A multiple window-based co-location pattern mining approach for various types of spatial data

M. Venkatesan* and Arunkumar Thangavelu

School of Computing Science and Engineering,
Vellore Institute of Technology,
Vellore-14, Tamilnadu, India
E-mail: mvenkatesan@vit.ac.in
E-mail: arunkumar.thangavelu@gamil.com
*Corresponding author

Abstract: Co-location pattern analysis represents the subsets of spatial events whose instances are found in close geographic proximity. Given a collection of Boolean spatial features, the co-location pattern discovery process finds the subsets of features frequently located together. Key challenges in co-location pattern analysis are modelling of neighbourhood in spatial domain, minimum prevalent threshold to generate collocation patterns and analysing extended spatial objects. We discuss the above key challenges using event centric approach and N-most prevalent co-location patterns approach. We propose a window-based model to find the neighbourhood for point spatial datasets and the multiple window model for extended spatial data objects. We also use N-most prevalent co-location patterns approach to filter the number of co-location pattern generation. We propose a more generic and efficient window-based model algorithm to find co-location patterns. Towards the end, we have done a comparative study of the existing approaches with our proposed approach.

Keywords: co-location; window; neighbourhood; spatial data.

Reference to this paper should be made as follows: Venkatesan, M. and Thangavelu, A. (2013) 'A multiple window-based co-location pattern mining approach for various types of spatial data', *Int. J. Computer Applications in Technology*, Vol. 48, No. 2, pp.144–154.

Biographical notes: M. Venkatesan is working as an Assistant Professor (selection grade) in School of Computing Science and Engineering, Vellore Institute of Technology, Vellore, Tamilnadu. He is working in the area of spatial data mining. His research area includes databases, data warehouse, data mining and applications of data mining in disaster management. He has good knowledge in data mining tools like Clementine, rapidminer. He is the principal investigator in ISRO funded project on data mining in a landslide. He completed his BE, MTech and doing his PhD in the area of spatial data mining.

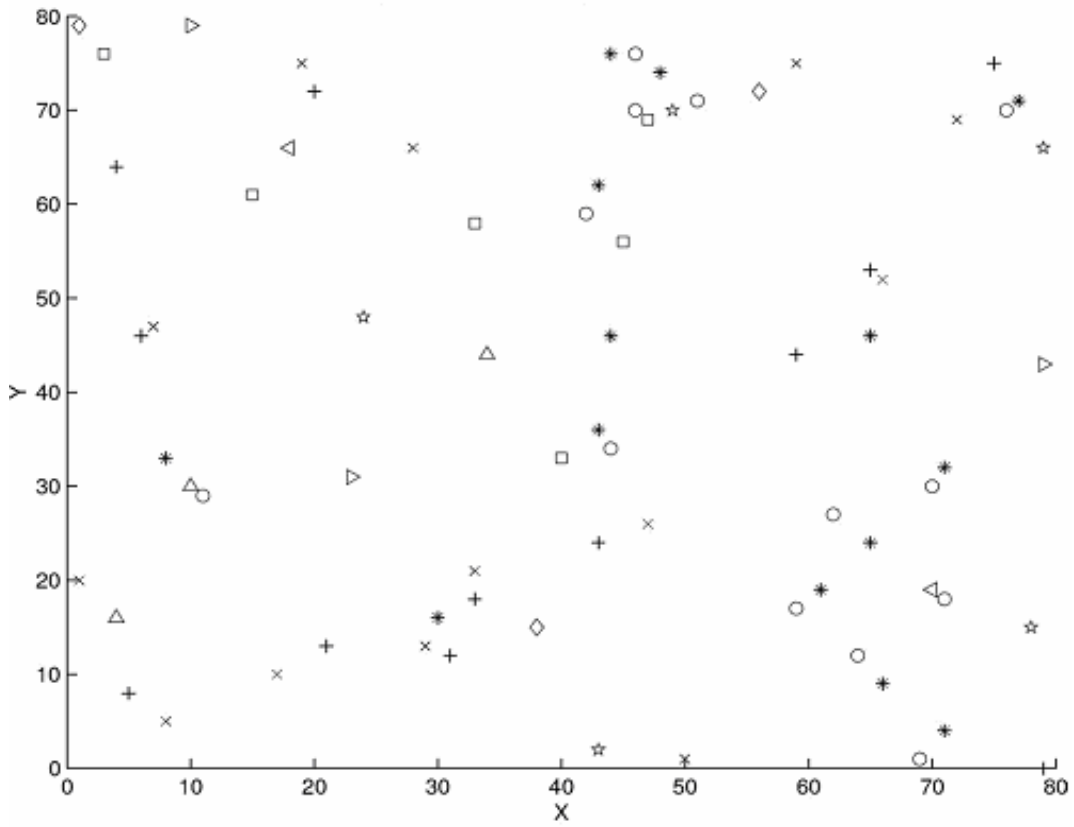
Arunkumar Thangavelu is working as a Professor in School of Computing Science and Engineering, Vellore Institute of Technology, Vellore. His research area includes data mining, and sensor networks. He has published a number of papers in the field of ad hoc networks and data mining. Currently, he is working as the Deputy Director Academics in Vellore Institute of Technology. He has completed his MCA and PhD. He is also an editor and reviewer for the number of journals.

1 Introduction

Spatial co-location pattern analysis is an important area in the spatial domain. Given collection of Boolean spatial features, the co-location pattern discovery process finds the subsets of features frequently located together. Co-location pattern analysis are used in many numbers of applications like e-services, ecology, real estate and disaster management like flood, landslide and earthquake…. Highways in large metropolitan area often have frontage roads nearby. Identification of such co-locations is useful in selecting test-sites for evaluating in-vehicle navigation technology. In flood analysis, areas which are near to river are useful to find the flood affected areas.

Co-location patterns represent subsets of Boolean spatial features whose instances are often located in close geographic proximity. Boolean spatial features describe the presence or absence of geographic object types at different locations in a two dimensional or three-dimensional metric space, such as the surface of the earth. Examples of Boolean spatial features include plant species, animal species, road types, cancers, crime, and business types. Figure 1 shows a dataset consisting of instances of several Boolean spatial features, each represented by a distinct shape.

Figure 1 Point spatial co-location pattern



A careful review reveals from the above figure that there are two co-location patterns as {'+', 'x'} and {'o', '*'}. Co-location rules (Shekhar and Chawala, 2003) are models to infer the presence of spatial features in the neighbourhood of instances of other spatial features. For example, 'HighwayRoad -> House' predicts the presence of the house in areas within highwayroad. The main idea of this paper is to overcome the key challenges in modelling of neighbourhood in a spatial domain. In this paper, we discuss an event centric approach and N-most prevalent approach, and we also discuss the proposed window-based model to find the neighbourhood for point spatial datasets and the multiple window model for extended spatial data objects. We finally compare our proposed approach with existing approaches.

