Spatio-Textual Similarity Joins

Panagiotis Bouros[1,2], Shen Ge[1], Nikos Mamoulis[1]

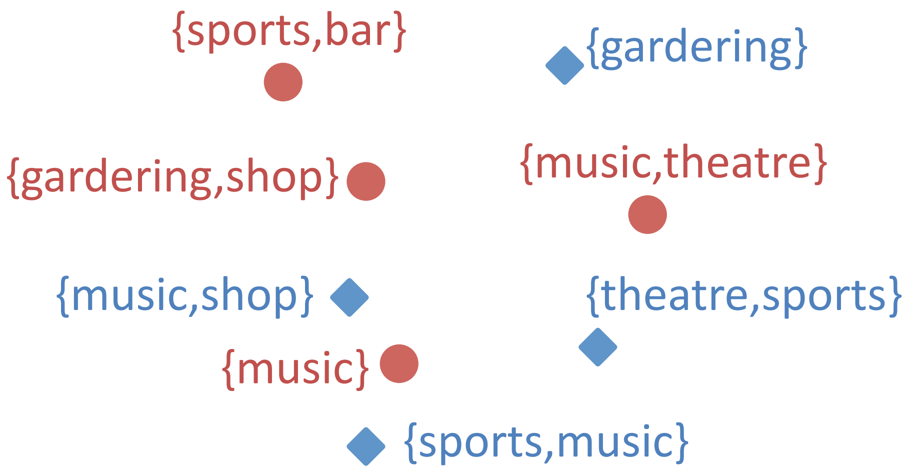
[1] University of Hong Kong
[2] Humboldt-Universität zu Berlin

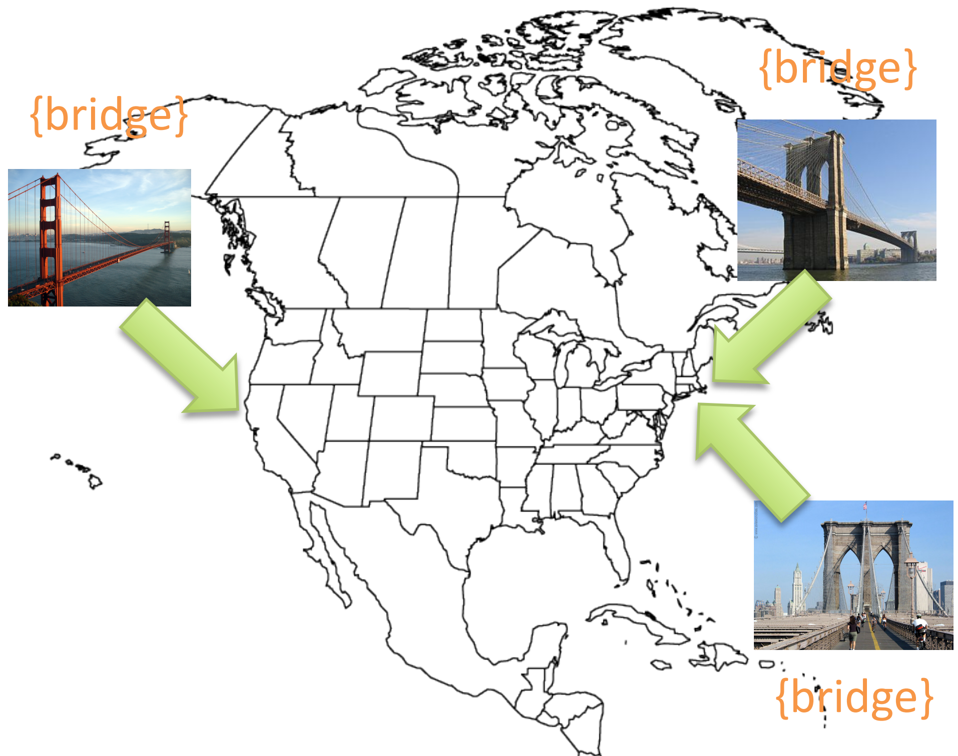
39th International Conference on Very Large Data Bases August 29, 2013

Complex data

- Data are becoming more complex
  - FLICKR, Foursquare, Twitter, Facebook…
    - Spatial locations
    - Textual description
    - Timestamps
    - Connectivity information (social)
  - Emerging geo-scientific fields, oceanography, seismology
    - Numerical attributes (measurements)
- Challenges for new complex queries
  - Research and industry, space as another dimension for set-value data

Motivation examples





- Social recommendation
  - Match men and women
  - Spatial locations
  - Interests

- Data de-duplication
  - Find FLICKR duplicates
  - Spatial locations
  - Tags description

Problem definition

- Spatio-textual objects $o(id, loc, text)$
- ST-SJOIN(R,S,ε,θ)
  - Pair of objects close in space with similar textual description
  - Euclidean spatial distance

$$dist_l(r,s) = dist(r.loc, s.loc)$$

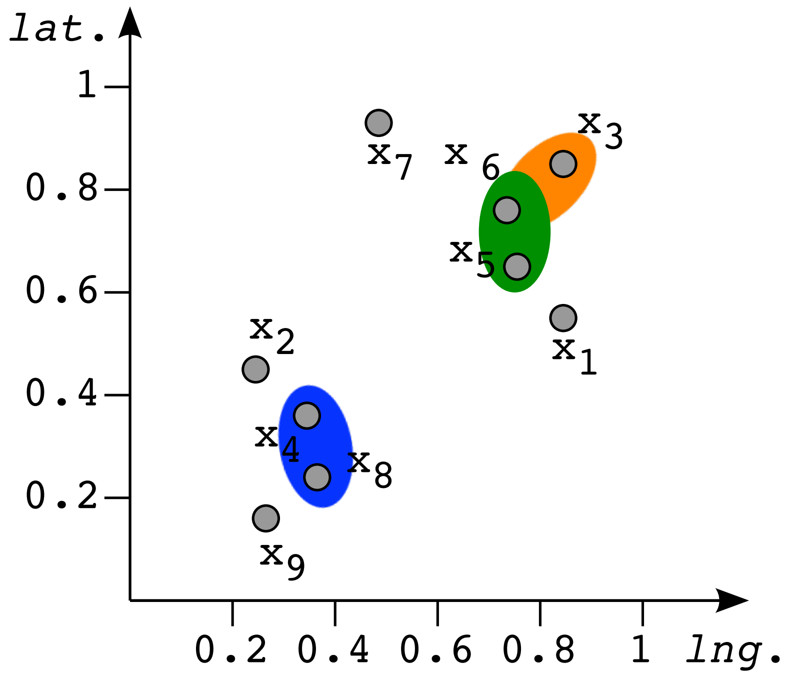
  - Jaccard textual similarity

$$sim_t(r,s) = \frac{|r.text \bigcap s.text|}{|r.text \bigcup s.text|}$$

  - Subset of R x S with $dist_l(r,s) \leq \varepsilon$ and $sim_t(r,s) \geq \theta$

Problem definition (cont'd)

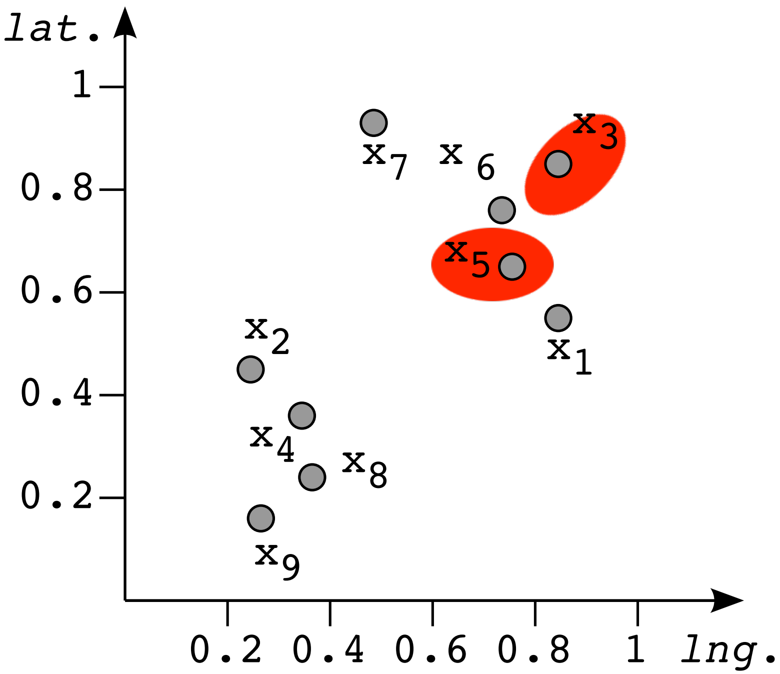
ST-SJOIN(R, R, ε = 0.2, θ = 0.7)



x_1	{B,C}	x_6	{C,D,E,F}
x_2	{E,F}	x_7	{A,B,C,D,F}
x_3	{D,E,F}	x_8	{A,B,D,E,F}
x_4	{A,B,E,F}	x_9	{A,B,C,D,E}
x_5	{C,D,E,F}		

Problem definition (cont'd)

