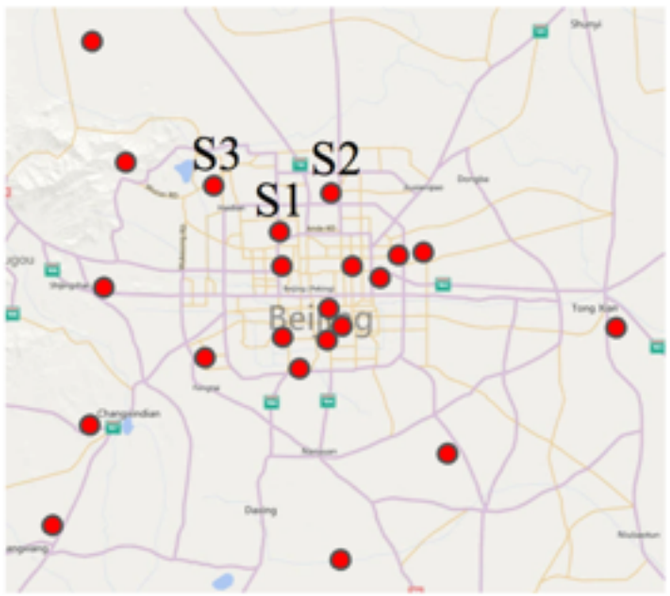
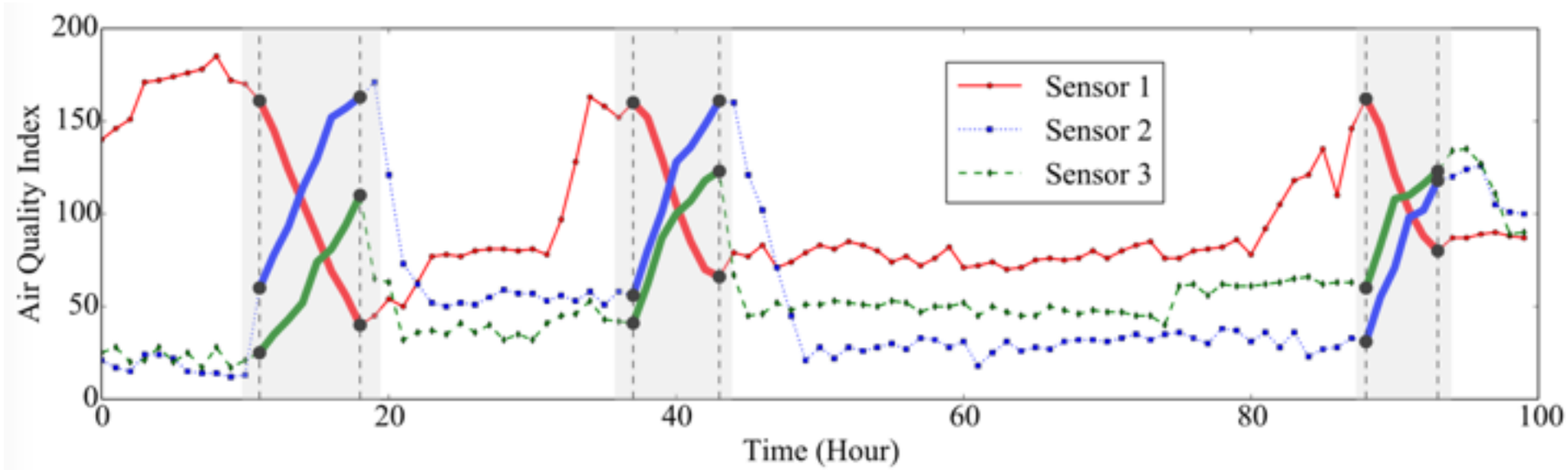
Efficient Discovery of Spatial Co-evolving Patterns in Massive Geo-sensory Data

Chao Zhang[1], Yu Zheng[2], Xiuli Ma[3], Jiawei Han[1]
[1]UIUC, [2]Microsoft, [3]PKU
czhang82@illinois.edu

Big Geo-Sensory Data is Ubiquitous

- Wireless sensor network (WSN): multiple sensors are deployed at different locations to monitor the target condition cooperatively.
- The geo-sensory data is becoming **big**
  - A modern WSN can contain hundreds of sensors, with each sensor collecting millions of records.

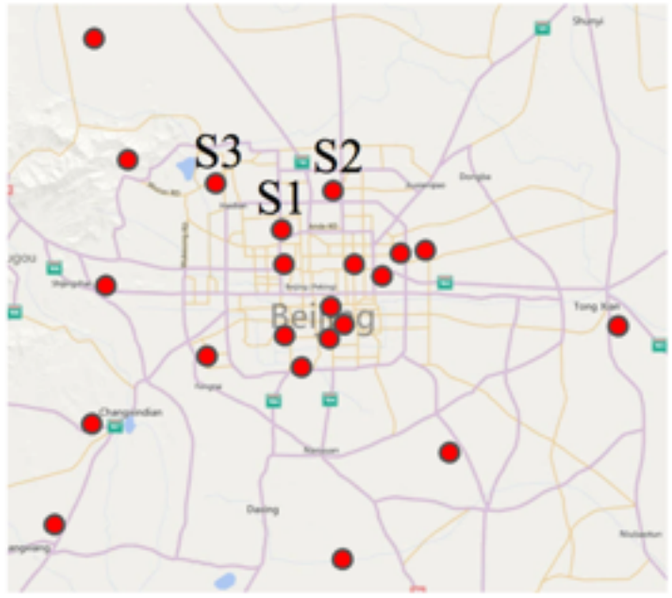




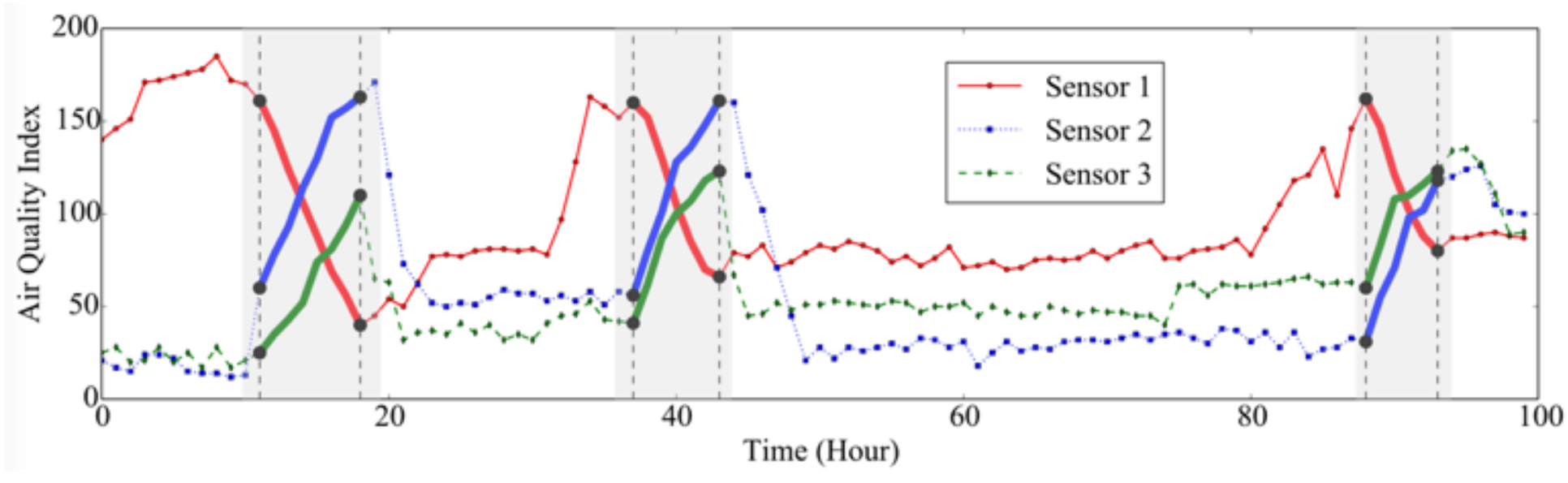
Spatial Co-evolving Pattern: An Example

- Goal: mining a set of ***spatially correlated sensors*** that exhibit ***frequent co-evolution.***
- Frequent co-evolution for [s1, s2, s3]
  - s1 decreases in AQI, s2 and s3 increase in AQI

