Mitigating Weight Explosion in Rare-Feature Co-location Pattern Mining: A Cauchy Distribution-Based Approach

Vanhiep Duong, Vanthuc Nguyen, Xuankhang Pham, Thanhduy Pham, and Vanha Tran(✉)

FPT University, 155514 Hanoi, Vietnam
{hiepdvhe181185,thucnvhe181431,khangpxhe180735,duypthe181990}@fpt.edu.vn, hatv14@fe.edu.vn

Abstract. Spatial co-location pattern (SCP) mining aims to identify groups of spatial features whose instances tend to appear near one another. A key difficulty is evaluating patterns that include rare features, particularly when feature frequencies are highly imbalanced. In such cases, the traditional participation index (PI) can be too strict and may overlook rare but meaningful spatial associations. While the proposed rarity-aware measures, they frequently utilize exponential weighting functions that provide insufficient compensation under moderate imbalance and suffer from numerical instability or excessive amplification under extreme disparity. To address these limitations, we propose a novel Cauchy distribution-based weighting framework for prevalence evaluation of SCPs. By defining a Cauchy Rare Intensity, we develop the Cauchy Weighted Participation Index (WPI_C), which offers more stable and proportionate rarity compensation across different imbalance regimes. To enhance mining efficiency, this measure is integrated into a clique-based framework that employs an optimized maximal clique enumeration and hash table-base query strategy. Extensive experiments on multiple real-world datasets demonstrate that our proposed method significantly improves the discovery of rare-feature co-locations while maintaining robust and efficient mining performance.

Keywords: Spatial data mining · Co-location pattern · Rare features · Cauchy Distribution · Maximal clique

1 Introduction

Today, with the rapid development of positioning technologies, massive amounts of data with coordinate information are generated and stored every day. Discovering valuable knowledge from such massive spatial data is extremely important in many fields, such as urban facility analysis, environmental monitoring, and geographic relationship mining [7,9,11]. Many data mining techniques have been

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2027
O. Gervasi et al. (Eds.): ICCSA 2026 Workshops, LNCS 16761, pp. 493–508, 2027.
https://doi.org/10.1007/978-3-032-30527-5_32

proposed, and among them, spatial co-location pattern (SCP) mining refers to finding a set of spatial features from a dataset whose instances frequently appear together in close geographic proximity, typically forming cliques. Traditionally, SCP mining relies on the participation index (PI) [7] to measure pattern prevalence, which is determined by the minimum participation ratio (PR) among all constituent features. However, this strictly minimum-based metric struggles significantly when spatial datasets contain rare features. A feature is rare if the number of its instances is obviously less than that of other features [13].

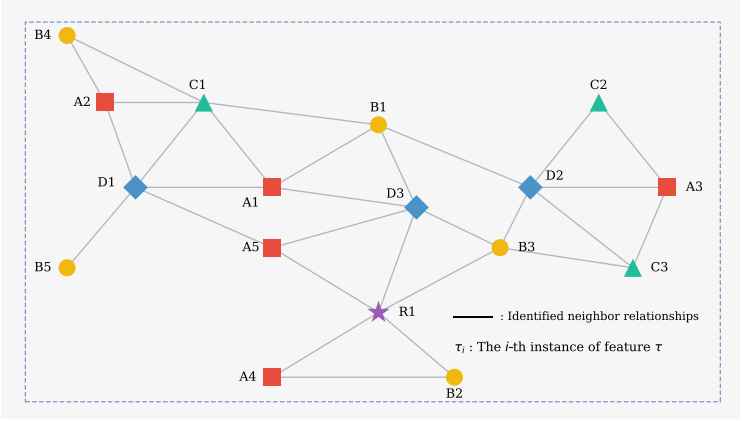


Fig. 1. An illustrative spatial dataset for SCP mining with varying feature frequencies and identified neighbor relationships.

