Knowledge and Information Systems (2021) 63:1365–1395
https://doi.org/10.1007/s10115-021-01559-3

REGULAR PAPER

Efficient discovery of co-location patterns from massive spatial datasets with or without rare features

Peizhong Yang[1] · Lizhen Wang[1] · Xiaoxuan Wang[1] · Lihua Zhou[1]

Received: 2 June 2019 / Revised: 1 March 2021 / Accepted: 10 March 2021 / Published online: 27 March 2021

Abstract
A co-location pattern indicates a group of spatial features whose instances are frequently located together in proximate geographic area. Spatial co-location pattern mining (SCPM) is valuable for many practical applications. Numerous previous SCPM studies emphasize the equal participation per feature. As a result, the interesting co-locations with rare features cannot be captured. In this paper, we propose a novel interest measure, i.e., the weighted participation index (WPI), to identify co-locations with or without rare features. The WPI measure possesses a conditional anti-monotone property which can be utilized to prune the search space. In addition, a fast row instance identification mechanism based on the ordered NR-tree is proposed to enhance efficiency. Subsequently, the ordered NR-tree-based algorithm is developed. To further improve efficiency and process massive spatial data, we break the ordered NR-tree into multiple independent subtrees, and parallelize the ordered NR-tree-based algorithm on MapReduce framework. Extensive experiments are conducted on both real and synthetic datasets to verify the effectiveness, efficiency and scalability of our techniques.

Keywords Spatial data mining · Co-location pattern · Rare feature · Parallel algorithm

1 Introduction

With the advancement of spatial technologies, such as Global Positioning System (GPS) and Remote Sensing Technology, spatial data presents explosive growth trend. These spatial data are considered as the nuggets which conceal valuable knowledge [10]. Mining of the

✉ Lizhen Wang
lzhwang@ynu.edu.cn

Peizhong Yang
pzyang0924@163.com

Xiaoxuan Wang
wangxiaoxuan1037@163.com

Lihua Zhou
lhzhou@ynu.edu.cn

[1] School of Information Science and Engineering, Yunnan University, Kunming 650091, China

interesting, potentially useful and previously unknown knowledge from spatial datasets is the vital goal of spatial knowledge discovery. Spatial co-location pattern mining (SCPM) is one of the spatial knowledge discovery techniques [11,17], and it is intended to capture the dependency relationship among spatial features. A co-location pattern (or co-location) corresponds to a subset of spatial features, whose instances are frequently located in close geographic proximity. For instance, Flower Store, Hospital and Drugstore constitute a co-location, as they constantly appear together in the proximity. SCPM is a significant field of research, and it may yield crucial insights in numerous practical applications [11,12, 17,27,32]. Ecologists can discover the symbiotic relationships among different species via SCPM [15]. Co-locations extracted from the mature city can be employed to plan layouts or arrange new facilities in the construction of new cities [28]. Other application domains include transportation, public health, ecological protection, etc.

Studies on SCPM commonly use the participation index as the interest measure, which quantifies the frequency that instances of features in a co-location are located closely. However, this interest measure emphasizes frequent co-occurrences of all features involved in a pattern, which makes some valuable co-locations including rare features missed. A feature is rare if the number of its instances is obviously less than that of other features. {Tricholoma Matsutake, Pine} is a realistic example of the co-locations with rare features. Tricholoma matsutakes are strongly co-located with pines, but this co-location cannot be reported due to the low participation index, since there is an unequal state in the quantity of instances for tricholoma matsutake and pine. In general, tricholoma matsutake is rare in the real world but pine is universal. In some cases, we are more interested in co-locations with rare features owing to the attention of rare features, e.g., the protection of cherish animals and plants.

To discover co-locations with rare features, an interest measure namely the maximal participation ratio (maxPR) was proposed [10]. However, some non-prevalent patterns where there is no rare feature included may be reported by maxPR. Afterward, Feng et al. [8] defined the concept of rare ratio and proposed the minimum weighted participation ratio based on rare ratio (WP-RR) as the interest measure to discover co-locations with or without rare features. However, WP-RR measure requires an user-specified minimum rare ratio threshold, and it is difficult for users to give a suitable threshold. Moreover, both maxPR and WP-RR measures are not efficient due to the weak monotonicity property of them. The details can refer to the part of related work.

In our previous work [25], we proposed the degree of dispersion to assess the difference in the quantity of instances for features in the spatial dataset. We continue that work, and propose the rare intensity in this paper to quantify the rare strength of features in a pattern. The rare intensity considers the degree of dispersion about the quantity of instances for features in the whole spatial dataset and that in a co-location at the same time. Considering the rare intensity and prevalence of features in a pattern, we propose a novel interest measure called the weighted participation index (WPI), which can discover co-locations with or without rare features; moreover, WPI satisfies a conditional anti-monotone property that can be used to prune the search space efficiently. Therefore, WPI measure can address the aforementioned issues existing in maxPR and WP-RR approaches. The following contributions are made by this paper.

1. The rare intensity based on the Gaussian kernel function is proposed to capture the rare degree of features in a co-location. Then, WPI is proposed as the interest measure, and we analyze the mechanism of WPI measure to discover co-locations with or without rare features. In addition, we prove the conditional anti-monotone property of WPI.

2. An efficient algorithm is developed, namely the ordered NR-tree-based algorithm, which employs the conditional anti-monotone property of WPI to prune the search space and utilizes the mechanism of extending previous table instances to accelerate the calculation of WPI. Moreover, we design the ordered NR-tree to materialize neighbor relationships and speed up the identification of row instances. The complexity, completeness and correctness of the proposed algorithm are discussed.
3. For enhancing efficiency further and processing massive spatial data, we parallelize the ordered NR-tree-based algorithm. The ordered NR-tree is broken down into multiple independent ordered NR-subtrees so that the mining task can be executed on each subtree in parallel. Based on the ordered NR-subtree, a novel pruning strategy is proposed. Furthermore, the parallel ordered NR-tree-based algorithm is implemented on MapReduce framework.

Substantial experiments are conducted on real world and synthetic spatial datasets to verify the superiority of WPI measure on the discovery of co-locations with or without rare features and the better efficiency of the proposed algorithms. Notably, experimental results show that the parallel ordered NR-tree-based algorithm has a great improvement in performance compared to the serial version, and it is more scalable to massive spatial data.

The remainder of this paper is structured as follows. Section 2 provides the overview of the basic concepts of SCPM, reviews related work, and discusses the motivation and challenges of this work. Section 3 presents WPI measure. The ordered NR-tree-based algorithm is described in Sect. 4 and its parallel version is presented in Sect. 5. Section 6 shows experimental evaluations, and the paper is concluded in Sect. 7.

2 Preliminary

2.1 Spatial co-location pattern mining

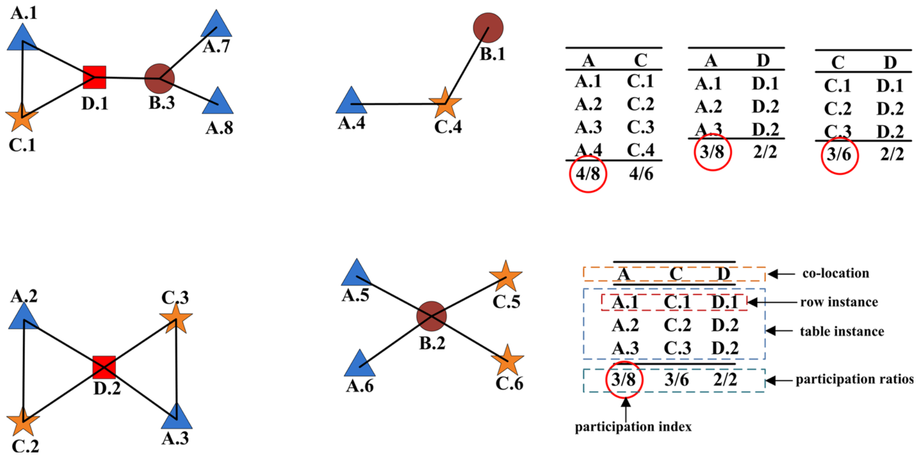
Notations. In a spatial dataset D, let F be the set of spatial features, O be the set of spatial instances of F and each instance is a three-ingredient tuple ⟨instance id, feature type, location⟩, and R be the neighbor relationship over pairwise instances. For ease of illustration, we use the Euclidean distance in this paper, and two instances are neighboring if the distance between them is no larger than a specified distance threshold d, i.e., $R(o, o') \Leftrightarrow distance(o, o') \leq d, o \in O, o' \in O$. Given a set of instances $S \subset O$, S is a **clique** if $R(i, j)$ holds for every pairwise instances $i \in S$, $j \in S$. A **co-location** C is a subset of features, $C \subseteq F$, whose instances frequently form cliques. The **size** of a co-location C is the number of features in C. A clique I is a **row instance** (or **co-location instance**) of C, if I includes instances of all features in C and no proper subset of I does so. The collection of all row instances of C is its **table instance**, denoted as $T(C)$. To capture the prevalence of a co-location, Huang et al. [11,17] defined the participation index. The **participation index** (PI) of a $k (k \geq 2)$ size co-location $C = \{f_1, \ldots, f_k\}$ is defined as $PI(C) = \min_{i=1}^{k} \{PR(f_i, C)\}$, where $PR(f_i, C)$ is the **participation ratio** (PR) of feature f_i in C, defined as

$$PR(f_i, C) = \frac{\text{number of distinct instances of } f_i \text{ in } T(C)}{\text{number of instances of } f_i} \tag{1}$$

$PR(f_i, C)$ reveals the probability that instances of feature f_i participate in C. A co-location C is prevalent, if $PI(C) \geq min\_prev$, where min_prev is an user-specified **minimum prevalence threshold**.

Table 1 Notation table

Notation	Definition
F	The set of spatial features
O	The set of spatial instances
o	A spatial instance with its id $o.i$, feature type $o.t$ and location $o.l$
d	The distance threshold
C	A co-location
k	The size of co-location
I	A row instance
$PR(f_i, C)$	The participation ratio of feature f_i in co-location C
$PI(C)$	The participation index of co-location C
min_prev	The minimum prevalence threshold



A	C
A.1	C.1
A.2	C.2
A.3	C.3
A.4	C.4
4/8	4/6

A	D
A.1	D.1
A.2	D.2
A.3	D.2
3/8	2/2

C	D
C.1	D.1
C.2	D.2
C.3	D.2
3/6	2/2

A	C	D
A.1	C.1	D.1
A.2	C.2	D.2
A.3	C.3	D.2
3/8	3/6	2/2

Fig. 1 An example spatial dataset and the table instances of co-locations {A, C}, {A, D}, {C, D} and {A, C, D}

The main notations used throughout the paper are summarized in Table 1.

Problem Definition. Given a spatial dataset with F, O, d, and min_prev, the objective of spatial co-location pattern mining is to discover all prevalent co-locations.

Example 1 Figure 1 shows an example spatial dataset with four features (i.e., A, B, C, D). Each instance is marked by its feature type and id, e.g., A.1 is the first instance of A. Two instances connected by a black line have the neighbor relationship R. We list the table instance for co-locations {A, C}, {A, D}, {C, D} and {A, C, D}, and calculate the participation ratio for features involved in each co-location, e.g., the participation ratio of C in {A, C, D} is $\frac{3}{6}$, since only three (C.1, C.2, C.3) out of six instances of C are included in the table instance of {A, C, D}. The participation index per co-location is marked by a red circle, e.g., the participation index of {A, C, D} is $\frac{3}{8}$. If $min\_prev = 0.5$, {A, C} and {C, D} are prevalent co-locations, but {A, D} and {A, C, D} are not.

Lemma 1 *The participation ratio and participation index are anti-monotone as the size of co-locations increases.*

That is, given two co-locations C and C', $C' \subset C$, if feature $f \in C$, $f \in C'$, $PR(f, C') \geq PR(f, C)$ holds, and naturally, $PI(C') \geq PI(C)$. The related proof can refer to [11,17]. Utilizing such a property, if co-location C is not prevalent, all super patterns of C must be non-prevalent. Thus, the anti-monotone property can be exploited to prune the search space of co-locations and enhance efficiency.

It is vital to note that some interesting co-locations involving features with a substantial difference in the quantity of instances may be missed by the participation index measure.

