Understanding urban mobility using crowdsourced GPS data

Dr. David McArthur, Dr. Yang Wang et al.

Urban Big Data Centre

- UK-wide data service for researchers who want to use big data to address a range of urban challenges, both local and global.
- Funded by the UK Economic and Social Research Council
- Objectives
  - **Develop novel solutions for using and sharing urban big data** including the infrastructure, tools and expertise to access such data;
  - **Provide high quality training and outreach activities** to equip researchers and decision-makers with the skills and knowledge to use big data to inform public policy debates and business innovations;
  - **Deliver cutting-edge research** to develop methods and tools to analyse urban big data as well as exemplar projects on substantive urban issues.

E·S·R·C ECONOMIC & SOCIAL RESEARCH COUNCIL

University of Glasgow

University of BRISTOL

THE UNIVERSITY of EDINBURGH

University of Reading

UIC UNIVERSITY OF ILLINOIS AT CHICAGO

The University Of Sheffield.

UNIVERSITY OF CAMBRIDGE

catch

Plan for the talk

- I'll be talking about one of my projects at the UBDC
- It's about one way in which we might get different (better?) transport data

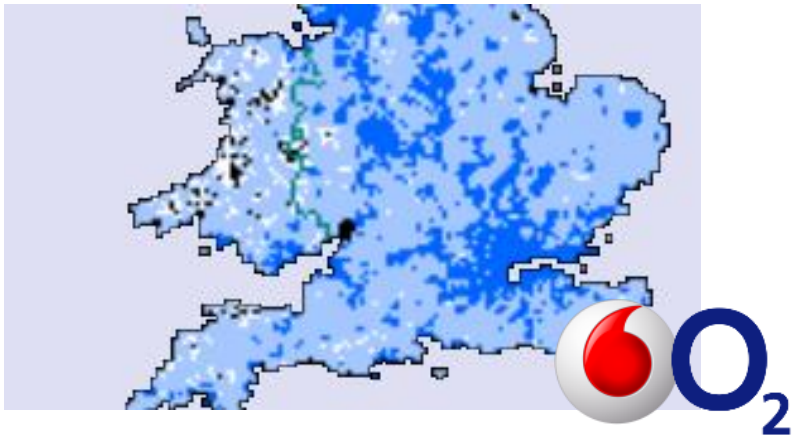
Some traditional data sources and limitations

- Surveys
  - Expensive
  - Small samples
  - Hard to complete
  - May lack detail
- Sensor networks
  - Requires (expensive) infrastructure
  - Lacks origin/destination info
  - Not multimodal

(Some) Big data

University of Glasgow

Urban Big Data Centre



catch

Some limitations

- Hard/expensive to access
- Pre-processing of data can be a black box
- May be tied to one mode
- Spatial coverage may be patchy
- Unknown biases

Smartphone data

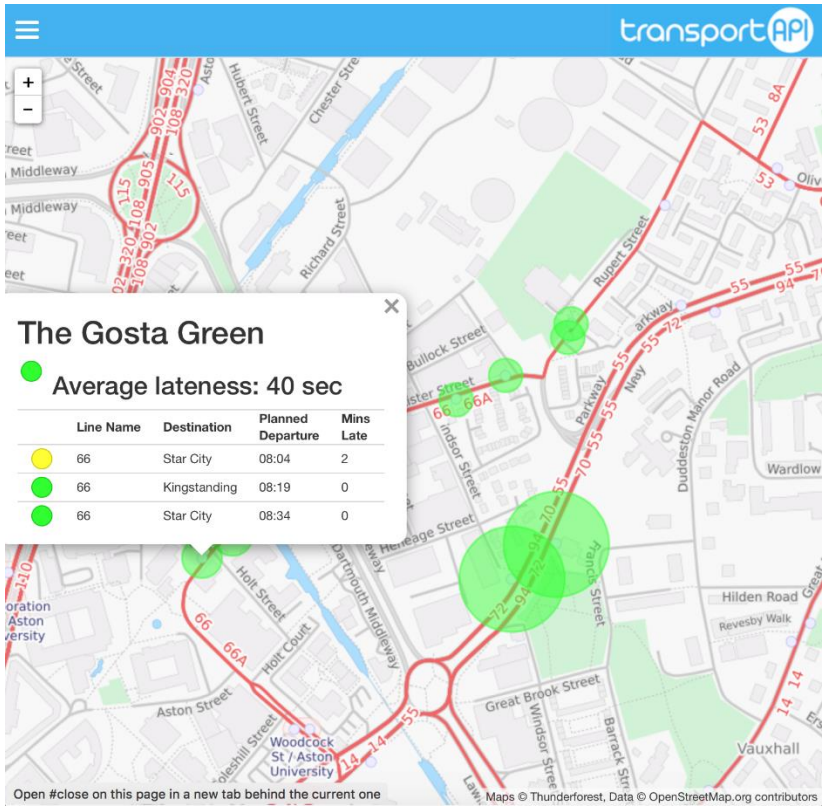
- Smartphones offer the chance to collect rich data
  - Independent of operator
  - Information for all travel modes
  - Full door-to-door OD and route data
  - Potentially real-time

Catch! Project

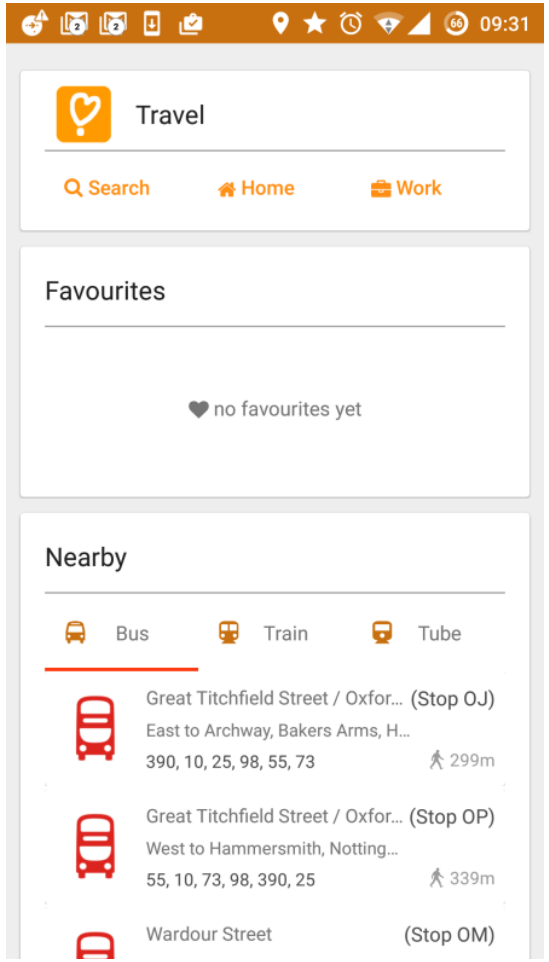
- The Catch! (citizens at the city's heart!) app is a journey planning app
- It passively collects GPS trajectories
- It utilises the phones' sensors to infer travel mode
- Users get to contribute data to improve transport planning in their city
- Insight from the data can feed back into better journey suggestions
- Funded by Innovate UK

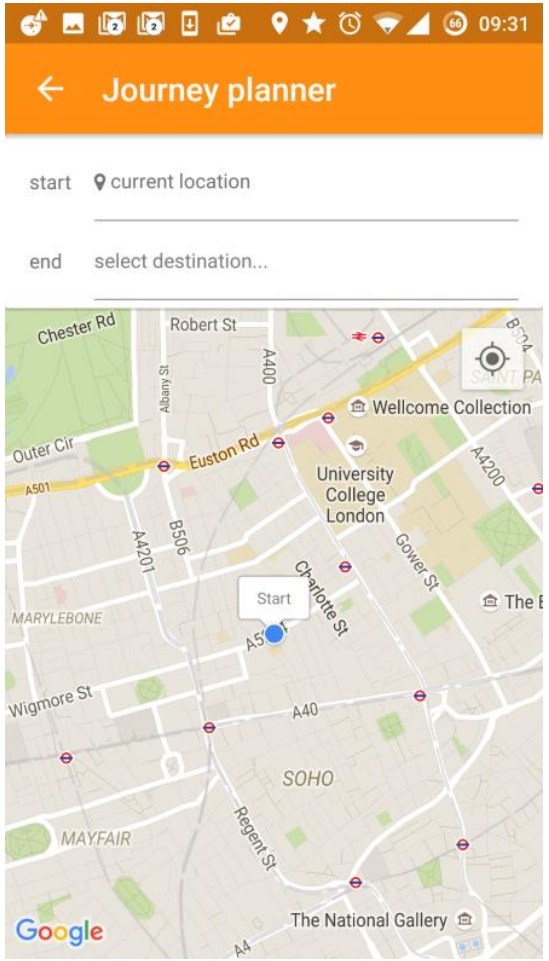
Catch! App

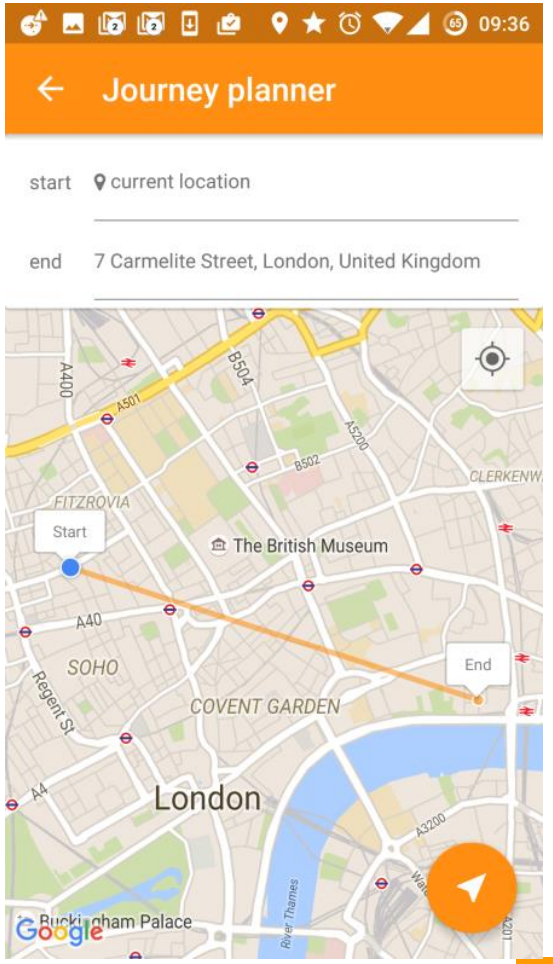
- Includes real time information on road and public transport performance