Figure 1 depicts an illustrative spatial dataset containing five features with varying instance counts: $|\mathrm{A}| = 5$, $|\mathrm{B}| = 5$, $|\mathrm{C}| = 3$, $|\mathrm{D}| = 3$, and a strictly rare feature $|\mathrm{R}| = 1$. A user sets a distance threshold ε to determine the neighbor relationships between instances, connected by solid lines. When applying the traditional SCP mining technique to evaluate the candidate combination $p = \{\mathrm{A}, \mathrm{D}, \mathrm{R}\}$, the identified neighbor relationships show that the sole instance R_1 is perfectly co-located with A_5 and D_3. Thus, the participation ratio of the rare feature R is $PR(\mathrm{R}|p) = 1/1 = 1.0$, while the ratios for the other features are $PR(\mathrm{D}|p) = 1/3 \approx 0.33$, and $PR(\mathrm{A}|p) = 1/5 = 0.20$. Consequently, the overall prevalence degree of this pattern is $PI(p) = \min\{0.20, 0.33, 1.0\} = 0.20$. If a user sets a minimum prevalence threshold *min_prev* (e.g., *min_prev* $= 0.30$), this highly meaningful rare-feature pattern would be prematurely pruned, even though the rare feature R relies entirely on this spatial association.

The example in Fig. 1 is merely a miniature representation of real-world datasets. The actual data contains a massive amount of instances, and the number of instances belonging to each feature varies drastically. To capture such valuable rare-feature co-locations, recent studies introduced the Weighted Participation Index (WPI) [13], which compensates for feature-frequency disparity

by utilizing a Gaussian kernel function to assign relative weights. While conceptually innovative, the exponential nature of the Gaussian weight ($w \propto \exp(v^2)$, where v is the frequency disparity ratio) introduces critical mathematical flaws when applied to massive spatial data [13]. For example, when the illustrative case is scaled to a more realistic setting with $|A| = 1000$ and $|R| = 10$, the frequency disparity becomes $v = 100$. When this value is used in the Gaussian weighting function, the resulting weight can become extremely large, even exceeding 10^{18}. Such a value may lead to numerical instability and can over-amplify very small participation ratios. Another limitation appears under moderate imbalance. For a pattern such as $p' = \{D, R\}$, where $|D| = 30$ and $|R| = 10$ ($v = 3$), the Gaussian weight remains close to 1.0. As a result, the method provides little adjustment and may still miss moderately rare but meaningful co-location patterns. Consequently, the algorithm offers virtually no compensation, easily leading to the loss of interesting moderately rare patterns.

This bipolar behavior makes the Gaussian-based WPI highly unreliable for datasets with skewed distributions. Therefore, this paper designs a novel weighting framework based on the Cauchy distribution to overcome these limitations. By featuring a smoother, heavy-tailed response curve, the Cauchy distribution-based approach accelerates earlier to adequately rescue moderately imbalanced features while growing polynomially rather than exponentially at the tail to definitively prevent weight explosion. To further ensure high mining efficiency, we integrate this robust prevalence measure into a clique-based mining framework. The specific contributions of this paper are as follows:

- We propose a Cauchy distribution-based rare-feature weighting framework, define a Cauchy Rare Intensity factor, and develop the Cauchy Weighted Participation Index (WPI_C) for robust prevalence evaluation in SCP mining. This design addresses two main limitations of exponential weighting: weight explosion under severe imbalance and weak compensation under moderate imbalance.
- We incorporate the proposed prevalence measure into a clique–hash-table-based SCP mining pipeline. The pipeline uses a hybrid Bron–Kerbosch-style maximal clique enumeration strategy, following prior graph mining studies [2,8], to support efficient pattern discovery.
- We evaluate the proposed method on several real-world datasets and compare it with existing baselines in terms of discovery effectiveness, numerical stability, and performance under highly imbalanced feature-frequency distributions.

2 Related Work

SCP mining is related to traditional frequent itemset mining, but it is more difficult because spatial instances are linked through neighborhood relationships rather than transaction records. Shekhar and Huang first introduced the SCP

mining problem and proposed the participation index (PI) to evaluate the prevalence of a co-location pattern [7,9]. Since then, several algorithmic frameworks have been developed to improve mining efficiency. Among them, clique-based methods have been widely used because they model neighborhood relations as graph structures [1,12]. By applying maximal clique enumeration, these methods can reduce redundant candidate generation. Bron–Kerbosch-type algorithms and their improved variants are commonly used for this task in graph mining [2,5,8]. Hybrid strategies that adapt to the structure of real-world graphs have also been shown to improve performance in dense neighborhoods [8].

The PI metric is commonly used for prevalence evaluation because it satisfies the anti-monotone property, which helps prune the search space. However, PI takes the minimum participation ratio among all features in a pattern. This makes it conservative when feature frequencies are highly imbalanced, since a low participation ratio from a frequent feature can cause a rare-feature pattern to be discarded. To address this problem, several alternative measures have been proposed. For example, MaxPR relaxes the strict bottleneck effect of PI to better support rare-feature discovery [6]. More recently, the Weighted Participation Index (WPI) introduced a rarity-aware weighting mechanism based on a Gaussian kernel function.