Example 2 Consider the example in Fig. 1, feature A has eight instances but feature D only has two instances. The number of instances for these two features has the unequal status. Even though all instances of D are surrounded by that of A, pattern $\{A, D\}$ is still non-prevalent as its participation index is only $\frac{3}{8}$ and less than $min\_prev = 0.5$.

2.2 Related work

The discovery of co-locations has become a research hot and been studied extensively. Approaches to identify co-locations can be divided into two categories, the statistics-based approach and the data mining-based approach. The statistics-based approaches apply statistical tools (e.g., cross-K function with Monte Carlo simulation) to characterize the relationship between distinct features [4,5]. However, the statistics-based methods are computationally expensive since the number of candidate patterns is exponential. Benefited from the high computational efficiency, the data mining-based approaches have attracted considerable attentions. SCPM was introduced by Shekhar and Huang [11,17]. In their work, the participation index was defined to measure the prevalence of a co-location, and an *Apriori-like* approach called the join-based algorithm was proposed to discover co-locations. Subsequently, some improved algorithms were presented, such as the partial-join approach [30], the join-less approach [31], the CPI-tree approach [22], and the iCPI-tree approach [23]. Moreover, some MapReduce-based [24,26,29] and GPU-based [1–3] parallel algorithms were developed for processing massive spatial data.

Some other studies related to SCPM task are reviewed as follows. Lu et al. [14,15] researched interesting relationships of features embedded in a co-location, such as symbiotic, competitive and causal relationships. In [28], a kernel-density-estimation-based model was presented to mine co-locations with the consideration of distance decay effects. Ge et al. [9] presented a framework to discover co-locations from extended spatial objects, e.g., polygons and lines. In [5], the problem of mining regional co-locations was concerned. Ouyang et al. [16] studied to identify co-locations from fuzzy objects. In [6], authors proposed a new support measure to address the problem that the contribution of objects is over-counted in the participation-based approach when multiple instances overlap. In [7], authors focused on mining the dominant-feature co-locations, which consider not only the prevalence but also the dominant role of features playing in a co-location. Wang et al. [21] proposed a new concept of sub-prevalent co-locations to address the issue that the participation index only considers clique instances and may overlook some important spatial correlations. Some models [13,19,20] were proposed to produce condensed co-locations without loss of information, so that the less and easy to understand co-locations are provided to users. To satisfy user preferences, an interactive probabilistic post-mining model was presented in [18] to discover the user-preferred co-locations.

The above methods have not considered the difference in the quantity of instances of features within a co-location, and thus some co-locations involving rare features are missed unfortunately. To discover such co-locations, Huang et al. [10] defined the maximal participation ratio (maxPR) as the interest measure. Distinct from PI measure, maxPR is the largest participation ratio of features in a co-location. Taking advantage of maxPR measure, co-locations with rare features can be identified. The judging condition of prevalent co-locations in maxPR measure is loose. A co-location is considered as prevalent by maxPR, as long as there is one involved feature with the participation ratio no less than min_prev. For this reason, some non-prevalent patterns may also be reported by maxPR, especially when there is no rare feature involved. Moreover, maxPR approach is not efficient, because it only satisfies a weak monotonicity property and so it can not efficiently prune the search space like PI approach. Thereafter, Feng et al. [8] defined the concept of rare ratio and used an user-specified rare ratio threshold t to clarify the concept of rarity, then they proposed the minimum weighted participation ratio (WP-RR) as the interest measure. Co-locations with or without rare features can be identified by WP-RR, but the effectiveness of this approach depends on the specified rare ratio threshold t. A smaller t may cause that some co-locations with rare features are missed, but some non-prevalent patterns without rare features may be reported inappropriately by a larger t. Thus, it is vital for users to set a suitable rare ratio threshold for WP-RR approach. As WP-RR just possesses a partial anti-monotonic property, this approach also faces the problem of low efficiency. To the best of our knowledge, only above two methods are systematic studies for mining co-locations with rare features.

2.3 Challenges

As can be seen from the above discussion, the problem of mining co-locations has still not been solved well when there is a huge difference in the quantity of instances for features in the spatial dataset. In this work, we commit to developing a novel approach to discover prevalent co-locations with or without rare features. To develop such an approach, the following issues should be considered. (1) The rareness is a fuzzy concept. A feature may be rare in a co-location but not rare in the others, so the problem that how to measure the rare intensity of features in a co-location should be addressed. (2) A novel interest measure which considers the prevalence and rare intensity of features within a co-location simultaneously is needed, and such an interest measure should not only identify co-locations with or without rare features but also do not increase the burden on users. (3) It is promising to investigate some properties of the new interest measure that can be used for enhancing efficiency, and develop a highly efficient algorithm. (4) Lastly, it is also necessary to study how to adapt the new algorithm to massive spatial data.

3 Novel interest measure

In this section, we present a rareness measure to evaluate the rare intensity of features within a co-location, and propose the weighted participation index measure. In addition, we investigate the properties of weighted participation index.

3.1 Rareness measure

The rarity of features is a relative concept, which should be related to the degree of dispersion about the quantity of instances for features in the whole spatial dataset and that in a co-location.

Definition 1 Let $F = \{f_1, f_2, \ldots, f_m\}$ be the set of all features in a spatial dataset D, and $num(f_i) \leq num(f_j)$ holds for every $1 \leq i < j \leq m$, where $num(f_i)$ is the number of instances of feature f_i. The degree of dispersion about the quantity of instances for features in D is defined as

$$\delta = \frac{2}{m(m-1)} \sum_{i<j} \frac{num(f_j)}{num(f_i)} \tag{2}$$

For feature pair (f_i, f_j), $j > i$, if the ratio $\frac{num(f_j)}{num(f_i)}$ is larger, the number of instances of f_j is obviously more than that of f_i; otherwise, if the ratio is close to 1, there is an equal status in the quantity of instances for features f_i and f_j. The degree of dispersion is the average value of all ratios for such feature pairs, and it was introduced in our previous work [25]. Given a spatial dataset, $\delta \in [1, +\infty)$ is a constant since $num(f_j) \geq num(f_i)$ holds for every $1 \leq i < j \leq m$ in $F = \{f_1, f_2, \ldots, f_m\}$. The bigger δ value is, the larger the difference in the quantity of instances for features in D is. Notably, $\delta = 1$ means that the number of instances for all features are equivalent.

Definition 2 Given a co-location $C = \{f_1, \ldots, f_k\}$, $k \geq 2$, $f_{\min}$ is the feature with the fewest number of instances in C. The degree that the quantity of instances of feature $f_i (1 \leq i \leq k)$ deviates from that of feature $f_{\min}$ in C is defined as

$$v(f_i, C) = \frac{num(f_i)}{num(f_{\min})} \tag{3}$$

In co-location C, $f_{\min}$ is considered as the rarest feature, and the rare intensity of other features decreases as the number of instances increases. In this work, we adopt the Gaussian kernel function to describe the rare strength of a feature in a co-location. The Gaussian kernel function is the most commonly used radial basis function, defined as

$$K(x) = \exp\left(-\frac{(x-c)^2}{2\sigma^2}\right) \tag{4}$$

where $\exp(\cdot)$ is an exponential function with the base of the natural constant e. The Gaussian kernel function is a bell curve [28], c is the center of the kernel function to determine the location of the peak value in the horizontal axis; σ is the bandwidth parameter to control the radial range of the kernel function, in other words, it determines the width of the curve. When the distance between x and c is within a certain range, the value of the Gaussian kernel function varies dramatically with the change of x, and the value will be smaller if the distance is larger. In our model, let x be the degree that the quantity of instances of f_i deviates from that of $f_{\min}$ in co-location C, i.e., $x = v(f_i, C)$; $c = v(f_{\min}, C) = 1$ be the center; and σ be the degree of dispersion about the quantity of instances for features in the whole spatial dataset, i.e., $\sigma = \delta$.

Definition 3 The rare intensity of feature f_i, $1 \leq i \leq k$, in co-location $C = \{f_1, \ldots, f_k\}$ is defined as

$$RI(f_i, C) = \exp\left(-\frac{\left(v(f_i, C) - 1\right)^2}{2\delta^2}\right) \tag{5}$$

$RI(f_i, C)$ takes into account both the degree of dispersion about the number of instances for all features in the whole spatial dataset (i.e., δ) and the degree to which feature f_i is off center in C (i.e., $v(f_i, C) - 1$). It is not difficult to know that the rare intensity satisfies following properties, (1) the rare intensity of feature $f_{\min}$ in C equals to 1; (2) the rare intensity of any feature in C falls in (0, 1]; (3) in co-location C, the larger the number of instances of feature f_i has, the lower the rare intensity of feature f_i is.

With the help of above definitions, the rare intensity of any feature in a co-location can be measured.

Example 3 Consider the example spatial dataset in Fig. 1 again, feature A has eight instances, feature B has three instances, feature C has six instances, and feature D has two instances, so D < B < C < A sorted by the number of instances in ascending order. Then, we have $\delta = 2.42$ calculated by Formula (2). In co-location {A, C, D}, the rarest feature $f_{\min}$ is D. The degree that the number of instances of A diverges from that of $f_{\min}$ in {A, C, D} is v(A, {A, C, D})$=\frac{num(A)}{num(D)} = \frac{8}{2} = 4$ by Definition 2, and then the rare intensity of feature A is RI(A, {A, C, D})$=\exp\left(-\frac{(4-1)^2}{2*2.42^2}\right) = 0.46$ by Formula (5). Similarly, we have RI(C, {A, C, D}) $= 0.71$, RI(D, {A, C, D}) $= 1$.

3.2 Weighted participation index

In this part, we present a novel interest measure which considers both the rare intensity and prevalence of features in a co-location. For a co-location including features with a significant difference in the quantity of instances, even though the participation ratio of the rare feature is high, its participation index may be low due to the smaller participation ratio of other features. Based on this observation, we extend the participation ratio with the consideration of the rare intensity of features.

Definition 4 Given a $k(k \geq 2)$ size co-location $C = \{f_1, \ldots, f_k\}$, the weighted participation ratio (**WPR**) of feature $f_i (1 \leq i \leq k)$ in C is defined as

$$WPR(f_i, C) = PR(f_i, C) * w(f_i, C) \tag{6}$$

where $w(f_i, C)$ is the relative weight of feature f_i in C, associated with its rare intensity in C, $w(f_i, C) = \frac{1}{RI(f_i,C)}$, $w(f_i, C) \in [1, +\infty)$.

Definition 5 For co-location $C = \{f_1, \ldots, f_k\}$, the weighted participation index (**WPI**) of C is the minimum weighted participation ratio for all features in C, i.e.,

$$WPI(C) = \min_{i=1}^{k}\{WPR(f_i, C)\} \tag{7}$$

It is easy to know that the WPI metric of a co-location falls in [0, 1]. So, users can set min_prev like the participation index framework, and a co-location C is prevalent if $WPI(C) \geq min\_prev$ holds. For a prevalent co-location C and any feature $f \in C$, we have $WPR(f, C) = \frac{PR(f,C)}{RI(f,C)} \geq min\_prev$, so $PR(f, C) \geq min\_prev * RI(f, C)$, which is equivalent to that the prevalence threshold for different features in C is not equal. (1) If all features in C have small difference in the quantity of instances, we can know that their rare intensity values are close to 1 or equal to 1, so the participation ratio of any feature in C is no less than min_prev. In this case, WPI and PI are almost equivalent to identify prevalent co-locations without rare features. (2) If the difference in the quantity of instances for features in C is substantial, $PR(f_{\min}, C)$ must be no less than min_prev since $RI(f_{\min}, C) = 1$.

For other features, their prevalence threshold, i.e., $min\_prev * RI(f, C)$, is decreased as the rare intensity decreases, which will loose the restriction of the participation ratio for other features, so that co-locations with rare features can be identified. Compared with maxPR, WPI is more strict. If $PR(f_{\min}, C) \geq min\_prev$, C is considered as a prevalent co-location by maxPR, but C may be non-prevalent in WPI measure, since WPI approach also requires that the participation ratio of other features no less than a specific threshold, i.e., $PR(f, C) \geq min\_prev * RI(f, C)$. As a result, co-locations discovered by WPI approach have the stronger dependency relationship. Moreover, WPI measure does not introduce additional thresholds on the basis of traditional SCPM framework, so it is more friendly for users than WP-RR approach.