Catch! App

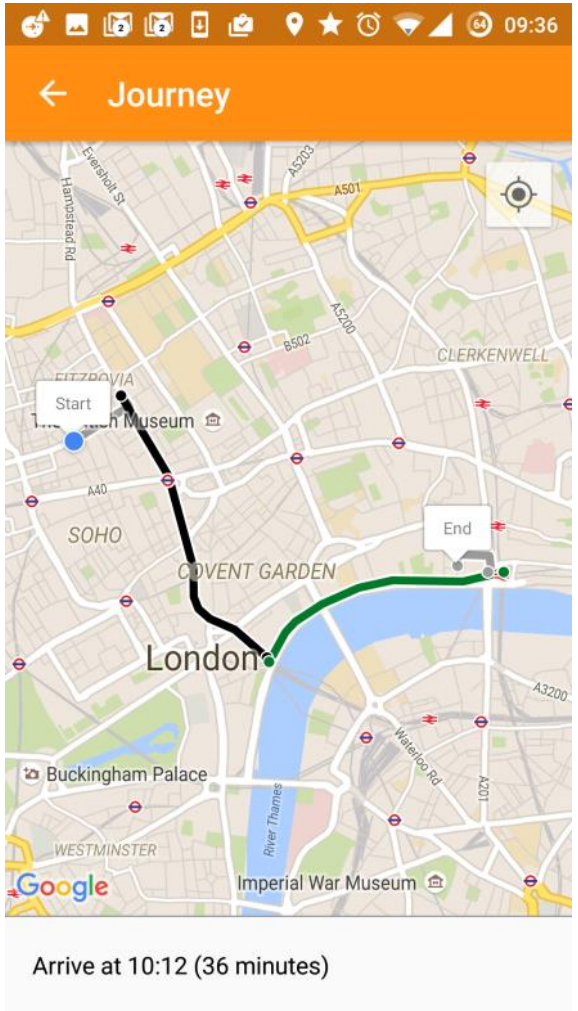


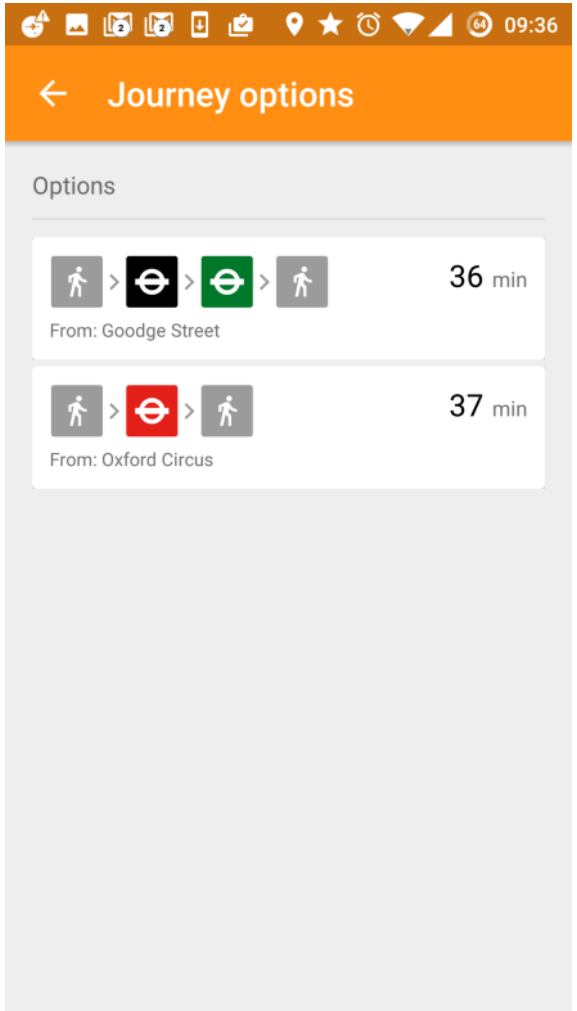


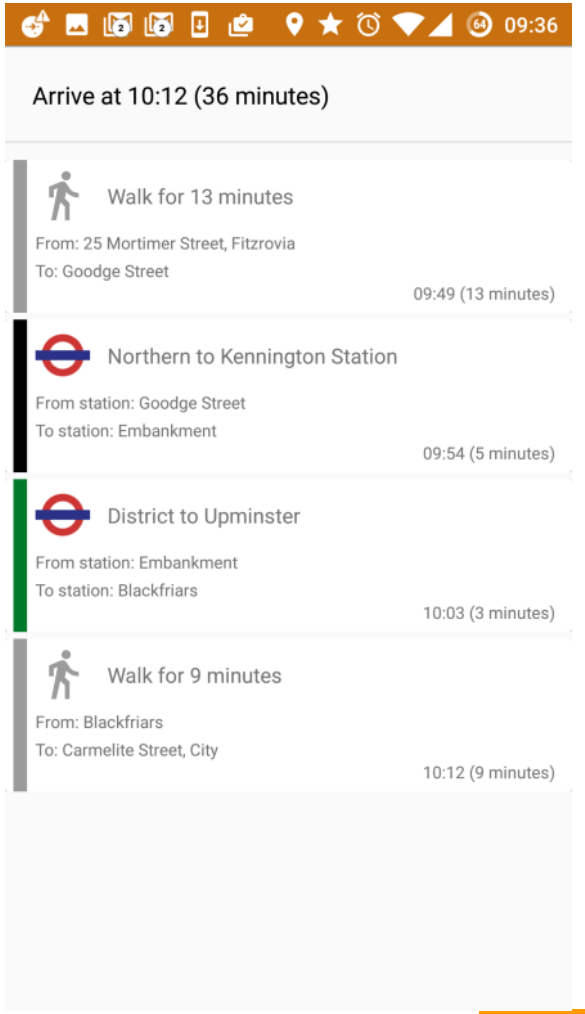


catch!

Catch! app







catch!


Plane
Car
Metro
Train
Bus
Cycle
Walk
0 miles
166 miles
197 miles
582 miles
213 miles
122 miles
81 miles
Crouch End
Camden Town
Chelsea
Biggin Hill
Hainault
Hornchurch
Dartford
River Tha
All journeys as if on one day
08:35
Morning rush hour in London
Analysing infrastructure at a number of levels and over long than normal periods can highlight city-wide as well as junction-specific traffic flow issues
itoi
catch

The Consortium

SMEs

- **TravelAi, The Behaviourlist, Elgin Roadworks, Placr**
- App development, data sources, citizen messaging, impact assessment

Research organisations

- **University of Glasgow (UBDC), University of Leeds (CDRC), Transport Systems Catapult**
- Data cleaning, anonymization, aggregation, analysis

Local Authorities and cities

- **Coventry, Ipswich, Leeds, Newcastle, Oxfordshire**
- Citizen access, sounding boards, pilots, data sources, advocates

Data from the app

- Person identifier
- Latitude, longitude
- Time
- Inferred mode
- Collected every 5 seconds (may change)

Processing

- Begin by removing nonsensical points e.g. points where the travel speed is unrealistic
- Assign the points onto the transport network (map matching)

Map matching approaches

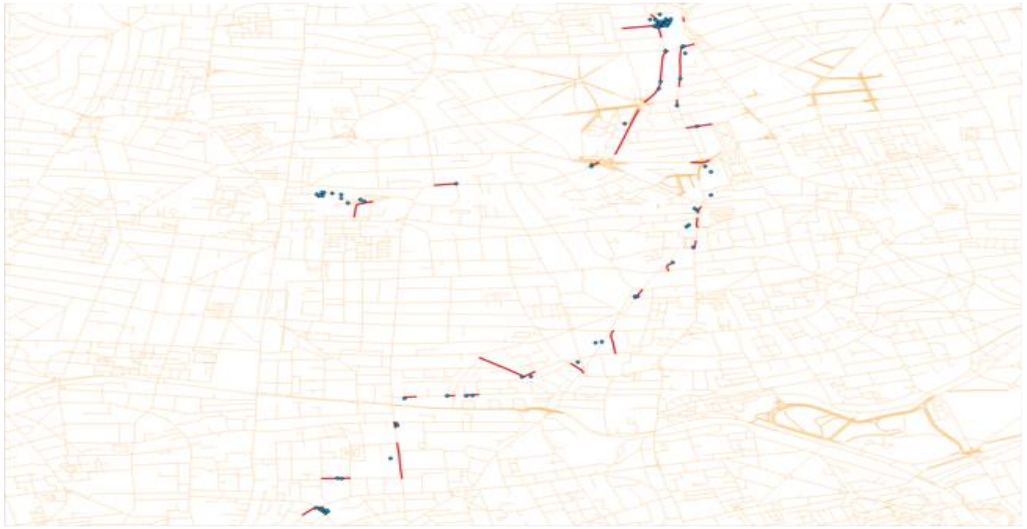
- Geometric approach
  - node-to-node, node-to-link, curve-to-link
- Topological approach
  - geometric approach plus connectivity of the road network
- Advanced approaches
  - Weight based or probabilistic algorithms

Four open source libraries

Name	Category	Logic
Mapillary, map matching (mm) in Python	Combined Geometric and Topological	• Selecting road links that are close to the sampling points, • connectivity of road network is of concern but only inside the points bounding box.
Multiple Hypothesis Technique (MHT) based library in Python	Combined Geometric and Topological	• Multiple route candidates are kept in memory when matching each sample point at a time, • connectivity, distance to samples and turning directions are of concern all the time, • path is determined by consistent updating scores the candidates road segments.
GraphHopper in Java	Advanced - probabilistic	• Select candidate closest to GPS points, • probability is calculated based on their distance to samples, • sorting route by minimizing probabilities, • assign points to road edges.

