International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

IJIEMR Transactions, online available on 15th Nov 2018. Link :http://www.ijiemr.org/downloads.php?vol=Volume-07&issue=ISSUE-12

Title: **IN NETWORKS: CLUE-BASED ROUTE SEARCH ON ROAD**

Volume 07, Issue 12, Pages: 222–226.

Paper Authors

JANI PASHA SHAIK MD, MD.IMRAN

Nimra College of Engineering & Technology, A.P., India.

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

IN NETWORKS: CLUE-BASED ROUTE SEARCH ON ROAD

[1]JANI PASHA SHAIK MD, [2]MD.IMRAN
[1]Student, M. Tech (CSE), NIMRA INSTITUTE OF SCIENCE & TECHNOLOGY, A.P., India.
[2]Assistant Professor, Dept. of Computer Science & Engineering, NIMRA INSTITUTE OF SCIENCE & TECHNOLOGY, A.P., India.
[1]jani.skmd@gmail.com

ABSTRACT: in this paper, investigate a novel query type, namely clue-based route search (CRS), which allows a user to provide clues on textual and spatial context along the route such that a best matching route w.r.t. the clues is returned. More specifically, a CRS query is defined over a road network G, and the input of the query consists of a source vertex vq and a sequence of clues, where each clue contains a query keyword and a user expected network distance. However, in many practical scenarios, an optimal route might not always be desirable. For example, a personalized route query is issued by providing some clues that describe the spatial context between PoIs along the route, where the result can be far from the optimal one. Therefore, propose a greedy algorithm and a dynamic programming algorithm as baselines. To improve efficiency, we develop a branch-and-bound algorithm that prunes unnecessary vertices in query processing. In order to quickly locate candidate, we propose an AB-tree that stores both the distance and keyword information in tree structure. To further reduce the index size, we construct a PB-tree by utilizing the virtue of 2-hop label index to pinpoint the candidate. Extensive experiments are conducted and verify the superiority of our algorithms and index structures.

Keywords: Clue-Based Route Search (CRS), Spatial Context,Networks.

INTRODUCTION

Particularly, in this paper, investigate a novel query type, namely clue-based route search (CRS), which allows a user to provide clues on textual and spatial context along the route such that a best matching route w.r.t. the clues is returned. More specifically, a CRS query is defined over a road network G, and the input of the query consists of a source vertex vq and a sequence of clues, where each clue contains a query keyword and a user expected network distance. A vertex contains a clue keyword is considered as a match vertex. The query returns a path P in G starting at vq, such that (i.) P passes through a sequence of match vertices (PoIs) w.r.t. the clues and (ii.) the network distances between two contagious matched vertices are close to the corresponding user specified distance such that the user's search intention is satisfied.Increased Flexibility in Trip Planning: As mentioned before, most existing work aims to find an optimal route with minimum travel distance. However, in many real scenarios, such an optimal route might not always be desirable. For instance, a user may have some personalized requirements on the distances between PoIs when planning a trip. Consider such a scenario, a user wants to find a buffet restaurant and a nearby cinema only in walking distance, say 3km, thus he can watch a movie after dinner. Therefore, after having delicious food, he can walk to the cinema in order to maintain a healthy lifestyle. These personalized requirements make the route search become distance-sensitive and more flexible such that the distance between PoIs along the route should be as close as possible to the user specified distance.

Clue-Based Route Navigation: Given a description including textual and distance information on an expected route, it is still not direct-viewing enough for users to obtain the exact route. This is usually the case when a user wants to know the way for a specific place and asks the others for help, she may still not be able to exactly figure out the route after obtaining the answers from them, where the answer usually comes in the form, for example, ―go straight on the way for about 100 meters, you will see a cafe, and turn right, you will arrive the Japanese restaurant after about 150 meters walk‖. Therefore, a novel type of route search which automatically interprets the clues contained in such answers becomes necessary. By augmenting it on current navigation services, a better user experience can be provided.