This paper is structured as follows: Section 2 discusses existing methods available to discover neighbourhood and co-location pattern approaches. Section 3 describes the overview of co-location analysis related concepts, where the method's event centric approach and N-most prevalent co-location pattern approach were explained with sample data. Section 4 discusses proposed multiple window-based model approach for co-location pattern analysis of the algorithm. Section 5 deals the implementation and result of spatial co-location patterns. Section 6 summarises the performance analysis and comparison our approach with the existing methods and Section 7 discusses the conclusions and future enhancements of the proposed system.

2 Related work

Discovering co-location patterns in the literature were categorised into three classes, namely spatial statistics, data mining, and the event centric approach (Huang et al., 2004). Spatial statistics-based approaches use measures of spatial correlation such as cross-K function with Monte Carlo simulation and quadrant count analysis to characterise the relationship between different types of spatial features. Computation costs of spatial correlation measures are more expensive due to the exponential number of candidate subsets in a large collection of spatial Boolean features. Data mining approach is further divided into a clustering-based map overlay approach and association rule-based approach. Association rule-based approach is divided into transaction-based approach and distance-based approach. Association rule-based approach uses an apriori algorithm (Agarwal and Srikant, 1994), for the creation of transactions over space to generate co-location patterns. Transactions over space can use a reference-feature centric (Koperski and Han, 1995) approach or a data-partition approach (Morimoto, 2001). In reference feature centric model, the transactions are created around instances of one user user-specified spatial feature. Data-partition approach defines transactions by dividing spatial datasets into disjoint partitions. There may be many distinct ways of partitioning the data, each yielding a distinct set of transactions, which in turn yields different values of support of a given co-location. A clustering-based map overlay approach (Estivill-Castro and Lee, 2001), treats every spatial attribute

as a map layer and considers spatial clusters of point-data in each layer as candidates for mining associations. Neighbourhood-based clustering (Zou et al., 2005) which discovers clusters, based on the neighbourhood characteristics of data.

Prevalence measures and conditional probability measures, called interest measures, are defined differently in different models of co-location mining (Salmenkivi, 2006). A distance-based approach which is also called k-neighbouring class sets. In this the number of instances for each pattern is used as the prevalence measure, which does not possess an anti-monotone property by nature. The reference feature centric and data partitioning models materialise transactions and thus can use traditional support and confidence as measures. The event centric approach defined new transaction free measures called the participation index (Shekhar and Huang 2001), which posses the desirable anti-monotone property. Co-location pattern mining general approach formalised the co-location problem and showed the similarities and differences between the co-location rules problem and the classic association rules problem as well as the difficulties in using traditional measures (e.g., support, confidence) created by implicit, overlapping and potentially infinite transactions in spatial datasets. User-specified proximity neighbourhoods (Qian et al., 2009; Celik et al., 2008) is used in place of transactions to specify groups of items and defined interest measures that are robust in the face of potentially infinite overlapping proximity neighbourhoods. A novel Joinless approach (Yoo and Shekhar, 2006) for efficient co-location pattern mining uses an instance-lookup scheme instead of an expensive spatial or instance join operation for identifying co-location instances. Multi-layer index and neighbour domain set (Wang et al., 2006) techniques were used generate co-location patterns for continuous data.

Mining co-location patterns with rare spatial features (Huang et al., 2006) proposes a new measure called the maximal participation ratio (maxPR) and shown that a co-location pattern with a relatively high maxPR value corresponds to a co-location pattern containing rare spatial events. The spatial neighbour relationships and the size-2 prevalence co-locations are compressed into extended prefix-tree structures in the novel order-clique-based (Wang et al., 2009) approach which mine candidate maximal co-locations and co-location instances efficiently. Two algorithms DF-NMColoc and BF-NMColoc were used for finding N-most prevalent co-location patterns (Wan and Zhou, 2008; Yoo and Bow, 2009), where N is the desired number of co-located event sets with the highest interest measure values per each pattern size. Mine top-k closed method (Yoo and Bow, 2011) is used to discover compact co-location patterns without minimum prevalence threshold specified by user.

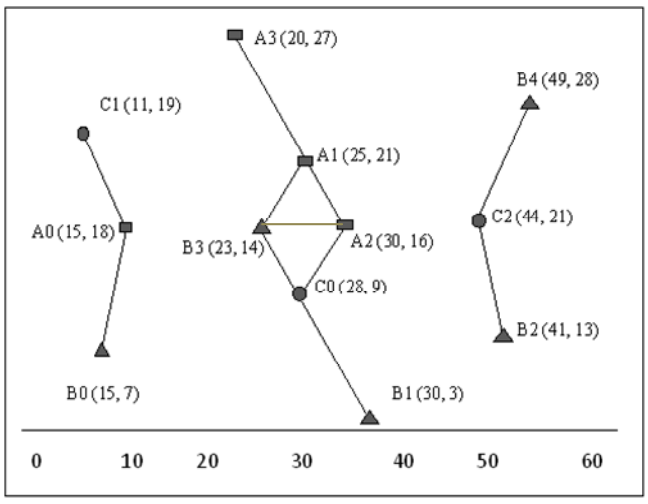
3 Overview of co-location analysis and related concepts

3.1 Event centric approach

Co-location patterns are discovered by using any one of the model such as the reference feature centric model, window centric model and event centric model. The prevalence measure and the conditional probability measure are called interesting measures used to determine useful co-location patterns from the spatial data. The interesting measures are defined differently in different models. We considered Figure 2 as an example spatial dataset to illustrate the event centric model. In the figure, each instance is uniquely identified by T: I, where T is the spatial feature type, and I is the unique id inside each spatial feature type. For example, B: 2 represent the instance 2 of the spatial feature B. Two instances are connected by edges if they have a spatial neighbour relationship. A co-location rule is of the form: c1 => c2 (p, cp), where c1 and c2 are co-locations, c1 ∩ c2 = Ø, p is a number representing the prevalence measure, and cp is a number measuring conditional probability. An important concept behind this method is proximity neighbourhood. User will provide a relationship between its types. Then based on the relationship the instances of different types are connected as shown in Figure 2. Now based on neighbourhood relationship, clique formation is checked. For e.g., in Figure 2, A0 instance is in the neighbourhood of B0 and C1. After getting all the relationship between all the instances, participation of each type in a relationship is calculated. The participation ratio pr(c, f_i) for feature type f_i in a size-k co-location C (f_1, …, f_k).

