Optimal Route Search Using Bounded Cost Informative Routes

Parwani Simran[1] , Jadhav Shweta[2] , Chavan Harshala[3]
Department of Computer Engineering
Jaihind College of Engineering, Kuran, Pune.
parwanisimran05@gmail.com[1]
jadhavshweta2015@gmail.com[2]
harshalachavan2015@gmail.com[3]

Prof. Nitesh Jadhav
Department of Computer Engineering
Jaihind College of Engineering, Kuran, Pune.
niteshjadhav5547@gmail.com

***Abstract* -** As travel is taking more significant part in our life, route recommendation service becomes user interested to visit new spots and the new short rout as well as long route with the interesting new places, a big business and attracts many major players in IT industry. Given a user specified origin and destination, a route recommendation service aims to provide users the routes with the best travelling experience according to criteria such as travelling distance, travelling time, etc. However, previous research shows that even the routes recommended by the big-thumb service providers can deviate significantly from the routes travelled by experienced drivers. It means travelers' preferences on route selection are influenced by many latent and dynamic factors that are hard to be modeled exactly with pre-defined formulas. In this work the approach to this challenging problem is with a completely different perspective – leveraging crowds' knowledge to improve the recommendation quality. The widespread location-aware applications produce a vast amount of spatio-textual data that contains both spatial and textual attributes. To make use of this enriched information for users to describe their preferences for travel routes, a Bounded-Cost Informative Route (BCIR) query is proposed to retrieve the routes that are the most textually relevant to the user-specified query keywords subject to a travel cost constraint. BCIR query is particularly helpful for tourists and city explorers to plan their travel routes. The proposed system will show that BCIR query is an NP-hard problem. To answer BCIR query efficiently, the exact solution is explained with effective pruning techniques and two approximate solutions with performance guarantees. Extensive experiments over real data sets demonstrate that the proposed solutions achieve the expected performance.

Index Terms - Route query, Query keywords, Informative routes, Bounded cost.

INTRODUCTION

Optimal Route Search using Bounded-Cost Informative Routes plays a vital role in our daily life. Thanks to the rapid development of GPS technologies and flourish of navigation service providers (e.g., Google Map, Bing Map, Tom-tom), we can now travel to unfamiliar places with much less effort, by simply following the recommended routes. THE constantly increasing volume of spatio-textual data brings abundant information about travel routes. Examples include comments and check-in from Foursquare, geo-tagged photos with textual descriptions from Facebook, geo-tagged posts from twitter, and location-based advertisements from various business promotion platforms. It is appealing to take full advantage of such spatio-textual data to retrieve interesting routes. Data mining is the means of extracting data from a dataset for users to use it in various purposes. The purpose of such data plays a significant role in keyword searching. Searching is a common activity happening in data mining. Searching for spatial objects from spatial database has recently sparked enthusiasm among researchers. This motivated to develop methods to retrieve spatial objects. Spatial objects consist of objects associated with spatial features. In other words, spatial objects involve spatial data along with longitude and latitude of location. Querying such data is called best keyword cover querying. Search is called best keyword cover search. Existing method to such data consider either minimum inter objective distance and keyword search. As a result new methods for best keyword cover search compute nearest neighbour search focus on finding nearest neighbours where keywords and spatial data plays a major impact.

PROBLEM STATEMENTS

1. Problem Statement:

When we are planning any trip, we limit to certain conditions such as location, time, budget, etc. In the proposed system, an efficient keyword aware representative travel route framework is explained that uses knowledge extraction from users historical mobility records and social interactions.

2. Goals & Objectives:

- Accurately find out the different places.
- To reduce the time required for finding rout related information.
- By using maximum number of resources and in minimum cost tourists can visit the interesting places according to their preferences.
- The system provides not only optimal route but also interesting and pleasant route.

LITERATURE SURVEY

For finding the optimized routes, early research focused on finding most efficient routes. For example, Xin Cao et al. [1] used keyword-aware optimal route query also known as KOR, which is used to find the optimal routes that satisfies the specified budget constraints and also it covers set of user specified keywords. The approximation algorithms like OSScaling, BucketBound and greedy algorithm are used.

Daniele Quercia et al. [2] suggests the optimized paths which are not only shortest but also pleasant and happy by using data provided by online services like Flickr or Four-square. Yifeng Zeng et al. [3] introduced the keyword coverage function and define the optimal route search for keyword coverage (ORS-KC) problem, which is used to find optimal route such that it can optimally satisfy the users preferences. For solving ORS-KC they used variants of A* algorithm which is based on heuristic function.