PROPOSED SYSTEM

Disadvantages

o There is no Clue-based Route Navigation to find exact Route path

o The searching is slow due to lack of Greedy Clue search algorithm

 In the proposed system, the system studies the problem of CRS on road networks, which aims to find an optimal route such that it covers a set of query keywords in a given specific order, and the matching distance is minimized. To answer the CRS query, we first propose a greedy clue-based algorithm GCS with no index where the network expansion approach is adapted to greedily select the current best candidates to construct feasible paths.

 Then, we devise an exact algorithm, namely clue-based dynamic programming CDP, to answer the query that enumerates all feasible paths and finally returns the optimal result. To further reduce the computational overhead, the system proposes a branch-and-bound algorithm BAB by applying filter-and-refine paradigm such that only a small portion of vertices are visited, thus improves the search efficiency.

 In order to quickly locate the candidate vertices, we develop AB-tree and PB-tree structures to speed up the tree traversal, as well as a semi dynamic index updating mechanism. Results of empirical studies show that all the proposed algorithms are capable of answering CRS query efficiently, while the BAB algorithm runs much faster, and the index size of PB-tree is much smaller than AB-tree.

Advantages

 Efficient Routing path due to Clue-based Route Navigation to find exact Route path

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

- The Searching technique is fast by Greedy Clue search algorithm

SPATIAL KEYWORD QUERYING

spatial web objects that possess both a geographical location and a textual description are gaining in prevalence, and spatial keyword queries that exploit both location and textual description are gaining in prominence. However, the queries studied so far generally focus on finding individual objects that each satisfy a query rather than finding groups of objects where the objects in a group collectively satisfy a query. We define the problem of retrieving a group of spatial web objects such that the group's keywords cover the query's keywords and such that objects are nearest to the query location and have the lowest inter-object distances. Specifically, we study two variants of this problem, both of which are NP-complete. We devise exact solutions as well as approximate solutions with provable approximation bounds to the problems. We present empirical studies that offer insight into the efficiency and accuracy of the solutions. problem of retrieving a group of spatial objects, each associated with a set of keywords, such that the group covers the query's keywords and has the lowest cost measured by their distance to the query point, and the distances between the objects in the group. We study two particular instances of the problem, both of which are NP-complete. We develop approximation algorithms with provable approximation bounds and exact algorithms to solve the two problems. Results of experimental evaluation offer insight into the efficiency and the accuracy of the approximation algorithms, and the efficiency of the exact algorithms. propose a new exact method for shortest-path distance queries on large-scale networks. Our method precomputes distance labels for vertices by performing a breadth-first search from every vertex. Seemingly too obvious and too inefficient at first glance, the key ingredient introduced here is pruning during breadth-first searches. While we can still answer the correct distance for any pair of vertices from the labels, it surprisingly reduces the search space and sizes of labels. Moreover, we show that we can perform 32 or 64 breadth-first searches simultaneously exploiting bitwise operations. We experimentally demonstrate that the combination of these two techniques is efficient and robust on various kinds of large-scale realworld networks. In particular, our method can handle social networks and web graphs with hundreds of millions of edges, which are two orders of magnitude larger than the limits of previous exact methods, with comparable query time to those of previous methods. A distance query asks the distance between two vertices in a graph. Without doubt, answering distance queries is one of the most fundamental operations on graphs, and it has wide range of applications. For example, on social networks, distance between two users is considered to indicate the closeness, and used in sociallysensitive search to help users to find more related users or contents, or to analyze influential people and communities. On web graphs, distance between web pages is one of indicators of relevance, and used in context-aware search to give higher ranks to web pages more related to the currently

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

visiting web page. Other applications of distance queries include top-k keyword queries on linked data, discovery of optimal pathways between compounds in metabolic networks, and management of resources in computer networks. Of course, we can compute the distance for each query by using a breadth first search (BFS) or Dijkstra's algorithm. However, they take more than a second for large graphs, which is too slow to use as a building block of these applications. In particular, applications such as socially-sensitive search or contextaware search should have low latency since they involve real-time interactions between users, while they need distances between a number of pairs of vertices to rank items for each search query. Therefore, distance queries should be answered much more quickly, say, microseconds. The other extreme approach is to compute distances between all pairs of vertices beforehand and store them in an index. Though we can answer distance queries instantly, this approach is also unacceptable since preprocessing time and index size are quadratic and unrealistically large. Due to the emergence of huge graph data, design of more moderate and practical methods between these two extreme approaches has been attracting strong interest in the database.

