Trường ĐH Bách Khoa Tp.HCM
Khoa Khoa Học và Kỹ Thuật Máy Tính

PPDM: Geospatial privacy-preserving data mining of social media

GV: PGS.TS ĐẶNG TRẦN KHÁNH

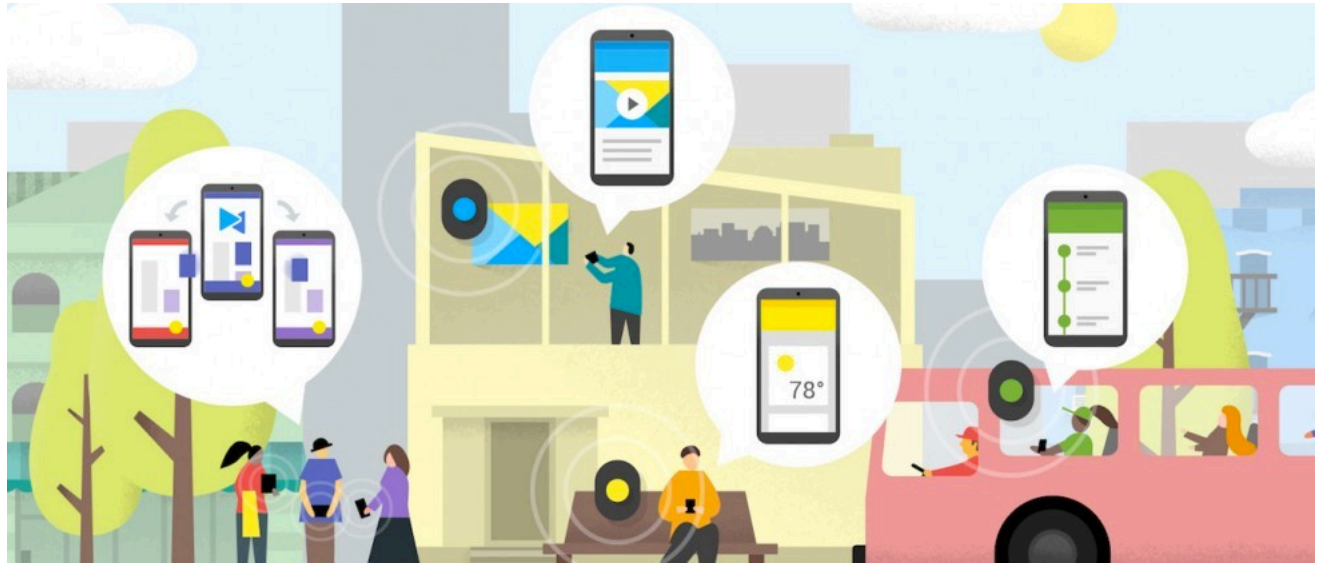
Chu Xuân Tình -1870583
Nguyễn Đức Huy -1870567

Outline

- Introduction
- Differential Privacy
- Differentially Tree Spatial Decomposition
- DBSCAN
- Conclusion

Introduction

- What is Location Privacy
- Basic Techniques
  - Private Information Retrieval
  - Probabilistic Approach
    - Stationary
    - Temporal



What is Location Privacy

- Location Based Services(LBS)
  - Google, zalo, facebook, Instagram, Twitter, Yelp
  - Restaurant check-in, finding the nearest gas station, navigation, tourist city guide, ...
- Location Sharing
  - Find Friends, Find my iphone, ...
- Risks?
  - Give your location data for the service
  - Give your location traces to Google, Apple or other Service providers
  - Enable malicious apps to know your locations
  - Locations may be leaked to other attackers through network

Features of Location Privacy

- Vs Standard Differential Privacy
  - Differential Privacy: the outputs are similar whether a user opts in or out
  - For Location base services, only one user
- Data Type
  - Standard Differential Privacy: tuples in Database
  - Location Privacy: place, user location
- Location data is only two-dimensional
  - Or at most three-dimensional

Techniques

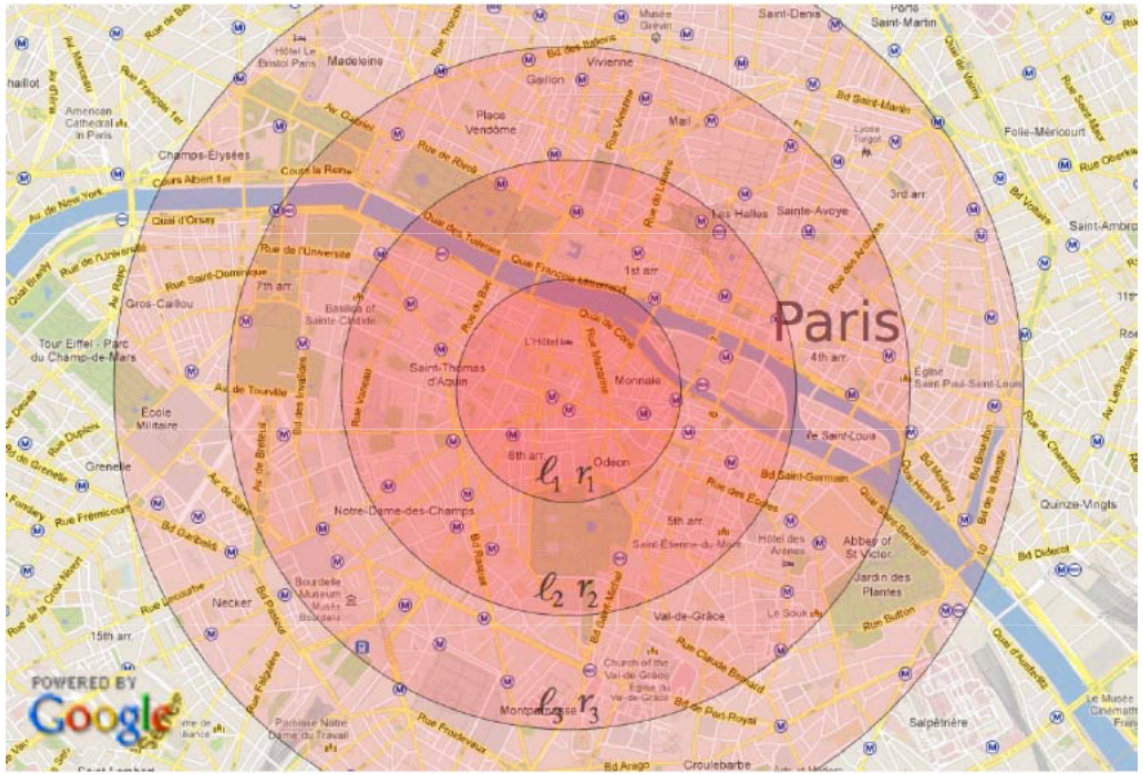
- Encryption-based Techniques
  - ✓ Private Information Retrieval Techniques
- Probabilistic Techniques
  - ✓ Location obfuscation, location cloaking
  - ✓ Location generalization

Probabilistic Techniques

- We develop a mechanism to achieve **geo-indistinguishability** by **perturbating** the user's location x.
- The inspiration comes from one of the most popular approaches for **differential privacy**, **namely the Laplacian noise**.
- We adopt a specific planar version of the Laplace distribution, allowing to draw points in a *geo-indistinguishable* way

Probabilistic Techniques

- Spatial Cloaking/Location Generalization
  - – Instead of sending the exact location to the service providers, a user can send a “general area”.



