M-Walk: Learning to Walk over Graphs using Monte Carlo Tree Search

Yelong Shen*[1], Jianshu Chen*[1], Po-Sen Huang*[2], Yuqing Guo [2], Jianfeng Gao [2]

*Equal Contribution, [1]Tencent AI Lab, [2]Microsoft Research

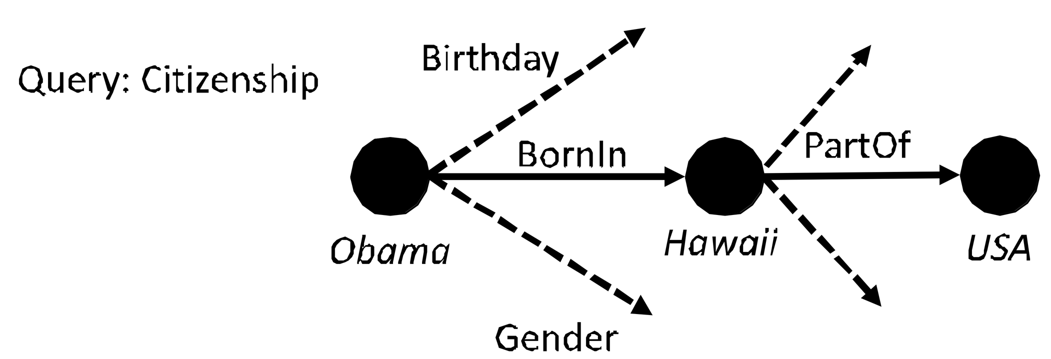
Microsoft Research

Overview

- Learning to walk over a graph towards a target node given input query and a source node.
- M-Walk consists a recurrent neural network and a Monte Carlo Tree Search (MCTS).
- MCTS is combined with the RNN policy to generate trajectories with more positive rewards.
- RNN policy is updated in an off-policy manner from trajectories.
- Experiment results: learn better policies from less number of rollouts compared to policy gradient methods.
- **Code: https://github.com/yelongshen/GraphWalk**

Problem Setting

- Given a pair of source node and query, learn to find a target node in a graph.



Training Algorithm

Algorithm 1 M-Walk Training Algorithm

Input: Graph $\mathcal{G}$; Initial node n_S; Query q; Target node n_T; Maximum Path Length $T_{\max}$; MCTS Search Number E;
for episode e in $[1..E]$ **do**
  Set current node $n_0 = n_S$; $q_0 = f_{\theta_q}(q, 0, 0, n_0)$
  for $t = 0 \ldots T_{\max}$ **do**
    Lookup from dictionary to obtain $W(s_t, a)$ and $N(s_t, a)$
    Select the action a_t with the maximum PUCT value:

$$a_t = \text{argmax}_a \left\{ c \cdot \pi_\theta(a|s_t)^\beta \frac{\sqrt{\sum_{a'} N(s_t, a')}}{1 + N(s_t, a)} + \frac{W(s_t, a)}{N(s_t, a)} \right\}$$

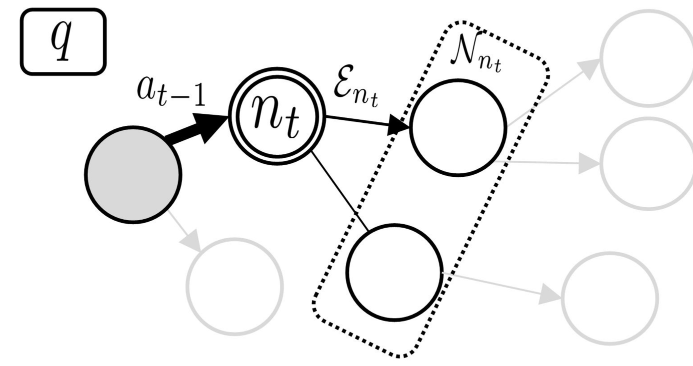
    Update $q_{t+1} = f_{\theta_q}(q_t, h_{A,t}, h_{a_t,t}, n_{t+1})$
    if a_t is STOP **then**
      Compute estimated reward value $V_\theta(s_t) = Q(s_t, a_t = \texttt{STOP})$
      Add generated path p into a path list
      Backup along the path p to update visit count $W(s_t, a)$ and $N(s_t, a)$
      Break
    end if
  end for
end for
for each path p in the path list **do**
  Set reward $r = 1$ if the end of the path $n_t = n_T$ otherwise $r = 0$
  Repeatedly update the model parameters with Q-learning:

$$\theta \leftarrow \theta + \alpha \cdot \nabla_\theta Q_\theta(s_t, a_t) \times \left(r(s_t, a_t) + \gamma \max_{a'} Q_\theta(s_{t+1}, a') - Q_\theta(s_t, a_t) \right)$$

