1. **Summary.** The paper presents a Route Interface framework based on Collective Knowledge (RICK) to construct the popular routes (top-k) from uncertain trajectories. The paper considers uncertain trajectories, that are sequences of spacial positions formed by few samples (sample are quite sparse in the time horizon), and no knowledge of the underlaying road network is given. The solution presented in the paper involves two key steps: **routable graph construction** and **route inference.** The first component is in charge of constructing an underlaying graph from the uncertain trajectories by connecting regions. The second component uses the graph to generate the route (answer the query) by searching the more likely trajectories. I found interesting the routable graph construction and how the author can discover connected geographical areas by learning among uncertain trajectories.

2. **Positive Points.**
   - No given knowledge of the underlaying road network is needed. This makes this approach suitable for a variety of applications.
   - The construction of the routable graph is performed efficiently by using the notion of enclosed regions and the index data structure proposed by the authors.
   - From the experiment sections RICK provides high utility results in the scenario tested by the authors even with sparse data.

3. **Negative Points.**
   - The route inference step is computationally intensive and the computational analysis is missing. This point in my opinion is an important drawback for this approach since the authors proposed the route inference as on-line step. Furthermore, it turns out that even the base line approach is considerable faster than RICK in answering queries.
• The inference approach for constructing the edges in the routable graph is based on the shortest path notion. This may not work properly in some cases. Since the road network is not given, the use of shortest distance may be misleading in those cases where the presence of obstacles force to take longer paths along the trajectory.

• The experiments are limited on short sequences of points, $|q| \leq 4$. I really doubt that this approach is suitable for longer sequences since from the experiments and from the generation strategy for the route seem that the time complexity respect $|q|$ grows very fast.

4. Discussion.

• The construction of the routable graph reminds me about density based clustering. So I was wondering if the construction of this graph could be casted as clustering problem. In this way this step could be addressed in a streaming fashion that could lead to an on-line algorithm to support the presence of new coming trajectories in the dataset and to modify the routable graph consequently.

• The optimization process for finding the trajectory with highest score is not clear to me. Some clarifications on convergence and complexity of the method are missing.

• It could be interesting to study the case where the user that issues the query could also assign a weight to the points in the trajectory that he/she is looking for. So that the user can specify how much each point is important for him/her. In this way, we could return some approximate solutions that are faster to compute but where for example we could drop those points where the weight is smaller.