Example 4 For pattern {A, D} in Fig. 1, it is easy to calculate RI(A, {A, D}) $= 0.46$ and RI(D, {A, D})=1 by Definition 3. So, we have w(A, {A, D}) $= 1 \div 0.46 = 2.17$ and w(D, {A, D}) $= 1$. Besides, we have known that PR(A, {A, D}) $= \frac{3}{8} = 0.375$ and PR(D, {A, D})$=1$. Thus, WPR(A, {A, D}) $= 0.375 * 2.17 = 0.814$ and WPR(D, {A, D}) $= 1$, then WPI({A, D}) $=$ min{0.814, 1} $= 0.814$. If min_prev is still 0.5, {A, D} is considered as a prevalent co-location. Similarly, we have WPI({A, C}) $= 0.504$ and WPI({A, C, D}) $= 0.704$.

3.3 Conditional anti-monotone property

Unfortunately, WPI measure does not satisfy the anti-monotone property. For instance, {A, C, D} is a super pattern of {A, C}, but WPI({A, C, D}) $>$ WPI({A, C}) from Example 4. Thus, all possible combinations of features have to be tested for discovering all prevalent co-locations, which is certainly time-consuming. Next, we investigate the properties of WPI measure to prune the search space.

Lemma 2 *Given a k size co-location $C = \{f_1, \ldots, f_k\}$, $k \geq 2$, $f_{\min} \in C$ is the feature with the fewest number of instances in C, suppose C' is a (k-1) size subset of C, $C' \subset C$, and $f_{\min} \in C'$, if $WPI(C') < min\_prev$, C must be not prevalent.*

Proof Given any feature $f \in C'$, then $f \in C$ holds, and $PR(f, C) \leq PR(f, C')$ according to Lemma 1. Besides, it is easy to know that $RI(f, C) = RI(f, C')$ by Definition 3, since feature $f_{\min}$ is included in both C and C'. Thus, $w(f, C) = w(f, C')$, and then $WPR(f, C) \leq WPR(f, C')$. Further, $WPI(C) \leq WPI(C') < min\_prev$. Therefore, C is not prevalent. □

Lemma 3 *For a k ($k \geq 2$) size co-location $C = \{f_1, \ldots, f_k\}$, $f_{\min} \in C$ is the feature with the fewest number of instances and $f_{\max} \in C$ is the feature with the largest number of instances, suppose C' is a (k-1) size subset of C, $C' \subset C$, and $f_{\min} \notin C'$, if $PI(C') * w(f_{\max}, C) < min\_prev$, C must be not prevalent.*

Proof Let $f \in C'$ is the feature with the minimum participation ratio in C', i.e., $PR(f, C') = PI(C')$. Then, $PR(f, C) \leq PR(f, C') = PI(C')$ by Lemma 1. Depending on Definition 2, we have $v(f, C) \leq v(f_{\max}, C)$ since $num(f) \leq num(f_{\max})$ in C, and thus $0 < RI(f_{\max}, C) \leq RI(f, C) \leq 1$ by Definition 3. For $w(f_i, C) = \frac{1}{RI(f_i, C)}$, $w(f, C) \leq w(f_{\max}, C)$ holds. Thus, $WPI(C) \leq WPR(f, C) = PR(f, C) * w(f, C) \leq PI(C') * w(f_{\max}, C) < min\_prev$. Therefore, C is not prevalent. □

According to Lemmas 2 and 3, if a $(k-1)$ size co-location is not prevalent, its k size super patterns can be pruned conditionally, and thus WPI possesses a conditional anti-monotone property. The pruning effectiveness of the conditional anti-monotone property of WPI is a bit weaker than the anti-monotone property of PI, but stronger than the weak monotonicity of maxPR and the partial anti-monotonic property of WP-RR, which was verified in the experimental evaluation.

4 Ordered NR-tree-based mining approach

In WPI measure, given a co-location whose features have a significant difference in the quantity of instances, and if the features with fewer instances have lower participation ratio, this co-location must be not prevalent. Therefore, we can search for co-locations by such a way that the features involved in a co-location are sorted by the quantity of instances in ascending order. That is, for a $k(k \geq 2)$ size co-location $C = \{f_1, \ldots, f_k\}$, $num(f_i) \leq num(f_j)$ holds for every $1 \leq i < j \leq k$, in particular, if $num(f_i) = num(f_j)$, $f_i < f_j$ in alphabetical order holds. For simplicity, the above constraints are applied to all patterns in subsequent. Due to the conditional anti-monotone property of WPI, it is not necessary to test all combinations of features. In the light of Lemma 2, the precondition for C to be prevalent is that both its $(k-1)$ size sub-patterns $C_1 = \{f_1, \ldots, f_{k-2}, f_{k-1}\}$ and $C_2 = \{f_1, \ldots, f_{k-2}, f_k\}$ are prevalent. Thus, a k size candidate pattern can be obtained by joining two $(k-1)$ size prevalent co-locations, e.g., C can be derived by joining C_1 and C_2. Then, some candidates can be pruned without calculating the WPI metric according to Lemmas 2 and 3. Naturally, searching for co-locations in a level-wise manner is a better way to prune the search space as much as possible. Consequently, we adopt the generation-and-test mechanism to discover all prevalent co-locations, similar to the join-based algorithm [11].

4.1 Ordered NR-tree

To calculate the WPI metric of a candidate pattern, the relative weight and participation ratio of the involved features are necessary. Calculating the participation ratio is difficult, since all row instances are required and generating row instances is expensive for computation. Next, we present a novel approach for speeding up the generation of row instances.

Definition 6 Given an instance $o \in O$, its feature type $o.t \in F$, the ordered neighbor set of o is defined as

$$Neigh(o) = \left\{o' | o' \in O \wedge R(o, o') \wedge (num(o.t) \leq num(o'.t))\right\} \tag{8}$$

The ordered neighbor set of o includes instances who have neighbor relationships with o and the quantity of instances of $o'.t$ is larger than that of $o.t$. Particularly, if $num(o.t) = num(o'.t)$ but $o.t < o'.t$ in alphabetical order, instance o' is included in $Neigh(o)$.

Definition 7 Given an instance o and its ordered neighbor set $Neigh(o)$, the ordered neighbor set of o on feature f is defined as

$$Neigh(o, f) = \left\{o' | o' \in Neigh(o) \wedge (o'.t = f)\right\} \tag{9}$$

$Neigh(o, f)$ is a subset of $Neigh(o)$, and it includes all instances of feature f in $Neigh(o)$.

Definition 8 Given a co-location $C = \{f_1, \ldots, f_k\}, k \geq 2, I = \{o_1, \ldots, o_k\}$ is a row instance of C and $o_i.t = f_i$ holds for every $1 \leq i \leq k$. $C' = \{f_1, \ldots, f_k, f\}$ is a super pattern of C, the extended set of I on feature f is defined as

$$S(I, f) = Neigh(o_1, f) \cap \cdots \cap Neigh(o_k, f) \tag{10}$$

Lemma 4 *For co-location $C = \{f_1, \ldots, f_k\}$ with a row instance $I = \{o_1, \ldots, o_k\}$ and the corresponding $S(I, f)$, if $S(I, f) \neq \emptyset$, appending any instance o from $S(I, f)$ to I can form a row instance $I' = \{o_1, \ldots, o_k, o\}$ of co-location $C' = \{f_1, \ldots, f_k, f\}$.*

Proof Since $o \in S(I, f)$, the feature type of o is f. Thus, I' includes all features of C' and none of its subset does so. In addition, o has neighbor relationships with all instances in I by Definitions 6 and 7. That is, all instances of I' form a clique. Therefore, I' is a row instance of C'. □

Example 5 In Fig. 1, $Neigh$(D.2) = {A.2, A.3, C.2, C.3}, further, $Neigh$(D.2, A) = {A.2, A.3} and $Neigh$(D.2, C) = {C.2, C.3}. Similarly, $Neigh$(C.2, A) = {A.2}. Let I = {D.2, C.2} is a row instance of co-location {D, C}, we have $S(I$, A) = $Neigh$(D.2, A)∩ $Neigh$ (C.2, A) = {A.2}. Appending A.2 to I can form a row instance I' = {D.2, C.2, A.2} of co-location {D, C, A}.

Lemma 4 provides a novel generation approach of row instances, where the (k+1) size row instance is produced by expanding the k size row instance. In the generation-and-test mechanism, such an approach can effectively reuse the previously processed information. To query the ordered neighbor set of any instance fastly, we design a tree structure to materialize neighbor relationships.

Definition 9 In a spatial dataset, given the set of all features $F = \{f_1, f_2, \ldots, f_m\}$ and all neighbor relationships among instances, the ordered neighbor relationship tree (**ordered NR-tree**, for short) is designed as follows.

(1) It consists of one root labeled as Null, and each feature is a children of the root.
(2) The feature f_i $(1 \leq i \leq m)$ sub-tree consists of the root f_i, and the ordered neighbor sets of all instances of f_i constitute the branch of the root f_i. Each branch records an instance and its ordered neighbor set on different features.

Example 6 The ordered NR-tree of the example spatial dataset in Fig. 1 is illustrated in Fig. 2. The subtree rooted at D consists of the branches of D.1 and D.2. The branch of D.2 records its ordered neighbors grouped by features, e.g., $Neigh$(D.2, C)={C.2, C.3} is recorded as a sub-branch (labeled as C) of the branch of D.2.

In particular, if $Neigh(o)$ is empty for any instance o, the branch of o can be pruned. Moreover, the subtree of feature f_i also can be pruned if there is no branch in this subtree. The ordered NR-tree is unique for the given spatial dataset and neighbor relationship R, and it materializes all neighbor relationships without duplication and loss of information. Furthermore, it is convenient and efficient to query any $Neigh(o, f)$ from the ordered NR-tree. For this reason, the extended set of any row instance can be calculated quickly, and then the row instance extension can be performed efficiently.

4.2 Ordered NR-tree-based algorithm

The ordered NR-tree-based algorithm is presented in Algorithm 1, and it adopts the generation-and-test mechanism.

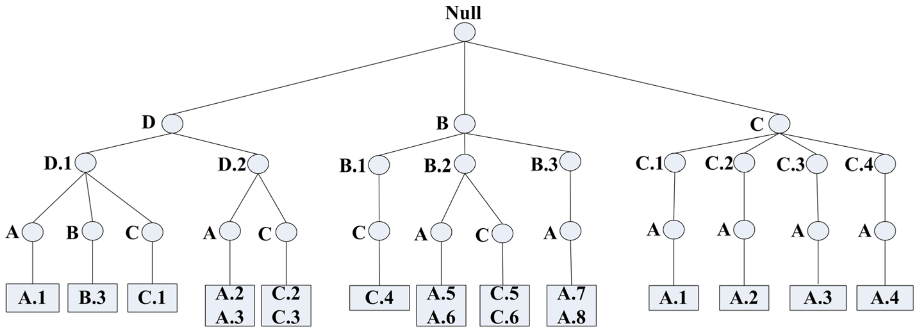


Fig. 2 Ordered NR-tree of the example spatial dataset in Fig. 1

Algorithm 1 ordered-NR-tree algorithm

Input: (a) spatial instance set O; (b) spatial feature set F; (c) neighbor relationship R; (d) minimum prevalence threshold min_prev

Output: A set of prevalent co-locations

Variables: (a) δ: the degree of dispersion; (b) k: co-location size; (c) C_k: set of k size candidate patterns; (d) T_k: set of the table instance of co-locations in C_k; (e) P_k: set of k size prevalent co-locations; (f) $Neigh$: the ordered neighbor set of all instances

Method:

```
1: counting the number of instances per feature
2: sorting features in ascending order of the quantity of instances
3: calculating $\delta$ for the spatial dataset
4: $Neigh$=gen_Neigh($F$, $O$, $R$)
5: ordered-NR-tree=gen_ordered-NR-tree($Neigh$, $F$)
6: let $P_1 = F, T_1 = O, k = 2$
7: while $P_{k-1}$ not empty do
8:     $C_k$=gen_candidate_patterns ($P_{k-1}$, $k$)
9:     filter_candidate_patterns ($C_k$, $P_{k-1}$)
10:    $T_k$ = gen_table_instances ($C_k$, $T_{k-1}$, ordered-NR-tree)
11:    calculate_WPI ($C_k$, $T_k$)
12:    $P_k$= select_prevalent_patterns ($C_k$, $T_k$, $min\_prev$)
13:    $k = k + 1$
14: end while
15: return union($P_2$, ..., $P_{k-1}$)
```

Initialization (Steps 1-6): Step 1 counts the number of instances per feature, and Step 2 sorts features in ascending order of the quantity of instances. Then, Step 3 calculates the degree of dispersion about the quantity of instances for features in the spatial dataset. Next, it finds all instance pairs with neighbor relationships using some geometric methods, e.g., the plane sweep, and generates the ordered neighbor set for all instances in Step 4. Step 5 scans all instances and their ordered neighbor set once to construct the ordered NR-tree. In Step 6, each feature $f_i \in F$ is considered as the size-1 prevalent co-location for the start of iterations, and its table instance is the set of instances of f_i.