Probabilistic Techniques

❖ Location Obfuscation

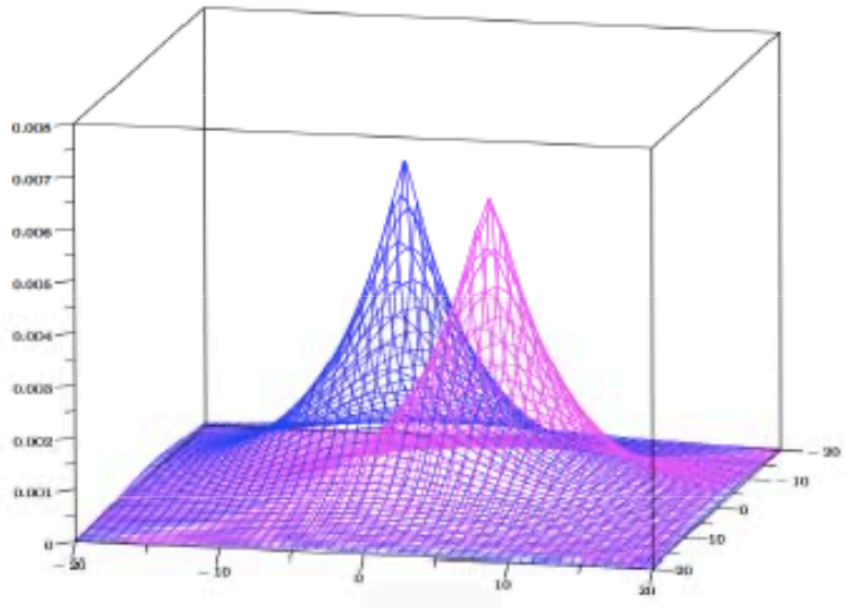
- Instead of sending the exact location to the service providers, a user can send a “noisy” location.
- Essentially, similar to spatial cloaking.
  - With the “general area”, a point can be randomly chosen to represent the “noisy” location.
  - The posterior probability of the “noisy” location will be the same as the “general area”. Can you prove it?

Probabilistic Techniques

- Privacy Guarantee
  - Uniform distribution in a circle
  - Uniform distribution in a polygon
  - Laplace distribution
  - Other distributions: 2D Gaussian distribution
- The trade-off between utility and privacy
  - What is the expected distance between the noisy location and the real location?
  - How much extra information does the noisy location give to attackers?
  - Can you derive the above distance function and the privacy function?

Geo-indistinguishability

- Geo-indistinguishability
  - ✓ A “differentially private” cloaking method
  - ✓ Based on the 2D Laplace distribution
  - ✓ Randomly draw a point from the distribution

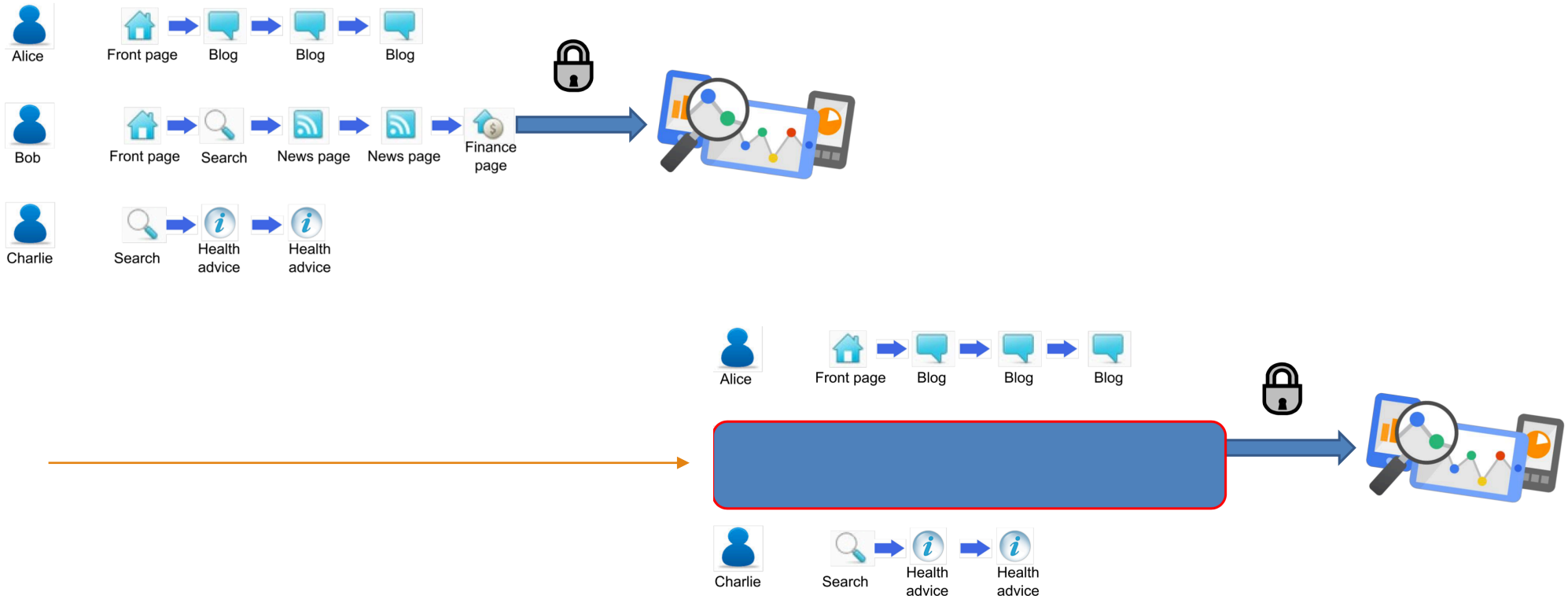


Geo-indistinguishability

- Definition
  - ✓ Pr(z|x)<=e^{epsilon}*Pr(z|x')
  - ✓ Where x and x' are any two locations in a circle with a radius r, z is the noisy location
- Features
  - ✓ Location data: x and x' are two points on a map
  - ✓ Neighboring databases: any points in the circle
  - ✓ Protection: indistinguishability in the circle

