Automatic User Identification Method across Heterogeneous Mobility Data Sources

Wei Cao

Tsinghua University, Beijing, China
Baidu Inc, Beijing, China

May 10, 2016

Joint work with Zhengwei Wu, Dong Wang, Jian Li and Haishan Wu

Outline

Background

- Ubiquitous location based services affect people's daily life deeply.
  - Baidu Map, Uber ...
- Mobility data is now collected routinely at a very large scale.
- The data is usually generated from heterogeneous data sources
  - Different devices, mobile apps, LBS providers ...
- Identifying the trajectories of the same user across different sources:
  - Fundamental ingredient of the mobility data integration.
  - Improve the quality and density of the data.

Q: Is it possible to identify the users across heterogeneous data sources?

Problem Statement

- Trajectory $T = \{p_1, p_2, \ldots, p_{|T|}\}$
  - Sequence of spatio-temporal points in temporal order
- Mobility data set D
  - Collection of trajectories from a single data source.
- Matching trajectories
  - Trajectories T_A and T_B which are collected from different sources
  - T_A and T_B are generated by the same user
  - Then, T_B is the matching trajectory of T_A
- **Objective:**
  - Given: Data sets D_A and D_B (collected from two different sources)
  - Our goal: Find the matching trajectory in D_B for each $T_A \in D_A$.

Related Work

- User trajectories similarity search [1]
  - Retrieve a subset of trajectories with similar patterns
- Human mobility uniqueness[2]
  - Each individual has her/his own mobility pattern
  - People tend to visit the places where they often visited in the past.

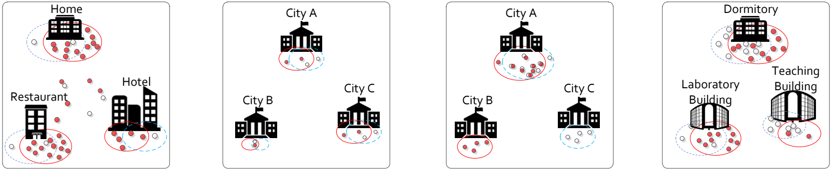
**Few prior work studies the case under heterogeneous data sources.*

[1] Li et al. SIGSPATIAL 2008, Chen et al. SIGMOD 2010, Ranu et al. ICDE 2015

[2] Zang et al. MobiCom 2008, Montjoye et al. Scientific reports. 2013

Challenges

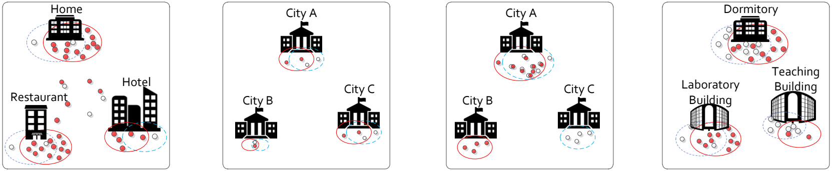
- Different sampling rates
  - Sampling rates of different sources can be extremely different
  - Prior work usually assumes uniform and dense sampling rates.



(a) ***Same person, different sampling rates*** (b) Same person, occurred in several places (c) Same person, disjoint in several cities (d) School mates, significant overlap

Challenges

- Hard to infer the user movements from sparse trajectories
  - Each user only generates one GPS point every 2.63 days on average



(a) Same person, different sampling rates

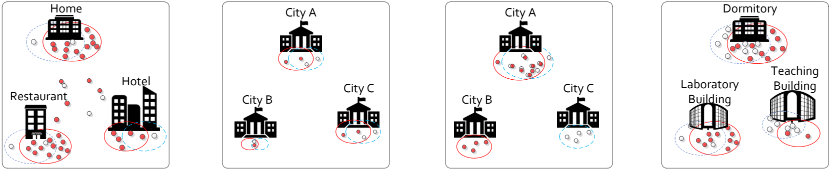
(b) ***Same person, sparse rate, occurred in several places***

