

Discovering Spatial Co-location Patterns

Yan Huang
Department of Computer Science
University of Minnesota
huangyan@cs.umn.edu

Advisor: Dr. Shashi Shekhar
Co-advisor: Dr. Dingzhu Du

Biography

* Education

- * Ph.D. Candidate, C.S., UMN, Fall 98 - May 03 (expected)
- * B.S., C.S., Beijing University, Fall 93 - Fall 97

* Research Interests

- * Database, Spatial Database, Data Mining, Geographic Information Systems

* Publications

* Spatial Co-location Patterns

- S. Shekhar and Y. Huang, Discovering Spatial Co-location Patterns : A Summary of Results, In *Proc. of 7th Intl Symposium on Spatial and Temporal Databases (SSTD)*, Springer-Verlag, Lecture Notes in Computer Science, LNCS 2121, p.236 ff, July 2001
- S. Shekhar and Y. Huang, Multi-resolution Co-location Miner: a New Algorithm to Find Co-location Patterns from Spatial Datasets, *SIAM SDM02 Workshop on Mining Scientific Datasets*, April 2002
- Y. Huang, H. Xiong, S. Shekhar, and J. Pei, Mining Confident Co-location Rules without A Support Threshold, in *Proc. of 18th ACM Symposium on Applied Computing (ACM SAC)*, March 2003
- Y. Huang, S. Shekhar, and H. Xiong, Discovering Co-location Patterns from Spatial Datasets: A General Approach, submitted to *IEEE Transactions on Knowledge and Data Engineering (TKDE)*

Biography

* Vector Map Compression

- S. Shekhar, Y. Huang, J. Djugash, and C. Zhou, Vector Map Compression: A Clustering Approach, in *Proc. of 10th ACM Intl. Symposium. on Advances in Geographic Information Systems (ACM-GIS)*, November 2002
- S. Shekhar, Y. Huang, and J. Djugash, Dictionary Design Algorithms for Vector Map Compression, In *Proc. of IEEE Data Compression Conference (DCC)*, April 2002

* Spatial Time-series Correlation Join

- P. Zhang, Y. Huang, S. Shekhar, and V. Kumar, Efficient Algorithms for Correlation Join over Spatial Time-series Datasets, to appear in *Proc. of 7th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD)*, 2003

* Misc - Spatial Data Mining

- S. Shekhar, Y. Huang, W. Wu, C.T. Lu, and S. Chawla, What's Spatial about Spatial Data Mining: Three Case Studies, book chapter: *Data Mining for Scientific and Engineering Applications*, R. Grossman, C. Kamath, P. Kegelmeyer, V. Kumar, R. Namburu (eds.), ISBN1-4020-0033-2, Kluwer Academic Publishers, 2001

Overview

- ⇒ Introduction
- * Related Work
- * Event Centric Approach
- * Co-location Miner Algorithm
- * Evaluation
- * Conclusions and Future Work

Spatial data mining (SDM)

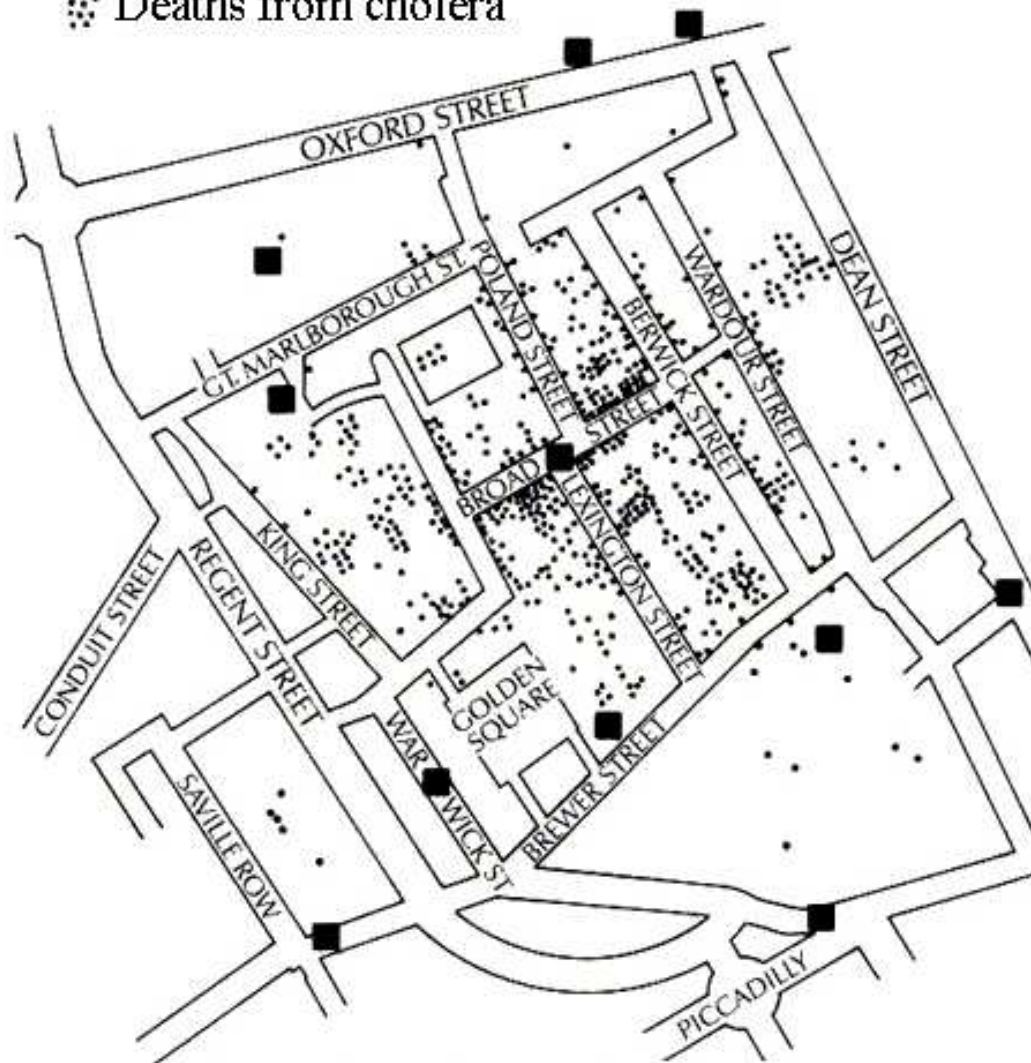
- * The process of discovering
 - * interesting, useful, non-trivial patterns
 - * from large spatial datasets
- * Spatial patterns
 - * Spatial outlier, discontinuities
 - bad traffic sensors on highways (DOT)
 - * Location prediction models
 - model to identify habitat of endangered species
 - * Spatial clusters
 - crime hot-spots (NIJ), cancer clusters (CDC)
 - * Co-location patterns
 - predator-prey species, symbiosis
 - Dental health and fluoride
 - Chromium 6 used by PG&E, health problems in Hinkley CA

Example Spatial Pattern: Spatial Cluster

★ 1854 cholera epidemic London map

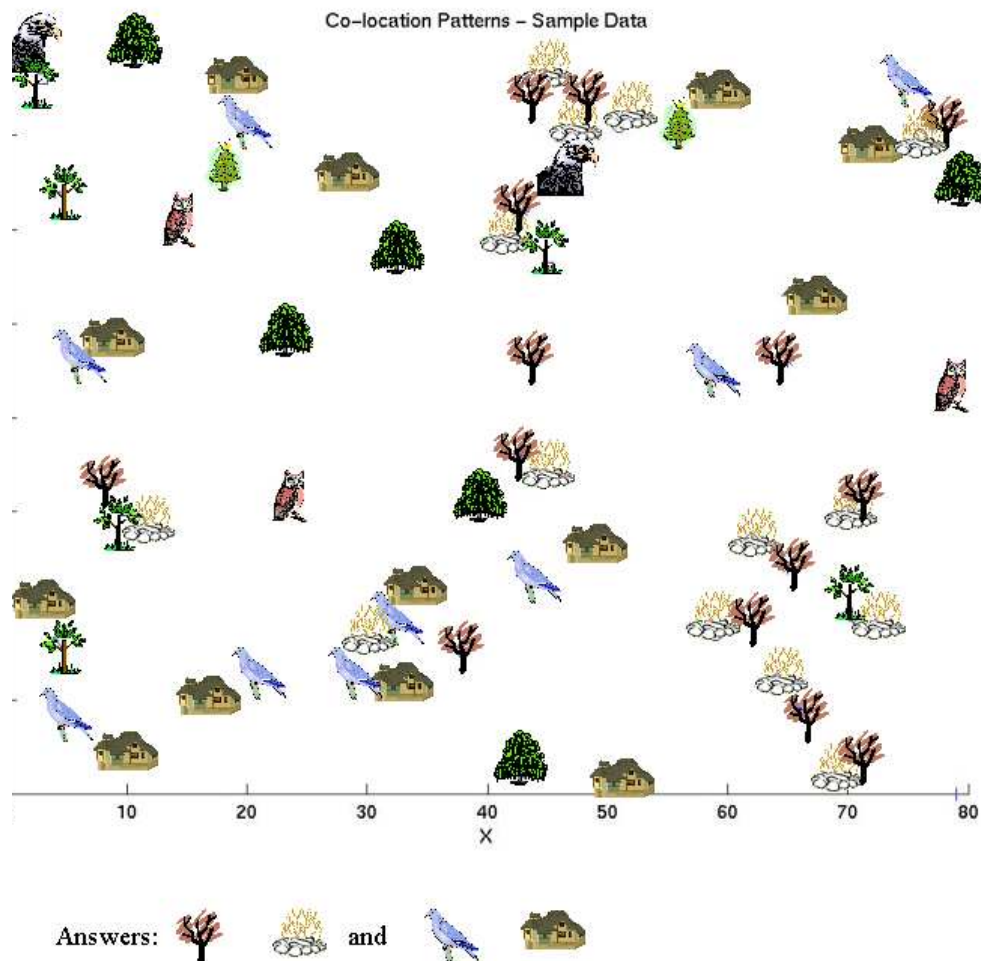
■ Pump sites

⋈ Deaths from cholera



Example Spatial Pattern: Co-locations

- ★ Given:
 - ★ A collection of different types of spatial events
- ★ Illustration



- ★ Find: Co-located subsets of event types

Overview

- * Introduction
- ⇒ Related Work
- * Event Centric Approach
- * Co-location Miner Algorithm
- * Evaluation
- * Conclusions and Future Work