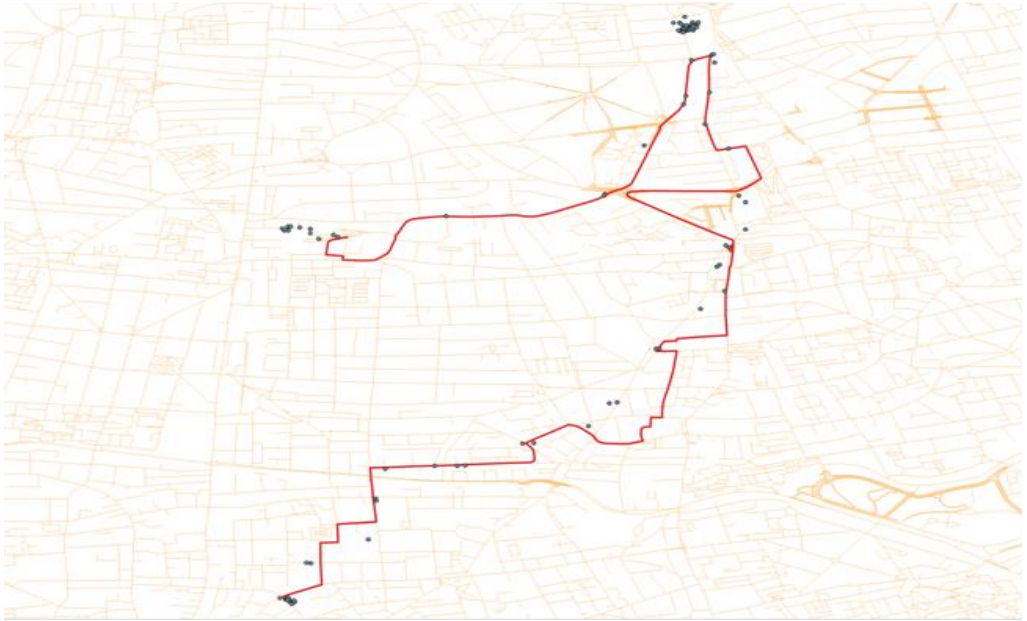
Barefoot in Java	Advanced -Hidden Markov Model	• Probability lattice is calculated where
		  • emission probabilities are the chances of GPS points observed at the road segment are reduced when distance increases.
		  • transition probabilities arethe chances of movement between road segments due to road connectivity at consecutive times.
		• the best path is calculated by Viterbi algorithm from the probability lattice.

catch

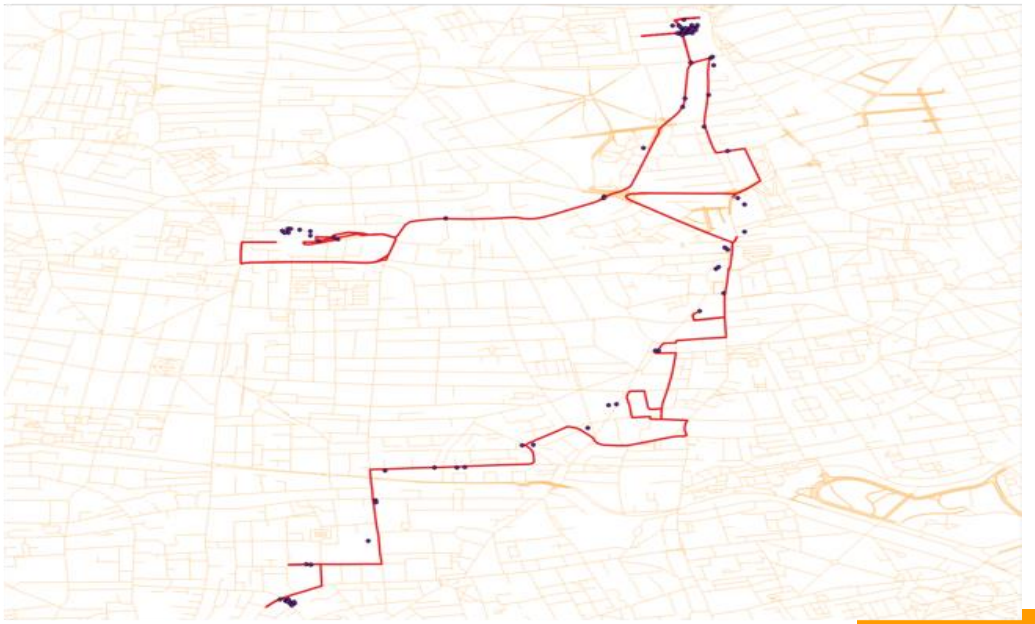


mm

MHT



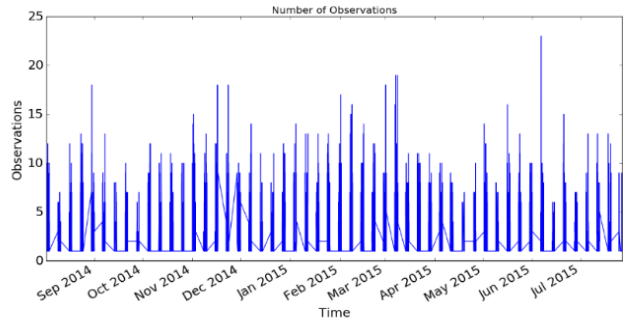
GraphHoppers

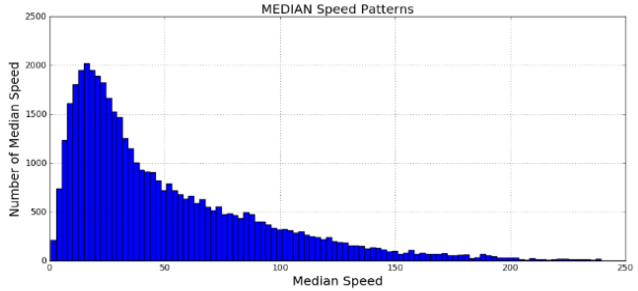


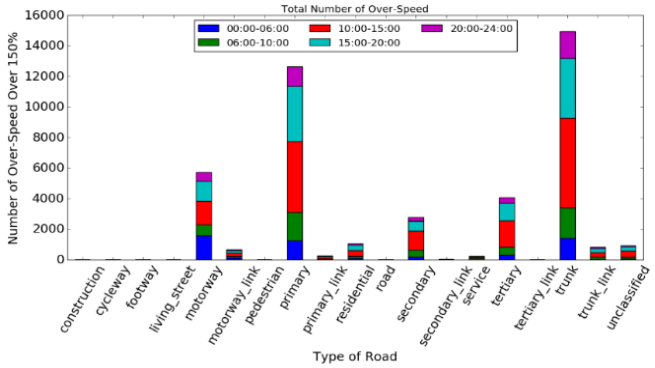
Barefoot

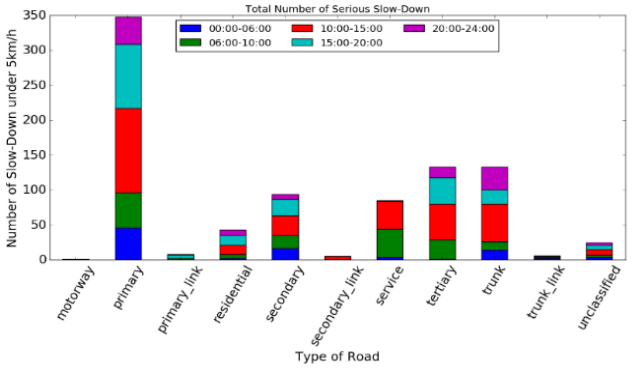
Aggregation of output

- Journey information can be aggregated at the link-level
- Can provide mode-specific counts of users/journeys
- Can provide a very fine-grain temporal scale





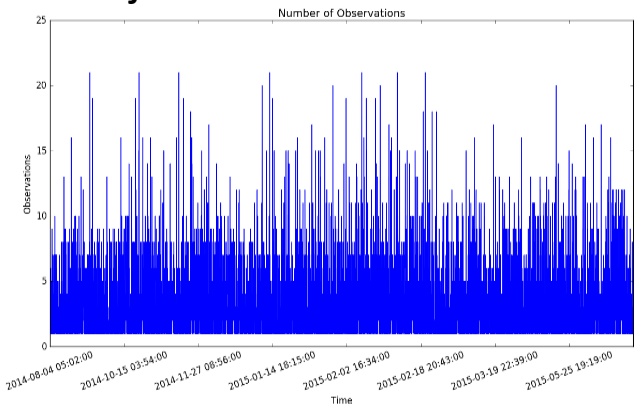


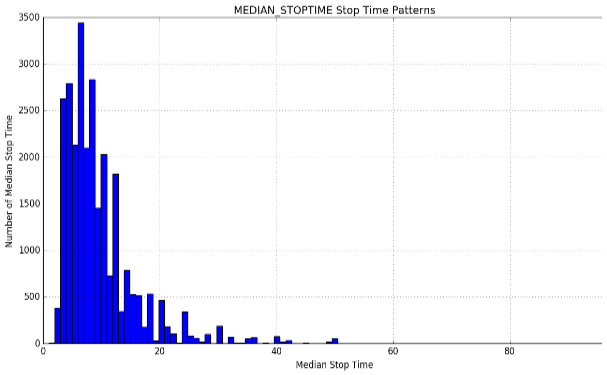


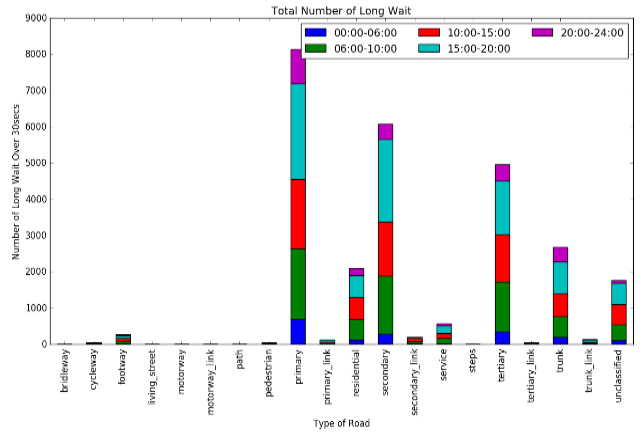
Average weekend speeds

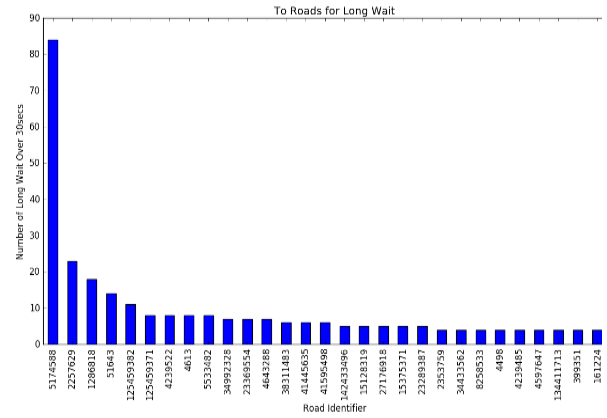
Aggregation output

- Information can be aggregates at the level of junction
- How long do different road users have to wait at junctions ?