CONCLUSION

In this paper, study the problem of CRS on road networks, which aims to find an optimal route such that it covers a set of query keywords in a given specific order, and the matching distance is minimized. To answer the CRS query, first propose a greedy clue-based algorithm GCS with no index where the network expansion approach is adopted to greedily select the current best candidates to construct feasible paths. First, users may prefer a more generic preference model, which combines PoI rating, PoI average menu price, etc, in the query clue. Second, it is of interest to take temporal information into account and further extend the CRS query. Each PoI is assigned with a opening hours time interval [To, Tc], and each clue contains a visiting time t, where the resulting query aims to find a path such that the time interval of each matched PoI covers the visiting time. Third, requiring users to provide exact keyword match is difficult sometimes as they are just providing ―clue‖, which may be inaccurate in nature. Thus, it is of interest to extend our model to support the approximate keyword match. Hence, the matching distance can be modified by incorporating both spatial distance and textual distance together through a linear combination.

References

[1] I. Abraham, D. Delling, A. V. Goldberg, and R. F. Werneck. Hierarchical hub labelings for shortest paths. In ESA, pages 24–35. Springer, 2012.
[2] T. Akiba, Y. Iwata, K.-i. Kawarabayashi, and Y. Kawata. Fast shortestpath distance queries on road networks by pruned highway labeling. In ALENEX, pages 147–154. SIAM, 2014.
[3] T. Akiba, Y. Iwata, and Y. Yoshida. Fast exact shortest-path distance queries on large networks by pruned landmark labeling. In SIGMOD, pages 349–360. ACM, 2013.

[4] T. Akiba, Y. Iwata, and Y. Yoshida. Dynamic and historical shortestpath distance queries on large evolving networks by pruned landmark labeling. In WWW, pages 237–248. ACM, 2014.

[5] J. L. Bentley and J. B. Saxe. Decomposable searching problems i. staticto dynamic transformation. Journal of Algorithms, 1(4):301–358, 1980.

[6] X. Cao, L. Chen, G. Cong, and X. Xiao. Keyword-aware optimal route search. PVLDB, 5(11):1136–1147, 2012.

[7] X. Cao, G. Cong, and C. S. Jensen. Retrieving top-k prestige-based relevant spatial web objects. PVLDB, 2010.

[8] X. Cao, G. Cong, C. S. Jensen, and B. C. Ooi. Collective spatial keyword querying. In SIGMOD, pages 373–384. ACM, 2011.

[9] H. Chen, W.-S. Ku, M.-T. Sun, and R. Zimmermann. The multi-rule partial sequenced route query. In SIGSPATIAL, page 10. ACM, 2008.

[10] L. Chen, G. Cong, C. S. Jensen, and D. Wu. Spatial keyword query processing: an experimental evaluation. PVLDB, 2013.

[11] N. Christofides. Worst-case analysis of a new heuristic for the travelling salesman problem. Technical report, DTIC Document, 1976.

[12] G. Cong, C. S. Jensen, and D. Wu. Efficient retrieval of the top-k most relevant spatial web objects. PVLDB, 2009.

[13] I. De Felipe, V. Hristidis, and N. Rishe. Keyword search on spatial databases. In ICDE, 2008.

[14] E. W. Dijkstra. A note on two problems in connexion with graphs. Numerische mathematik, 1(1):269–271, 1959.

[15] T. Guo, X. Cao, and G. Cong. Efficient algorithms for answering the m-closest keywords query. In SIGMOD, pages 405–418. ACM, 2015.