YIKE2

FYT Proposal

Differentially Private Range Counting in General Metric Spaces

by

Shr En Chen

YIKE2

Advised by

Prof. Ke Yi

Submitted in partial fulfillment of the requirements for COMP 4981H

in the

Department of Computer Science

The Hong Kong University of Science and Technology

2025-26

Date of submission: Sep 11 2025

Table of Contents

1 Introduction

1.1 Overview

1.1.1 Differential Privacy (DP)

Differential privacy (DP) has been the standard for privacy-preserving database query analysis [1]. Under differential privacy, the distribution of the output from a query will be similar if the two databases differ only in one data point. Hence, the outsider could not infer whether an individual data is inside the database or not. However, privacy is meaningless if the output has large error, which means the utility of the output is useless.

1.1.2 (Approximated) Range Counting

Range counting is a fundamental query that counts the number of data points inside the given range. Many downstream applications such as nearest neighbors query and KNN-classifiers. One may relax the problem to approximate range counting that counts the total number of points in a specified error tolerant range. This is practical in real world since data by its nature contain noise.

1.1.3 Spatial Decomposition

Spatial decomposition is the major approach for range counting. It is usually realized through tree structure. The split at each node represents a decomposition of the original space and store some information (e.g. the number of data points) at the node. To answer a range counting, it only needs to traverse the relevant nodes and sum the number of points inside. Hence, the runtime is bounded by the height of the tree.

There are many efficient ranges-counting hierarchical data structure for range counting for data in Euclidean spaces with theoretical guarantees. They take advantage of $\mathbb{R}^d$ to separate the data into two parts along axis so that the range of space or the number of points in each part decreases by a constant factor and results in desirable tree depth. However, in many common data space, there are no axis and total order of data, but are only equipped with a metric. Taking strings as an example, edit distance is a natural metric but it is hard to have a meaningful order of all strings, so the aforementioned methods do not apply. Metric trees are the main approach to the problem [2], which decompose the space based on the property of metric only. However, it's impossible to have a data structure that support all metric space. For example, in *discrete metric space*, the distance between two points is 1 only if they are different, and 0 otherwise. The distance gives no information about the data, so some works [3] [4] introduce an abstract notion of dimension of make the discussion feasible.

To achieve differential privacy when using space decomposition, there are two modifications needed [5].

1. When summing up the amount of data at each node, we need to add appropriate noise
2. If the spatial decomposition structure itself is dependent on data, it is necessary to include randomness during construction.

There are works [5] [6] adapting hierarchical structure for Euclidean to become differentially private, which mostly include private median operation. However, there are, by far as I know, limited to no work on augmenting metric tree structure to differential private version. Therefore, it will be the main purpose of this thesis.

1.2 Objectives

This final year thesis will focus on range-counting methods for general metric spaces and adapt them to become differential private.

1.2.1 DP data structure for (approximated) range-counting

The first objective is converting existing hierarchical structure supporting range counting to differentially private ones.

1.2.2 Empirical Experiments and Evaluation

The second objective is to implement the proposed structure and conduct experiments on dataset from different metric spaces.

1.2.3 Theoretical Analysis

Beyond experiments, the third objective is to give theoretical guarantees on privacy, accuracy, and runtime of the proposed data structures.

1.3 Literature Survey

The mainstream approaches to (approximated) range-counting are hierarchical spatial decomposition that recursively separates the data space into parts, create nodes to store information (e.g. the number of data points) of each subspace. This results in a tree structure. To do range-counting, one starts from the root and traverse down the relevant nodes and sum up the counts. The key lies in how to properly separate the space so that it can cover any query range, but the tree itself is not too deep to be inefficient. There are two types of spatial decompositions [5]:

- **Data-independent spatial decomposition**

The data structure is predefined, regardless of where data instances are. A familiar example is Quadtree which equally splits the space, and hence the total range of space is decreased by a constant factor. This may result in $O(log\ u)$ tree depth, where u is the size of the original range, if the tree is balanced.