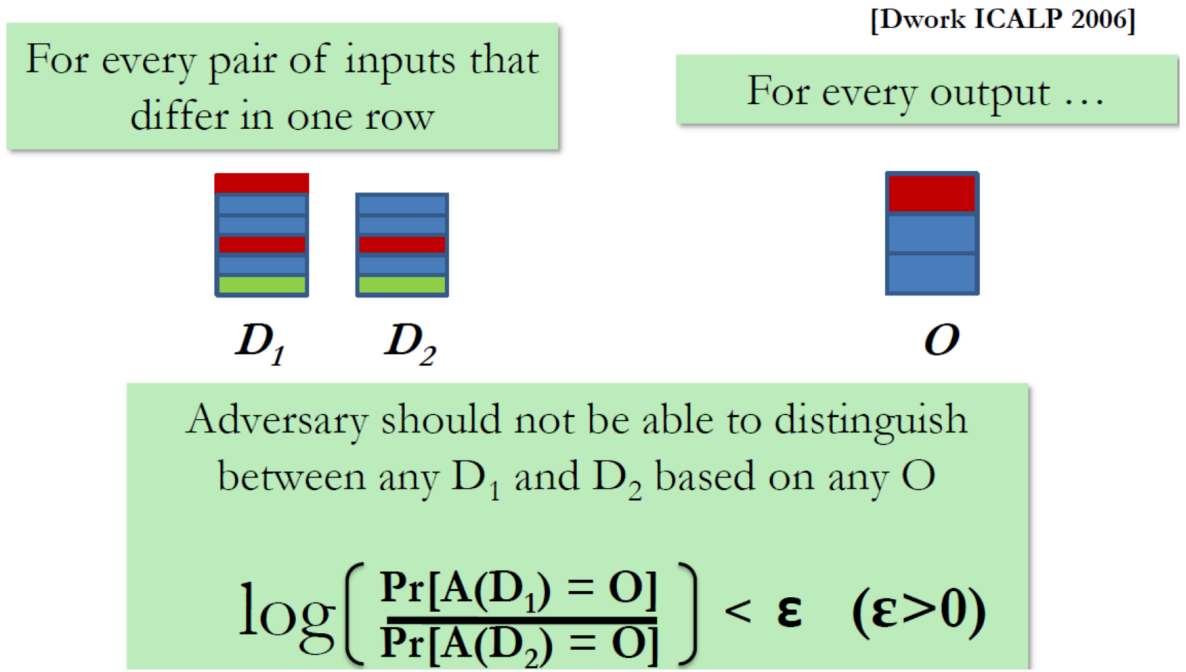
Differential Privacy

- Statistical outcome is indistinguishable regardless whether a particular user (record) is included in the data



Differential Privacy

- Differential privacy was introduced by Dwork et al. (2006). It ensures that useful information can be inquired and mined from a statistical database comprised of individually identifying information, while protecting a given individual's privacy.



Laplace Mechanism

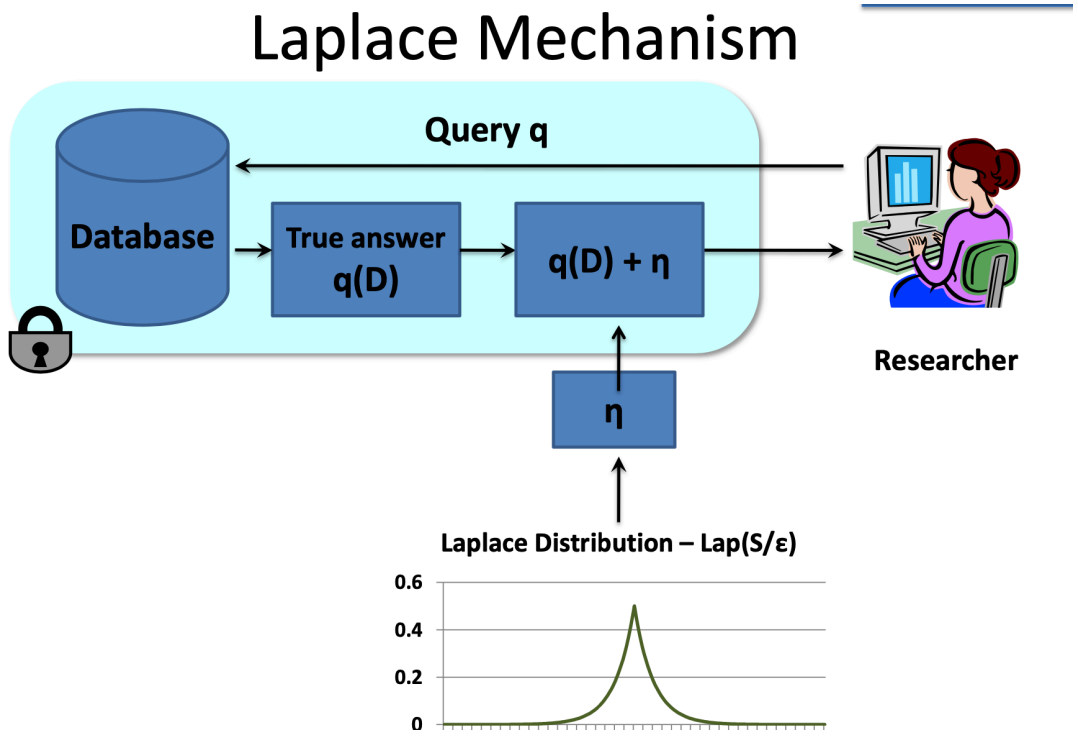
- It provides privacy guarantees as to whether or not a single element is present inside a database or not without explicitly identifying the individual. Several efforts have explored how to apply differential privacy to protect location privacy.

- One example is to support Geo- indistinguishability (Andre´s et al. 2013) using a disturbance technique, whereby a Laplace distribution

- Why use Laplace ?
  - When the possible locations of the users are modeled by a continuous region, the Laplacian was formally proved to provide the minimal noise required to satisfy geo-indistinguishability on this region
  - The generation of Laplace noise can be done efficiently and at low-cost using an analytic expression, so the mechanism can be implemented easily even in a computationally limited device such as a smart phone

Laplace Mechanism

- They explored three ways to add noise: adding global noise to the whole trace; adding noise to each point (x, y) independently and adding noise to each x- and y- coordinate independently.



Laplace Mechanism

- A mechanism for the continuous plane

The idea is that When ever the actual location is x0 $\in R^2$, we report, instead, apoint x $\in R^2$ generated randomly according to the noise function.

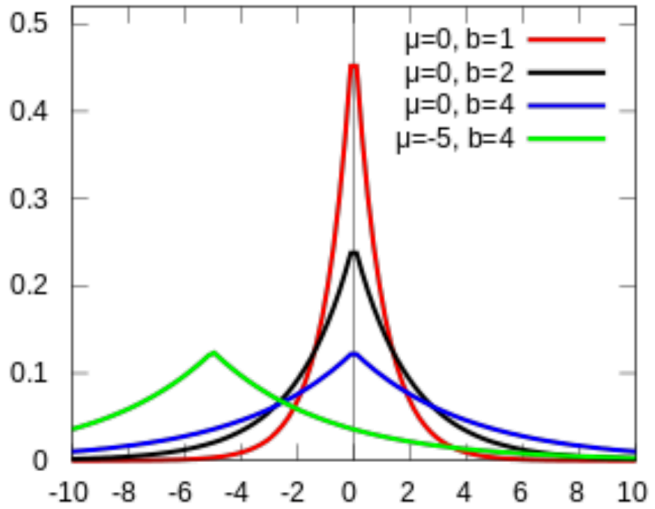
The latter needs to be such that the probabilities of reporting a point in a certain (infinitesimal) area around x, when the actual locations are x0 and x'0 respectively, differs at most by a multiplicative factor $e^{-\varepsilon d(x0,x'0)}$

"Geo-Indistinguishability: Differential Privacy for Location-Based Systems"

Laplace Distribution

- Intuitively, this property is achieved if the noise function is such that the probability of generating a point in the area around x decreases exponentially with the distance from the actual location x0. In a linear space this is exactly the behavior of the Laplace distribution, whose probability density function (PDF) is:

  - PDF: $f(x \mid \mu, b) = \frac{1}{2b} \exp\left(-\frac{|x-\mu|}{b}\right)$
  - Denoted as Lap(b) when u=0
  - Mean u
  - Variance $2b^2$



Laplace Distribution

How much noise for privacy?