Related Work

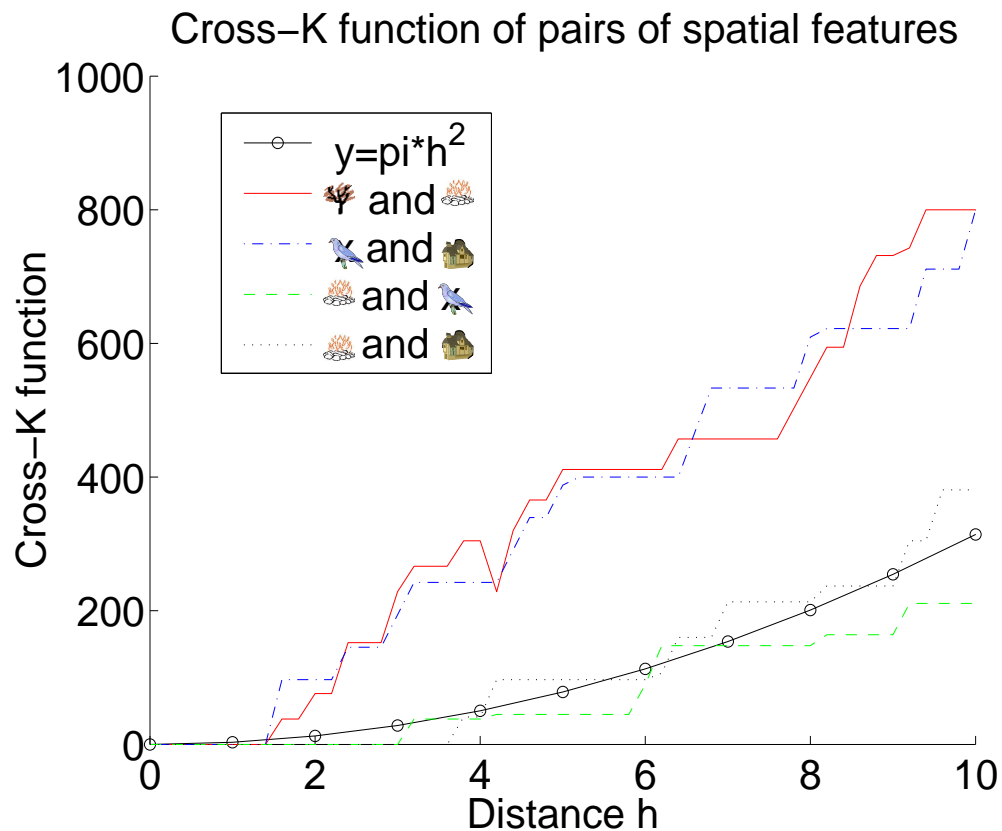
- * Spatial statistical approach
- * Classical data mining association rule approach
 - * Reference feature approach
 - * Partitioning approach

Related Work: Statistical Approach

★ Ripley's K-function:

★ $K_{ij}(h) = \lambda_j^{-1} E$ [number of type j event within distance h of a randomly chosen type i event]

★ Ripley's K-function of some pair of spatial feature types








★ Properties:

★ Not well defined for size ≥ 3

★ Expensive Monte Carlo simulation for confidence band

Association Rules - An Analogy

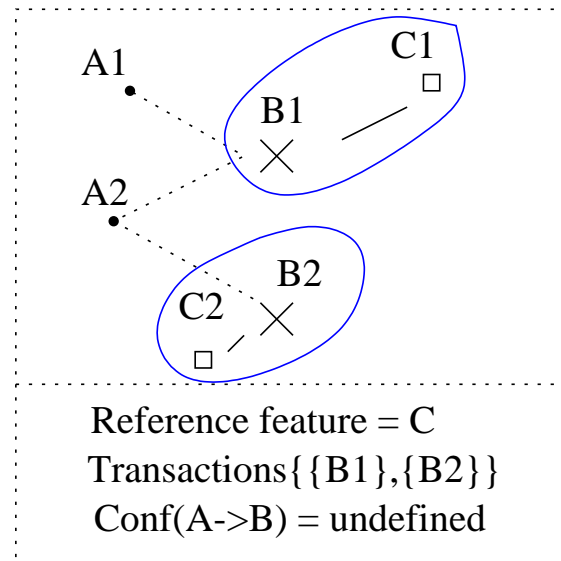
- * Association rule e.g. (Diaper in T \Rightarrow Beer in T)

rans.	Items Bought
	{socks,  milk,  , beef, egg, ... }
	{ pillow,  , toothbrush, ice-cream, muffin, ... }
	{  ,  , pacifier, formula, blanket, ... }
.	...
	{battery, juice, beef, egg, chicken, ... }

- * Support: probability(Diaper and Beer in T) = 2/5
- * Confidence: probability(Beer in T|Diaper in T) = 2/2
- * Algorithm Apriori [Agrawal, Srikant, VLDB94]
 - * Support based pruning using monotonicity
- * Note: **Transaction is a core concept!**

Related Work: Association Rule Approach

- ★ Reference feature centric model [Koperski, Han, SSD95]



★ Properties

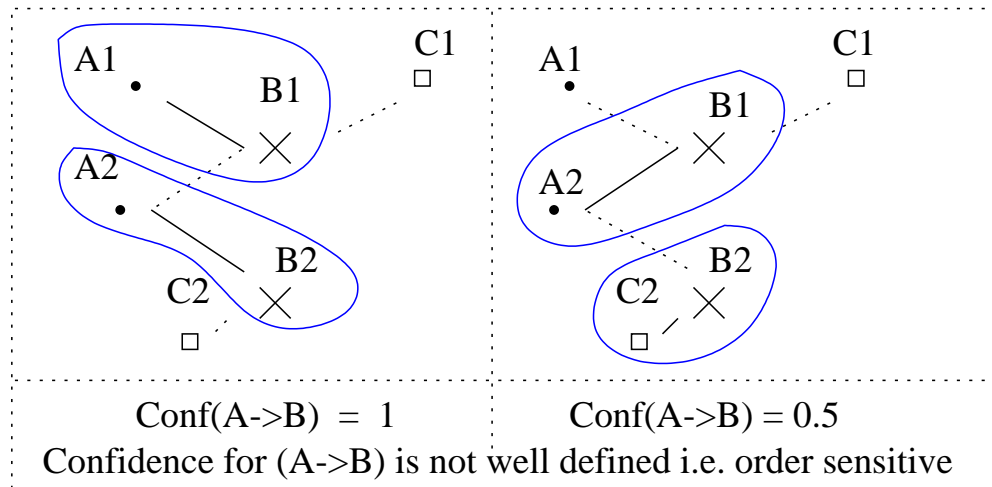
- ★ All relevant co-locations reference to one feature
- ★ Item types = boolean spatial features
- ★ Transactions = defined around instances of reference feature
- ★ Force-fit notion of transaction

★ Limitations

- ★ May under-count support for a pattern, e.g (A,B)
- ★ May over-counter support
- ★ Results not comparable with spatial statistical approach

Related Work: Association Rule Approach

* Partitioning approach [Morimoto, SIGKDD01]



* Properties

- * Divide dataset into partitions
- * Item types = boolean spatial features
- * Transactions = partitions

* Limitations

- * Order sensitive transactions
- * Support and confidence are ill-defined

Limitation of Related Work and Our Contributions

- * Limitation of Related Work
 - * Expensive computation
 - * Force-fit transaction on spatial dataset

- * Our Contributions
 - * Event centric co-location model
 - Robust in face of overlapping neighborhoods
 - * Co-location Miner algorithm
 - Computational efficiency
 - * High confidence low prevalence co-location patterns
 - * Validity of inferences

Overview

- * Introduction
- * Related Work
- ⇒ Event Centric Approach
- * Co-location Miner Algorithm
- * Evaluation
- * Conclusions and Future Work

Our Approach: Event Centric Model

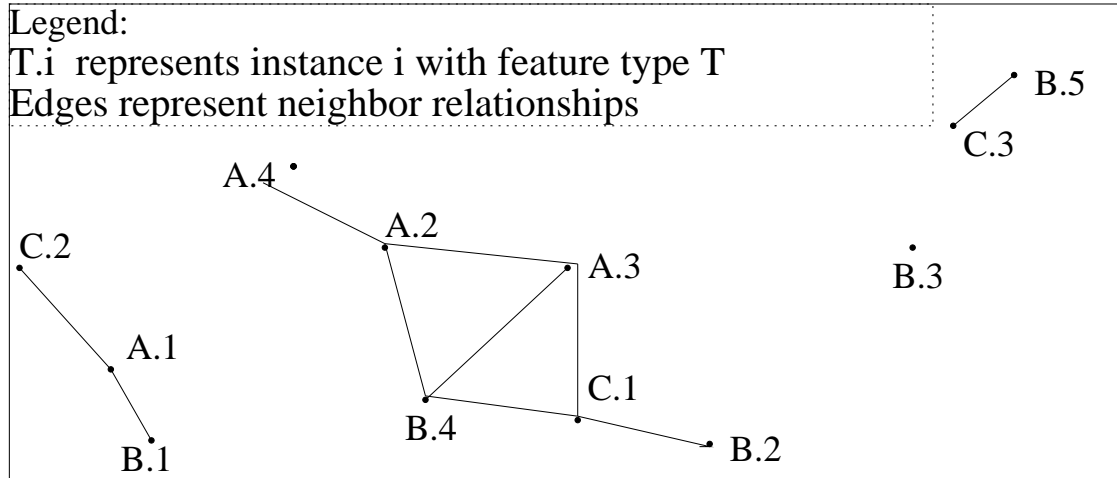
★ Association Rules Vs. Co-location Rules

Criteria	Association Rule	Co-location Rule
Underlying Space	Discrete Sets	Continuous Space
Item Types	Product types	Spatial Features(Boolean)
Item Collections	Transactions $\{T_i\}$	Neighborhoods
Prevalence ($A \rightarrow B$)	Support: $p(A \cup B \in T_i)$	Participation Index
Conditional Probability ($A \rightarrow B$)	$p(B \in T_i A \in T_i)$	$p(B \in \text{Nbr}(L) A \text{ at } L)$

- ★ An example: A happens $\rightarrow B$ happens in A 's neighborhood with 100% conditional probability