- **Data-dependent spatial decomposition**

This kind of data structure take input into accounts, and the structure depends on input. A well-known example is kd-trees which recursively split view lines passing through the median data. The number of data points will decrease by a constant as we traverse down the tree. This may result in $O(\log n)$ tree depth, where n is the number of data points, if the tree is balanced.

1.3.1 Spatial Decompositions in Euclidean Spaces

The well-known quad trees and kd-trees are designed for Euclidean spaces decomposition. Some approach such as Balanced Aspect Ratio Tree (BAR-tree) [7] combine kd-tree and Octrees to balance the tree height. An optimal approach for approximated range counting is Balanced Box-Decomposition tree (BBD-tree) [8].

1.3.2 Spatial Decompositions in General Metric Spaces

The common approach for spatial decomposition in general metric space is metric trees [2]. The central idea is to use the triangular inequality property in metric space to prune out the search space. Previous works adopt the intuition to decompose spaces through simple balls and clusters [9] [10] [11], while the more recent works adopt more complicated structures like covering nets [3] [4].

Balls and Clusters

The simplest example of separating space through balls is vp-tree [9]. It separate the spaces recursively by selecting a vantage point p and choosing appropriate radius r so that the ball $B(p;r)$ contains half of the points. However, this approach is asymmetric which make pruning inefficient. A common, symmetric approach is through clustering. The works pick $k \geq 2$ representatives and decompose the spaces by assigning other points to the nearest representatives [10] [11]. The theoretical results are weak, relying heavily on experimental results.

Covering Nets

Another branch of approach is through the idea of covering nets [3] [4]. It is a hierarchical structure as well. At each level, the nodes are far enough from each other and the balls centered at them together cover the nodes in the next level. It turns out to be efficiently prune out search spaces during range counting query.

Since it is impossible to handle all kinds of general metric space as discussed in section 1.1.3, many works introduced another abstract notion of dimension, such as doubling dimension, to perform analysis. The theoretical guaranteed becomes meaningful when the dimension of the real-world data is small enough.

1.3.3 Spatial Decompositions under Differential Privacy

There is a general framework in [5] to calibrate noise on both data-independent and data-dependent hierarchical structure. It gives each node a non-uniform privacy budge with careful analysis and post-processes the noises to further increase accuracy.

Since the works in spatial decomposition in Euclidean space takes advantage of the well-defined notion of median, they can be easily adapted to be differentially private by adding noise to counts and injecting randomness in structure construction. However, in general metric space, there is no natural notion of median. It leaves privatizing metric trees a relatively unexplored problem.

2 Methodology

2.1 Design and Implementations

2.1.1 Preliminaries

Differential Privacy and Basic Mechanisms

Let $D \sim D'$ denote two neighboring databases, where one contains one more point than the other.

Definition 1 (Differential Privacy [1]) For $\epsilon > 0, \delta > 0$, a randomized algorithm M is (ϵ, δ)-differentially private if for any neighboring databases $D \sim D'$ and any output range $S \subseteq Range(M)$, $\Pr[M(D) \in S] \leq e^{\epsilon} \cdot \Pr[M(D') \in S] + \delta$

The case when $\delta = 0$, we say the mechanism satisfies ϵ-differential privacy.

Definition 2 (Sensitivity [1]) For a numeric query f, the maximum difference between the output of two neighboring databases is the *sensitivity of* f, denoted as $\Delta f = \max_{\{D \sim D'\}} |f(D) - f(D')|$.

The following lemma is the most common way to achieve differentially private numeric query.

Lemma 1 (Laplace Mechanism [1]) For a numeric query f over a database D, an $(\epsilon, 0)$-differentially private mechanism is to output $f(P) + X$, where $X \sim Lap(\Delta f / \epsilon)$

An differentially private algorithm is usually a sequential composition of differentially private operations. The following lemma gives us a theoretical guarantee under composition.