(c) Same person, disjoint in several cities

(d) School mates, significant overlap

Challenges

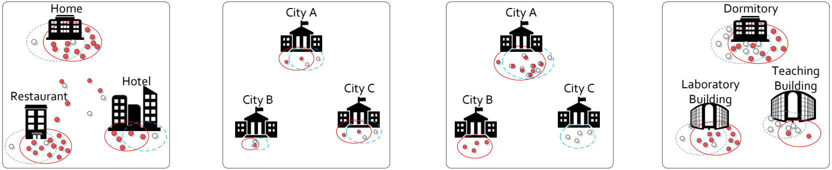
- Data sets can be temporally disjoint
  - They can be collected in different time intervals



(a) Same person, different sampling rates (b) Same person, occurred in several places (c) ***Same person, disjoint in several cities*** (d) School mates, significant overlap

Challenges

- Trajectories of users with close relationships have significant overlaps.
  - Hard to distinguish the users in this case.



(a) Same person, different sampling rates

(b) Same person, occurred in several places

(c) Same person, disjoint in several cities

(d) ***School mates, significant overlap***

Contribution

- Present a Mapreduce-based framework called **Automatic User Identification (AUI)**.
- Design an effective filtering strategy for the large scale data.
- Design a novel similarity measure called **Signal Based Similarity (SIG)**.
- Adopt a rejection strategy to reduce the mis-identification cases.

Outline

Overview

Our system consists of three stages:

- Pre-processing
  - Transform each trajectory into a set of *stay points*
- Multi-resolution filtering
  - Partition the map with multiple resolutions
  - Select a small subset as the candidates of T_A
- Verification
  - Evaluate candidates with SIG
  - Select the matching trajectories carefully

Outline

Pre-processing

Goal: Transform each trajectory into a set of *stay points*

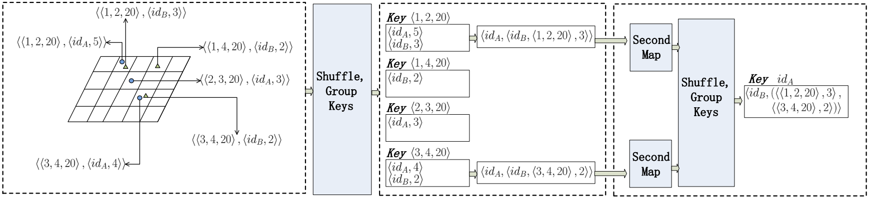
- Recall: it is hard to infer the movement due to the sparsity.
- Accumulate the GPS points during a long period (e.g. half a year)
- Extract the locations where the user stay for a while (*stay points*)
  - Denote the stay point as *sp*
  - $sp.loc \rightarrow$ location of stay point
  - $sp.cnt \rightarrow$ frequency of *sp* occurred in the data

Outline

Multi-Resolution Filtering

Goal: Select a small subset as the candidates of trajectory T_A.

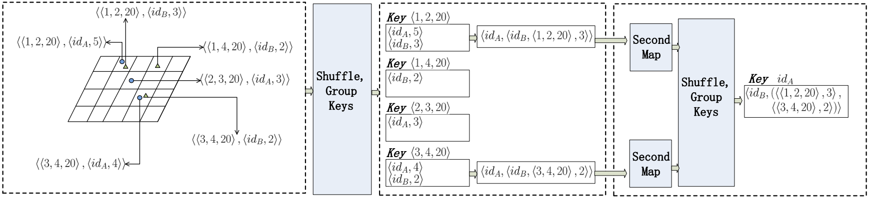
- **First phase**: Find the trajectories which co-occurred with T_A.
- **Map stage**: Input $(\langle T.id, S \rangle)$
  - Partition our map into cells with different granularities.
  - For each stay point $sp \in S$ occurred in cell c
    - Emit $(\langle c, \langle T.id, sp.cnt \rangle \rangle)$ $T.id$ occurred in cell c for $sp.cnt$ times



Multi-Resolution Filtering

Goal: Select a small subset as the candidates of trajectory T_A.

