EGR-470

OPTIMIZING SEARCH IN LOCATION DATA

Author(s): Lalitha Sangamithra, Krishna Chaitanya, Sairam Abhishikth

Abstract

Our project addresses the challenge of optimizing location-based data queries, critical for applications like DoorDash, Uber, and Google. The benchmark k-Nearest Neighbor algorithm, while effective, incurs significant computational costs. To enhance efficiency, we propose a k-means clustering approach that minimizes the number of comparisons required for nearest point identification. By grouping data into k-clusters, our method streamlines the search process, reducing computational load and improving query times. This approach offers a practical solution for accurate and swift point-to-point matching in GPS and road network scenarios.

Introduction

Inherent errors in GPS data collection necessitate strategies to optimize the identification of nearest points on a map. The k-NN algorithm poses computational challenges due to its exhaustive comparison of all data points. In response, this paper proposes a novel approach leveraging the k-means clustering technique to streamline the process.

Research Question(s)

1. Compare the benchmark k-NN algorithm in terms of computational efficiency for location-based data queries?

2. What impact does the clustering of data points into k-groups have on the overall search time for identifying nearest points in location-based data.

3. Can the proposed method effectively mitigate the challenges associated with GPS errors during data collection, leading to improved accuracy in identifying nearest points on a map?

Materials and Methods

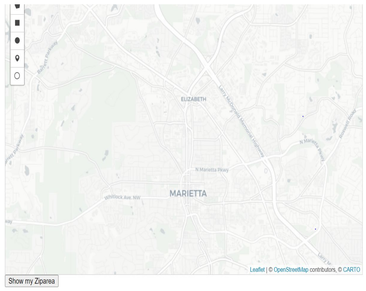
The problem statement states that our objective is to identify the nearest locations by optimizing the number of comparisons.
The process of clustering involves grouping the data points into k-clusters. To minimize the search space, we first choose the closest cluster based on distance, and then we search for the required point k inside that cluster.
This point-to-point matching technique is mostly used to determine the distance between GPS points and road network points by matching the GPS positioning point to the nearest point as the correction result.

Results

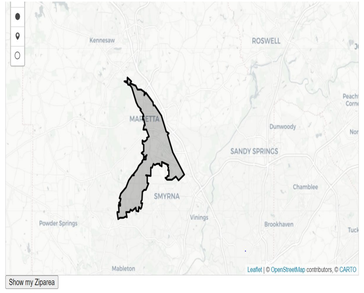
```
Given Point: [33.952301, -84.549049]
Algorithm: kNN
       It took 0.01202 seconds to predict
       The predicted zipcode: 30060
Algorithm: kMeans
       It took 0.00209 seconds to predict
       The predicted zipcode: 30060
$
```

```
Algorithm: SVM
       It took 17.61626 seconds to predict
       Accuracy: 0.9564036807965002
Algorithm: Naive Bayes
       It took 0.008 seconds to predict
       Accuracy: 0.972846583195052
Algorithm: KNN
       It took 0.28129 seconds to predict
       Accuracy: 0.975863629506712 9
Algorithm: kMeans
       It took 0.001 seconds to predict
       The predicted clusters: [0 4 0 2 1]
```

Our main Goal here is to predict the nearest zip code based on our latitude and longitude points without searching the whole data set. Here we have predicted nearest zip code 30060 using our latitude and longitude point [33.952301, -84.549049]



Your zipcode appears here



Your Zipcode is 30060

Project Future Development

- We have finished the coding part and implemented it.
- In the next coming weeks, we will refine it further to work on solving the problems encountered and run simulations on various test cases. We will also develop the GUI.
- Then we will evaluate the final time and space efficiency for our algorithm.
- The running time of the benchmark and the proposed algorithm will be compared in a tabular form.
- Prepare the final presentation and report

Conclusions

Through our project, we have successfully detected or searched for the desired area of zip code. The running time of our chosen algorithm is slightly efficient when compared to the benchmark algorithm. Our project also uses the simplest technique to produce the results and has made search easier. Running time of proposed algorithm is better than benchmark algorithm. This project helped us learn how to evaluate a problem, research and several types of algorithms that can help us evaluate this kind of problem.

Acknowledgments

List of people who helped to work in this project:
Mahmud Karakaya
Organization: Kennesaw State University

Contact Information

lmantha@students.Kennesaw.edu
svadlam1@students.Kennesaw.edu
knarapar@students.Kennesaw.edu

References

[1] T. Liu, Z. Chen, C. Chen, Z. Duan and B. Zhao, "A Dynamic K-nearest Neighbor Map Matching Method Combined with Neural Network," 2019 IEEE Intelligent Transportation Systems Conference (ITSC), 2019, pp. 3573-3578, doi:10.1109/ITSC.2019.8916909. A Dynamic K-nearest Neighbor Map Matching Method Combined with Neural Network | IEEE Conference Publication | IEEE Xplore

[2] k-Nearest Neighbors on Road Networks: A Journey in Experimentation and In-Memory Implementation Technical Report Tenindra Abeywickrama, Muhammad Aamir Cheema, David Taniar Faculty of Information Technology, Monash University, Australia . {tenindra.abeywickrama,aamir.cheema,david.taniar}@monash.edu www.vldb.org/pvldb/vol9/p492-abeywickrama.pdf

KENNESAW STATE UNIVERSITY
COLLEGE OF COMPUTING AND SOFTWARE ENGINEERING