Yu Zheng et al. [4] used the GPS trajectories generated by multiple users so that a person can able to find some places that attract them from other people' travel routes, hence, plan an interesting and efficient journey based on multiple users experience. Mao Ye et al. [5] aimed to provide a point-of-interest (POI) recommendation service for the rapid growing location-based social networks (LBSNs), e.g., Four-square, Flickr, etc.

Muhammad Aamir Saleem et al. [6] studied how the users of LBSN navigate between locations and based on that information they select the most influential location. For this purpose they described two models namely absolute influential model and relative influential model and also proposed an Oracle data which can be used to find the top-k influential locations.

Kunjie Chen et al. [7] used Aggregate-keyword routing query, which focuses on multiple query points in a spatial keyword search. They also designed approximation algorithm called as Centre Based Assignment which tries to find aggregate points and task points near the centre of the query point and then uses the greed approach to assign those task points.

PROPOSED SYSTEM

In proposed system Given a user specified origin and destination, a route recommendation service aims to provide users the routes with the best travelling experience according to criteria such as travelling distance, travelling time, etc. However, previous research shows that even the routes recommended by the big-thumb service providers can deviate significantly from the routes travelled by experienced drivers. In this system, Crowd Planner – a novel crowd-based route recommendation system has been developed, which requests human workers to evaluate candidates routes recommended by different sources and methods, and determine the best route based on the feedbacks of these workers. Finally we apply Sentiment Analysis for to display positive reviews only regarding that place.

ALGORITHMS

In the proposed system K-Nearest Neighbors (KNN) algorithm is used for finding the nearest interesting places in between our route in section 1, and also the comment analyzer that is sentimental analysis in section 2, which is used for analyzing the users positive reviews.

1. KNN (K-NEAREST NEIGHBORS)

1. Determine parameter k = number of nearest neighbour.
2. Calculate the distance between the query instance and all the training samples.
3. Sort the distance and determine nearest neighbour based on th k th minimum distance.
4. Gather the category y of the nearest neighbour.
5. Use simple majority of the category of nearest neighbour as the prediction value of query instance.

K-nearest neighbor (Knn) algorithm pseudocode:

Let (X_i, C_i) where i = 1, 2……., n be data points. X_i denotes feature values & C_i denotes labels

For X_i for each i.

Assuming the number of classes as 'c'

$_{Ci} \in \{1, 2, 3, \ldots\ldots, c\}$ for all values of i

Let x be a point for which label is not known, and we would like to find the label class using k-nearest neighbor algorithms.

1. Calculate "$d(x, x_i)$" i =1, 2, …..n; where d denotes the Euclidean distance between the points.
2. Arrange the calculated n Euclidean distances in non-decreasing order.
3. Let k be a +ve integer, take the first k distances from this sorted list.
4. Find those k-points corresponding to these k-distances.
5. Let k_i denotes the number of points belonging to the i^{th} class among k points i.e. $k \geq 0$
6. If $k_i > k_j \; \forall \; i \neq j$ then put x in class i.

2. SENTIMENT ANALYSIS

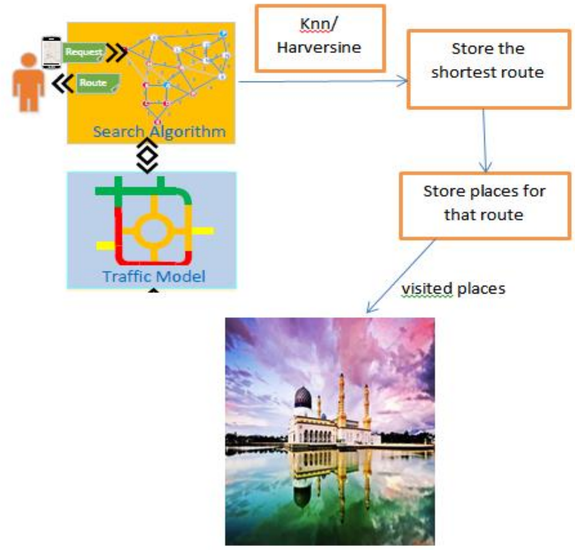
- Get Terms - Reduce each review to the list of words
- Filtering - Remove unnecessary words that will not add value for sentiment analysis - is, among, but, and, it, that
- Base Word - Convert all inflections to their root word
  - fry, fries, fried -> fry
  - going, go , went, goes -> go
  - movies, movie -> movie
- Make Features - Use the words thus extracted from a review as features to indicate the positiveness or negativeness of that review
- Classifier - Train a classifier to predict positivity