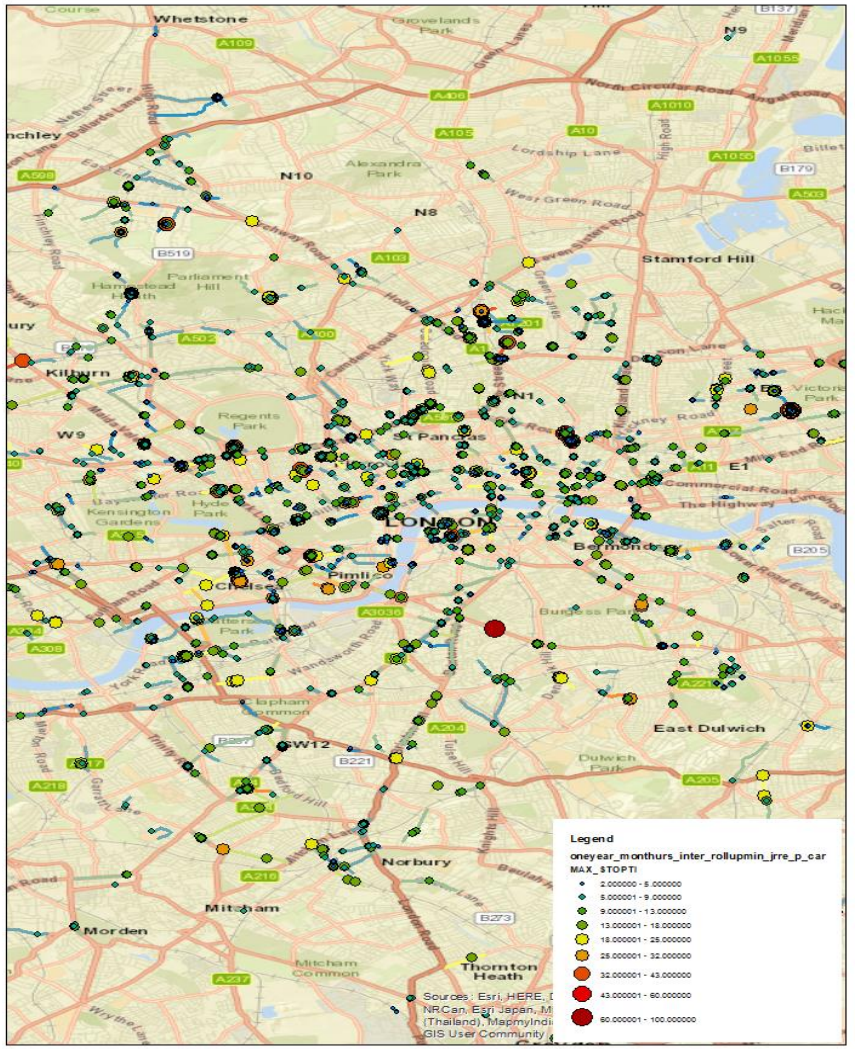






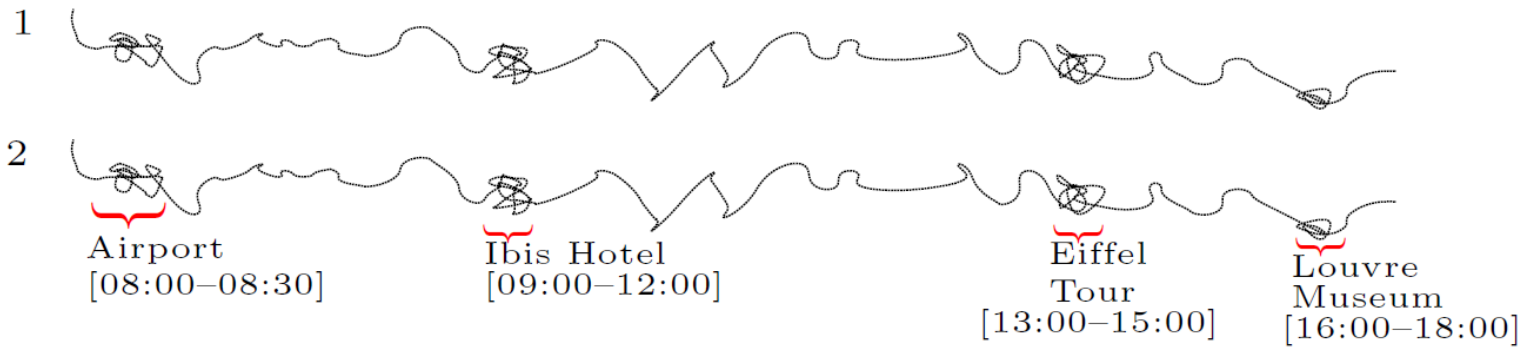


Average vehicle waiting time



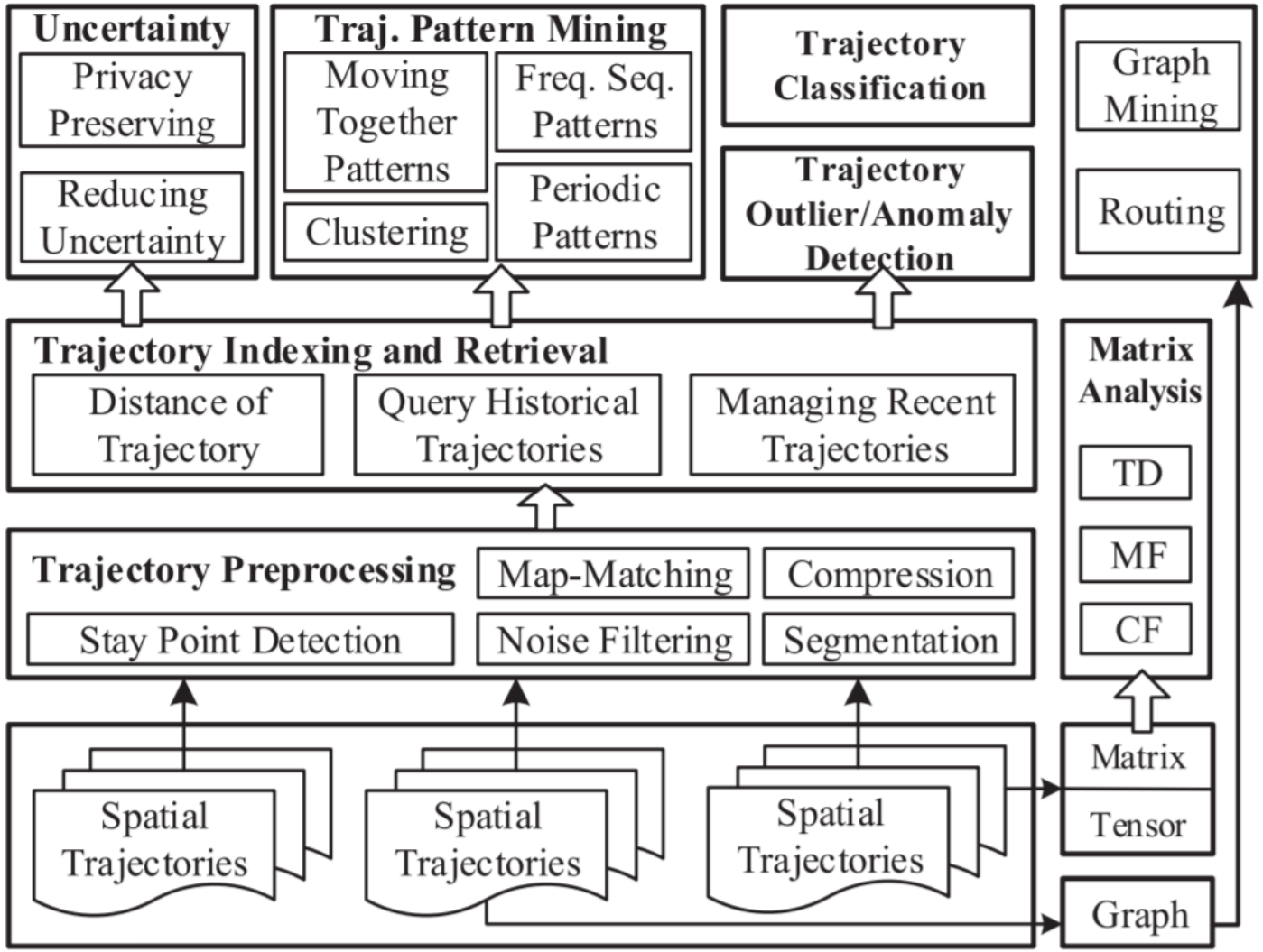
Semantic Trajectory

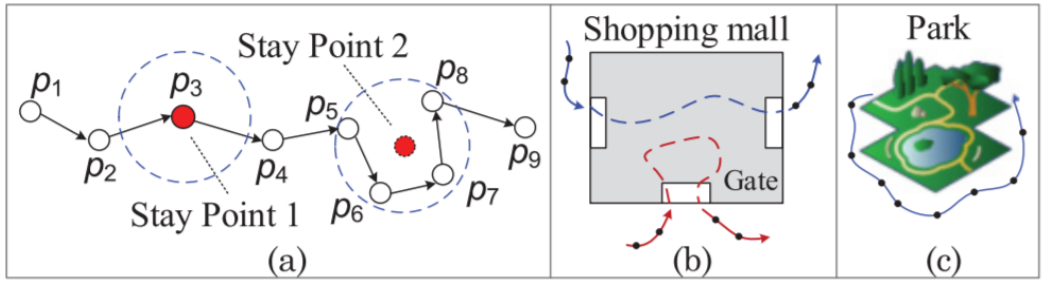
- Another strand of our work deals with the semantic annotation of the GPS trajectories



