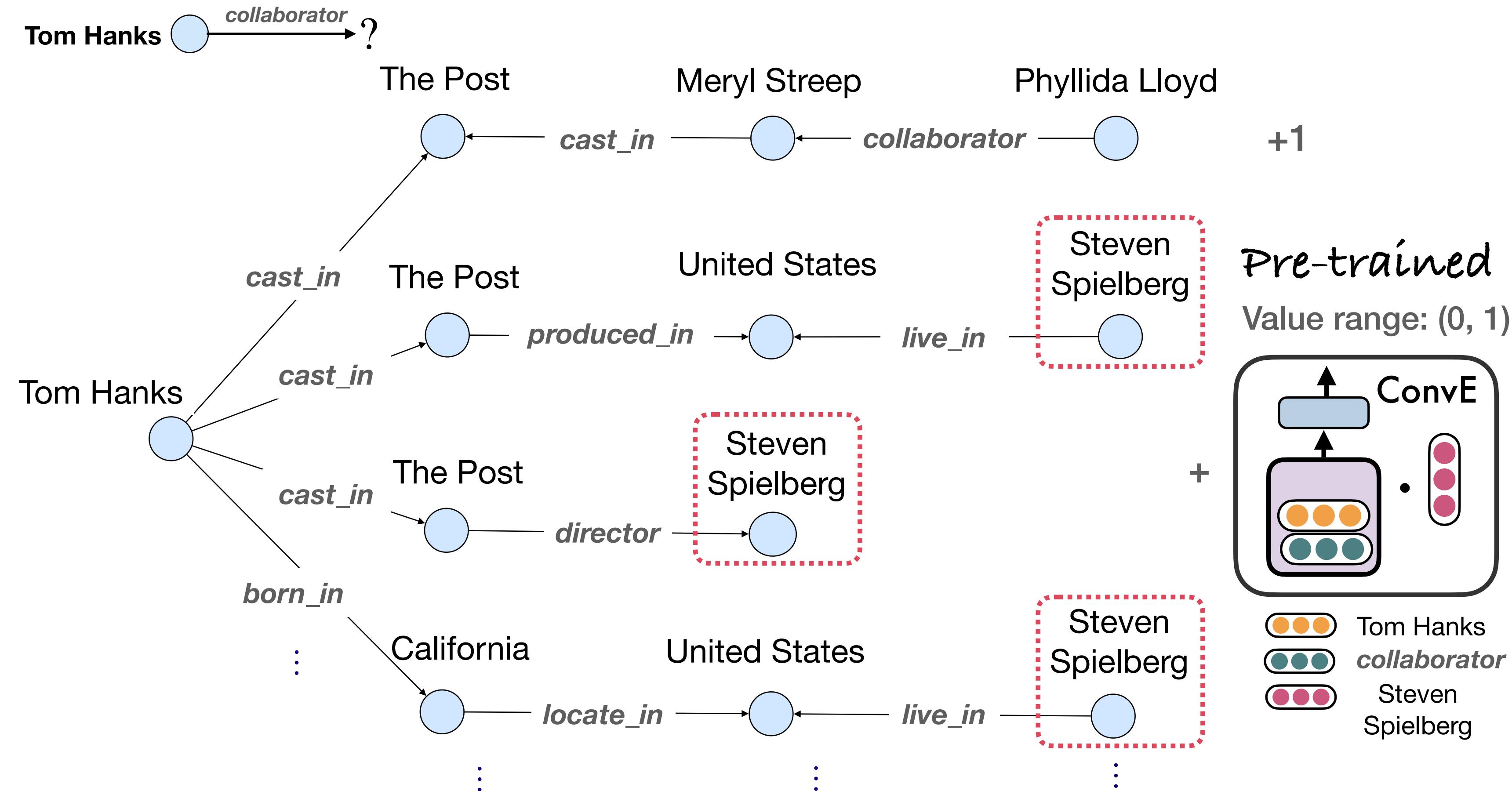
Soft correctness



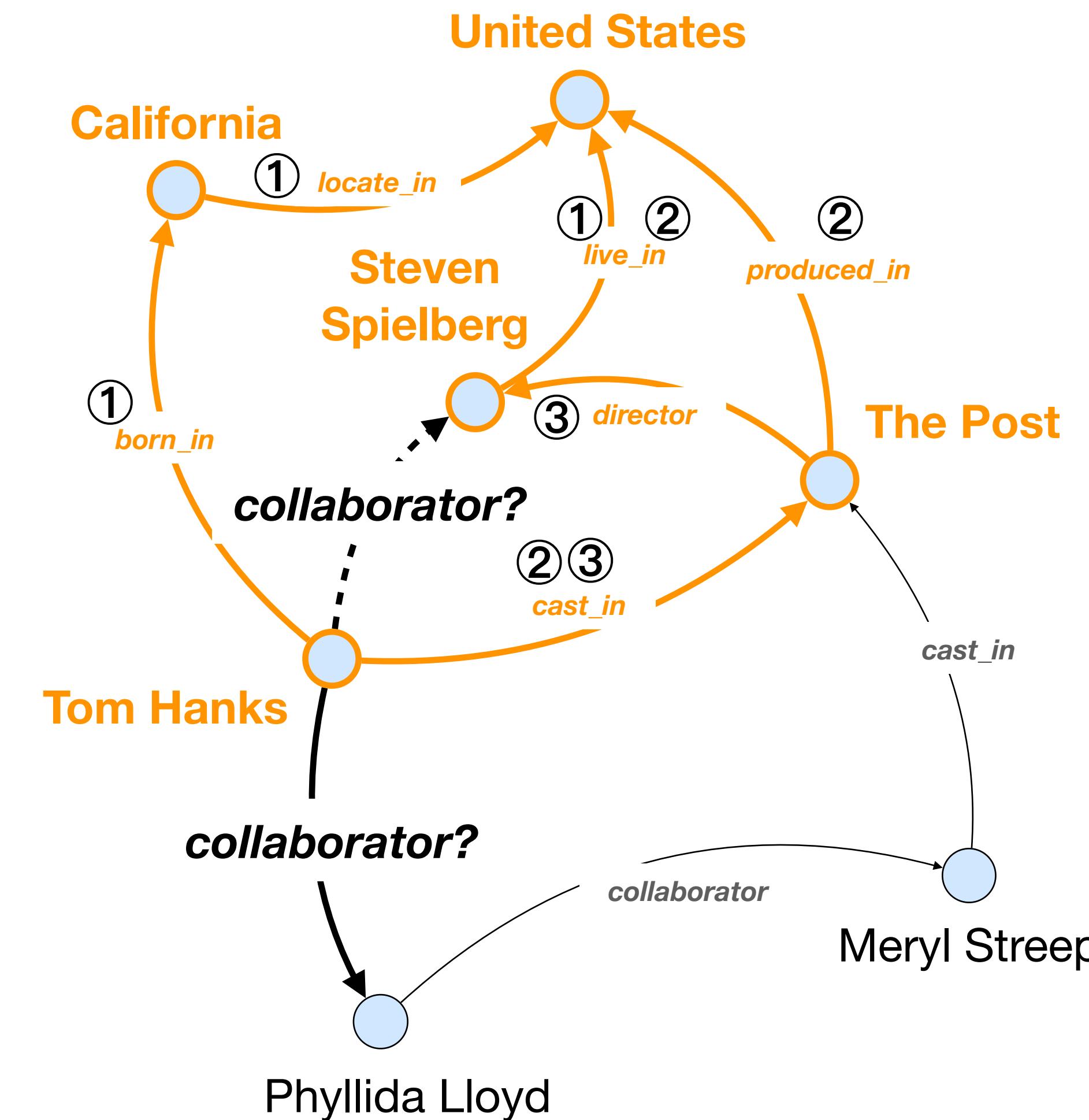
Reward Shaping



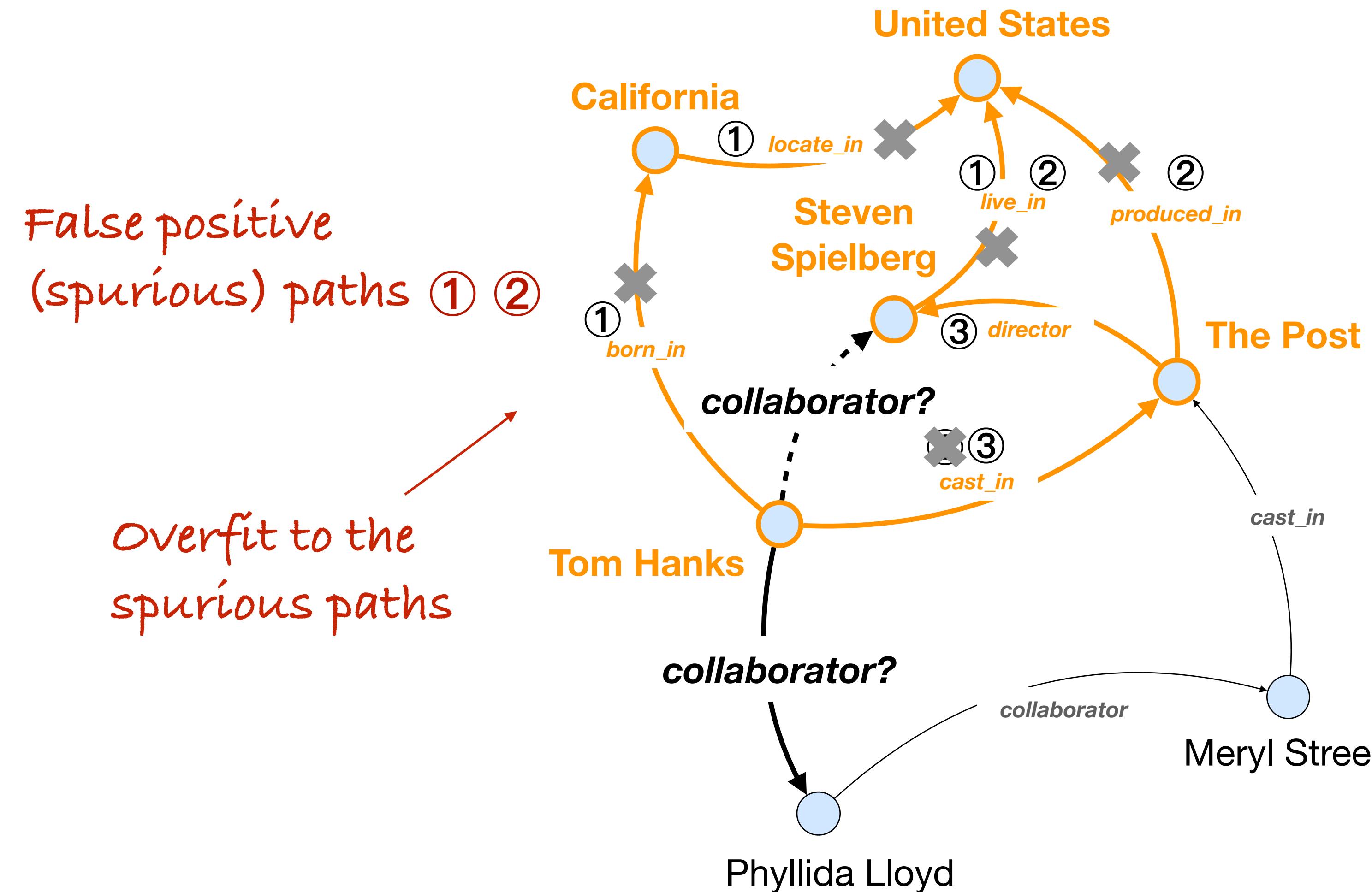
Reward Shaping



Spurious Path



Spurious Path



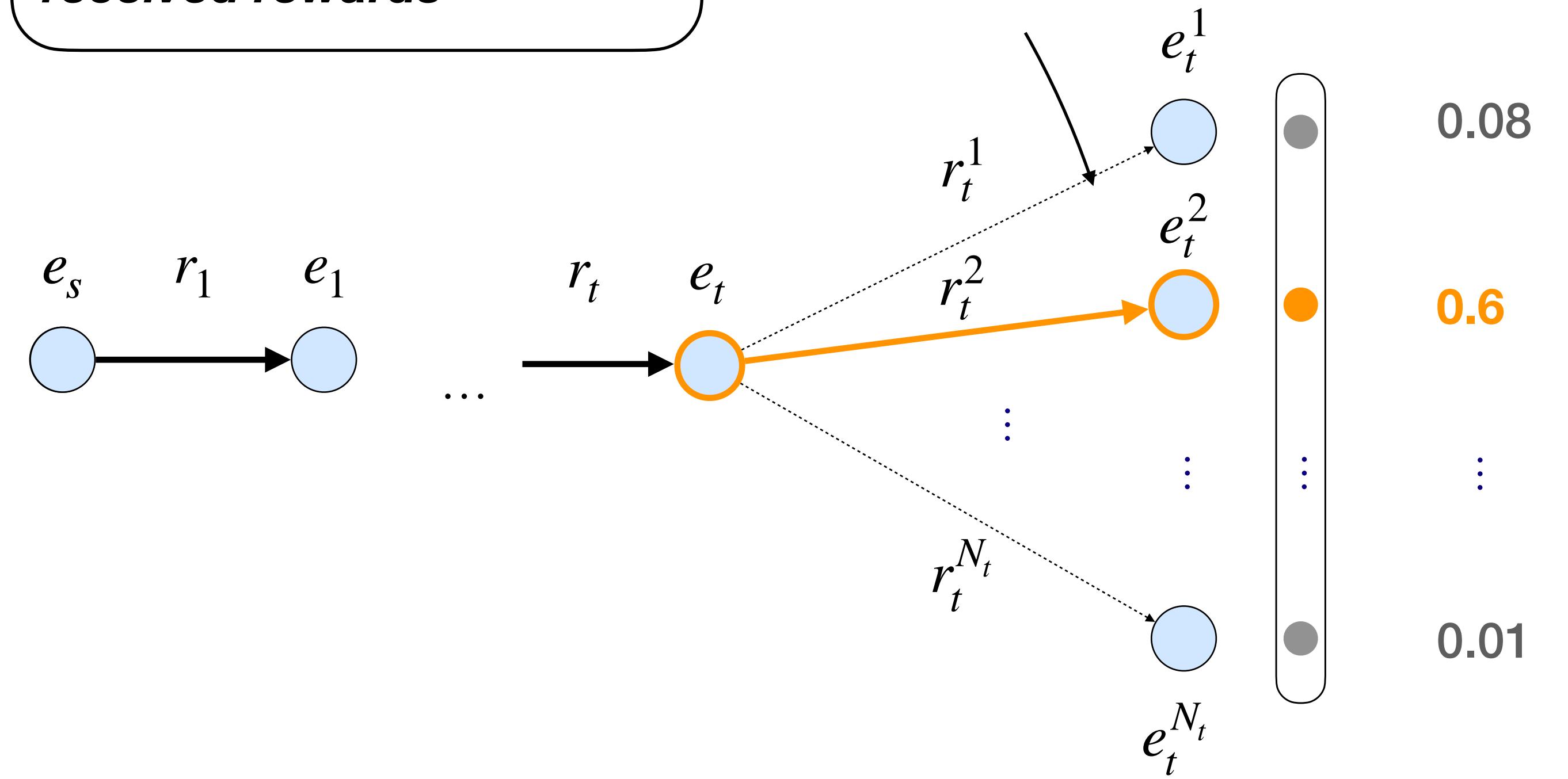
Action Dropout

Intuition: ***avoid sticking to***
past actions that ***had***