Generate Candidate Patterns (Steps 8-9): As the previous analysis, in Step 8, the k size candidate pattern is generated by joining two prevalent $(k-1)$ size co-locations. For the sake of maintaining the consistent, features within the generated candidate pattern still keep the ascending order of the number of instances. Then, Step 9 prunes the candidate set using Lemmas 2 and 3, and only the leftover candidate patterns are needed to be tested.

Generate Table Instances (Step 10): For a candidate pattern $C = \{f_1, \ldots, f_k\}, k \geq 2$, its table instance is obtained by expanding all row instances of the sub-pattern $C' = \{f_1, \ldots, f_{k-1}\}$. For each row instance I of C', we generate the extended set of I on feature f_k, i.e., $S(I, f_k)$, where the required set $Neigh(o, f_k)$, $o \in I$, can be inquired fastly from the ordered NR-tree, and then append each instance of $S(I, f_k)$ to I to generate row instances of C.

Select Prevalent Co-locations (Steps 11-12): For each candidate pattern, Step 11 calculates the relative weight and participation ratio for each involved feature, and then calculates the weighted participation ratio. The minimum weighted participation ratio, i.e., WPI, is utilized as the interest measure of the candidate pattern. In Step 12, only the patterns whose WPI metric no less than min_prev are selected as the prevalent co-location.

Steps 8-13 are repeated with the increment in the size of patterns. All prevalent co-locations can be identified level-by-level through executing the ordered NR-tree-based algorithm.

For Steps 1-3, it needs to scan the spatial dataset once, so the complexity is $O(n)$, where n is the amount of instances. In Step 4, the complexity of the procedure to find all neighboring instance pairs is $O(n^2)$ in the worst case. However, the actual cost of Step 4 is much lower than $O(n^2)$ due to some heuristics. To construct the ordered NR-tree, all instances and their ordered neighbor set are scanned once, so the complexity is $O(n * \max(|Neigh(o)|))$, where $|Neigh(o)|$ is the size of the ordered neighbor set of any instance o. For the loop of Steps 7-14, generating table instances dominates the time cost, so we focus on this phase. The complexity to generate table instances for all candidate patterns is $O(|C_k| * \max(|T(c_{k-1})|) * \max(|S(I, f)|))$, where $|C_k|$ is the quantity of the k size candidate patterns, $|T(c_{k-1})|$ is the number of row instances of any $(k-1)$ size prevalent co-location, $|S(I, f)|$ is the size of the extended set of any row instance I on any feature f. In addition, the cost of generating $S(I, f)$ from the ordered NR-tree is $O(k * \max(|Neigh(o, f)|))$, $o \in I$, if a hash method is used to save the information of $Neigh(o, f)$. Overall, the cost of Algorithm 1 mainly depends on the number of candidate patterns, the number of table instances, the number of instances and their neighbors.

4.3 Completeness and correctness

Completeness means the ordered NR-tree-based algorithm discovers all prevalent co-locations. Correctness indicates that all co-locations discovered by the ordered NR-tree-based algorithm have the WPI metric no less than min_prev.

Theorem 1 *The ordered NR-tree-based algorithm is complete.*

Proof All $k(k \geq 2)$ size potential patterns have to be tested in the $(k-1)$th iteration. First, if a k size candidate pattern $C = \{f_1, \ldots, f_k\}$ is not generated in Step 8, it means that at least one of its sub-patterns $C_1 = \{f_1, \ldots, f_{k-2}, f_{k-1}\}$ and $C_2 = \{f_1, \ldots, f_{k-2}, f_k\}$ is not prevalent. Thus, C must be not prevalent according to Lemma 2. Second, for the generated candidate patterns in Step 8, the candidates pruned by Step 9 must be not prevalent, which can be guaranteed by Lemmas 2 and 3. So, there is no k size potential pattern omitted in the $(k-1)$th iteration. Therefore, the ordered NR-tree-based algorithm can discover all prevalent co-locations. □

Theorem 2 *The ordered NR-tree-based algorithm is correct.*

Proof The correctness of the ordered NR-tree-based algorithm can be shown in two aspects. The first is that the calculation of WPI for each candidate pattern is correct. The second is that only candidates whose WPI metric no less than min_prev are selected. The second is obvious by Step 12, next, we confirm the first. The key of calculating WPI is to generate the table instance for the candidate pattern completely and correctly. The method to generate row instances is correct according to Lemma 4. For each generated candidate pattern $C = \{f_1, \ldots, f_k\}, k \geq 2$, let $C' = \{f_1, \ldots, f_{k-1}\}$ be a sub-pattern of C, so C' must be a prevalent co-location discovered in the $(k-1)$th iteration. For any row instance $I = \{o_1, \ldots, o_k\}$ of C, its subset $I' = \{o_1, \ldots, o_{k-1}\}$ is a row instance of C'. In addition, instance o_k is an element of $S(I', f_k)$ by Definition 8. The ordered NR-tree does not miss and duplicate any neighbor relationship, so all row instances of C can be generated completely and correctly by expanding the table instance of C'. Therefore, the calculated WPI metric for each candidate pattern is correct. □

5 Parallel ordered NR-tree-based mining approach

Even though the ordered NR-tree-based algorithm is highly efficient, it has difficulty to process massive spatial data due to the limitation in the computing and storage of a single machine. Next, we adapt the ordered NR-tree-based algorithm to a distributed parallel algorithm for responding massive spatial data.

5.1 Independent ordered NR-subtree

It is a promising method to decompose the ordered NR-tree into multiple independent subtrees, so that the mining task can be executed on these subtrees in parallel.

Definition 10 Given a spatial dataset and its ordered NR-tree, the independent ordered neighbor relationship subtree (**ordered NR-subtree**, for short) is defined as follows.

(1) An ordered NR-subtree is a part of the ordered NR-tree, and the partitioning is at the level of features.
(2) An ordered NR-subtree consists of one root labeled as a feature $f \in F$, denoted as ordered NR-subtree(f).
(3) The subtree f of the ordered NR-tree is included in ordered NR-subtree(f), i.e., the ordered neighbor set of all instances with feature f is a branch of ordered NR-subtree(f).
(4) For an instance o and $o.t \neq f$, if there is an instance o' of f and $o \in Neigh(o')$, the ordered neighbor set of o is one branch of ordered NR-subtree(f).

Example 7 As Fig. 3 displayed, the ordered NR-tree shown in Fig. 2 is divided into three subtrees. In ordered NR-subtree(D), all branches of instances of D are included. Since B.3 $\in Neigh$(D.1), the branch of B.3 are contained in ordered NR-subtree(D).

Lemma 5 *Given a feature $f \in F$ and ordered NR-subtree(f), for the k ($k \geq 2$) size co-location $C = \{f_1, \ldots, f_k\}$ where $f_1 = f$, its any row instance can be generated from ordered NR-subtree(f).*

Proof Apparently, all row instances of the size-2 co-location whose first feature type is f can be generated from ordered NR-subtree(f), since all branches of instances of f are included in this subtree. For a $k(k > 2)$ size co-location C, $C' = \{f_1, \ldots, f_{k-1}\}$ is a subset of C. Given any row instance $I' = \{o_1, \ldots, o_{k-1}\}$ of C', the extended set of I' on feature f_k,

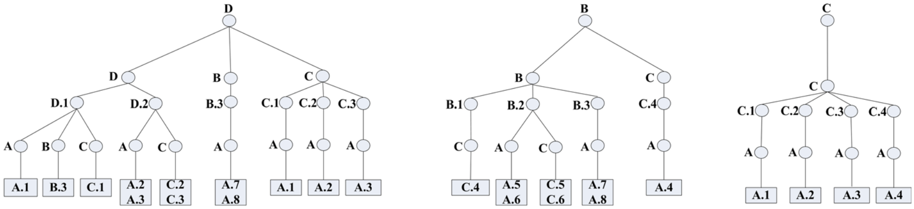


Fig. 3 Ordered NR-subtrees of the example spatial dataset in Fig. 1

i.e., $S(I', f_k)$, can be calculated from ordered NR-subtree(f), as the branch of any instance $o \in I'$ is included in ordered NR-subtree(f) by the conditions (3) and (4) of Definition 10. Therefore, all row instances of C can be generated through expanding the table instance of C' on ordered NR-subtree(f). To sum up, the table instance of any k size co-location whose first feature is f can be generated from ordered NR-subtree(f). □

Under the guarantee of Lemma 5, given ordered NR-subtree(f), the group of co-locations whose first feature is f can be searched by executing Steps 7–14 of Algorithm 1 on this subtree independently. Therefore, we can perform the mining task on multiple ordered NR-subtrees in parallel. Based on Lemma 5, as well as the completeness and correctness of the ordered NR-tree-based algorithm, it is trivial to prove that the parallel ordered NR-subtree-based approach is complete and correct.

Lemma 6 *Given a feature $f \in F$ and its ordered NR-subtree(f), $C = \{f_1, \ldots, f_k\}$ is a k size prevalent co-location where $f_1 = f$, and the number of branches of subtree f_k in ordered NR-subtree(f) is q, if $\frac{q}{num(f_k)} * w(f_k, C) < min\_prev$, all super patterns of C must be not prevalent.*

Proof Let $C' = \{f_1, \ldots, f_k, f_{k+1}\}$ is a super pattern of C, we have $w(f_k, C') = w(f_k, C)$ since $RI(f_k, C') = RI(f_k, C)$ by Definition 3. Besides, $PR(f_k, C') \le \frac{q}{num(f_k)}$, because at most q instances of f_k participate in the table instance of C' by the condition (4) of the ordered NR-subtree. Thus, $WPI(C') \le PR(f_k, C') * w(f_k, C') \le \frac{q}{num(f_k)} * w(f_k, C) < min\_prev$. Therefore, C' is not prevalent. □

Lemma 6 provides a new pruning strategy based on the ordered NR-subtree to accelerate the process of performing the mining task on each independent subtree. Next, we parallelize the ordered NR-tree-based algorithm on MapReduce framework.

5.2 MapReduce algorithm

MapReduce is a programming model which provides a highly scalable and flexible framework for data-oriented parallel computing. Two key functions, namely Map and Reduce, are provided in MapReduce framework. Map function receives a pair of (key, $value$) data as input and generates a list of intermediate ($key1$, $value1$) pairs to output. Then, the intermediate pairs produced by Map function are moved to Reduce function by Shuffle phase. When Shuffle phase is finished, values associated with the same key are gathered together, saved as $Set(key)$. In Reduce function, the formatted ($key1$, $Set(key1)$) pairs are input and processed, then the new formatted ($key2$, $value2$) pairs are generated to output.

Next, we present the parallel ordered NR-tree-based algorithm using MapReduce framework. Some preparations need performing in advance, (1) counting the number of instances per feature and calculating the degree of dispersion about the quantity of instances for features in the spatial dataset, which are easy to perform by scanning the spatial dataset once, (2) finding all neighboring instance pairs, which can be achieved using the parallel plane sweep technique [29]. In this paper, we only focus on the key steps, i.e., generating the ordered neighbor set per instance, building ordered NR-subtree for each feature and mining prevalent co-locations on each subtree in parallel.

At first, the task of generating the ordered neighbor set per instance is presented in Algorithm 2. The neighboring instance pair (o_i, o_j) where $o_i.t < o_j.t$ in alphabetical order is fed for Map function. Then, Map function outputs pairs (o_i, o_j) and (o_j, o_i), so that all instances neighboring with the same instance o can be collected together as $NS(o)$ by Shuffle process. In Reduce side, for each instance o and its all neighbors $NS(o)$, we generate the ordered neighbor set of o (i.e., $Neigh(o)$) by Definition 6. Moreover, the ordered NR-subtrees to which instance o and $Neigh(o)$ belong are recorded in $subtree(o)$ according to Definition 10.

