Splitter: Mining Fine-Grained Sequential Patterns in Semantic Trajectories

Chao Zhang[1], Jiawei Han[1], Lidan Shou[2], Jiajun Lu[1], Thomas La Porta[3]

czhang82@Illinois.edu

[1] University of Illinois at Urbana-Champaign
[2] Zhejiang University
[3] Penn State University

Frequent Sequential Pattern for Discrete Data

- Sequential pattern: a subsequence that matches at least n sequences (n is the support threshold) in the database.

Sequence Database:

C -> A -> B

A -> C -> B -> A

A -> B

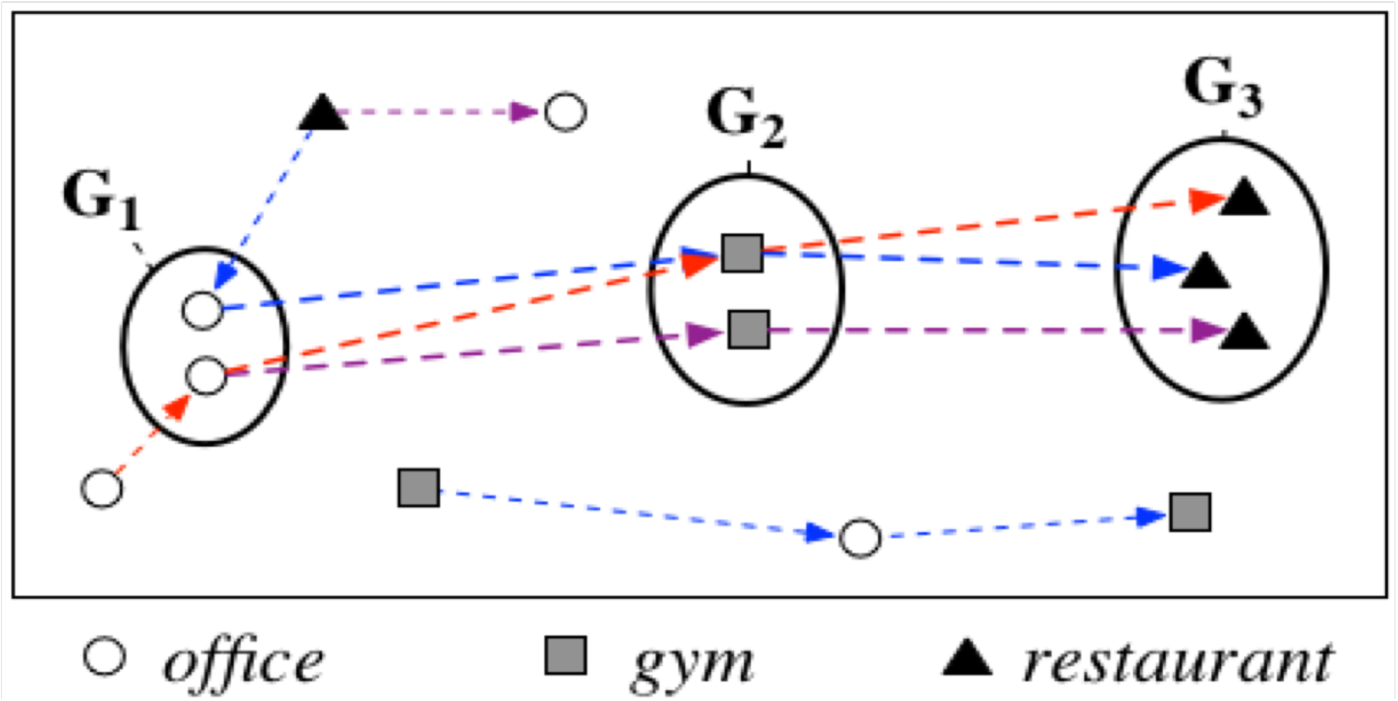
B -> C

Sequential Pattern (n=3):

A -> B

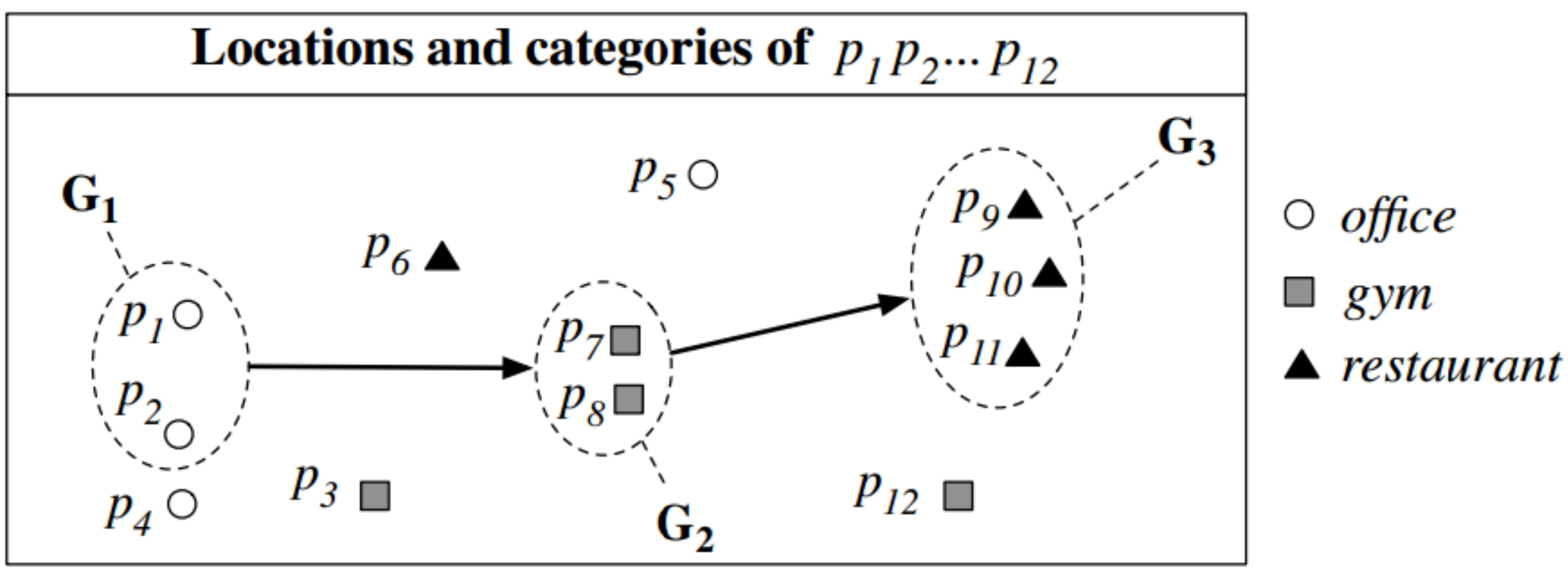
How do We Define Sequential Movement Pattern?