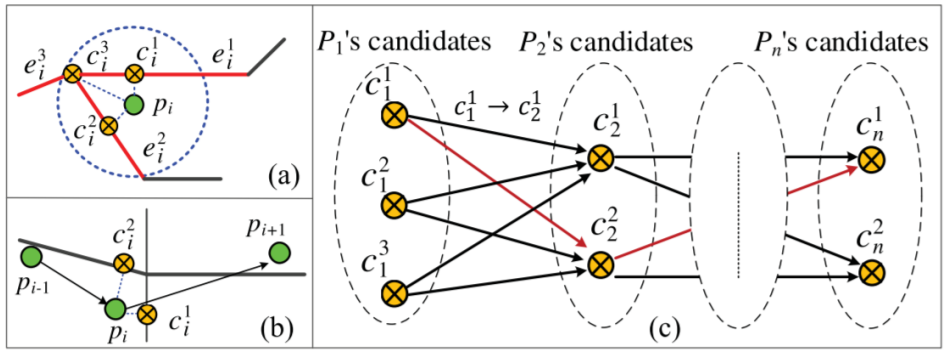
Alvares, L.O., Bogorny, V., Kuijpers, B., Macedo, J.A., Moelans, B. and Vaisman, A. (2007a), "A model for enriching trajectories with semantic geographical information", Proceedings of the 15th ACM International Symposium on Advances in Geographic Information Systems, ACM, Seattle, WA, pp. 1-8.

Models or Frameworks for Semantic Trajectory Development Paradigm of Trajectory Data Mining (Y. Zheng)



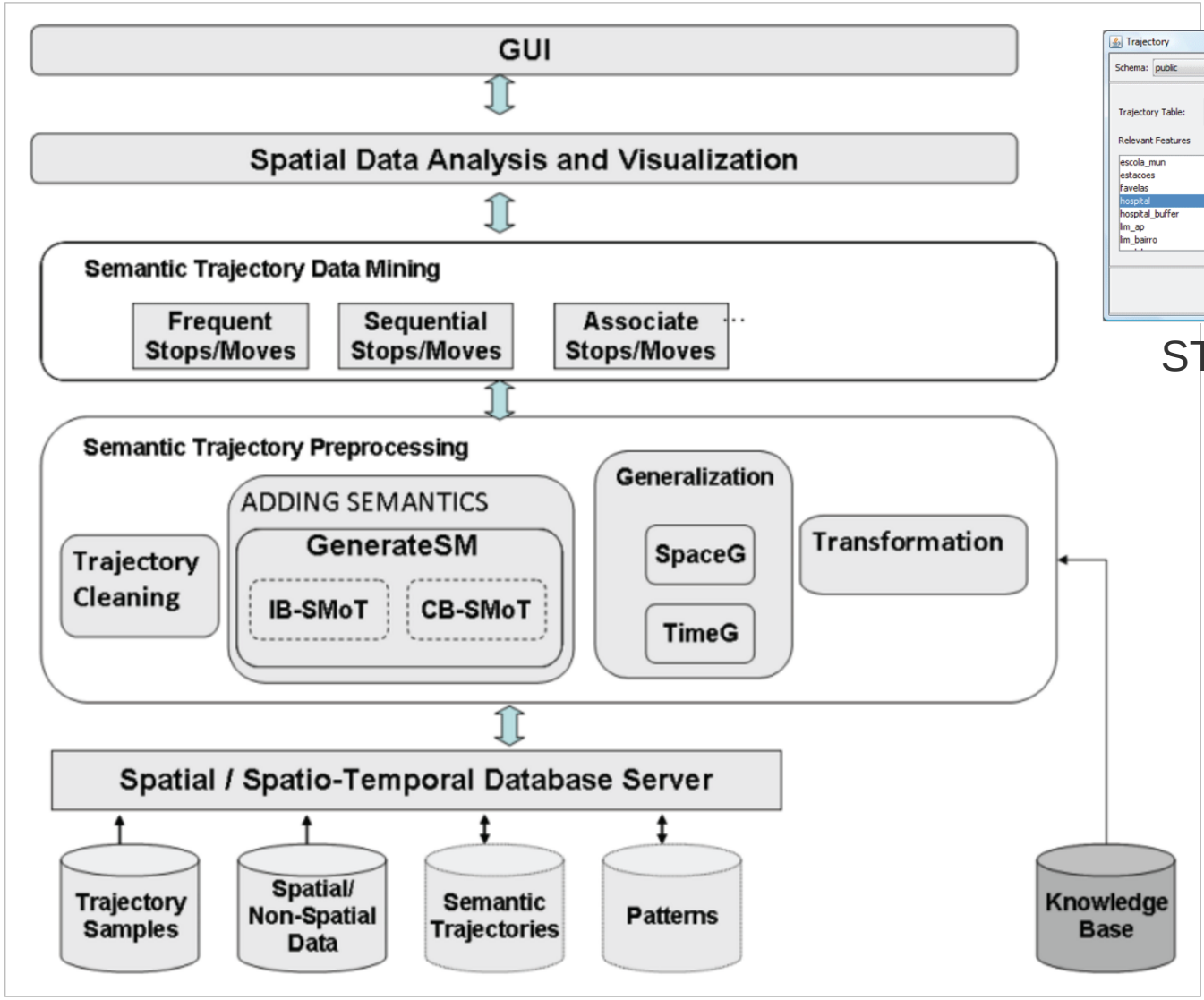


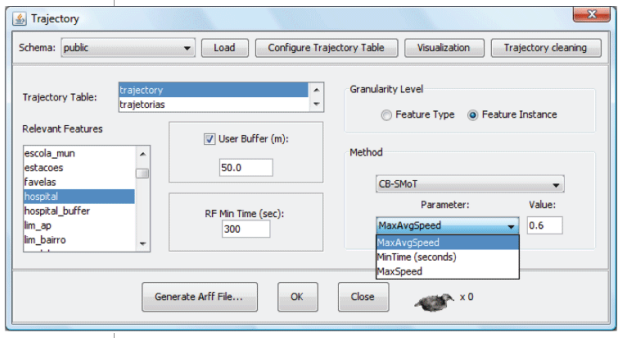
Stay points in a trajectory



An advanced map-matching algorithm

Models or Frameworks for Semantic Trajectory Development
Weka-STPM – an open source toolkit

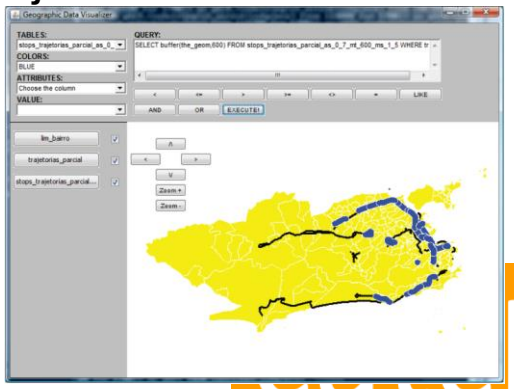




STPM main interface

tid	stopid	stop_name	stop_gid	start_time	end_time	the_geom
2236	0	0_unknown	0	2004-08-04 18:10:18	2004-08-04 18:25:29	667015.36217975 7464208.41043153, ...
2234	0	1_unknown	0	2004-07-20 18:35:58	2004-07-20 19:06:29	679405.188285265 7469918.46315512, ...
2233	0	2_unknown	0	2004-07-20 15:39:33	2004-07-20 15:51:50	680479.071827787 7466287.37028003, ...
2232	0	3_unknown	0	2005-05-23 08:04:08	2005-05-23 10:06:25	673805.183177759 7455273.90615614, ...
1236	0	1_unknown	0	2005-02-21 17:20:42	2005-02-21 18:04:10	681063.492700548 7469816.57366467, ...
1235	0	1_unknown	0	2005-02-18 16:39:03	2005-02-18 17:08:24	681203.357858659 7469257.89224045, ...
1234	0	4_unknown	0	2005-02-17 15:33:33	2005-02-17 15:46:48	686860.9302822 7463298.01652945, ...
1233	0	1_unknown	0	2005-02-16 07:47:22	2005-02-16 08:11:14	680939.399430723 7469585.19999004, ...
1232	0	1_unknown	0	2005-02-15 16:45:57	2005-02-15 17:24:24	681037.05860043 7469842.71077073, ...
1231	0	1_unknown	0	2005-02-14 07:37:42	2005-02-14 08:24:26	679634.452558103 7471157.38290711, ...