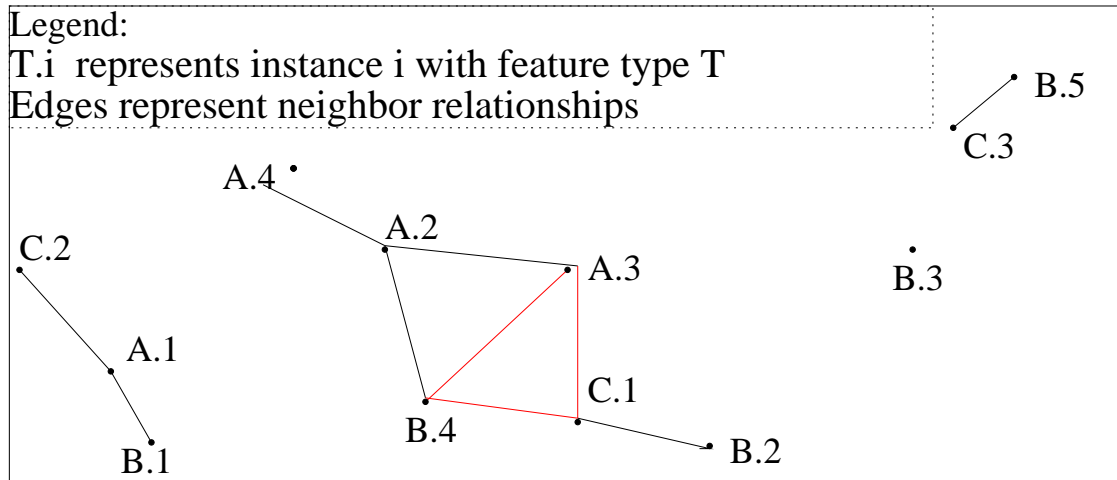
Key Concepts

* Example Dataset



Key Concepts

* Example Dataset

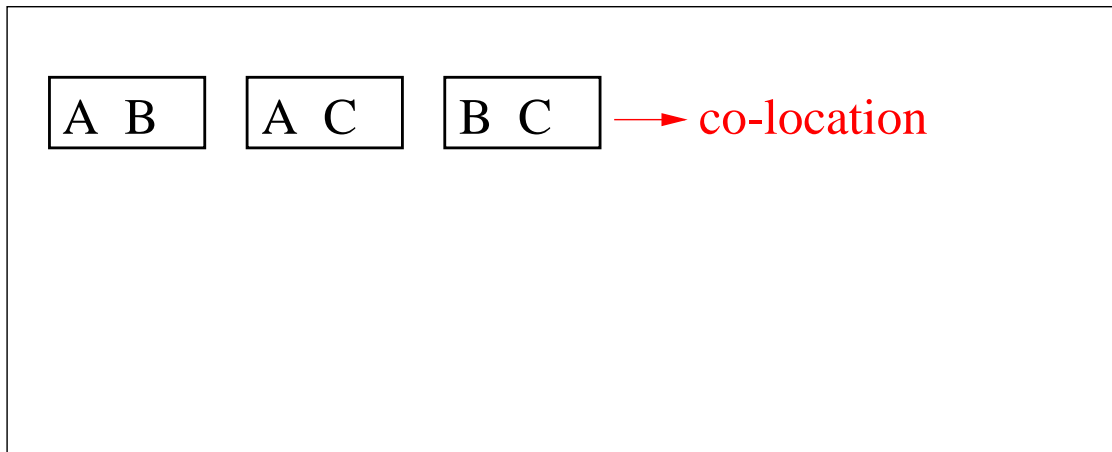
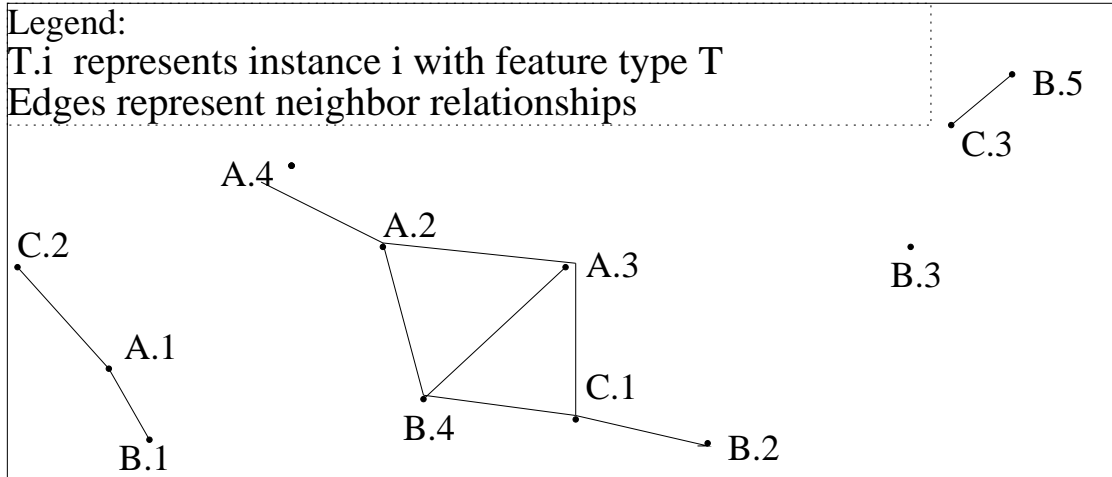


* A neighborhood:

- * A clique in a graph of neighbor relation R

Key Concepts

★ Example Dataset

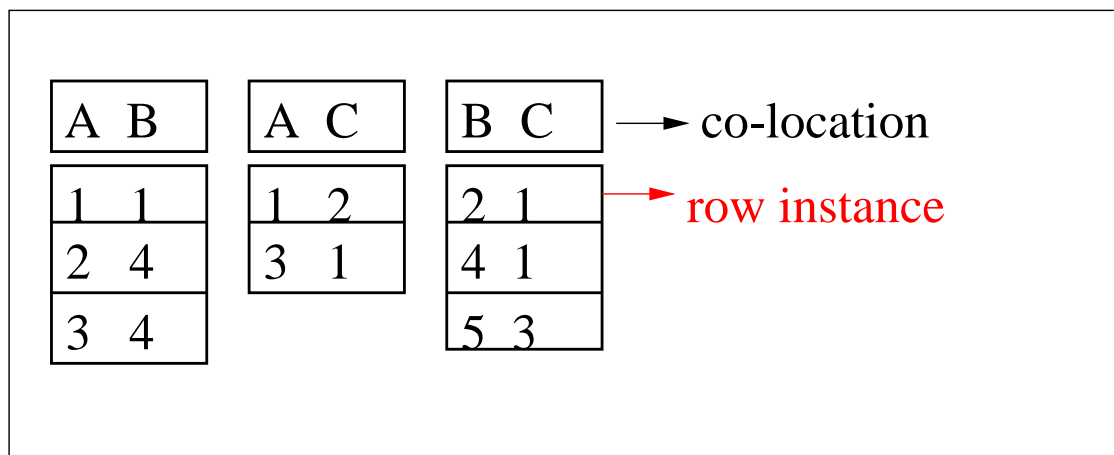
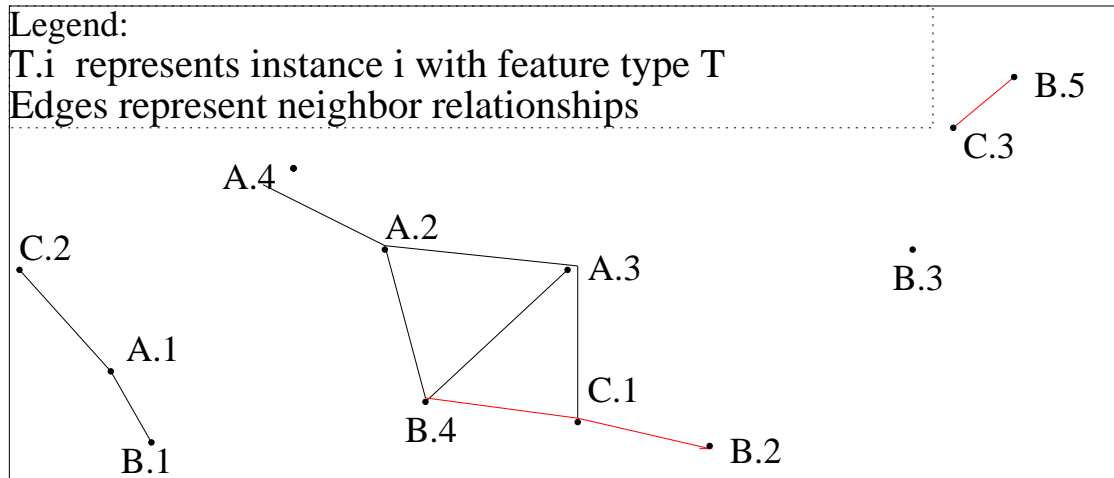


★ A co-location C :

- ★ A subset of boolean spatial features

Key Concepts

* Example Dataset



* A row instance I of a co-location $C = \{f_1, \dots, f_k\}$:

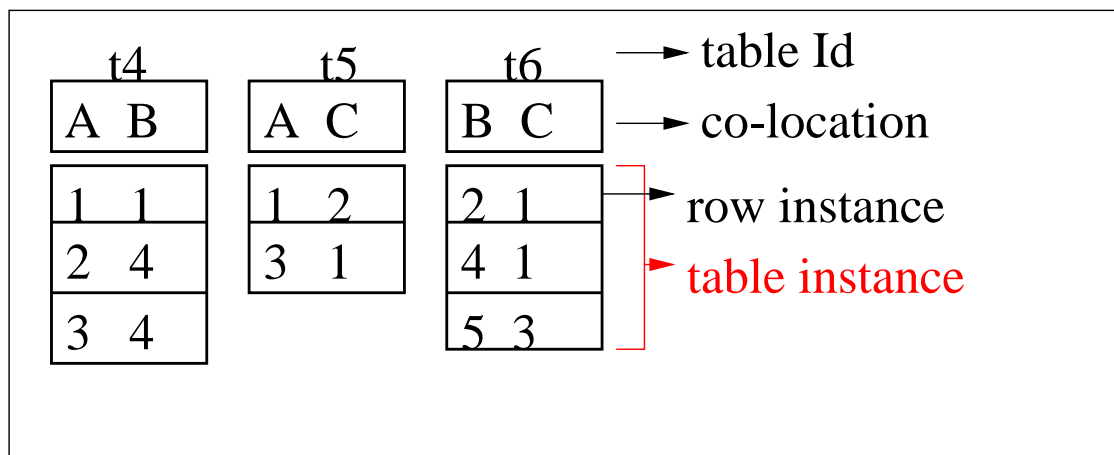
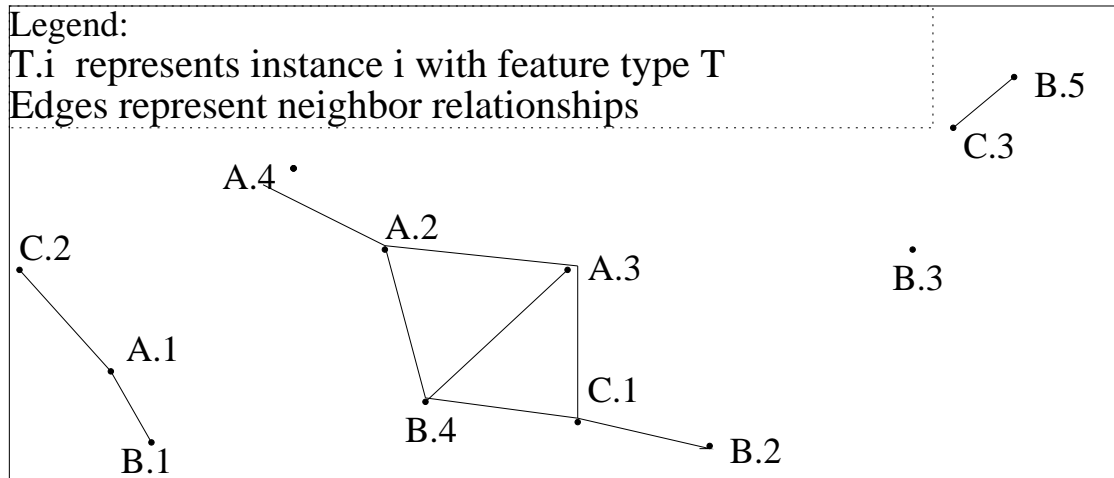
* $I = \{i_1, \dots, i_k\}$