- Can we define it as a *place sequence* that matches at least n trajectories? No.
- Due to space continuity, similar places need to be grouped to collectively form frequent patterns.



How do We Define Sequential Movement Pattern?

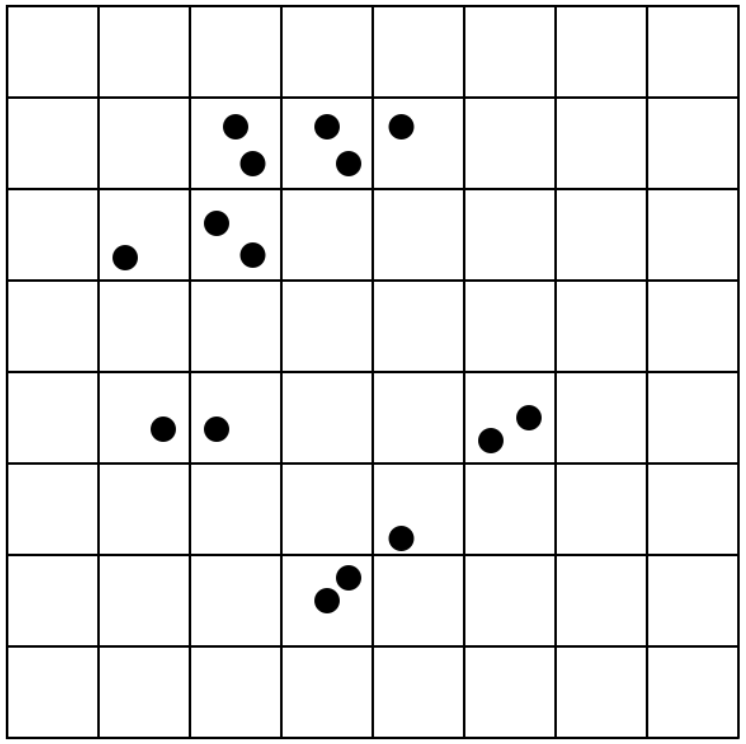
- Meaningful patterns must satisfy three constraints:
  - Semantic consistency
  - Spatial compactness
  - Temporal continuity



Sequential Pattern: G_1 -> G_2 -> G_3

Existing Approaches

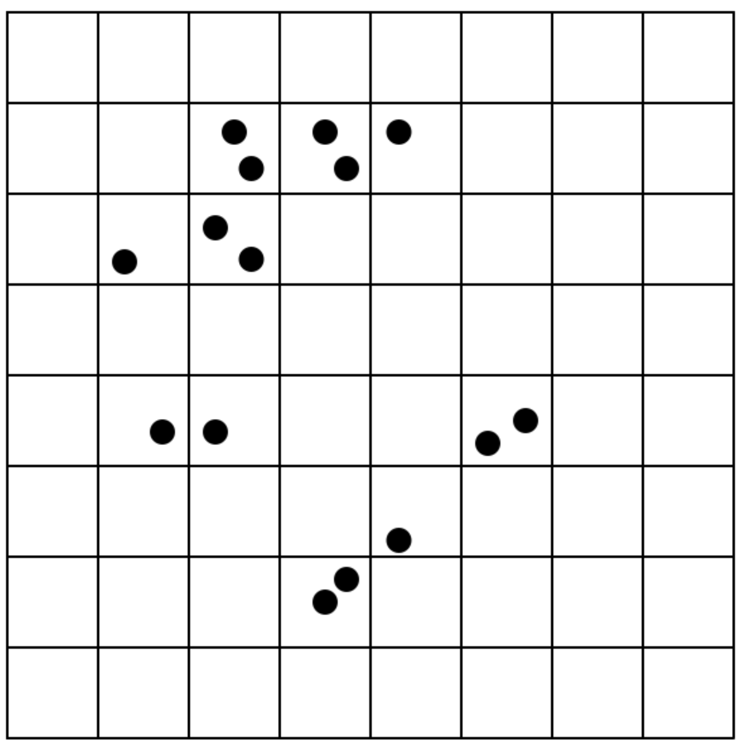
- Trajectory Pattern Mining [1, 2, 3]:
  - Partition the space into small grids
  - Group the places in the same grid (or several neighboring grids)
  - Mine frequent sequential patterns.



[1] I. Tsoukatos et. al., Efficient mining of spatiotemporal patterns, SSTD, 2001.
[2] J. Wang et. al., Flowminer: Finding flow patterns in spatio-temporal databases, ICTAI, 2004
[3] F. Giannotti et. al., Trajectory Pattern Mining, KDD, 2007.

Existing Approaches

- Drawbacks of rigid space partitioning:
  - It suffers from the sharp boundary problem.
  - It is hard to pre-specify the partition granularity.

