Spatial Crowdsourcing: Challenges, Techniques, and Applications

Yongxin Tong  
Beihang University

Lei Chen  
Hong Kong University of Science and Technology

Cyrus Shahabi  
University of Southern California
Outline

- Overview of Spatial Crowdsourcing (20min)
  - Motivation
  - Workflow
  - Core Issues
  - Difference from Related Tutorials

- Fundamental Techniques (50min)
  - Task Assignment
  - Quality Control
  - Incentive Mechanism
  - Privacy Protection

- Spatial Crowdsourced Applications (15min)
  - Spatial Crowdsourcing Intrinsic Applications
  - Crowd-powered Spatial Applications

- Open Questions (5 min)
Crowdsourcing: Concept

- Crowdsourcing
  - Organizing the crowd (Internet workers) to do micro-tasks in order to solve human-intrinsic problems
Crowdsourcing: Applications

- Wikipedia
  - Collaborative knowledge

- reCAPTCHA
  - Digitalizing newspapers

- Yahoo Answer
  - Question & Answer

- ImageNet
  - Image labelling

- Amazon Mechanical Turk (AMT)
  - General purpose crowdsourcing
Crowdsourcing in the Internet era connects requesters and workers on the Internet.
Spatial Crowdsourcing: Concept

- Crowdsourcing
  - Organizing the crowd (Internet workers) to do micro-tasks in order to solve human-intrinsic problems

- Spatial Crowdsourcing
  - Organizing the crowd (Mobile Internet workers) to do spatial tasks by physically moving to other locations
Spatial Crowdsourcing: Concept

Location-based Social Networks

Spatiotemporal Data Processing

Spatial Crowdsourcing

Mobile Computing

Crowdsourcing

a.k.a. mobile crowdsourcing, mobile crowdsensing, participatory sensing, location-based crowdsourcing
Spatial Crowdsourcing: Applications

- Open Street Map
  - Collaborative map

- Waze
  - Live traffic

- Facebook
  - Check-in

- Uber
  - Smart transportation

- Gigwalk
  - General purpose spatial crowdsourcing
## Application Comparison

<table>
<thead>
<tr>
<th>Information Collection</th>
<th>(Internet) Crowdsourcing</th>
<th>Spatial Crowdsourcing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Passive Participation</td>
<td><img src="image1" alt="Wikipedia" /></td>
<td><img src="image2" alt="Waze" /></td>
</tr>
<tr>
<td>Question &amp; Answer</td>
<td><img src="image3" alt="Yahoo Answers" /></td>
<td><img src="image4" alt="Facebook" /></td>
</tr>
<tr>
<td>Specialized Platforms</td>
<td><img src="image5" alt="ImageNet" /></td>
<td><img src="image6" alt="Uber" /></td>
</tr>
<tr>
<td>General Platforms</td>
<td><img src="image7" alt="Amazon" /></td>
<td><img src="image8" alt="Gigwalk" /></td>
</tr>
</tbody>
</table>
Spatial Crowdsourcing: Other Apps

- Local Search-and-discovery Service: Foursquare
- Mobile Market Research & Audits: Field Agent
- Repair & Refresh Your Home: TaskRabbit
- Intelligent Transportation: DiDi
- Food Delivery: Yelp
- Location-based Game: Pokémon GO
Outline

- Overview of Spatial Crowdsourcing (20min)
  - Motivation
  - Workflow
  - Core Issues
  - Difference from Related Tutorials

- Fundamental Techniques (50min)
  - Task Assignment
  - Quality Control
  - Incentive Mechanism
  - Privacy Protection

- Spatial Crowdsourced Applications (15min)
  - Spatial Crowdsourcing Intrinsic Applications
  - Crowd-powered Spatial Applications

- Open Questions (5min)
Spatial Crowdsourcing: Workflow

- Requesters
  - Submit tasks

- Platforms
  - Task management

- Workers
  - Perform tasks
Spatial Crowdsourcing: Workflow

- **Requesters**
  - Submit tasks

- **Platforms**
  - Task management

- **Workers**
  - Perform tasks

- Submit tasks
- Get answers
- Assign tasks
- Submit results
Spatial Crowdsourcing: Task

- General spatial tasks
  - Inventory identification
  - Placement checking
  - Data collection
  - Service evaluation
Spatial Crowdsourcing: Task

- Specific spatial tasks
  - Taxi calling service
  - Ridesharing service
Spatial Crowdsourcing: Workflow

- **Requesters**
  - Submit tasks

- **Platforms**
  - Task management

- **Workers**
  - Perform tasks

Submit tasks —> Get answers
Assign tasks —> Submit results
Spatial Crowdsourcing: Platform

- **Management Modes**
  - **Worker Selected Tasks (WST)**
    - Workers *actively* select tasks
  - **Server Assigned Tasks (SAT)**
    - Workers *passively* wait for the platform to assign tasks

Most studies focus on SAT mode because more optimization techniques can be designed by platforms.
Other Platforms

- **gMission**
  - An open sourced spatial crowdsourcing platform
  - [http://gmission.github.io](http://gmission.github.io)
Other Platforms

- **MediaQ**
  - An online spatial-crowdsourcing-based media management system
  - [http://mediaq.usc.edu/](http://mediaq.usc.edu/)
Other Platforms

- **iRain**
  - A Spatial Crowdsourcing System for Real-time Rainfall Observation
  - [http://irain.eng.uci.edu/](http://irain.eng.uci.edu/)
Spatial Crowdsourcing: Workflow

- **Requesters**
  - Submit tasks

- **Platforms**
  - Task management

- **Workers**
  - Perform tasks

![Diagram showing the workflow of spatial crowdsourcing]

- Submit tasks
- Get answers
- Assign tasks
- Submit results
Spatial Crowdsourcing: Worker

- Influential factor
  - Distance
  - Socioeconomic status

Workers tend to perform nearby tasks

Spatial Crowdsourcing: Worker

- Influential factor
  - Distance
  - Socioeconomic status

Workers tend to perform tasks in high income regions

Outline

- Overview of Spatial Crowdsourcing (20min)
  - Motivation
  - Workflow
  - Core Issues
  - Difference from Related Tutorials

- Fundamental Techniques (50min)
  - Task Assignment
  - Quality Control
  - Incentive Mechanism
  - Privacy Protection

- Spatial Crowdsourced Applications (15min)
  - Spatial Crowdsourcing Intrinsic Applications
  - Crowd-powered Spatial Applications

- Open Questions (5min)
Core Issues in Spatial Crowdsourcing

- **Quality Control**: Crowd might err, spatiotemporal factors might influence quality.
- **Task Assignment**: Location of crowd is sensitive.
- **Incentive Mechanism**: Recruitment of mobile crowd is not free.
- **Privacy Protection**: Core issue: a basis of QC, IM, and PP.
A Spatial Crowdsourcing System

- Requester
  - Query
  - Result

Database

Optimization
  - Index

Crowd-powered Spatial Operators
  - SC Path Selection
  - SC Speed Estimation
  - SC POI Labelling
  - ...

Spatial Crowdsourcing Executor
  - Task Assignment
  - Quality Control
  - Incentive Mechanism
  - Privacy Protection

Tasks
- Worker
  - Answers
Existing Works

- VLDB
- ICDE
- SIGMOD
- GIS
Outline

- Overview of Spatial Crowdsourcing (20min)
  - Motivation
  - Workflow
  - Core Issues
  - Difference from Related Tutorials

- Fundamental Techniques (50min)
  - Task Assignment
  - Quality Control
  - Incentive Mechanism
  - Privacy Protection

- Spatial Crowdsourced Applications (15min)
  - Spatial Crowdsourcing Intrinsic Applications
  - Crowd-powered Spatial Applications

- Open Questions (5min)
Difference from Related Tutorials

- **VLDB 2012**
  - Crowdsourced platforms and principles

- **CIKM 2014**
  - Traditional crowdsourced knowledge management

- **ICDE 2015**
  - Traditional crowdsourced queries, mining and applications

- **VLDB 2016**
  - Human factors involved in task assignment

- **SIGMOD 2017**
  - Traditional crowdsourced data management

- **Our Tutorial**
  - **Spatial** crowdsourced data management
  - **Theories** and **applications** in spatial crowdsourcing
Outline

- Overview of Spatial Crowdsourcing (20min)
  - Motivation
  - Workflow
  - Core Issues
  - Difference from Related Tutorials

- Fundamental Techniques (50min)
  - Task Assignment
  - Quality Control
  - Incentive Mechanism
  - Privacy Protection

- Spatial Crowdsourced Applications (15min)
  - Spatial Crowdsourcing Intrinsic Applications
  - Crowd-powered Spatial Applications

- Open Questions (5min)
Why Task Assignment?