Although these methods improve SCP mining from different perspectives, they do not fully consider how the weighting function behaves under different imbalance levels. As discussed in Sect. 1, exponential weighting may provide limited compensation under moderate imbalance but can produce excessively large weights under extreme imbalance. This motivates the use of a more stable weighting function for rare-feature co-location mining. The literature still lacks a prevalence function explicitly designed to balance sensitivity and control. Therefore, in the following sections, we introduce a Cauchy distribution-based weighting framework and integrate it into an efficient clique-based SCP mining framework to overcome these shortcomings.

3 The Proposed Algorithms

3.1 Basic Concepts and Problem Formulation

A spatial dataset, denoted as $\mathcal{O}$, comprises a finite collection of spatial objects. Each object $x \in \mathcal{O}$ is characterized by a specific feature type $\tau \in \mathcal{T}$ and a geographical coordinate $\mathbf{s}(x)$, where $\mathcal{T} = \{\tau_1, \tau_2, \ldots, \tau_m\}$ represents the universal set of distinct features. The subset of objects belonging to a specific feature τ is denoted by $\mathcal{O}_\tau$, with its cardinality given by $cnt(\tau) = |\mathcal{O}_\tau|$.

Definition 1 (Neighborhood and Co-location Instance). *A spatial neighborhood relationship, denoted as $x \sim_\varepsilon y$, exists between two objects x and y if their spatial distance satisfies a given threshold ε under a specific metric (e.g., Euclidean distance) [7]. Given a candidate SCP, $p \subseteq \mathcal{T}$, ($|p| \geq 2$), a strict type-unique instance of p is defined as a subset of objects $S \subseteq \mathcal{O}$ satisfying two structural constraints: (1) $|S| = |p|$, ensuring a bijective mapping between the*

objects in S and the feature types in p, and (2) all objects in S form a spatial clique (i.e., they are pairwise neighbors) [9]. The set of all such instances for pattern p is denoted as $\mathcal{T}(p)$.

Definition 2 (Participation Ratio and Participation Index). *To evaluate the significance of a pattern, we quantify the contribution of each individual feature. The Participation Ratio (PR) of a feature $\tau \in p$ is the fraction of its objects that appear in at least one valid instance of p:*

$$PR(\tau \mid p) = \frac{|\mathcal{O}_\tau^p|}{cnt(\tau)} \tag{1}$$

where $\mathcal{O}_\tau^p = \{x \in \mathcal{O}_\tau \mid \exists S \in \mathcal{I}(p), x \in S\}$. Consequently, the Participation Index (PI) of the entire pattern p is bounded by its least participating feature:

$$PI(p) = \min_{\tau \in p}\{PR(\tau \mid p)\} \tag{2}$$

A pattern p is deemed a Prevalent Co-location Pattern (PCP) if $PI(p) \geq min_prev$, where min_prev is a user-specified threshold.

3.2 Robust Cauchy Distribution-Based Interest Measure

To alleviate the strict bottleneck of PI under heavy-tailed feature distributions, recent works apply exponential weights. However, as frequency discrepancies grow infinitely large, exponential kernels succumb to "weight explosion," generating excessive false positives. To overcome this mathematical bottleneck, we propose shifting the weighting mechanism to the Lorentzian form of the Cauchy distribution kernel [3].

Definition 3 (Original Weighted Participation Index [13]). *To compensate for the rare-feature dilemma, the traditional Weighted Participation Index (WPI) evaluates the prevalence of a co-location pattern p by incorporating a rarity-aware weighting mechanism. The metric is defined as:*

$$WPI(p) = \min_{\tau \in p}\{PR(\tau \mid p) \cdot weight(\tau, p)\} \tag{3}$$

where $weight(\tau, p)$ is the dynamic weight assigned to feature τ, traditionally formulated using a Gaussian kernel function to exponentially amplify the participation ratio of rare features:

$$weight(\tau, p) = \exp\left(\frac{(r(\tau, p) - 1)^2}{2\kappa^2}\right) \tag{4}$$

with $r(\tau, p)$ representing the relative frequency disparity of the feature within the pattern, and κ serving as the global dispersion parameter of the dataset.