Lemma 2 (Compositions [11]) Let $M_1, \ldots, M_t$ be t algorithms such that M_i satisfies ϵ_i-differential privacy, $1 \leq i \leq t$. Then their sequential composition $M_t \circ M_{\{t-1\}} \circ \cdots \circ M_1$ satisfies ϵ-differential privacy, for $\epsilon = \sum_{i=1}^{t} \epsilon_i$

2.1.2 Adapting Hierarchical Structure

For hierarchical spatial decomposition structures in Euclidean spaces, we can adapt it to be differentially private with method [5] [6] . For example, the key operation in construction of k-d tree is selecting a median along an axis and splitting the data into two parts. It is necessary to privatized median, otherwise, the structure of k-d tree itself will reveal information of true data. Cormode et al. [5] gives an algorithm for selecting median under differential privacy. The theoretical guarantee is achieved by viewing every range counting query (traversal on tree) of k-d tree as a sequence of median query on the database. Hence, the composition lemma applies.

My idea to construct differentially private metric tree is similar. As we discussed in section 1.1.3, there are two adjustments needed. The first adjustment is to add noise to when each node returns the range count of the range that it represented. The result from [5] can be directly applied. The second adjustment is to construct the tree structure under differential privacy. Although we can naively add noise when ever need in construction, the resulted structure may not be ideal. Due to the noise, the data structure may not equip the original, ideal properties as the original version, so the theoretical guarantees on privacy, accuracy and, runtime may all not exist anymore. However, it is still possible to give probabilistic bounds like Huang et. al did in [6].

2.2 Experiments and Evaluation

The proposed algorithms will be evaluated on two aspects, accuracy and runtime. I will test the algorithms on several types of datasets ranging from images, genetics, and texts from UCI Machine Learning (https://kdd.ics.uci.edu/summary.data.type.html).

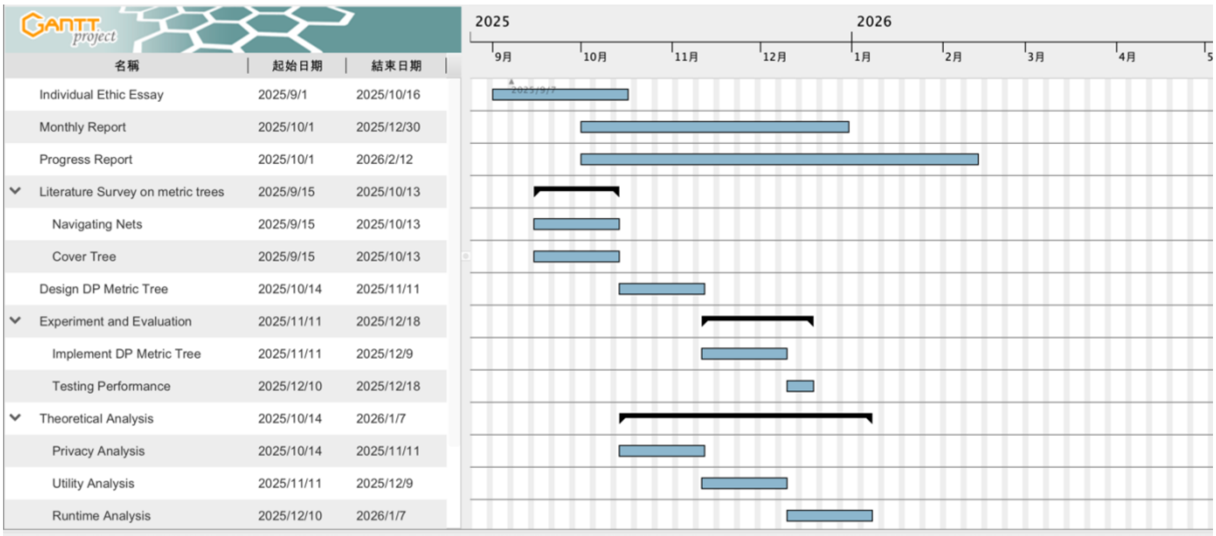
2.3 Theoretical Analysis

The theoretical analysis is quite non-trivial, so I am not sure how it will go. I better first fully understand the analysis in [3] [4].