Participation ratio = No. of instances of fi in the relationship / Total no of instances of fi.

Figure 2 Sample spatial data



For e.g., participation ratio of A in A – B relationship is 3/4. As three instances of A, i.e., A0, A1, A2, out of total four participate in A-B relationship. The participation index pi (cp) of a co-location C (f1 … fk) is $\min_{i=1}^{k}(pr(c, f_i))$.

The participation index is used as the measure of prevalence of a co-location.

For e.g., participation index of AB is min (pr (A), pr (B)), i.e., min (3/4, 2/5), therefore pi of AB is 2/5.

Methodology

Step 1 Initialisation: user provides the neighbourhood relationship between the objects.

Step 2 For k in (2, 3, …, K – 1) and prev. co-location found do

1 Generate size k candidate co-locations
2 Check for cliques and generate table instances
3 Calculate prevalence and select prevalent co-locations
4 Generate co-location rules of size k.

Table 1 Event centric co-location patterns

(a) Patterns where K = 1					
A		*B*		*C*	
0		0		0	
1		1		1	
2		2		2	
3		3			
		4			
Pr 1		1		1	
(b) Patterns where K = 2					
A	*B*	*B*	*C*	*A*	*C*
0	0	1	0	0	1
1	3	3	0	2	0
2	3	4	2		
Pr 0.75	0.4	0.6	0.67	0.5	0.67
PI	0.4	0.6	0.5		
(c) Pattern where K = 3					
A		*B*		*C*	
2		3		0	
Pr 0.25		0.2		0.33	
PI				0.2	

Event centric approach co-location patterns are shown in Table 2 for various size k. first one candidate pattern is generated, and then two candidate patterns are generated. Neighbourhood instances are identified based on the clique formation between the event instances. Table 2(b) shows the candidate sets based on the cliques between instances of A-B, B-C, A-C. In the event A-B, there are three cliques A0-B0, A1-B3, A2-B3. Therefore, Pr of both A and B in A-B is calculated as 0.75 and 0.4, respectively. And PI of A-B is calculated as min (0.75, 0.4), i.e., 0.4. This is compared with the prevalent threshold given by the user.

When k = 3, Size-3 candidates are generated as shown in Table 2(c). In size-3 candidates, only one clique is formed, i.e., A2-B3-C0. The Pr of A, B and C in A-B-C is 0.25, 0.2 and 0.33, respectively. Therefore, value of Pi is 0.2 as its minimum in all the Pr's. As in the example, we do not have cliques of size-4. So algorithm stops with output as all the prevalent co-location patterns.

Table 2 N-most prevalent co-location patterns

(a) Neighbourhood transactions		
S. no.	*Items (neighbour objects)*	
1	A0	B0,C1
2	A1	B3
3	A2	B3,C0
4	B0	A0
5	B1	C0
6	B3	A1,A2,C0
7	B4	C2
8	C0	A2,B1,B3
9	C1	A0
10	C2	B4
(b) Event neighbourhood transactions		
S. no.	*Items (neighbour objects)*	
1	A	B,C
2	A	B
3	A	B,C
4	B	A
5	B	C
6	B	A,C
7	B	C
8	C	A,B
9	C	A
10	C	B
(c) Candidate generation outcomes		
Star candidates		*Co-located candidates*
K = 2		
A B: 3/3	A C:2/3	A B: 1/4
B A: 2/4	B C:3/4	A C: 2/3
C A: 2/3	C B: 2/3	B C: 2/3
K = 3		
A B C: 2/3	C A B: 1/3	A B C: 1/4
B A C: 1/4		

3.2 N-most prevalent co-location patterns approach

Most studies of spatial co-location mining require the specification of two parameter constraints to find interesting co-location patterns. One is a minimum prevalent threshold of co-locations, and the other is a distance threshold to

define spatial neighbourhood. However, it is difficult for users to decide appropriate threshold values without prior knowledge of their task-specific spatial data. To improve on the first constraint, the problem of finding N-most prevalent co-located event set was introduced where N is the desired number of co-located event sets with the highest interest measure values per each pattern size. If the prevalence threshold is set too high, there may be only a small number of result sets or even no result. If the threshold is too low, too many result patterns can be generated with an exceedingly long computational time. They make the analysis of the discovered patterns impractical and even useless. So, to improve it N-most prevalent co-located event set was introduced. In particular, the task of mining N-most prevalent co-location patterns from a spatial dataset is to find N co-located event sets with the highest participation index values per each pattern size. For example, in the process of Figure 2, if N is 2, the N-most prevalent mining, spatial co-location patterns co-located event sets of size 2 is {B, C} and {A, C} since they have higher participation index values than other event sets, {A, B}.

K-co-located event set is a co-location containing k event types.

The N-most prevalent k-co-located event sets: Let L be a list of all k-co-located event sets by descending their participation index values, and let p be the participation index of the Nth k-co-located event set in the list L. The N-most prevalent k-co-located event sets is a set of k-co-located event sets having a participation index ≥ p.

The N-most prevalent co-location patterns are the union of the N-most prevalent k-co-located event sets for $2 \leq k \leq$ kmax, where kmax is the maximum size of co-location patterns.

Given spatial objects $o_i \in S$, the neighbourhood transaction of o_i is defined as set of spatial objects $\{o_i, o_j \in S \mid R(o_i, o_j) = \text{true} \wedge o_i\text{'s event type} \neq o_j\text{'s event type}\}$, where R is neighbour relationship. For example, in Figure 2, C0 has neighbour relationships with each A2, B1 and B3. The neighbourhood transaction of C0 is {A2, B1, B3} including itself as shown in Table 2(a).

Given spatial objects $o_i \in S$, the event neighbourhood transaction of o_i is defined as set of spatial objects $\{o_i\text{'s event type, distinct event type } o_j \in S \mid R(o_i, o_j) = \text{true} \wedge o_i\text{'s event type} \neq o_j\text{'s event type}\}$, where R is neighbour relationship.

For example, in Figure 2, C0 has neighbour relationships with each A2, B1 and B3. The event neighbourhood transaction of C is {A, B} including itself as shown in Table 2(b).