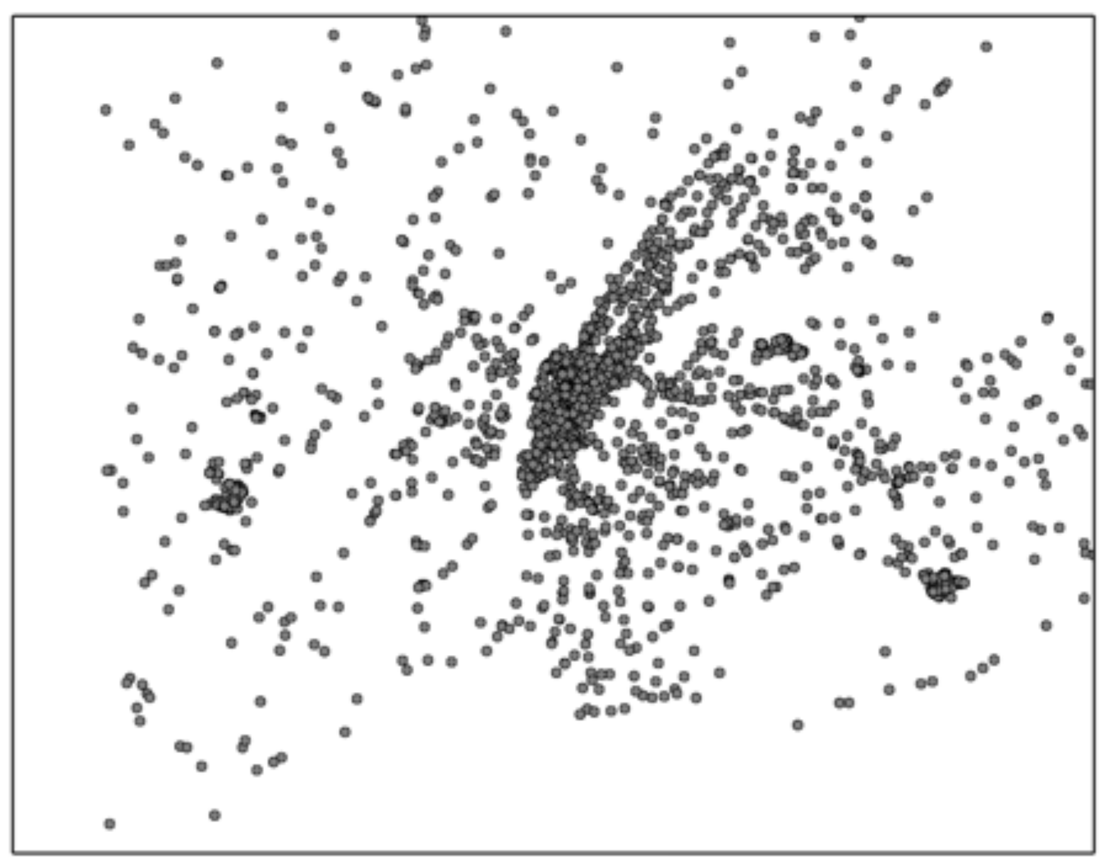


An Overview of Splitter

- Splitter is a two-step approach.
  - Step 1: mining coarse patterns that satisfy the semantic and temporal constraints.
  - Step 2: splitting each coarse patterns into fine-grained ones to meet the spatial constraint.

Mining Coarse Patterns

- Group the places by category such that the places having the same category go to the same group.
  - We obtain groups like office, gym, restaurant, etc.
  - Each group can be viewed as an independent item.



Mining Coarse Patterns

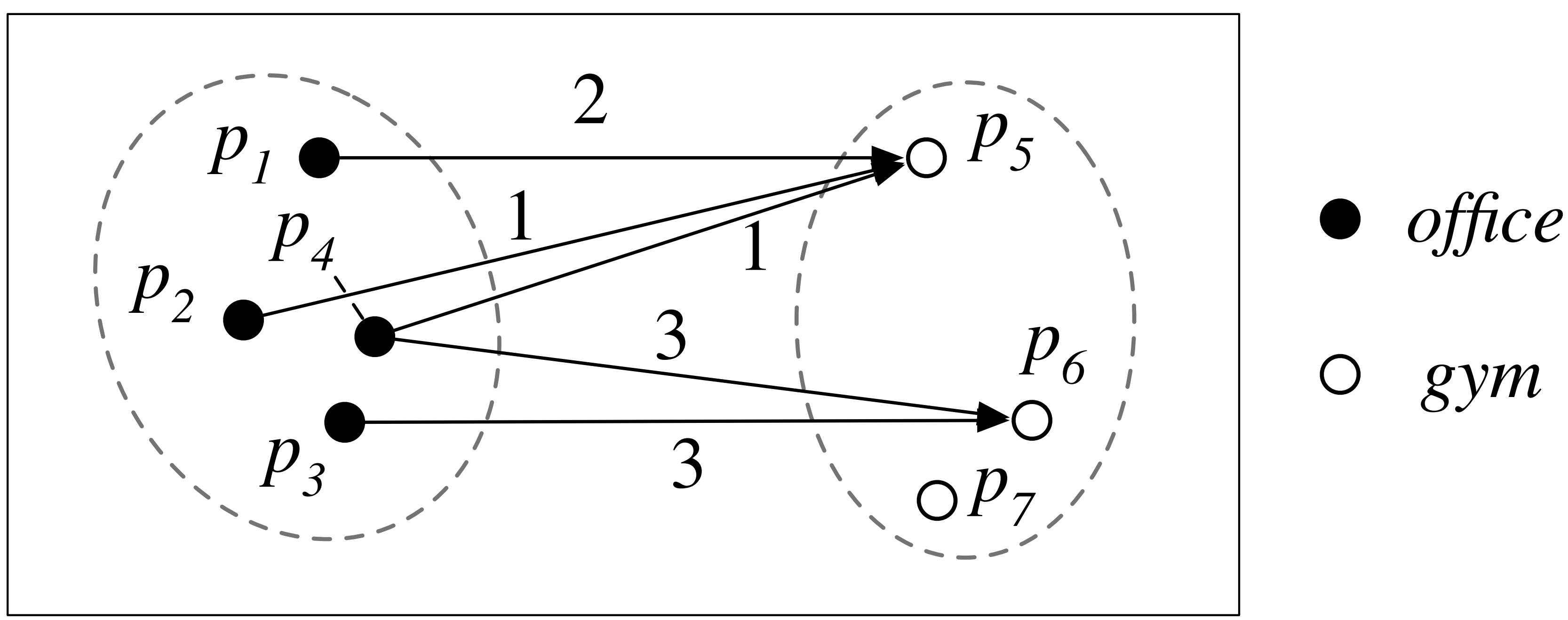
- Transform trajectories into item sequences, by mapping the place ids to the group ids.

Object	Semantic Trajectory
o_1	$<(p_3, 0), (\boldsymbol{p_1}, 10), (\boldsymbol{p_7}, 30), (\boldsymbol{p_9}, 40)>$
o_2	$<(p_5, 0), (p_7, 30), (\boldsymbol{p_2}, 360), (\boldsymbol{p_7}, 400), (\boldsymbol{p_{10}}, 420)>$
o_3	$<(p_3, 0), (p_6, 30)>$
o_4	$<(p_2, 0), (\boldsymbol{p_1}, 120), (p_6, 140), (\boldsymbol{p_8}, 150), (\boldsymbol{p_{11}}, 180)>$
o_5	$<(p_{12}, 50), (p_8, 80), (p_{11}, 120), (p_4, 210)>$

Object	Timestamped item sequence
o_1	$\langle (G_2, 0), (G_1, 10), (G_2, 30), (G_3, 40) \rangle$
o_2	$\langle (G_1, 0), (G_2, 30), (G_1, 360), (G_2, 400), (G_3, 420) \rangle$
o_3	$\langle (G_2, 0), (G_3, 30) \rangle$
o_4	$\langle (G_1, 0), (G_1, 120), (G_3, 140), (G_2, 150), (G_3, 180) \rangle$
o_5	$\langle (G_2, 50), (G_2, 80), (G_3, 120), (G_1, 210) \rangle$