* i_j : instance of $f_j (\forall j \in 1, \dots, k)$

* I is a neighborhood

Key Concepts

* Example Dataset

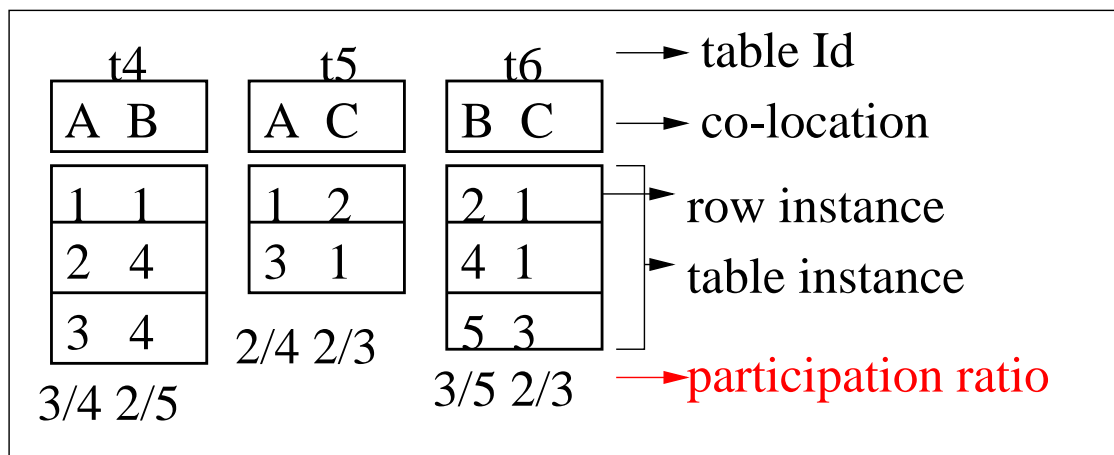
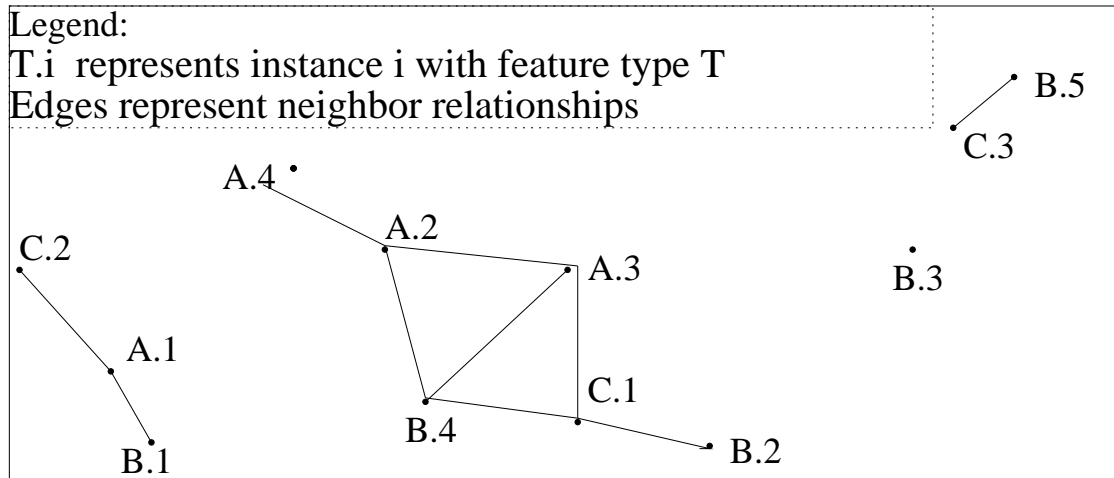


* Table instance(co-location $C = \{f_1, \dots, f_k\}$):

- * Collection of all its row instances
- * Spatial join interpretation

Key Concepts

* Example Dataset



* Participation ratio

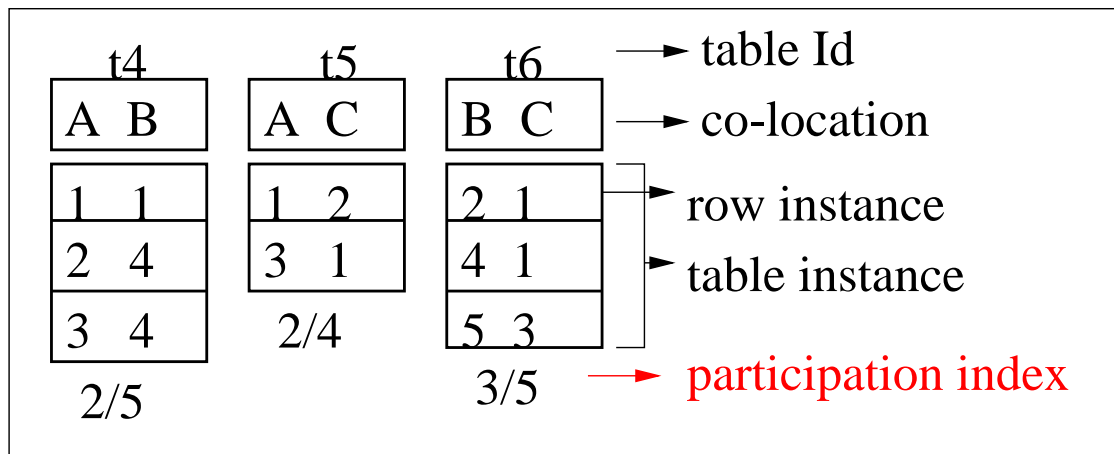
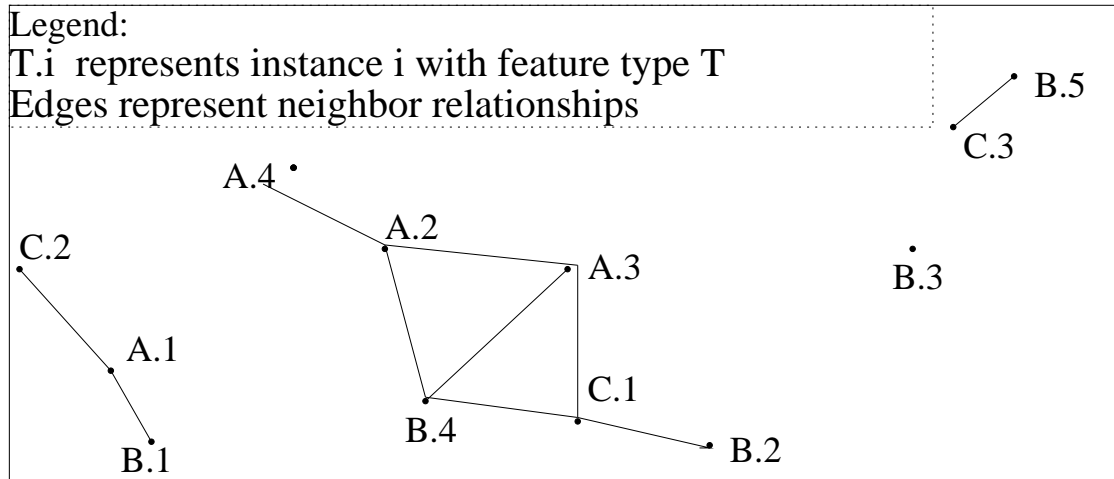
$$* pr(C, f_i) = |\pi_{f_i} table instance(C)| / |instances(f_i)|$$

$$* C = \{f_1, f_2, \dots, f_k\}$$

* Co-location strength of a spatial feature in a pattern

Key Concepts

★ Example Dataset



★ The participation index

$$* pi(C) = \min_{i=1}^k pr(C, f_i)$$

* Co-location strength of a pattern

Key Concepts

- * **A neighborhood:**
 - * A clique in a graph of neighbor relation R
- * **A co-location C :**
 - * A subset of boolean spatial features
- * **A row instance I of a co-location $C = \{f_1, \dots, f_k\}$:**
 - * $I = \{i_1, \dots, i_k\}$
 - * i_j : instance of $f_j (\forall j \in 1, \dots, k)$
 - * I is a neighborhood
- * **Table instance(co-location $C = \{f_1, \dots, f_k\}$):**
 - * Collection of all its row instances
 - * Spatial join interpretation

Key Concepts

★ **Participation ratio (PR)**

★ $pr(C, f_i) = |\pi_{f_i} \text{table instance}(C)| / |\text{instances}(f_i)|$

★ $C = \{f_1, f_2, \dots, f_k\}$

★ **Participation index (PI)**

★ $pi(C) = \min_{i=1}^k pr(C, f_i)$

★ **Lemma 1** [Monotonicity] Participation ratio and participation index are monotonically decreasing with respect to co-location size

★ **Proof:**

- An instance of A participates in $\{A, B, \dots\}$, it must participate in $\{A, B\}$
- PR is monotonic
- PI is the minimal of PR, monotonic too