$$\{[-20/h, -15/h], [+15/h, +20/h], [+15/h, +20/h]\}$$

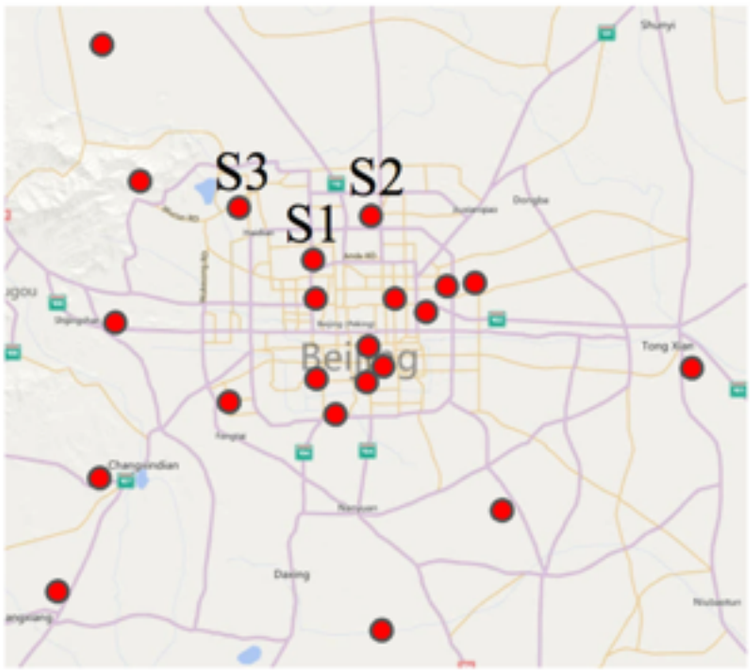


  - Caused by traffic flow in off-work hours



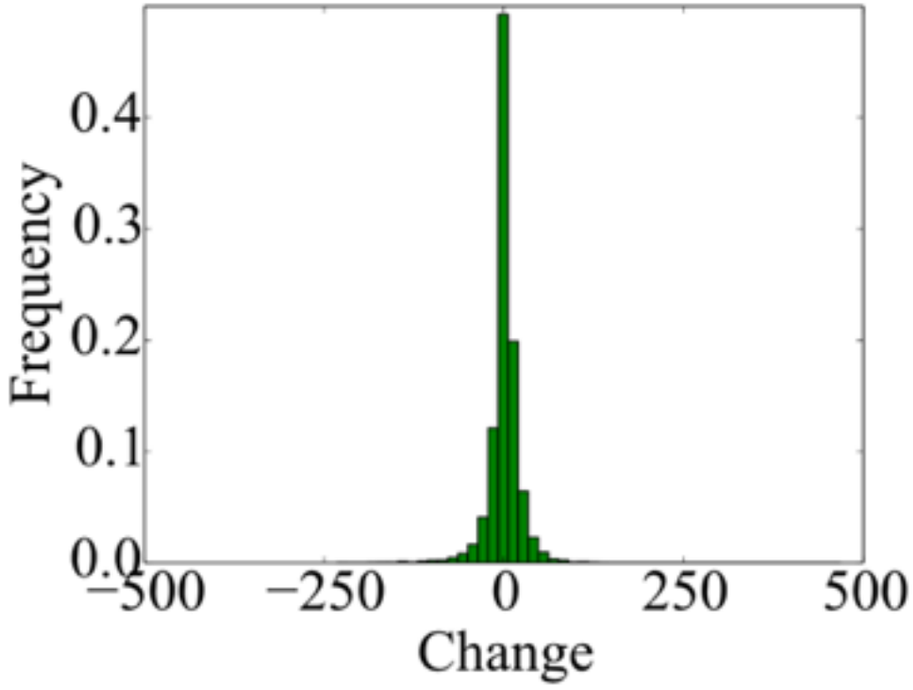
Spatial Co-evolving Pattern Mining

- A spatial co-evolving pattern (SCP) contains
  - a set of spatially connected sensors
  - the frequent co-evolution in their readings
- We aim to find all SCPs from the input geo-sensory data.



Why is it a Challenging Problem?

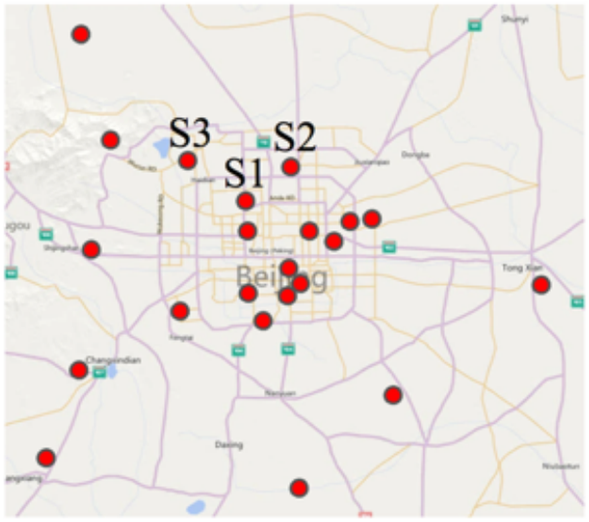
- The truly interesting evolutions are often flooded by numerous trivial fluctuations.
  - Existing motif discovery methods can only find trivial motifs from such data.



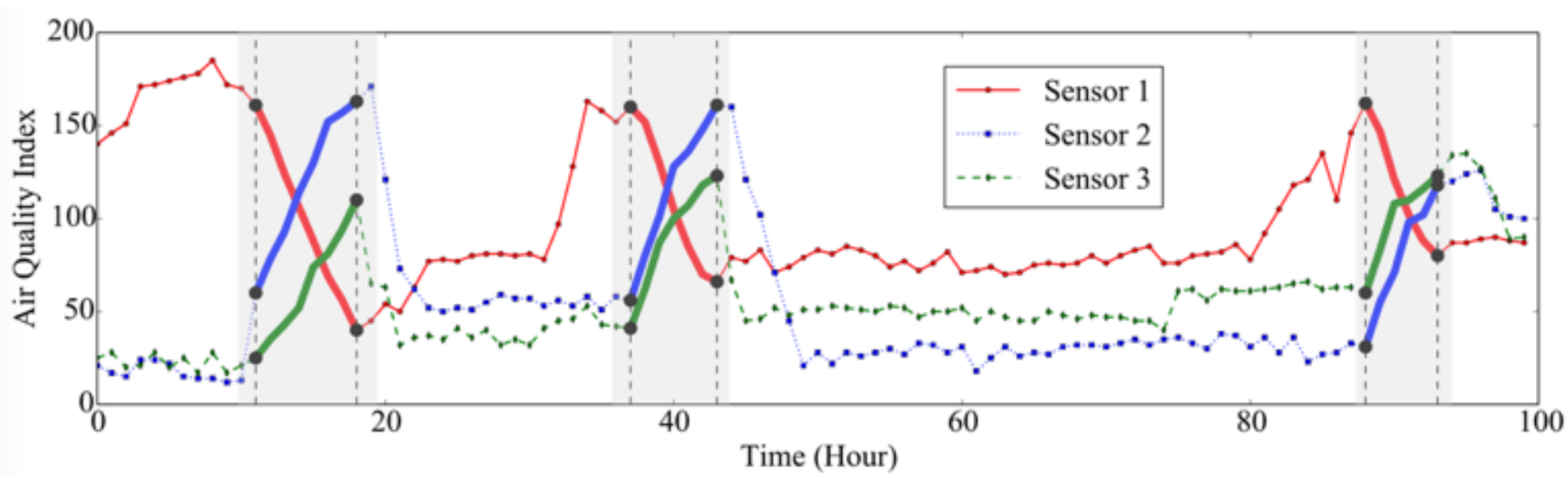
AQI Change Distribution

Why is it a Challenging Problem?