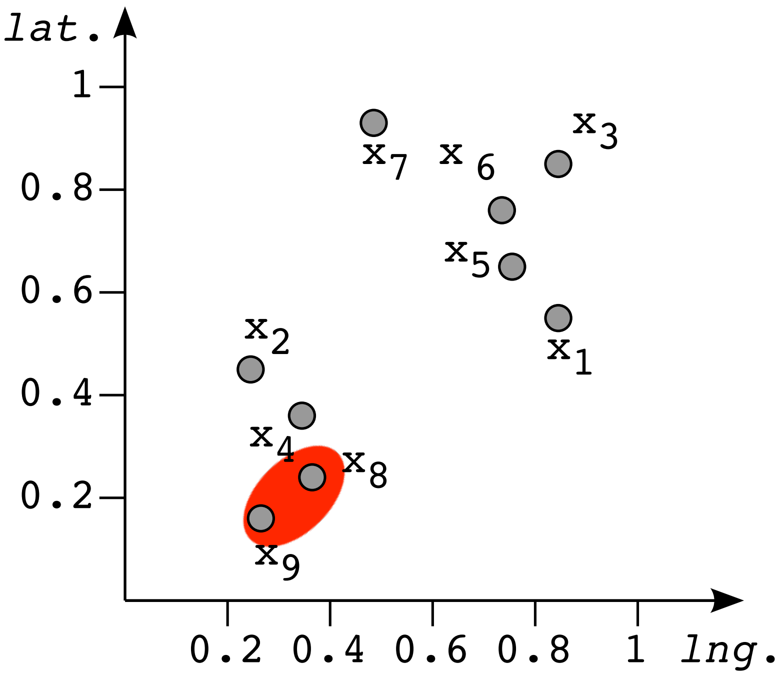
ST-SJOIN(R, R, ε = 0.2, θ = 0.7)



x_1	{B,C}	x_6	{C,D,E,F}
x_2	{E,F}	x_7	{A,B,C,D,F}
x_3	{D,E,F}	x_8	{A,B,D,E,F}
x_4	{A,B,E,F}	x_9	{A,B,C,D,E}
x_5	{C,D,E,F}		

Problem definition (cont'd)

ST-SJOIN(R, R, ε = 0.2, θ = 0.7)



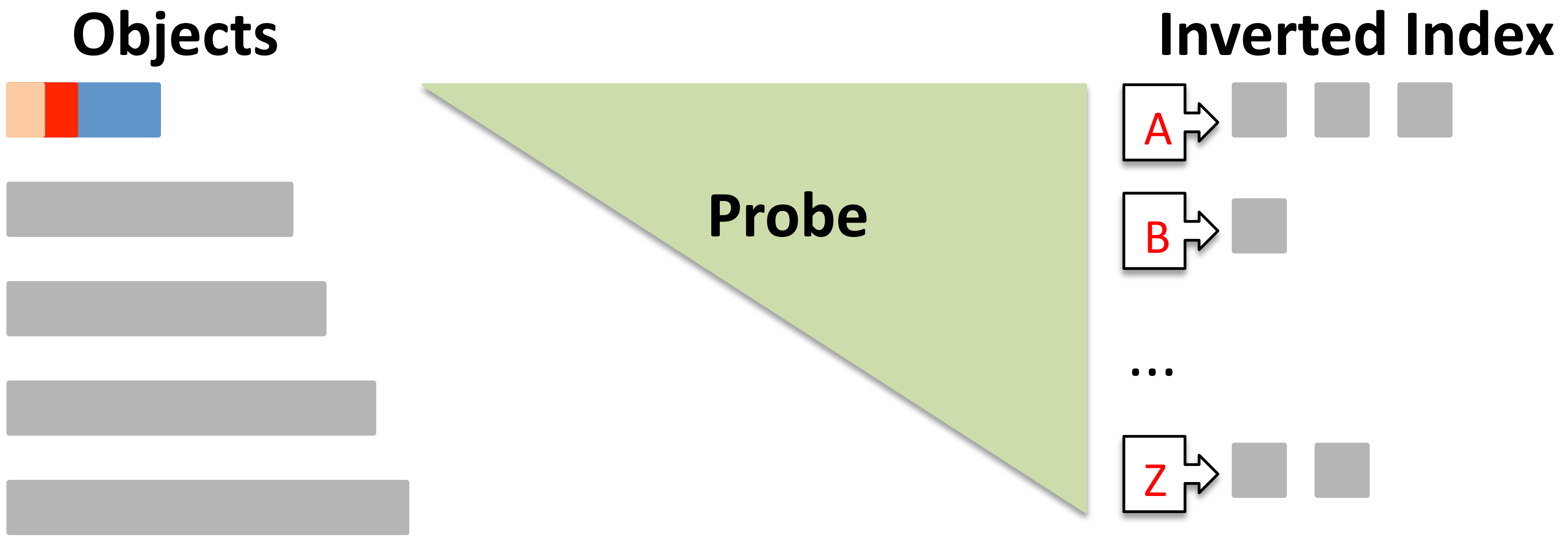
x_1	{B,C}	x_6	{C,D,E,F}
x_2	{E,F}	x_7	{A,B,C,D,F}
x_3	{D,E,F}	x_8	{A,B,D,E,F}
x_4	{A,B,E,F}	x_9	{A,B,C,D,E}
x_5	{C,D,E,F}		

Outline

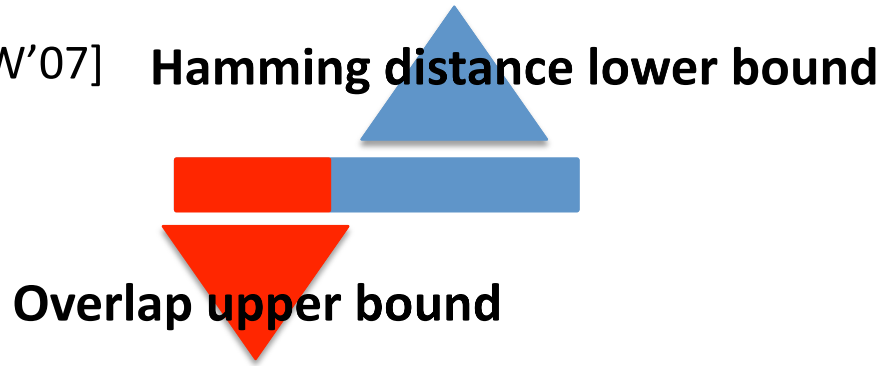
- Introduction
- Background on set similarity joins
- Computing spatio-textual similarity joins
- Experimental analysis
- Conclusions and future work

Set similarity joins and PPJOIN

[Xiao et al @ WWW'08]



- Inverted index to compute overlaps [Sarawagi et al @ SIGMOD'04]
- Prefix filtering [Chaudhuri et al @ ICDE'06]
- Two-phase method [Bayardo et al @ WWW'07]
  - Objects by length
  - Read-Probe-Index
- Positional filter
- Suffix filter



Computing ST-SJOIN

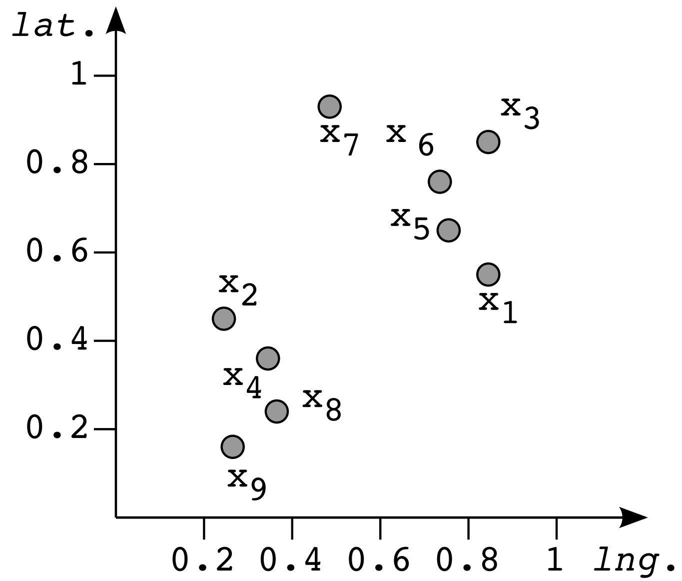
- Textual similarity join
  - Build upon PPJOIN
- Spatial distance join
  - Filtering, dynamic grid partitioning, R-tree
- Methods
  - PPJ
  - PPJ-I
  - PPJ-C
  - PPJ-R
- Grouping

Spatial filtering and PPJ

- Straightforward approach
  - Extend PPJOIN
  - Add another filter before positional and suffix

    $dist_l(r,s) \le \varepsilon$

ST-SJOIN(R, R, ε = 0.2, θ = 0.7)



x_1	{B,C}	x_6	{C,D,E,F}
x_2	{E,F}	x_7	{A,B,C,D,F}
x_3	{D,E,F}	x_8	{A,B,D,E,F}
x_4	{A,B,E,F}	x_9	{A,B,C,D,E}
x_5	{C,D,E,F}		