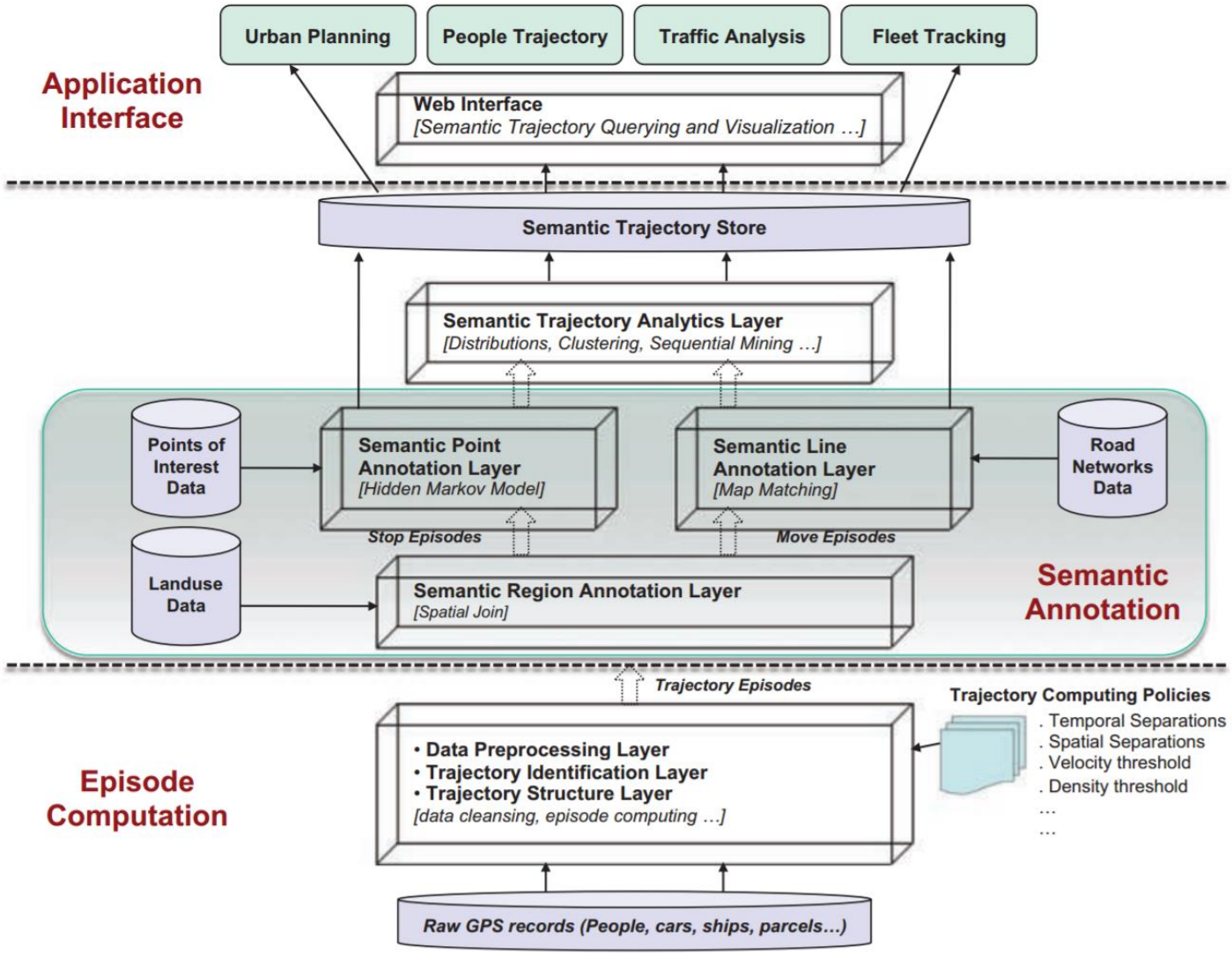
Preprocessing Trajectories with CB-SMoT

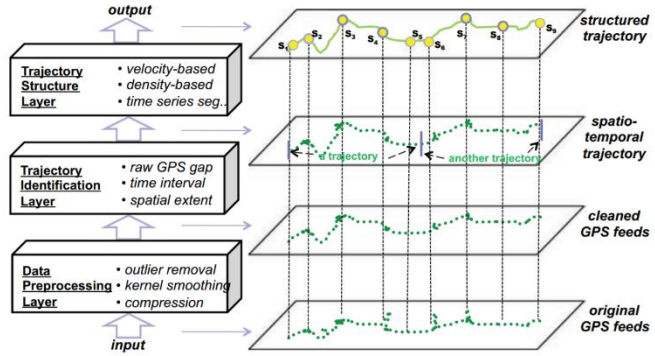


Preprocessing Trajectories with CB-SMoT

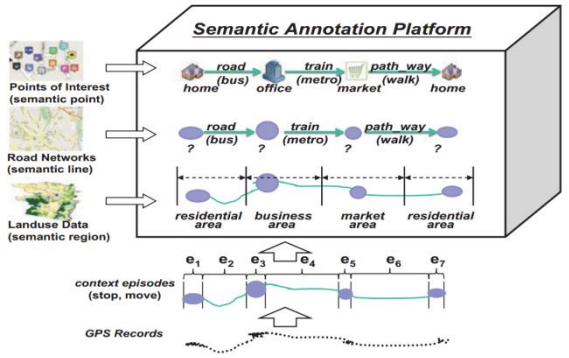
Models or Frameworks for Semantic Trajectory Development

Semantic Trajectory Platform Architecture in (Z. Yan et al)





Trajectory computing platform



Trajectory annotation platform

An Extra Data Anonymization Layer to Semantic Trajectory Framework 1

- Current frameworks have no facilities for data anonymization and data sharing;

- Two main sources of contextual information: road network and geographical regions or points through Map Matching and stop/move Detection and Annotation;

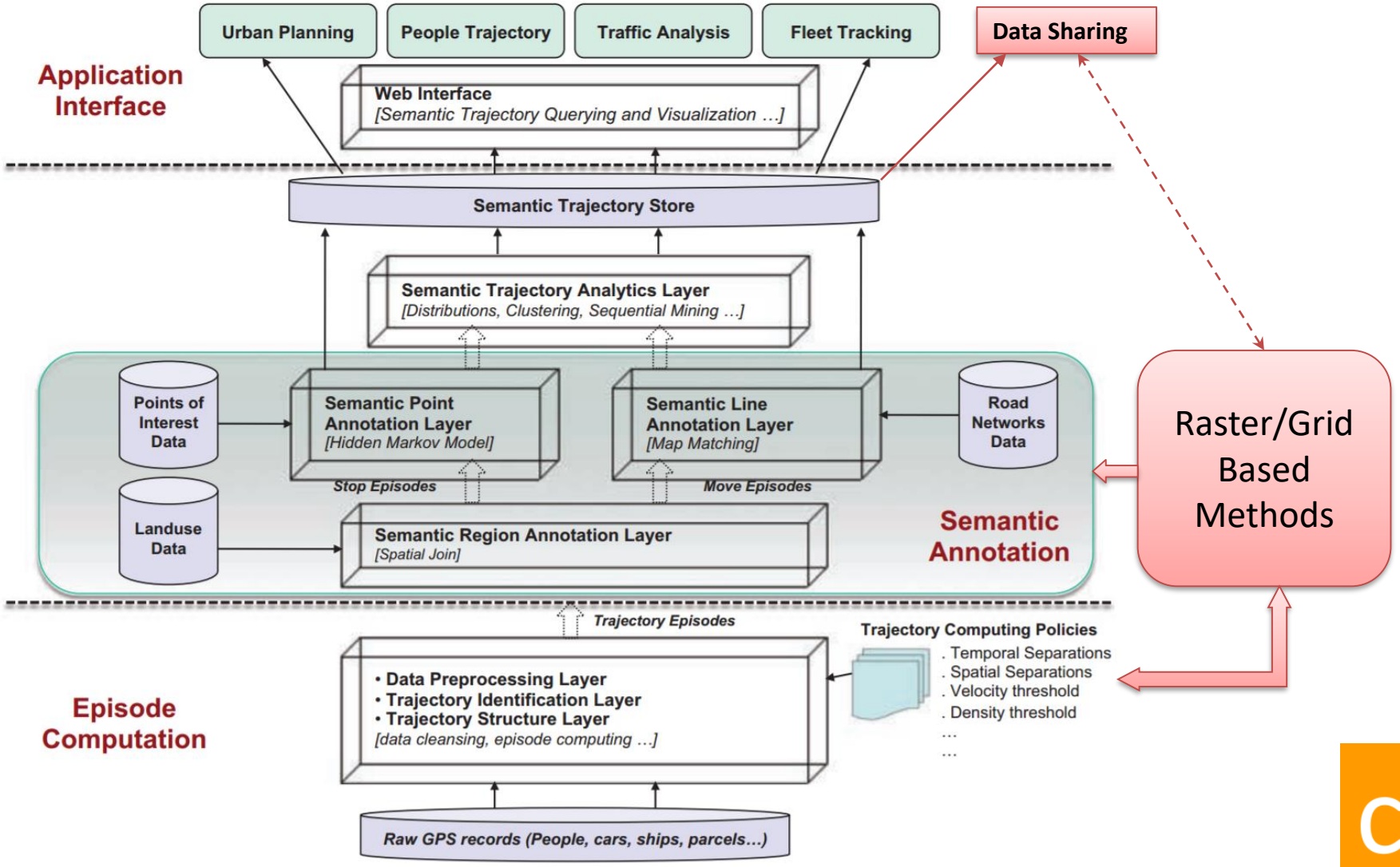
- Our contribution:

Adding an extra data anonymization layer to the framework to better:

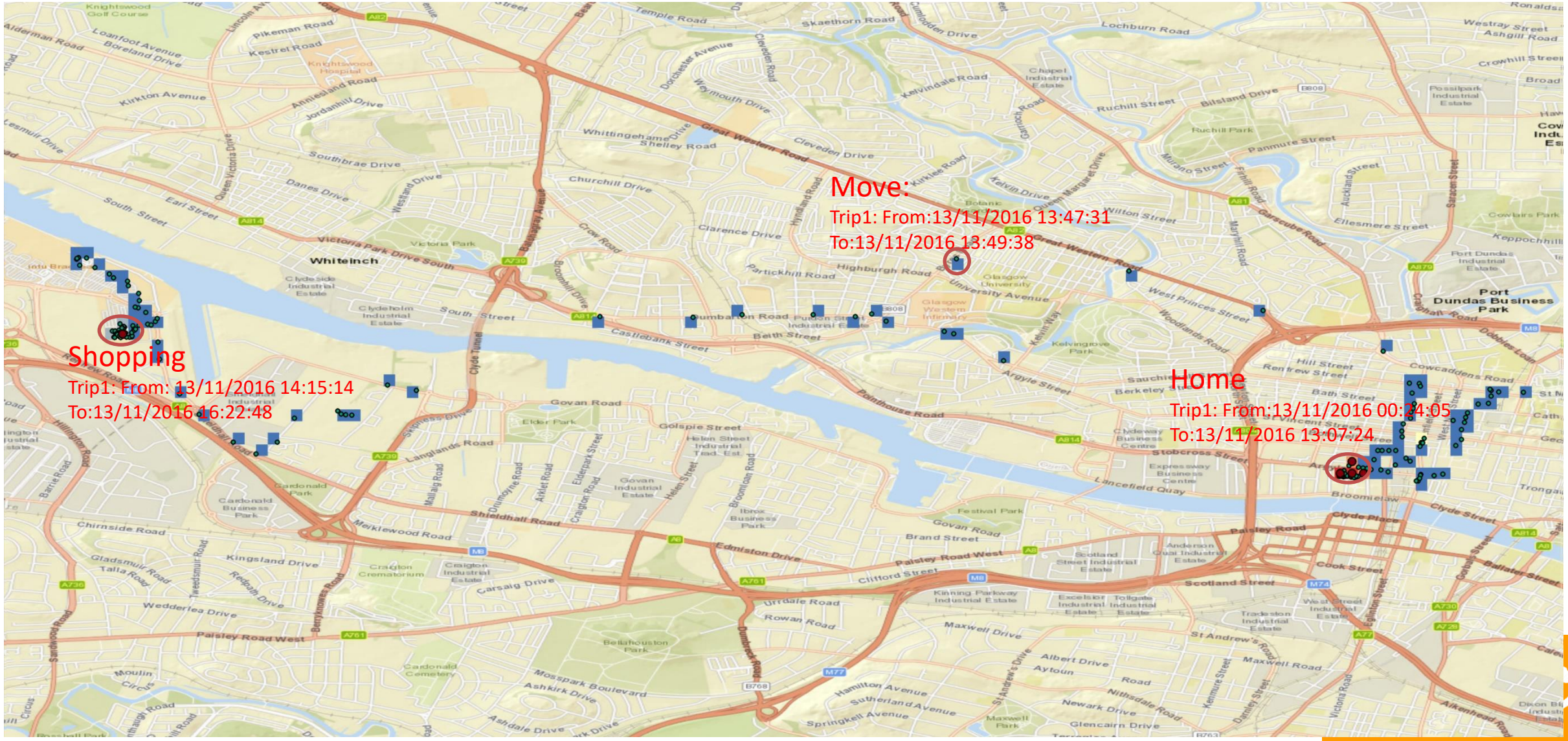
(a) Protect individual users' privacy

(b) Develop a workflow including methods and algorithms towards such a goal using a raster/grid based generalization structure.

An Extra Data Anonymization Layer



Example


Shopping
Trip1: From: 13/11/2016 14:15:14
To:13/11/2016 16:22:48
Move:
Trip1: From:13/11/2016 13:47:31
To:13/11/2016 13:49:38
Home
Trip1: From:13/11/2016 00:24:05
To:13/11/2016 13:07:24

Stop Detection

Stay point detection is

- the set of geographical locations that an individual stays at for a certain amount of time (Li *et al.,* 2008)

Knowing the stay points allows us to

- infer activities that are conducted at different locations (Liao *et al.* 2007, Ye *et al*. 2009),
- Segment the trajectory with separate travel purposes (Zheng 2015)
- Find points where modes are switched e.g. walking to train (Zheng *et a*l. 2008, , Patterson *et al*. 2003, Liao *et al*. 2007, Gonzalez *et al*. 2008).

Stop Detection Methods

Threshold based Approaches

taking GPS embedded or calculated parameters, such as speed, dwelling time, clustering density and 'power-off' gap durations (Ashbrook and Starner 2003, Schuessler and Axhausen 2008, Srinivasan *et al*. 2008). Some problems:

- threshold settings are arbitrary and require additional information about the raw GPS data
- speed values are unreliable due to limitations of GPS

Density based Approaches

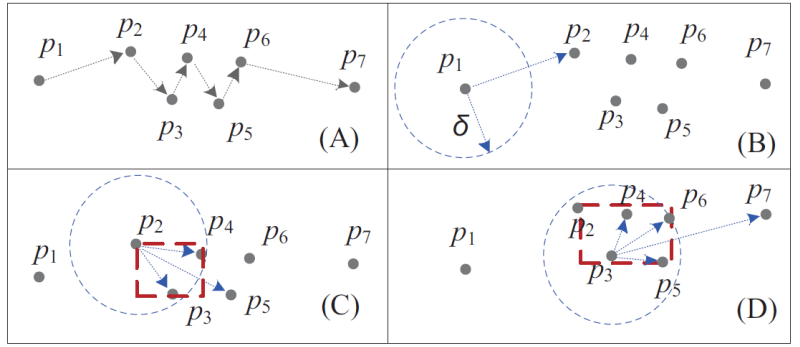
- spatial clustering algorithms (e.g. DBSCAN) assumes there are a larger number of points clustered around significant locations (Schoier and Borruso, 2011).
- The algorithm scans for a minimum number of tracking points ($MinPts$) around a randomly selected unvisited points within a pre-specified search radius (eps) then further aggregates clusters if they are densely connected (Ester *et al*. 1996).
- Approaches such as Hinneburg and Keim (1998), Ankerst et al (1999) and recent work proposed by Campello et al (2013) try to simplify the parameters
- Other proposed improvements include introducing temporal and other dimensions (Birant and Kut 2007, Hwang 2013]. ST-DBSCAN (Birant and Kut 2007), temporal DBSCAN (Hwang *et al.* 2013), interpolate missing GPS points (Hwang et al. 2017).
- DBSCAN is less sensitive to noise and can detect stops with arbitrary shapes. It doesn't work well with large temporal gaps, loss of GPS signal or movement inside a house

Stop Detection in Semantic Trajectory Mining (Y. Zheng et al.)

- Density based method: distance between each points to all other points until the final distance and the duration of the set of points exceeds the thresholds. The algorithm loop and add points into the candidate stop until the clusters is no longer expansible.

- A supervise model: features including a) minimum bounding ratio (MBR), average and centre distance to road segments, duration and speed for last stop, b) term frequency invers document frequency (*tf-idf*) for Point of Interests, c) repetitive historical visits, to filter out the clusters caused by slow speed.

- The method is designed for taxi stop location detection, therefore, it is transport network constrained.



Parking candidates detection

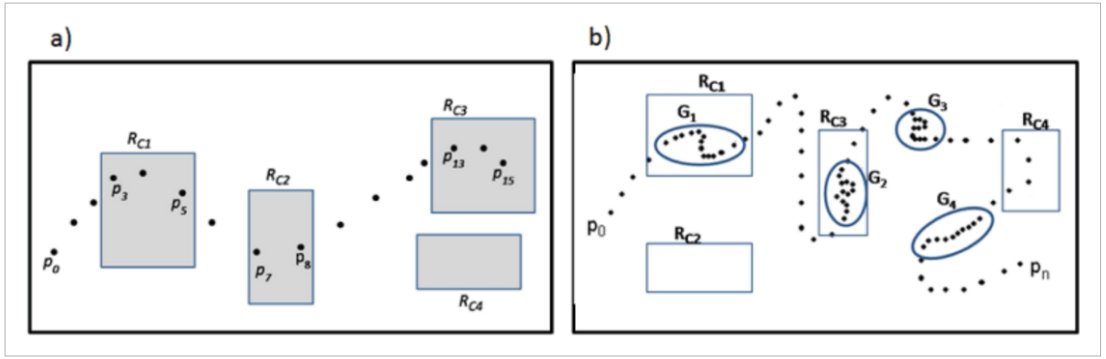
Algorithm 1: ParkingCandidateDetection

Input: A road network G, a trajectory Tr, distance threshold δ, time threshold τ
Output: A set of parking candidates $\mathbb{P} = \{P\}$

```
1  i ← 0, M ← ||Tr||, P ← ∅, ℙ ← ∅;
2  while i < (M − 1) do
3      j ← i + 1; flag ←false;
4      while j < M do
5          dist ←Distance (p_i, p_j);
6          if dist < δ then j ← j + 1;flag =true;
7          else break;
8      if p_{j−1}.t − p_i.t > τ and flag =true then
9          foreach point p ∈ Tr[i, j) and p ∉ P do
10             P.Add(p);/* build a candidate          */
11         if i = j − 1 then
12             ℙ.Add(MB (P)); P ← ∅;
               /* add the minimum bounding box of P
                  into ℙ                              */
13         i ← i + 1;
14     return ℙ
```

Stop Detection in Semantic Trajectory Mining Weka-STPM

- An Intersection based Stop and Move Trajectories (**IB-SMoT**): approach where GPS trajectories are spatially intersected with pre-defined geographical file to look for durations that spend inside each stop shape to determine a stop.
- A clustering-based Stop and Moves of Trajectories (**CB_SMoT**) algorithm: based on DBSCAN but clusters speed values of the trajectory. By apply the algorithm, slower speed part of trajectories are clustered. If stay duration is greater than a threshold, stay point is detected.
- A **direction-based** stops and Moves of trajectories: similar direction change controlled by a minimal direction change threshold, minimal amount of stop duration is used to verify if the direction change is noise or direction change reaches its end.

