Spatio-temporal Keyword Query in Semantic Trajectories

Xia WU, Jiankun YU, Xiaoming ZHAO

Frontiers of Computer Science, DOI: 10.1007/s11704-020-0039-4

Problems & Ideas

- Problem
  - We study the problem of Spatio-Temporal Keyword query in Semantic Trajectories (STKST).
  - Given the position of a user and a set of keywords with temporal constraint, we aim to find the trajectories that contain the most semantically and temporally relevant keywords and also are close to the position of the user.
- Similarity Measure
  - Semantic Similarity Measure(sim_{ss}): Inspired by WordNet,we build a LM-Tree to index all the classes of landmarks in a spatial space; based on the intuition that two classes are similar if their superclasses/subclasses are similar, we propose a measurement by the Jaccard similarity of keywords' extended class sets.
  - The Spatial(dis_{sd}) and Temporal(dis_{td}) Distances: Euclidean distance
  - Overall Similarity Measure:

$$Sim = \delta \cdot sim_{ss}$$

Where $\delta = 2/(1+e^{\alpha \cdot dissd + (1-\alpha) \cdot distd})$

Main Contributions

- We give the formalized definition of the problem of Spatio-Temporal Keyword query in Semantic Trajectories.
- We define a new measurement to evaluate the similarity between the spatial keywords and landmark labels. This similarity can estimates two words if they are semantic similar but literal dissimilar.
- We take an experimental evaluation on real trajectory dataset. We add two existing methods (GAT and DPA) into our experiment as baselines. We find that they may make mistakes on similarity matching with temporal constrains, or if a keyword and a stop are semantic similar but literal dissimilar. But proposed measurement can answer the STKST problem with 100% accuracy.

Method	k=5	k=10	k=15	k=20	k=25
STKST-f	1.00	1.00	1.00	1.00	1.00
GAT-t	0.12	0.15	0.16	0.16	0.21
DPA-t	0.12	0.16	0.17	0.18	0.19
GAT	0.21	0.22	0.23	0.24	0.25
DPA	0.61	0.64	0.66	0.69	0.71

The Accurate Rate of Proposed Measurement

* Here k is the number of trajectories returned by the algorithms