3 Project Planning

There are three main objectives in this thesis project. Since my current understanding of existing metric trees is not thorough enough, it is no clear how to convert them to be differentially private. I decided to spend a month to dive deeper in articles including but not limited to navigating nets and cover trees. Then I will try to adapt them to be differentially private. The design and analysis of privacy go hand by hand. After having a desirable DP metric tree, I will conduct experiments to evaluate its runtime and the accuracy of the output. While conducting experiments, I will start researching on utility analysis and runtime analysis.

The schedule is summarized in the GAANT chart below.



名稱	起始日期	結束日期
Individual Ethic Essay	2025/9/1	2025/10/16
Monthly Report	2025/10/1	2025/12/30
Progress Report	2025/10/1	2026/2/12
Literature Survey on metric trees	2025/9/15	2025/10/13
Navigating Nets	2025/9/15	2025/10/13
Cover Tree	2025/9/15	2025/10/13
Design DP Metric Tree	2025/10/14	2025/11/11
Experiment and Evaluation	2025/11/11	2025/12/18
Implement DP Metric Tree	2025/11/11	2025/12/9
Testing Performance	2025/12/10	2025/12/18
Theoretical Analysis	2025/10/14	2026/1/7
Privacy Analysis	2025/10/14	2025/11/11
Utility Analysis	2025/11/11	2025/12/9
Runtime Analysis	2025/12/10	2026/1/7

4 Required Hardware & Software

The algorithms in my plan may be simple enough to implement and run entirely on my laptop. The code will be written in python with libraries *random, numpy, scipy.*

5 References

[1] C.Dwork, F.McSherry, K.Nissim, and A.Smith, "Calibrating Noise to Sensitivit in Private Data Analysis," *Theory of Crptography,* pp. 265-284, 2006.

[2] J. Uhlmann, "Satisfying general proximity / similarity queries with metric trees," *Information Processing Letters,* pp. 175-9, Nov 1991.

[3] A. Beygelzimer, S. M. Kakade, and J. Langford, "Cover trees for nearest neighbor," *International Conference on Machine Learning,* Jun 2006.

[4] R. Krauthgamer and J. R. Lee, "Navigating nets: simple algorithms for proximity search," pp. 798-807, Jan 2004.

[5] G. Cormode, C. M. Procopiuc, D. Srivastava, E. Shen, and T. Yu, "Differentially Private Spatial Decompositions," *International Conference on Data Engineering,* Apr 2012.

[6] Z. Huang and K. Yi, "Approximate Range Counting under Differential Privacy," *International Symposium on Computational Geometry (SoCG 2021),* Jun 2021.

[7] C.A. Duncan, M. T. Goodrich, and S. Kobourove, "Balanced Aspect Ratio Trees: Combining the Advantages of k-d Trees and Octrees," *Journal of Algorithms,* vol. 38, pp. 303-333, Jan 2001.

[8] S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman, and A. Y. Wu, "An Optimla Algorithm for Approximate Nearest Neighbor Searching Fixed Dimensions," *Journal of the ACM,* pp. 891-923, 1998.

[9] P. N. Yianilos, "Data structures and algorithms for nearest neighbor search in general metric spaces," *ACM-SIAM Symposium on Discrete Algorithms,* pp. 311-321, 1993.

[10] S. Brin, "Near Neighbor Search in Large Metric Spaces," *Very Large Data Bases,* pp. 574-584, Sep 1995.

[11] F. McSherry and I. Mironov, "Differentially private recommender systems,," *Knowledge Discovery and Data Mining,* pp. 627-636, Jun 2009.

6 Appendix A: Meeting Minutes

Here are the meetings summary with advisor.

1. The 1st meeting

Date: Apr 25 2025

Time: 4:00 pm

Place: Zoom

Summary: Prof. Yi gives two main branches of research, differential privacy and reservoir sampling.

2. The 2nd meeting

Date: Jun 17 2025

Time: 4:00 pm

Place: Zoom

Summary: We narrowed down the topic to geometric privacy on strings under edit distance.

3. The 3rd meeting

Date: Sep 1 2025

Time: 10:00 am

Place: CYT 3002

Summary: After exploration on edit distance, we think edit distance does not have many useful properties. We redirect our interest to range counting on general metric spaces with hierarchical data structures.