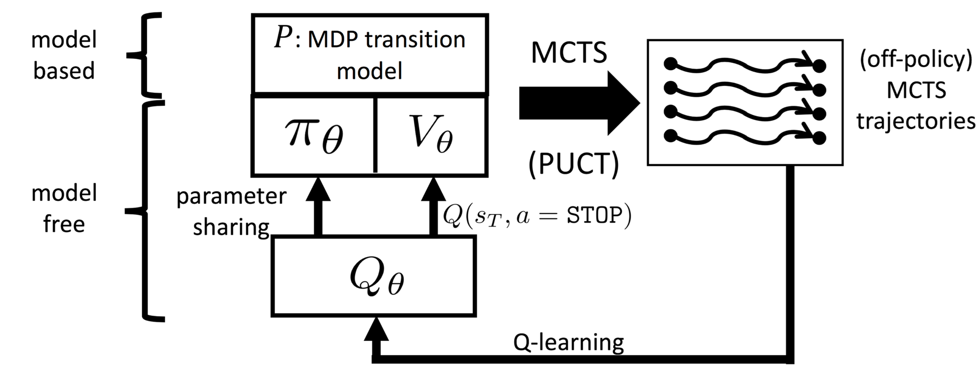
end for

Model

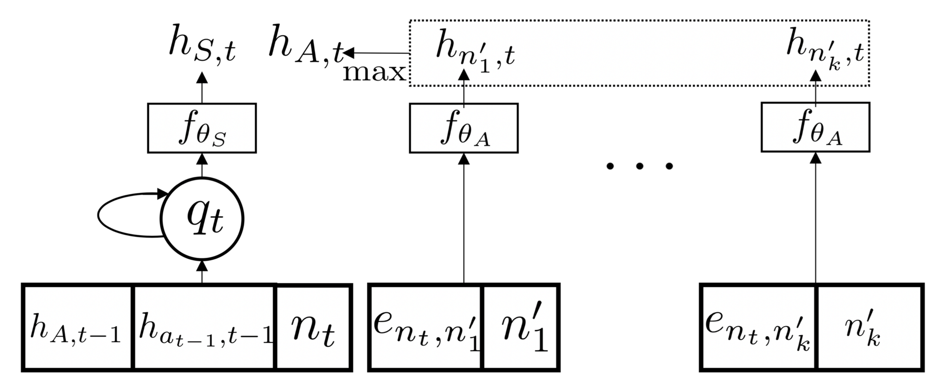
- Markov decision process



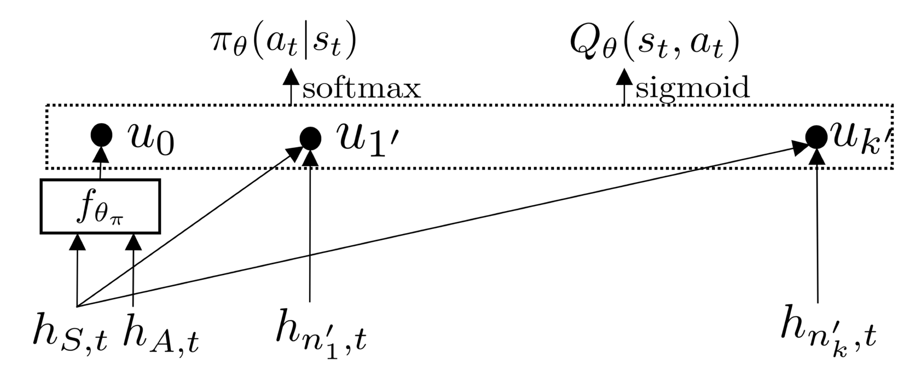
- Iterative policy improvement



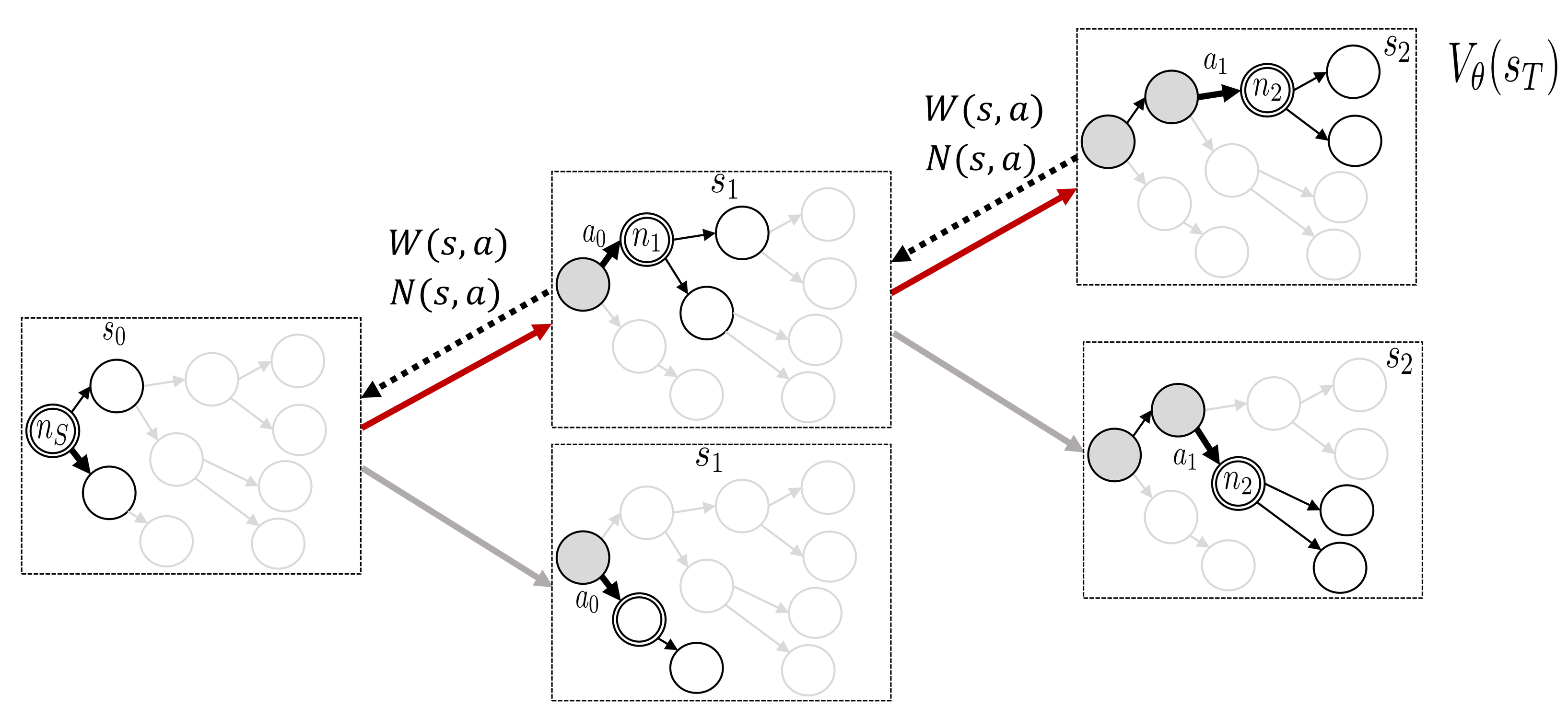
- Use fully connected neural networks to encode q_t along with other quantities $\mathcal{E}_{n_t}$ and $\mathcal{N}_{n_t}$ into a high-level embedding vectors $h_{S,t}, h_{n_1',t}, \ldots, h_{n_k',t}, h_{A,t}$



- Mapped them into the Q-value, the policy and the state value at different output units.



- Jointly train the RNN to model Q_θ and π_θ
- The Monte Carlo Tree Search in M-Walk. The path is a trajectory generated by MCTS using the PUCT (Rosin 11, Silver 17)