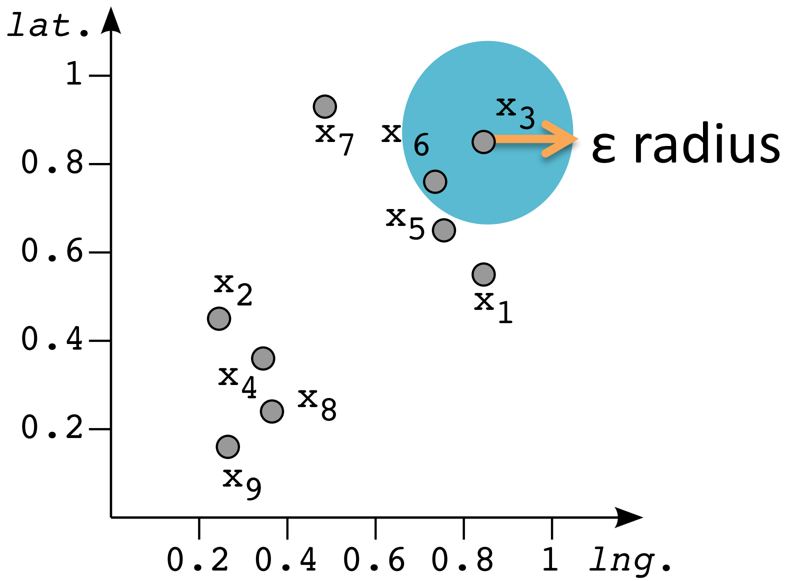
Spatial filtering and PPJ

- Straightforward approach
  - Extend PPJOIN
  - Add another filter before positional and suffix

$$dist_l(r,s) \leq \varepsilon$$

- Problem
  - Lack of spatial indexing
  - Examines objects no matter how far from x_3

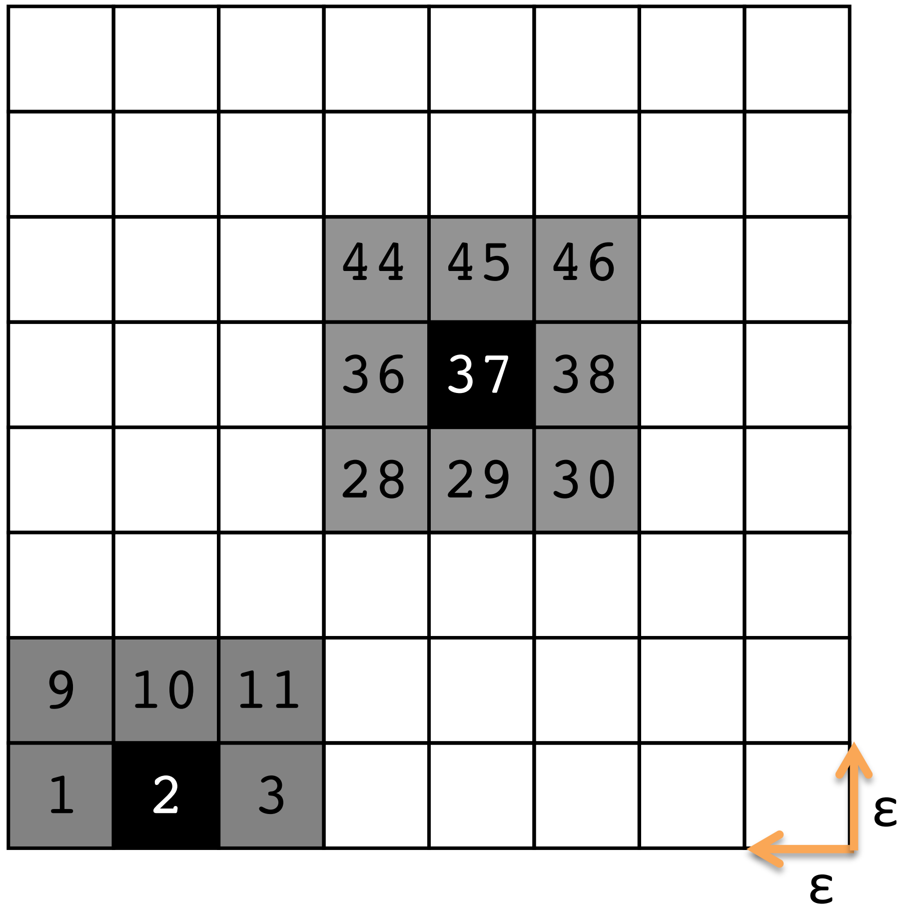
ST-SJOIN(R, R, ε = 0.2, θ = 0.7)



x_1	{B,C}	x_6	{C,D,E,F}
x_2	{E,F}	x_7	{A,B,C,D,F}
x_3	{D,E,F}	x_8	{A,B,D,E,F}
x_4	{A,B,E,F}	x_9	{A,B,C,D,E}
x_5	{C,D,E,F}		

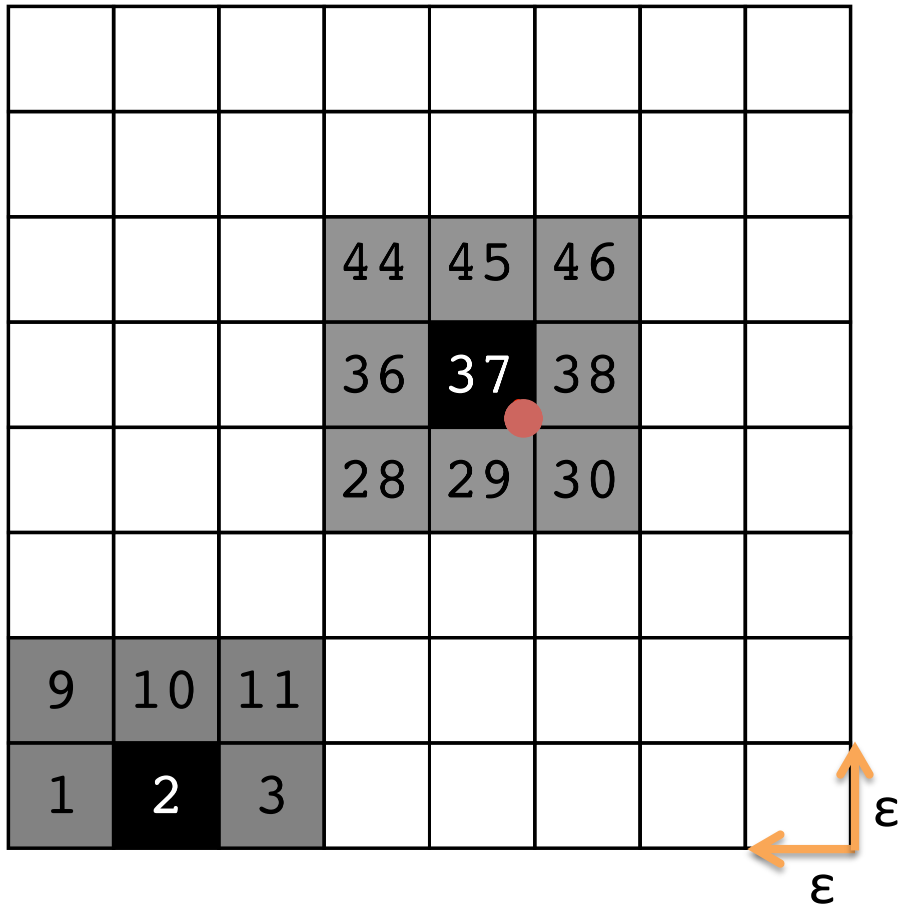
Dynamic grid partitioning

- Grid partitioning
  - On the fly
  - Extend of a grid cell equals ε
  - Numbering from left to right from bottom to top



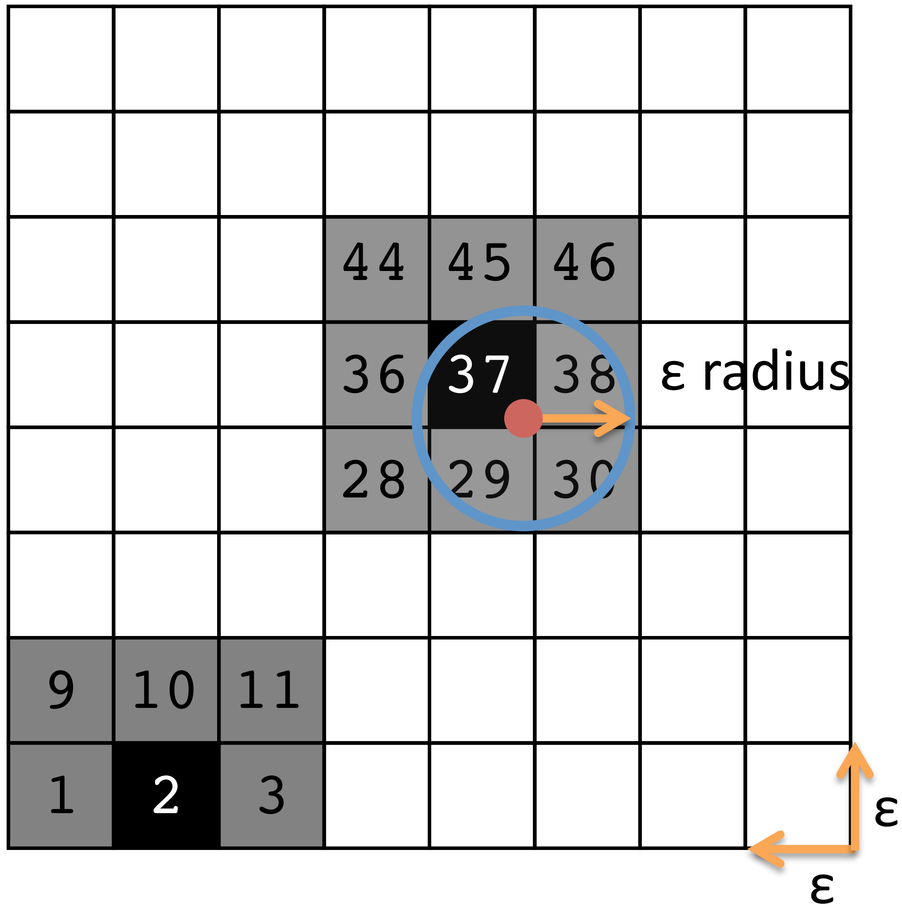
Dynamic grid partitioning

- Grid partitioning
  - On the fly
  - Extend of a grid cell equals ε
  - Numbering from left to right from bottom to top
- Property
  - Objects spatially joinable inside at most 9 cells
  - Still need to verify w.r.t. ε