Sensitivity: Consider a query q: I->R. S(q) is the smallest number s.t. for any neighboring tables D, D',

$$| q(D) - q(D') | \leq S(q)$$

Theorem: If sensitivity of the query is S, then the algorithm

$$A(D) = q(D) + Lap(S(q)/\varepsilon)$$

guarantees ε- differential privacy

Privacy of Laplace Mechanism

- Consider neighboring databases D and D’
- Consider some output O

$$\frac{\Pr\left[A(D) = O\right]}{\Pr\left[A(D') = O\right]} = \frac{\Pr\left[q(D) + \eta = O\right]}{\Pr\left[q(D') + \eta = O\right]}$$

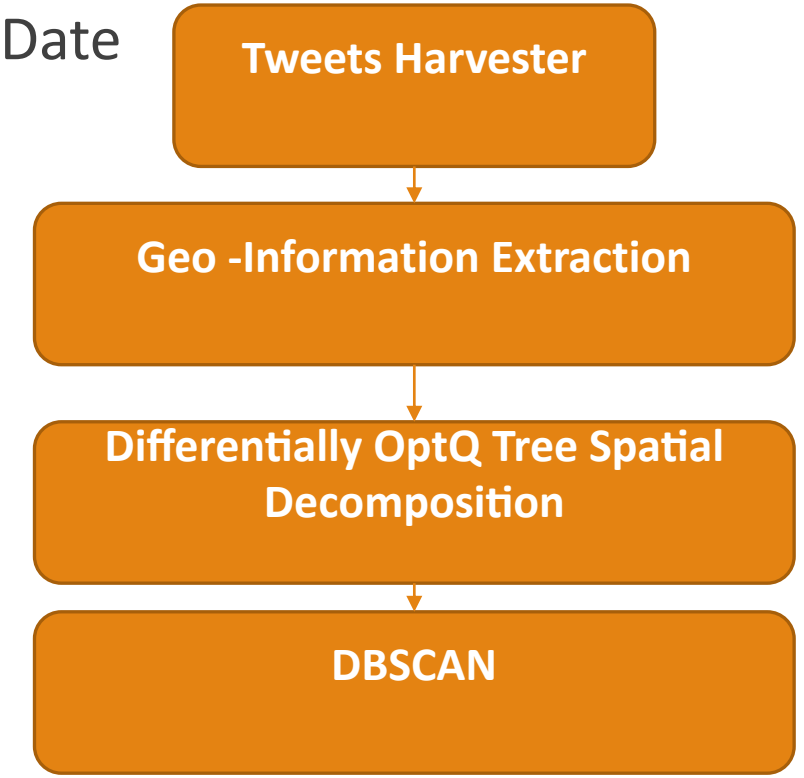
$$= \frac{e^{-|O - q(D)|/\lambda}}{e^{-|O - q(D')|/\lambda}}$$

$$\leq e^{|q(D) - q(D')|/\lambda} \leq e^{S(q)/\lambda} = e^{\varepsilon}$$

Example: Traffic information alerting.

Input: UserId| PointID| Longitude|Latitude|Date

Output: Traffic density



Example: range queries over spatial data

Input: sensitive data D

1	Latitude	Longitude
2	39.98105	116.30142
3	39.9424	116.30587
4	39.93691	116.33438
5	39.94354	116.3532
6	...	...

Scatter plot of input data

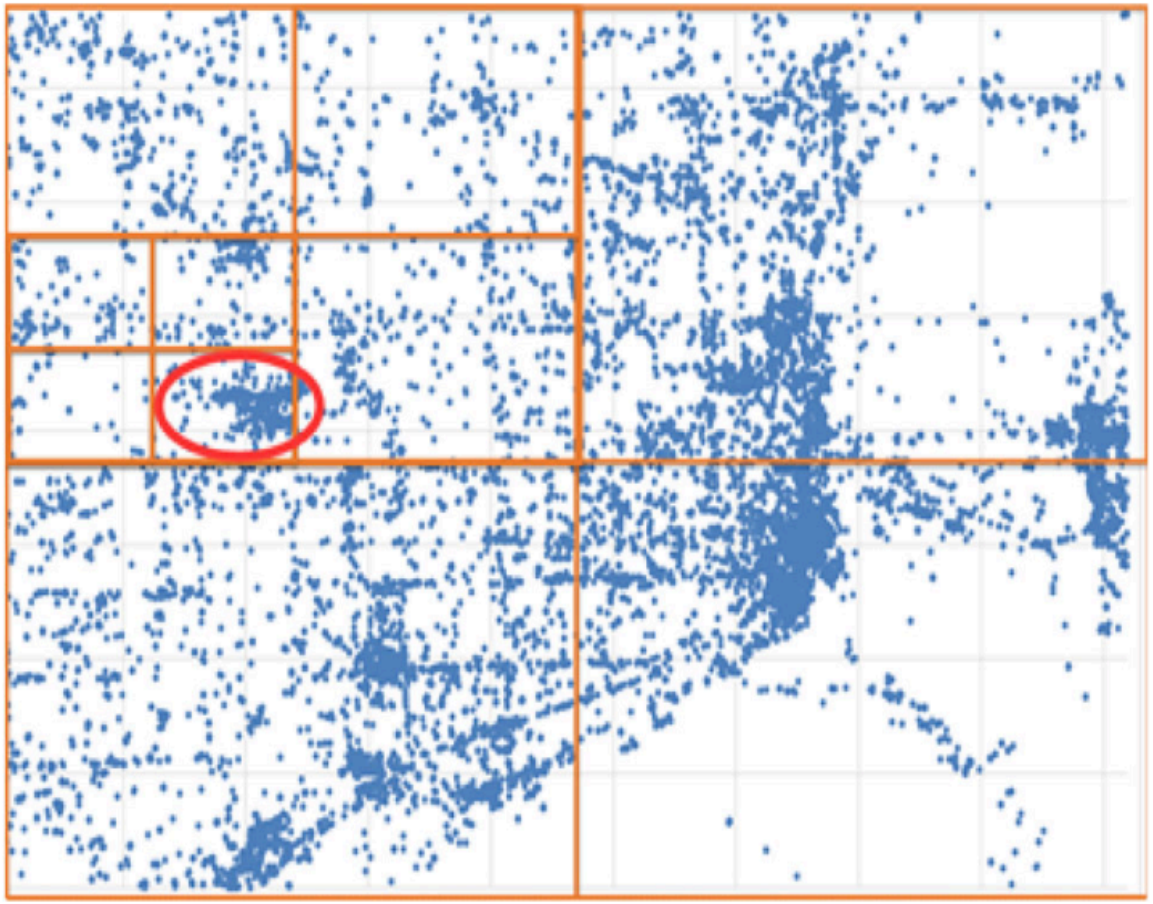
Input: range query workload W

Shown is workload of **3 range queries**

Task: compute answers to workload W over private input D

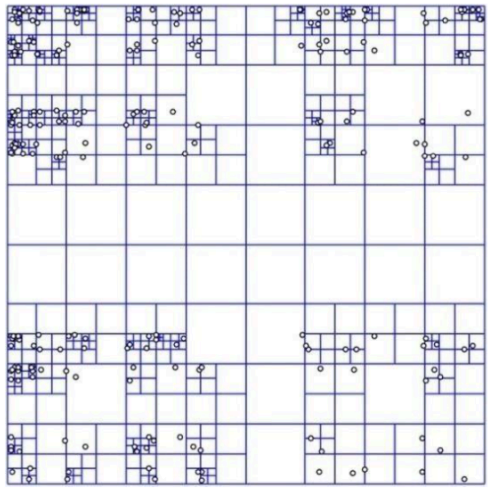
Differential privacy-based spatial decomposition