Algorithm 2 generating ordered neighbor set

```
procedure MAP(key=$o_i$, value=$o_j$)
    emit($o_i$, $o_j$), emit($o_j$, $o_i$)
end procedure
procedure REDUCE(key=$o$, value=$NS(o)$)
    $Neigh(o)$=[ ], $subtree(o)$=[$o.t$]
    for all $x \in NS(o)$ do
        if $num(x.t) > num(o.t)$ then
            append $x$ into $Neigh(o)$
        else
            append $x.t$ into $subtree(o)$
        end if
    end for
    emit($o$, ($Neigh(o)$, $subtree(o)$))
end procedure
```

Algorithm 3 parallel ordered-NR-subtree based mining

```
procedure MAP(key=$o$, value=($Neigh(o)$, $subtree(o)$))
    for all $f \in subtree(o)$ do
        emit($f$, ($o$, $Neigh(o)$))
    end for
end procedure
procedure REDUCE(key=$f$, value=$S(f)$)
    ordered-NR-subtree=gen_ordered-NR-tree($S(f)$, $f$)
    let $P_1$=$f$, $T_1$=$O(f)$, $k$=2
    while $P_{k-1}$ not empty do
        $C_k$=gen_candidate_patterns ($P_{k-1}$, $k$)
        filter_candidate_patterns ($C_k$, $P_{k-1}$)
        $T_k$ = gen_table_instances ($C_k$, $T_{k-1}$, ordered-NR-subtree)
        calculate_WPI ($C_k$, $T_k$)
        $P_k$= select_prevalent_patterns ($C_k$, $T_k$, $min\_prev$)
        $k = k + 1$
    end while
    emit union($P_2$, ..., $P_{k-1}$)
end procedure
```

For Algorithm 2, all neighboring instance pairs are scanned by Map function, so the complexity of Map phase is $O(|R|/P))$, where $|R|$ is the number of neighboring instance pairs and P is the number of worker nodes in the cluster. The number of instance pairs shuffled in the network is $2|R|$ and thus the complexity of Shuffle phase is $O(|R|)$. In Reduce phase, all instances and their neighbors are scanned, so the complexity is $O(n * \max(|NS(o)|)/P)$, where n is the number of all instances and $|NS(o)|$ is the number of neighbors of any instance o. To sum up, the neighbor relationship R determines the complexity of Algorithm 2.

When Algorithm 2 is finished, its outputs are fed to Map function of Algorithm 3. Each instance o with $Neigh(o)$ is assigned to different groups according to features recorded in $subtree(o)$. After Shuffle phase, all pairs $(o, Neigh(o))$ belonging to the same group (labeled as f) are collected together as $S(f)$. In Reduce side, we construct the ordered NR-subtree for each group. Then, on the ordered NR-subtree rooted at f, let $P_1 = f$ as the beginning of iteration, and the next steps are same as Algorithm 1. Note that Lemmas 2 and 6 are used to filter candidate patterns in Step 11, but Lemma 3 is invalid as the sub-pattern $\{f_2, \ldots, f_k\}$ of $\{f_1, f_2, \ldots, f_k\}$ is not searched on ordered NR-subtree(f_1).

In Map phase of Algorithm 3, all instances are scanned once to assign the subtree to which they belong, so the complexity is $O(n*\max(|subtree(o)|)/P)$, where n is the amount of instances, $|subtree(o)|$ is the number of subtrees to which instance o belongs, P is the number of worker nodes. All pairs generated by Map function are shuffled, so the complexity of Shuffle phase is $O(n*\max(|subtree(o)|))$. In Reduce phase, constructing the ordered NR-subtree and performing the mining task on each subtree is same as Steps 5–14 of Algorithm 1, so the complexity analysis is similar. The most costly step of Algorithm 3 is the task that mining co-locations on each subtree.

6 Experimental results

In this section, we demonstrate the experimental evaluation on real-world and synthetic spatial datasets. First, we evaluate the effectiveness of WPI measure. Second, we assess the efficiency of the ordered NR-tree-based algorithm. Third, we examine the efficiency gain and scalability of the parallel ordered NR-tree-based algorithm.

6.1 Effectiveness of WPI

We assess the effectiveness of WPI measure on two real-world spatial datasets. The first is the plant dataset of Three Parallel Rivers of Yunnan Protected Area, including ten features and 3845 instances. The detail is shown in Table 2 and $\delta = 4.1639$ for this dataset. The second is the plant dataset of Gaoligong Mountain, containing ten features and 11,062 instances. Table 3 outlines the detail, and the δ value of this dataset is 12.2583. For comparison, three interest measures (i.e., PI, maxPR and WP-RR) mentioned in related work are used as competitors.

(1) Comparison of different interest measures

At first, the experiment is performed on the plant dataset of Three Parallel Rivers of Yunnan Protected Area with the distance threshold $d = 1000$ m. For WP-RR measure, the minimum rare ratio threshold t is set to 0.05 and 0.7 respectively. Table 4 shows six co-locations with their different interest metrics. For co-locations whose features with a great variation in the quantity of instances, e.g., {H, A} and {H, G, A}, their PI metrics are much lower so that they can not be identified. In co-location {H, A}, the quantity of instances of H is significantly less than that of A. Figure 4a displays the distribution of instances of features H and A,

Table 2 Plant dataset of Three Parallel Rivers of Yunnan Protected Area

Spatial feature type	Label	Number of instances
Temperate coniferous forest	A	1535
Ice and snow vegetation	B	479
Warm coniferous forest	C	438
Warm temperate sparse tree shrub	D	301
Dry hot shrub	E	282
Coniferous and broad-leaved mixed forest	F	269
Alpine meadow	G	236
Cold temperate shrub	H	169
Cold and warm mountain evergreen oak forest	I	82
Cultivated vegetation	J	54

and we can see that always there are instances of A around instances of H. Statistically, the participation ratio of H in {H, A} is 0.9527. For maxPR measure, such co-locations whose rare feature has high participation ratio can be highlighted, since the maximum participation ratio is used as the interest measure. The WPI metric of the co-location which involves rare features with high participation ratio is also larger, because WPI measure considers both the prevalence and rare intensity of features in a pattern. For the co-location whose features with a smaller difference in the number of instances, e.g., {H, G}, {G, B}, {E, C}, {D, C}, PI measure reflects the minimum probability that instances of features participate in the co-location; thus, PI is the more suitable measure when the co-location does not involve rare features. The maxPR metric of co-locations without rare features is usually higher than that of PI, which may make some non-prevalent patterns without rare features identified by maxPR. For example, as visualized in Fig. 4b, we can observe that there is no instance of E occured around the most instances of C, but its maxPR metric is high (0.8262). For co-locations without rare features, their WPI metrics are close to PI, because the rare intensity of the involved features is close to 1 and the participation ratio of features dominates WPI measure. Moreover, we can see that the WP-RR metric of a co-location is greatly affected by the minimum rare ratio threshold t. If t is too small, e.g., $t = 0.05$, WP-RR measure may degenerate as PI and co-locations with rare features can not be captured. Instead, a larger t (e.g., 0.7) may make the WP-RR metric of co-locations without rare features much higher, just like maxPR measure.

Table 5 shows the example co-locations in the plant dataset of Gaoligong Mountain, with $d = 2000\,\text{m}$. For WP-RR measure, the minimum rare ratio threshold t is set to 0.01 and 0.5 respectively. Similar conclusions are given in Table 5. Concretely, co-locations with rare features have low PI metric even though the rare feature has high participation ratio. Taking co-location {S, L} shown in Fig. 5a as an example, the number of instances of L is 38.85 times that of S. Though 85.71% of instances of S have neighbor relationships with instances of L, its PI metric is only 0.0607 due to the lower participation ratio of L. The maxPR and WPI metrics of {S, L} are high since both they consider the impact of rare features. However, maxPR measure may magnify the interest metric of a co-location when the involved features have a less difference in the quantity of instances. As an illustration, for co-location {O, K} shown in Fig. 5b, there is no rare features in this pattern intuitively, and the probability that instances of K participate in {O, K} is only 0.3852, but the maxPR metric of {O, K} is as high as 0.7418. For co-locations without rare features, WPI and PI measures are almost the

Table 3 Plant dataset of Gaoligong Mountain

Spatial feature type	Label	Number of instances
Abies	K	2866
Tsuga	L	2176
Yunnan Pine	M	1958
Arrow Bamboo	N	1288
Miscellaneous irrigation	O	1282
Alder Wood	P	802
Quercus	Q	471
Rhododendron simsii planch	R	110
Larch	S	56
Spruce	T	53

Table 4 Example co-locations in the plant dataset of Three Parallel Rivers of Yunnan Protected Area

Co-locations	PI	WPI	maxPR	WP-RR	
				$t = 0.05$	$t = 0.7$
{H, A}	0.3114	0.9527	0.9527	0.3114	0.9527
{H, G, A}	0.1921	0.6470	0.7988	0.1921	0.6441
{H, G}	0.7288	0.7321	0.8639	0.7288	0.7288
{G, B}	0.6534	0.6737	0.7034	0.6534	0.7034
{E, C}	0.4361	0.4399	0.8262	0.4361	0.6773
{D, C}	0.4064	0.4088	0.6777	0.4064	0.5914

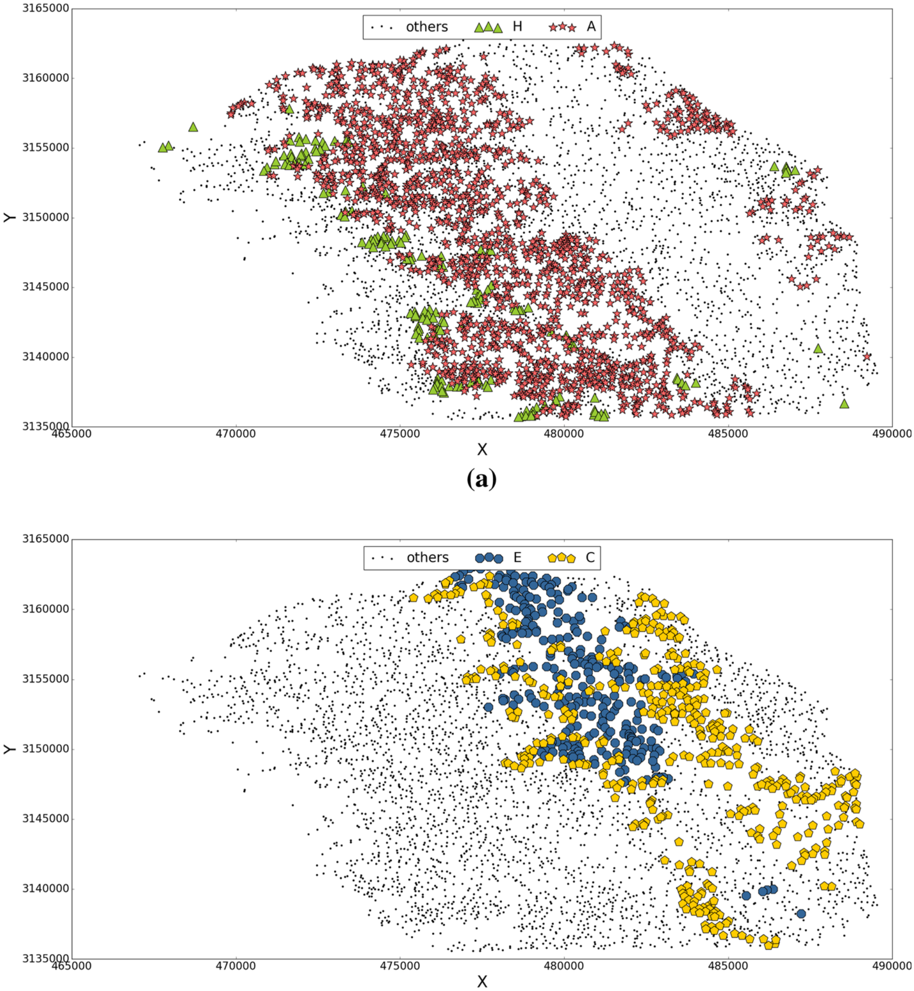
Table 5 Example co-locations in the plant dataset of Gaoligong Mountain

Co-locations	PI	WPI	maxPR	WP-RR	
				$t = 0.01$	$t = 0.5$
{S, L}	0.0607	0.8571	0.8571	0.0607	0.8571
{S, L, K}	0.0516	0.8036	0.8036	0.0516	0.8036
{L, K}	0.5935	0.5937	0.7472	0.5935	0.5935
{P, M}	0.5531	0.5570	0.8591	0.5531	0.8591
{O, K}	0.3852	0.3872	0.7418	0.3852	0.7418
{P, L}	0.2224	0.2246	0.5823	0.2224	0.5822

same, because the rare intensity of features within such co-locations plays a weak role in WPI measure. Similarly, the threshold t has a great impact on WP-RR measure. A smaller t, e.g., 0.01, may cause that co-locations with rare features have a smaller WP-RR metric; however, WP-RR faces the same problem as maxPR if a larger t is given.