received rewards

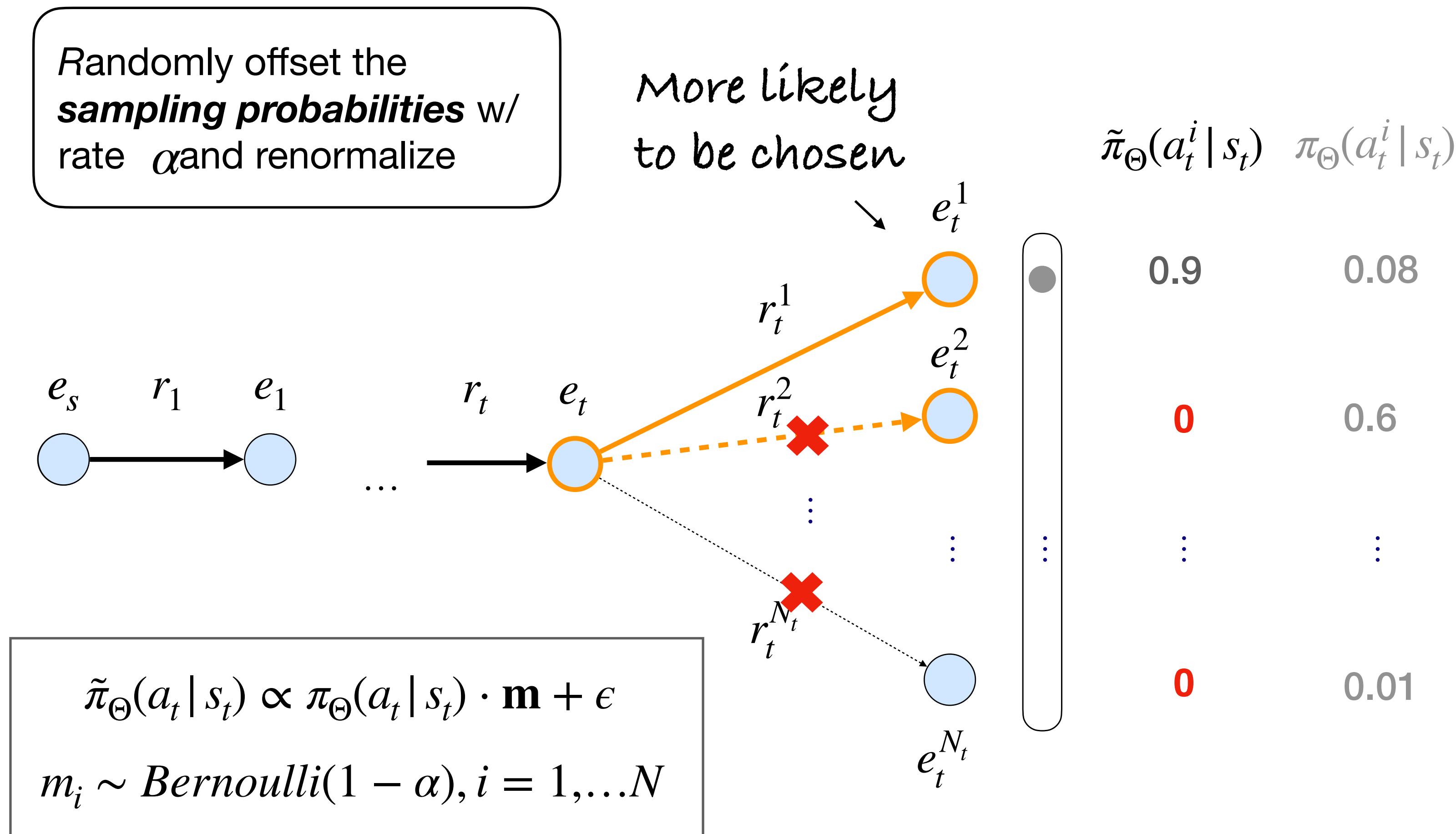
Action Dropout

Intuition: **avoid sticking to**
past actions that **had received rewards**

More likely
to be chosen



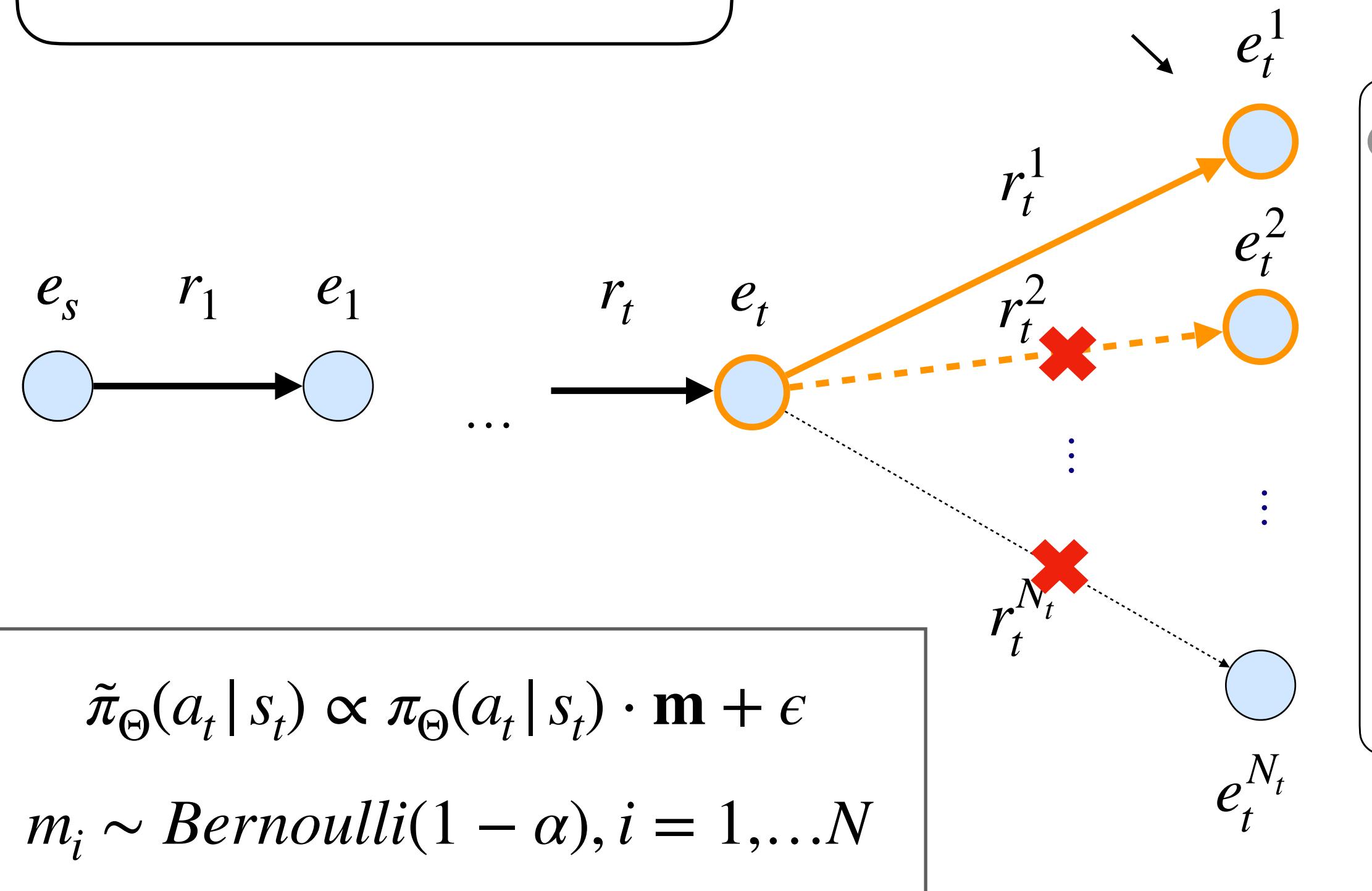
Action Dropout



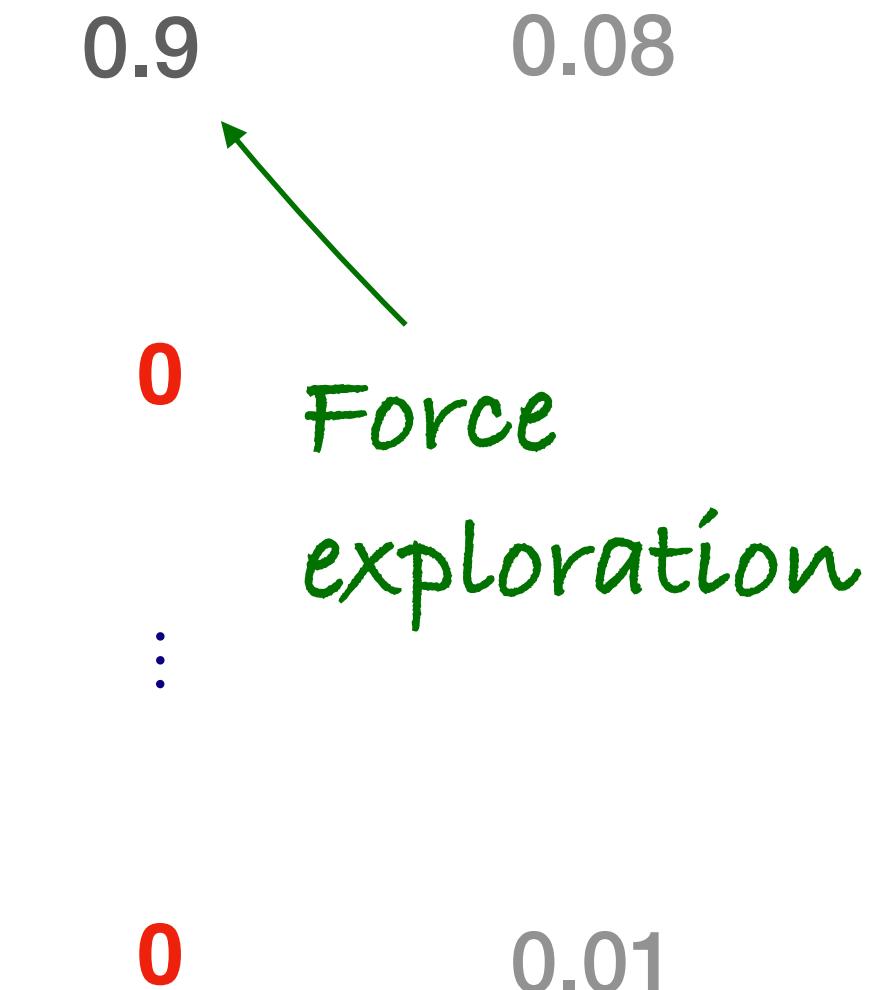
Action Dropout

Randomly offset the **sampling probabilities** w/