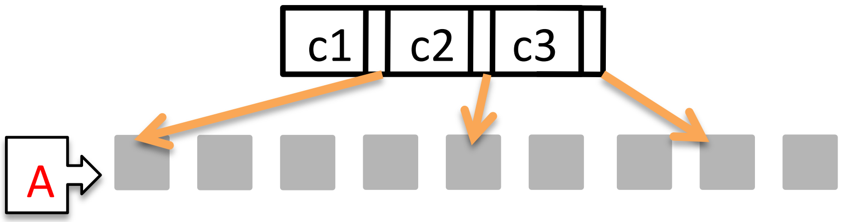
Dynamic grid partitioning

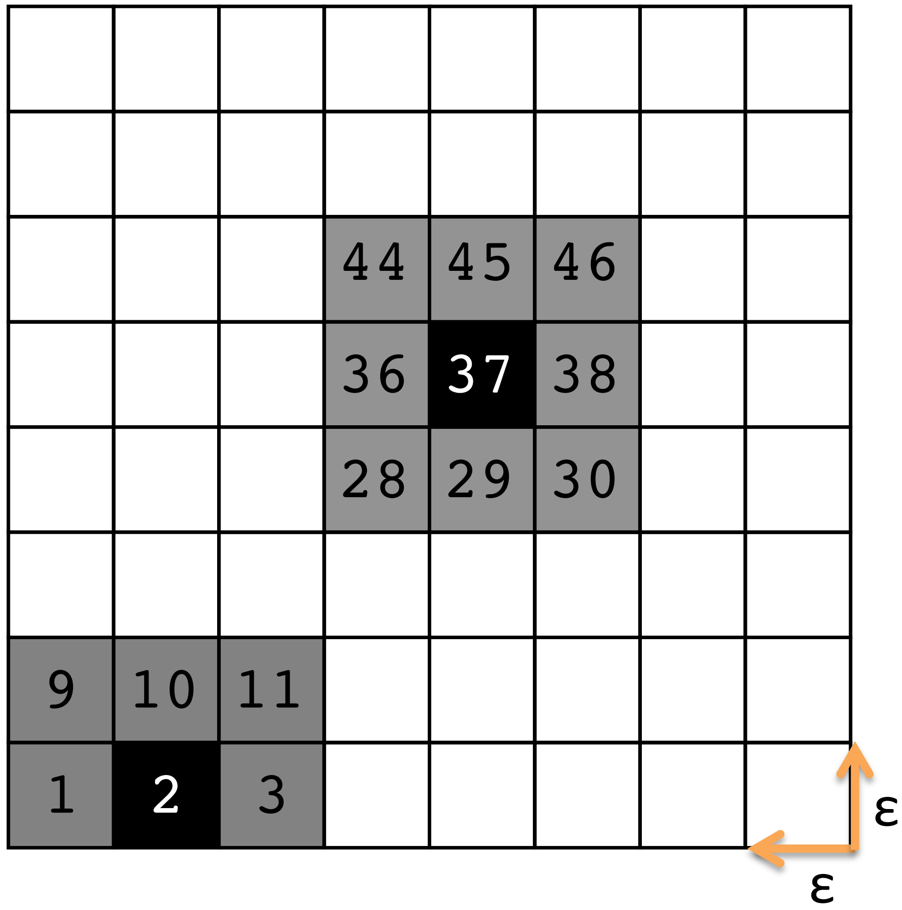
- Grid partitioning
  - On the fly
  - Extend of a grid cell equals ε
  - Numbering from left to right from bottom to top
- Property
  - Objects spatially joinable inside at most 9 cells
  - Still need to verify w.r.t. ε



Dynamic grid partitioning and PPJ-I

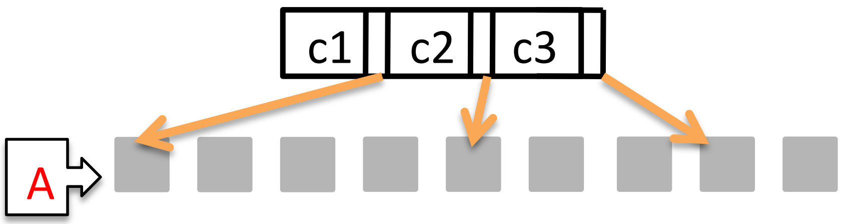
- Spatial information inside inverted index
  - Sort postings by cell id
  - Lightweight index on top of postings



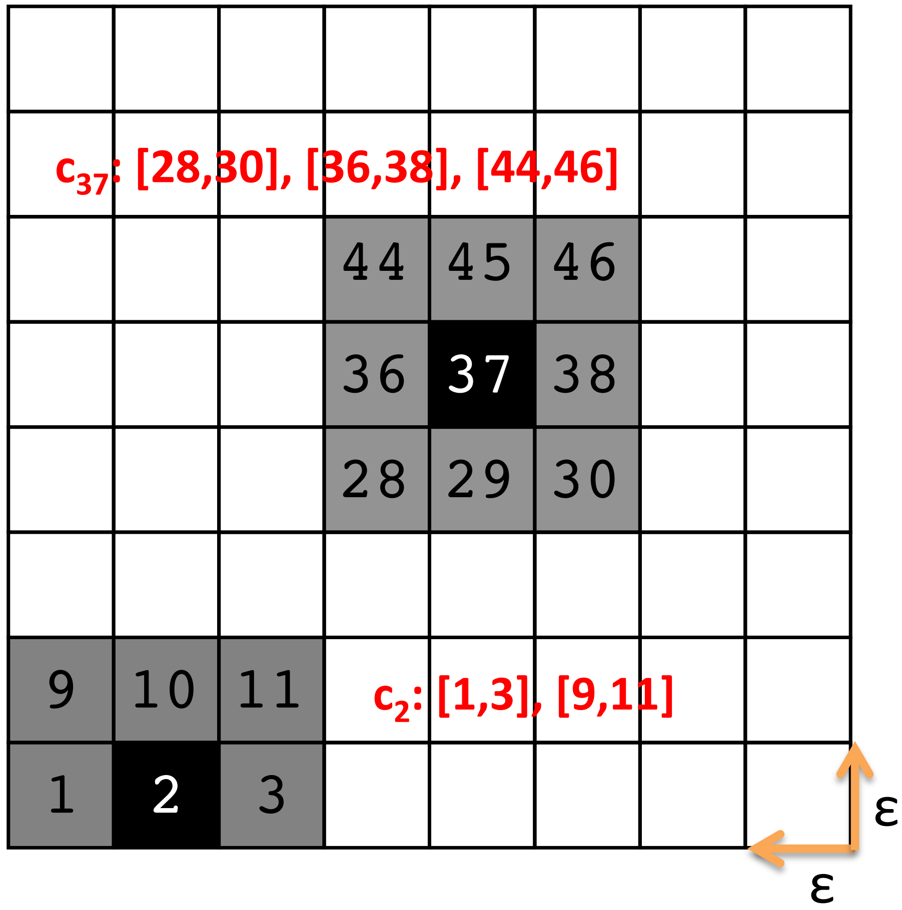


Dynamic grid partitioning and PPJ-I

- Spatial information inside inverted index
  - Sort postings by cell id
  - Lightweight index on top of postings

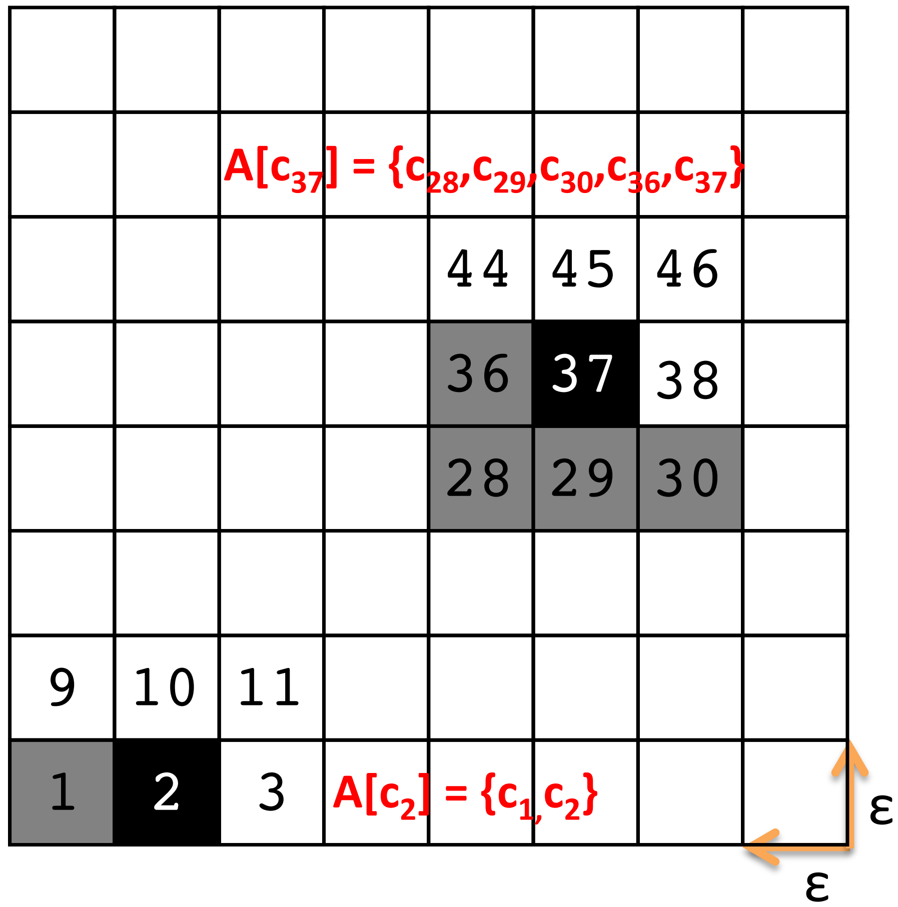


- Joinable neighborhood
  - At most three cell intervals
- Spatial distance join with space filling curve