Methodology

Step 1 Preprocess

1 Initialisation: user provides the neighbourhood relationship.

2 ST = gen_neighbour_transactions(S, R)

Step 2 Candidate generation

1 for i = 1 to m do

2 Tree i = build_CP-tree (fi, ST)

3 end do

4 C = gen_candidates(Tree1, . . . , Tree m)

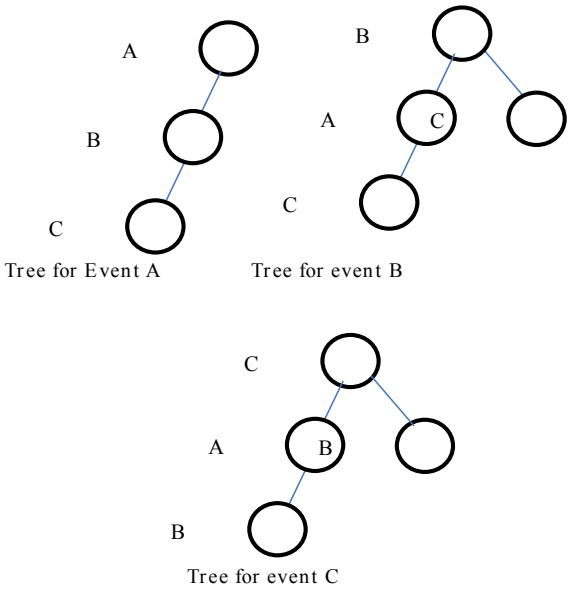
5 calculate_upper_pi(C)

Step 3 Pruning

1 Generate the N-most prevalence co-location patterns.

In preprocessing step, the neighbourhood transactions are generated based on the neighbourhood relationships obtained from the user. Using data from Figure 2 the neighbourhood transactions and event neighbourhood transactions are generated as shown in Tables 2(a) and 2(b). The Cp tree is constructed for each event from the event neighbourhood transactions, as shown in Figure 3. The star candidates are generated for various sizes k by mining Cp tree as shown in Table 2(c). Finally, the co-located candidates are generated from the star candidates who satisfy the prevalence measures. After generating co-location patterns for each size, their upper bound PI is sorted and then pruned on the basis of N value specified by the user. For example, suppose N value specified by the user is 2.

Figure 3 CP tree generation (see online version for colours)



Then N-most prevalent patterns for size 2 are {A C} and {B C}. And for size 3, we have only one co-location pattern {A B C} as shown in Table 2(c).

4 Multiple window-based model approach for co-location pattern analysis

Most studies of spatial co-location mining require the specification of two parameter constraints to find interesting co-location patterns. First, one is a minimum prevalent

threshold of co-locations. The second is a distance threshold to define spatial neighbourhood. However, it is difficult for users to decide appropriate threshold values without prior knowledge of their task-specific spatial data.

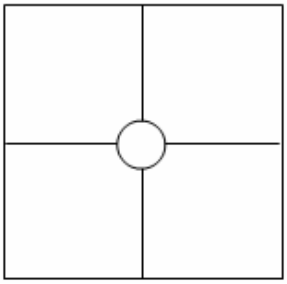
To overcome the above constraints, we have proposed the following solution as our contribution in the area of co-location pattern analysis.

To improve on the first constraint, the problem of finding N-most prevalent co-located event sets was introduced where N is the desired number of co-located event sets with the highest interest measure values per each pattern size. If the prevalence threshold is set too high, there may be only a small number of result sets or even no result. If the threshold is too low, too many result patterns can be generated with an exceedingly long computational time. They make the analysis of the discovered patterns impractical and even useless. So, to improve it N-most prevalent co-located event was introduced.

To improve on the distance neighbourhood constraint the 'window' model is introduced.

In this window-based model, a distance measure is taken from the user, and an Euclidean bounding window is created around that point spatial object. The spatial objects which are within this neighbourhood are said to be in the neighbourhood of that particular spatial object as shown in Figure 4. Window model avoids the relationship specification by the user as an input to the system. For example, in Figure 5, A0 has C1 and B1 in its window area. Therefore, A0 has the neighbourhood relationship with C1 and B1. Another example as B2 has no other object in its window area; therefore, B2 does not have any neighbourhood relationship with other objects. This window model will generate the neighbourhood relationship between the objects which then can be used for further calculations. As in general, event centric approach and N-most prevalent approach, user has to provide the relationship between objects but by using window model user can get the neighbourhood relationship by just provide the size of the window.

Figure 4 Single window



On the extended data-types (i.e., points including lines and polygons) the multiple – window model is introduced.

The range for creating the multiple windows for the spatial dataset is decided dynamically, as the user might not be having the prior knowledge of their task-specific spatial data. So we are proposing the concept of multiple windows. In this method, we do not require the user-specified values to define the window and neighbourhood relationship. A number of windows are generated around each object as shown in Figure 6. Each window defines a unique relationship with another object based on that we proceed. For each window, the algorithm will generate co-location candidates which in turn are pruned to generate co-location patterns. From the output, user can itself decide what is required for the task.

Figure 5 Window model objects illustration

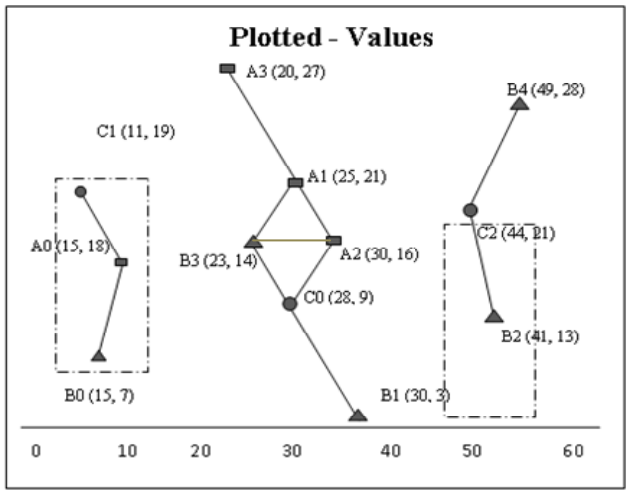
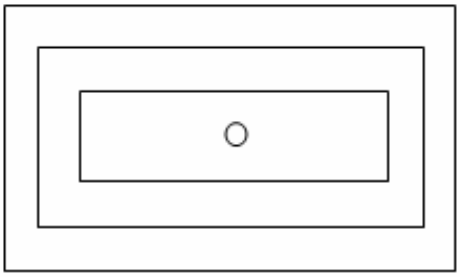


Figure 6 Multiple window



4.1 Modelling neighbourhood

We have used different techniques to define the neighbourhood. We initially apply the 'window'-based system. The window-based system took the input from the user for defining the Euclidean distance around an object. Taking input from the user suffers from a lot of setbacks. Many co-location patterns may not be detected if the user input is too small. We proposed the 'multi-window'-based model, which defines a range of windows for detecting the neighbourhood around an object. This multi-window model does not suffer from the earlier flaw. In geospatial environment, a window is a zone of specified distance around spatial objects. Neighbourhood is being calculated as the boundary should be equidistance from the object. E.g., in case of point object boundary will be a circle, in case of extended spatial objects (such as line or polygon) boundary will be the isoline equidistance to the edges of objects. On the user specified spatial data, multiple-window model is applied to define the neighbourhood relationship objects. In this step, we check the neighbourhood of the spatial objects and based on their attribute values, we decide a

neighbourhood and using a dynamic boundary, we generate the co-location patterns.