- Almost all other core issues in spatial crowdsourcing depend on task assignment.

- The core operation connects tasks, workers and the platform.

[Logos for DiDi, Gigwalk, Uber, and Taskrabbit]
Problem Definition

Given a set of workers and a set of tasks distributed in the physical space, the objective is to assign tasks to proper workers.

Dependent on specific objectives: e.g., minimum total distance.
Categories of Existing Research

- Arrival Scenarios: Static vs. Dynamic
- Algorithmic Models: Matching vs. Planning
Categories of Existing Research

- **Arrival Scenarios: Static vs. Dynamic**
- **Algorithmic Models: Matching vs. Planning**
Static Scenario vs. Dynamic Scenario

- **Static Scenario**
  - The platform is assumed to **know all spatiotemporal information of tasks and workers at the beginning**

  Refer to all arrival times and locations of tasks/workers

<table>
<thead>
<tr>
<th>Arrival Time</th>
<th>8:00</th>
<th>8:01</th>
<th>8:02</th>
<th>8:07</th>
<th>8:08</th>
<th>8:09</th>
<th>8:09</th>
<th>8:15</th>
<th>8:18</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$t_1$</td>
<td>$w_1$</td>
<td>$t_2$</td>
<td>$t_3$</td>
<td>$w_2$</td>
<td>$t_4$</td>
<td>$w_3$</td>
<td>$w_4$</td>
<td>$t_5$</td>
</tr>
</tbody>
</table>
Static Scenario vs. Dynamic Scenario

- Dynamic Scenario
  - Tasks/workers usually dynamically appear
  - They usually need to be assigned immediately based on real-time partial information
    - Fast Food Delivery Service
    - Real-Time Taxi-Calling Service
Static Scenario vs. Dynamic Scenario

- Dynamic Scenario: An Example
  - New tasks and workers dynamically arrive at the platform during task assignment
Static Scenario vs. Dynamic Scenario

- Dynamic Scenario: An Example
  - New tasks and workers dynamically arrive at the platform during task assignment

Arrival Time 8:00

$t_1$
Static Scenario vs. Dynamic Scenario

- **Dynamic Scenario: An Example**
  - New tasks and workers dynamically arrive at the platform during task assignment.

<table>
<thead>
<tr>
<th>Arrival Time</th>
<th>8:00</th>
<th>8:01</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_1$</td>
<td></td>
<td>$w_1$</td>
</tr>
</tbody>
</table>
Static Scenario vs. Dynamic Scenario

- **Dynamic Scenario: An Example**
  - New tasks and workers dynamically arrive at the platform during task assignment

<table>
<thead>
<tr>
<th>Arrival Time</th>
<th>8:00</th>
<th>8:01</th>
<th>8:02</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_1$</td>
<td>$w_1$</td>
<td>$t_2$</td>
<td></td>
</tr>
</tbody>
</table>
Static Scenario vs. Dynamic Scenario

- **Dynamic Scenario:** An Example
  - New tasks and workers dynamically arrive at the platform during task assignment

<table>
<thead>
<tr>
<th>Arrival Time</th>
<th>8:00</th>
<th>8:01</th>
<th>8:02</th>
<th>8:07</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$t_1$</td>
<td>$w_1$</td>
<td>$t_2$</td>
<td>$t_3$</td>
</tr>
</tbody>
</table>
Static Scenario vs. Dynamic Scenario

- **Dynamic Scenario: An Example**
  - New tasks and workers dynamically arrive at the platform during task assignment

<table>
<thead>
<tr>
<th>Arrival Time</th>
<th>8:00</th>
<th>8:01</th>
<th>8:02</th>
<th>8:07</th>
<th>8:08</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$t_1$</td>
<td>$w_1$</td>
<td>$t_2$</td>
<td>$t_3$</td>
<td>$w_2$</td>
</tr>
</tbody>
</table>
Static Scenario vs. Dynamic Scenario

- **Dynamic Scenario: An Example**
  - New tasks and workers dynamically arrive at the platform during task assignment.

<table>
<thead>
<tr>
<th>Arrival Time</th>
<th>8:00</th>
<th>8:01</th>
<th>8:02</th>
<th>8:07</th>
<th>8:08</th>
<th>8:09</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$t_1$</td>
<td>$w_1$</td>
<td>$t_2$</td>
<td>$t_3$</td>
<td>$w_2$</td>
<td>$t_4$</td>
</tr>
</tbody>
</table>
Static Scenario vs. Dynamic Scenario

- **Dynamic Scenario: An Example**
  - New tasks and workers dynamically arrive at the platform during task assignment

<table>
<thead>
<tr>
<th>Arrival Time</th>
<th>8:00</th>
<th>8:01</th>
<th>8:02</th>
<th>8:07</th>
<th>8:08</th>
<th>8:09</th>
<th>8:09</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_1$</td>
<td>$w_1$</td>
<td>$t_2$</td>
<td>$t_3$</td>
<td>$w_2$</td>
<td>$t_4$</td>
<td>$w_3$</td>
<td></td>
</tr>
</tbody>
</table>
Static Scenario vs. Dynamic Scenario

- Dynamic Scenario: An Example
  - New tasks and workers dynamically arrive at the platform during task assignment

<table>
<thead>
<tr>
<th>Arrival Time</th>
<th>8:00</th>
<th>8:01</th>
<th>8:02</th>
<th>8:07</th>
<th>8:08</th>
<th>8:09</th>
<th>8:09</th>
<th>8:15</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$t_1$</td>
<td>$w_1$</td>
<td>$t_2$</td>
<td>$t_3$</td>
<td>$w_2$</td>
<td>$t_4$</td>
<td>$w_3$</td>
<td>$w_4$</td>
</tr>
</tbody>
</table>
Static Scenario vs. Dynamic Scenario

- Dynamic Scenario: An Example
  - New tasks and workers dynamically arrive at the platform during task assignment

<table>
<thead>
<tr>
<th>Arrival Time</th>
<th>8:00</th>
<th>8:01</th>
<th>8:02</th>
<th>8:07</th>
<th>8:08</th>
<th>8:09</th>
<th>8:09</th>
<th>8:15</th>
<th>8:18</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$t_1$</td>
<td>$w_1$</td>
<td>$t_2$</td>
<td>$t_3$</td>
<td>$w_2$</td>
<td>$t_4$</td>
<td>$w_3$</td>
<td>$w_4$</td>
<td>$t_5$</td>
</tr>
</tbody>
</table>

The platform does not know information of subsequent tasks/workers when it performs task assignment dynamically.
Categories of Existing Research

- Arrival Scenarios: Static vs. Dynamic
- Algorithmic Models: Matching vs. Planning
Matching vs. Planning

Matching Model

- Formulate task assignment problem as classic "weighted bipartite graph matching" problem.

Weight: utility score of assigning this task to the worker.
Matching vs. Planning

- **Planning Model**
  - Plan a route for a worker or workers to complete a number of tasks
  - Each worker usually performs multiple tasks instead of performing a single task in the matching model

Plan 1: $t_2 \rightarrow t_3 \rightarrow t_4 \rightarrow t_1$
Plan 2: $t_1 \rightarrow t_3 \rightarrow t_4 \rightarrow t_2$
Static Matching

Static Matching

Dynamic Matching

Static Planning

Dynamic Planning

Planning
Static Matching

- Problem Definition

- Existing Research
  - Objective 1: MaxSum Matching
  - Objective 2: MinSum Matching
  - Objective 3: Stable Marriage Matching
Static Matching

- **Problem Definition**
  - Given **all spatiotemporal information** of a set of workers and tasks, the problem is to assign the tasks to the proper workers based on the specific objective function.
Static Matching

- Problem Definition

- Existing Research
  - Objective 1: MaxSum Matching
  - Objective 2: MinSum Matching
  - Objective 3: Stable Marriage Matching

Spatial information acts as constraints

Spatial information is the optimization goal
Static Matching

- Problem Definition

- Existing Research
  - Objective 1: MaxSum Matching
  - Objective 2: MinSum Matching
  - Objective 3: Stable Marriage Matching
MaxSum Matching

- **Objective 1:** Maximizing the total utility/number of the matching

The problem of static matching with Objective 1 equals to the Maximum Weighted Bipartite Matching (MWBM) problem.
MaxSum Matching

- The MWBM problem can be solved by various of classical Max-Flow algorithms
  - Ford-Fulkerson Algorithm, etc.