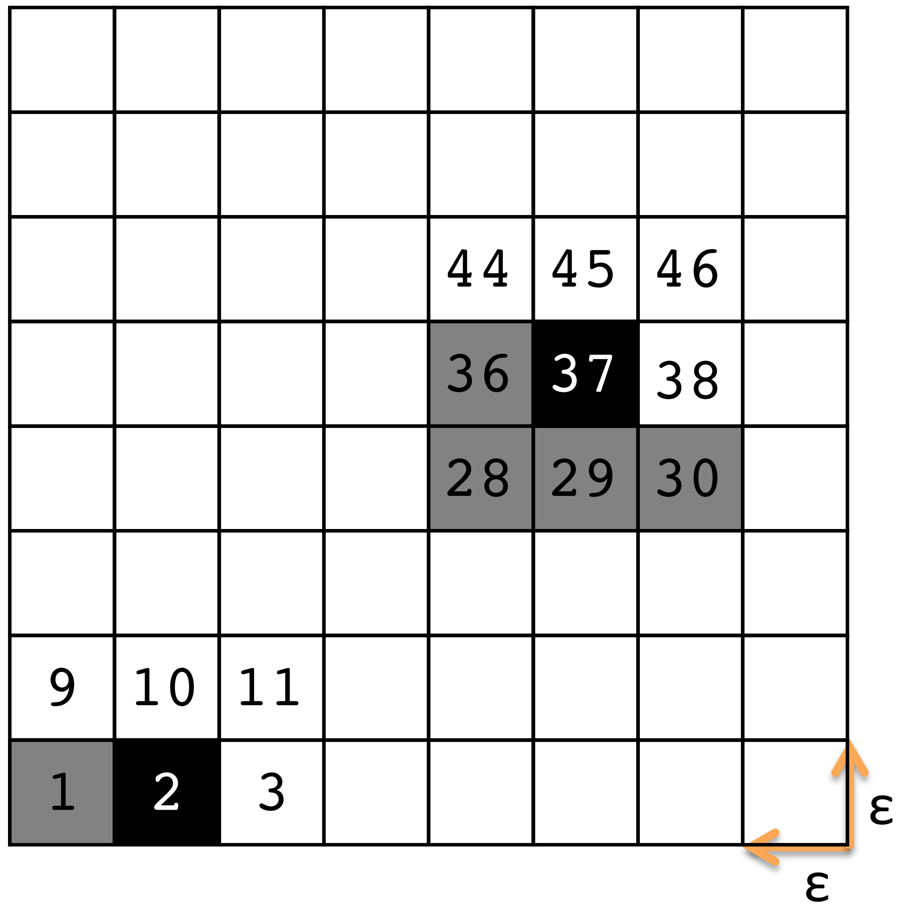
Dynamic grid partitioning and PPJ-C

- Working at the cell-level
- For each cell
  - Build inverted index
  - Define set A[c], cells among 9 adjacent with smaller or equal id



Dynamic grid partitioning and PPJ-C

- Working at the cell-level
- For each cell
  - Build inverted index
  - Define set A[c], cells among 9 adjacent with smaller or equal id
  - ST-SJOIN(c, c, ε, θ)
  - ST-SJOIN(c, c’, ε, θ) for each cell c’ in A[c]
  - Discard c after finish with all cell in A[c]



R-tree and PPJ-R

- Similar to PPJ-C but:
  - Static partitioning, objects indexed offline by R-tree
  - No connection to ε
- ST-SJOIN based on ε-distance join using R-trees
  - Traversing R-tree determines which partitions to join

Grouping

object	x.text	ppref(x)
x_1	{B,C}	{B}
x_2	{E,F}	{E}
x_3	{D,E,F}	{D}
x_4	{A,B,E,F}	{A,B}
x_5	{C,D,E,F}	{C,D}
x_6	{C,D,E,F}	{C,D}
x_7	{A,B,C,D,F}	{A,B}
x_8	{A,B,D,E,F}	{A,B}
x_9	{A,B,C,D,E}	{A,B}

- Problems
  - Same prefix index more than once

Grouping

object	x.text	ppref(x)
x_1	{B,C}	{B}
x_2	{E,F}	{E}
x_3	{D,E,F}	{D}
x_4	{A,B,E,F}	{A,B}
x_5	{C,D,E,F}	{C,D}
x_6	{C,D,E,F}	{C,D}
x_7	{A,B,C,D,F}	{A,B}
x_8	{A,B,D,E,F}	{A,B}
x_9	{A,B,C,D,E}	{A,B}

- Problems
  - Same prefix index more than once
  - Some overlap calculated more than once

Grouping

group	object	x.text	ppref(x)
g_1	x_1	{B,C}	{B}
g_2	x_2	{E,F}	{E}
g_3	x_3	{D,E,F}	{D}
g_4	x_5	{C,D,E,F}	{C,D}
	x_6	{C,D,E,F}	{C,D}
g_5	x_4	{A,B,E,F}	{A,B}
	x_7	{A,B,C,D,F}	{A,B}
	x_8	{A,B,D,E,F}	{A,B}
	x_9	{A,B,C,D,E}	{A,B}

- Problems
  - Same prefix index more than once
  - Some overlap calculated more than once
- Grouping objects by prefix
  - Massive pruning

Grouping for ST-SJOIN

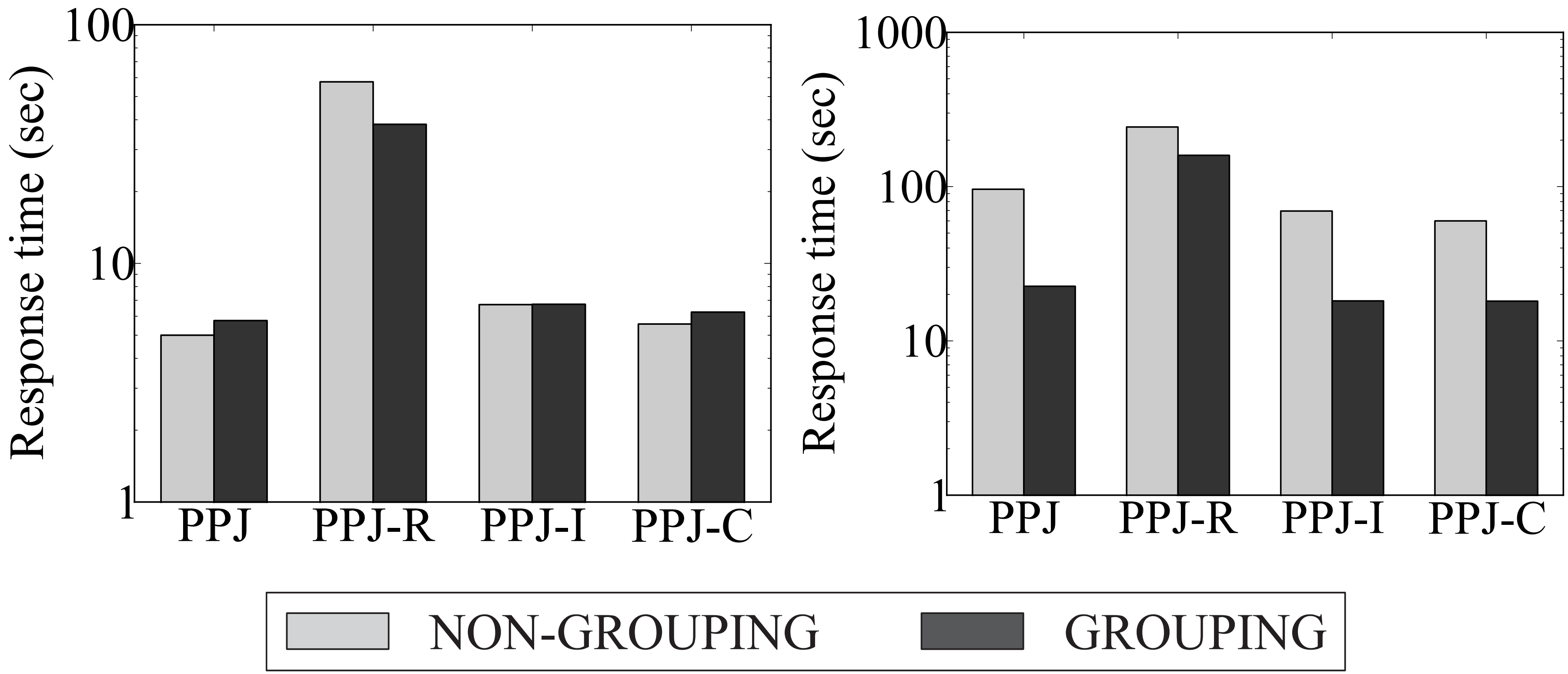
- Textually
  - Group objects by length and prefix
  - Examination order retained, PPJOIN fully applicable
  - If $|g_x| \geq |g_y|$ then $|x| \geq |y|$ for x in g_x and y in g_y
- Spatially
  - PPJ: group objects no matter how far
  - PPJ-I,PPJ-C: group objects inside grid cells
- Join process
  - Probing and indexing over groups
  - Suffix filter not useful
  - Unfold groups during verification