Figure 7 Window boundaries around various spatial objects

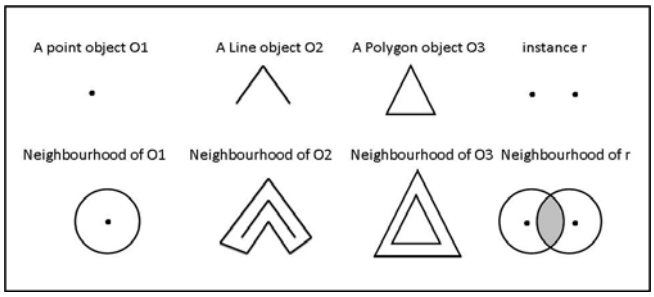


Figure 7 shows a window operation on different types of spatial objects. Objects in space frequently have some sort of impact on the objects and areas around them. Houses near to the highways will have more value in the real estate field.

The Euclidean neighbourhood $G(f_j)$ of a feature f_j is the union of $N(i_l)$ for every instance i_l of the feature f_j.

The Euclidean neighbourhood $G(f_1, f_2 \ldots\ldots f_k)$ for a feature set $C = \{f_1, f_2 \ldots\ldots f_k\}$ is the intersection of $G(f_i)$ for every feature f_i in C.

The support ratio $Pr(f_1, f_2 \ldots\ldots f_k)$ for a feature set $C = \{f_1, f_2 \ldots\ldots f_k\}$ is $G(f_1, f_2 \ldots\ldots f_k)$ / the total area of the plane, where $G(f_1, f_2 \ldots\ldots f_k)$ is the Euclidean neighbourhood of the set C. The support ratio serves as the prevalence measure in this method. The window-based model has a major challenge in dealing with the large number of overlapping operations, which find intersection area among windows of spatial objects through geometric intersections. To reduce the overlay, overlapping should be avoided and that can be done using below formula.

$$\bigcup_{i=1}^{n} BN(A_i) = \sum_{i=1}^{n} BN(A_i) - \sum_{i<j}^{n} BN(A_i A_j) + \sum_{i<j<k}^{n} BN(A_i A_j A_k)$$

$$- \sum_{i<j<k<l}^{n} BN(A_i A_j A_k A_l) + \ldots\ldots + \left((-1)^{n+1}\right) BN(A_1 A_2 \ldots\ldots A_n)$$

The bounding neighbourhood of a spatial object, BN(o) is defined as MBBR(window(MOBR(Spatial Object O),d)) as shown in Figure 7, where MOBR is the minimum object bounding box, window is the window with size d, and MBBR is the minimum window bounding box.

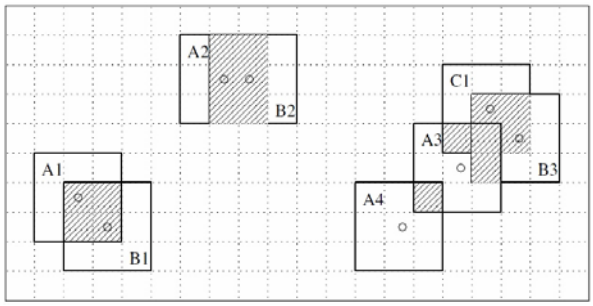
For example, Figure 8 shows eight objects with their bounding neighbourhoods. Four instances of feature A, A1, A2, A3, A4 and only the bounding neighbourhood of A3 has one cell overlapping with the bounding neighbourhood of A4. If we set area the area of a cell to be one unit, the Euclidean bounding neighbourhood BN(A) of feature, A is 4*9 – 1 = 35, which is the union of the bounding neighbourhoods of these four instances.

SPr(A) = BN(A) / total area of the plane = 35 / 200 = 0.175,

SPr(B) = BN(AB) / total area of the plane = 12 / 200 = 0.06

The conditional probability Pr (C2|C1) of a co-location rule s C1->C2 is the probability of finding the neighbourhood of C2 in the neighbourhood of C1. It can be computed as $N(C1 \cup C2) / N(C1)$.

Figure 8 Bounded neighbourhood illustration



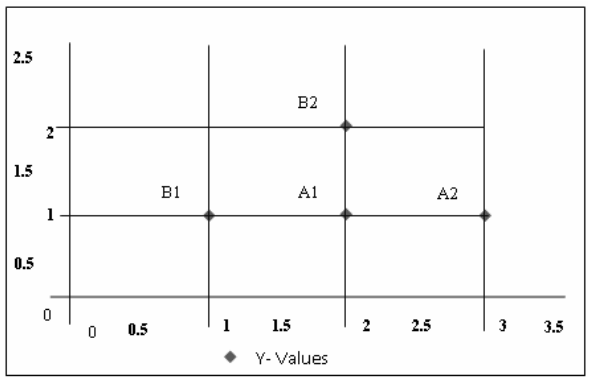
Suppose we have the sample data shown in Figure 9, with two types A and B. And both A and B with two instance each. Let the spatial working area be 8*8.

Therefore, the range of possible for the window is calculated as:

$$Limit = \sqrt{\frac{spatial\ working\ area}{4 * total\ no.\ of\ objects}}$$

Hence, range of the window varies from 1 to limit at integer values. In our example, Figure 9, we have a total four object, therefore limit = 2. So, the range is defined from 1 to 2. Therefore, there would be two neighbourhood relationship between objects one with window size 1 and other with size 2. So for each window size support ratios are calculated to generate the co-location patterns.