Experimental Results

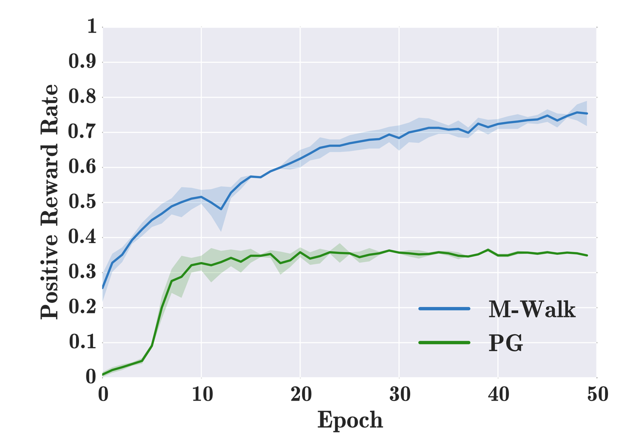
- NELL-995 Link Prediction Performance (MAP)

Tasks	M-Walk	PG-Walk	Q-Walk	MINERVA	DeepPath	PRA	TransE	TransR
AthletePlaysForTeam	**84.7** (1.3)	80.8 (0.9)	82.6 (1.2)	82.7 (0.8)	72.1 (1.2)	54.7	62.7	67.3
AthletePlaysInLeague	**97.8** (0.2)	96.0 (0.6)	96.2 (0.8)	95.2 (0.8)	92.7 (5.3)	84.1	77.3	91.2
AthleteHomeStadium	91.9 (0.1)	91.9 (0.3)	91.1 (1.3)	**92.8** (0.1)	84.6 (0.8)	85.9	71.8	72.2
AthletePlaysSport	98.3 (0.1)	98.0 (0.8)	97.0 (0.2)	**98.6** (0.1)	91.7 (4.1)	47.4	87.6	96.3
TeamPlaySports	**88.4** (1.8)	87.4 (0.9)	78.5 (0.6)	87.5 (0.5)	69.6 (6.7)	79.1	76.1	81.4
OrgHeadquaterCity	**95.0** (0.7)	94.0 (0.4)	94.0 (0.6)	94.5 (0.3)	79.0 (0.0)	81.1	62.0	65.7
WorksFor	**84.2** (0.6)	84.0 (1.6)	82.7 (0.2)	82.7 (0.5)	69.9 (0.3)	68.1	67.7	69.2
BornLocation	81.2 (0.0)	**82.3** (0.6)	81.4 (0.5)	78.2 (0.0)	75.5 (0.5)	66.8	71.2	81.2
PersonLeadsOrg	**88.8** (0.5)	87.2 (0.5)	86.9 (0.5)	83.0 (2.6)	79.0 (1.0)	70.0	75.1	77.2
OrgHiredPerson	**88.8** (0.6)	87.2 (0.4)	87.8 (0.9)	87.0 (0.3)	73.8 (1.9)	59.9	71.9	73.7
Overall	**89.9**	88.9	87.8	87.6	78.8	69.7	72.3	77.5

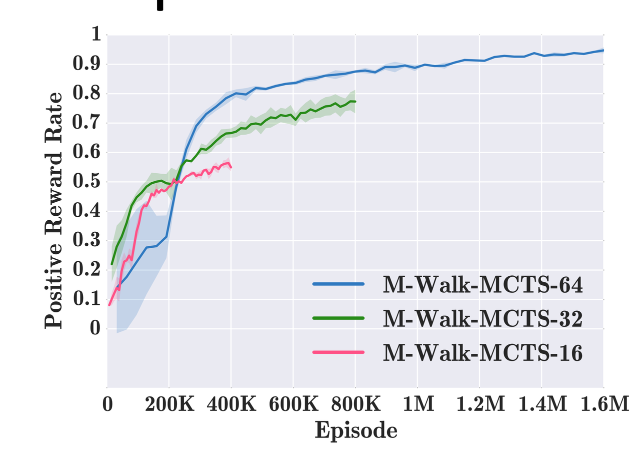
- WN18RR Link Prediction Performance

Metric (%)	M-Walk	PG-Walk	Q-Walk	MINERVA	ComplEx	ConvE	DistMult	NeuralLP
HITS@1	**41.4 (0.1)**	39.3 (0.2)	38.2 (0.3)	35.1 (0.1)	38.5 (0.3)	39.6 (0.3)	38.4 (0.4)	37.2 (0.1)
HITS@3	44.5 (0.2)	41.9 (0.1)	40.8 (0.4)	44.5 (0.4)	43.9 (0.3)	**44.7 (0.2)**	42.4 (0.3)	43.4 (0.1)
MRR	**43.7 (0.1)**	41.3 (0.1)	40.1 (0.3)	40.9 (0.1)	42.2 (0.2)	43.3 (0.2)	41.3 (0.3)	43.5 (0.1)