(2) Comparison of the discovered co-locations

Next, we compare co-locations discovered by different interest measures. Tables 6 and 7 list co-locations identified from two datasets, respectively, where the prevalence threshold min_prev is set to 0.55 and other thresholds remain the same as above. Intuitively, the number of co-locations found by PI, maxPR and WPI holds a relationship PI < WPI < maxPR. As the previous analysis, PI measure misses some co-locations with rare features, but WPI and maxPR have ability to discover such co-locations; besides, WPI measure is more strict than maxPR, since maxPR only requires there is at least one feature having the participation ratio



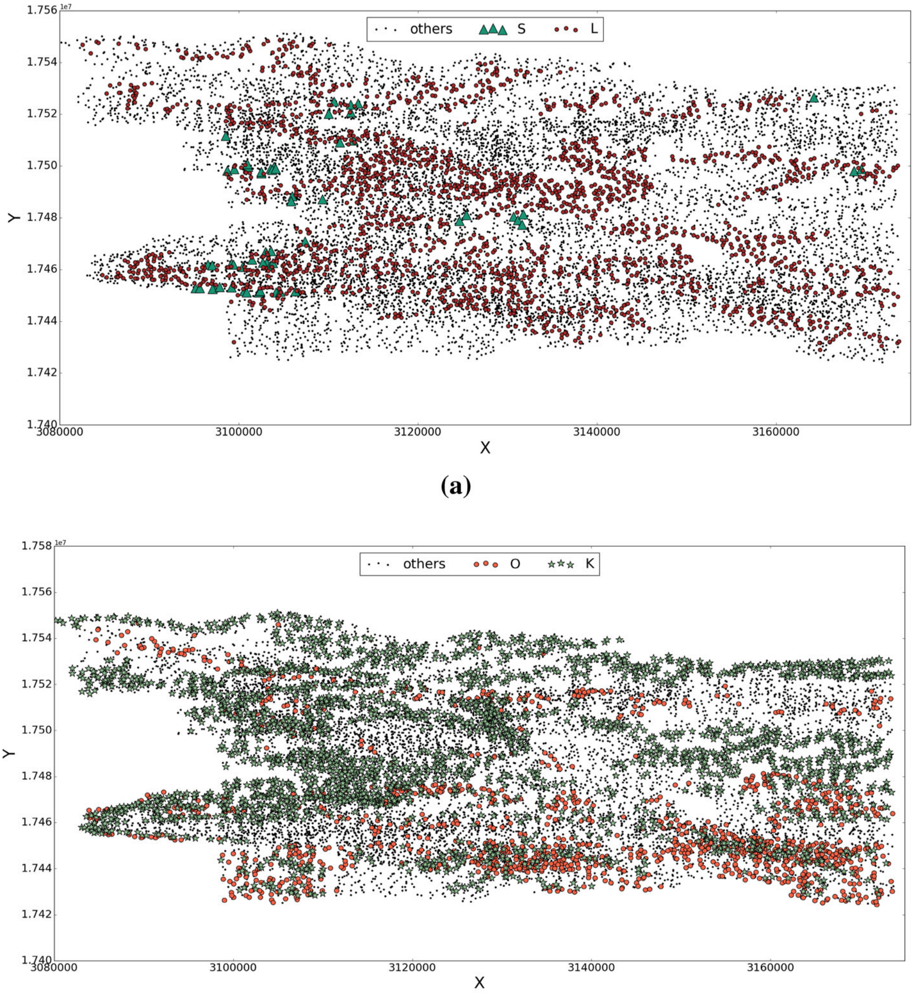
(a)

(b)

Fig. 4 Distribution of instances on the plant dataset of Three Parallel Rivers of Yunnan Protected Area, X, Y is the plane coordinate axis and points with different shapes (or colors) represent instances of different features, **a** co-location {H, A}, **b** co-location {E, C}

no smaller than min_prev, but WPI requires all features in a pattern should have a certain percentage of instances to participate in this pattern. For WP-RR approach, the number of discovered co-locations varies greatly with different threshold t. The co-locations discovered by WP-RR with a smaller t are same as that of PI, and the results of WP-RR are close to that of maxPR if a larger t is specified.

Figure 6 shows the number of co-locations discovered by different approaches, with changing min_prev from 0.1 to 0.9. As min_prev increases, the number of co-locations identified by all approaches decreases, since fewer patterns satisfy the condition of prevalent co-locations. Under the same min_prev, similar conclusions as above can be obtained.



(a)

(b)

Fig. 5 Distribution of instances on the plant dataset of Gaoligong Mountain, X, Y is the plane coordinate axis and points with different shapes (or colors) represent instances of different features, **a** co-location {S, L}, **b** co-location {O, K}

(3) Quality of the discovered co-locations

Then, we evaluate the quality of co-locations found by different interest measures. Given a prevalence threshold min_prev, the prevalent co-locations without rare features should satisfy the condition that the participation ratio per feature is no smaller than min_prev. However, identifying prevalent co-locations with rare features is difficult since the rare feature is a fuzzy concept. In a co-location $C = \{f_1, \ldots, f_k\}$, let $f_{\max}$ be the feature with the largest number of instances in C, and f_i can be considered as a rare feature in C if the ratio $r(f_i, C) = \frac{num(f_i)}{num(f_{\max})}$ is less than a given rare threshold min_r, i.e., $r(f_i, C) < min\_r$. For a prevalent co-location with rare features, the participation ratio per rare feature should be no less than min_prev. Under different min_r, the prevalent co-locations are different.

Table 6 Results on the plant dataset of Three Parallel Rivers of Yunnan Protected Area

PI	WPI	maxPR	WP-RR	
			$t = 0.05$	$t = 0.7$
{H, G}	{H, A}	{H, A}	{H, G}	{H, A}
{G, B}	{H, G}	{G, A}	{G, B}	{G, A}
	{F, A}	{H, G}		{J, C}
	{G, B}	{J, C}		{H, G}
	{I, A}	{E, C}		{F, A}
	{G, A}	{F, A}		{G, B}
	{J, D}	{G, B}		{E, C}
	{J, C}	{D, C}		{C, A}
	{H, G, A}	{C, A}		{I, A}
		{I, A}		{H, B}
		{H, B}		{J, D}
		{J, D}		{D, C}
		{H, G, A}		{H, G, A}
		{H, G, B}		{G, B, A}
		{G, B, A}		

Table 7 Results on the plant dataset of Gaoligong Mountain

PI	WPI	maxPR	WP-RR	
			$t = 0.01$	$t = 0.5$
{L, K}	{R, K}	{R, K}	{L, K}	{R, K}
{P, M}	{S, K}	{S, K}	{P, M}	{S, K}
	{T, L}	{T, L}		{T, L}
	{S, L}	{P, M}		{P, M}
	{T, K}	{S, L}		{S, L}
	{L, K}	{Q, L}		{Q, L}
	{P, M}	{N, K}		{N, K}
	{S, L, K}	{L, K}		{O, K}
	{T, L, K}	{O, K}		{T, K}
		{T, K}		{Q, M}
		{Q, M}		{L, K}
		{P, O}		{P, L}
		{P, L}		{S, L, K}
		{O, N}		{T, L, K}
		{S, L, K}		
		{T, L, K}		

In a spatial dataset, let T be the co-location set where each co-location satisfies one of the following conditions: (1) the participation index is no smaller than min_prev; (2) the participation ratio per feature with $r(f_i, C) < min\_r$ is no smaller than min_prev. Let S_* be the co-locations discovered by different approaches under min_prev, e.g., S_{PI} is the results of PI. Then, we use precision and recall metrics to evaluate the difference between the mined results and the set T, $precision = \frac{|TP|}{|TP|+|FP|}$, $recall = \frac{|TP|}{|TP|+|FN|}$,

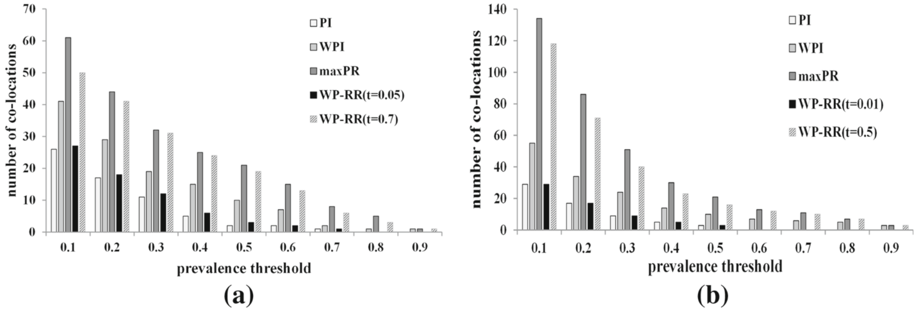


Fig. 6 Number of prevalent co-locations discovered by different interest measure: **a** the plant dataset of Three Parallel Rivers of Yunnan Protected Area, **b** the plant dataset of Gaoligong Mountain

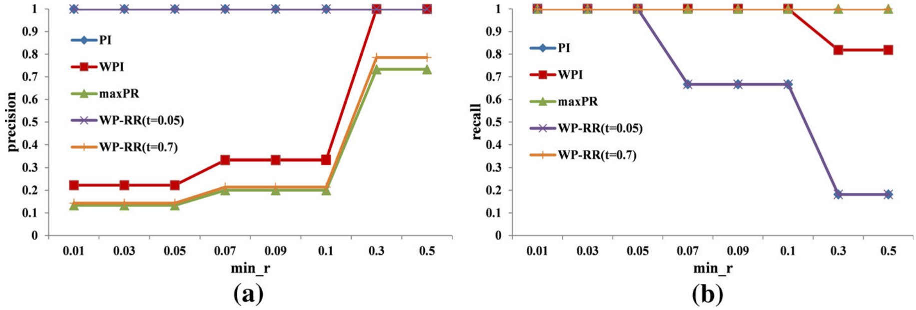


Fig. 7 Evaluation of the results on the plant dataset of Three Parallel Rivers of Yunnan Protected Area: **a** precision, **b** recall

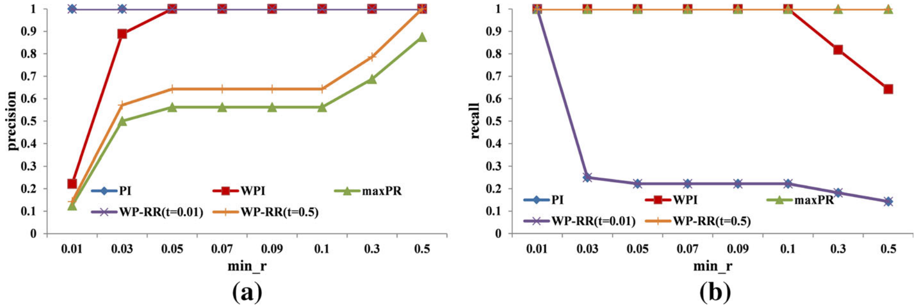


Fig. 8 Evaluation of the results on the plant dataset of Gaoligong Mountain: **a** precision, **b** recall

where $TP = S_* \cap T$, $FP = S_* - T$ and $FN = T - S_*$. In the experiment, we compare the results discovered by different approaches under $min\_prev = 0.55$ to the set T with min_r changing. Experimental results are shown in Figs. 7 and 8.