rate α and renormalize

More likely
to be chosen



$$\tilde{\pi}_{\Theta}(a_t^i | s_t) \quad \pi_{\Theta}(a_t^i | s_t)$$



$$\tilde{\pi}_{\Theta}(a_t | s_t) \propto \pi_{\Theta}(a_t | s_t) \cdot \mathbf{m} + \epsilon$$

$$m_i \sim \text{Bernoulli}(1 - \alpha), i = 1, \dots, N$$

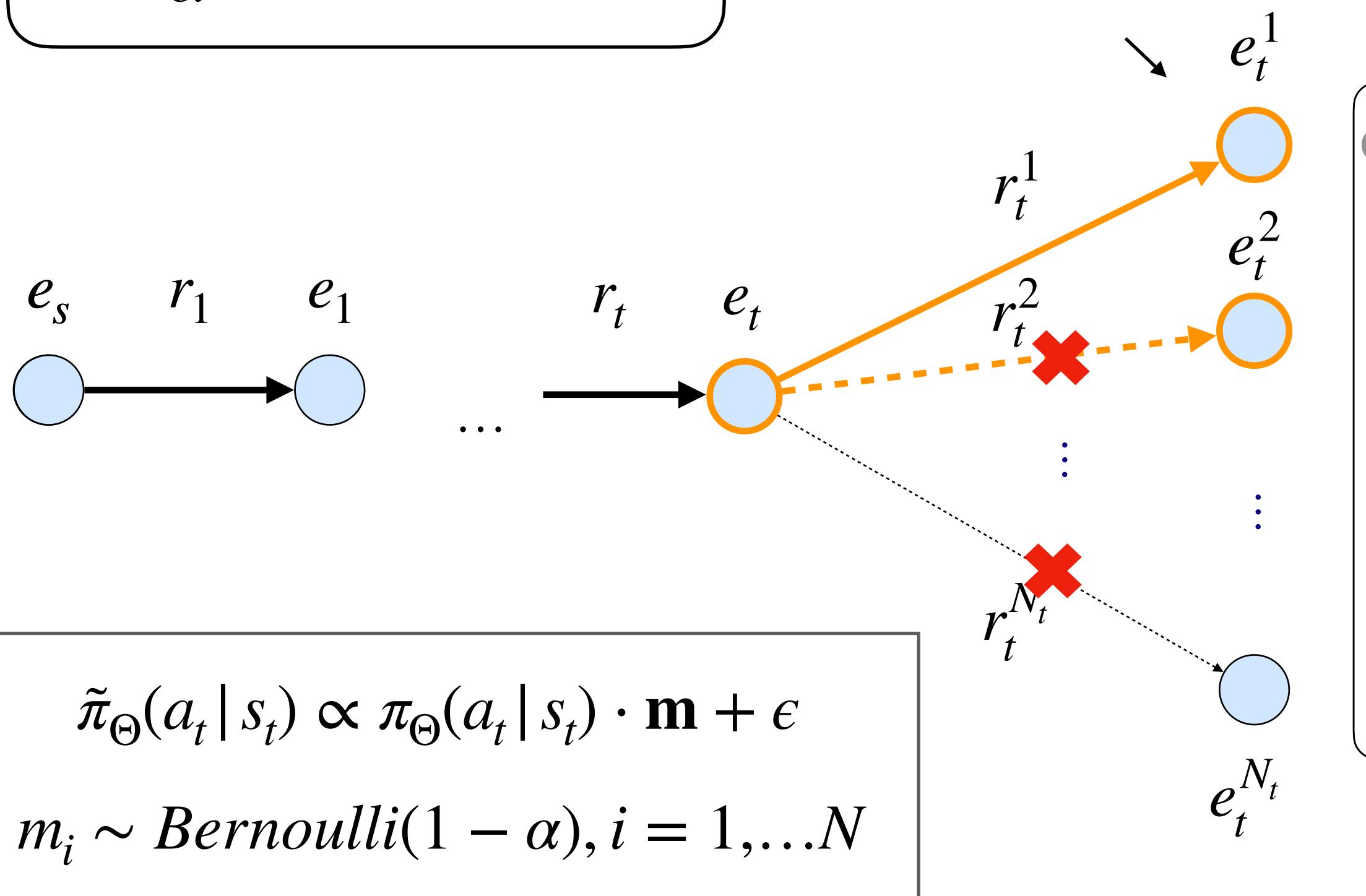
Action Dropout