★ **A co-location rule $C_1 \rightarrow C_2(p, cp)$:**

★ C_1 and C_2 are co-locations

★ p = prevalence measure, e.g. participation index

★ $cp = \Pr[C_2 \in N(L) \mid C_1 @ L] = \frac{|(\pi_{C_1}(\text{table instance of } (C_1 \cup C_2)))|}{|\text{instance of } C_1|}$

- π is a projection operation

Overview

- * Introduction
- * Related Work
- * Event Centric Approach
- ⇒ Co-location Miner Algorithm
- * Evaluation
- * Conclusions and Future Work

Problem Formulation

★ Given:

- ★ K Boolean spatial feature types
- ★ Instances $\langle \text{id}, \text{feature type } t, \text{location } l \rangle$
- ★ A neighbor relation R over locations
- ★ Prev_threshold and cp_threshold

★ Find:

- ★ Co-location rules with prevalence $>$ prev_threshold and conditional probability $>$ cp_threshold

★ Objectives:

- ★ Efficiency

★ Constraints:

- ★ Correctness
 - Every co-location found has prevalence $>$ prev_threshold and conditional probability $>$ cp_threshold
- ★ Completeness
 - Find all the co-locations with prevalence $>$ prev_threshold and conditional probability $>$ cp_threshold
- ★ Monotonic prevalence measure
- ★ Event centric model

Revisit related work in light of problem formulation

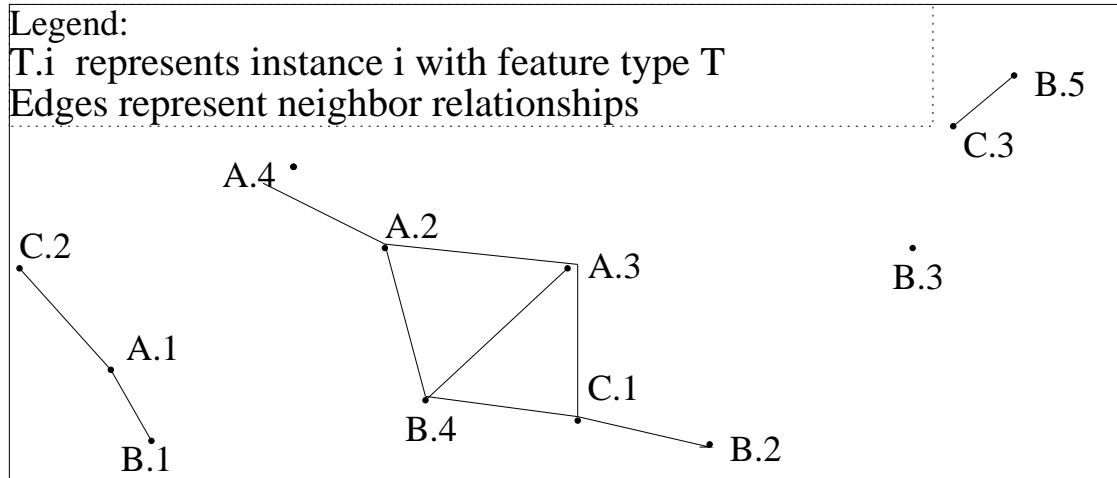
	Correct	Complete	Efficient
K function	Y	Y	N
Reference feature centric	N	N	Y
Partitioning	N	N	Y
Event centric	Y	Y	Y

Co-location Miner Algorithm: Basic Idea

- * Initialization
- * **for** k in $(2, 3, \dots, K - 1)$ **and** prev. co-location found **do**
 - * 1. Generate size k candidate co-locations
 - * 2. Multi-resolution or other filtering methods
 - * 3. Generate table instances
 - * 4. Calculate prevalence and select prevalent co-locations
 - * 5. Generate co-location rules of size k
- * **end**
- * Note: Step 3 not needed in mining association rules
 - * because item collections (i.e. transactions) are given