- Differential privacy spatial decomposition can be divided into adding noise to counts and index structures satisfying differential privacy.
- A **quad-tree**- based spatial decomposition was adopted here to create sets of locations that group points within a certain area from the leaf of the quad-tree.
- Perturb the count of the sub-regions to protect the differential privacy of the count query outputs

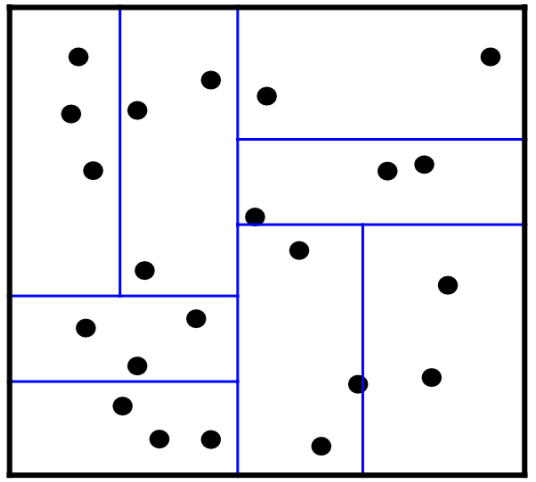


Private Spatial decompositions

- Build: partitioning with differential privacy
- Release: a private description of data distribution (in the form of bounding boxes and noisy counts)



quadtree



kd-tree

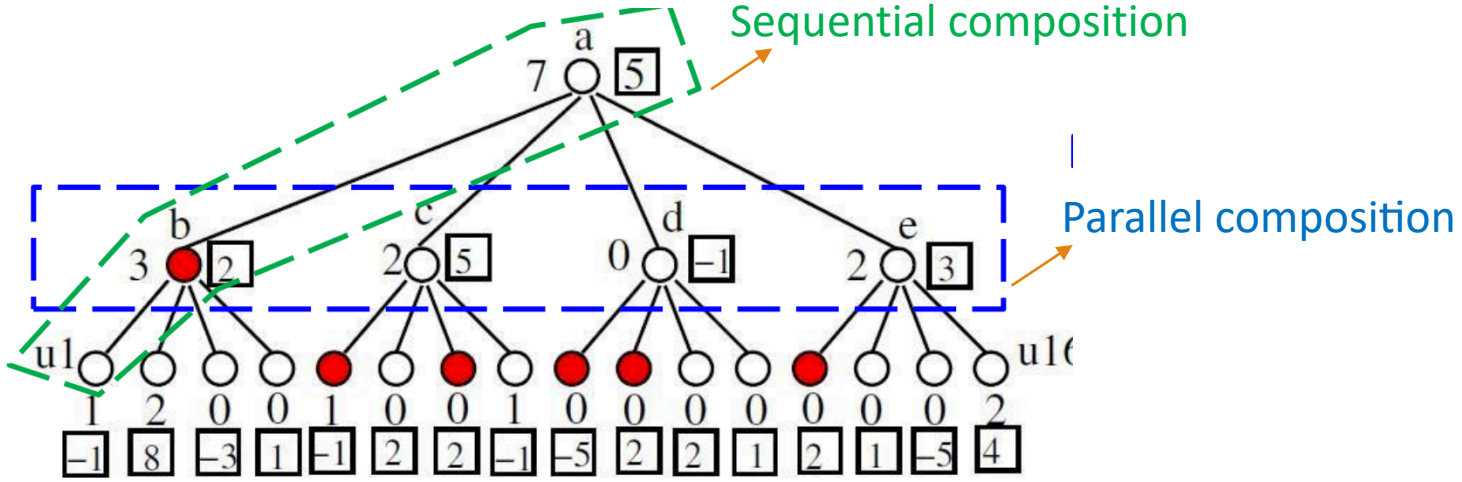
Building a Private quad-tree

- Process to build a private **quad-tree or (Kd-tree)**
  - Input: maximum height h, minimum leaf size L, data set
  - Choose dimension to split
  - Get (private) median in this dimension
  - Create child nodes and add noise to the counts
  - Recurse until:
    - Max height is reached
    - Noisy count of this node less than L
    - Budget along the root-leaf path has used up
- The entire PSD satisfies DP by the composition property

Building a Private quad-tree

Building PSDs – privacy budget allocation

- ❖ Budget is split between medians and counts at each node – Tradeoff accuracy of division with accuracy of counts
- ❖ Budget is split across levels of the tree
- – Privacy budget used along any root-leaf path should total
- – Optimal budget allocation
- – Post processing with consistency check



Differential privacy-based spatial decomposition

Algorithm 1. Optimal quad-tree spatial decomposition with differential privacy (OptQ-SDDP)

Variables: $P = \{\}$; Sp $= \{\}$; $H = 8$; $T = 3L$

OptQ -SDDP (S, R, T)

0: Obtain ε_i according to geometric privacy budget strategy

1: CountWithNoise $= |S| + \text{Lap}(\Delta f/\varepsilon_i)$;

2: **if** $h > 8$ then

3: $P = P \cup \{R\}$; Sp $=$ Sp$\cup\{S\}$;

4: return

5: **else if** CountWithNoise $< L$ then

6: $P = P \cup \{R\}$; Sp $=$ Sp$\cup\{S\}$;

7: return

8: **else**

9: Split spatial region R into 4 equal quadrants

10: OptQ -SDDP (S{q1}; Rn{q1}; T);

11: OptQ -SDDP (S{q2}; Rn{q2}; T);

12: OptQ -SDDP (S{q3}; Rn{q3}; T);