Experimental analysis

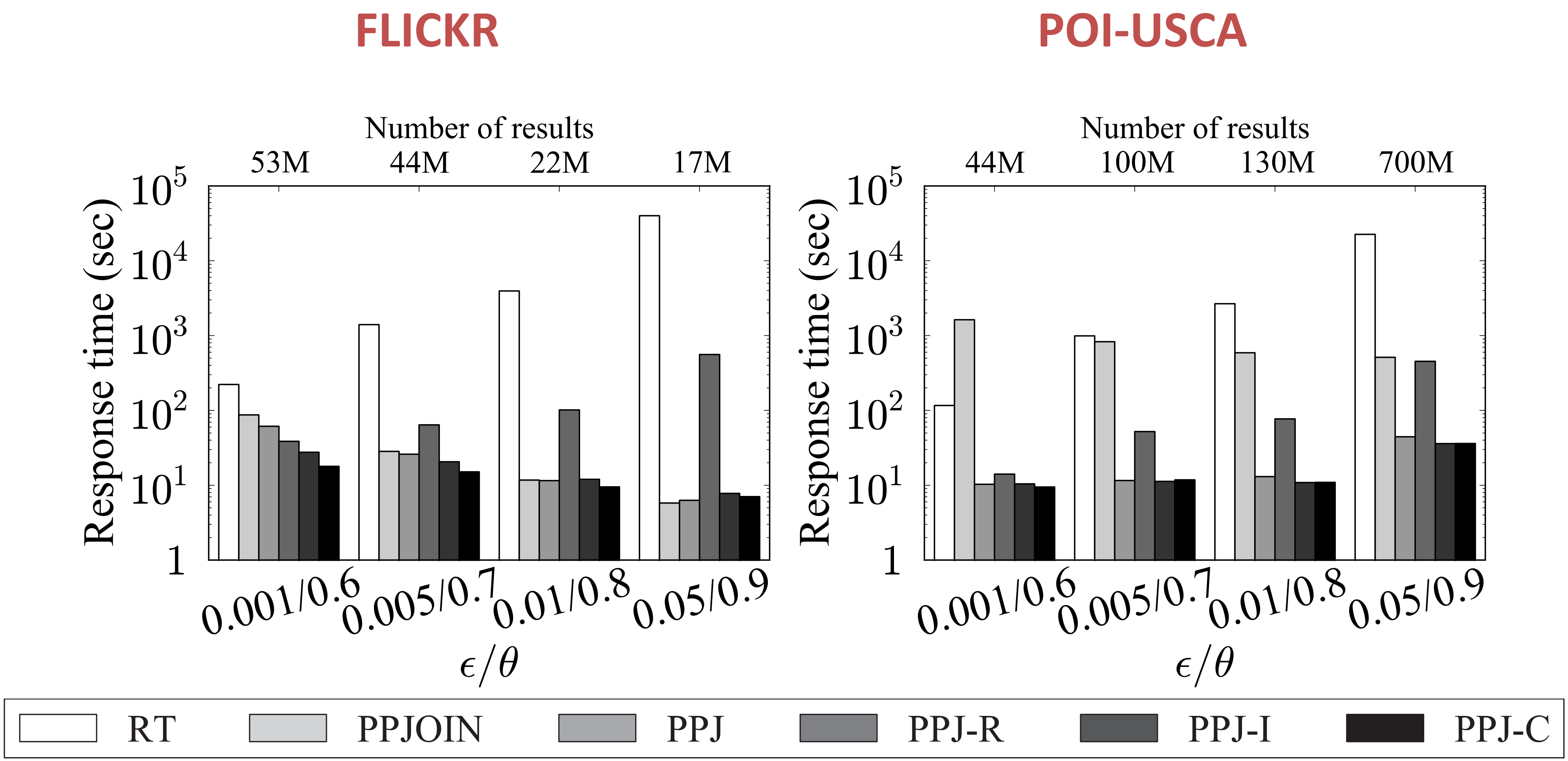
- Real collections
  - FLICKR, NY, |R| = 1.5M, |T| = 730K, avg size 10.5
  - POI-USCA, California state, |R| = 1.5M, |T| = 16K, avg size 4.4
  - POI-AU, Australia, |R| = 700K, |T| = 2.6K, avg size 4.7
- Synthetic collections
  - |R| = {30K, 100K, 500K, 1M, 3M}
  - |T| = {5K, 10K, 50K, 100K, 300K}
  - Spatial distribution, uniform or clustered
  - Correlated
- Experiments
  - Measure response time
  - Vary ε = {0.001,0.005,0.01,0.05,0.1} synthetic {0.001,0.005,0.01,0.05} real
  - Vary θ = {0.5,0.6,0.7,0.8,0.9} synthetic, {0.6,0.7,0.8,0.9} real

To group or not to group

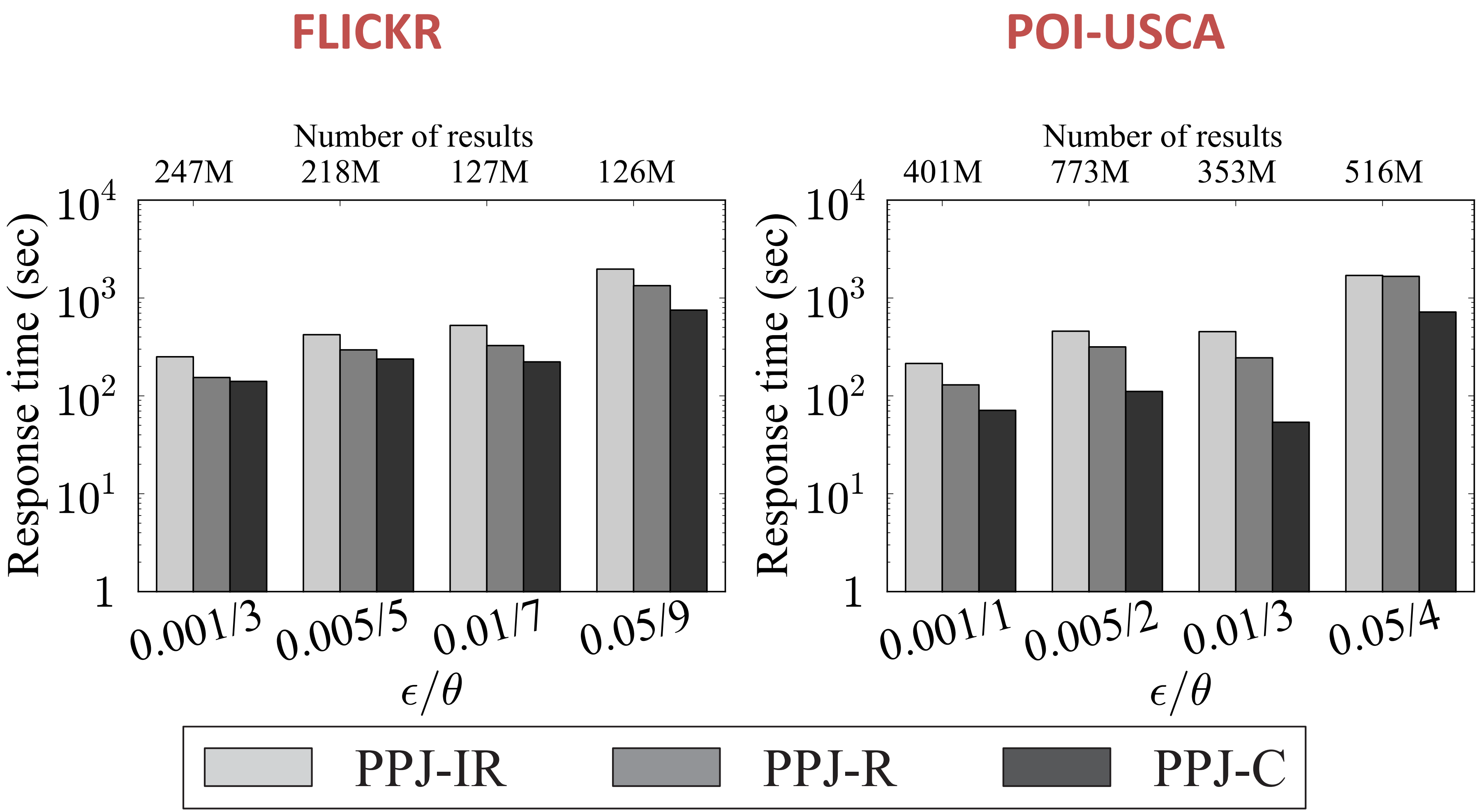
FLICKR ε = 0.005, θ = 0.9 | **POI-AU ε = 0.05, θ = 0.7**