- The combinatorial nature of SCP leads to an extremely large search space.
  - An arbitrary number of sensors.
  - The occurring time intervals of an SCP is uncertain.



Spatial combination

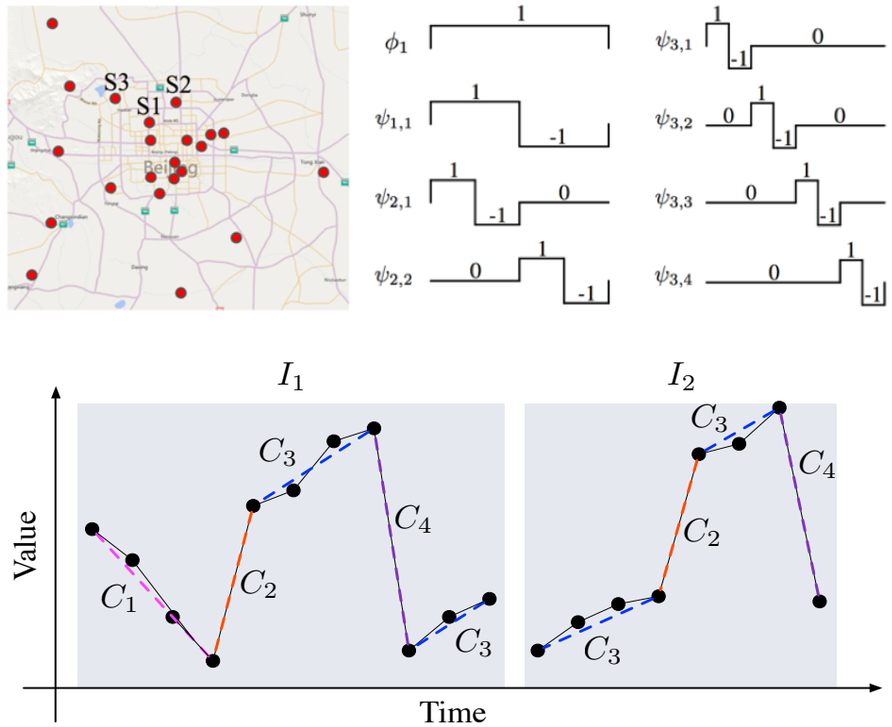


Temporal combination

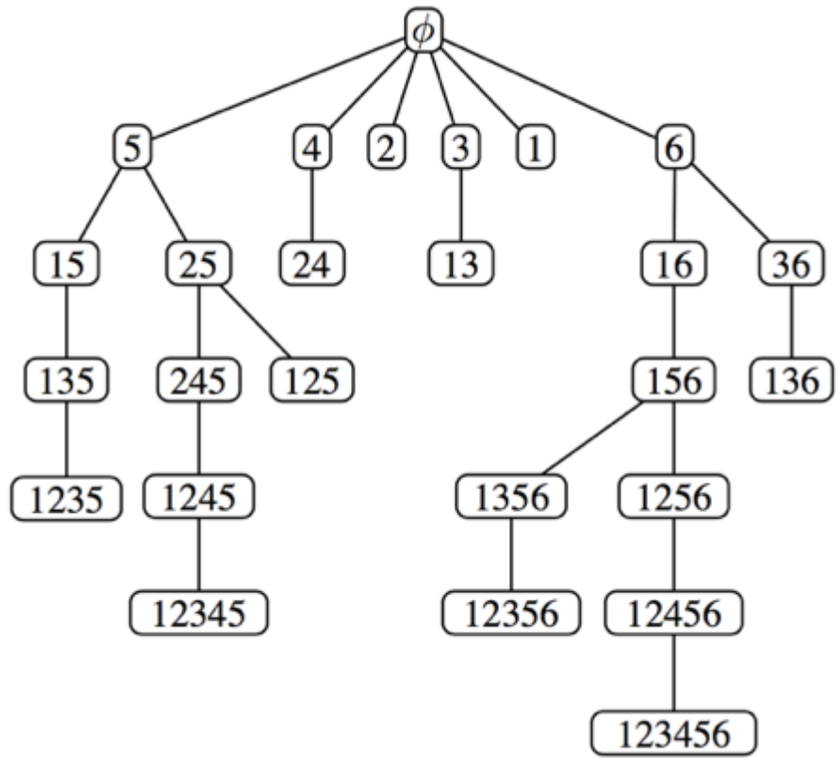
Assembler: A Two-stage Approach

- Stage I: find frequent evolutions for individual sensors.
- Stage II: assemble individual evolutions into SCPs based on the spatial constraint.

Stage 1:

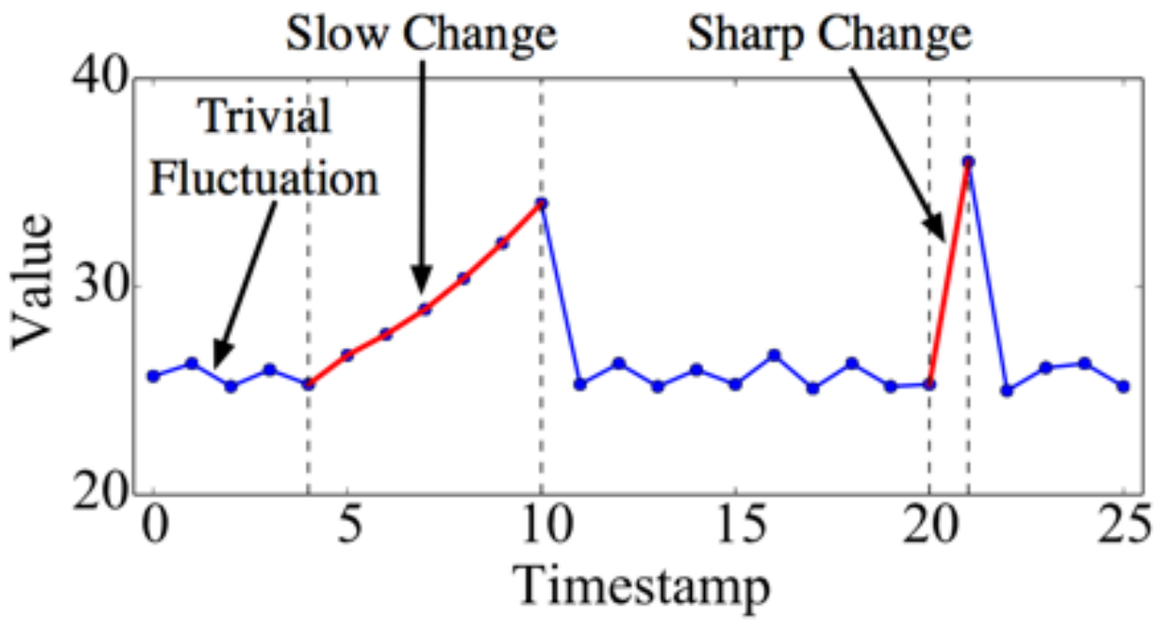


Stage 2:



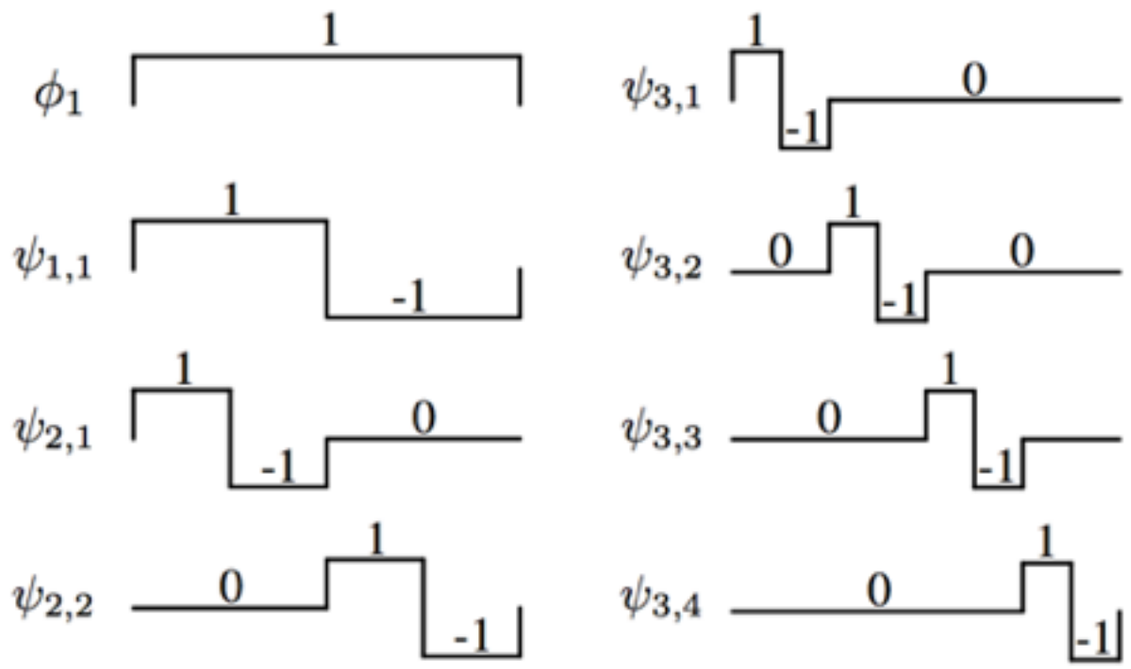
Stage I: Mining Frequent Evolutions for Individual Sensors

- To find interesting evolutions, we must filter trivial fluctuations and identify evolving intervals.
- In geo-sensory data, the changes occur with different rates and durations.



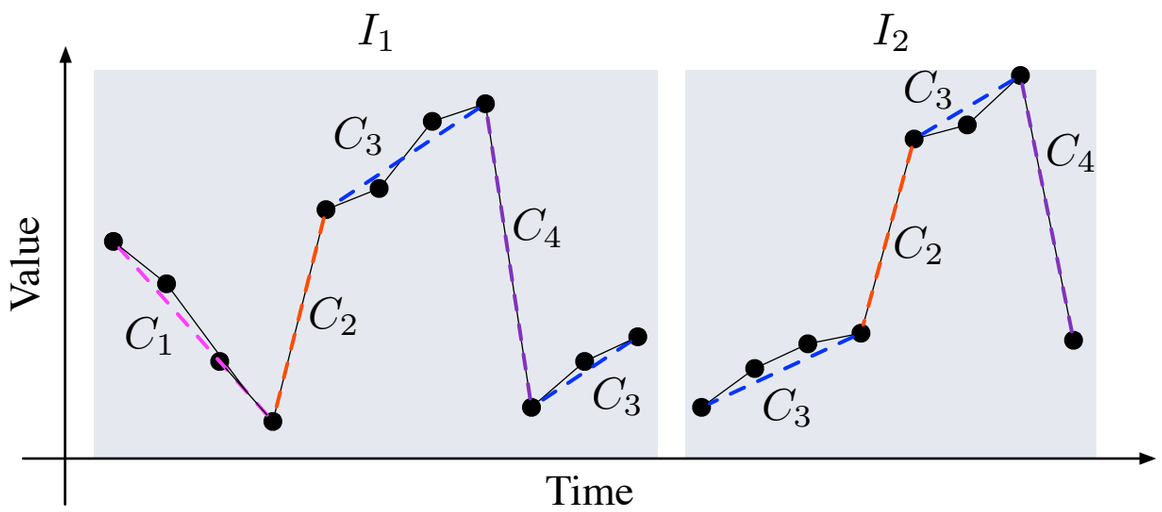
Extract Evolutions using Wavelet Transform

- We capture multi-scale changes using wavelet transform.
  - In the wavelet space, the coefficients of different bases measure the strengths of changes.
  - We preserve large coefficients and discard small coefficients.

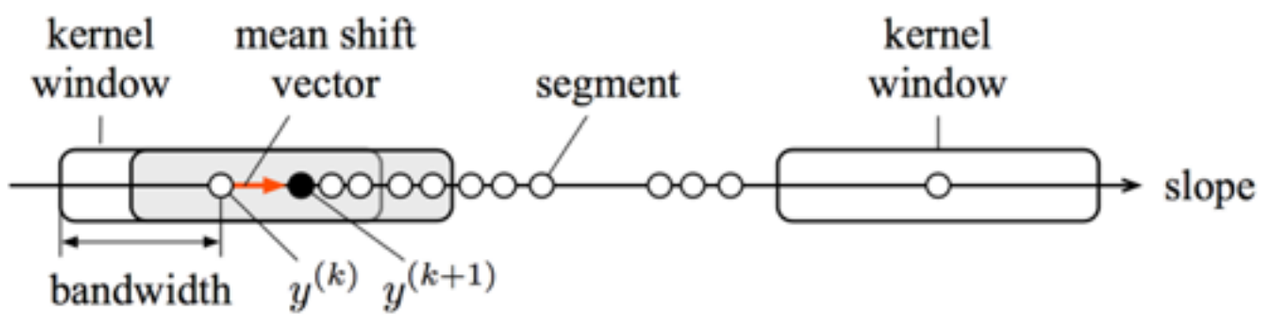


Detecting Frequent Evolutions

- After extracting evolving intervals in the time series, we detect frequent evolutions via clustering.
- A segment-and-group approach:
  - Partition each interval into line segments.
  - Use mean shift to cluster the line segments based on slope (change rate).



Segment



Mean shift clustering

Stage II: SCP Generation

- From individual sensors to groups of sensors
  - Pattern assembling via timestamp intersection.