The PI approach has the precision to 1.0 for different min_r, but its recall is the lowest than other approaches, as it only finds the co-locations without rare features. The maxPR approach has the recall to 1.0 always, because co-locations in T must have their maximum participation ratio no less than min_prev and so they can be identified by maxPR; but, the precision of maxPR is the lowest since it may identify lots of non-prevalent patterns. For WPI, the recall is always to 1.0 when min_r is lower indeed, which means that WPI can discover all co-locations in T. When min_r is larger, i.e., the difference in the quantity of instances for features in a pattern is not obvious, WPI may miss some patterns in T. Under a larger min_r, more features can be considered as rare and the condition (2) of the set T is easily achieved because it only requires that the participation ratio of the feature with $r(f, C) < min\_r$ no smaller than min_prev. However, WPI measure has a certain requirement for the participation ratio of other features, as we analyzed in Sect. 3.2. Therefore, some patterns are discarded by WPI owing to the lower participation ratio of other features. As min_r increases, the precision of WPI also rises due to the fact that more co-locations identified by WPI can be included in the set T. Moreover, the precision of WPI is always larger than maxPR, which benefits from that WPI measure is more strict than maxPR to determine whether a co-location is prevalent. At last, we can note that the precision and recall of WP-RR have a larger effect on the threshold t specified by users.

To sum up, WPI measure has ability to discover prevalent co-locations without rare features; furthermore, it also can identify co-locations whose features have the significant difference in the quantity of instances and the participation ratio of the rare feature is high. Besides, WPI can filter some non-prevalent co-locations compared to maxPR.

6.2 Efficiency of the ordered NR-tree-based algorithm

In this section, we examine the efficiency of WPI compared with other interest measures, and the improvement of the ordered NR-tree-based algorithm. All four interest measures (i.e., PI, maxPR, WP-RR and WPI) need to generate row instances per candidate pattern. For the sake of fairness, we use the state-of-the-art instance-lookup scheme used in the join-less algorithm [31] to generate row instances for them. All algorithms are implemented in JAVA and run on a 3.60 GHz Intel Core i7 machine with 16GB main memory.

(1) Effect of distance thresholds

First, we examine the effect of distance thresholds. Figure 9a shows the result on the plant dataset of Three Parallel Rivers of Yunnan Protected Area and Fig. 10a presents the result on the plant dataset of Gaoligong Mountain, with min_prev to 0.2. For WP-RR measure, $t = 0.2$ in Fig. 9a and $t = 0.1$ in Fig. 10a, respectively. Seen from experimental results, the running time of five approaches increases as the distance threshold increases. A larger distance threshold means more neighbor relationships, and thus more instances form cliques. Therefore, more execution time is needed to identify row instances for testing candidate patterns. In terms of efficiency, WPI is slightly lower than PI, but PI can not discover co-locations with rare features. Besides, WPI is better than maxPR and WP-RR, as WPI measure satisfies a conditional anti-monotone property and can filter more candidate patterns, but maxPR and WP-RR measures have the weaker ability to prune the search space. In particular, maxPR is the least efficient among all approaches due to the worst pruning technique. Compared with the basic algorithm of WPI, the ordered NR-tree-based algorithm has a significant improvement in efficiency especially under a larger distance threshold. Because the latter adopts the ordered NR-tree-based mechanism to generate row instances efficiently.

(2) Effect of prevalence thresholds

Second, we examine the effect of prevalence thresholds. The result on the plant dataset of Three Parallel Rivers of Yunnan Protected Area is shown in Fig. 9b, with $d = 2500\,\text{m}$. Figure 10b displays the result on the plant dataset of Gaoligong Mountain, with $d = 4500\,\text{m}$. For WP-RR measure, $t = 0.2$ in Fig. 9b and $t = 0.1$ in Fig. 10b similarly. In summary,

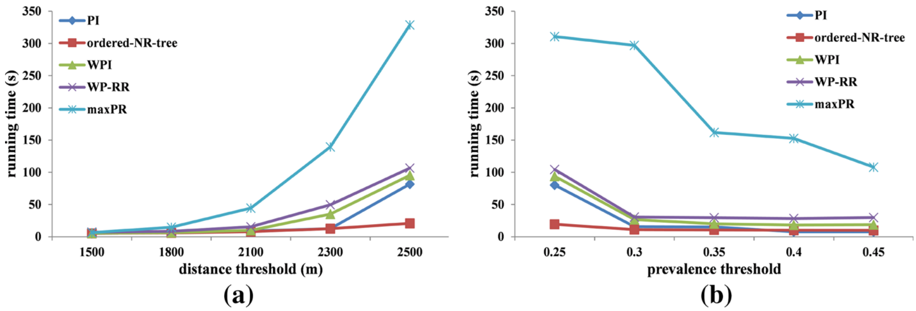


Fig. 9 Running time over the plant dataset of Three Parallel Rivers of Yunnan Protected Area: **a** by distance thresholds, **b** by prevalence thresholds

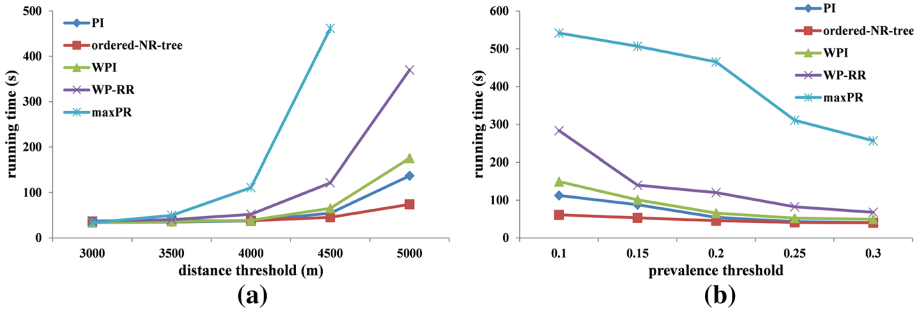


Fig. 10 Running time over the plant dataset of Gaoligong Mountain: **a** by distance thresholds, **b** by prevalence thresholds

the running time of five approaches decreases with the increase in prevalence thresholds. Under a higher prevalence threshold, more co-locations are considered as non-prevalent and more candidate patterns can be filtered out by the pruning techniques, so the execution time is reduced. Similarly, the efficiency of WPI is slightly worse than PI, but better than maxPR and WP-RR, and maxPR is still the worst in performance. Besides, the ordered NR-tree-based algorithm is superior to the basic algorithm of WPI obviously, especially when the prevalence threshold is smaller.

(3) Effect of the number of instances

Third, we evaluate the scalability of five algorithms on synthetic datasets. We generate instances by the spatial data generator [31] and distribute them into a 5000 × 5000 space. The number of features is set to 50 and the number of total instances is increased from 100K to 300K. For the sake of fairness, the number of instances of features in all synthetic datasets obeys the same normal distribution. We set d to 30 and min_prev to 0.3, besides, the threshold t of WP-RR measure defaults to 0.2. The result is shown in Fig. 11a, and we can observe that the running time of five approaches raises as the increasing of the amount of instances. The density of instances becomes more dense with increasing the total instances in the same space. Likewise, the efficiency of maxPR is significantly weaker than others. WPI is slightly lower than PI but better than WP-RR. In addition, the ordered NR-tree-based algorithm is the most efficient, and it achieves a great improvement compared with the basic algorithm of WPI, especially when the amount of instances is larger. That is, the ordered NR-tree-based algorithm shows the better scalability to large dense spatial datasets.

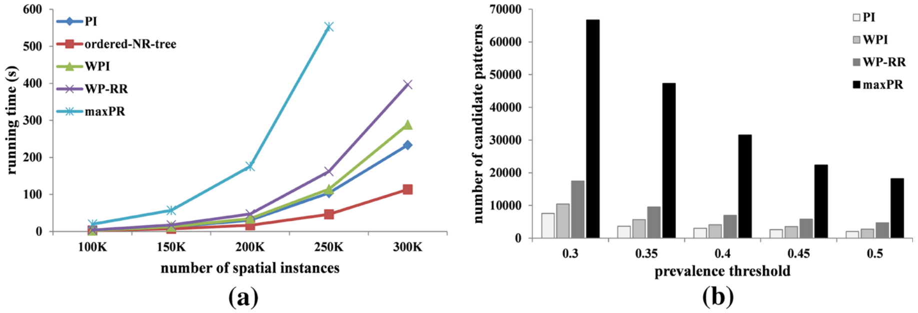


Fig. 11 Experiments on synthetic spatial datasets: **a** the effect of the number of instances, **b** the effect of pruning techniques

(4) Effect of pruning techniques

Fourth, we assess the effect of pruning techniques for four interest measures on the synthetic dataset with 300K instances and 50 features, where other parameter settings are same as the experiment in Fig. 11a. The result is demonstrated in Fig. 11b. We utilize the number of candidate patterns which have to be tested by different approaches, i.e., the search space after pruning, to quantify the effectiveness of pruning techniques of four interest measures. The more the number of candidates, the worse the pruning effect. With the increasing of prevalence thresholds, the pruning effect becomes better for all approaches. With the same prevalence threshold, the pruning effect of maxPR is the worst and the pruning effect of WPI is between PI and WP-RR. This is consistent with the property possessed by them, i.e., PI possesses an anti-monotone property, WPI satisfies a conditional anti-monotone property, WP-RR has a partial monotonicity property, but maxPR only has a weak monotonicity property.

In summary, WPI has the higher efficiency than maxPR and WP-RR due to the conditional anti-monotone property. Furthermore, the ordered NR-tree-based algorithm outperforms the basic algorithm of WPI in efficiency, which means that the proposed row instance generation mechanism is beneficial for the efficiency improvement.

6.3 Efficiency of the parallel ordered NR-tree-based algorithm

In this section, we evaluate the efficiency of the parallel ordered NR-tree-based algorithm on real and synthetic spatial datasets. We implement this parallel algorithm in Scala, Spark RDD library functions. Experiments are conducted on the cluster that deployed Hadoop and Spark. The cluster includes one master and ten slaves, where each machine has 3.60 GHz Intel Core i7 processor with 16GB RAM.

(1) Efficiency gain

The experiment is conducted on the POI data of Beijing firstly to assess the performance gain of the parallel ordered NR-tree-based algorithm compared to its serial version. The dataset contains 63 features and 303,895 instances, and the number of instances of features ranges from 929 to 31,991. The experimental results are presented in Fig. 12, where the prevalence threshold is 0.4 in Fig. 12a, and the distance threshold is 200m in Fig. 12b. The parallel ordered NR-tree-based algorithm outperforms the serial version significantly, especially for the larger distance threshold or the smaller prevalence threshold. In the parallel algorithm, the ordered NR-tree is disintegrated into some independent subtrees, so that the

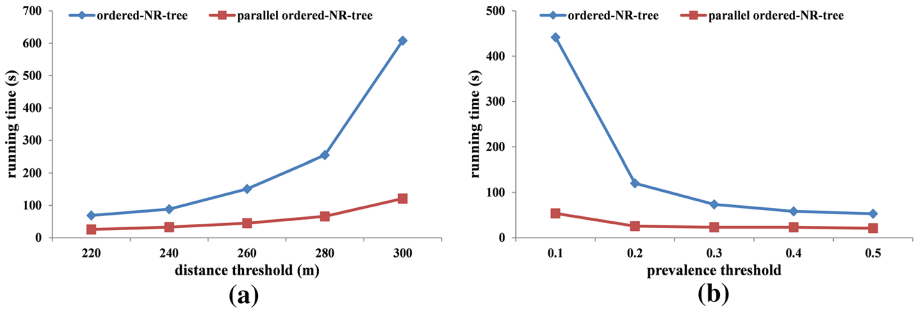


Fig. 12 Running time over the POI data of Beijing: **a** by distance thresholds, **b** by prevalence thresholds

mining task is assigned to multiple machines to execute in parallel. Naturally, the parallel algorithm based on ordered NR-subtrees is superior to the serial version in efficiency, and this advantage is more distinct when the amount of calculation is larger, e.g., the parallel ordered NR-tree-based algorithm improves efficiency by 5 times when $min\_prev = 0.2$, but by 8 times when $min\_prev = 0.1$.

(2) Scalability

Next, we test the scalability of the parallel ordered NR-tree-based algorithm with several workloads.

First, we examine the effect of the amount of worker nodes. Figure 13a shows the experimental results on the POI data of Beijing with $d = 300\,\text{m}$ and $min\_prev = 0.4$, where task-1 indicates the execution time of Algorithm 2 to generate the ordered neighbor set for all instances, and task-2 is the execution time of Algorithm 3 to construct ordered NR-subtrees and perform mining tasks on each subtree. As the number of worker nodes increases, the execution time is reduced, and it is valid for both two tasks. Moreover, the execution time of task-2 as far exceeds that of task-1, since the expensive row instance identification is performed in task-2.