Comparison with baseline methods



IR-tree and PPJ-IR



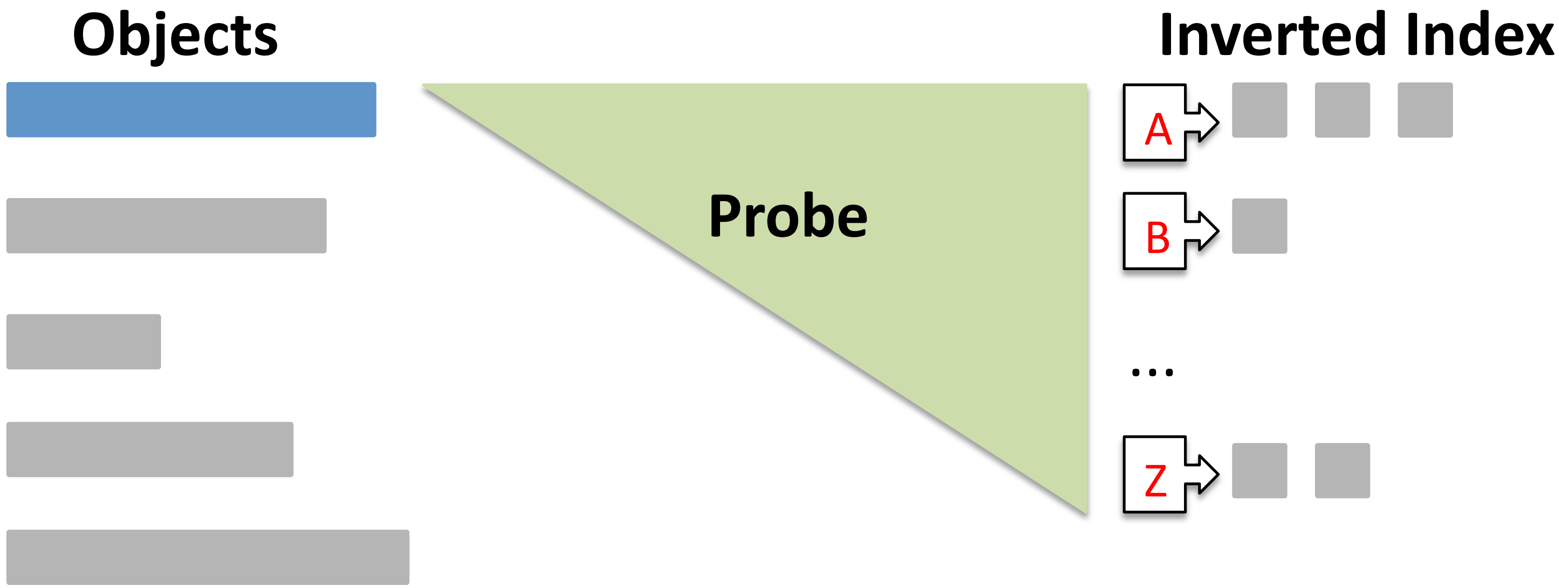
Conclusions and future work

- Conclusions
  - New join query, ST-SJOIN
  - Evaluation algorithms
    - State-of-the-art on set similarity joins
    - Spatial indexing
    - PPJ-C in general most efficient method
- Future work
  - Study PPJ-C's advantage on distributed environments
  - Consider other dimensions, e.g., time or graph

Questions?

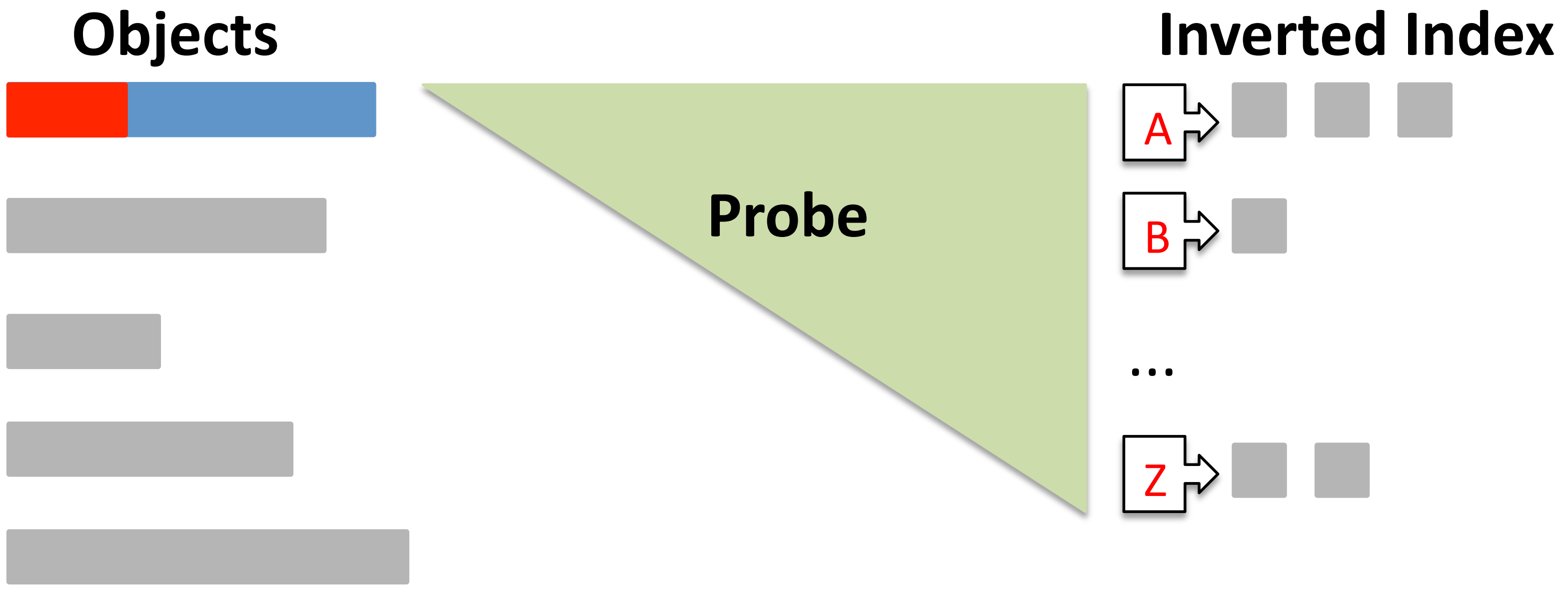
Backup slides

Set similarity joins



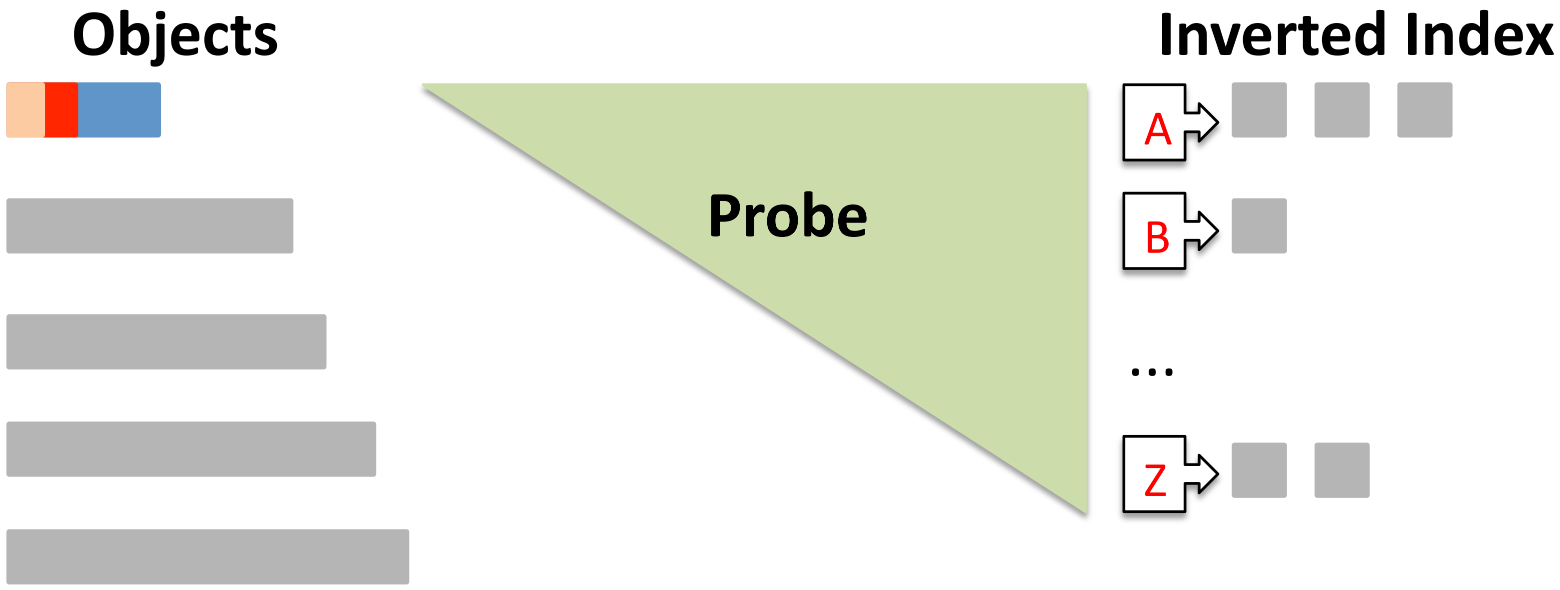
- For every term t in object [Sarawagi et al @ SIGMOD'04]
  - Probe inverted index, traverse postings list L_t
  - Compute overlap O[,] with every object
- Optimization
  - Build inverted index on the fly, incrementally
  - Compute overlap between two object only once