①

Pattern P_1	[+20/h, +50/h]
Timestamps	$\{t_1, t_3, t_4, t_7, t_9, t_{10}, t_{11}, t_{12}, t_{13}, t_{14}\}$

②

Pattern P_2	[-30/h, -10/h]
Timestamps	$\{t_2, t_4, t_5, t_7, t_9, t_{10}, t_{11}, t_{12}, t_{13}, t_{17}\}$

① — ②

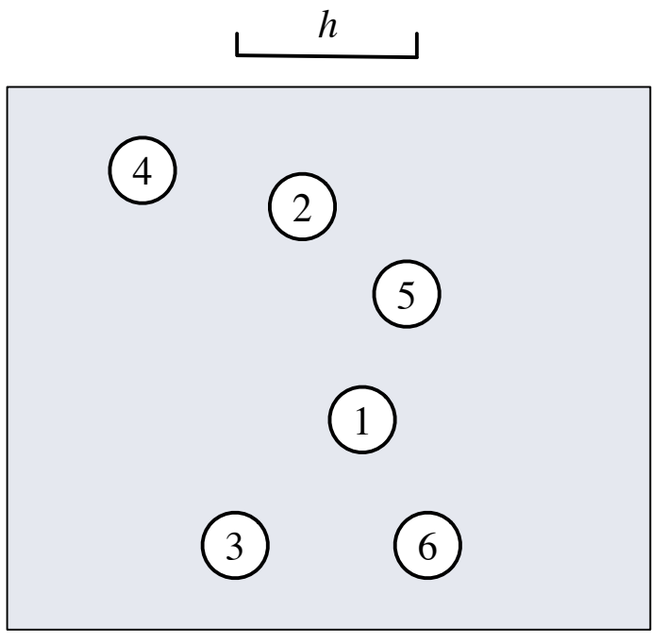
Pattern P_{12}	{[+20/h, +50/h], [-30/h, -10/h]}
Timestamps	$\{t_4, t_7, t_9, t_{10}, t_{11}, t_{12}, t_{13}\}$

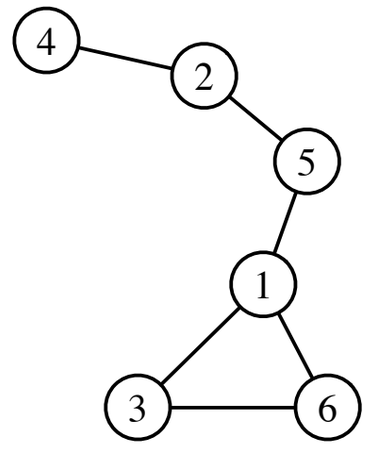
Stage II: SCP Generation

- **Anti-monotonicity**: if one set of sensors have SCP, any of its subsets must also have SCP.
- A baseline method based on Apriori: starting with patterns on individual sensors, obtain SCPs in a bottom-up manner.
- The baseline method is not efficient enough
  - It generates numerous candidates and keep them in memory.
  - It performs pair-wise comparison to examine whether two candidates can be joined.

Stage II: SCP Generation

- Can we leverage the spatial constraint to generate SCPs more efficiently?
- The connectivity graph
  - Each set of spatially connected sensors corresponds to a connected component in the connectivity graph.

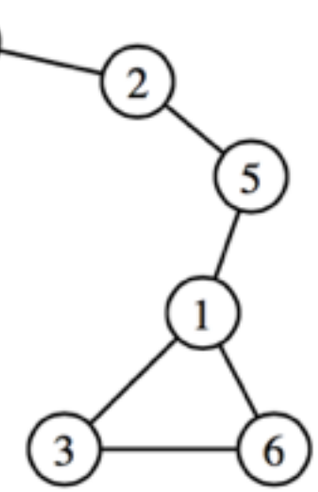




Parent Relation

- We define the parent relation between two connected components.

DEFINITION 10 (PARENT). *Let Y be a size-$(k+1)$ connected component in a connectivity graph G. Given a vertex ordering $\mathcal{V}$, the roll-up operation on Y removes one vertex s from Y such that: (1) the result set $X = Y - \{s\}$ is still connected; (2) s is the first possible vertex in $\mathcal{V}$ on the premise of satisfying Condition (1). We say X is the parent of Y, and Y is a child of X.*



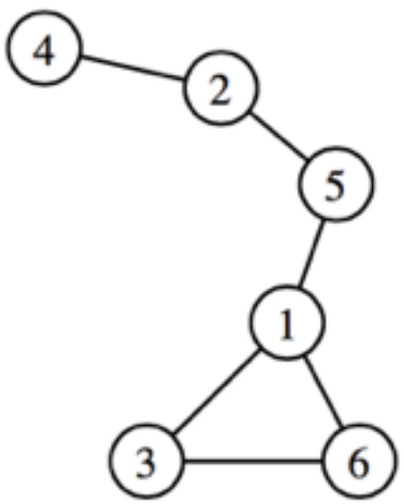
Example:

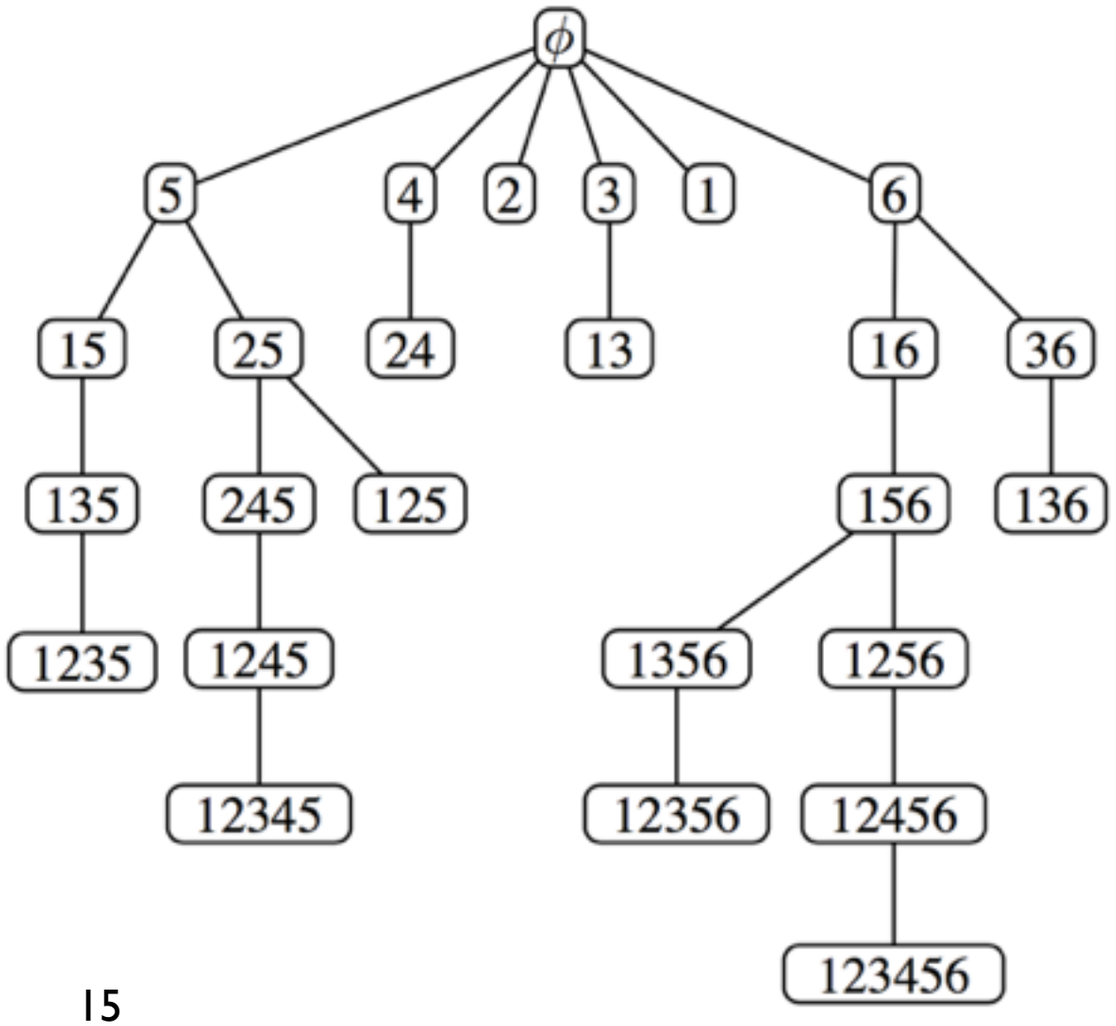
Suppose V = 1 -> 2 -> 3 -> 4 -> 5 -> 6, then the parent relation generates:

{245} -> {25} -> {5} -> ϕ

The SCP Search Tree

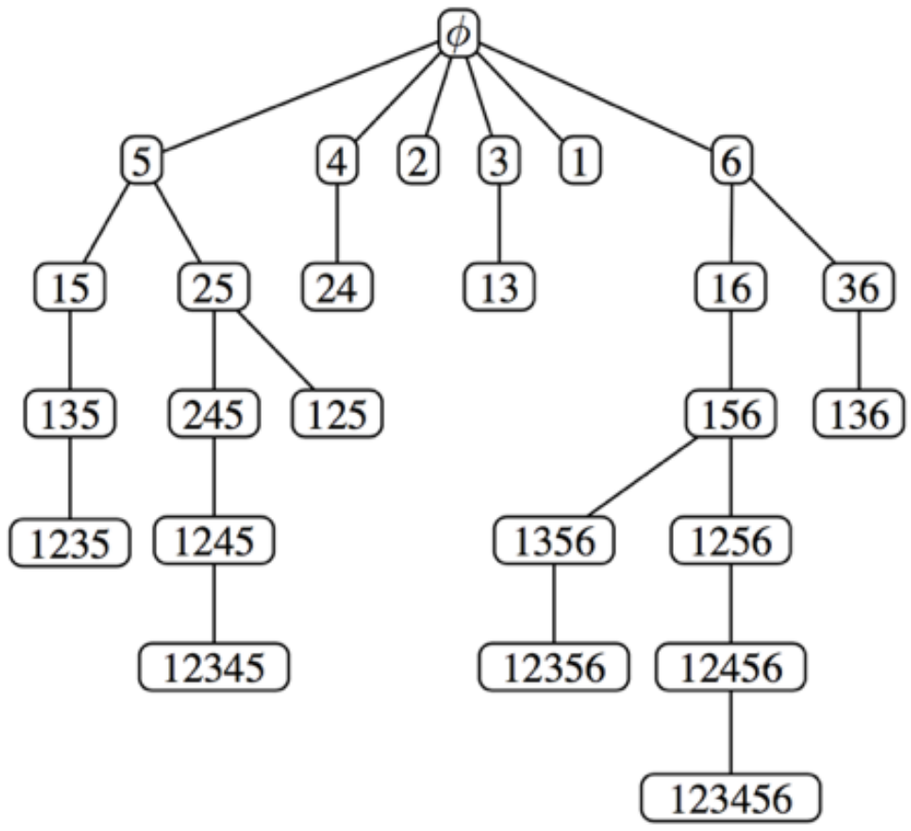
- Starting from any connected component, by performing the roll-up operation, we can reach the same node ϕ.
- A tree structure: each node is a connected component along with the SCPs occurring on it.





Reverse Search of SCPs

- Starting from the root node, we perform depth-first construction of the SCP search tree
  - SCPs are obtained on-the-fly
  - Unqualified branches are pruned with anti-monotonicity.

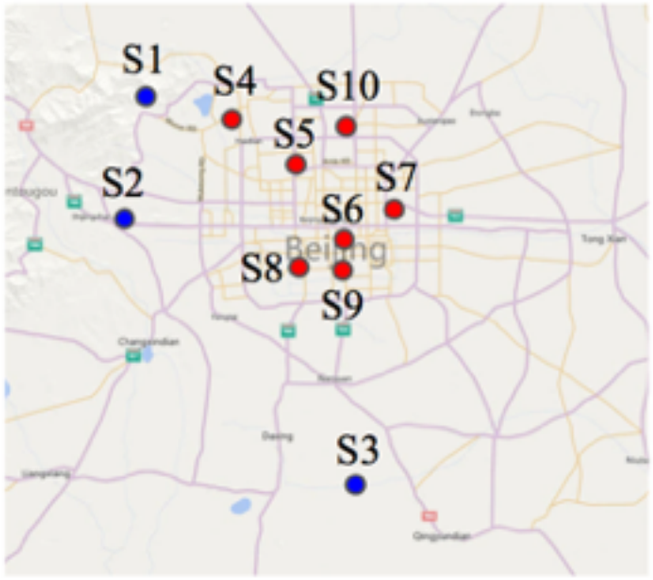


Experimental Evaluation

- Data sets
  - Air: the AQI data collected by 180 sensors in northern China during 1.5 years.
  - Bike: the bike rental data of 332 docks in New York during one year.

Example SCPs

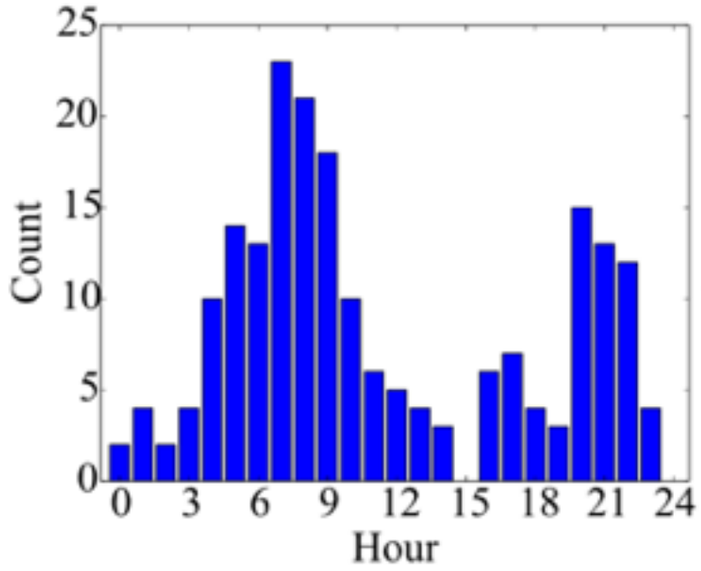
- On the Air data set:



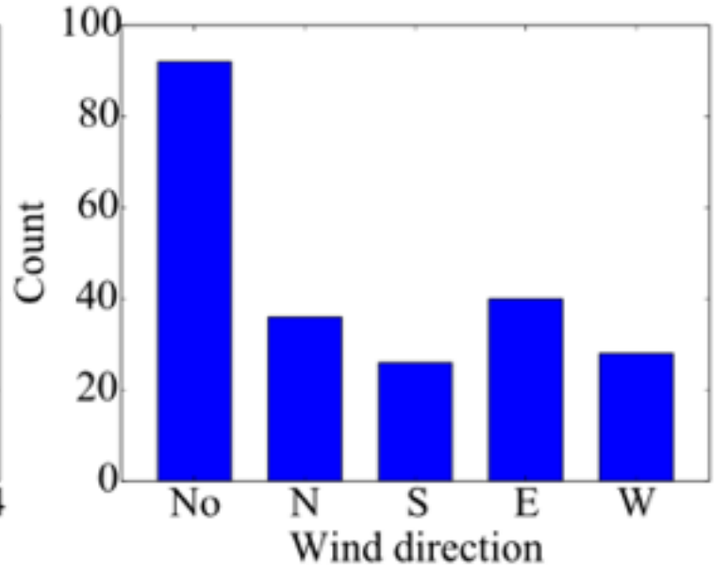
Pattern 1

$$\begin{pmatrix} s_1: & -17.4 \pm 5.1 \\ s_2: & -18.6 \pm 6.5 \\ s_3: & -29.1 \pm 4.5 \\ s_4: & +16.2 \pm 3.7 \\ s_5: & +45.9 \pm 3.1 \\ s_6: & +56.5 \pm 7.3 \\ s_7: & +45.3 \pm 6.2 \\ s_8: & +31.0 \pm 3.7 \\ s_9: & +26.1 \pm 4.9 \\ s_{10}: & +21.6 \pm 8.6 \end{pmatrix}$$

