Motivation

- Planning trips is a complex decision problem
- Many decisions to be made at once: duration, costs, places to visit, food and many more!
- Overload of information on the Web renders task tedious

Markov Decision Process (MDP) Framework

- State: a sequence of at most k places the user visited until now
- Actions: all POI categories present in the city
- Transition & reward function estimate by maximum-likelihood

\[
T(s, a, s') = \frac{\text{count}(s'|s)}{\sum_{s''} \text{count}(s''|s)} \quad R(s, a) = \frac{\text{count}(s, a)}{\text{count}(s)}
\]

Optimizing the MDP via Value Iteration algorithm:

\[
V(s) = \max_a \left( R(s, a) + \gamma \sum_{s'} T(s, a, s') V(s') \right)
\]

State-action values, \(Q(s, a)\), are obtained from the learned value function which serve as scores for recommendation:

\[
Q(s, a) = R(s, a) + \gamma \sum_{s'} T(s, a, s') V^*(s')
\]

⇒ The goal is to recommend a sequence of POIs given individual user preferences based on previous visited places.

Conclusions

- An RL approach to recommend user itinerary
- Utilize freely available data from social media with minimal manual intervention
- Computationally inexpensive
- Outperforms standard path planning methods

Online Personalization

- Duration-based: Amount of time a user spends on a category
- Frequency-based: Frequency of visiting a certain category

⇒ A POI is recommended from the optimal category: Weighted(distance + personalized score)

Empirical Study

- Leave-one-out cross-validation method
- Baselines: Breadth first search, Dijkstra, Heuristic Search, A*

Partial path accuracy in terms of order of Markov chain:

<table>
<thead>
<tr>
<th>Path Length</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st order</td>
<td>0.041</td>
<td>0.041</td>
<td>0.042</td>
<td>0.042</td>
<td>0.041</td>
<td>0.034</td>
</tr>
<tr>
<td>2nd order</td>
<td>0.098</td>
<td>0.090</td>
<td>0.096</td>
<td>0.106</td>
<td>0.100</td>
<td>0.103</td>
</tr>
<tr>
<td>3rd order</td>
<td>0.097</td>
<td>0.090</td>
<td>0.093</td>
<td>0.105</td>
<td>0.090</td>
<td>0.087</td>
</tr>
<tr>
<td>4th order</td>
<td>0.089</td>
<td>0.084</td>
<td>0.083</td>
<td>0.094</td>
<td>0.077</td>
<td>0.060</td>
</tr>
<tr>
<td>5th order</td>
<td>0.074</td>
<td>0.071</td>
<td>0.058</td>
<td>0.072</td>
<td>0.070</td>
<td>0.058</td>
</tr>
</tbody>
</table>

⇒ Encoding more history in the state improves the performance. Personalization techniques:

Photos without coordinate information: Using Latent Semantic Analysis (LSA) to compute the semantic similarity between the tags of geo-tagged and non-geo-tagged photos

An example of non-geotagged photo

An example of geo-tagged photo

⇒ Duration-based outperforms frequency-based. Exact (left) and partial path accuracy (right) for Paris:

⇒ Our approach outperforms the baselines.