Information Sciences 450 (2018) 343–360

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Group-based keyword-aware route querying in road networks

Sen Zhao[a,c,*], Lei Zhao[b], Sen Su[a], Xiang Cheng[a], Li Xiong[c]

[a] Beijing University of Posts and Telecommunications, China
[b] Renmin University of China, China
[c] Emory University, USA

ARTICLE INFO

Article history:
Received 17 March 2017
Revised 22 March 2018
Accepted 23 March 2018
Available online 27 March 2018

Keywords:
Group-based
Keyword-aware
Route query
Road network
Approximation algorithms

ABSTRACT

This paper addresses a group-based keyword-aware route (GKAR) query problem in road networks. The road network with Points of Interest (POIs) is modeled as an undirected graph, where each node with coordinates represents a road intersection or a POI, and each weighted edge represents a road segment. The GAKR query aims to find a POI route such that it passes by a set of required keywords sequentially, and the cost of the route is minimized. The GKAR query problem is NP-hard. To solve this problem, we develop a series of query processing algorithms. We first propose two efficient algorithms which give an *(n+3)*-factor approximation and an *(n+1)*-factor approximation respectively to find a feasible POI route, where *n* represents the number of query keywords. The cost of this result is further used to limit the search space in the following algorithms. We then present two exact algorithms, and one greedy algorithm. The first exact algorithm finds the optimal result by enumerating all feasible POI routes. To improve the efficiency, we propose another exact algorithm based on the property of the cost function. At last, we propose a greedy algorithm to make comparisons. Extensive experiments with two real datasets confirm the efficiency, accuracy, and scalability of our proposed algorithms.

1. Introduction

With the rapid popularization of GPS-enabled devices such as cell phone, smart watch and so on, location-based services (LBS) have attracted much attention in recent studies. As a basic query problem, route query is an important part of applications in LBS. For example, we can find a shortest route between two places by using Google Maps[1] or Baidu Map[2]. However, user needs may exist that are satisfied by a route covering multiple objects instead of a simple route. For example, a user may want to find a route to the house such that it passes by a swimming pool, a movie theater and a shopping mall. Some studies (e.g., [1,41]) have considered this type of route query for a user.

Recently, new applications in LBS need to target not only individual users but also user groups. For example, consider three users at different locations want to watch a movie together at the movie theater nearby. The processing of such a query requires taking into account the location of each user. Many studies (e.g., [12,26,32,37,42]) have focused on the queries by a group of users. Consider some close friends living at different places who intend to have a travel together.

* Corresponding author.
E-mail addresses: zhaoseneasy@outlook.com, sen.zhao@emory.edu (S. Zhao).

[1] https://www.google.com/maps.
[2] https://map.baidu.com/.

https://doi.org/10.1016/j.ins.2018.03.058

They are going to meet in a park nearby, followed by a joint visit to a movie theater, and then go to a restaurant for dinner. After the travel, they may go to different destinations such as their offices, workplaces, and homes respectively. The required route should meet the following needs: (1) the route must pass by a park, a movie theater, and a restaurant sequentially; (2) the park in the route is close to all of them such that none of them need to travel a long distance for aggregating; (3) the restaurant in the route is close to all destinations of them; (4) the travel distance of the route is minimized such that they can spend more time relaxing together. Inspired by this, we introduce a new type of query called **g**roup-based **k**eyword-**a**ware **r**oute (GKAR) query, to enable a group of users to find a route which passes by all required keywords sequentially and has minimum cost in a road network.

Specifically, a GKAR query is defined over a road network with POIs which is modeled as an undirected graph G, where each node locating in a two-dimensional space represents a road intersection or a POI attached with a set of keywords, and each weighted edge represents a road segment. The input of the query consists of the following four parameters: $G, \mathcal{U}, \mathcal{D}$ and Ψ, where G is the input road network graph, $\mathcal{U}$ is the group of query nodes in G which represent the users' locations, $\mathcal{D}$ is the group of destination nodes which represent the users' destinations, and Ψ is a set of query keywords. The query returns a POI route that covers all the keywords in Ψ sequentially, and the cost of this route is minimized. In particular, the cost function consists of three parts: (1) the distance cost between the POI route and $\mathcal{U}$ (referred to as query distance cost); (2) the distance cost between the POI route and $\mathcal{D}$ (referred to as destination distance cost); (3) the distance cost of the POI route (referred to as route distance cost). To the best of our knowledge, none of existing studies in the literature is applicable to the GKAR query.

We show the problem of answering GKAR queries is NP-hard. To solve this problem, we first abstract three basic query operations, implemented using the distance oracle based index [33]. Based on these three basic query operations, we design a series of query processing algorithms. Specifically, we first propose two efficient algorithms which give an *(n+3)*-factor approximation and an *(n+1)*-factor approximation respectively to find a feasible POI route, where n represents the number of query keywords. The cost of this POI route is considered as an upper bound, and can be used to limit the search space in following algorithms, including two exact algorithms, and one greedy algorithm. The first exact algorithm is enumeration based and finds the optimal POI route by enumerating all feasible POI routes. However, this algorithm is not efficient because many unnecessary POI routes are enumerated. We observe that the lower bound cost of a candidate route can be determined by the group of query nodes $\mathcal{U}$, the group of destination nodes $\mathcal{D}$, the previously selected POI nodes and the candidate POI node which will be selected, and utilizing the property of the cost function we propose another exact algorithm, called SepPro, which can progressively prune the search space by updating the upper bound with the cost of current best POI route. At last, we propose a greedy algorithm to make comparisons with the above proposed algorithms.

We carry out extensive experiments on two real datasets: the Flickr dataset[3] and the Florida road network dataset[4]. The experimental results on these datasets show that the two approximation algorithms FRU and FRG, and the exact algorithm SepPro are efficient and scalable. In particular, the exact algorithm SepPro can achieve both excellent efficiency and accuracy.

The contributions of this paper are summarized as follows:

1. We define the group-based keyword-aware route (GKAR) query problem, and prove that this problem is NP-hard.
2. We present two approximation algorithms to find a feasible POI route for answering the GKAR query, which give an *(n+3)*-factor approximation and an *(n+1)*-factor approximation, respectively.
3. We present two exact algorithms for answering the GKAR query. Specifically, one is an enumeration based algorithm, and the other (denoted by SepPro) is designed to make use of the property of the cost function.
4. Extensive experimental results on two real datasets show that the proposed algorithms are scalable and capable of excellent performance.

The rest of paper is organized as follows. We formally define the GKAR query and show that the problem of answering GKAR query is NP-hard in Section 2. We present three basic query operations in Section 3. In Section 4 we present two efficient algorithms to find the first feasible POI route, and in Section 5 we propose two exact algorithms and a greedy algorithm for answering GKAR queries. In Section 6 we report on empirical studies. Finally, we discuss related work in Section 7 and conclude the paper in Section 8.

2. Problem statement

We first formally define the group-based keyword-aware route (GKAR) query problem, and then show the hardness of this problem.

Definition 1. Road network graph. A road network with POIs (as shown in Fig. 1(a)) is modeled as a road network graph $G = (V, E)$ (as shown in Fig. 1(b)) which is an undirected graph and consists of a set of nodes V and a set of edges $E \subseteq V \times V$. Each node $v \in V$,which locates in a two-dimensional space, represents a road intersection or a POI. A POI is attached with a set of keywords $v.K$ as its description. Each edge $e \in E$ represents a road segment between two nodes in V, and the edge

[3] https://www.flickr.com/.

[4] http://www.dis.uniroma1.it/challenge9/download.shtml.