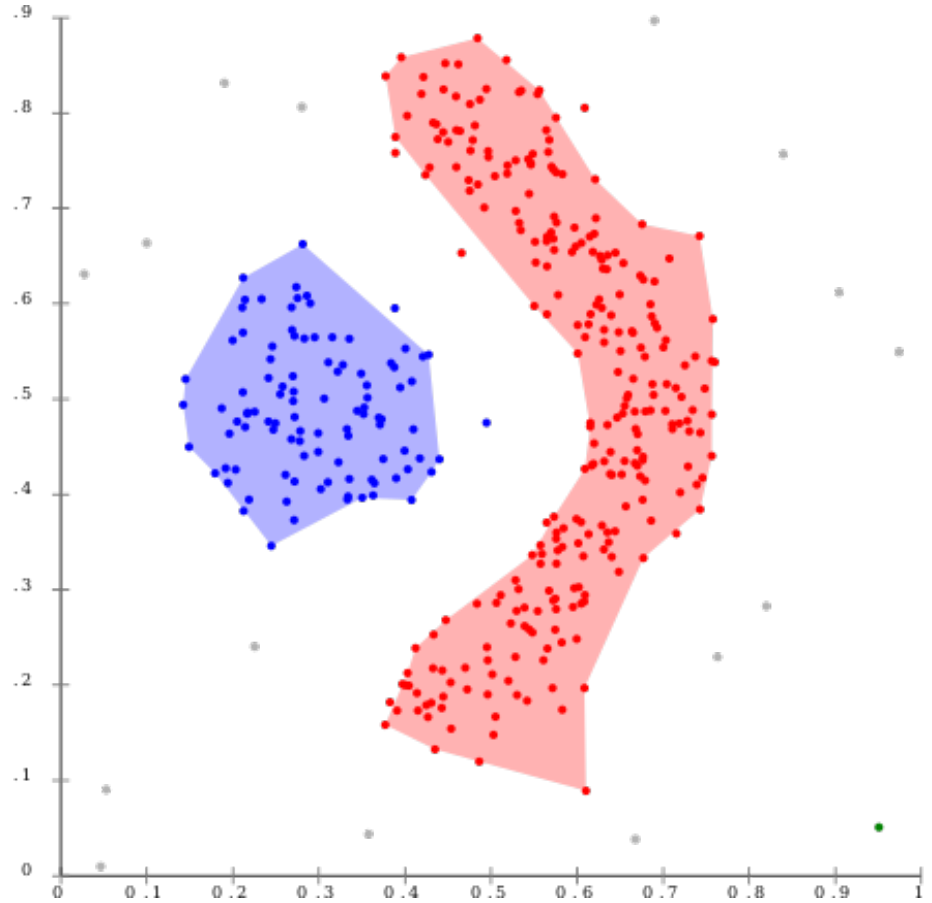
13: OptQ -SDDP (S{q4}; Rn{q4}; T);

14: **end if**

15: **return**

DBSCAN Algorithm

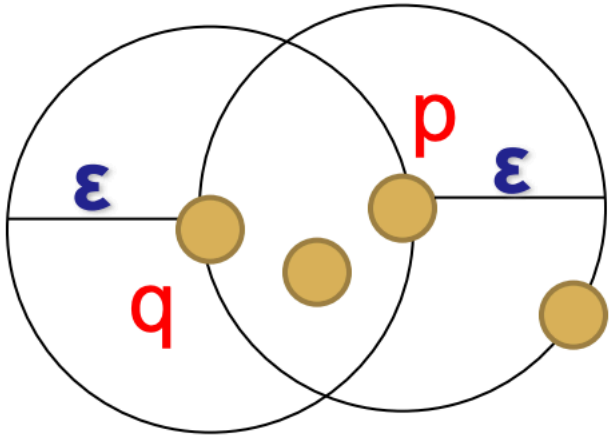
- Density based clustering



DBSCAN Algorithm

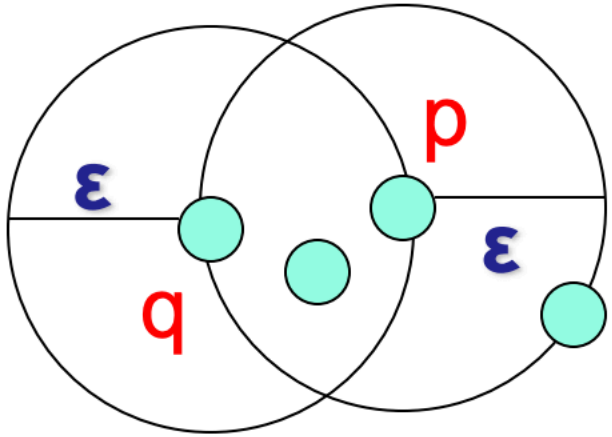
- Parameters:
  - ε: the radius of a neighborhood with respect to some point, called ε-neighborhood.
  - MinPts: the min number of points that are required in ε-neighborhood of a point.

=> A point *p* is a *core point* if at least minPts points are within distance *ε* of it.



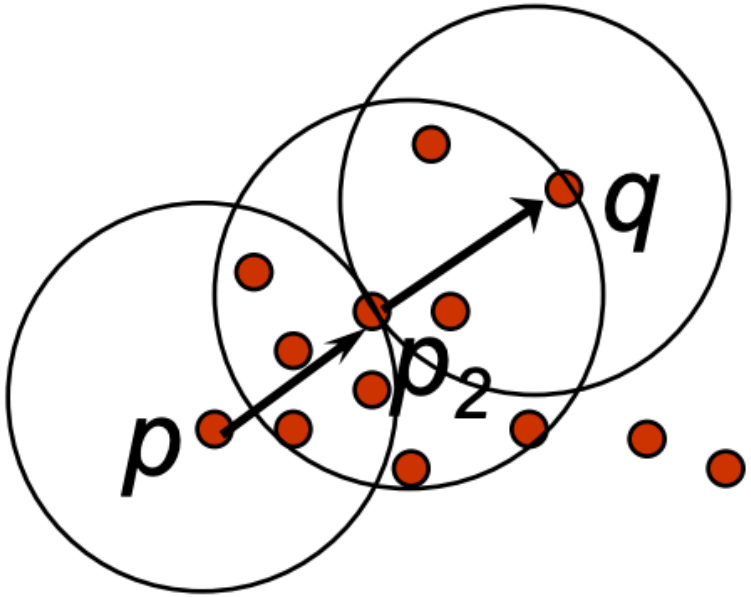
DBSCAN Algorithm

- A point *q* is *directly reachable* from *p* if point *q* is within distance *ε* from core point *p*. Points are only said to be directly reachable from core points.



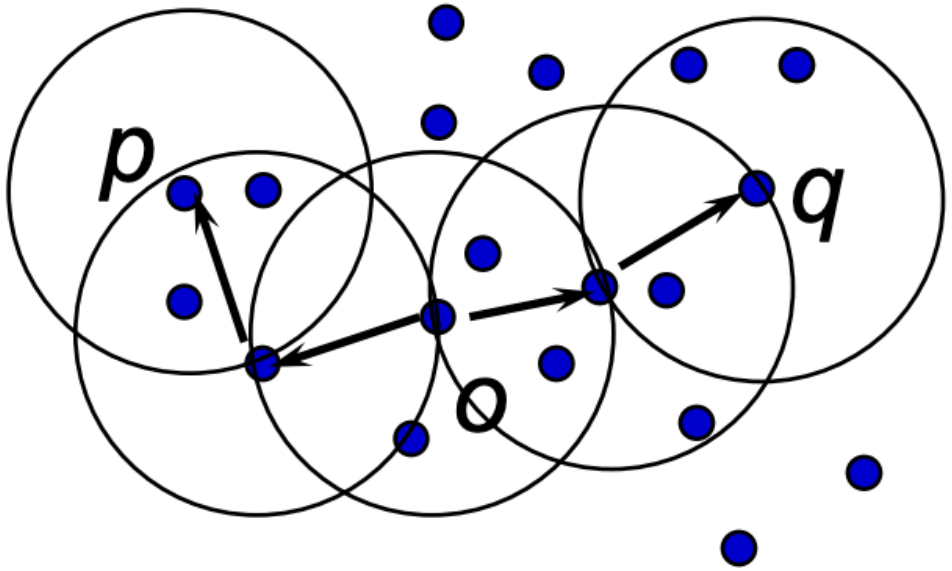
DBSCAN Algorithm

- A point q is *reachable* from p if there is a path p_1, ..., p_n with $p_1 = p$ and $p_n = q$, where each p_{i+1} is directly reachable from p_i. Note that this implies that the initial point and all points on the path must be core points, with the possible exception of q.