Algorithm Trace

* Running Example



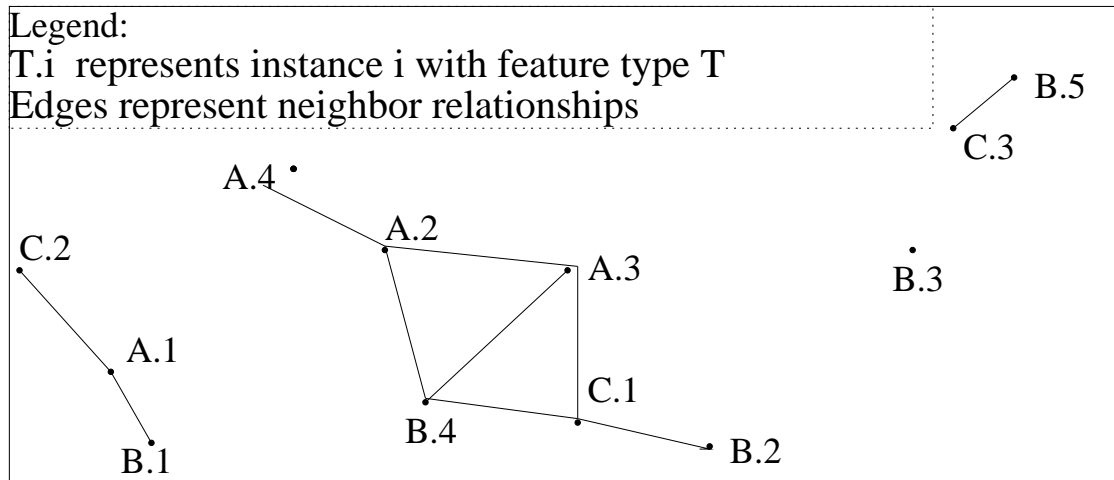
* Running Example

k=1		
t1	t2	t3
A	B	C
1	1	1
2	2	2
3	3	3
4	4	1
1	5	
	1	

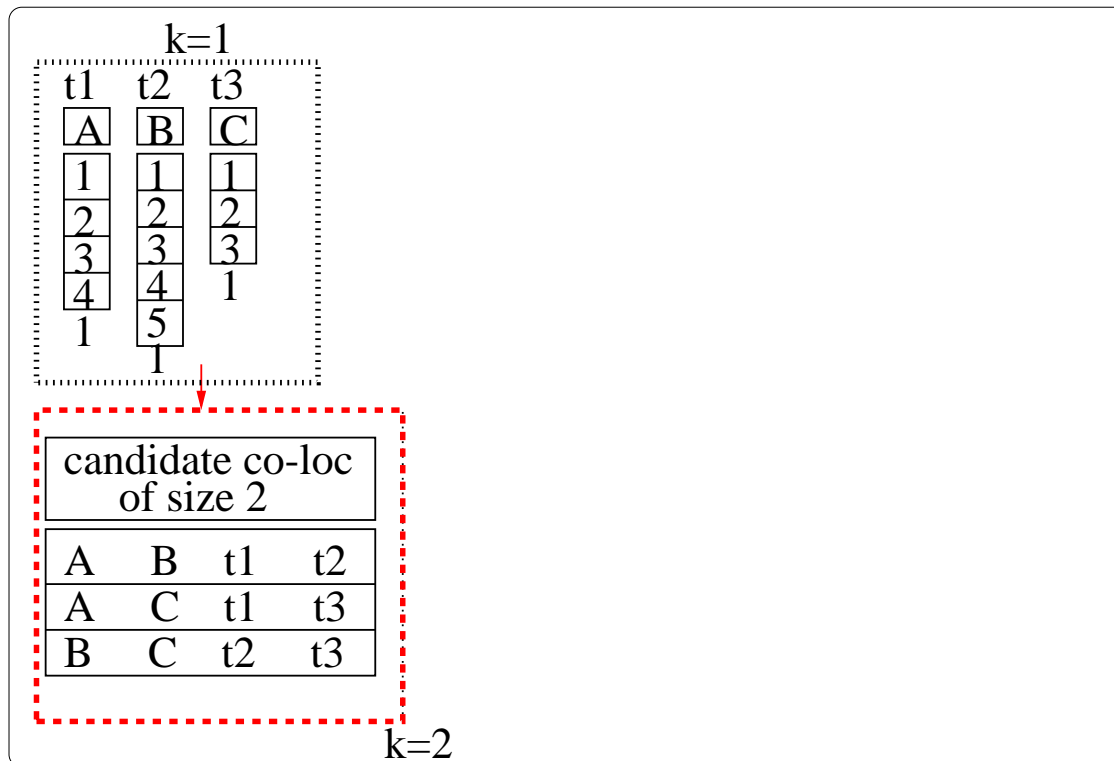
* Initialization

Algorithm Trace

* Running Example



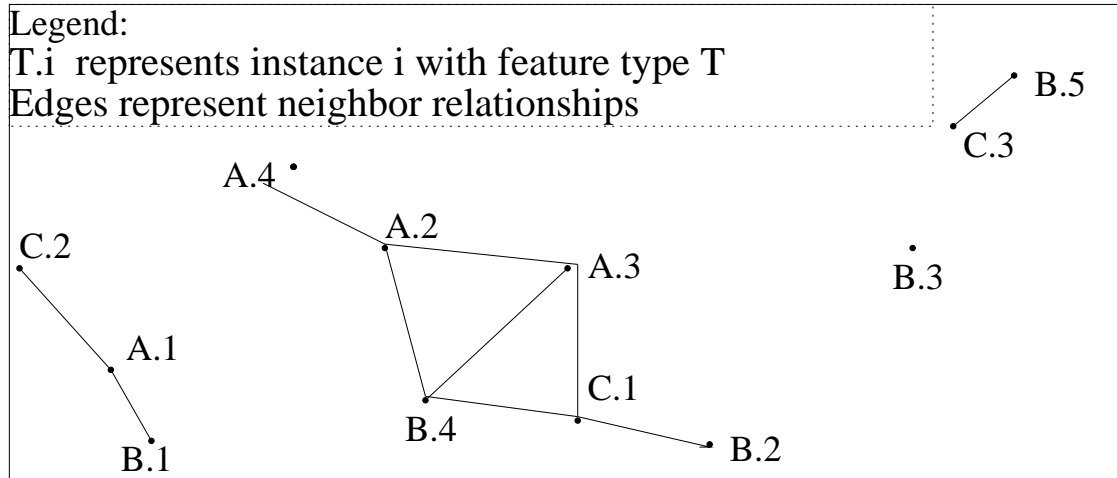
* Running Example



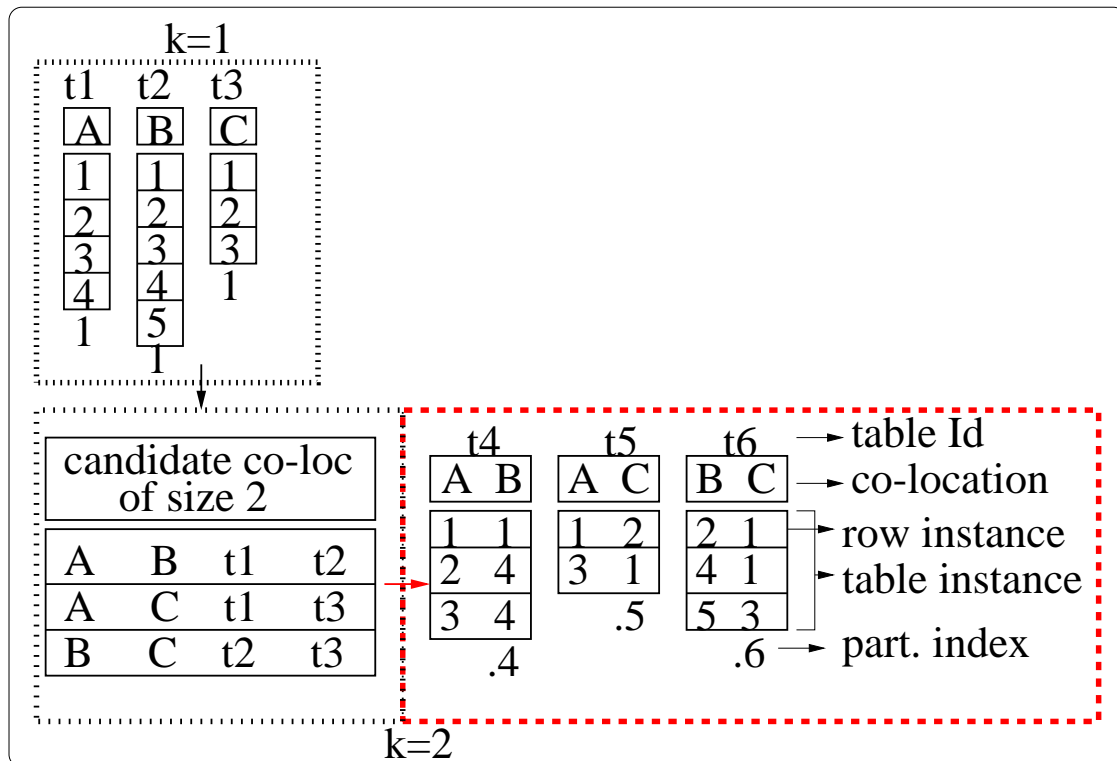
* $k = 2$, generate size 2 candidate co-locations (step 1)

Algorithm Trace

* Running Example



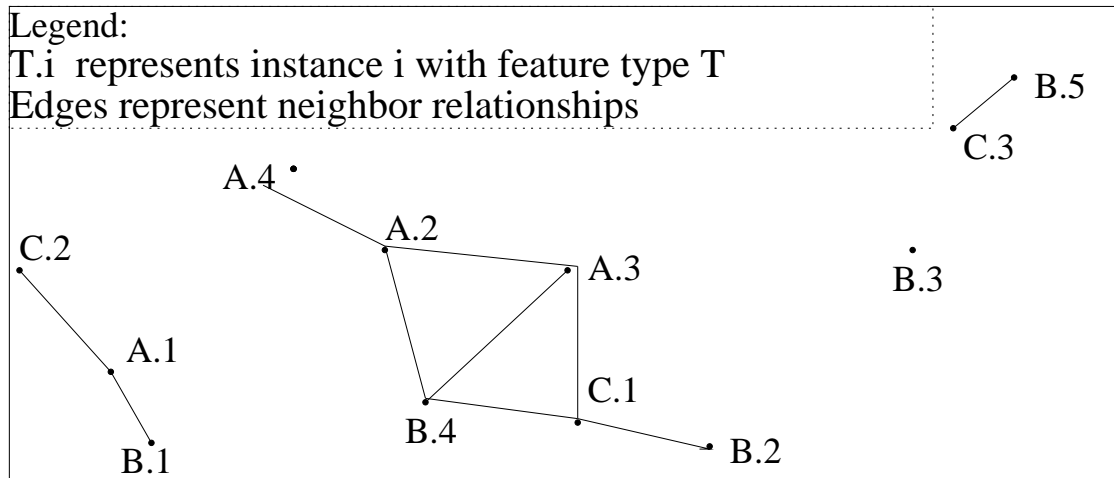
* Running Example



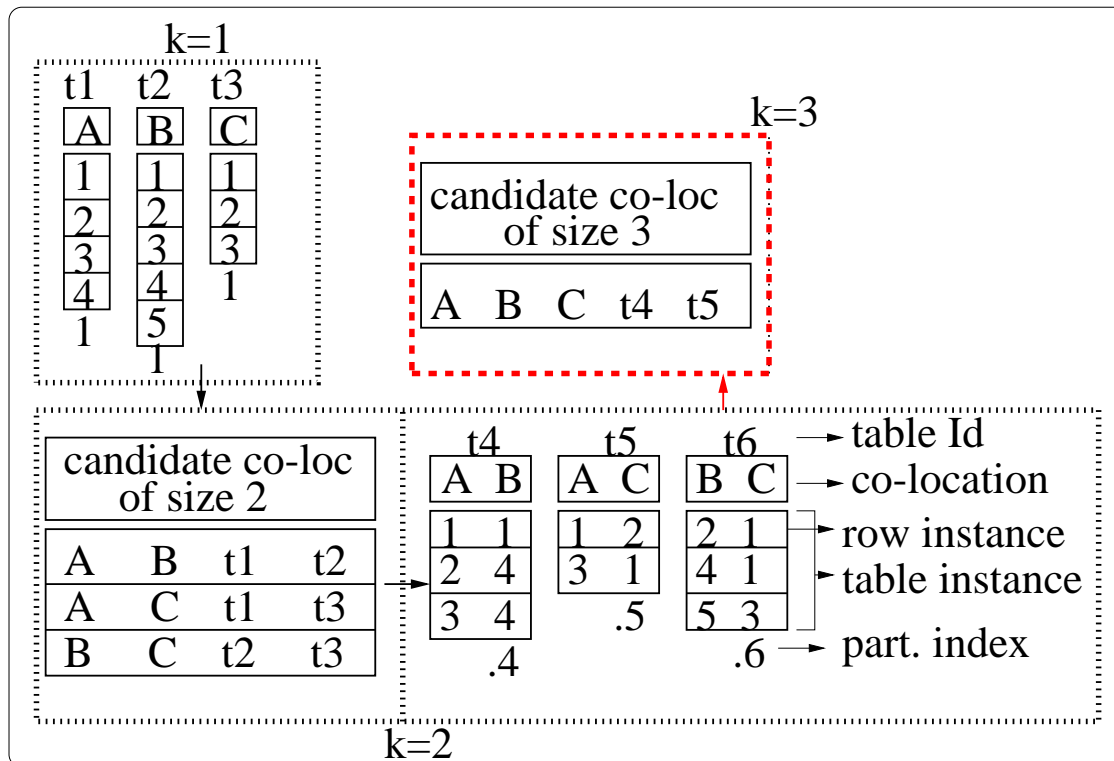
* $k = 2$, generate size 2 table instances ... (steps 3,4,5)