Figure 9 Sample dataset for windows-based model

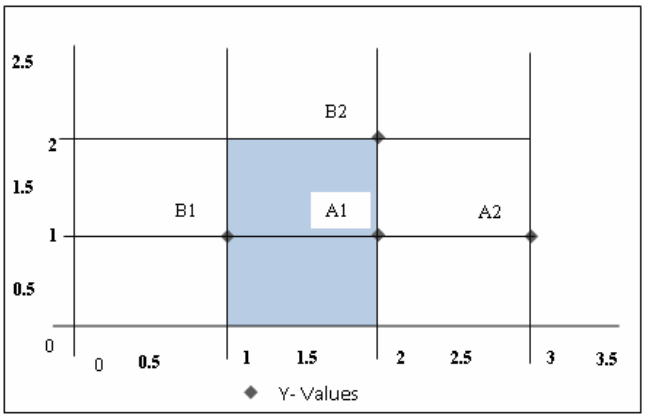


4.2 Candidate co-location generation

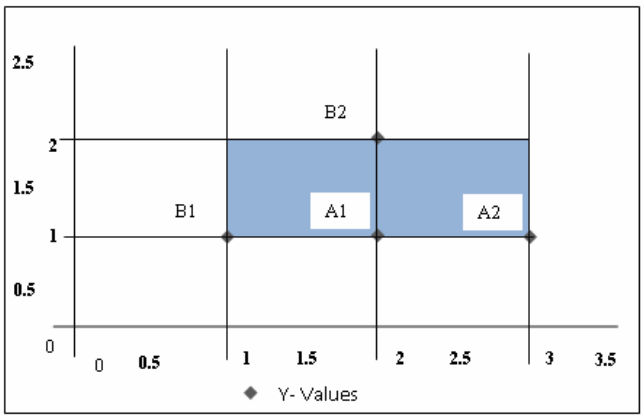
Based on the neighbourhood relationship defined by the multiple windows, for each individual window candidate are generated. All generated candidates are then processed for pruning. After finding the neighbourhood relationship, we have calculated the support ratio for every relationship. In above dataset, we have only one relationship A-B. So we have to calculate the support ratio for A-B.

$$Support\ ratio = BN(AB) \Big/ total\ area\ of\ the\ plane$$

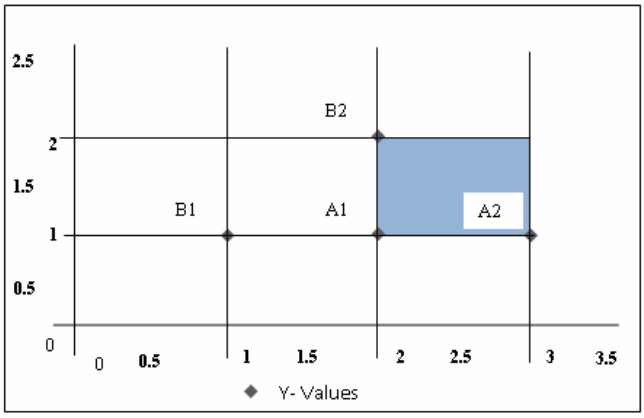
Figure 10 Modelling of neighbourhood using window, (a) intersection between A1 and B1 (b) intersection between A1 and B2 (c) intersection between A2 and B2 (see online version for colours)



(a)



(b)



(c)

And BN of two different type is calculated by area intersection between them. Therefore, for window size 1, the area between A1 and B1 is two units, and the area between A1 and B2 is again two units. And area between A2 and B2 is one unit as shown in Figure 10.

Therefore, BN (AB) is 2 + 2 + 1 = 5 units. So support ratio (AB) is 5/64. Similar way we proceed for window size 2. And support ratio (AB) for window size 2 is 41/64.

4.3 Pruning

After finding support ratio for each relationship, we compare it with the minimum threshold given by the user. Co-location candidates having a support ratio greater than the user-specified threshold value are termed as co-location patterns.

Multiple window based co-location pattern algorithm

Input:

1. A spatial working area 'W' of D1×D2 dimensions.
2. S = {A set of Spatial features of points and lines.}
3. F= {Instance-id, Feature-Type, Location in Space} which would be representing a set of instances of features.
4. A minimum support ratio threshold 'β'
5. A conditional probability limit 'μ' for generating the co-location rules

Output:

1. A set of co-location patterns with coverage ratios greater than the minimum total ratio threshold 'β' which is user-specified.
2. A set of co-location rules with conditional probability greater than the limit 'μ'.

Variables:

1. i = the co-location size.
2. BB_2 = A set of candidate size-2 coarse level co-location patterns.
3. BP_2 = A set of size-2 coarse level co-location patterns having total ratios > 'β'.
4. B_i = A set of candidate size-i co-location patterns.
5. P_i = A set of size-i co-location patterns.
6. T_i = A set of co-location rules derived from size-i co-location patterns.

The geometric filter:

1. Initialisation
2. BB_2 = Pattern_search (S, F, r[])
3. BP_2 = Prevalence_prune(BB_2, β)

Pattern search:

1. Initialisation
2. P_2 = overlay(BP_2, r[]); i=2
3. While(not empty Pi) do {
4. B_{i+1} = generate_candidate_co-location(Pi);
5. P_{i+1} = Prevalence_prune(B i+1, β)
6. R_{i+1} = generate_co-location_rule(μ)
7. i = i + 1 }
8. SAVE: union(P_2,……..P_{i+1})
9. SAVE: union(R_2,……..R_{i+1})

The above multiple window-based co-location pattern algorithm is proposed to discover co-location patterns from different types of spatial data such as point, line and polygon.

5 Implementation and result discussion

Dev-C++ is used for the implementation. It is easy and convenient to use this tool for implementation. Datasets are taken as input from the user and tested on the implemented algorithms. The algorithm was tested with synthetic dataset, and the co-located patterns generated were found out in different cases as shown in Figures 11 and 12.

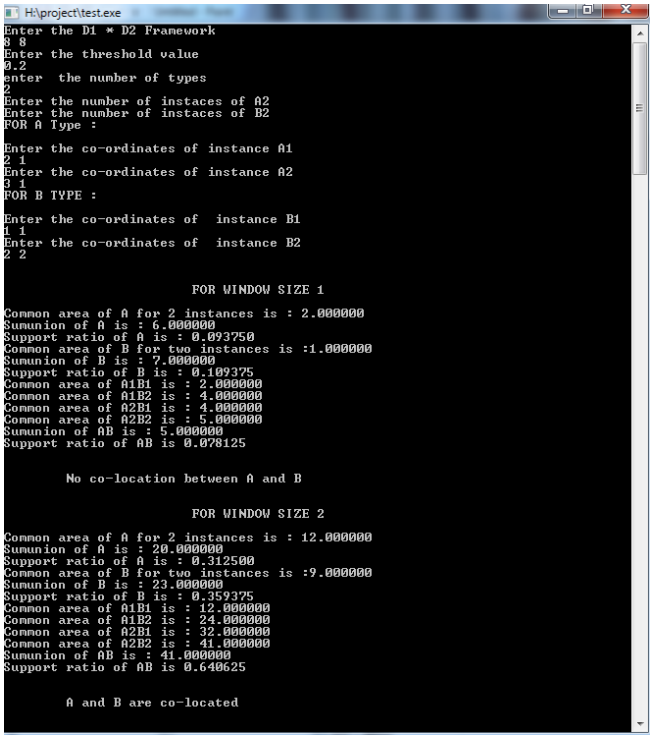
Figure 11 Co-location pattern for single window approach (see online version for colours)

The single window approach generates fewer numbers of patterns, which omit important patterns due to limited neighbours in the window as in Figure 11. The multiple window approach includes important pattern and generates more co-location patterns as shown in Figure 12. The co-located patterns generated were found to be more optimised and dynamic with our proposed methodology.