Comment Analyzer

Input: Preprocessed comment
Output: Comment categorized as positive negative or neutral.
overallPol: Polarity of the whole comment.
sentPol: Polarity of the sentence within the comment.
POS: Part of Speech of the word.

```
BEGIN
1. overallPol = 0
2. For each sentence in comment
3. f
4. sentPol = 0
5. For each word in sentence
6. f
7. If Polarity[word] == Positive
8. f
9. If POS[word] == Adjective
10. sentPol = sentPol + 4
11. Else if POS[word] == Adverb
12. sentPol = sentPol + 3
13. Else if POS[word] == Verb
14. sentPol = sentPol + 2
15. Else:
16. sentPol = sentPol + 1
17. g
18. Else if Polarity[word] == Negative
19. f
20. If POS[word] == Adjective
21. sentPol = sentPol 4
22. Else if POS[word] == Adverb
23. sentPol = sentPol 3
24. Else if POS[word] == Verb
25. sentPol = sentPol 2
26. Else
27. sentPol = sentPol 1
28. g
29. Else if word is a Negation word
30. sentPol = - sentPol
31. g
32. overallPol = overallPol + sentPol
33. g
34. If overallPol > 0
35. Return Positive Comment
36. Else if overallPol < 0
37. Return Negative Comment
38. Else
39. Return Neutral Comment
```

SYSTEM ARCHITECTURE



HARDWARE AND SOFTWARE REQUIREMENTS

Hardware Requirements Specification:

There should be required devices to interact with software.

- System : Dual core /quad core processor.
- Hard Disk : 40 GB.
- Ram : 1GB.

Software Requirements Specification:

- Operating system : Windows 7 and above.
- Coding Language : JAVA/J2EE.
- IDE : Android studio/Eclipse.

CONCLUSION AND FUTURE SCOPE

In this paper, the BCIR query to retrieve the route that is most textually relevant to the user-specified query keywords within a travel cost budget. To efficiently process BCIR queries, in proposed system an exact solution with effective pruning techniques and two approximate solutions regarding the response time and the quality of results, respectively. As demonstrated via extensive experiments, the proposed solutions achieve satisfying performance over different data sets. BCIR provides a new type of route query that can be applied in various applications ranging from route planning to location-aware recommendation.

ACKNOWLEDGMENT

We would prefer to give thanks the researchers likewise publishers for creating their resources available. We are conjointly grateful to guide, reviewer for their valuable suggestions and also thank the college authorities for providing the required infrastructure and support.

REFERENCES

[1] X. Cao, L. Chen, G. Cong, and X. Xiao, "Keyword-aware optimal route search," PVLDB, vol. 5, no. 11, pp. 1136–1147, 2012.

[2] D. Quercia, R. Schifanella, and L. M. Aiello, "The shortest path to happiness: recommending beautiful, quiet, and happy routes in the city," in HT, 2014, pp. 116–125.

[3] Y. Zeng, X. Chen, X. Cao, S. Qin, M. Cavazza, and Y. Xiang, "Optimal route search with the coverage of users' preferences," in IJCAI, 2015, pp. 2118–2124.

[4] Y. Zheng, L. Zhang, X. Xie, and W.-Y. Ma, "Mining interesting locations and travel sequences from GPS trajectories", in Proc. 18th Int. Conf. World Wide Web, 2009, pp. 791800.

[5] M. Ye, P. Yin, W.-C. Lee, and D.-L. Lee, Exploiting geographical influence for collaborative point-of-interest recommendation, in Proc. 34th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2011, pp. 325334.

[6] Muhammad Aamir Saleem, Rohit Kumar, Toon Calders, Xike Xie, Torben Bach Pedersen, "Location influential in Location-based social networks, " in WSDM, 2017.

[7] Kunjie Chen, Weiwei Sun, Chuanchuan Tu, Chunan Chen, Yan Huang, "Aggregate keyword routing in spatial database," in ACM SIGSPATIAL, 2012.

[8] V. S. Tseng, E. H.-C. Lu, and C.-H. Huang, Mining temporal mobile sequential patterns in location-based service environments, in Proc. Int. Conf. Parallel Distrib. Syst., 2007, pp. 18.

[9] W. T. Hsu, Y. T.Wen, L. Y.Wei, andW. C. Peng, Skyline trave l routes: Exploring skyline for trip planning, in Proc. IEEE 15th Int. Conf. Mobile Data Manage., 2014, pp. 3136.

[10] R. Geisberger, P. Sanders, D. Schultes, and D. Delling, "Contraction hierarchies: Faster and simpler hierarchical routing in road networks," in WEA, 2008, pp. 319–333.

[11] D. Deng, C. Shahabi, and U. Demiryurek, "Maximizing the number of worker's self-selected tasks in spatial crowdsourcing," in SIGSPATIAL, 2013, pp. 314–323.

[12] W. Souffriau, P. Vansteenwegen, G. V. Berghe, and D. V. Oudheusden, "The planning of cycle trips in province of eat flanders," Omega, vol. 39, no. 2, pp. 209–213, 2011.