Algorithm Trace

* Running Example



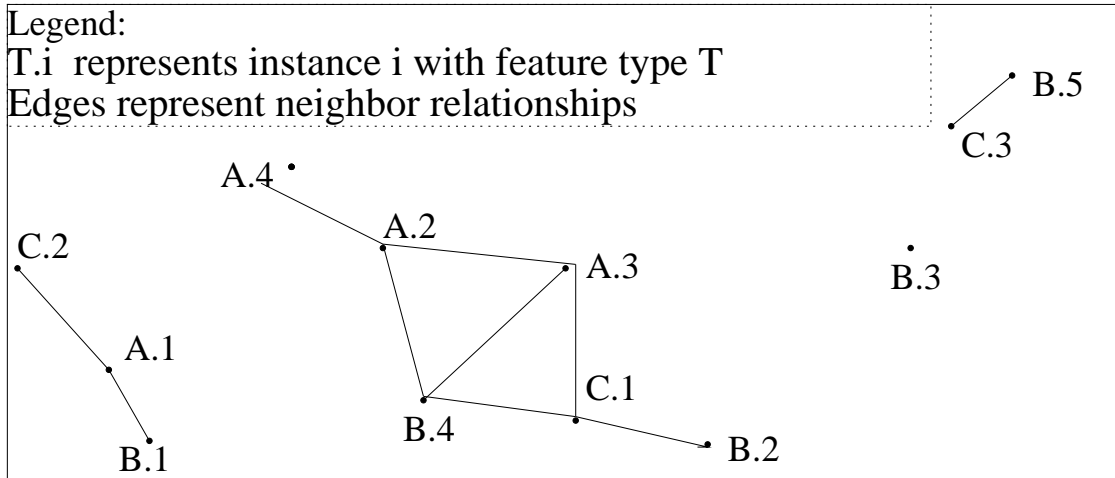
* Running Example



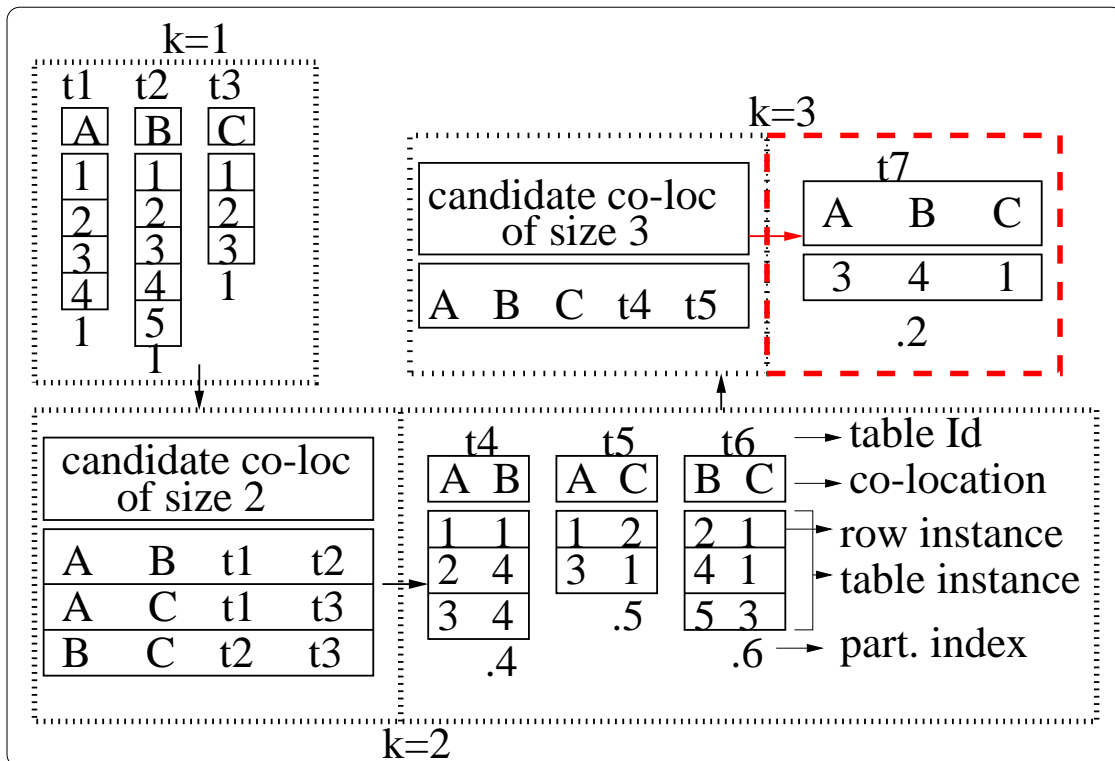
* $k = 3$, generate size 3 candidate co-locations (step 1)

Algorithm Trace

* Running Example



* Running Example



* $k = 3$, generate size 3 table instances .. (steps 3,4,5)

Some Details of Co-location Miner

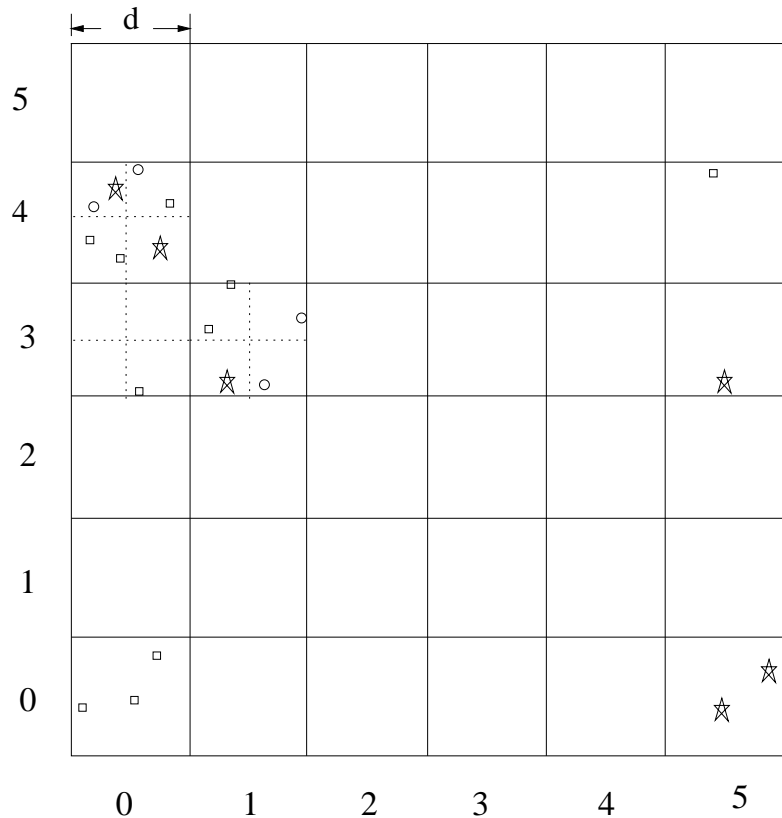
- ★ Generate candidate co-locations
 - ★ Similar to that in association rule mining
- ★ Participation indexes calculation
 - ★ Bitmap index based
 - ★ One scan of table instances in current iteration
- ★ Co-location rule generation
 - ★ Conditional probability of co-location rule $C_1 \rightarrow C_2$
$$= \frac{|\{\pi_{C_1}(\text{table instance of } (C_1 \cup C_2))\}|}{|\text{instance of } C_1|}$$
 - ★ Bitmaps or other data structures
 - ★ Similar strategies for prevalence based pruning

Performance Tuning

- ★ An optional filter
 - ★ Multi-resolution filter
 - ★ Hierarchical structure, e.g. grid files and R-tree
 - ★ Reuse bitmaps in the previous iteration
- ★ Join strategies for generating table instances
 - ★ Geometric: plane sweep, space partition, and tree matching
 - ★ Combinatorial
 - ★ Hybrid

A Multi-resolution Filter

★ Illustration:



Co-location of size 1
and coarse level table instances

c1	c2	c2
★	□	○
(0, 4)	(0, 0)	(0, 4)
(1, 3)	(0, 3)	(1, 3)
(5, 0)	(0, 4)	1
(5, 3)	(1, 3)	
1	(5, 4)	
	1	

★ Process

- ★ Summarize data at a coarse resolution
- ★ Generate coarse level table instances
- ★ Calculate over-estimated participation index
- ★ Eliminates a co-location if its over-estimated index falls below user give threshold

Join Strategies

- ★ Geometric
 - ★ In practice use filter and refine
 - ★ Minimum bounding rectangle
 - ★ then exact geometry and predicates are considered

- ★ Combinatorial
 - ★ Sort-merge join strategy
 - Match the first k-1 instances
 - Efficient since instances of co-locations are sorted already
 - ★ then check if the last two instances are neighbors

- ★ Hybrid
 - ★ Choose the more promising of the
 - spatial and combinatorial approaches
 - in each iteration

Overview

- * Introduction
- * Related Work
- * Event Centric Approach
- * Co-location Miner Algorithm
- ⇒ Evaluation
- * Conclusions and Future Work

Analytical Evaluation: Correctness and Completeness

- * Definition:
 - * Completeness:
Find all rules with prevalence $>$ prev_threshold and conditional probability $>$ cp_threshold
 - * Correctness:
Any rules found have prevalence $>$ prev_threshold and conditional probability $>$ cp_threshold
- * Lemma
 - * Co-location Miner is complete and correct
- * Proof Sketch
 - * Participation index is monotonic in size of co-location
 - * Any subset of a prevalent co-location is prevalent
 - * Table join will not miss any row instance

Analytical Evaluation: Ascertaining the Quality of the Inferences

- * $pi(A, B)$ is an upper bound on $\frac{\hat{K}_{AB}(h)}{W}$
 - * $\hat{K}_{AB}(h)$ is the estimation of the $K(A, B)$
 - * W is the total area defined by distance $\leq h$
- * Table instance $t(A, B)$ of a binary co-location (A, B)
 - * has enough information to compute $\hat{K}_{AB}(h)$
 - * for $h = d$
 - * $\frac{\hat{K}_{AB}(h)}{W} = \frac{1}{|A|} \cdot \frac{|t(A, B)|}{|B|}$

Analytical Evaluation: Choice of Join Strategies

- * Geometric
 - * keep information of nearby regions
 - * Lack spatial feature type level pruning
- * Combinatorial
 - * benefits from spatial feature type level pruning
 - * do not keep spatial proximity information
- * Hybrid: integrate the best features of the two join strategies

Analytical Evaluation: When to Use Additional Filtering

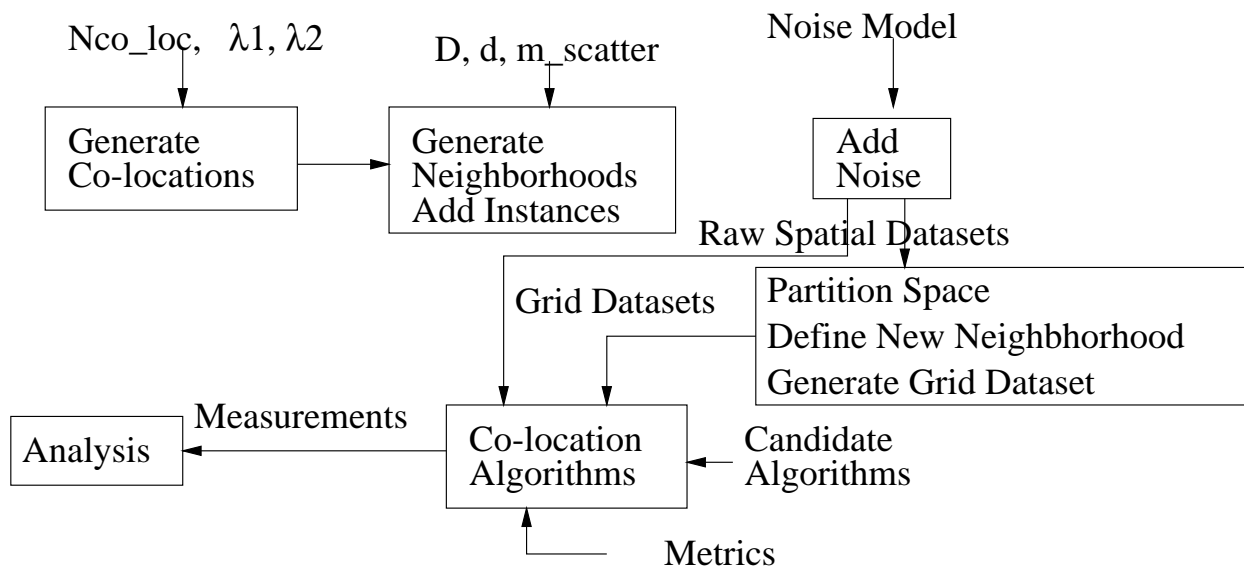
- * Running time ratio without/with filtering:

$$\begin{aligned} \frac{t_{filter}(k)}{t(k)} &\approx \frac{|C_{k+1}| \times T_{grid}(k) + |C'_{k+1}| \times T_{orig}(k)}{|C_{k+1}| \times T_{orig}(k)} \\ &= \frac{T_{grid}(k)}{T_{orig}(k)} + \frac{|C'_{k+1}|}{|C_{k+1}|} \end{aligned} \quad (1)$$

- * C_{k+1} : number of size $k+1$ candidates before filtering
 - * C'_{k+1} : number of size $k+1$ candidates after filtering
 - * $T_{grid}(k)$: average time for a coarse level table instance
 - * $T_{orig}(k)$: average time for a fine level table instance
- * Choice of filtering is affected by
 - * Filtering ratio
 - * Dataset clustering level