Considering the fact that user might not be having prior knowledge of datasets, we tried to define the neighbourhood relationship with the more dynamic approach.

In this paper, we proposed algorithm for co-location mining for different types of spatial data objects. We studied the existing techniques and had implemented them. We started with an event centric model where the algorithm faced two major constraints, first is a minimum prevalent threshold of co-locations, and the other is a distance threshold to define spatial neighbourhood. We implemented the N-most prevalent co-location pattern algorithm where the first constraint was improved, such that users can control their interesting patterns in the number of desired patterns. As the user might not be having prior knowledge of their task-specific dataset, we proposed a window model to improve the second constraint in both the above methods. Both the methods are modified and are implemented and tested with window model. Window model is used to provide the distance threshold to define spatial neighbourhood. Then we tried to make it applicable to the extended spatial data objects by introducing the multiple-window model to make it more dynamic in its neighbourhood selection.

Figure 12 Co-location pattern for multiple window approach (see online version for colours)



6 Performance analysis

Here, we compared our proposed approach with existing approach. We also have shown the comparison in Table 3(a) and Table 3(b). We have used the dataset shown in Figure 13 for the result analysis purposes.

Table 3(a) Result comparison between event centric and N-most prevalent approach

Event centric approach	*N-most prevalent co-location patterns*
Inputs:	Inputs:
1 Above dataset.	1 Above dataset.
2 Value of the window = 7	2 Value of the window = 7
3 Min threshold value = 0.2	3 No. of patterns (N) = 2
Output:	Output :
1 A-B	1 For size 2 A-C and B-C are co-located.
2 A-C	2 For size 3 A-B-C are co-located.
3 B-C and A-B-C are co-located.	

Table 3(b) Result comparison between single window approach and multiple window approach

Existing window approach	*Multiple-window model*
Input: • Above dataset • Workspace area (8*8) • Window size (1) • Threshold value (0.4)	Input: • Above dataset • Workspace area (8*8) • Threshold value (0.4)
Output: • A and B are not co-located	Output: • For window size 1: A and B are not co-located. • For window size 2: A and B are co-located.

From the above results [Table 3(a)] we conclude that event centric approach depends on the user-specified threshold value for generating co-location patterns. So, the user must have a prior knowledge of their datasets, whereas, in N-most, prevalent patterns approach the number of patterns generated are controlled by the user. N specifies the number of patterns to be generated for each size.

Comparison for extended data objects.

Figure 13 Sample dataset for result analysis results

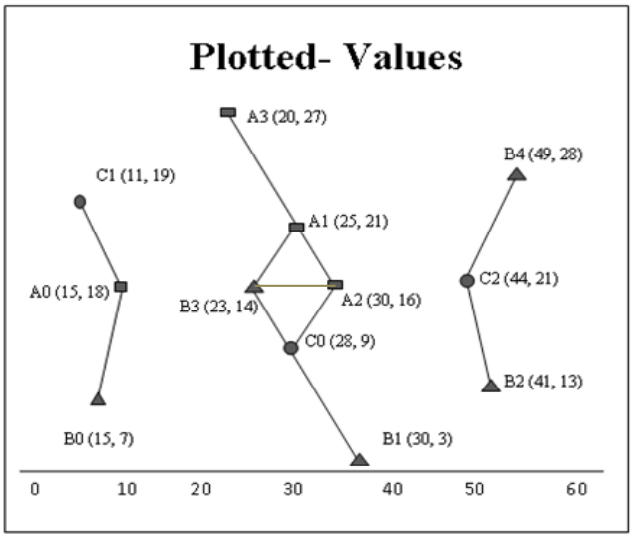


Figure 14 Example dataset for result analysis

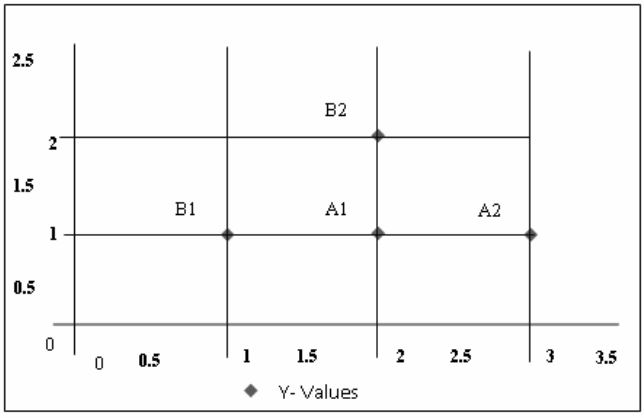
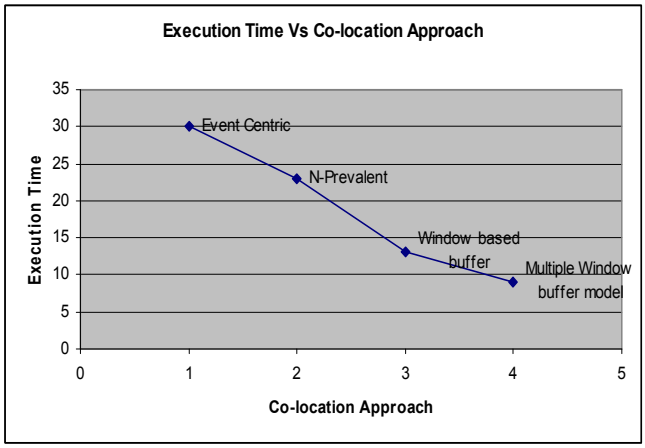
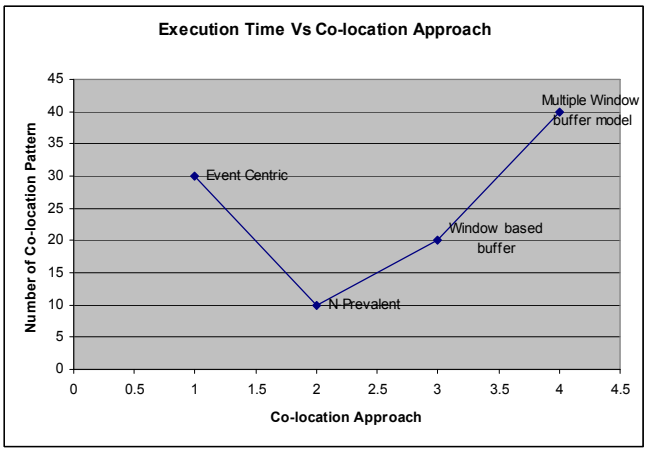


