MobiSensing: Exploiting Human Mobility for Multi-Application Mobile Data Sensing with Low User Intervention*

Kang Chen¹ and Haiying Shen²

¹Dept. of ECE, Southern Illinois University, IL, USA
²Dept. of ECE, Clemson University, SC, USA

* Work was done when at Clemson
Outline

• Introduction
• System Design
• Performance Evaluation
• Conclusion
Introduction

• Explosive growth of mobile devices
  • Smart phones, tablets, and wearable devices
  • Around 2 billion smart phone users now

• Rich sensing & computing capability on those mobile devices
  • GHz level CPUs
  • Various sensors: GPS/Microphone/Accelerometer
Introduction

• Inspires mobile device based sensing systems
  • Exploit otherwise underutilized resources
  • Does not need dedicated infrastructure
  • A bunch of existing endeavors
    • Bus arrival time prediction (MobiSys’12)
    • Red light advisory (MobiSys’11)
    • Remote sensing (MobiSys’10)
    • Road status sensing and monitoring (SenSys’08)
    • etc.
Introduction

• Mobile devices holders are not dedicated
  • Do not want to move or operate purposely for sensing
  • Want to be interrupted as few as possible
  • A major barrier for such distributed sensing systems

• The passive mode appears to be more attractive
  • Collect data without explicit user interruption
  • Can exploit more resources
Introduction

• How to realize passive data sensing effectively?
  • Accurate mobility prediction
    • Understand where mobile devices are going
  
  • Efficient task assignment
    • Send the task to the device that is most likely to finish it
  
  • Reduce explicit user participation as much as possible
    • Enhance the accuracy of the above two steps “silently”
Introduction

• MobiSensing is proposed in this direction
  • Exploit the heterogeneous semi-markov model for mobility modeling

• Mobility model based task assignment
  • A task is always assigned to devices that are most likely to complete them

• Exploit mobile device’s intermittent connection to server for
  • Mobility model refinement
  • Task assignment and data collection
Outline

• Introduction
• **System Design**
• Performance Evaluation
• Conclusion
System Design: Mobility Modeling

• Treat device mobility as transition between landmarks
  - $\mathcal{L} = \{L_1, L_2, L_3, ..., L_m\}$: landmark space consisting of $m$ landmarks

• Using semi-Markov process (SMP) to model those transitions
  - $X_n$: the status (i.e., which landmark) after the $n$-th transition
  - $T_n$: the time when the node changes to $X_n$
  - Kernel:
    \[
    Q_{ij}(t) = P[X_{n+1} = L_j, T_{n+1} - T_n \leq t | X_n = L_i]
    \]
  - It means the probability of transiting to $L_j$ after staying at $L_i$ for less than $t$ time slots
System Design : Mobility Modeling

• How to obtain $Q_{ij}(t)$?

\[
Q_{ij}(t) = P[X_{n+1} = L_j, T_{n+1} - T_n \leq t | X_n = L_i] \\
= P[X_{n+1} = L_j | X_n = L_i] \cdot P[T_{n+1} - T_n \leq t | X_{n+1} = L_j, X_n = L_i] \\
= p_{ij} U_{ij}(t)
\]

• $p_{ij}$: the probability of transiting to $L_j$ after staying at $L_i$  
  \[
p_{ij} = \frac{V_{ij}}{V_i}
\]

• $V_{ij}$: the number of transits from $L_i$ to $L_j$

• $V_i$: the number of transits leaving $L_i$
System Design : Mobility Modeling

• How to obtain $Q_{ij}(t)$?

$$Q_{ij}(t) = P[X_{n+1} = L_j, T_{n+1} - T_n \leq t | X_n = L_i]$$

$$= P[X_{n+1} = L_j | X_n = L_i] \times P[T_{n+1} - T_n \leq t | X_{n+1} = L_j, X_n = L_i]$$

$$= p_{ij} U_{ij}(t)$$

• $U_{ij}(t)$: the probability that the sojourn time at $L_i$ is no more than $t$ on the condition that the node transits to $L_j$ after staying at $L_i$

$$U_{ij}(t) = V_{ij}(t) / V_{ij}$$

• $V_{ij}(t)$: the number of transits from $L_i$ to $L_j$ with the sojourn time at $L_i$ less than $t$

• $V_{ij}$: the number of transits from $L_i$ to $L_j$
System Design : Mobility Modeling