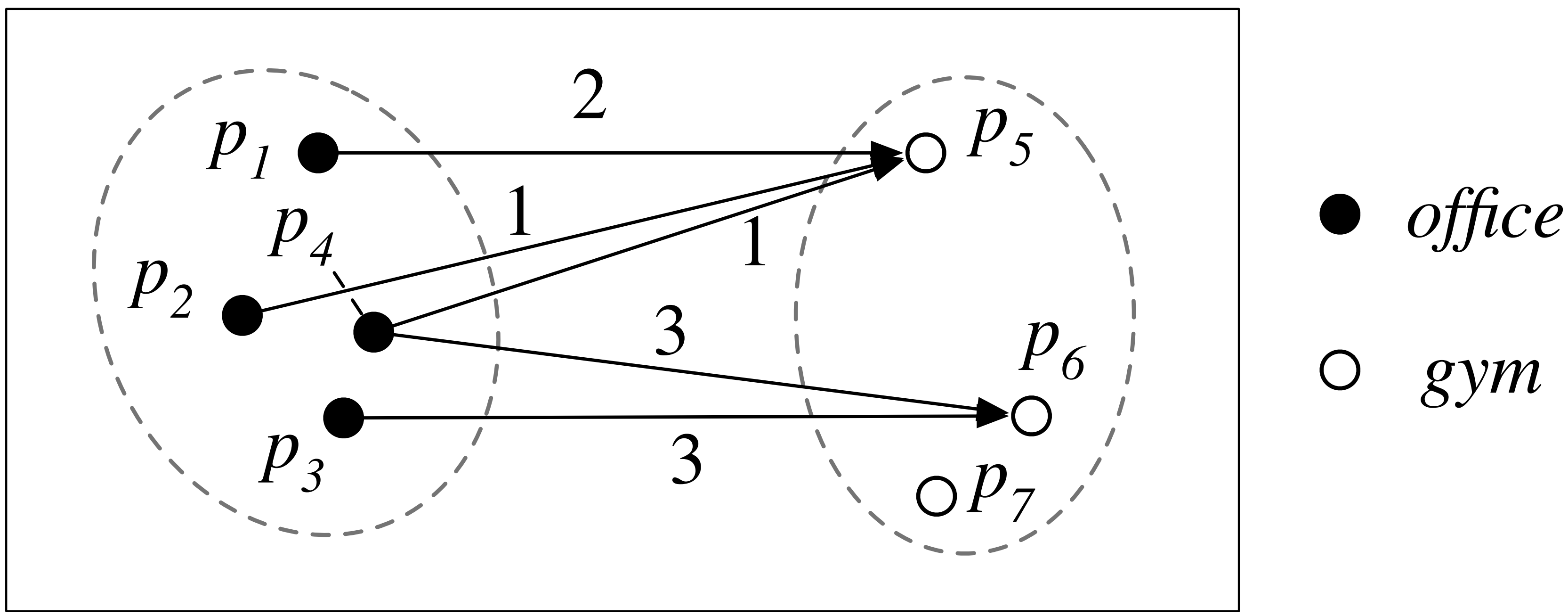
Mining Coarse Patterns

- We modify PrefixSpan by using the full projection principle.
  - It guarantees the result patterns satisfy the time constraint.
  - It extracts the snippets (place sequences) for each coarse pattern.



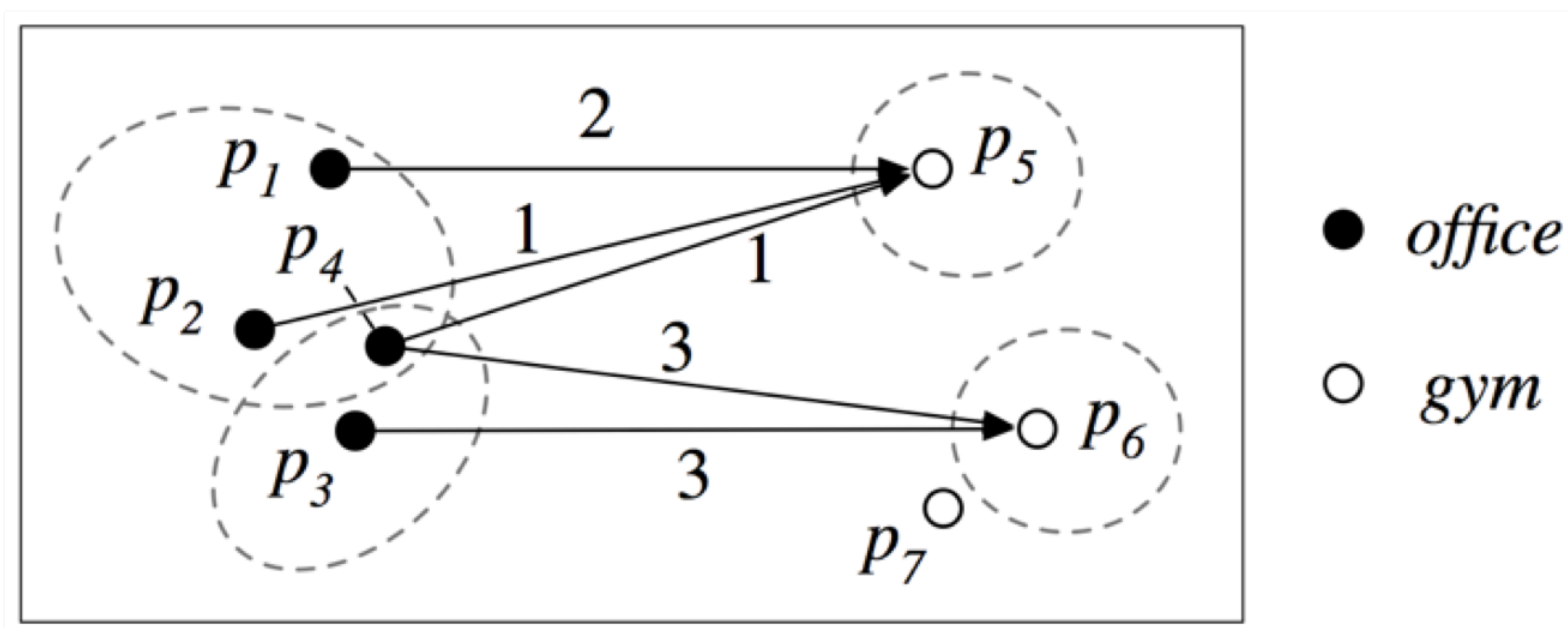
What do We Have Now?

- A set of coarse patterns.
- The snippets for each coarse pattern.