Randomly offset the **sampling probabilities** w/
rate α and renormalize

More likely
to be chosen

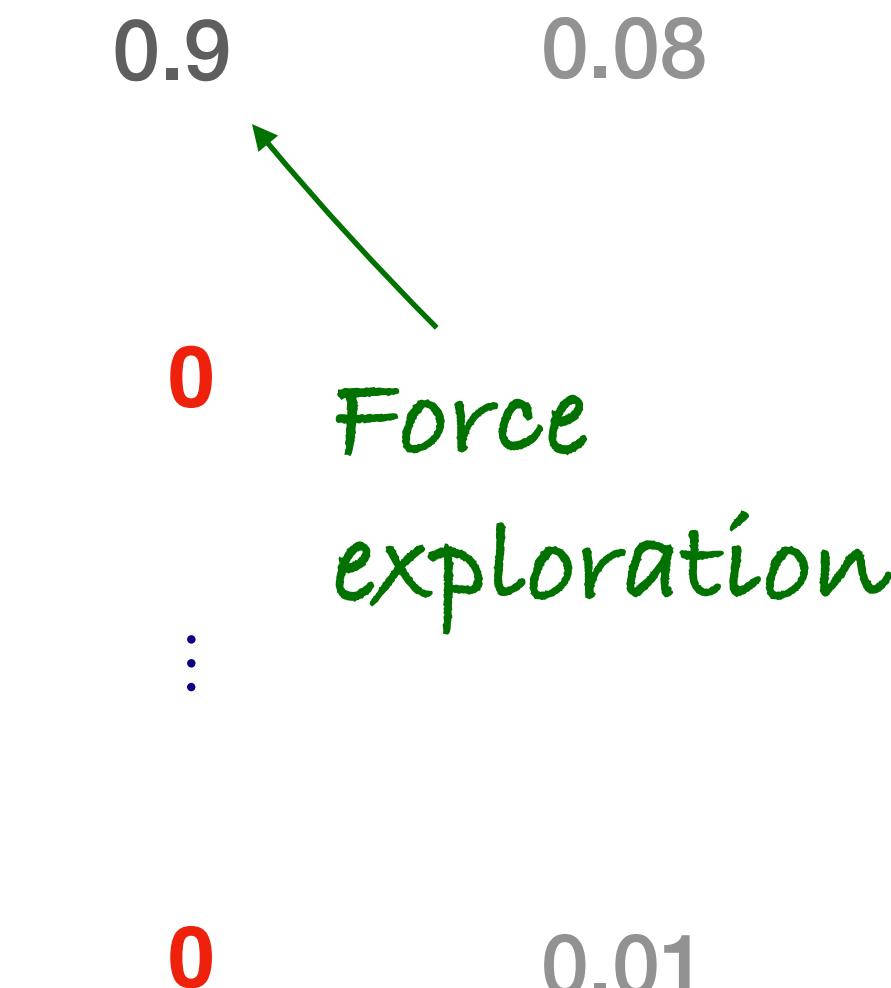
Up to $\times 8$ # path
traversed

$$\tilde{\pi}_{\Theta}(a_t^i | s_t) \quad \pi_{\Theta}(a_t^i | s_t)$$



$$\tilde{\pi}_{\Theta}(a_t | s_t) \propto \pi_{\Theta}(a_t | s_t) \cdot \mathbf{m} + \epsilon$$