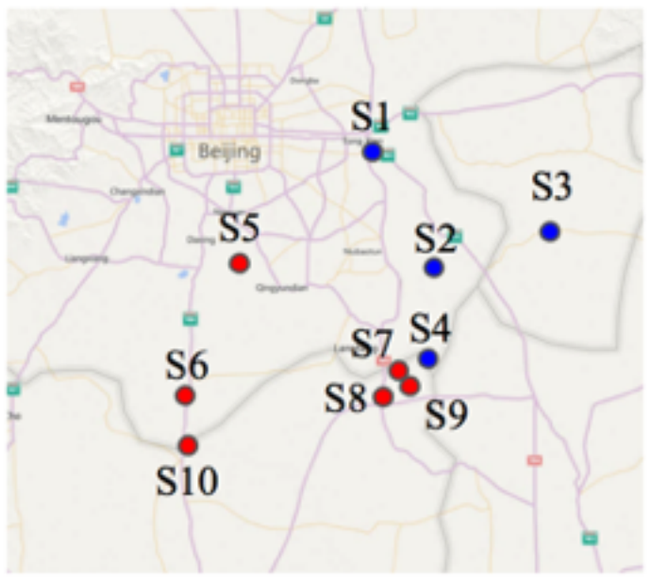
AQI Change



Hour Distribution



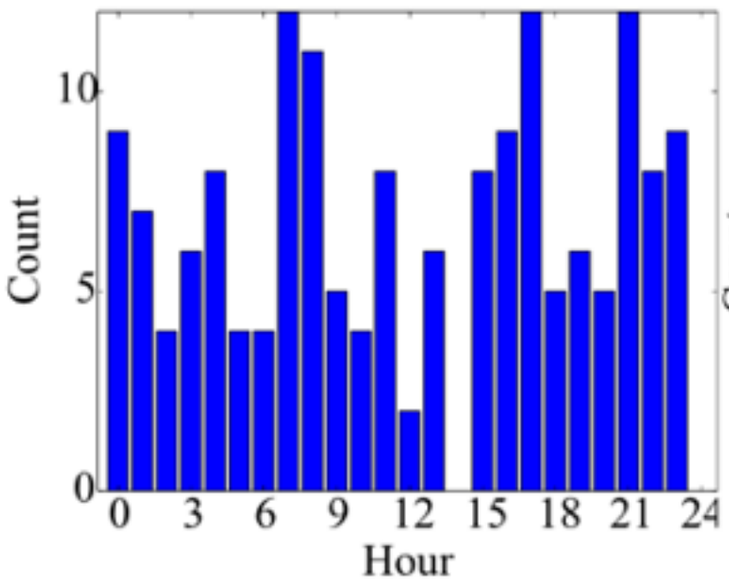
Wind Direction



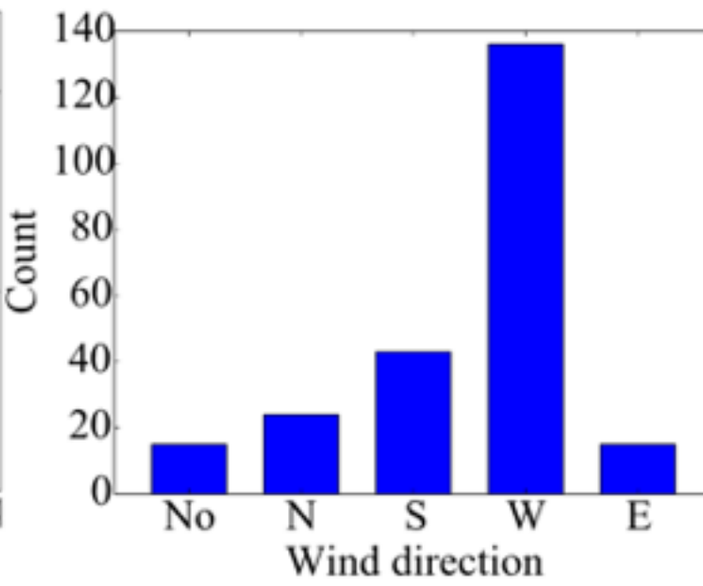
Pattern 2

$$\begin{pmatrix} s_1: & -44.3 \pm 3.6 \\ s_2: & -41.2 \pm 7.8 \\ s_3: & -51.6 \pm 6.5 \\ s_4: & -31.7 \pm 9.5 \\ s_5: & +51.8 \pm 5.5 \\ s_6: & +31.2 \pm 8.5 \\ s_7: & +56.5 \pm 6.6 \\ s_8: & +45.5 \pm 8.3 \\ s_9: & +35.2 \pm 7.2 \\ s_{10}: & +36.9 \pm 7.3 \end{pmatrix}$$

AQI Change



Hour Distribution

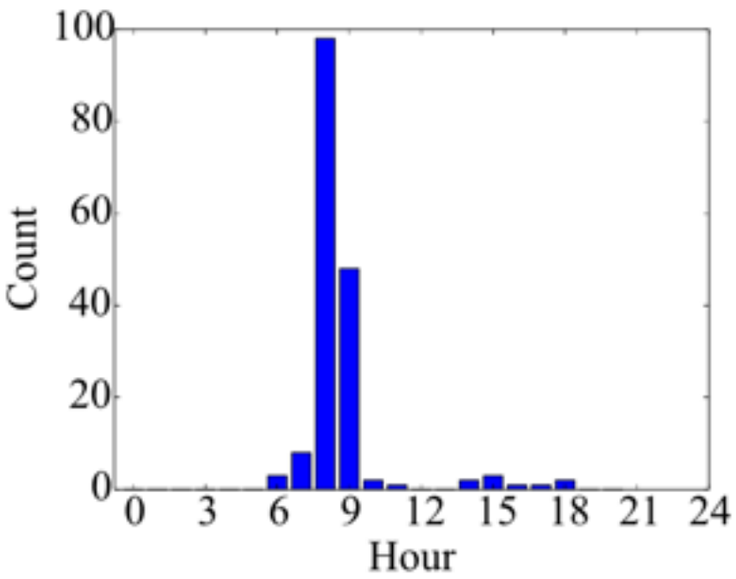


Wind Direction

Example SCPs

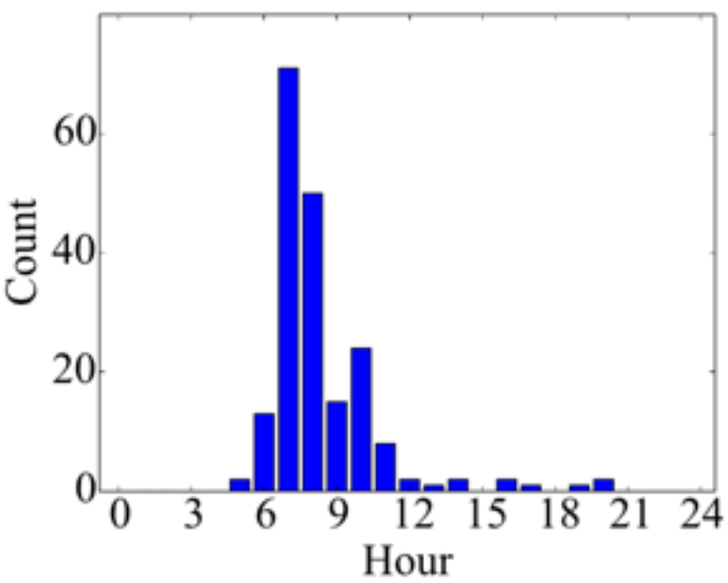
- On the Bike data set:

Pattern 1



Hour Distribution

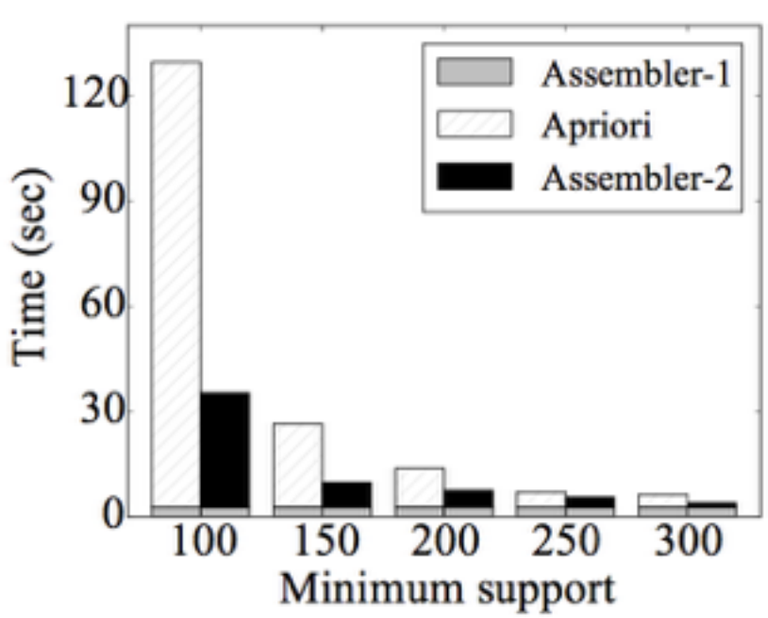
Pattern 2



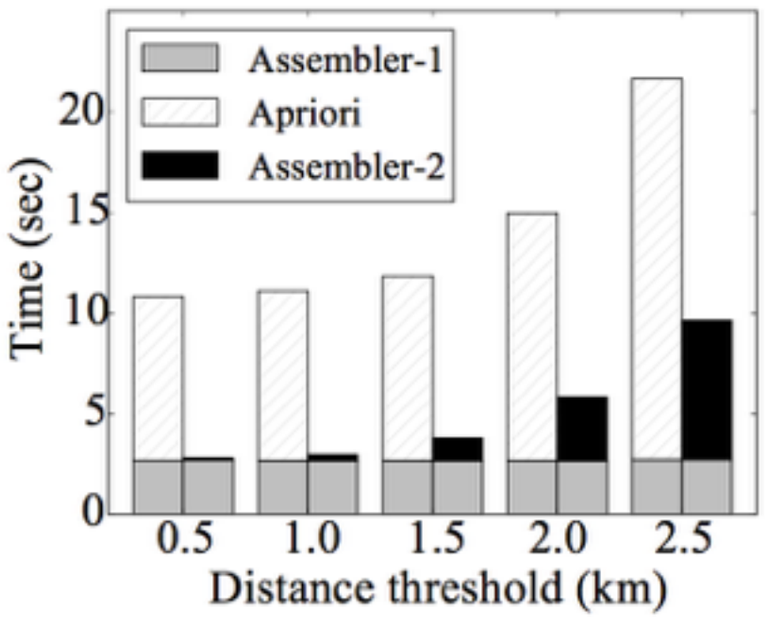
Hour Distribution

Running Time Comparison

- Efficiency

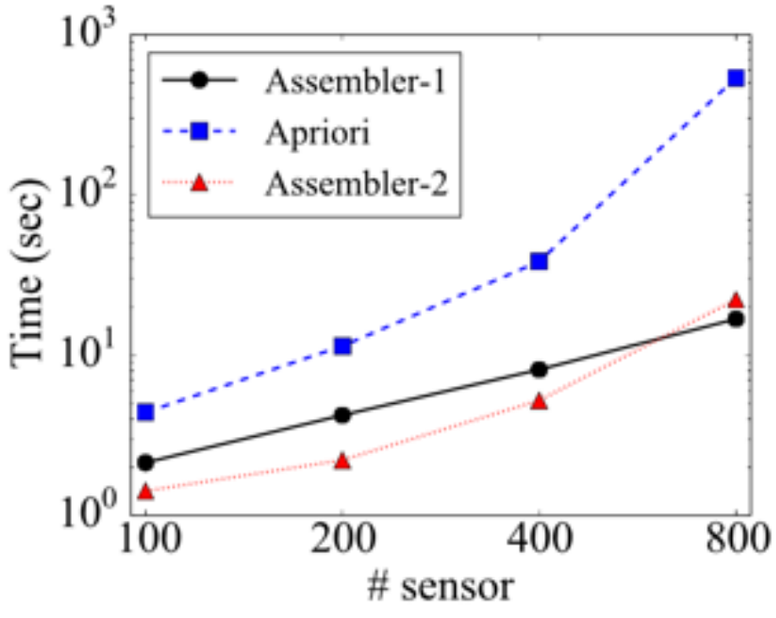


(a) Time *v.s.* θ.

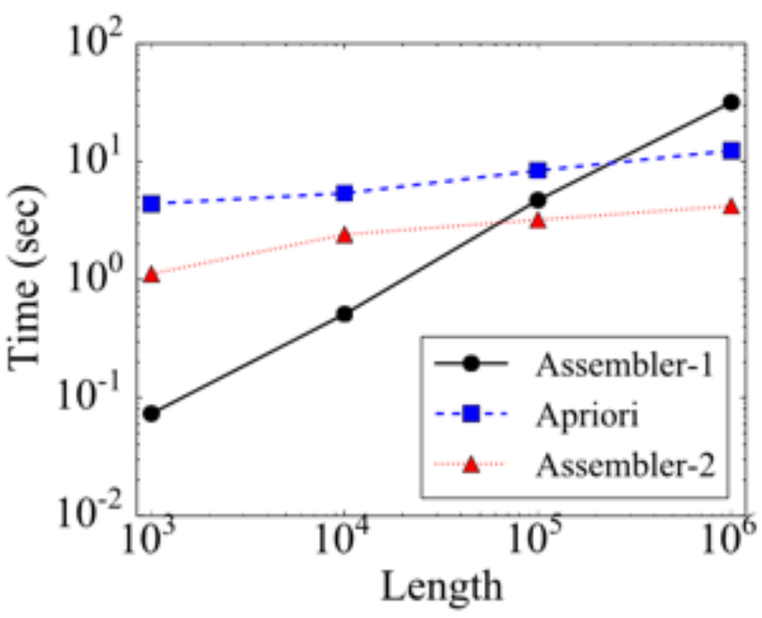


(b) Time *v.s.* h.

- Scalability



(a) Time *v.s.* number of sensors n.



(b) Time *v.s.* sequence length m.

Summary

- We study the problem of mining spatial co-evolving patterns from massive geo-sensory data.
- We propose the two-stage method Assembler:
  - Stage 1: it obtains frequent evolutions for individual sensors.
  - Stage 2: it assembles single patterns into SCPs.
- The experiment results show that Assembler is effective and efficient.