---

H. To, C. Shahabi, L. Kazemi: A server-assigned spatial crowdsourcing framework. TSAS 2015
Static Matching

- Problem Definition

- Existing Research
  - Objective 1: MaxSum Matching
  - Objective 2: MinSum Matching
  - Objective 3: Stable Marriage Matching
MinSum Matching

- **Objective 2**: Minimizing the total distance cost of the maximum-cardinality matching (prefect matching)

The problem of static matching with Objective 2 equals to the **Maximum Flow with Minimum Cost problem**.

MinSum Matching

- The MinSum problem can be solved by a serial of classical assignment algorithms
  - Successive Shortest Path Algorithm (SSPA)
  - Leverage index and I/O optimization techniques

Static Matching

- Problem Definition

- Existing Research

  - Objective 1: MaxSum Matching
  - Objective 2: MinSum Matching
  - Objective 3: Stable Marriage Matching
Stable Marriage Matching

- Stable Marriage: Given the preference lists of $n$ women and $n$ men, the problem is to find a perfect matching with no unstable pairs.

- Spatial Stable Marriage: Given the preference lists of $n$ tasks and $n$ workers based on the order of their distances, the problem is to find a perfect matching with no unstable pairs.

Stable Marriage Matching

- A worker-task pair \((w, t)\) is an unstable pair if
  - \(|w, t| < \text{the distance between } t \text{ and } t's \text{ partner in the given matching}\)
  - \(|w, t| < \text{the distance between } w \text{ and } w's \text{ partner in the given matching}\)

Stable Marriage Matching

- A worker-task pair \((w, t)\) is an **unstable pair** if
  - \(|w, t| < \text{the distance between } t \text{ and } t’s \text{ partner in the given matching} \)
  - \(|w, t| < \text{the distance between } w \text{ and } w’s \text{ partner in the given matching} \)

Stable Marriage Matching

- Classical Gale-Shapley algorithm is impractical for large-scale spatial data
- Design a Chain algorithm, which iteratively utilizes NN operation to enhance the efficiency

Iteratively find NN

Reference: Static Matching

Dynamic Matching

- Static Matching
- Dynamic Matching
- Static Planning
- Dynamic Planning
Dynamic Matching

- **Batch-based Matching**
  - Problem Definition
  - Existing Research

- **Online Matching**
  - Problem Definition
  - Existing Research
    - Objective 1: Online MaxSum Matching
    - Objective 2: Online MinSum Matching

- **Summary**
Batch-based Matching

- A set of workers and tasks dynamically appear in the physical space, this problem needs to perform a series of static matching for new arrival workers/tasks per time slot based on different objectives.
Dynamic Matching

- Batch-based Matching
  - Problem Definition
  - Existing Research

- Online Matching
  - Problem Definition
  - Existing Research
    - Objective 1: Online MaxSum Matching
    - Objective 2: Online MinSum Matching

- Summary
Batch-based Matching

- **Objective:** Maximizing the total utility/number of matching in all batches (batch size=4min)

Platform

Batch-based Matching

- Perform a MaxSum static matching for new arrival tasks/workers per time slot (e.g., 4min)

Platform gradually knows information (past and present) as time goes on

<table>
<thead>
<tr>
<th>Arrival Time</th>
<th>8:00</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_1$</td>
<td></td>
</tr>
</tbody>
</table>

Batch-based Matching

- Perform a MaxSum **static** matching for new arrival tasks/workers **per time slot** (e.g., 4min)

Platform gradually knows information (past and present) as time goes on

<table>
<thead>
<tr>
<th>Arrival Time</th>
<th>8:00</th>
<th>8:01</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$t_1$</td>
<td>$w_1$</td>
</tr>
</tbody>
</table>

Batch-based Matching

- Perform a MaxSum **static** matching for new arrival tasks/workers **per time slot** (e.g., 4min)

<table>
<thead>
<tr>
<th>Arrival Time</th>
<th>8:00</th>
<th>8:01</th>
<th>8:02</th>
</tr>
</thead>
<tbody>
<tr>
<td>t₁</td>
<td>w₁</td>
<td>t₂</td>
<td></td>
</tr>
</tbody>
</table>

Batch-based Matching

- Perform a MaxSum **static** matching for new arrival tasks/workers **per time slot** (e.g., 4min)

Platform gradually knows information (past and present) as time goes on.

<table>
<thead>
<tr>
<th>Arrival Time</th>
<th>8:00</th>
<th>8:01</th>
<th>8:02</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$t_1$</td>
<td>$w_1$</td>
<td>$t_2$</td>
</tr>
</tbody>
</table>

Perform an assignment at 8:04

Existing Research

- Perform a **local static assignment in each time slot**
  - Exact methods: Ford-Fulkerson algorithm
  - Approximation methods: Greedy algorithm

- Aggregate the total number of the assignments

The total number of assignment is 3

<table>
<thead>
<tr>
<th>Arrival Time</th>
<th>8:00</th>
<th>8:01</th>
<th>8:02</th>
<th>8:07</th>
<th>8:08</th>
<th>8:09</th>
<th>8:09</th>
<th>8:11</th>
</tr>
</thead>
<tbody>
<tr>
<td>t₁</td>
<td>w₁</td>
<td>t₂</td>
<td>t₃</td>
<td>w₂</td>
<td>t₄</td>
<td>w₃</td>
<td>w₄</td>
<td></td>
</tr>
</tbody>
</table>

Dynamic Matching

- **Batch-based Matching**
  - Problem Definition
  - Existing Research

- **Online Matching**
  - Problem Definition
  - Existing Research
    - Objective 1: Online MaxSum Matching
    - Objective 2: Online MinSum Matching

- **Summary**
Online Matching

- A set of tasks/workers dynamically appear one by one in the physical space, the online matching based on a specific objective must be performed immediately and irrevocably, when a new task or worker arrives
The platform immediately decides how to assign \( w_3 \).

In fact, this assignment is better.

The arrival order significantly affects the assignment (assume \( t_4 \) appears first).

But the assignment cannot be changed once made.
Online Matching

- Evaluation of Online Matching Algorithms
  - Competitive Ratio (CR)
    \[ CR = \frac{\text{Value of an online algorithm}}{\text{Value of the corresponding offline algorithm}} \]

- Input Models
  - Adversarial Model (Worst-Case Analysis)
    \[ CR_A = \min_{G(T,W,U) \text{ and } \forall v \in V} \frac{\text{MaxFunc}(M)}{\text{MaxFunc}(OPT)} \]
    \[ CR_A = \max_{G(T,W,U) \text{ and } \forall v \in V} \frac{\text{MinFunc}(M)}{\text{MinFunc}(OPT)} \]
  - Random Order Model (Average-Case Analysis)
    \[ CR_{RO} = \min_{G(T,W,U)} \frac{\mathbb{E}[\text{MaxFunc}(M)]}{\text{MaxSum}(OPT)} \]
    \[ CR_{RO} = \max_{G(T,W,U)} \frac{\mathbb{E}[\text{MinFunc}(M)]}{\text{MinSum}(OPT)} \]
Dynamic Matching

- Batch-based Matching
  - Problem Definition
  - Existing Research

- Online Matching
  - Problem Definition
  - Existing Research
  - Objective 1: Online MaxSum Matching
  - Objective 2: Online MinSum Matching

- Summary
Online MaxSum Matching

- Objective 1: Assign tasks to workers in the online scenario to maximize the total utility/number of the matching
Online MaxSum Matching

- Baseline
  - Greedy
    - When a task/worker appears, the algorithm assigns it to a worker/task with the current maximum utility
Online MaxSum Matching

- **Baseline**
  - **Greedy**
    - When a task/worker appears, the algorithm assigns it to a worker/task with the current maximum utility

\[ w_1 \]

\[ w_2 \]

Online MaxSum Matching

- Baseline
  - Greedy
    - When a task/worker appears, the algorithm assigns it to a worker/task with the current maximum utility

Online MaxSum Matching

- **Baseline**
  - **Greedy**
    - When a task/worker appears, the algorithm assigns it to a worker/task with the current maximum utility

Online MaxSum Matching

- Baseline
  - Greedy
    - When a task/worker appears, the algorithm assigns it to a worker/task with the current maximum utility

Online MaxSum Matching

- Improvements
  - Extended Greedy
    - Choose an integer $k$ from 0 to $\lceil \ln(U_{max} + 1) \rceil$ randomly.
    - Filter the edges with weights lower than the threshold $e^k$.
    - Use a greedy strategy on the remaining edges.

