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.

Query: Citizens\( \triangleright \) Query: Obama

Birthday

BornIn

Gender

StateOf

USA

Algorithm 1 M-Walk Training Algorithm

1. Equal: Graph \( G \). Initial node \( s_0 \); Query \( q \); Target node \( t \). Maximum Path Length \( T_{\text{max}} \). MCTS Search-Number \( N \).
2. for episode \( e \in [1..E] \) do
3. Set current node \( n_0 \) is \( s_0 \); \( q_0 \) is \( q \); \( o_0 \) is \( 0 \), \( n_0 \).
4. for \( t = 0 \) to \( T_{\text{max}} \) do
5. Lookup from dictionary to obtain \( V(n, o) \) and \( N(n, o) \).
6. Select the action \( a_t \) with the maximum PUCT value:
   \[
   a_t = \arg \max \left\{ \frac{N_t(n, o_t) + \alpha N_t(n, o_t)}{1 + N_t(n, o_t)} \cdot \frac{W(n, o_t)}{N(n, o_t)} \right\}
   \]
7. Update \( n_{t+1} = f(q, h_{t}, a_{t}, h_{t-1}, n_{t}, o_{t}) \).
8. if \( a_t \) is STOP then
9. Compute estimated reward value \( V(n_{t+1}) = Q(n_{t+1}, o_{t+1}) \).
10. Add generated path \( p \) into a path list.
11. Break along the path \( p \) to update visit count \( V(n, o) \) and \( N(n, o) \).
12. if end then
13. end if
14. end for
15. end for
16. end for
17. Set reward \( \gamma \) if the end of the path \( n_{T_{\text{max}}} = t \), otherwise \( \gamma = 0 \).
18. Repeatedly update model parameters with Q-learning:
   \[
   \theta \leftarrow \theta + \alpha \left( Q_{\theta}(s_{t-1}, a_{t}) - Q_{\theta}(s_{t}, n_{t-1}, a_{t}) \right)
   \]
19. end for

Model

- Markov decision process
- Iterative policy improvement

![Diagram of Model](image)

- Use fully connected neural networks to encode \( q_t \) along with other quantities \( E_{n_t} \) and \( N_{n_t} \) into a high-level embedding vectors \( h_{S_{t}}, h_{A_{t}}, h_{B_{t}}, h_{T_{t}} \).

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

- Jointly train the RNN to model \( Q_{\theta} \) and \( \pi_{\theta} \).

- 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)

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Hits@1</th>
<th>Hits@3</th>
<th>Hits@10</th>
<th>Hits@20</th>
<th>Hits@50</th>
<th>Hits@100</th>
</tr>
</thead>
<tbody>
<tr>
<td>DeepPath</td>
<td>43.7 (0.1)</td>
<td>43.7 (0.1)</td>
<td>40.9 (0.1)</td>
<td>42.2 (0.2)</td>
<td>43.3 (0.3)</td>
<td>43.5 (0.1)</td>
</tr>
<tr>
<td>M-Walk</td>
<td>43.7 (0.1)</td>
<td>43.7 (0.1)</td>
<td>40.9 (0.1)</td>
<td>42.2 (0.2)</td>
<td>43.3 (0.3)</td>
<td>43.5 (0.1)</td>
</tr>
<tr>
<td>MINERV A</td>
<td>43.7 (0.1)</td>
<td>43.7 (0.1)</td>
<td>40.9 (0.1)</td>
<td>42.2 (0.2)</td>
<td>43.3 (0.3)</td>
<td>43.5 (0.1)</td>
</tr>
</tbody>
</table>

- WN18RR Link Prediction Performance

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Hits@1 (%)</th>
<th>Hits@3 (%)</th>
<th>Hits@10 (%)</th>
<th>Hits@20 (%)</th>
<th>Hits@50 (%)</th>
<th>Hits@100 (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MINERV A</td>
<td>83.0 (2.6)</td>
<td>79.0 (1.0)</td>
<td>70.0</td>
<td>75.1</td>
<td>77.2</td>
<td>38.4 (4.6)</td>
</tr>
<tr>
<td>M-Walk</td>
<td>84.6 (0.8)</td>
<td>85.9</td>
<td>71.8</td>
<td>72.2</td>
<td>78.6</td>
<td>43.2 (0.3)</td>
</tr>
<tr>
<td>DeepPath</td>
<td>84.3 (0.8)</td>
<td>85.9</td>
<td>71.8</td>
<td>72.2</td>
<td>78.6</td>
<td>43.2 (0.3)</td>
</tr>
<tr>
<td>ConvE</td>
<td>78.2 (0.3)</td>
<td>75.5 (0.5)</td>
<td>66.8</td>
<td>71.2</td>
<td>81.2</td>
<td>35.2 (0.1)</td>
</tr>
</tbody>
</table>

- Positive Reward Rate Comparison

- Train Rollouts = 32
- MCTS Comparison
- Relation: WorksFor

Hyperparameter and Error Analysis on WN18RR

Examples of Paths found by M-Walk

- Examples 3: Athlete's silver paces...