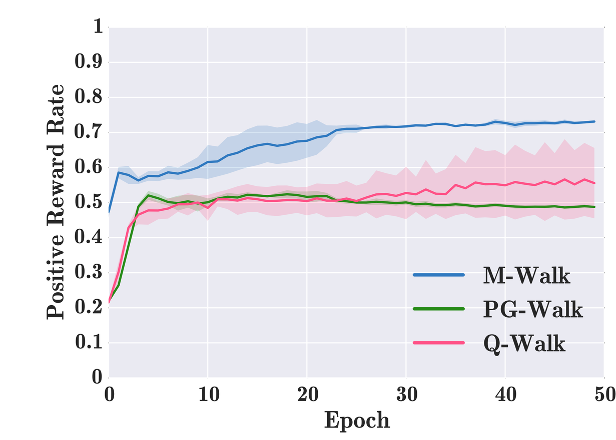
Positive Reward Rate Comparison



Train Rollouts = 32

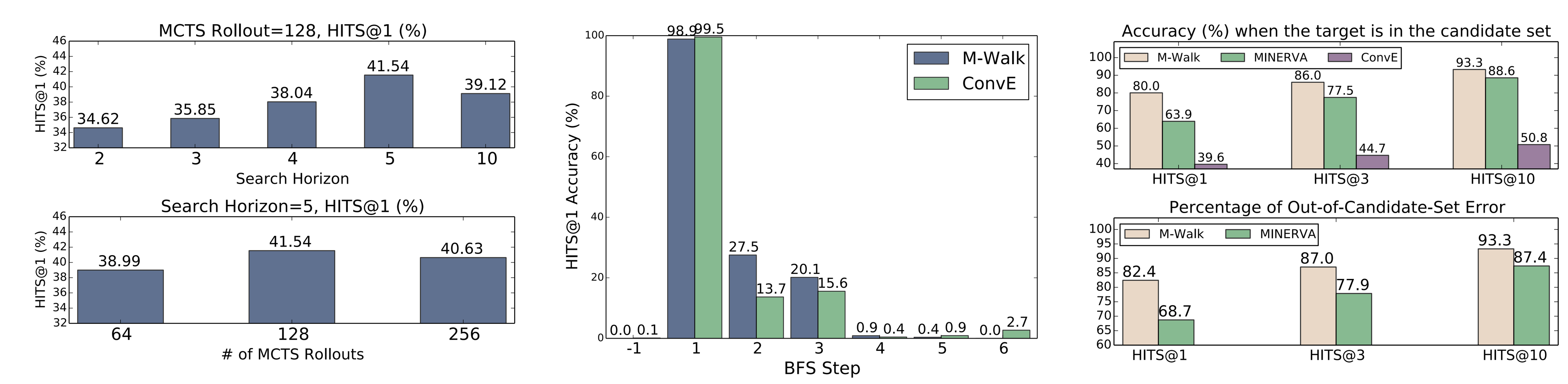


MCTS Comparison



Relation: WorksFor

Hyperparameter and Error Analysis on WN18RR



Examples of Paths found by M-Walk

AthleteHomeStadium:

Example 1: athlete ernie banks $\xrightarrow{\text{AthleteHomeStadium}}$?

athlete ernie banks $\xrightarrow{\text{AthletePlaysInLeague}}$ SportsLeague mlb $\xrightarrow{\text{TeamPlaysInLeague}^{-1}}$ SportsTeam chicago cubs $\xrightarrow{\text{TeamHomeStadium}}$ StadiumOrEventVenue wrigley field, (True)

Example 2: coach jim zorn $\xrightarrow{\text{AthleteHomeStadium}}$?

coach jim zorn $\xrightarrow{\text{CoachWonTrophy}}$ AwardTrophyTournament super bowl $\xrightarrow{\text{TeamWonTrophy}^{-1}}$ SportsTeam redskins $\xrightarrow{\text{TeamHomeStadium}}$ StadiumOrEventVenue fedex field, (True)

Example 3: athlete oliver perez $\xrightarrow{\text{AthleteHomeStadium}}$?

athlete oliver perez $\xrightarrow{\text{AthletePlaysInLeague}}$ SportsLeague mlb $\xrightarrow{\text{TeamPlaysInLeague}^{-1}}$ SportsTeam chicago cubs $\xrightarrow{\text{TeamHomeStadium}}$ StadiumOrEventVenue wrigley field, (False)