---

Online MaxSum Matching

- Improvements
  - Extended Greedy
    - Choose an integer $k$ from 0 to $\lfloor \ln(U_{max} + 1) \rfloor$ randomly.
    - Filter the edges with weights lower than the threshold $e^k$.
    - Use a greedy strategy on the remaining edges.

---

Dynamic Matching

- Batch-based Matching
  - Problem Definition
  - Existing Research

- Online Matching
  - Problem Definition
  - Existing Research
    - Objective 1: Online MaxSum Matching
    - Objective 2: Online MinSum Matching

- Summary
Online MinSum Matching

- **Objective 2:** Assign tasks to workers in the online scenario to minimize the total utility.
- A task/worker is assigned immediately on arrival based on partial information.

Y. Tong, J. She, B. Ding, L. Chen, T. Wo, K. Xu. Online minimum matching in real-time spatial data: experiments and analysis. VLDB 2016.
Review: Static MinSum Matching

- **Objective:** Minimize the total distance cost of maximum-cardinality matching

- Assume that all spatiotemporal information is known in advance (offline OPT)

Y. Tong, J. She, B. Ding, L. Chen, T. Wo, K. Xu. Online minimum matching in real-time spatial data: experiments and analysis. VLDB 2016.
Review: Static MinSum Matching

- **Objective:** Minimize the total distance cost of maximum-cardinality matching
- Assume that all spatiotemporal information is known in advance (offline OPT)

Y. Tong, J. She, B. Ding, L. Chen, T. Wo, K. Xu. Online minimum matching in real-time spatial data: experiments and analysis. VLDB 2016.
## Online MinSum Matching

- **Four representative algorithms**

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>Time Complexity per Each Arrival Vertex</th>
<th>Randomization</th>
<th>Data Structure</th>
<th>Competitive Ratio (Worst-Case Analysis)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Greedy [SODA’1991]</td>
<td>$O(k)$</td>
<td>Deterministic</td>
<td>No</td>
<td>$O(2^k)$</td>
</tr>
<tr>
<td>Permutation [SODA’1991]</td>
<td>$O(k^3)$</td>
<td>Deterministic</td>
<td>No</td>
<td>$O(2k-1)$</td>
</tr>
<tr>
<td>HST-Greedy [SODA 2006]</td>
<td>$O(k)$</td>
<td>Randomized</td>
<td>HST</td>
<td>$O(\log^3 k)$</td>
</tr>
<tr>
<td>HST-Reassignment [ESA 2007]</td>
<td>$O(k^2)$</td>
<td>Randomized</td>
<td>HST</td>
<td>$O(\log^2 k)$</td>
</tr>
</tbody>
</table>

**Hierarchically Separated Tree**

**Is the greedy algorithm really the worst?**
Online MinSum Matching

- **Greedy Revisited**

  \[
  \begin{array}{cccc}
  w_1 & w_2 & w_3 & w_4 \\
  t_1 & t_2 & t_3 & t_4 \\
  1+\varepsilon & 1 & 2 & 4 \\
  \end{array}
  \]

  (a)Locations

---

Y. Tong, J. She, B. Ding, L. Chen, T. Wo, K. Xu. Online minimum matching in real-time spatial data: experiments and analysis. VLDB 2016.
Online MinSum Matching

- **Greedy Revisited**

(a) Locations

(b) Matching of Offline OPT

---

Y. Tong, J. She, B. Ding, L. Chen, T. Wo, K. Xu. Online minimum matching in real-time spatial data: experiments and analysis. VLDB 2016.
Online MinSum Matching

- **Greedy Revisited**

  (a) Locations
  - $w_1$ at $t_1$
  - $w_2$ at $t_2$
  - $w_3$ at $t_3$
  - $w_4$ at $t_4$

  (b) Matching of Offline OPT
  - $w_1$ at $t_1$
  - $w_2$ at $t_2$
  - $w_3$ at $t_3$
  - $w_4$ at $t_4$

  (c) Matching of Worst-Case Greedy
  - $w_1$ at $t_1$
  - $w_2$ at $t_1$
  - $w_3$ at $t_3$
  - $w_4$ at $t_4$

When $t_1$ appears, $w_2$ will be assigned to $t_1$ by Greedy

---

Y. Tong, J. She, B. Ding, L. Chen, T. Wo, K. Xu. Online minimum matching in real-time spatial data: experiments and analysis. VLDB 2016.
Online MinSum Matching

- **Greedy Revisited**

(a) Locations

(b) Matching of Offline OPT

(c) Matching of Worst-Case Greedy

When $t_2$ appears, $w_3$ will be assigned to $t_2$ by Greedy.

Y. Tong, J. She, B. Ding, L. Chen, T. Wo, K. Xu. Online minimum matching in real-time spatial data: experiments and analysis. VLDB 2016.
Online MinSum Matching

- Greedy Revisited

(a) Locations

(b) Matching of Offline OPT

(c) Matching of Worst-Case Greedy

When $t_3$ appears, $w_4$ will be assigned to $t_3$ by Greedy

Y. Tong, J. She, B. Ding, L. Chen, T. Wo, K. Xu. Online minimum matching in real-time spatial data: experiments and analysis. VLDB 2016.
Online MinSum Matching

- **Greedy Revisited**

  Competitive Ratio $CR_{RO}=3.195$

  - (a) Locations
  - (b) Matching of Offline OPT
  - (c) Matching of Worst-Case Greedy

When $t_4$ appears, $w_1$ has to be assigned to $t_4$ by Greedy

The probability that the worst-case happens is only 1 over the factorial of $k$, which is the number of workers (taxis)

---

Y. Tong, J. She, B. Ding, L. Chen, T. Wo, K. Xu. Online minimum matching in real-time spatial data: experiments and analysis. VLDB 2016.
Online MinSum Matching

Their extensive experiments on real datasets and synthetic datasets shows that the greedy algorithm is not bad and always outperforms other state-of-the-art online algorithms!

Y. Tong, J. She, B. Ding, L. Chen, T. Wo, K. Xu. Online minimum matching in real-time spatial data: experiments and analysis. VLDB 2016.
Online MinSum Matching

Open question: the average-case competitive ratio under the random order model of Greedy for the online MinSum matching problem should be constant or $O(k)$?

Y. Tong, J. She, B. Ding, L. Chen, T. Wo, K. Xu. Online minimum matching in real-time spatial data: experiments and analysis. VLDB 2016.
Dynamic Matching

- Batch-based Matching
  - Problem Definition
  - Existing Research

- Online Matching
  - Problem Definition
  - Existing Research
    - Objective 1: Online MaxSum Matching
    - Objective 2: Online MinSum Matching

- Summary
## Summary of Matching Models

<table>
<thead>
<tr>
<th>Static/Dynamic Scenario</th>
<th>Effect of Spatial Factors</th>
<th>Optimization Target</th>
</tr>
</thead>
<tbody>
<tr>
<td>U et al. SIGMOD08</td>
<td>Static</td>
<td>Target</td>
</tr>
<tr>
<td>Wong et al. VLDB07</td>
<td>Static</td>
<td>Target</td>
</tr>
<tr>
<td>Long et al. SIGMOD13</td>
<td>Static</td>
<td>Target</td>
</tr>
<tr>
<td>She et al. ICDE15</td>
<td>Static</td>
<td>Constraint</td>
</tr>
<tr>
<td>Alfarrarjeh et al. MDM15</td>
<td>Static</td>
<td>Target</td>
</tr>
<tr>
<td>Kazemi et al. GIS12</td>
<td>Batch-based</td>
<td>Constraint</td>
</tr>
<tr>
<td>To et al. TSAS15</td>
<td>Batch-based</td>
<td>Constraint</td>
</tr>
<tr>
<td>Liu et al. ICDE16</td>
<td>Batch-based</td>
<td>Constraint</td>
</tr>
<tr>
<td>Cheng et al. ICDE17</td>
<td>Batch-based</td>
<td>Prediction-based</td>
</tr>
</tbody>
</table>

- **MinSum Distance**
- **Spatial Stable Marriage**
- **MinMax Distance**
- **Conflict-based MaxSum Utility**
- **MaxSum Number**
- **MaxSum Utility**
- **MaxSum Utility**
- **Prediction-based MaxSum Utility**
### Summary of Matching Models