- **Reduce stage**: Input $(\langle c, list(\langle T.id, T.cnt\rangle)\rangle)$
  - For each pair $(T_A.id \in D_A, T_B.id \in D_B)$
    - $\text{output}(T_A.id, \langle T_B.id, c, o\rangle)$ T_A co-occurred with T_B in cell c for o times.



Multi-Resolution Filtering

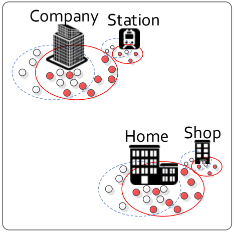
- **Second phase**: Select the candidate set of each trajectory T_A
  - After the first stage, all the co-occurrences are obtained.
  - Suppose trajectory T_B co-occurred with T_A in cell c for o times
    - Add a score $r_c \cdot o$ to T_B
    - The finer granularity they co-occurred at, the larger r_c is.
  - For each T_A, select the top Q trajectory ids with the largest scores as the candidates.
    Set a large Q to ensure the actually matching trajectory is not missing.

Outline

Signal Based Similarity

Description

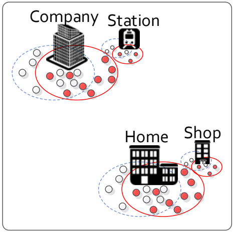
- Take each co-occurrence as a signal.
- The signal indicates whether the trajectories belong to the same user
- Illustration:
  - Two trajectories co-occurred at home/company/station/shop
  - These co-occurrences are signals for user identification



Signal Based Similarity

Description

- Distinguish *observed signal* and *stimulus signal*.
  - Initially, the "kernel cells" emits a positive stimulus signal.
  - The stimulus signal spreads out with an attenuation factor $\alpha < 1$
  - The observed signal is the superposition of decaying stimulus signals.
- Stimulus signals can better capture the mobility pattern.
- Our goal: Recover the stimulus signals from the observations.



Algorithm

- $(c_1, o_1), \ldots, (c_m, o_m)$: observed co-occurrences (in arbitrary order).
- Calculate the observed signal in cell c_k: $\mathrm{ob}(c_k)$

$$\mathrm{ob}(c_k; \eta, \gamma) = \frac{\eta}{1 + e^{-\gamma o_k}} - \frac{\eta}{2}$$

- Recover the stimulus signals $\mathrm{st}(c_k)$ approximately:

$$\mathrm{st}(c_k) = \max \begin{cases} \mathrm{ob}(c_k) - \sum_{l<k} \mathrm{st}(c_l) \cdot \alpha^{\mathrm{Dis}_{\mathrm{grid}}(c_k, c_l)} \\ 0 \end{cases}$$

- Kernel cells: $K = \{k : \mathrm{st}(c_k) > 0\}$

Algorithm

- We also consider the distances between kernel cells.
  - Case 1: Two stimulus signals in Tsinghua U. and Aalto U.
  - Case 2: Two stimulus signals in Tsinghua U. and Peking U.

  Case 1 is more significant than case 2!
- $\mathrm{md}(c_k)$: Minimal distance from cell c_k to the previous kernel cells.
- sig_i: The signal at granularity i.

$$\mathrm{sig}_i = \mathrm{st}(c_1) + \sum_{k \in K \setminus \{1\}} \mathrm{st}(c_k) \cdot (1 + f_d(\mathrm{md}(c_k)))$$

  $f_d()$: sigmoid-like function
- The final signal is the combination of the signals at different granularities.

Identification

Finally, how to identify whether there exists a matching trajectory of T_A?

- Find $T_B \in D_B$ with the largest signal.
- Identify whether they are matched if the signal is large enough.

To reduce misidentification, we need a stronger rejection strategy.

- Utilize **Weighted Jaccard Similarity (WJS)** (Ioffe et al ICDM 2010.).
- T_B is the matching trajectory of T_A if it has both the largest WJS and SIG score among all the candidates.