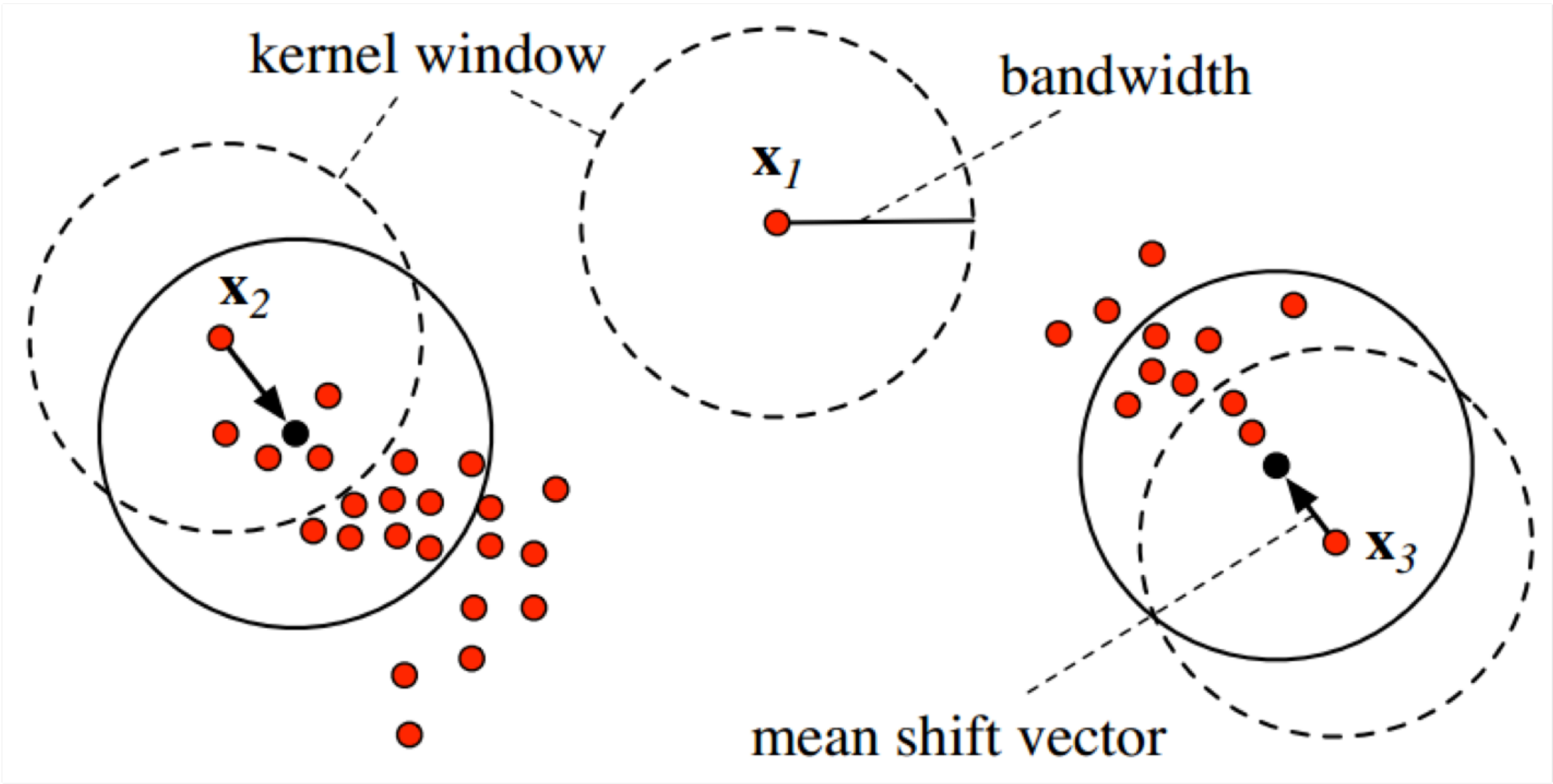
Splitting Coarse Patterns

- We find fine-grained patterns by merging close snippets.
  - Each snippet is mapped to a weighted high-dimensional point (e.g., length-2 snippets are mapped to 4D points).
  - Detect dense and compact clusters in the high-dimensional space spanned by the snippets.



Splitting Coarse Patterns

- Finding snippet clusters via *weighted snippet shift*:



Splitting Coarse Patterns

- Top-down pattern discovery:
  - Start with an initially large bandwidth.
  - Gradually dampen the bandwidth and find patterns on-the-fly.
  - Terminate until no more pattern can be found.
- We introduce a divide-and-conquer strategy to speed up the top-down discovery process.

Experimental Data

- A Foursquare check-in data set:
  - ~15K users in New York.
  - ~50K places.
  - 15 categories.
- Two synthetic data sets generated by the Brinkhoff's network-based generator.

Compared Methods

- Grid
  - Trajectory pattern mining [1] based on space partitioning.
- HC
  - Group the places via top-down hierarchical clustering.
  - Mine movement patterns using PrefixSpan.

[1] F. Giannotti et. al., Trajectory Pattern Mining, KDD, 2007.