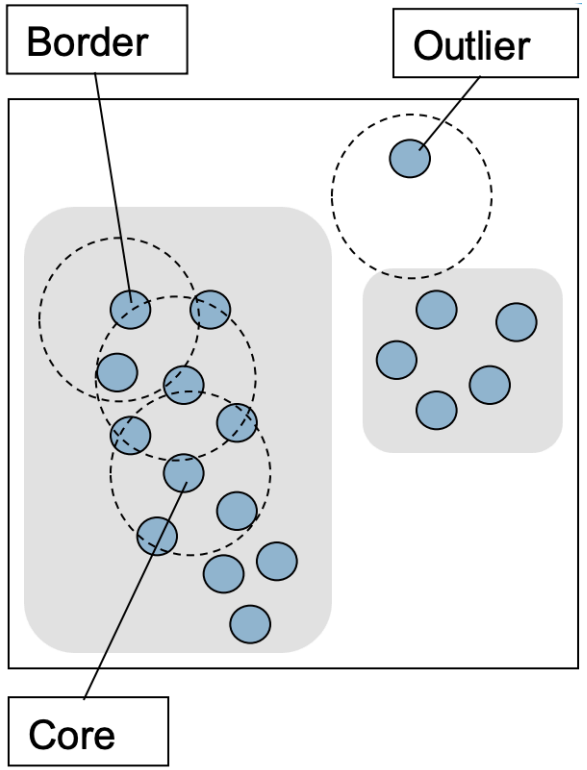
DBSCAN Algorithm

- Two points *p* and *q* are density-connected if there is a point *o* such that both *p* and *q* are reachable from *o*.