$$m_i \sim \text{Bernoulli}(1 - \alpha), i = 1, \dots, N$$



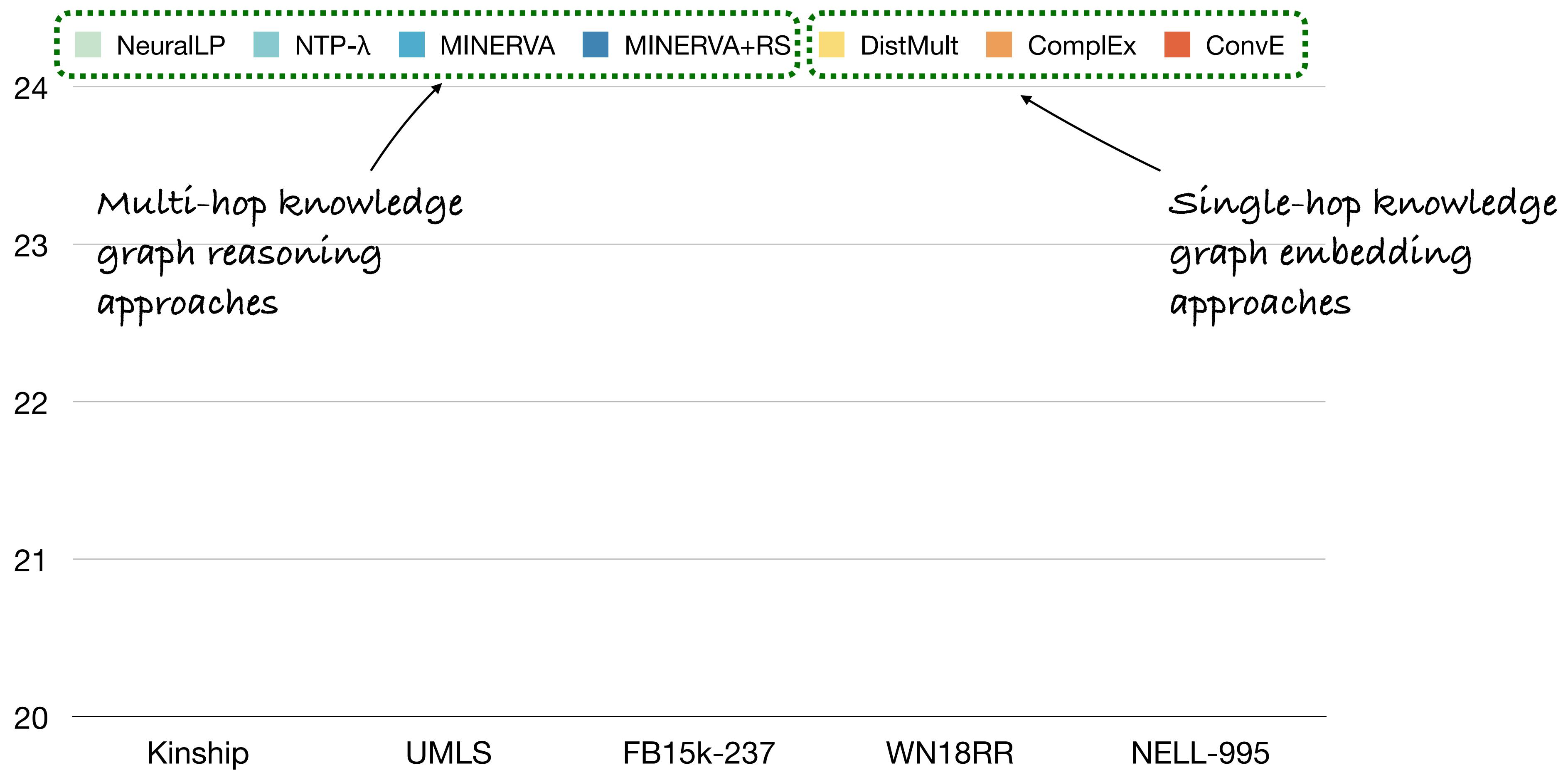
Experiment Setup

KG Benchmarks

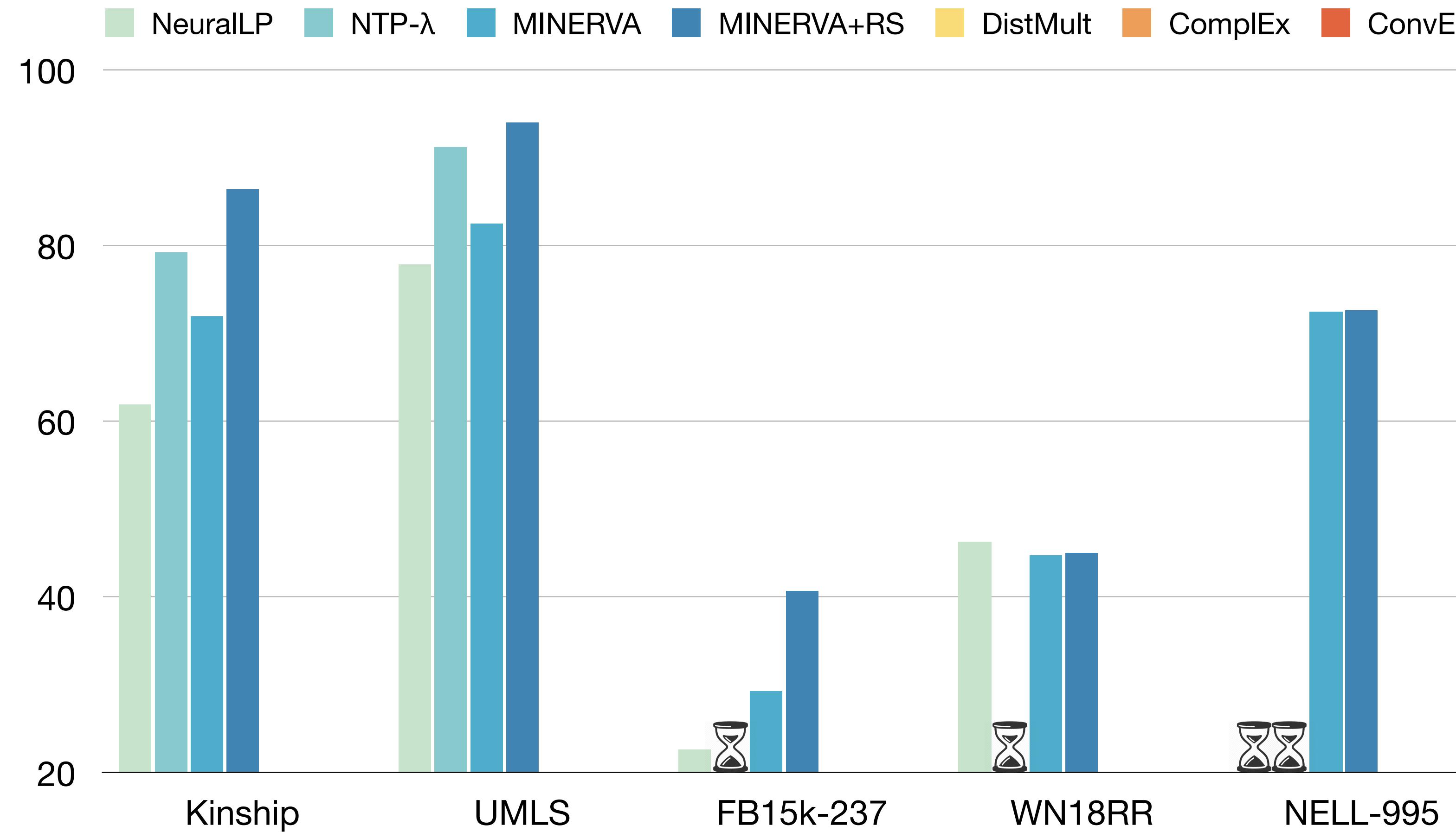
Name	# Ent.	# Rel.	# Fact	# Degree Avg	# Degree Median
Kinship	104	25	8,544	85.15	82
UMLS	135	46	5,216	38.63	28
FB15k-237	14,505	237	272,115	19.74	14
WN18RR	40,945	11	86,835	2.19	2
NELL-995	75,492	200	154,213	4.07	1