<table>
<thead>
<tr>
<th>Static/Dynamic Scenario</th>
<th>Effect of Spatial Factors</th>
<th>Optimization Target</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tong et al. ICDE16</td>
<td>Constraint</td>
<td>MaxSum Utility</td>
</tr>
<tr>
<td>Tong et al. VLDB16</td>
<td>Target</td>
<td>MinSum Distance</td>
</tr>
<tr>
<td>She et al. TKDE16</td>
<td>Constraint</td>
<td>Conflict-based MaxSum Utility</td>
</tr>
<tr>
<td>Song et al. ICDE17</td>
<td>Constraint</td>
<td>Trichromatic MaxSum Utility</td>
</tr>
<tr>
<td>Tong et al. VLDB17</td>
<td>Constraint</td>
<td>Prediction-based MaxSum Number</td>
</tr>
<tr>
<td>Hassan et al. UIC14</td>
<td>Constraint</td>
<td>MaxSum Utility</td>
</tr>
<tr>
<td>To el al. Percom16</td>
<td>Constraint</td>
<td>MaxSum Utility</td>
</tr>
</tbody>
</table>
Reference: Dynamic Matching


Reference: Dynamic Matching


Static Planning
Static Planning

- **Problem Definition**
- **Background**
  - Orienteering Problem (OP)
- **Existing Research**
  - One-Worker-To-Many-Task Planning
  - Many-Worker-To-Many-Task Planning
Static Planning

- **Problem Definition**
  - Given a set of spatial tasks with some constraints (e.g., deadline), and the *spatiotemporal information of tasks is known*, the problem is to find a *proper route of tasks* for workers.
Static Planning

- Problem Definition

- Background
  - Orienteering Problem (OP)

- Existing Research
  - One-Worker-To-Many-Tasks Planning
  - Many-Worker-To-Many-Tasks Planning
Orienteering Problem (OP)

Given a set of vertices, each associated with a score, a travel budget $b$, a start vertex $s$, an end vertex $e$, and the distance matrix among them, the problem is find a route from $s$ to $e$ such that:

- The total travel distance is no more than the travel budget $b$
- The total score of travelled vertices are maximized

A variant of the Traveling Salesman Problem (TSP)
Static Planning

- Problem Definition

- Background
  - Orienteering Problem (OP)

- Existing Research
  - One-Worker-To-Many-Task Planning
  - Many-Worker-To-Many-Task Planning
One-Worker-To-Many-Tasks Planning

- **Objective:** Find a route for one worker to maximize the number of the tasks satisfying travel/time budgets

- **Difference from Orienteering Problem**
  - The utility score of each task is zero or one
  - The end vertex of the path is not given
One-Worker-To-Many-Tasks Planning

- **Heuristic Methods**
  - Nearest Neighbor first
  - Limitation first (e.g., time expiration)

\[ t_e = 10 \]

One-Worker-To-Many-Tasks Planning

- **Heuristic Methods**
  - Nearest Neighbor first
  - **Limitation first** (e.g., time expiration)

Static Planning

- Problem Definition

- Background
  - Orienteering Problem (OP)

- Existing Research
  - One-Worker-To-Many-Task Planning
  - Many-Worker-To-Many-Task Planning
Many-Worker-To-Many-Tasks Planning

- **Objective:** Find routes for all workers to maximize the total utility/number satisfying travel/time budgets

![Diagram of worker routes]

J. She, Y. Tong, L. Chen. Utility-aware event-participant planning. SIGMOD 2015
Many-Worker-To-Many-Tasks Planning

- Greedy-based Algorithm
  - Greedily choose the pair $<w,t>$ with the maximum utility/travel cost ratio for each worker

J. She, Y. Tong, L. Chen. Utility-aware event-participant planning. SIGMOD 2015
Dynamic Planning

- Static Matching
- Dynamic Matching
- Static Planning
- Dynamic Planning
Dynamic Planning

Problem Definition (One-worker-to-many-tasks)

- Given one worker and a set of spatial tasks, which are dynamically released and have some constraints (e.g., deadline), the problem is find a route for the worker such that the number of the finished tasks is maximized.

- The route can only be updated when a task is released.
Dynamic Planning

- Task-layer Greedy
  - Greedily choose the next nearest task after finishing the current task

There is no constant competitive ratio

Y. Li, M. Yiu, W. Xu. Oriented online route recommendation for spatial crowdsourcing task workers. SSTD 2015.
Reference: Planning


## Summary

<table>
<thead>
<tr>
<th>Computation Complexity</th>
<th>Matching Model</th>
<th>Planning Model</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Static Scenario</strong></td>
<td>Weighted Bipartite Matching [Polynomial Time Solvable Problem]</td>
<td>Traveling Salesman Problem (TSP) or Orienteering Problem (OP) [NP-hard Problem]</td>
</tr>
<tr>
<td><strong>Dynamic Scenario</strong></td>
<td>Constant-competitive-ratio Online Algorithms</td>
<td>Non-constant-competitive-ratio Online Algorithms</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Workload per Worker</th>
<th>One Task</th>
<th>Multiple Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Representative Applications</td>
<td>Taxi Calling Service Bike Sharing</td>
<td>Food Delivery Service Ridesharing Service</td>
</tr>
</tbody>
</table>
# Summary

<table>
<thead>
<tr>
<th>Computation Complexity</th>
<th>Matching Model</th>
<th>Planning Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static Scenario</td>
<td>Weighted Bipartite Matching [Polynomial Time Solvable Problem]</td>
<td>Traveling Salesman Problem (TSP) or Orienteering Problem (OP) [NP-hard Problem]</td>
</tr>
<tr>
<td>Dynamic Scenario</td>
<td>Constant-competitive-ratio Online Algorithms</td>
<td>Non-constant-competitive-ratio Online Algorithms</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Workload per Worker</th>
<th>One Task</th>
<th>Multiple Tasks</th>
</tr>
</thead>
</table>

| Representative Applications | Taxi Calling Service Bike Sharing | Food Delivery Service Ridesharing Service |
## Summary

<table>
<thead>
<tr>
<th>Computation Complexity</th>
<th>Matching Model</th>
<th>Planning Model</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Static Scenario</strong></td>
<td>Weighted Bipartite Matching [Polynomial Time Solvable Problem]</td>
<td>Traveling Salesman Problem (TSP) or Orienteering Problem (OP) [NP-hard Problem]</td>
</tr>
<tr>
<td><strong>Dynamic Scenario</strong></td>
<td>Constant-competitive-ratio Online Algorithms</td>
<td>Non-constant-competitive-ratio Online Algorithms</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Workload per Worker</th>
<th>Matching Model</th>
<th>Planning Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>One Task</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Multiple Tasks</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Representative Applications</th>
<th>Matching Model</th>
<th>Planning Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Taxi Calling Service Bike Sharing</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Food Delivery Service Ridesharing Service</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Summary

<table>
<thead>
<tr>
<th>Computation Complexity</th>
<th>Matching Model</th>
<th>Planning Model</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Static Scenario</strong></td>
<td>Weighted Bipartite Matching [Polynomial Time Solvable Problem]</td>
<td>Traveling Salesman Problem (TSP) or Orienteering Problem (OP) [NP-hard Problem]</td>
</tr>
<tr>
<td><strong>Dynamic Scenario</strong></td>
<td>Constant-competitive-ratio Online Algorithms</td>
<td>Non-constant-competitive-ratio Online Algorithms</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Workload per Worker</th>
<th>One Task</th>
<th>Multiple Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Representative Applications</strong></td>
<td>Taxi Calling Service Bike Sharing</td>
<td>Food Delivery Service Ridesharing Service</td>
</tr>
</tbody>
</table>
### Summary

<table>
<thead>
<tr>
<th>Computation Complexity</th>
<th>Matching Model</th>
<th>Planning Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static Scenario</td>
<td>Weighted Bipartite Matching [Polynomial Time Solvable Problem]</td>
<td>Traveling Salesman Problem (TSP) or Orienteering Problem (OP) [NP-hard Problem]</td>
</tr>
<tr>
<td>Dynamic Scenario</td>
<td>Constant-competitive-ratio Online Algorithms</td>
<td>Non-constant-competitive-ratio Online Algorithms</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Workload per Worker</th>
<th>One Task</th>
<th>Multiple Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Representative Applications</td>
<td>Taxi Calling Service Bike Sharing</td>
<td>Food Delivery Service Ridesharing Service</td>
</tr>
</tbody>
</table>
Outline

- Overview of Spatial Crowdsourcing (20min)
  - Motivation
  - Workflow
  - Core Issues
  - Difference from Related Tutorials
- Fundamental Techniques (40min)
  - Task Assignment
  - Quality Control
  - Incentive Mechanism
  - Privacy Protection
- Spatial Crowdsourced Applications (20min)
  - Spatial Crowdsourcing Intrinsic Applications
  - Crowd-powered Spatial Applications
- Open Questions (10min)
Quality Control