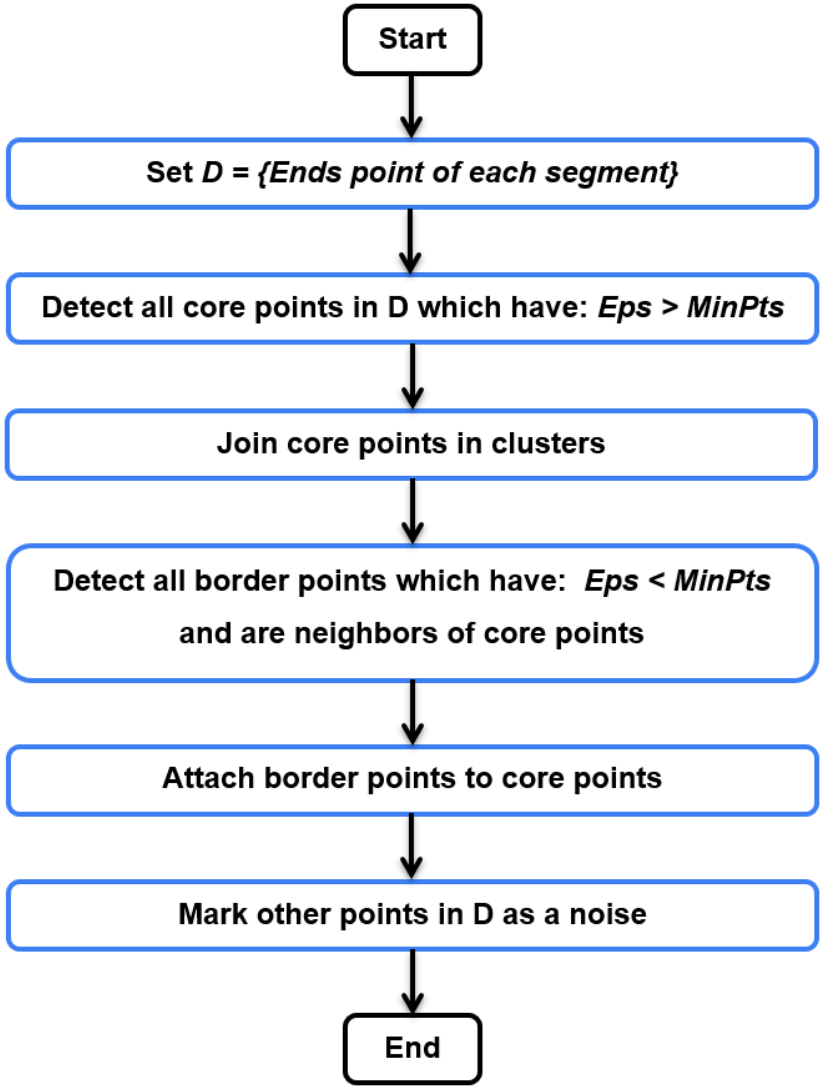


DBSCAN Algorithm

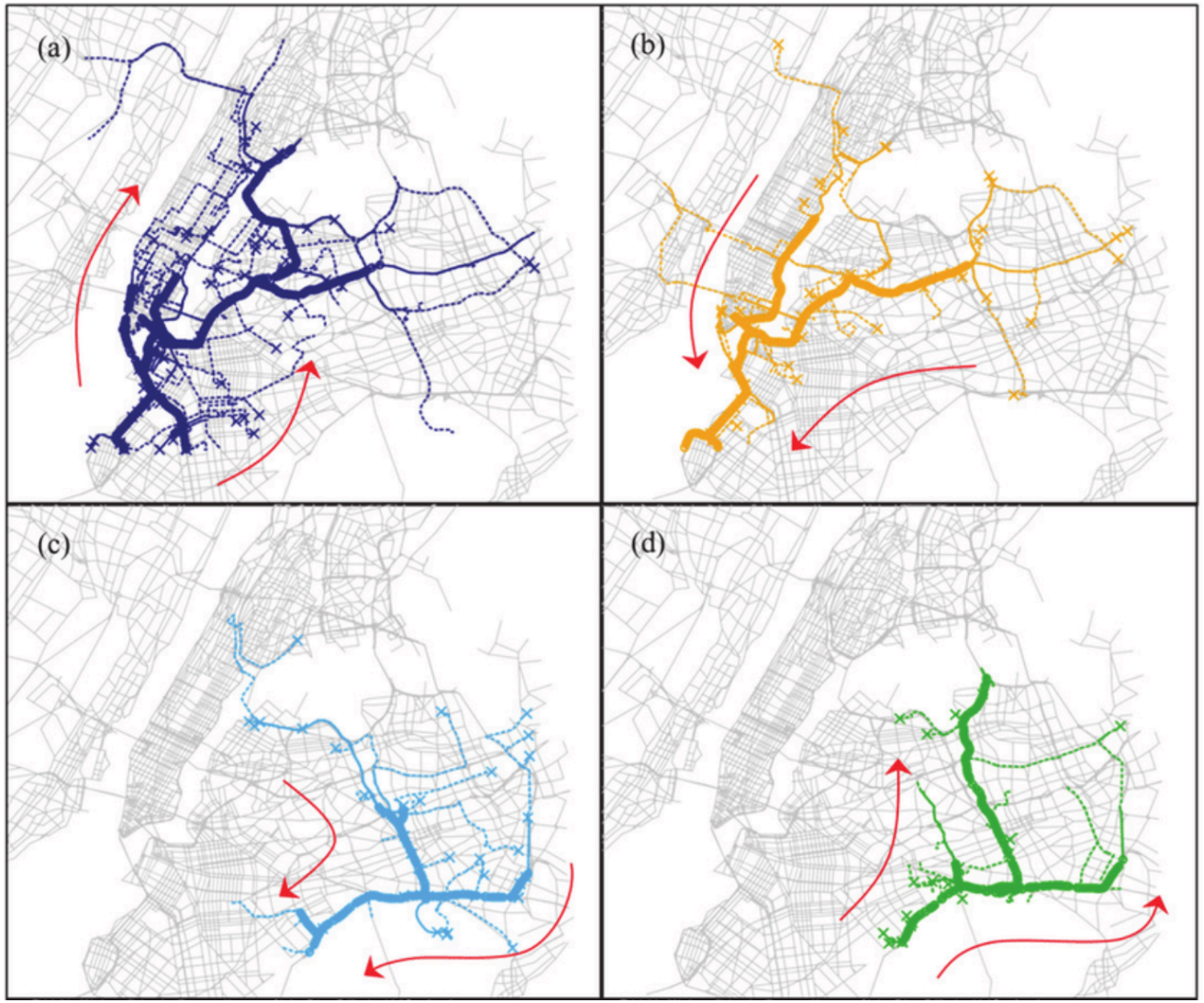
- A cluster then satisfies two properties:
  - All points within the cluster are mutually density-connected.
  - If a point is density-reachable from some point of the cluster, it is part of the cluster as well.



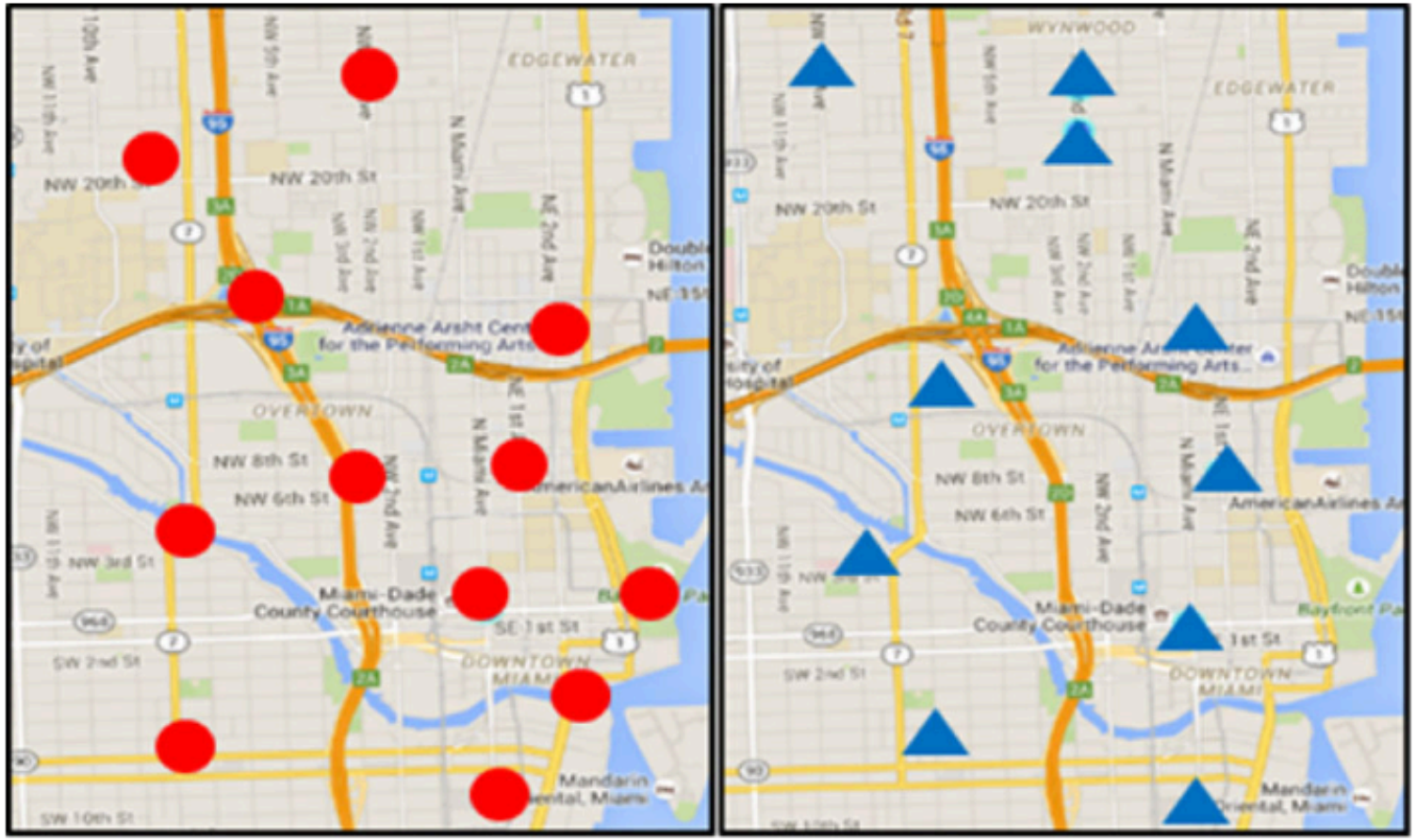
DBSCAN Algorithm



DBSCAN Algorithm



Result


EDGEWATER
NW 20th St
OVERTOWN
NW 8th St
NW 6th St
NW 3rd St
SW 2nd St
SE 1st St
DOWNTOWN MIAMI
Miami-Dade County Courthouse
WYNWOOD

Conclusion

- This method can not only protect a user's location privacy while efficiently ensuring the accuracy of the location-based service through differential privacy, but also protect the privacy of each individual user by adding noise to the statistical reports so that a user's tweets cannot significantly change the alert status.
- We explored adding differential privacy capabilities to Twitter data. Through the application of RDBC to cluster sub-regions split by differentially privacy optimal quad-tree spatial decomposition
- We showed that privacy and precision are trade-offs
- One key area of application of Twitter is real-time information on transport. Tweets about traffic conditions such as traffic congestion or traffic accidents provide near real-time traffic information that is useful for travelers and could allow them to take alternative routes.

References

[1] Shuo Wang1 & Richard O “Supporting geospatial privacy-preserving data mining of social media”

[2] Miguel Andre´s, Nicola´s Bordenabe “Geo-Indistinguishability: Differential Privacy for Location-Based Systems”

[3] Internet

THANK YOU!