Decreasing
connectivity

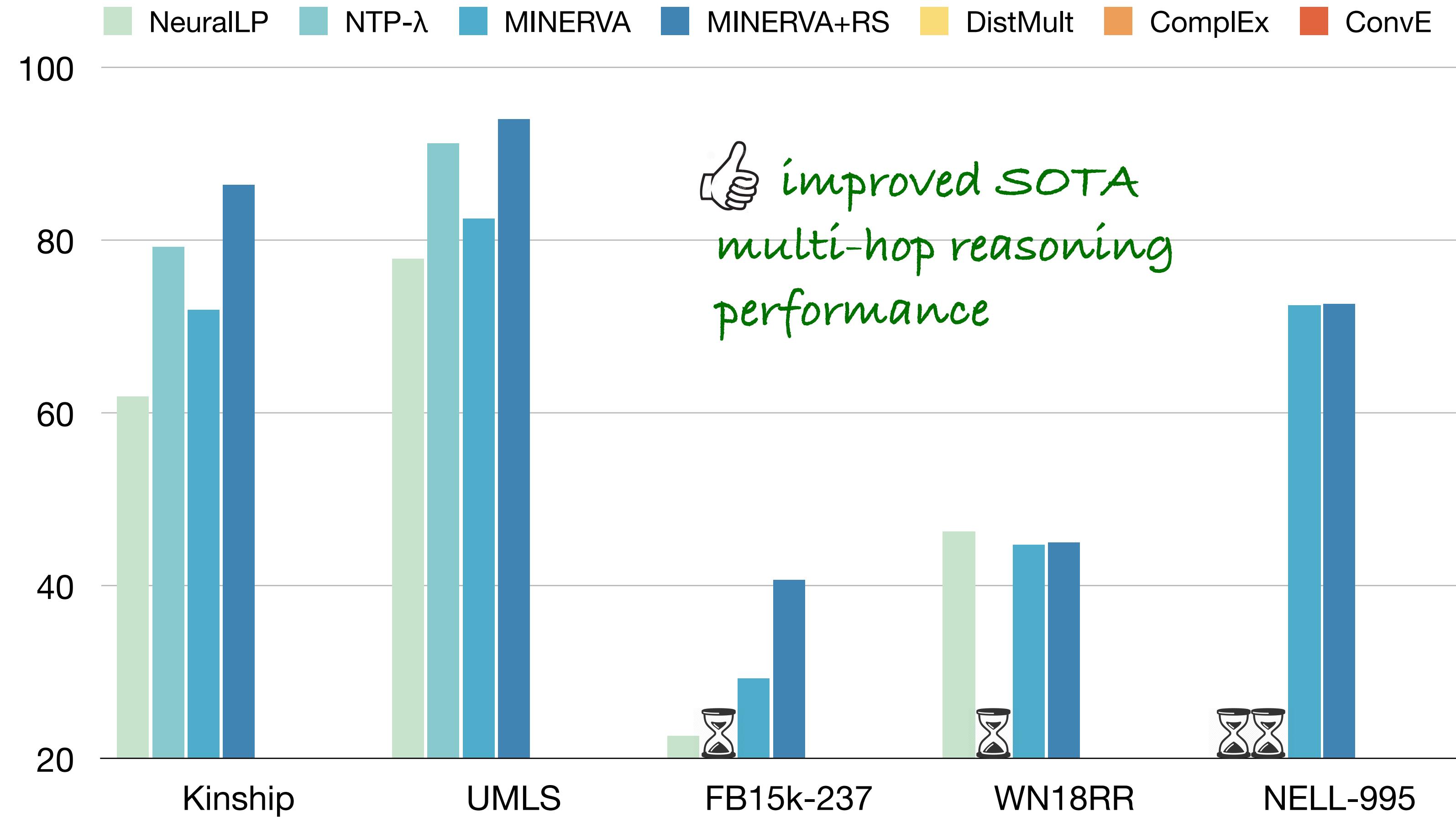
Main Results



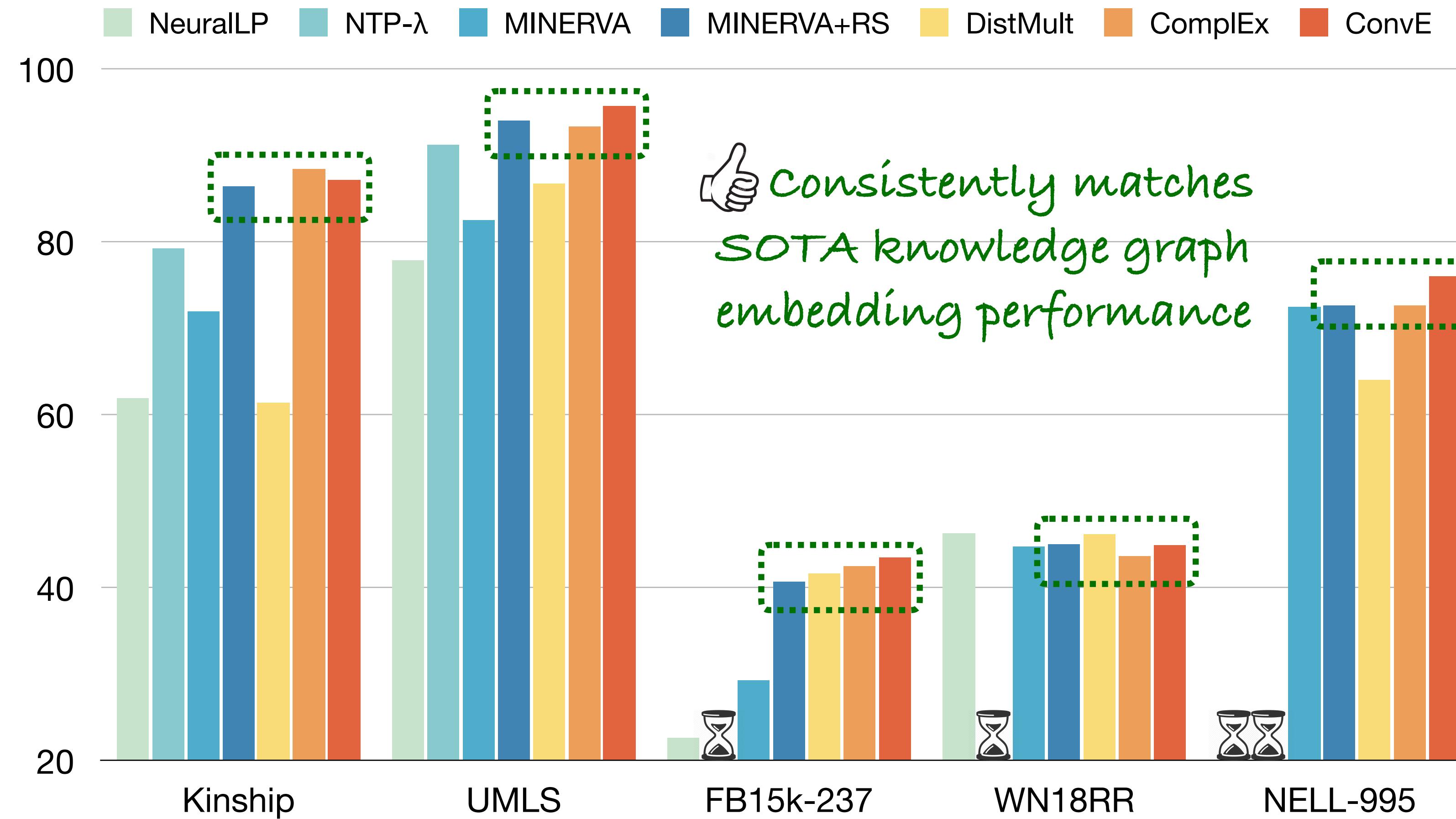
Main Results



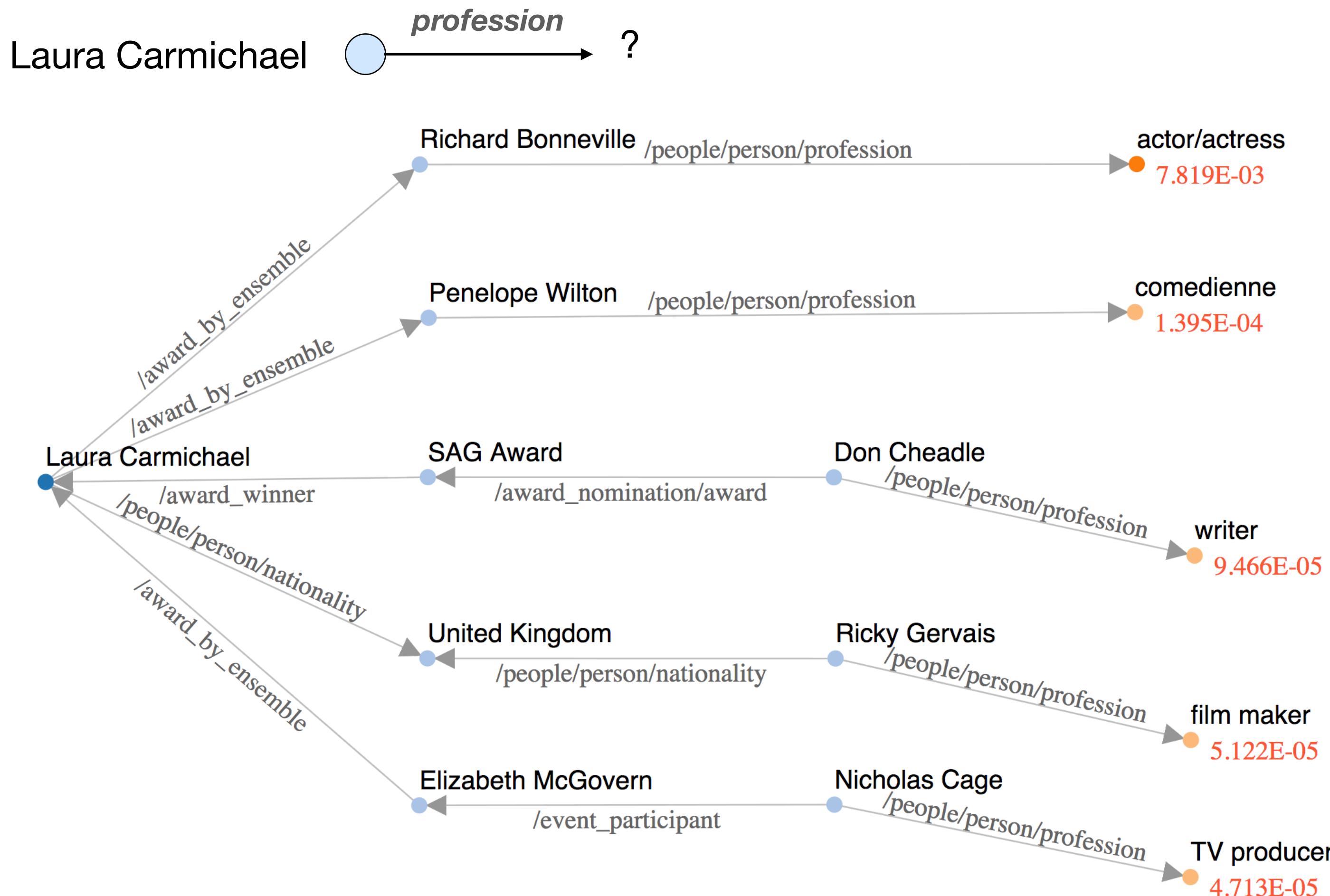
Main Results



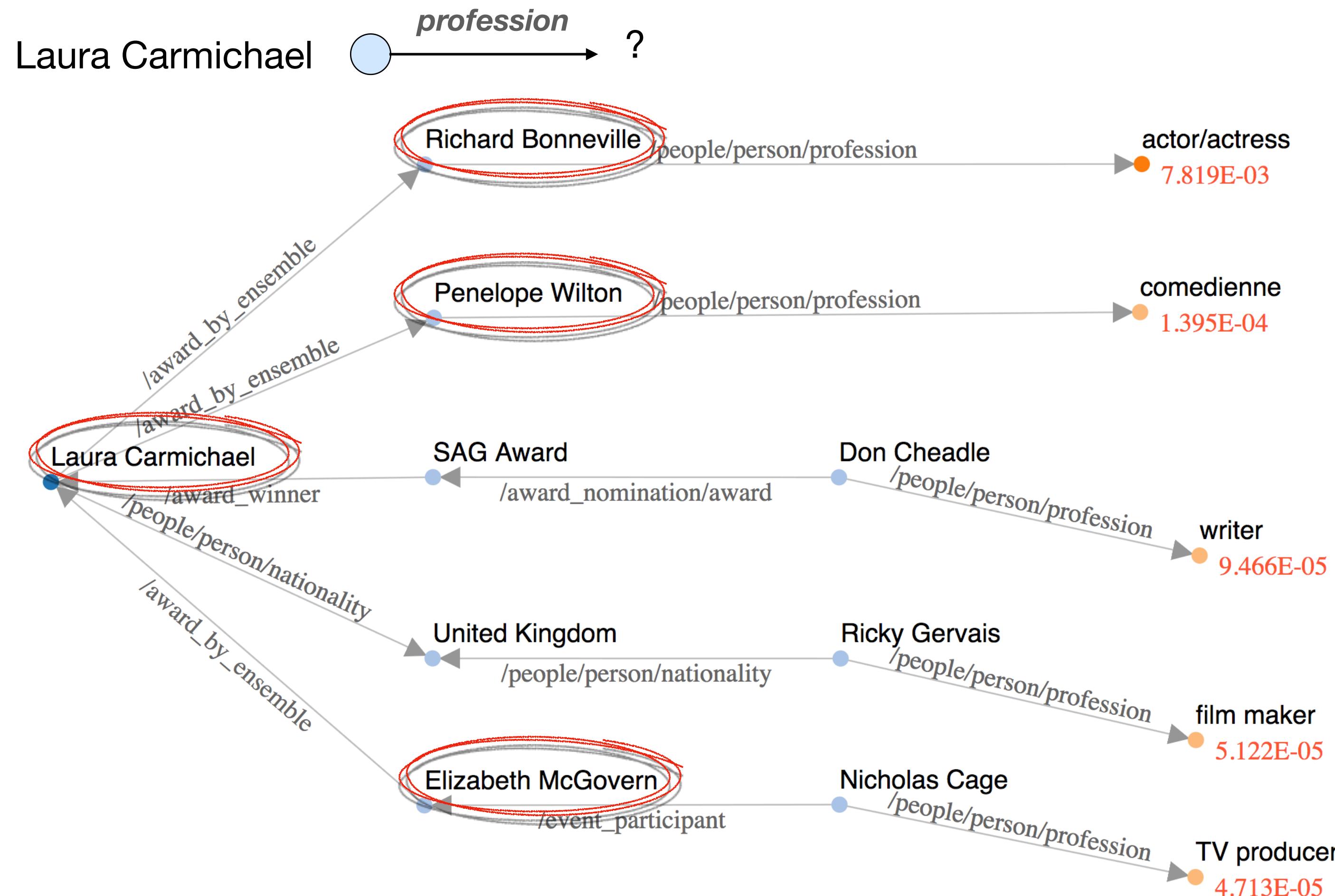
Main Results



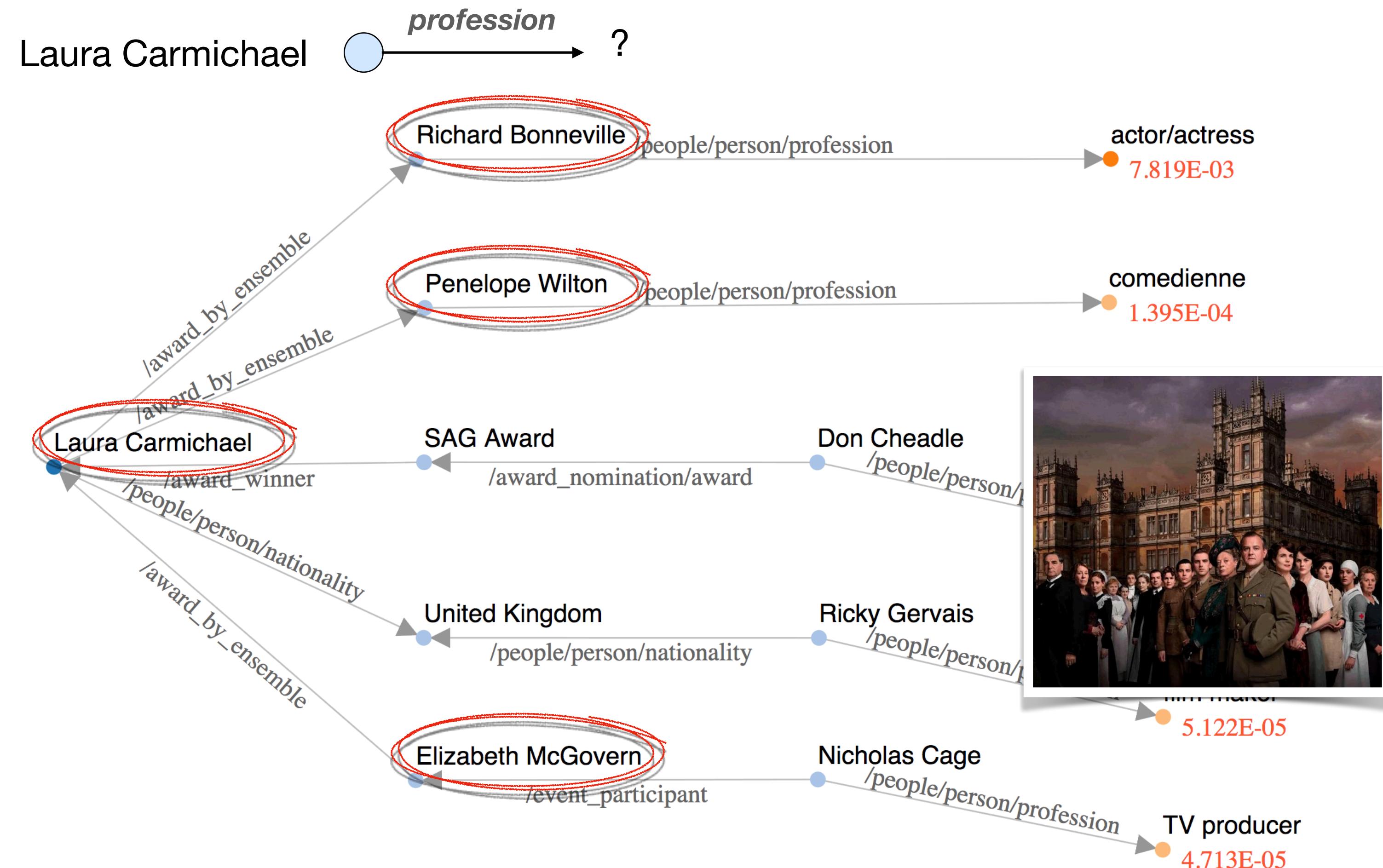
Interpretable Results



Interpretable Results

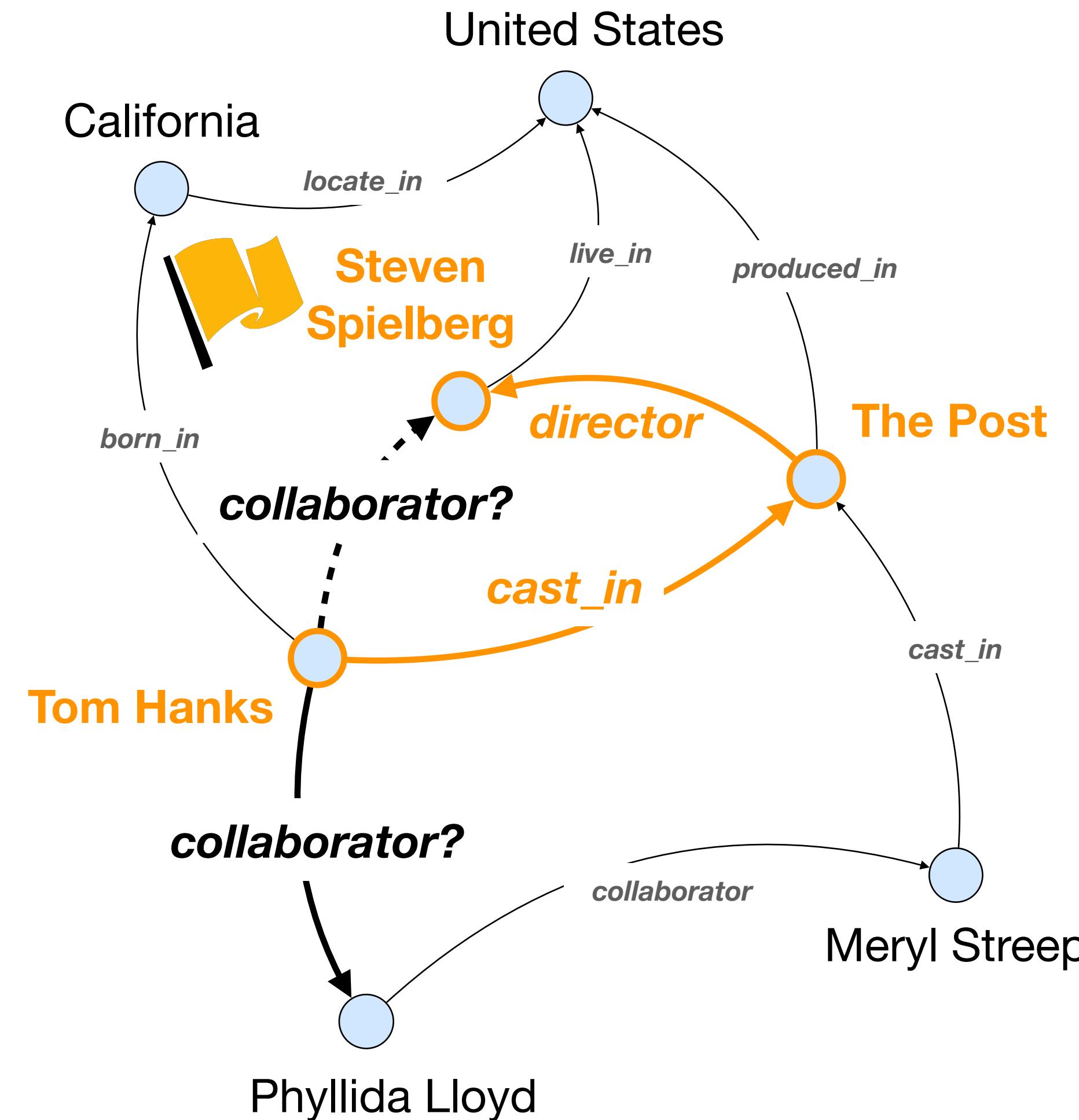


Interpretable Results



Sequential Multi-Hop Reasoning

- Efficient for $e_s \xrightarrow{r_q} ?$ queries
- Return multiple predicted target entities with best-first search
- Can resolve $e_s \xrightarrow{r_q ?} e_t$ queries



Open Source: <https://github.com/salesforce/MultiHopKG>

- Knowledge graph reasoning is critical for KG-based applications as KGs are intrinsically incomplete
- **(Deep) reinforcement learning** provides a strong family of algorithms for learning **informative** reasoning paths while being **time and space efficient**
- Our work combines policy network with KG embedding based reward shaping is **the first sequential multi-hop reasoning approach that matches the performance of KG embedding based approaches** on multiple benchmarks
- Future work could learn more from core RL research to resolve generic (e.g. sparse reward) and KG-specific learning challenges



thank
you