Performance Evaluation

- * Experiment goals
 - * How do join strategies affect the performance?
 - * When to use additional filtering?
- * Experiment Design



- * Setup
 - * Sun Ultra 10 work station
 - * with a 440 MHz CPU
 - * 128 Mbytes memory
 - * running the SunOS 5.7 operating system

Performance Evaluation

★ Parameters

Parameter	Definition	C
N_{co_loc}	The number of core co-locations	5
λ_1	The parameter of the Poisson distribution to define the size of the core co-locations	5
λ_2	The parameter of the Poisson distribution to define the size of the table instance of each co-location when $m_{clump} = 1$	50
$D_1 \times D_2$	The size of the spatial framework	$10^6 \times 10^6$
d	The size of the square to define a co-location	10
r_{noise_f}	The ratio the of number of noise features over the number of features involved in generating the maximal co-locations	.5
r_{noise_n}	The number of noise instances	50,000
$m_{overlap}$	The number of co-location generated by appending one more spatial feature for each core co-location	1
m_{clump}	The number of instances generated for each spatial feature in a neighborhood for a co-location	1

★ Report results on a representative dataset C

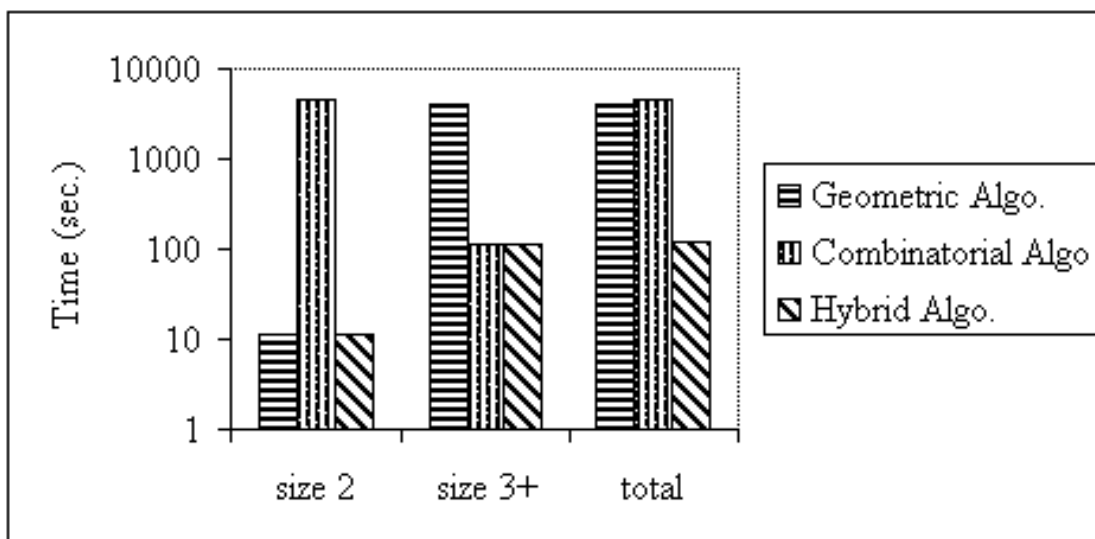
- ★ Variable parameters of dataset C are reported for each experiment

Performance Evaluation

- * Relative performance of geometric, combinatorial, and hybrid join strategies

- * Prevalence threshold set to 0.9

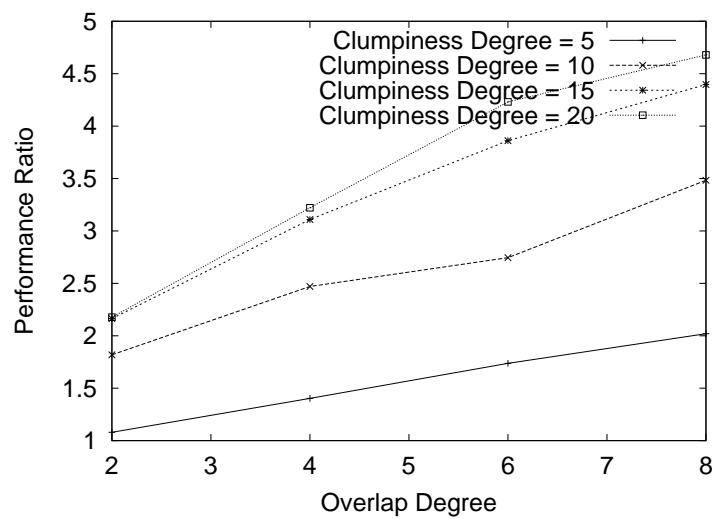
- * Result



- * Geometric: faster to generate co-locations of size 2
- * Combinatorial: faster (magnitude of 2) to generate co-locations of size 3+
- * Hybrid: combine geometric and combinatorial

Performance Evaluation

- ★ Effect of multi-resolution filtering
- ★ Variable parameter: $m_{overlap}$ from 2 to 8
- ★ Result:

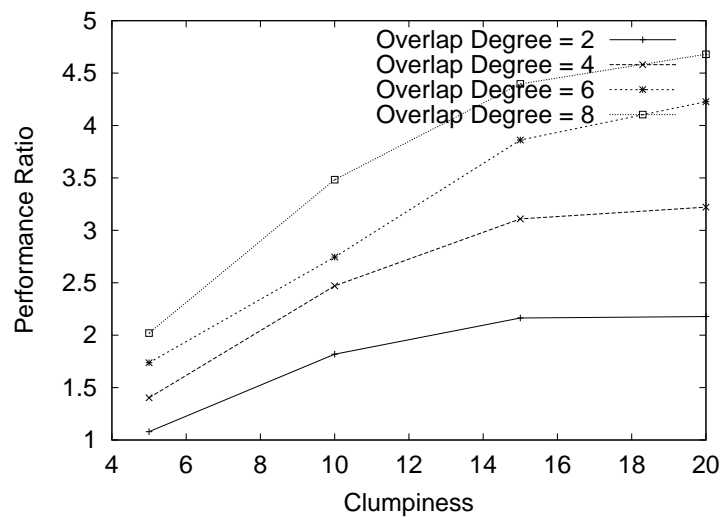


- ★ Multi-resolution filtering is effective especially when overlapping degree is high
- ★ Algebraic explanation:

$$\frac{t_{filter}(k)}{t(k)} \approx \frac{T_{grid}(k)}{T_{orig}(k)} + \frac{|C'_{k+1}|}{|C_{k+1}|} \quad (2)$$

Performance Evaluation

- ★ Effect of multi-resolution filtering
- ★ Variable parameter: m_{clump} from 5 to 20
- ★ Result:

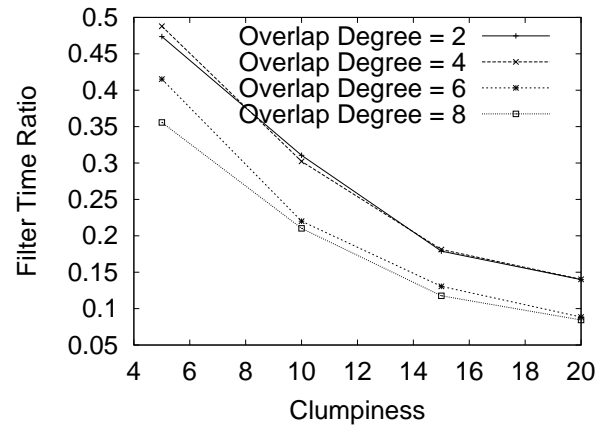
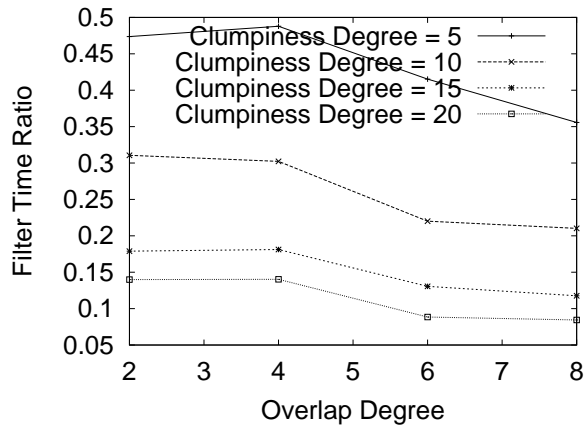


- ★ Multi-resolution filtering is effective especially when dataset is clustered
- ★ Algebraic explanation:

$$\frac{t_{filter}(k)}{t(k)} \approx \frac{\mathbf{T}_{grid}(\mathbf{k})}{\mathbf{T}_{orig}(\mathbf{k})} + \frac{|C'_{k+1}|}{|C_{k+1}|} \quad (3)$$

Performance Evaluation

- ★ Effect of multi-resolution pruning: filter time ratio



- ★ Filter time ratio
 - ★ Filter time is 10% to 50% of the total running time

Overview

- * Introduction
- * Related Work
- * Event Centric Approach
- * Co-location Miner Algorithm
- * Evaluation
- ⇒ Conclusions and Future Work

Conclusions and Future Work

- ★ Our contributions described today
 - ★ Event centric co-location model
 - Robust in face of overlapping neighborhoods
 - ★ Co-location Miner algorithm
 - Computational efficiency
 - Correctness and completeness with various performance tuning
 - ★ Validity of inferences
- ★ Other contributions in my thesis
 - ★ High-confidence Low-prevalence (HCLP) Patterns
 - Prevalence base pruning: hard to retain HCLP patterns
 - Proposed a measure to retain such patterns
 - Proved a weak monotonicity of the proposed measure
 - Designed an algorithm using the weak monotonicity
 - ★ May find pattern
 - chromium 6 → lung disease, breast cancer in spatial proximity

Future Work

- * Co-location patterns involving lines and polygons
- * Temporal co-incidence mining
 - * No natural concept of transactions over temporal datasets
 - * Arbitrary windowing may not be desirable
- * Spatio-temporal dataset

Future Work in a Longer Term

- ★ Environmental Biology
 - ★ Jane Goodall's Chimpanzee behavior dataset analysis
- ★ Emergency Evacuation Planning
 - ★ Heuristic approaches
- ★ Scientific Data Management
 - ★ EOS by NASA collecting terabyte of information each day
 - ★ Spatial and temporal in nature
- ★ Moving Object Databases/Location Based Services
 - ★ Data mining: location based recommendation
 - ★ Database systems
 - support millions of triggers
 - answer proximity queries
 - keep trajectories of moving objects

Thanks!