- Why Quality Control

- Existing Research
  - Spatiotemporal-constraint-based Quality Control
  - Spatiotemporal-target-based Quality Control
    - Minimum Latency
    - Maximum Spatiotemporal Diversity

- Summary
Why Quality Control

- Traditional crowdsourcing
  - Erroneous
    - “To err is human” by Marcus Tullius Cicero
  - Existing research
    - Estimate the quality of workers
    - Infer the truth of tasks

- Spatial crowdsourcing
  - Spatiotemporal factors as constraints
    - Workers still err / truth of tasks still need to be inferred
    - Spatiotemporal factors are only used as constraints
  - Spatiotemporal factors as goals
    - Quality of tasks is assessed by spatiotemporal factors
Quality Control

- Why Quality Control

- Existing Research
  - Spatiotemporal-constraint-based Quality Control
  - Spatiotemporal-target-based Quality Control
    - Minimum Latency
    - Maximum Spatiotemporal Diversity

- Summary
Given a set of tasks and a set of workers with their reputation, the problem is to maximize the total number of assigned tasks such that

- Quality requirement of each task is satisfied
- Spatiotemporal constraint: tasks should locate in the service range of the assigned workers
Spatial-constraint-based QC

- Quality requirement depends on quality control methods e.g. Majority Voting

Aggregate Reputation Score (ARS):

\[
ARS(Q) = \sum_{k=\lceil |Q|/2 \rceil + 1}^{\lceil |Q| \rceil} \sum_{A \subset F_k} \prod_{w_j \in A} r_j \prod_{w_j \notin A} (1 - r_j)
\]

Quality Control

- Why Quality Control

- Existing Research
  - Spatiotemporal-constraint-based Quality Control
  - **Spatiotemporal-target-based Quality Control**
    - Minimum Latency
    - Maximum Spatiotemporal Diversity

- Summary
Quality Control

- Why Quality Control

- Existing Research
  - Spatiotemporal-constraint-based Quality Control
  - Spatiotemporal-target-based Quality Control
    - Minimum Latency
    - Maximum Spatiotemporal Diversity

- Summary
Minimum Latency

- Latency is a more important concern in spatial crowdsourcing than in traditional crowdsourcing.

**Latency:** only care the longest completion time of the tasks

**Spatial Crowdsourcing**

**Latency:** also care the total completion time of all the tasks

(Internet) Crowdsourcing
Minimum Latency

- Latency is a more important concern in spatial crowdsourcing than in traditional crowdsourcing.

- Solutions
  - Modelled and solved by online matching or online planning.

Online Matching
Online Planning
Quality Control

- Why Quality Control

- Existing Research
  - Spatiotemporal-constraint-based Quality Control
  - Spatiotemporal-target-based Quality Control
    - Minimum Latency
    - Maximum Spatiotemporal Diversity

- Summary
Maximum Spatiotemporal Diversity

- Spatial Diversity

Take photos of a landmark

Post Task

P. Cheng, X. Lian, Z. Chen, R. Fu, L. Chen, J. Han, J. Zhao. Reliable Diversity-Based Spatial Crowdsourcing by Moving Workers. VLDB 2015.
Maximum Spatiotemporal Diversity

Temporal Diversity

Monitor available parking spaces over a time period

Begin
2:00PM

End
6:00PM

P. Cheng, X. Lian, Z. Chen, R. Fu, L. Chen, J. Han, J. Zhao. Reliable Diversity-Based Spatial Crowdsourcing by Moving Workers. VLDB 2015.
Maximum Spatiotemporal Diversity

- Spatiotemporal Diversity

Spatial Diversity is given as

$$SD(t_i) = -\sum_{j=1}^{r} \frac{A_j}{2\pi} \cdot \log\left(\frac{A_j}{2\pi}\right)$$

Temporal Diversity is given as

$$TD(t_i) = -\sum_{j=1}^{r+1} \frac{I_j}{e_i - s_i} \cdot \log\left(\frac{I_j}{e_i - s_i}\right)$$

Balance Spatial Diversity and Temporal Diversity:

$$STD(t_i, W_i) = \beta \cdot SD(t_i) + (1 - \beta) \cdot TD(t_i)$$

The goal is to maximize the spatiotemporal diversity of a given task

P. Cheng, X. Lian, Z. Chen, R. Fu, L. Chen, J. Han, J. Zhao. Reliable Diversity-Based Spatial Crowdsourcing by Moving Workers. VLDB 2015.
Quality Control

- Why Quality Control

- Existing Research
  - Spatiotemporal-constraint-based Quality Control
  - Spatiotemporal-target-based Quality Control
    - Minimum Latency
    - Maximum Spatiotemporal Diversity

- Summary
Summary

- “To err is human”

- Spatiotemporal-constraint-based Quality Control
  - Tasks have quality requirement for truth of answers
  - Workers have spatiotemporal constraints

- Spatiotemporal-target-based Quality Control
  - Minimizing Latency
  - Maximizing Spatiotemporal Diversity
## Reference: Quality Control


Reference: Quality Control


Outline

- Overview of Spatial Crowdsourcing (20min)
  - Motivation
  - Workflow
  - Core Issues
  - Difference from Related Tutorials

- Fundamental Techniques (40min)
  - Task Assignment
  - Quality Control
  - Incentive Mechanism
  - Privacy Protection

- Spatial Crowdsourced Applications (20min)
  - Spatial Crowdsourcing Intrinsic Applications
  - Crowd-powered Spatial Applications

- Open Questions (10min)
Incentive Mechanism

- **Motivation**

- **Existing Research**
  - Monetary Reward
    - Game Theory based incentive mechanism
    - Auction based incentive mechanism
  - Gamification
  - Volunteer

- **Summary**
Incentive Mechanism

- Stimulate workers to complete tasks efficiently using proper reward
- A tradeoff between workers and tasks
  - Excessively low reward discourages the workers to finish the task
  - Excessively high reward hurts the benefit of the platform

Platform

It hurts my benefit because of the high reward

Too little reward that I cannot work efficiently
Incentive Mechanism: Design Space

Task Execution Manner

Active

Passive

Monetary Reward

Gamification

Volunteer

Incentive

Uber

Tourality

Open Street Map

Ingress

Waze
Incentive Mechanism

- Motivation

- Existing Research
  - Monetary Reward
    - Game Theory based incentive mechanism
    - Auction based incentive mechanism
  - Gamification

- Volunteer

- Summary
Incentive Mechanism

- Motivation

- Existing Research
  - Monetary Reward
    - Game Theory based incentive mechanism
    - Auction based incentive mechanism
  - Gamification
  - Volunteer

- Summary
Game Theory Based Mechanism

- **Platform-Centric Model**
  - We need to compete for the total budget
  - I have a spatial task that needs to be completed collaboratively by multiple workers
  - I also have a total budget

  The total budget is allocated to the workers according to their contributions.
Game Theory Based Mechanism

- Platform-Centric Model

I have a small unit cost

I have a large unit cost

Platform
Game Theory Based Mechanism

- Stackelberg Game
  - Leader
  - Followers

- Stackelberg Equilibrium
  - I have the knowledge of the followers’ behavior, and want to maximize my utility

Game Theory Based Mechanism

Step 1: Estimate an optimal total budget and publish it

Step 2: Submit their actual plans of performing the tasks

Step 3: Allocate the budget to the workers according to their actual plans

Incentive Mechanism

- Motivation

- Existing Research
  - Monetary Reward
    - Game Theory based incentive mechanism
    - Auction based incentive mechanism
  - Gamification

- Volunteer

- Summary
Background of Auction

- Auction
  - Buyers compete to obtain goods or services by offering increasingly higher prices.
Background of Auction

- **Auction**
  - Buyers compete to obtain goods or services by offering increasingly higher prices

- **Reverse auction**
  - The sellers compete to obtain a business from the buyer and prices will typically decrease as the sellers underbid each other

I want a pc

Buyer

My price is 700$

My price is 600$

My price is only 500$

Seller

Seller

Seller
Auction Based Mechanism

Objective

- Given a set of workers dynamically coming, each of which is associated with a bidding price and a utility score, the platform needs to hire $m$ workers and irrevocably decide the payment of accepted workers to maximize the total utility.

$w_1 = 10, b_1 = 10$

$w_2 = 6, b_2 = 8$

$w_3 = 6, b_3 = 6$

$w_4 = 4, b_4 = 4$

Worker (Seller)

Task (Buyer)

Platform

$m = 2$

Too far to accomplish the task

Auction Based Mechanism

- State-of-the-art
  - Based on the secretary problem

- Secretary Problem
  - $n$ applicants, which can be ranked in strictly total order, apply for a single position. The applicants are interviewed sequentially and immediately after an interview, the interviewed applicant is either accepted or rejected irrevocably. The objective is to have the highest probability of selecting the best applicant of the whole group.

