Learning to Rank Paths in Spatial Networks

Yang, Sean Bin; Yang, Bin

Publication date:
2019

Citation for published version (APA):
Learning to Rank Paths in Spatial Networks

Sean Bin Yang and Bin Yang
Department of Computer Science, Aalborg University, Denmark
(seanb.yang@cs.aau.dk)

Introduction
A routing service quality study shows that local drivers often choose paths that are neither shortest nor fastest, rendering classic routing algorithms often impractical in many real-world routing scenarios.

In addition, commercial navigation systems, such as Google Maps and TomTom, often follow a similar strategy by suggesting multiple candidate paths to drivers, although the criteria for selecting the candidate paths are often confidential.

Challenges:
- Constructing an appropriate training path set $P$ is non-trivial.
- Effective training models often rely on meaningful feature representation of input data—how to learning path representation.

Our approach:
- Training Data Generation: A compact set of diversified paths using trajectories as training data.
- Path Representation: An end-end deep learning framework is presented to solve the regression problem.
  - A spatial network embedding is proposed to embed each vertex to a feature vector by considering the road network topology.
  - Since a path is represented by a sequence of vertices, recurrent neural network is applied to model the sequence.
- The RNN finally outputs an estimated similarity score, which is compared against the ground truth similarity.

Solution Overview
- We propose a data-driven ranking framework PathRank, which ranks candidate paths by taking into account the paths used by local drivers in their historical trajectories.
- Most importantly, PathRank models ranking candidate paths as a "regression" problem—for each candidate path, PathRank estimates a ranking score for the candidate path.
- Solution Overview.

Training Data Generation
- We proceed to elaborate how to generate a set of training paths for a trajectory path $P$ from source $s$ to destination $d$.
- We propose the strategy using the diversified top-$k$ shortest paths.

Algorithm 1: Top-$k$ Diversified Paths

```
Input: Road network $G$, source $s$, destination $d$, integer $k$, similarity threshold $\delta$
Output: The diversified top-$k$ paths: $DkPS$
1. Add the shortest path $P_1$ into $DkPS$;
2. while $DkPS < k$ do
3. Identify the next shortest path $P_2$;
4. Boolean tag $\leftarrow$ true;
5. for each path $P \in DkPS$ do
6. if $\text{sim}(P_2, P) \geq \delta$ then
7. tag $\leftarrow$ false;
8. break;
9. if tag then
10. Add $P_2$ into $DkPS$;
11. return $DkPS$;
```

PathRank

PathRank Overview.

Vertex Embedding:
- Node2vec is used to embed road network and initialize vertex embedding layer.

Recurrent Neural Network (RNN):

Experiments

Experiments Setup
- Road Network and Trajectories: North Jutland, Denmark, 180 million GPS records from 183 vehicles.
- Ground Truth Data: For each trajectory $P_r$. We generate two sets of training paths: Top-$k$ shortest paths (TkDI) and diversified top-$k$ shortest paths (D-TkDI).
- For each training path $P$, we employed Jaccard similarity $\text{WeightedJaccard}(P, P_1)$ as $P_r$’s ground truth ranking score.

Evaluation Metrics:
- Mean Absolute Error (MAE) and Mean Absolute Relative Error (MARE)
- Kendall Rank Correlation Coefficient ($\tau$) and Spearman’s Rank Correlation Coefficient ($\rho$)

Experiments Results
- Table 1 shows that (1) when using the diversified top-$k$ paths for training, we achieve higher accuracy compared to when using top-$k$ paths; (2) a larger embedding feature size $M$ achieves better results.
- Table 2 shows the results. In addition, PR-A2 achieves better accuracy than does PR-A1, meaning that updating embedding matrix $B$ is useful.

<table>
<thead>
<tr>
<th>Strategies</th>
<th>MAE</th>
<th>MARE</th>
<th>$\tau$</th>
<th>$\rho$</th>
</tr>
</thead>
<tbody>
<tr>
<td>TkDI</td>
<td>0.1433</td>
<td>0.2300</td>
<td>0.6638</td>
<td>0.7044</td>
</tr>
<tr>
<td>D-TkDI</td>
<td>0.1169</td>
<td>0.1875</td>
<td>0.6913</td>
<td>0.7330</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Strategies</th>
<th>MAE</th>
<th>MARE</th>
<th>$\tau$</th>
<th>$\rho$</th>
</tr>
</thead>
<tbody>
<tr>
<td>TkDI</td>
<td>0.1140</td>
<td>0.1830</td>
<td>0.6959</td>
<td>0.7346</td>
</tr>
<tr>
<td>D-TkDI</td>
<td>0.0955</td>
<td>0.1533</td>
<td>0.7077</td>
<td>0.7492</td>
</tr>
</tbody>
</table>

Table 1: Training Data Generation Strategies, PR-A1

Table 2: Training Data Generation Strategies, PR-A2