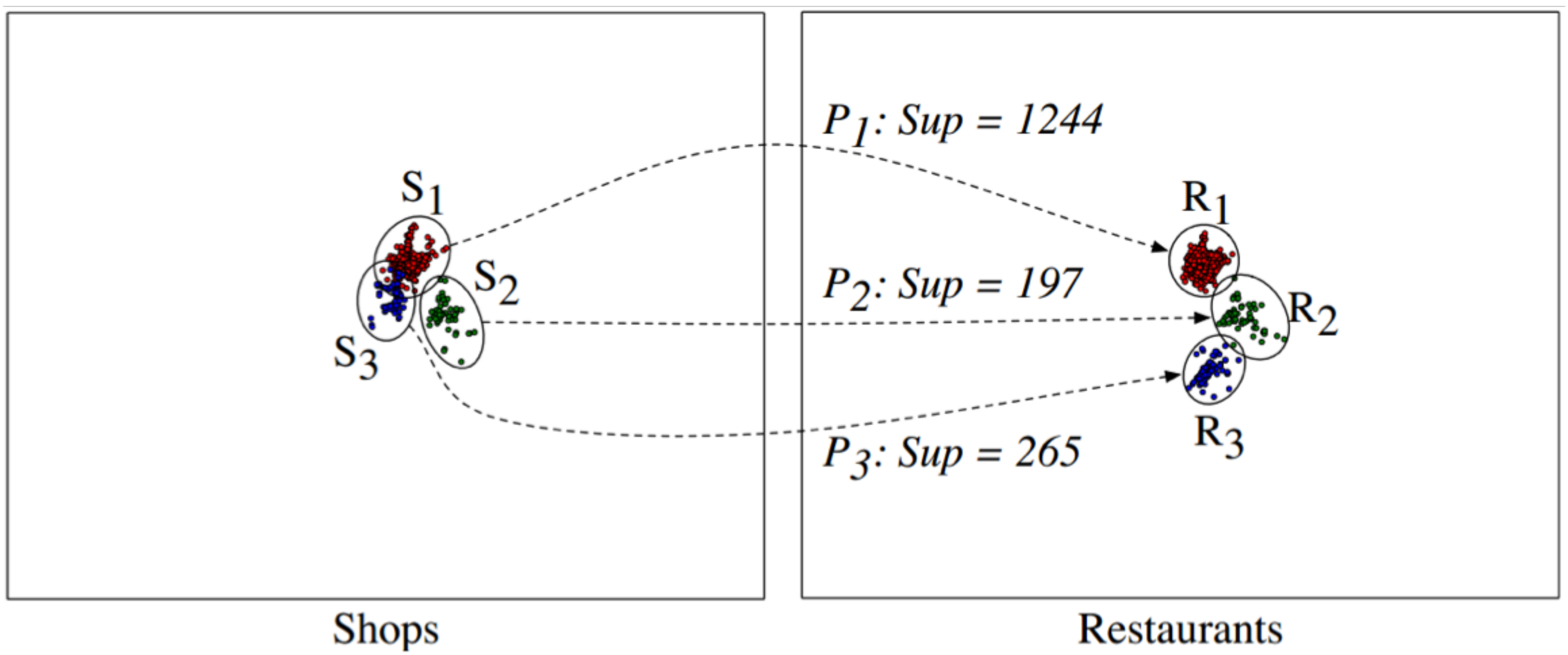
Example Coarse Patterns

- Support threshold n = 100

	Pattern	Sup
length=2	Shop → Food	1819
	Food → Shop	1464
	Professional → Nightlife Spot	1121
	Outdoor → Food	947
	Residence → College & University	647
length=3	Shop → Food → Shop	262
	Professional → Food → Nightlife Spot	240
	Entertainment → Food → Shop	178
	Transportation → Shop → Shop	174
	Residence → Outdoor → Food	163

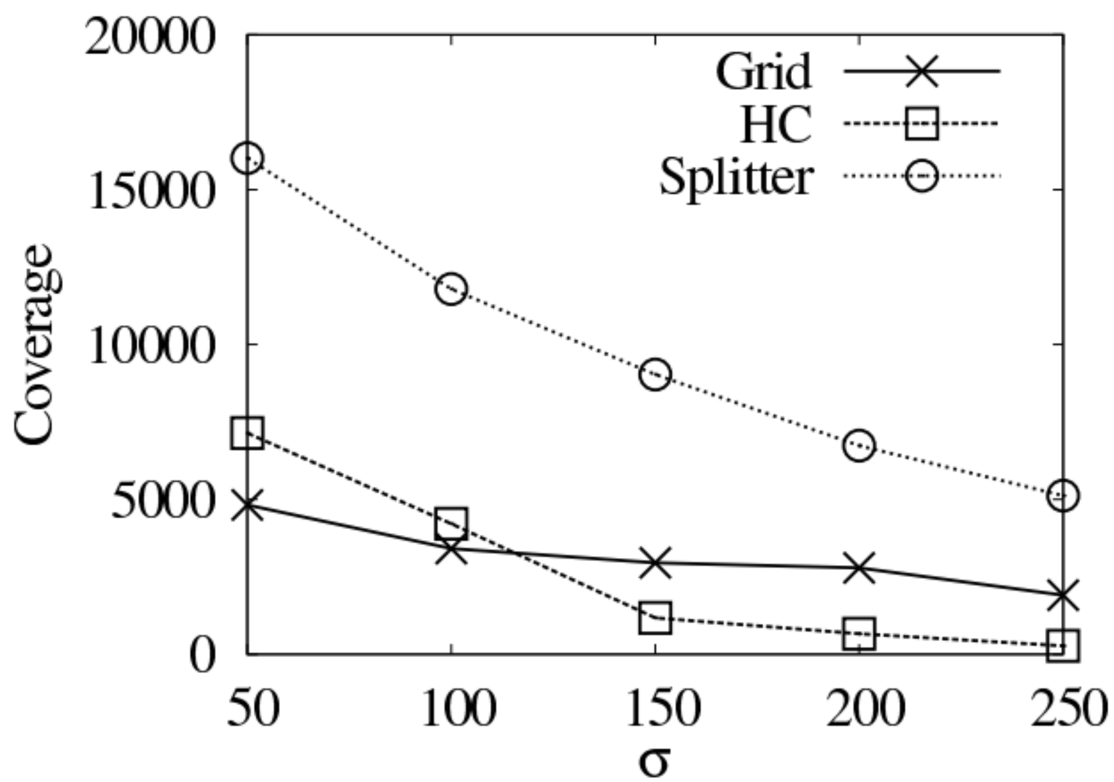
Example Fine-Grained Patterns

- Length-2 patterns for Shop -> Food:

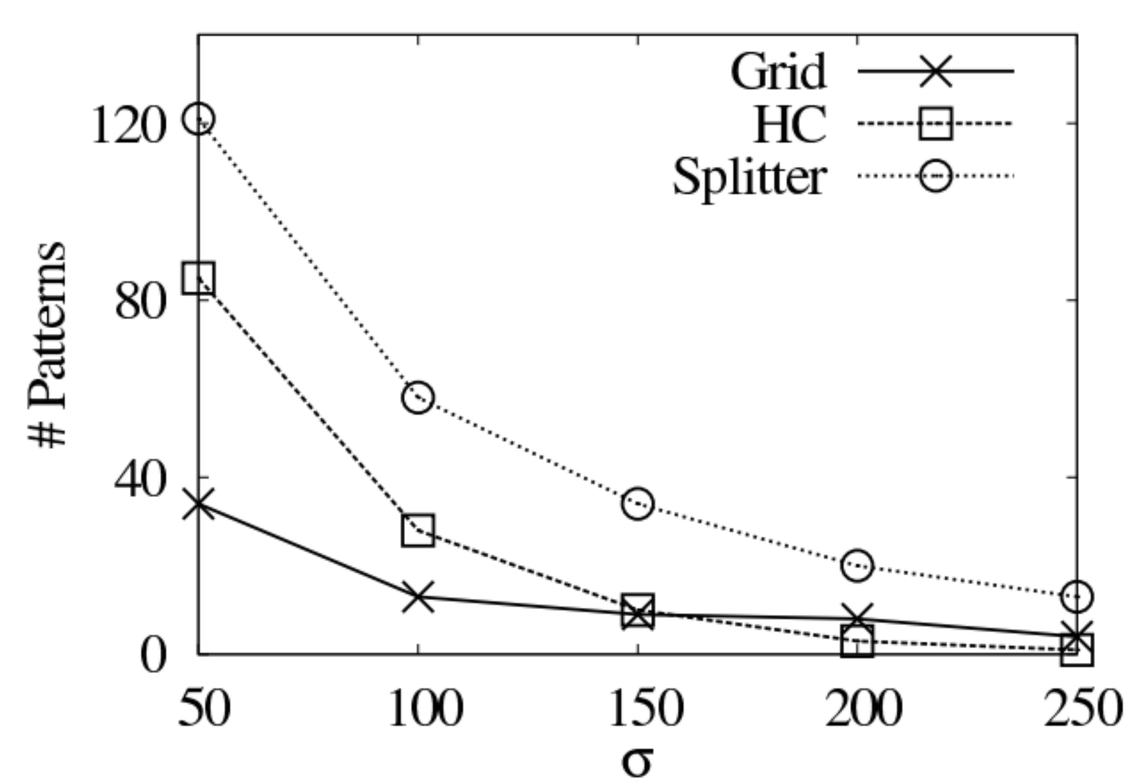


Effectiveness Comparison

- Varying the support threshold:



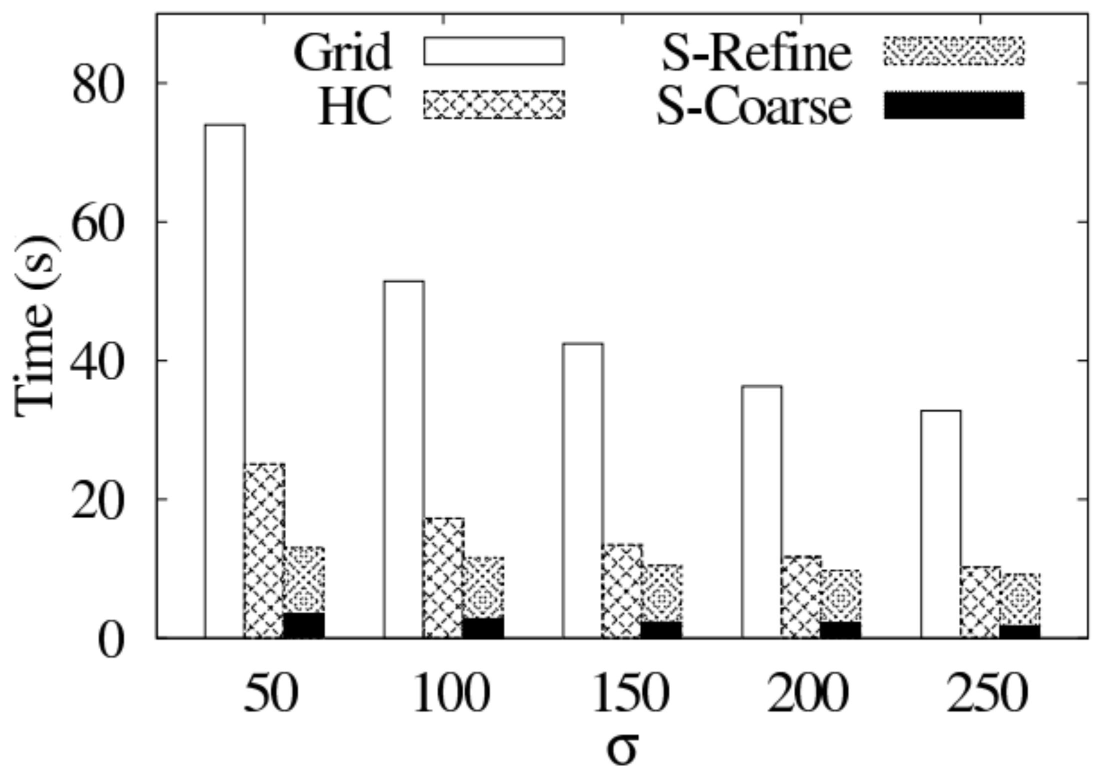
(a) Coverage *w.r.t.* σ.



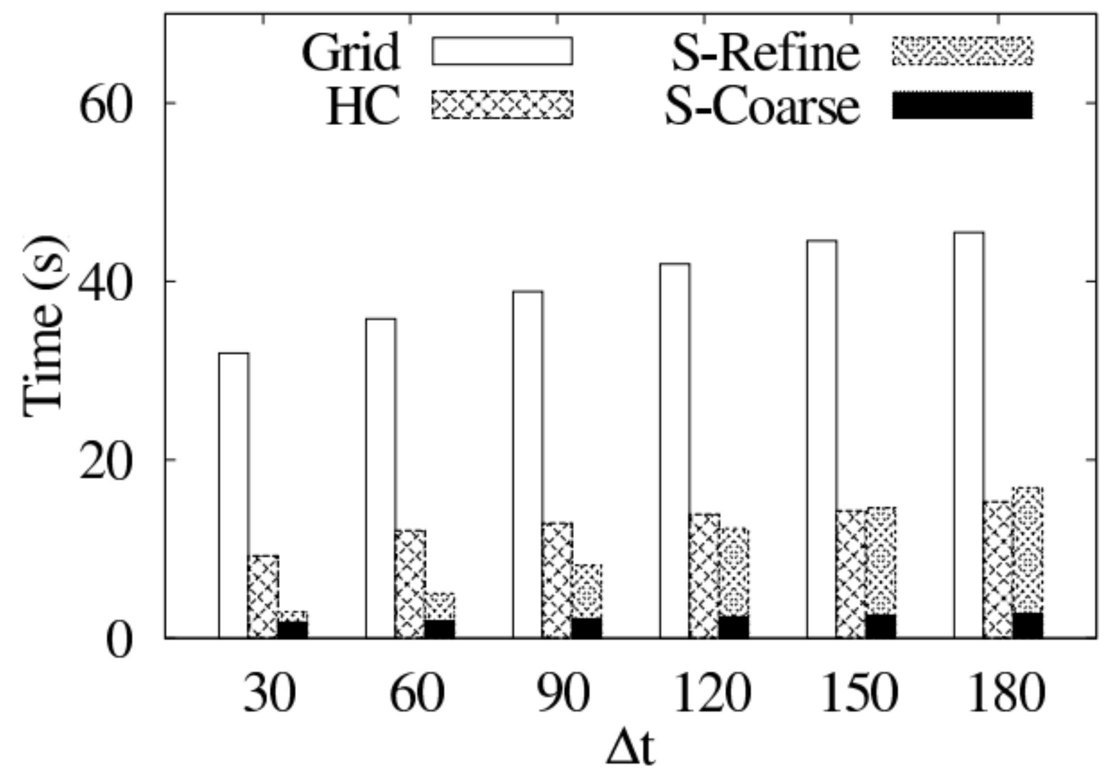
(b) Pattern number *w.r.t.* σ.