Second, on the synthetic spatial datasets we examine the efficiency gain of the parallel ordered NR-tree-based algorithm for different amount of instances. The number of features is fixed at 200, and the number of instances is increased from 1000 to 4000 K and all instances are distributed into a 10,000 × 10,000 space. Figure 13b shows the experimental result with $d = 12$ and $min\_prev = 0.1$. As shown in Fig. 13b, the execution time is decreased with the increase in number of worker nodes. In particular, the efficiency gain is more higher when the amount of instances is larger, e.g., 4000 K. For the dataset with sparse distribution of instances, the efficiency gain decreases when the number of worker nodes reaches a certain level. According to the Amdalh's law, the speedup can not be always linear. When the execution time of each node is very short, continue to add nodes cannot improve efficiency dramatically, but the communication cost among nodes is increased. Moreover, we can note that the larger the number of instances is, the later the Amdalh's law takes effect.

Third, we investigate the efficiency improvement of the parallel ordered NR-tree-based algorithm on different thresholds. Experiments are conducted on the synthetic spatial dataset with 200 features and 3000 K instances. In Fig. 13c, we fix the prevalence threshold to 0.2 and the distance threshold to 10, 12 and 14, respectively. In Fig. 13d, we set the distance threshold to 12 and the prevalence threshold to 0.1, 0.2 and 0.3, respectively. With the increase in worker nodes, the execution time is reduced significantly for all experimental settings. Moreover, the efficiency gain is more noticeable when the distance threshold is larger or the prevalence

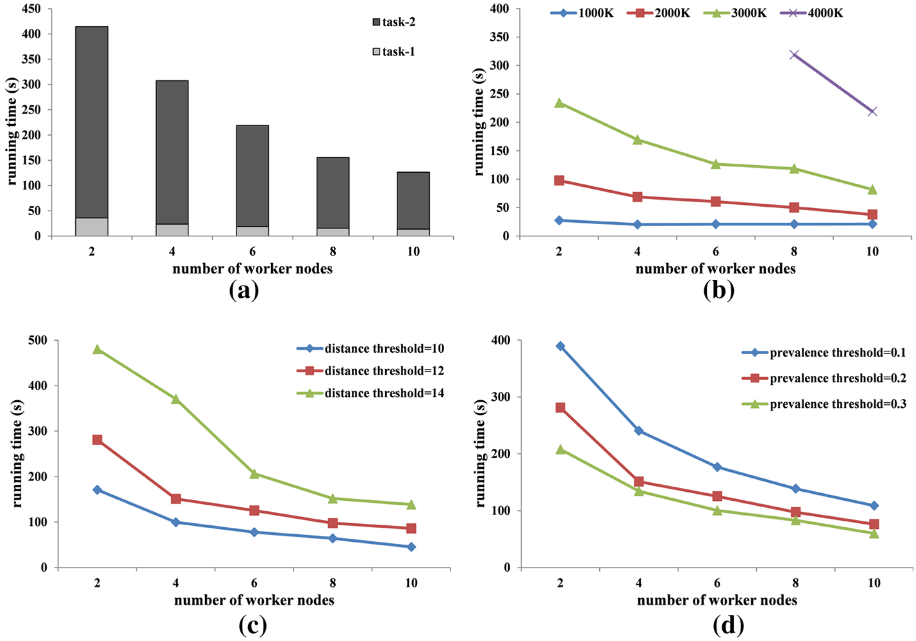


Fig. 13 Scalability experiments: **a** by two main tasks, **b** by the number of instances, **c** by distance thresholds, **d** by prevalence thresholds

threshold is smaller, since the amount of calculation is larger under such parameter settings and the parallel algorithm is more suitable for the huge computation. Similarly, the Amdalh's law takes effect for all experimental settings, but it will be postponed by a larger distance threshold or a smaller prevalence threshold.

In conclusion, the parallel ordered NR-tree-based algorithm further enhances efficiency against with the serial version, and it shows the better scalability for massive spatial datasets.

7 Conclusion and future work

In this paper, we are determined to address the issue of mining co-locations from spatial datasets equipped with or without rare features. At first, we show that existing interest measures suffer from some weaknesses to discover co-locations with rare features, and propose a novel interest measure WPI to identify co-locations with or without rare features. Then, we prove that WPI measure satisfies a conditional anti-monotone property, and develop an efficient approach namely the ordered NR-tree-based algorithm for WPI measure to discover all prevalent co-locations. For processing massive spatial data, the ordered NR-tree-based algorithm is parallelized on MapReduce framework. Lastly, we conduct abundant experiments on both real and synthetic datasets. Experimental results reveal that WPI is superior to other interest measures with the same capabilities. Furthermore, the ordered NR-tree-based algorithm runs significantly faster than the baseline approaches and its parallel version shows better scalability for massive spatial data.

Although WPI measure shows better superiority than other approaches, the distance threshold and prevalence threshold are still required. In particular, the distance threshold is difficult to specify since it is related to the distribution of instances. In future work, we will research how to assist users select suitable thresholds or extend our approach to the version without thresholds. The issue of load balance is common in the distributed parallel algorithm, so it is promising to study some strategies to address the load balance problem existing in the parallel ordered NR-tree-based algorithm. In addition, it is an interesting task to discover co-locations from spatio-temporal data.

Acknowledgements This work is supported by the National Natural Science Foundation of China (61966036, 61662086, 61762090), and the Project of Innovative Research Team of Yunnan Province (2018HC019).

References

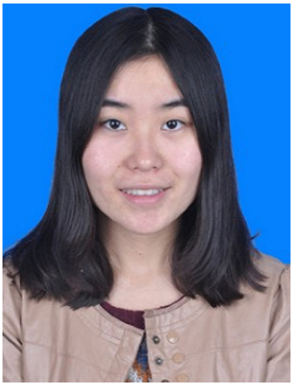
1. Andrzejewski W, Boinski P (2015) Parallel GPU-based plane-sweep algorithm for construction of iCPI-Trees. J Database Manag 26(3):1–20
2. Andrzejewski W, Boinski P (2018) Efficient spatial co-location pattern mining on multiple GPUs. Expert Syst Appl 93:465–483
3. Andrzejewski W, Boinski P (2019) Parallel approach to incremental co-location pattern mining. Inf Sci 496:485–505
4. Barua S, Sander J (2014) Mining statistically significant co-location and segregation patterns. IEEE Trans Knowl Data Eng 26(5):1185–1199
5. Cai J, Liu Q, Deng M, Tang J, He Z (2018) Adaptive detection of statistically significant regional spatial co-location patterns. Comput Environ Urban Syst 68:53–63
6. Chan HK, Long C, Yan D, Wong RC (2019) Fraction-score: a new support measure for co-location pattern mining. In: IEEE international conference on data engineering (ICDE), pp 1514–1525
7. Fang Y, Wang L, Wang X, Zhou L (2017) Mining co-location patterns with dominant features. In: International conference on web information systems engineering (WISE), pp 183–198
8. Feng L, Wang L, Gao S (2012) A new approach of mining co-location patterns in spatial datasets with rare features. J Nanjing Univ Nat Sci 48(1):99–107 ((**in Chinese**))
9. Ge Y, Yao Z, Li H (2021) Computing co-location patterns in spatial data with extended objects: a scalable buffer-based approach. IEEE Trans Knowl Data Eng 33(2):401–414
10. Huang Y, Pei J, Xiong H (2006) Mining co-location patterns with rare events from spatial data sets. GeoInformatica 10(3):239–260
11. Huang Y, Shekhar S, Xiong H (2004) Discovering colocation patterns from spatial data sets: a general approach. IEEE Trans Knowl Data Eng 16(12):1472–1485
12. Li J, Adilmagambetov A, Jabbar MSM, Osornio-Vargas A, Wine O (2016) On discovering co-location patterns in datasets: a case study of pollutants and child cancers. Geoinformatica 20(4):651–692
13. Liu B, Chen L, Liu C, Zhang C, Qiu W (2015) RCP mining: towards the summarization of spatial co-location patterns. In: International symposium on spatial and temporal databases (SSTD), pp 451–469
14. Lu J, Wang L, Fang Y, Li M (2017) Mining competitive pairs hidden in co-location patterns from dynamic spatial databases. In: Pacific Asia knowledge discovery and data mining (PAKDD), pp 467–480
15. Lu J, Wang L, Fang Y, Zhao J (2018) Mining strong symbiotic patterns hidden in spatial prevalent co-location patterns. Knowl Based Syst 146:190–202
16. Ouyang Z, Wang L, Wu P (2017) Spatial co-location pattern discovery from fuzzy objects. Int J Artif Intell Tools 26(2):1750003. https://doi.org/10.1142/S0218213017500038
17. Shekhar S, Huang Y (2001) Discovering spatial co-location patterns: a summary of results. In: International symposium on spatial and temporal databases (SSTD), pp 236–256
18. Wang L, Bao X, Cao L (2018) Interactive probabilistic post-mining of user-preferred spatial co-location patterns. In: IEEE international conference on data engineering (ICDE), pp 1256–1259
19. Wang L, Bao X, Chen H, Cao L (2018) Effective lossless condensed representation and discovery of spatial co-location patterns. Inf Sci 436:197–213
20. Wang L, Bao X, Zhou L (2018) Redundancy reduction for prevalent co-location patterns. IEEE Trans Knowl Data Eng 30(1):142–155
21. Wang L, Bao X, Zhou L, Chen H (2019) Mining maximal sub-prevalent co-location patterns. World Wide Web 22(5):1971–1997
22. Wang L, Bao Y, Lu J, Yip J (2008) A new join-less approach for co-location pattern mining. In: IEEE international conference on computer and information technology (CIT), pp 197–202

23. Wang L, Bao Y, Lu Z (2009) Efficient discovery of spatial co-location patterns using the iCPI-tree. Open Inf Syst J 3(1):69–80
24. Yang P, Wang L, Wang X (2018) A parallel spatial co-location pattern mining approach based on ordered clique growth. In: International conference on database systems for advanced applications (DASFAA), pp 734–742
25. Yang P, Wang L, Wang X (2019) An effective approach on mining co-location patterns from spatial databases with rare features. In: IEEE international conference on mobile data management (MDM), pp 53–62
26. Yang P, Wang L, Wang X, Fang Y (2018) A parallel joinless algorithm for co-location pattern mining based on group-dependent shard. In: International conference on web information systems engineering (WISE), pp 240–250
27. Yang P, Zhang T, Wang L (2018) TSRS: trip service recommended system based on summarized co-location patterns. In: APWEB/WAIM, pp 451–455
28. Yao X, Chen L, Peng L, Chi T (2017) A co-location pattern-mining algorithm with a density-weighted distance thresholding consideration. Inf Sci 396:144–161
29. Yoo JS, Boulware D, Kimmey D (2020) Parallel co-location mining with MapReduce and NoSQL systems. Knowl Inf Syst 62:1433–1463
30. Yoo JS, Shekhar S (2004) A partial join approach for mining co-location patterns. In: the 12th Annual ACM international workshop on geographic information systems, pp 241–249
31. Yoo JS, Shekhar S (2006) A joinless approach for mining spatial colocation patterns. IEEE Trans Knowl Data Eng 18(10):1323–1337
32. Yu W (2016) Spatial co-location pattern mining for location-based services in road networks. Expert Syst Appl 46:324–335

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Peizhong Yang received the B.S. degree in Computer Science from Yunnan University, in 2016. He has been working toward the Ph.D. degree in Computer Science and Technology at Yunnan University, since 2016. His main research interest is spatial data mining and parallel computing.

Lizhen Wang received the B.S. and M.Sc. degrees in computational mathematics from Yunnan University, in 1983 and 1988, respectively, and the PhD degree in computer science from the University of Hundersfield, UK, in 2008. She is a professor, PhD supervisor at the School of Computer Science and Engineering, Yunnan University. Her research interests include spatial data mining, interactive data mining, big data analytics, and their applications.



Xiaoxuan Wang was born in 1991, Ph.D. candidate. She received the B.S. degree in computer science from Minzu University of China, in 2014. Her current research interests include spatial database and Data mining.

Lihua Zhou received the B.S. and M.Sc. degrees in electronics and information system from Yunnan University, in 1989 and 1992, respectively, and the PhD degree in communication and information system from Yunnan University, in 2010. She is a professor, PhD supervisor in the Department of Computer Science and Engineering, Yunnan University. Her main research interests include data mining, machine learning and social network analysis.