Set similarity joins



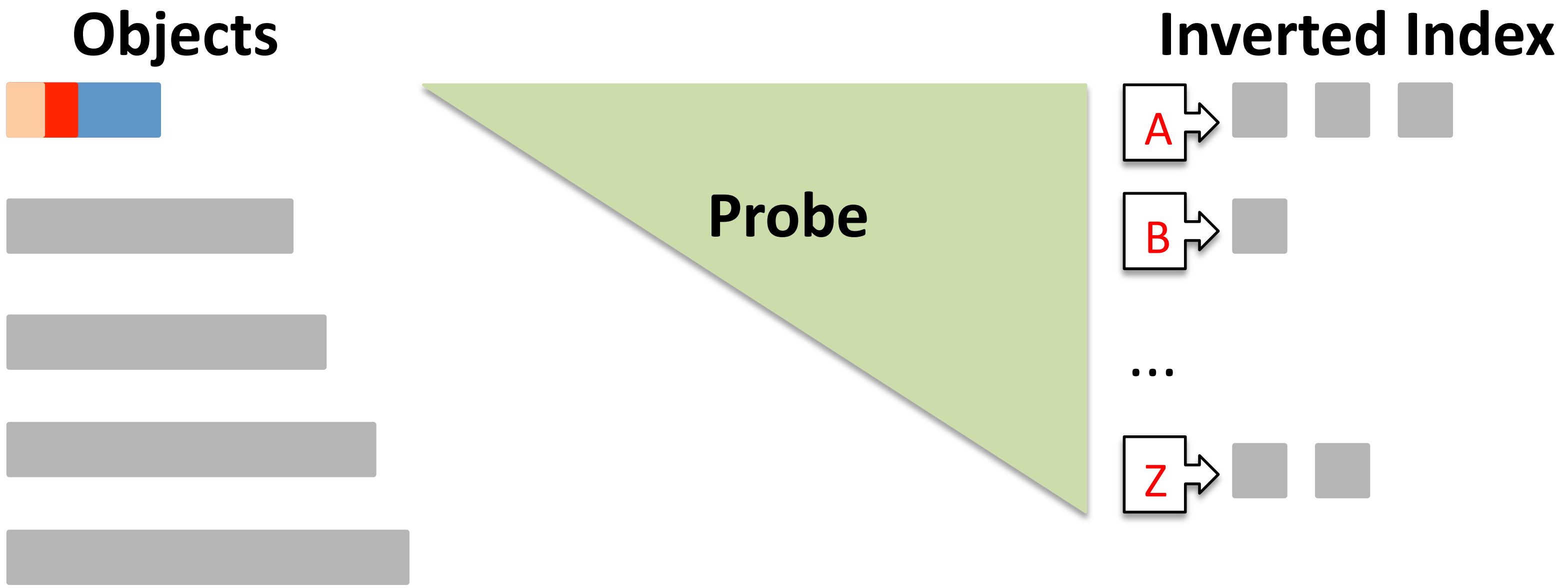
- Prefix filtering [Chaudhuri et al @ ICDE'06]
  - Global ordering of terms, canonicalized objects
  - Prefixes w.r.t. θ should share at least one term

Set similarity joins

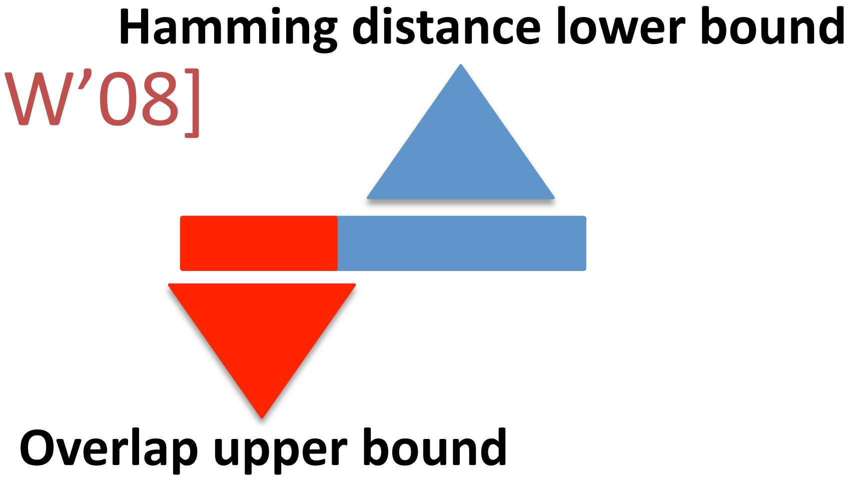


- AllPairs [Bayardo et al @ WWW'07]
  - Builds upon prefix-filtering
  - Examine objects by length, ascending
  - Reduce indexing cost
    - Index prefix of an object
  - Length filter

Set similarity joins

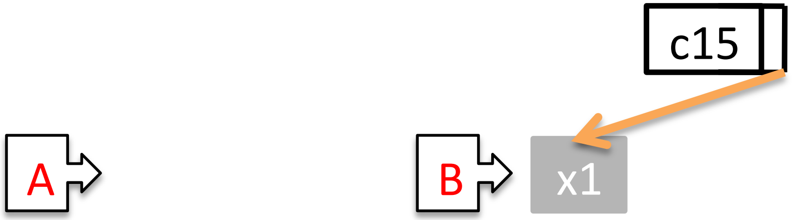


- PPJOIN [Xiao et al @ WWW'08]
  - Builds upon AllPairs
  - Positional filter
  - Suffix filter



Dynamic grid partitioning and PPJ-I

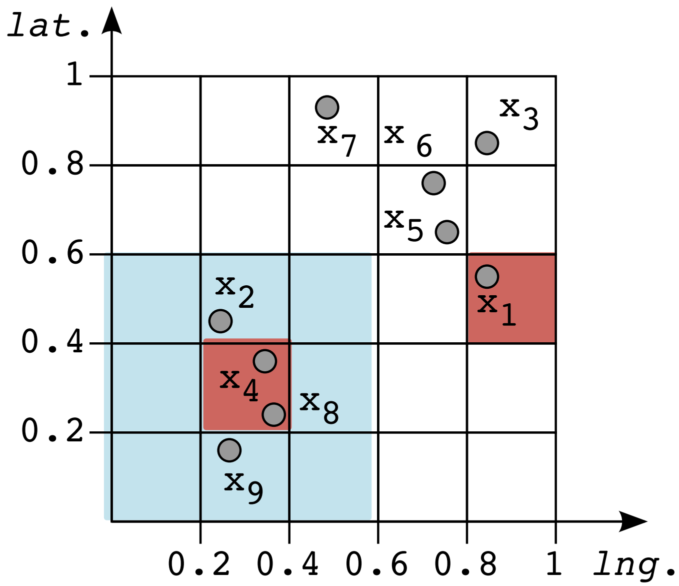
- When examining x_4 in c_7



c_7: [1,3], [6,8], [11,13]

  – c_{15} is not inside the joinable neighborhood of c_7

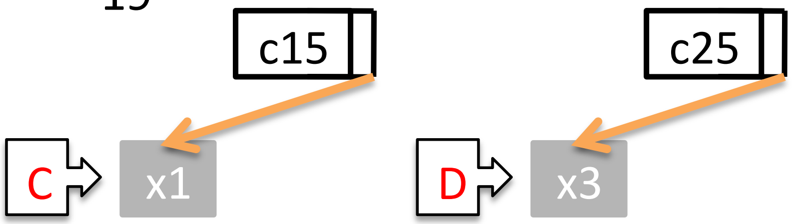
ST-SJOIN(R, R, ε = 0.2, θ = 0.7)



x_1	{B,C}	x_6	{C,D,E,F}
x_2	{E,F}	x_7	{A,B,C,D,F}
x_3	{D,E,F}	x_8	{A,B,D,E,F}
x_4	{A,B,E,F}	x_9	{A,B,C,D,E}
x_5	{C,D,E,F}		

Dynamic grid partitioning and PPJ-I

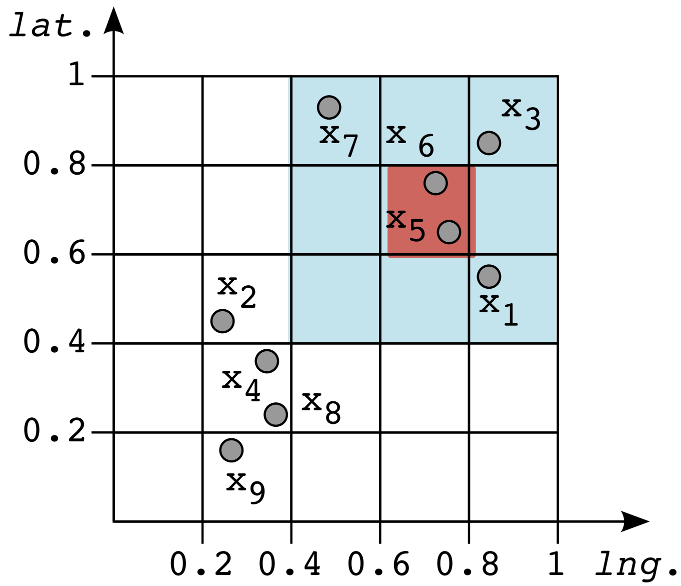
- When examining x_5 in c_{19}



c_{19}: [13,15], [18,20], [23,25]

  - c_{25} is inside the joinable neighborhood of c_{19}
  - Need to check Euclidean distance

ST-SJOIN(R, R, ε = 0.2, θ = 0.7)



x_1	{B,C}	x_6	{C,D,E,F}
x_2	{E,F}	x_7	{A,B,C,D,F}
x_3	{D,E,F}	x_8	{A,B,D,E,F}
x_4	{A,B,E,F}	x_9	{A,B,C,D,E}
x_5	{C,D,E,F}		