- Solution of Secretary Problem
  - Reject the first $n/e$ applicants and hire the first applicant thereafter who has a higher score than all preceding applicants.
  - With probability $1/e$ we can select the best applicant.
Auction Based Mechanism

- **Threshold-based Auction**
  - Filter and observe the first half of workers and calculate a $m$-dimension threshold vector $\delta$.
  - For the second half of workers, the framework decides whether the ratio of utility over bidding price by the new arrival worker is larger than the corresponding threshold in the vector $\delta$.
  - If so, the platform hires the worker, otherwise rejects the worker.
Threshold-based Auction

- Filter and observe the first half of workers and calculate a $m \times 1$-dimension threshold vector $\delta$
- For the second half of workers, the platform decides whether to pay according to the vector $\delta$

Platform

$n = 6$

$m = 2$

$\delta = < 7, 4 >$

Incentive Mechanism

- Motivation

- Existing Research
  - Monetary Reward
    - Game Theory based incentive mechanism
    - Auction based incentive mechanism
  - Gamification

- Volunteer

- Summary
Incentive Mechanism: Gamification

- **Ingress (Niantic, Google)**
  - A location-based mobile game
  - The gameplay consists of capturing "portals" at places of cultural significance such as public art
  - Portals are set at specific positions to collect data during the capturing process using phones’ camera

Data from Ingress was used to populate the locations for Pokéstops and gyms in Pokémon Go, released in July 2016.
Incentive Mechanism: Gamification

- **Pokémon Go (Niantic, Nintendo)**
  - A location-based, augmented-reality mobile game
  - The game utilizes the player's mobile device to locate and capture virtual creatures called Pokémon
  - Pokémon can be set at specific positions to collect data during the capturing process using phones’ camera

Incentive Mechanism

- Motivation
- Existing Research
  - Monetary Reward
    - Game Theory based incentive mechanism
    - Auction based incentive mechanism
  - Gamification
- Volunteer
- Summary
Incentive Mechanism: Volunteer

- **Open Street Map**
  - A collaborative project to create a free editable map
  - A prominent example of volunteered geographic information
  - Map data is collected from scratch by volunteers using tools such as a mobile phone, handheld GPS unit, a notebook, digital camera, or a voice recorder

Incentive Mechanism

- Motivation

- Existing Research
  - Monetary Reward
    - Game Theory based incentive mechanism
    - Auction based incentive mechanism
  - Gamification

- Volunteer

- Summary
Summery

- Workers always need to be motivated

- Tradeoff between workers and tasks

- State-of-the-art techniques
  - Monetary-Reward-based incentives
  - Gamification-based incentives
  - Volunteer-based incentives


Outline

- Overview of Spatial Crowdsourcing (20min)
  - Motivation
  - Workflow
  - Core Issues
  - Difference from Related Tutorials

- Fundamental Techniques (40min)
  - Task Assignment
  - Quality Control
  - Incentive Mechanism
  - Privacy Protection

- Spatial Crowdsourced Applications (20min)
  - Spatial Crowdsourcing Intrinsic Applications
  - Crowd-powered Spatial Applications

- Open Questions (10min)
Privacy Protection

- **Motivation & A Unified Framework**

- **Existing Research**
  - Cloaked Locations-based Protection
  - Differential Privacy-based Protection
  - Encrypted Data-based Protection

- **Summary**
Motivation and Framework

- **Motivation**
  - Protect the privacy of workers’ locations
- **A unified framework**
  - The locations of the workers are transformed by some techniques
  - The platform performs task assignment based on the transformed locations of the tasks
  - The workers confirm/refine the task assignment results based on their true locations
Motivation and Framework

- **Motivation**
  - Protect the privacy of workers’ locations
- **A unified framework**
  - The locations of the workers are transformed by some techniques
  - The platform performs task assignment based on the transformed locations of the tasks
  - The workers confirm/refine the task assignment results based on their true locations

*Diagram showing the process of task assignment with transformed location data.*
Motivation and Framework

- Motivation
  - Protect the privacy of workers’ locations
- A unified framework
  - The locations of the workers are transformed by some techniques
  - The platform performs task assignment based on the transformed locations of the workers
  - The workers confirm/refine the task assignment results based on their true locations
Motivation and Framework

- **Motivation**
  - Protect the privacy of workers’ locations
  - A unified framework
    - The locations of the workers are transformed by some techniques
    - The platform performs task assignment based on the transformed locations of the tasks
    - The workers confirm/refine the task assignment results based on their true locations
Privacy Protection

- Motivation & A Unified Framework

- Existing Research
  - Cloaked Locations-based Protection
  - Differential Privacy-based Protection
  - Encrypted Data-based Protection

- Summary
Cloaked Area Based Protection

- The location of the worker is transformed as a **cloaked area**
  - A cloaked area is a pair \(<a, f>\), where \(a\) is a spatial range and \(f\) is the probability density function of the worker at each point in \(a\)

L. Pournajaf, L. Xiong, V. S. Sunderam, X. Xu. STAC: spatial task assignment for crowd sensing with cloaked participant locations. GIS 2015.
Privacy Protection

- Motivation & A Unified Framework

- Existing Research
  - Cloaked Locations-based Protection
  - Differential Privacy-based Protection
  - Encrypted Data-based Protection

- Summary
Differential Privacy based Protection

- **Differential Privacy (DP)**
  - Basic idea: limit the impact of any particular record on the output
  - **Definition**
    - \( \Pr[(X) \in ] \leq e^\varepsilon \Pr[(X') \in ] \)
    - \( \varepsilon > 0 \) controls the level of privacy

- **DP in spatial data**
  - Protect location information of **points**
  - Protect location information of **trajectories**
DP: Point Location Protection

- **Point** location protection
- **Core Technique**
  - Private spatial decompositions (PSD)
  - Decompose a geometric space, data points partitioned among the leaves

A trusted third party sanitizes the location of workers according to differential privacy techniques and release it.

The platform performs task assignment according to PSD.

Workers refine the assignment using their exact locations.

- Workers send their locations to a trusted third party.
- The third party sanitizes the location of workers according to differential privacy techniques and release it.
- The platform performs task assignment according to PSD.
- Workers refine the assignment using their exact locations.

DP: Point Location Protection

- **Workers send their locations to a trusted third party**
- **The third party sanitizes the location of workers according to differential privacy techniques and release it**
- **The platform performs task assignment according to PSD**
- **Workers refine the assignment using their exact locations**

Workers send their locations to a trusted third party
The third party sanitizes the location of workers according to differential privacy techniques and release it
The platform performs task assignment according to PSD
Workers refine the assignment using their exact locations

DP: Point Location Protection

- Workers send their locations to a trusted third party
- The third party sanitizes the location of workers according to differential privacy techniques and release it
- The platform performs task assignment according to PSD
- Workers refine the assignment using their exact locations

DP: Point Location Protection

- Workers send their locations to a trusted third party
- The third party sanitizes the location of workers according to differential privacy techniques and release it
- The platform performs task assignment according to PSD
- Workers refine the assignment using their exact locations

**DP: Trajectory Location Protection**

- **Trajectory** location protection
  - Structure of PSD is computed at the first time instance and is reused for all remaining instances (use a fixed privacy budget)
  - The remaining budget is used for each grid cell across multiple time instances using FAST – a time-series perturbation technique

---


Privacy Protection

- Motivation & A Unified Framework

- Existing Research
  - Cloaked Locations-based Protection
  - Differential Privacy-based Protection
  - Encrypted Data-based Protection

- Summary
Encryption Based Protection

- **Homomorphic Encryption**
  - The *exact* distances between tasks and workers can be computed based on their encrypted locations

- **Secure Indexing**
  - A secure indexing technique which combines homomorphic encryption with KD-tree

Encryption Based Protection

- The locations of tasks and workers are encrypted by homomorphic encryption.
- The platform performs task assignment based on the distances computed by the encrypted data.
- The workers receive the encrypted location of the task assigned by the platform, and decrypts it to get the location.

Encryption Based Protection

- The locations of tasks and workers are encrypted by homomorphic encryption.
- The platform performs task assignment based on the distances computed from the encrypted data.
- The workers receive the encrypted location of the task assigned by the platform, and decrypts it to get the location.

Locations of both the workers and the tasks are protected.