Efficiency Comparison

- Varying the support threshold and the time interval:



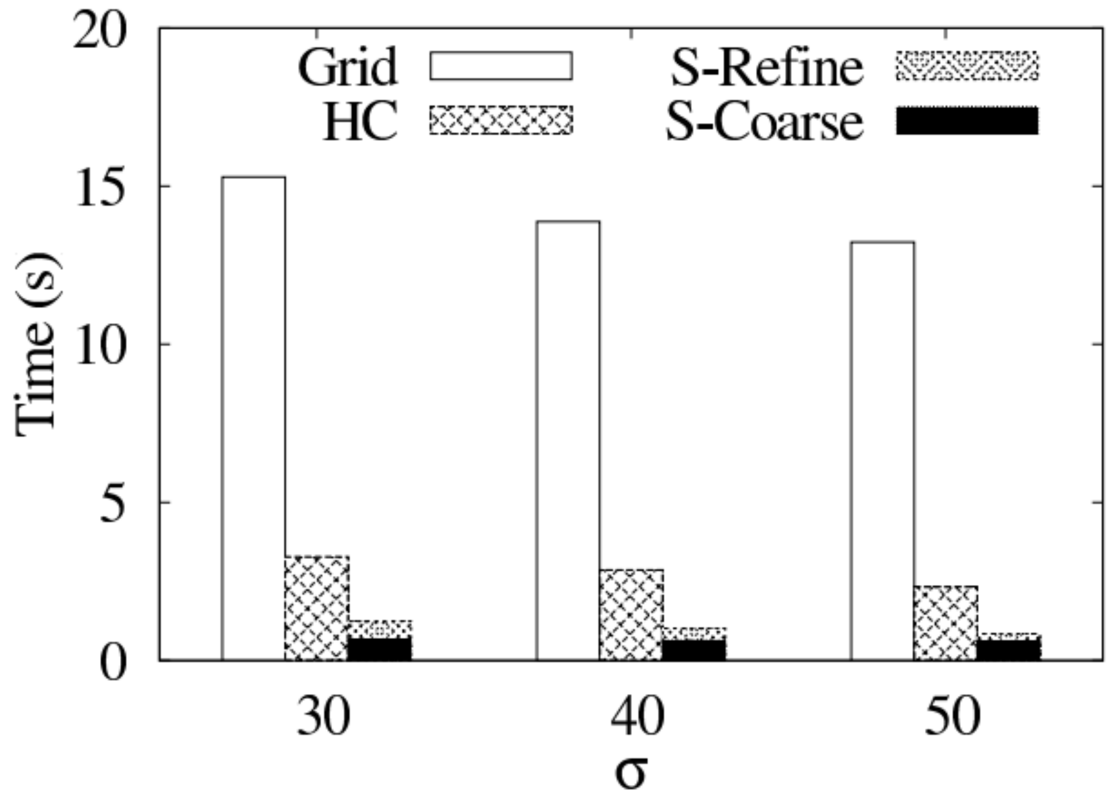
(a) Running time *w.r.t.* σ.



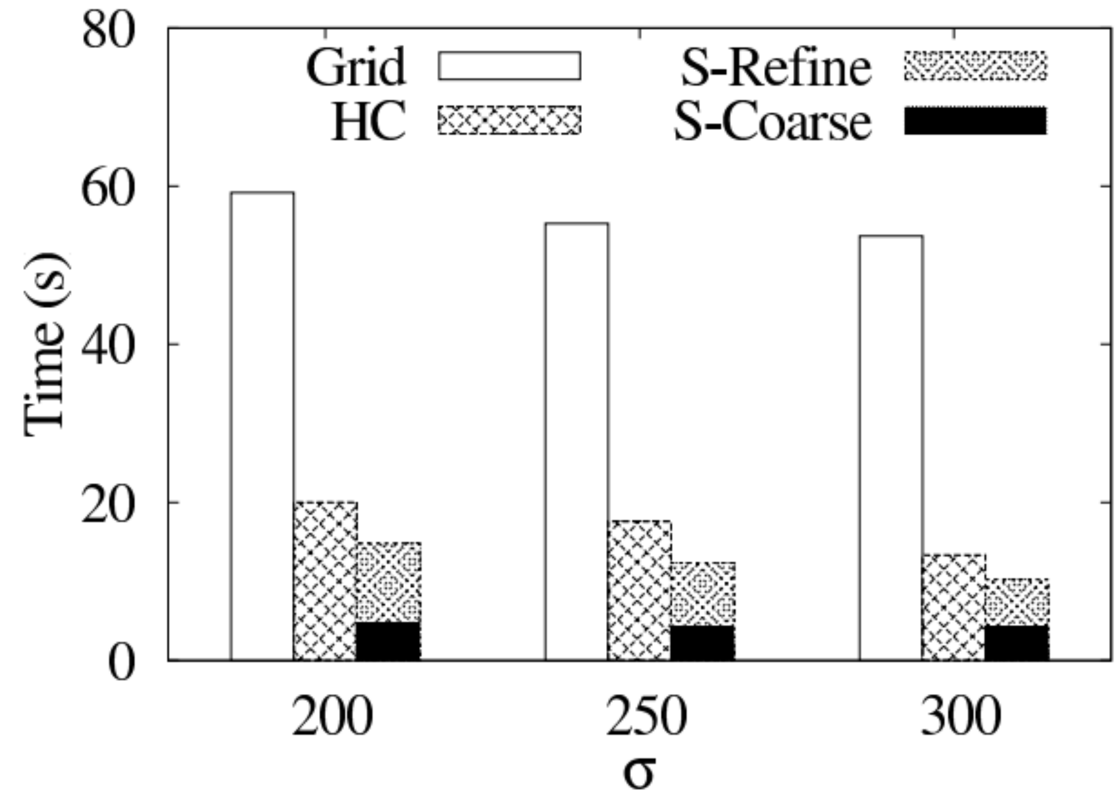
(b) Running time *w.r.t.* Δt.

Efficiency Comparison

- Efficiency on the synthetic data sets:



(a) Running time on S1K.



(b) Running time on S10K.

Summary

- Finding fine-grained sequential movement patterns is a critical yet challenging task.
- We develop a two-step method for mining fine-grained sequential movement patterns in semantic trajectories.
  - Step 1: mining coarse patterns
  - Step 2: splitting each coarse pattern into fine-grained ones.
- Our method significantly outperforms existing ones in both effectiveness and efficiency.