Figure 15 Various co-location approaches performance, (a) execution time vs. co-location approach (b) number of co-location pattern vs. approach (see online version for colours)



(a)



(b)

The above results infer the following:

- In existing window approach, the window value is user defined. So the outcomes are biased in terms of window value. As the number of generated co-location patterns can be very low if a window value is low, whereas in multiple window models, the window value will be a range of values with integer intervals.
- In existing window approach, the number of co-location patterns generated would be less compared to the multiple window models as the multiple window model works on range of window values.
- In existing window approach, the user needs to have a prior knowledge of the existing condition to make a correct estimate for the number of co-location patterns to be generated whereas the multiple window model does have any such requirement. It is more generic.
- Existing window approach is less efficient as compared to the multiple window model because of the window value is user defined, and it may miss out important co-location patterns.

- Number of co-location pattern generation is more in multiple window approach compare than single window approach. The execution time for multiple windows is more than single window and less than an even centric and n-prevalent approach.

7 Conclusions

In this paper, we proposed multiple window-based algorithm for co-location mining for different types of spatial data objects. We have discussed the different co-location pattern analysis algorithms like the event centric approach and N-most prevalent co-location patterns. We implemented the N-most prevalent co-location pattern algorithm where the first constraint was improved, such that users can control their interesting patterns in the number of desired patterns. As the user might not be having prior knowledge of their task-specific dataset, we proposed a window model to improve the second constraint in both the above methods. Both the methods are modified and are implemented and tested with window model. Window model is used to provide the distance threshold to define spatial neighbourhood. Then we tried to make it applicable to the extended spatial data objects by introducing the multiple-window model to make it more dynamic in its neighbourhood selection. Here, we have tested only the Boolean features in co-location pattern generation. In future, the algorithm can be tested for continuous and categorical features.

References

Agarwal, R. and Srikant, R. (1994) 'Fast algorithms for mining association rules', *Proceedings of the 20th VLDB Conference*, pp.487–499, Santiago, Chile.

Celik, M., Shekhar, S., Rogers, J.P. and Shine, J.A. (2008) 'Mixed-drove spatiotemporal co-occurrence pattern mining', *IEEE Transactions on Knowledge and Data Engineering*, Vol. 20, No. 10, pp.1322–1335.

Estivill-Castro, V. and Lee, I. (2001) 'Data mining techniques for autonomous exploration of large volumes of geo-referenced crime data', *Proc. Sixth Int'l. Conf. Geo Computation*, Brisbane, Australia.

Huang, Y., Pei, J. and Xiong, H. (2006) 'Mining co-location patterns with rare events from spatial data sets', *Journal Geoinformatica*, Vol. 10, No. 3, pp.239–260.

Huang, Y., Shekhar, S. and Xiong, H. (2004) 'Discovering colocation patterns from spatial data sets: a general approach', *IEEE Transactions on Knowledge and Data Engineering*, Vol. 16, No. 12, pp.1472–1485.

Koperski, K. and Han, J. (1995) 'Discovery of spatial association rules in geographic information databases', *Proceedings of the 4th International Symposium on Advances in Spatial Databases*, pp.47–66, Springer-Verlag, London.

Morimoto, Y. (2001) 'Mining frequent neighbouring class sets in spatial databases', *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp.353–358, ACM, New York, USA.

Qian, F., He, Q. and He, J. (2009) 'Mining spatial co-location patterns with dynamic neighbourhood constraint', *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases: Part II, LNAI*, Vol. 5782, pp.238–253, Springer-Verlag, Berlin, Heidelberg.

Salmenkivi, M. (2006) 'Efficient mining of correlation patterns in spatial point data', *Proceedings of the 10th European Conference on Principle and Practice of Knowledge Discovery in Databases, LNAI*, Vol. 4213, pp.359–370, Springer-Verlag, Berlin, Heidelberg.

Shekhar, S. and Chawla, S. (2003) *Spatial Databases: A Tour*, Prentice Hall, ISBN 013-017480-7.

Shekhar, S. and Huang, Y. (2001) 'Discovering spatial co-location patterns: a summary of results', *Proceedings of the 7th International Symposium on Advances in Spatial and Temporal Databases*, pp.236–256, Springer-Verlag, London.

Wan, Y. and Zhou, J. (2008) 'Knfcom-T: a k-nearest features-based co-location pattern mining algorithm for large spatial data sets by using T-trees', *International Journal of Business Intelligence and Data Mining*, Vol. 3, No. 4, pp.375–389, Inderscience Publishers.

Wang, L., Zhou, L., Lu, J. and Yip, J. (2009) 'An order-clique-based approach for mining maximal co-locations', *Information Sciences*, Vol. 179, No. 19, pp.3370–3382.

Wang, Z-Q., Chen, H-B. and Yu, H-Q. (2006) 'Spatial co-location rule mining research in continuous data', *Proceedings of the Fifth International Conference on Machine Learning and Cybernetics*, pp.1362–1367, Dalian, China.

Yoo, J.S. and Bow, M. (2009) 'Finding N-most prevalent colocated event sets', *Proceedings of the 11th International Conference on Data Warehousing and Knowledge Discovery*, pp.415–427, Springer-Verlag, Berlin.

Yoo, J.S. and Bow, M. (2011) 'Mining spatial colocation patterns: a different framework', *Data Mining and Knowledge Discovery*, Vol. 24, No. 1, pp.159–194, Kluwer Academic Publishers.

Yoo, J.S. and Shekhar, S. (2006) 'A joinless approach for mining spatial co-location patterns', *IEEE Transactions on Knowledge and Data Engineering*, Vol. 18, No. 10, pp.1323–1337.

Zou, S., Zhao, Y., Guan, J. and Huang, J. (2005) 'A neighbourhood-based clustering algorithm', *Proceedings of the 9th Pacific-Asian conference on Advances in Knowledge Discovery and Data Mining*, pp.361–371, Springer-Verlag, Berlin, Heidelberg.