- Usage of $Q_{ij}(t)$
  - Deduce $\Phi_{ij}(t)$: the probability of at status $L_j$ after $t$ time slots if the node is at status $L_i$ at time 0.
  - Refer to the paper for the detail of getting $\Phi_{ij}(t)$

- Usage of $\Phi_{ij}(t)$
  - If a node is at $L_i$ now, it has a probability of $\Phi_{ij}(t)$ to be at $L_j$ after $t$ time slots
  - Use it to determine what tasks can be assigned to the node
A sensing task is a tuple with four elements:

\{\text{DataType, Location, Date, TimeSlot}\}

- The \textit{DataType} must be supported by mobile devices
- The \textit{Location} refers to landmarks
- Time is slotted
- A job that spanning a longer period of time is divided into consecutive sensing tasks
System Design : Task Assignment

• Mobility prediction based task assignment
  • Predict future mobility

• How long we need to look into the future?
  • User mobility has uncertainty
  • The longer we look, the low prediction accuracy we have
  • The prediction for the next hour is more accurate that for 1 day later
System Design : Task Assignment

• How long we need to look into the future?
  • Exploit user’s intermittent connection to the network (WiFi or LTE)
  • Conduct a task assignment at each connection
  • Only need to predict the mobility between this connection and the next connection
    • Between $T_{ik}$ and $T_{i(k+1)}$, where $T_{ik}$ denotes the $k$-th connection to the server
    • Can refine the accuracy of mobility prediction
    • Can reduce the overhead, as we only need to consider sensing tasks in this period of time
System Design: Task Assignment

• When is the next connection?
  • It is unknown as we do not require user participation
    • To reduce user interruption
  • Predict it with historical data
    • Using the average interval between two task assignments
    • \( \bar{F}_{ik} \) in the figure

\[ T_{ik} \quad F_{ik} \quad \bar{F}_{ik} \quad T'_{i(k+1)} \]

- k-th task assignment
- expected (k+1)-th task assignment
System Design : Task Assignment

• Incorrectly predicted next task assignment may miss tasks
  • When actual next task assignment happens after the predicted one, i.e., $T_i(k+1)$ is after $T'_i(k+1)$
  • As we only consider tasks between $T_{ik}$ and $T'_{i(k+1)}$, tasks between $T'_{i(k+1)}$ and $T_i(k+1)$ will not be assigned to any node
System Design : Task Assignment

- Solve the problem with a redundancy factor $\alpha$
  - Consider tasks in the period of $\alpha F_{ik}$
  - Can effectively reducing the chance of missing tasks
  - Duplicate period of time is fine as it only indicates a little more overhead
System Design : Task Assignment

• Summary of task assignment
  • When a node connects to the server, determine the period of future time slots to consider, i.e., $αF_{ik}$

  • Calculate the node’s probability of appearing at each landmark, i.e., $Φ_{ij}(t)$ at time slots $1, 2, 3, ..., αF_{ik}$ for landmarks $L_j, j = 1, 2, 3, ..., m$,

  • Tasks corresponding to $Φ_{ij}(t)$ that is larger than a threshold is assigned to the node
Outline

- Introduction
- System Design
- Performance Evaluation
- Conclusion
Evaluation

• **Traces**
  - MIT Reality: encountering of mobile devices on a campus
  - DieselNet: encountering of buses in a community

• **Metrics**
  - Success rate: percentage of successfully completed sensing tasks
  - Overhead: total # of assigned tasks

• **Methods**
  - Optimal: assign a copy of a task to every node
  - MaxFreq: assign a task to nodes most frequently visit the target landmark
Evaluation: MIT Reality Trace

- Success Rate:
  - MobiSensing
  - MaxFreq
  - Optimal

- Total Number of Assigned Tasks (x10^5):
  - MobiSensing
  - MaxFreq

Graphs show comparison of success rates and total assigned tasks across different task rates.
Evaluation: DieselNet

- MobiSensing leads to a high success rate and low overhead compared to MaxFreq
Conclusion

• The popularity of mobile devices leads to rich sensing capability.
• Ideally to exploit such capability passively with a low interruption to users.

Solution:
• Assign tasks to mobile devices during their intermittent connection to the network.
• Modeling user mobility with semi-Markov process for efficient task assignment.

Future work:
• Privacy protection: not all users are willing to share their locations.
• Social properties: mobility patterns reflected in social structures.
Thank you!
Questions & Comments?