Outline

Experiment

Data set:

- User shared data of Baidu Inc. during 6 months (anonymized by hashing).
- Evaluate on different sampling rates:
  - *Dense set*: GPS location, navigation...
  - *Sparse set*: check-in, map queries...
- Evaluate on temporally disjoint case:
  - *Part-1*: aggregate the data of the first 3 months
  - *Part-2*: aggregate the data of the second 3 months

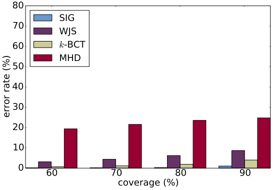
Experiment

Setting:

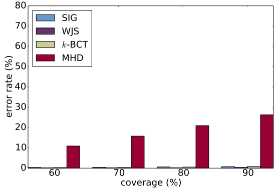
- Three different data sets:
  - *CN*: 31,511 users all over China
    - D_A = CN-Dense, D_B = CN-Sparse
    - D_A = CN-Part1, D_B = CN-Part2
  - *UT*: 14,115 users of the same university
  - *CB*: 4,323 users of the same company
- Coverage:
  - For each $T_A \in D_A$, find $T_B \in D_B$ with largest similarity
  - If the similarity is not large enough, reject to identify this trajectory
  - Percentage of trajectories we do not reject
- Compare with existing algorithms under the same "coverage":
  - modified Hausdorff distance (Adelfio et al. EPJ Data Science 2015)
  - k-Best Connected Trajectories (Chen et al. SIGMOD 2010)

Experiment

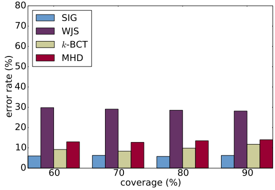
Error rates on different experiments:



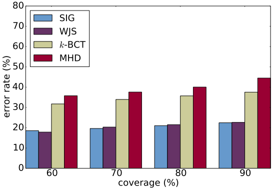
(a) CN-Sparse vs. CN-Dense



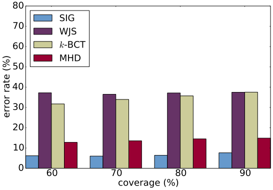
(b) CN-Part1 vs. CN-Part2



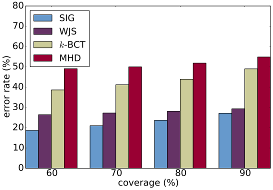
(c) CB-Sparse vs. CB-Dense



(d) CB-Part1 vs. CB-Part2



(e) UT-Sparse vs. UT-Dense



(f) UT-Part1 vs. UT-Part2

Experiment

Performance of AUI[3]

- AUI combines **signal based similarity** and **weighted jaccard similarity**.
- Determines the coverage automatically according to the data set.
- Compare AUI with other methods under the same coverage.

Experiment	AUI	SIG	WJS	k-BCT	MHD
CN-Sparse vs. CN-Dense (coverage = 59.72%)	99.94%	99.84%	99.08%	98.80%	78.13%
CN-Part1 vs. CN-Part2 (coverage = 88.35%)	99.80%	99.57%	99.41%	98.20%	70.39%
CB-Sparse vs. CB-Dense (coverage = 73.31%)	97.39%	97.53%	70.87%	91.59%	87.37%
CB-Part1 vs. CB-Part2 (coverage = 70.66%)	91.36%	81.67%	80.43%	65.93%	62.40%
UT-Sparse vs. UT-Dense (coverage = 72.84%)	95.63%	95.20%	76.02%	71.48%	87.32%
UT-Part1 vs. UT-Part2 (coverage = 60.81%)	90.09%	80.96%	74.97%	61.38%	50.89%

[3] See our paper for more experimental results.

Outline

Conclusion

- Studied the user identification across heterogeneous data sources.
- Presented an algorithm which handles large scale of mobility data.
- Proposed a novel similarity for extremely noisy data.
- Adopted an effective rejection strategy.
- Conducted extensive experiments.

Thank you!