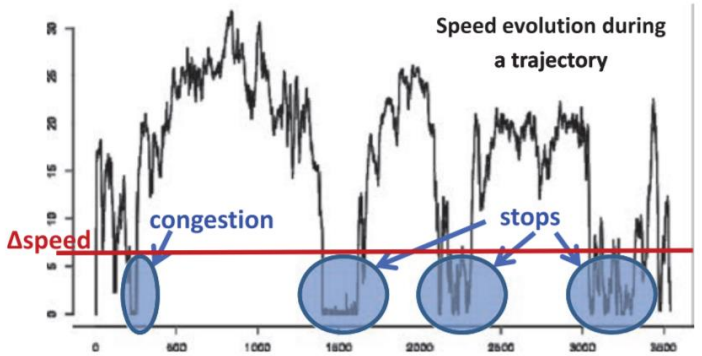


(a) Example of the IB-SMoT method, and (b) Example of the CB-SMoT method

Stop Detection in Semantic Trajectory Mining
Yan (et al. 2013)

A combined **velocity-based** and **density based** methods.

- For the velocity-based method, the speed of successive GPS points are compared with the minimum value of observed speed on the by-passing road segment and the average travel speed of the user.
- further compare the time duration of the groupings of the stop episode with a minimal stop time threshold to eliminate congestion stops.
- Since speed value is not always reliable to indicate stop, the authors apply another supplementary density-based method to cover generic cases.



Velocity-based stop identification

ALGORITHM 2: *Velocity-based trajectory structure*

Input: a raw trajectory $\mathcal{T}_{raw} = \{p_1, p_2, \ldots, p_n\}$
Output: a structured trajectory $\mathcal{T}_{str} = \{e_1, e_2, \ldots, e_m\}$ where e_i is a tagged trajectory episode (stop $\mathcal{S}$ or move $\mathcal{M}$)

begin
  /* initialize: calculate GPS instant speed if needed */
  ArrayList$\langle x, y, t, tag\rangle$ *gpsList* ← getGPSList($\mathcal{T}_{spa}$);
  if *no instant speed from GPS device* **then**
    compute GPS instant speed s_i for all $p_i = (x, y, t) \in gpsList$;
  /* episode annotation: tag each GPS point with 'S' or 'M' */
  forall the $p_i = (x, y, t) \in gpsList$ **do**
    // *get dynamic* $\Delta^{(i)}_{speed}$ *by Algorithm 1*
    $\Delta^{(i)}_{speed}$ ← getDynamicΔ_{speed} (p, obj_{id}, δ);
    // *tag GPS point as a stop point 'S' or a move point 'M'*
    if *instant speed* $s_i < \Delta^{(i)}_{speed}$ **then**
      tag current point $p_i(x, y, t)$ as a stop point 'S';
    else
      tag current point $p_i(x, y, t)$ as a move point 'M';
  /* compute episodes: grouping consecutive same tags*/
  forall the *consecutive points with the same tag 'S'* **do**
    // *compute stop episode*
    get the total time duration $t_{interval}$ of these points;
    if $t_{interval} > \tau$ *the minimal possible stop time* **then**
      $stop \leftarrow (time_{from}, time_{to}, center, boundingRectangle)$;
      $\mathcal{T}_{str}$.(*stop*, 'S'); // *add the stop episode*
    else
      change the 'S' tag to 'M' for all these points; // *as "congestion"*
  forall the *consecutive points with the same tag 'M'* **do**
    // *compute move episode*
    $move \leftarrow (stop_{from}, stop_{to}, duration)$ // *create a move episode*
    $\mathcal{T}_{str}$.(*move*, 'M'); // *add the move episode*
  return the structured trajectory $\mathcal{T}_{str}$;

ALGORITHM 1: getDynamicΔ_{speed} (gpsPoint, obj_{id}, δ)

input : gpsPoint $p = (x, y, t)$, moving object obj_{id}
output: dynamic speed threshold Δ_{speed}

get the average speed of this moving object obj_{id}: $\overline{objectAvgSpeed}$;
if *network-constrained trajectory* **then**
  get the average speed of the nearest road crossing to p: $\overline{crossingAvgSpeed}$;
  get the average speed of the map matched road segment of p: $\overline{segmentAvgSpeed}$;
  $positionAvgSpeed \leftarrow min\{\overline{crossingAvgSpeed}, \overline{segmentAvgSpeed}\}$
else
  get the average speed of the cell that (x,y) belongs to: $\overline{cellAvgSpeed}$;
  $positionAvgSpeed \leftarrow \overline{cellAvgSpeed}$
compute the dynamic speed threshold by Definition 4;
return Δ_{speed}

A Raster Sampling based Method

A ‘top-down’ raster sampling method which directly queries a set of GPS records and samples those with significant differences

- Geographical attributed raster cells by nature impose spatial constrains while we try to sample temporal and other attributes inferred from the GPS records
- A data clustering method is performed at the final stage
- It does not sampling the density of GPS records inside grid cells, but rather information such as total dwelling time

Advantages:

- requires only the setting of the raster cell size
- fast and accurate (compared to a travel diary)

Exploring GPS Indicators for Stop Detection

Data: a day-to-day episodes of one user's one month, from 2016-07-12 to 2016-8-10 which are cleaned, and a travel dairy containing locations of stops in chronological order

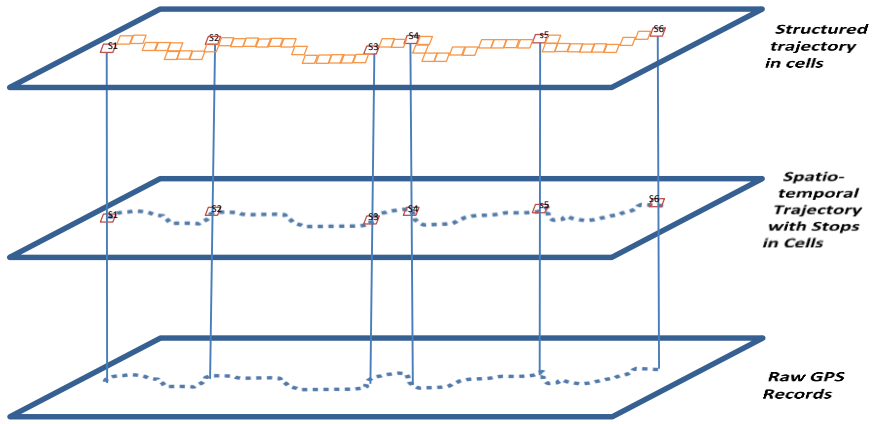
Method:

- Top-down sampling method which depend on indicators including
  - (a) time difference between two consecutive GPS tuples,
  - (b) an rough estimation of single trip GPS dwelling time at a given cell,
  - (c) a dwelling time deducing the travel time observed before and after a given GPS record,
  - (d) an estimation by pulling actual dwelling time per visit.
- Natural Break (Jenks) with goodness of variance fit over 0.8, to cluster the cell values into groups then select stops
- Two baseline methods:
  - (e) using thresholds to select stops with higher GPS dwelling time
  - (f) detecting stops less 'bounded' with the road network through a map matching process, are chosen as baselines for comparison.

Summery: An extra raster/grid layer

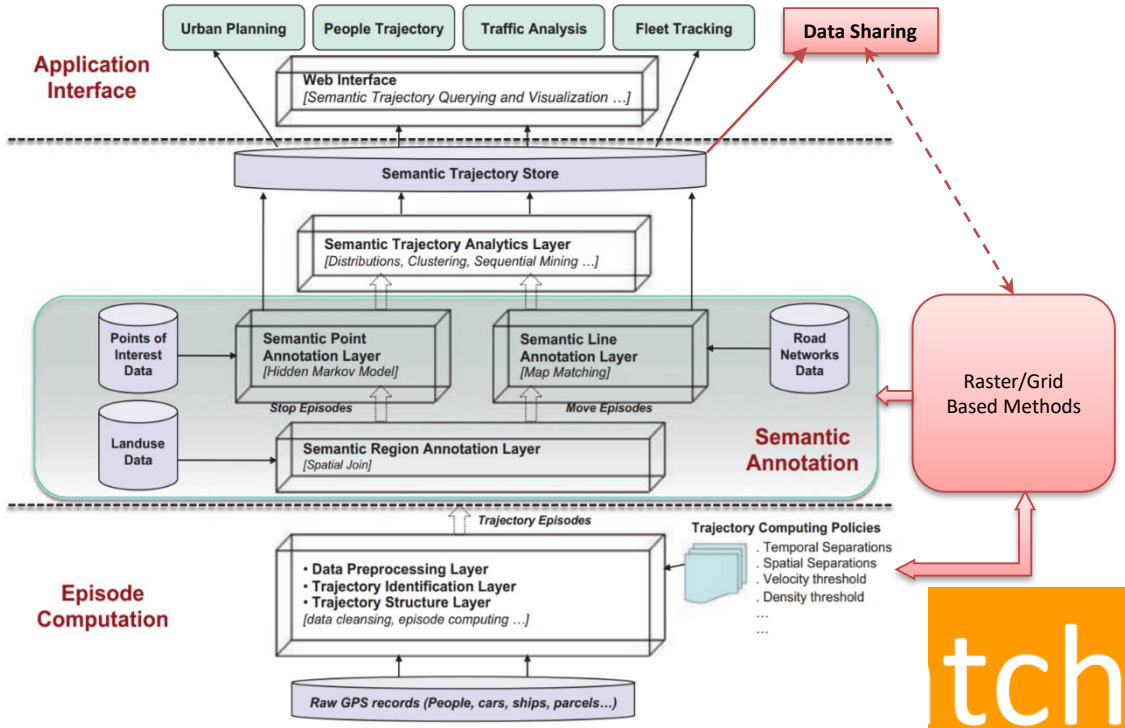
Advantages:

- enable multi-level data sharing while protect privacy;
- facilitate stop detection and further stop/move segmentation;
- support fast and semantic enriched GPS queries without performing expensive spatial joins or intersections;
- ease further spatial/temporal activity pattern mining and place/route recommendations.



Limitations:

- extra processing time;
- stop/move segmentation is still a raster/vector combined method
- top-down approach missing shorter stays
- scalability needs to be investigated.



(Some) limitations of the project

- Will the app be used by enough people
- Will it drain people's batteries?
- The data are biased; but how? Will this change over time?

Thank you for your attention.
Questions?

www.ubdc.ac.uk

@UrbanBigData

urbanbigdata

Urban Big Data Centre

An ESRC Data Investment