The locations of tasks and workers are encrypted by homomorphic encryption.

The platform performs task assignment based on the distances computed by the encrypted data.

The workers receive the encrypted location of the task assigned by the platform, and decrypts it to get the location.
The locations of tasks and workers are encrypted by homomorphic encryption.

The platform performs task assignment based on the distances computed by the encrypted data.

The workers receive the encrypted location of the task assigned by the platform, and decrypts it to get the location.
Privacy Protection

- Motivation & A Unified Framework

- Existing Research
  - Cloaked Locations-based Protection
  - Differential Privacy-based Protection
  - Encrypted Data-based Protection

- Summary
Summary

- **Challenge**
  - Location Security vs. Distance Accuracy

- **Framework**
  - Transform the locations of the workers
  - The platform performs task assignment based on the transformed locations of the workers
  - The workers confirm/refine the task assignment results based on their true locations

- **State-of-the-art techniques**
  - Cloaked Locations-based protection
  - Differential Privacy-based protection
  - Encrypted Data-based protection
Reference: Privacy Protection


Outline

- Overview of Spatial Crowdsourcing (20min)
  - Motivation
  - Workflow
  - Core Issues
  - Difference from Related Tutorials
- Fundamental Techniques (50min)
  - Task Assignment
  - Quality Control
  - Incentive Mechanism
  - Privacy Protection
- Spatial Crowdsourced Applications (15min)
  - Spatial Crowdsourcing Intrinsic Applications
  - Crowd-powered Spatial Applications
- Open Questions (5min)
Outline

- Overview of Spatial Crowdsourcing (20min)
  - Motivation
  - Workflow
  - Core Issues
  - Difference from Related Tutorials

- Fundamental Techniques (50min)
  - Task Assignment
  - Quality Control
  - Incentive Mechanism
  - Privacy Protection

- Spatial Crowdsourced Applications (15min)
  - Spatial Crowdsourcing Intrinsic Applications
  - Crowd-powered Spatial Applications

- Open Questions (5min)
Intrinsic Applications

- What are the **Intrinsic Applications**
  - Naturally modelled by spatial crowdsourcing
  - Share the same core issues with spatial crowdsourcing

- Two kinds of representative applications
  - Real-time taxi calling service
    - Uber
    - DiDi
  - Food delivery service
    - Yelp
    - Grubhub
Intrinsic Applications

- Real-time taxi calling service

Passenger is the task requester

The task is to pick up the passenger at location A and send her/him to location B
Intrinsic Applications

- Real-time taxi calling service

A typical dynamic matching problem

Passengers and available taxis dynamically appear on the platform

Uber is the platform, and its core concern is how to assign taxis to pick up different passengers efficiently
Intrinsic Applications

- Real-time taxi calling service
Intrinsic Applications

- Food delivery service

User is the task requester

The task is to pick up the food and send the food to the user
Intrinsic Applications

- Food delivery service

A typical **dynamic planning problem**

Grubhub is the platform, and its core concern is how to design effective plans for the deliverers.

Orders and available deliverers dynamically appear on the platform.
Intrinsic Applications

- Food delivery service

Orders and available deliverers dynamically appear on the platform

Task Assignment

Platform

Incentive Mechanism

Quality Control

Privacy Protection
Intrinsic Applications

- Ridesharing Service
Intrinsic Applications
Outline

- Overview of Spatial Crowdsourcing (20min)
  - Motivation
  - Workflow
  - Core Issues
  - Difference from Related Tutorials

- Fundamental Techniques (50min)
  - Task Assignment
  - Quality Control
  - Incentive Mechanism
  - Privacy Protection

- Spatial Crowdsourced Applications (15min)
  - Spatial Crowdsourcing Intrinsic Applications
  - Crowd-powered Spatial Applications

- Open Questions (5min)
Outline

- Crowd-powered Spatial Applications
  - Crowdsourced Path Selection
  - Crowdsourced Speed Estimation
  - Crowdsourced POI Labelling
Path Selection

- **Motivation**
  - Route recommendation service is indispensable in daily life

- **Solutions** *without crowdsourcing*
  - Using popular routes mined from historical trajectories as recommended routes
    - Most Popular Route
    - Local Driver Route
    - Most Frequent Path

- **Drawbacks**
  - **Sparsity**: insufficient historical trajectories for inference
  - **Diversity**: different algorithms, different routes

*I need more data!*

![Graph showing similarity proportion over length (km)](image-url)
Path Selection

- **Solutions using spatial crowdsourcing**
  - Leverage crowds’ knowledge to improve the recommendation quality

- **Main Idea**
  - Consolidate candidate routes from different sources (e.g., map service providers, popular routes)
  - Request experienced drivers to select amongst them

---

Outline

- Crowd-powered Spatial Applications
  - Crowdsourced Path Selection
  - Crowdsourced Speed Estimation
  - Crowdsourced POI Labelling
Traffic Speed Estimation

- **Motivation**
  - Real-time traffic speed represents the congestion of road and is one of the most important aspects for traffic monitoring

- **Speed Estimation without Crowdsourcing**
  - Use sensor or trajectory data for speed estimation
  - It is hard to infer the speed of remote roads
Traffic Speed Estimation

- **Solutions using Spatial Crowdsourcing**
  - Assign crowd workers to physically check traffic speed of some ambiguous roads or remote roads.

- **Key Idea**
  - A two-layer framework for speed estimation:
    - Assign $k$ roads (called seed road) to crowd workers to generate their real speeds.
    - Infer speeds of other roads according to seed roads and historical speed information.

---

Outline

- Crowd-powered Spatial Applications
  - Crowdsourced Path Selection
  - Crowdsourced Speed Estimation
  - Crowdsourced POI Labelling
POI Labelling

- Motivation
  - POI (Point of Interest) is one of the most useful and fundamental spatial data

- Incorrect labels exist
  - Low quality POI labels from volunteers
  - Limited accuracies of AI algorithms
POI Labelling

- Solutions using Spatial Crowdsourcing
  - An iterative framework
    - Assign tasks to workers
    - Collect the results and infer the truth of POIs
    - Repeat until the budget is exhaustive

POI Labelling

- Experimental Findings
  - The distance between POIs and workers has a huge impact on POI labelling

The longer distance, the lower accuracy

### Summary

<table>
<thead>
<tr>
<th>Spatial crowdsourced intrinsic applications</th>
<th>Crowd-powered spatial applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Taxi Calling</td>
<td>SC Path Selection</td>
</tr>
<tr>
<td>Ride Sharing</td>
<td>SC Speed Estimation</td>
</tr>
<tr>
<td>Food Delivery</td>
<td>SC POI Labelling</td>
</tr>
</tbody>
</table>

| Task Assignment                          | ✔️                               |
| Quality Control                          | ✔️                               |
| Incentive Mechanism                      | ✔️                               |
| Privacy Protection                       | ✔️                               |

**Spatial crowdsourced intrinsic applications:** Pure human effort

**Crowd-powered spatial applications:** Hybrid human-machine effort
Reference: Applications


Outline

- Overview of Spatial Crowdsourcing (20min)
  - Motivation
  - Workflow
  - Core Issues
  - Difference from Related Tutorials

- Fundamental Techniques (50min)
  - Task Assignment
  - Quality Control
  - Incentive Mechanism
  - Privacy Protection

- Spatial Crowdsourced Applications (15min)
  - Spatial Crowdsourcing Intrinsic Applications
  - Crowd-powered Spatial Applications

- Open Questions (5min)
Question 1: Index

- Abundant indexes in spatial data management
  - Grid, R-tree, Quadtree, kd-tree, etc

- How to leverage existing indexes or design new indexes to optimize the efficiency of fundamental issues in spatial crowdsourcing?
Question 2: Dynamic Scenarios

- Although some recent research has focused on dynamic task assignment, there are several big gaps between theory and practice.
  - E.g., good performance in practice versus bad theoretical results of the simple greedy algorithm for the online minimum matching problem.
Question 2: Dynamic Scenarios

- The research of other core issues of spatial crowdsourcing in dynamic scenario is limited
  - How to design adaptive incentive mechanisms which can handle the dynamic supply and demand between tasks and workers?
  - How to protect the location privacy of the dynamic arrival tasks and workers
Question 3: Benchmark

- Abundant benchmarks for many classical spatial data management
  - E.g., DIMACS for shortest path

- Although there are a few synthetic data generators for spatial crowdsourcing, public real datasets are still inadequate
  - Platforms owning abundant real data are usually commercial, and would not like to open their data
  - Open sourced platforms do not have enough money recruitment enough workers
Thank you