Definition 4 (Global Pairwise Dispersion). *Assuming the feature types in $\mathcal{T}$ are indexed in ascending order of their frequencies such that $cnt(\tau_1) \leq cnt(\tau_2) \leq \ldots \leq cnt(\tau_m)$, the global dispersion parameter κ is formulated as the arithmetic mean of all pairwise frequency ratios:*

$$\kappa = \frac{2}{m(m-1)} \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} \frac{cnt(\tau_j)}{cnt(\tau_i)} \tag{5}$$

Functionally, κ serves as a global bandwidth parameter that encapsulates the intrinsic frequency skewness across the entire spatial domain.

Definition 5 (Cauchy Rare Intensity). *For any candidate pattern p, let $\tau_{\min} = \arg\min_{\tau \in p} cnt(\tau)$ be its limiting (rarest) feature. We quantify the relative abundance of any feature $\tau \in p$ using the within-pattern ratio $r(\tau, p) = cnt(\tau)/cnt(\tau_{\min})$. To model the statistical rarity of a feature without aggressive penalization, we define its Rare Intensity (RI_C) using the Cauchy distribution:*

$$RI_C(\tau, p) = \frac{1}{1 + \left(\frac{r(\tau,p)-1}{\kappa}\right)^2} \tag{6}$$

Definition 6 (Cauchy Distribution-based Weighting). *Unlike the exponential function used in the original WPI (Eq. 2), our framework calculates the feature weight $\omega(\tau, p)$ simply as the reciprocal of the Cauchy Rare Intensity:*

$$\omega(\tau, p) = \frac{1}{RI_C(\tau, p)} = 1 + \left(\frac{r(\tau, p) - 1}{\kappa}\right)^2 \tag{7}$$

Weight Analysis and Mathematical Properties: The polynomial formulation (Eq. 5) resolves two main drawbacks of the Gaussian kernel (Eq. 2):

- **Asymptotic Stability (Preventing Weight Explosion):** Extreme frequency imbalances (large r) cause exponential functions to output unmanageable values. Given a global dispersion $\kappa = 8$ and a ratio $r(\tau, p) = 81$, the Gaussian kernel yields $\exp(10^2) \approx 2.68 \times 10^{43}$. This inflation pushes random noise above the prevalence threshold. The Cauchy distribution avoids this via its heavy-tailed polynomial decay [4]. Under the same parameters, our Cauchy weight is strictly bounded at $\omega = 1 + 10^2 = 101$. This naturally limits the maximum influence of rare features and prevents false positives.
- **Early Sensitivity (Recovering Missed Patterns):** Exponential curves offer minimal compensation for moderate discrepancies (small r). Consequently, meaningful spatial patterns with slight imbalances are often pruned. The Cauchy function grows more steadily early on. In our evaluation of a moderately rare pair, the Gaussian model produced a weight of 1.08, leaving the pattern below the threshold. Our framework assigned a weight of 1.16. This adjustment is sufficient to retain valid spatial relationships.

Definition 7 (Cauchy Weighted Participation Index). *Replacing the exponential weight with our robust Cauchy weight, we define the Cauchy Weighted Participation Ratio (WPR_C) by multiplying a feature's raw PR with its dynamically calculated Cauchy weight:*

$$WPR_C(\tau \mid p) = PR(\tau \mid p) \cdot \omega(\tau, p) \tag{8}$$

A spatial co-location pattern is considered prevalent when its Cauchy Weighted Participation Index (WPI_C) reaches or exceeds the threshold min_prev. We define $WPI_C(p)$ as the bottleneck WPR_C across all features present in p:

$$WPI_C(p) = \min_{\tau \in P} WPR_C(\tau \mid p) \tag{9}$$

We formulate the Cauchy Weighted Participation Ratio (WPR_C) by scaling the raw PR with the proposed Cauchy weight:

Impact of Cauchy Weight on Feature Selection. Equation (6) avoids the instability of exponential scaling. The Cauchy kernel preserves the statistical validity of the prevalence score. In traditional models, inflated weights often eclipse the actual PR. Even negligible, coincidental co-occurrences (e.g., $PR = 10^{-5}$) are artificially pushed above the *min_prev* threshold by weights like 10^{43}, generating severe noise. By replacing the Gaussian kernel with $\omega(\tau, p)$ to ensure a predictable polynomial growth curve while providing early sensitivity, WPI_C guarantees that rare-feature compensation is proportionate across all levels of imbalance. The raw PR thus retains its discriminative power: a pattern achieves prevalence only if it exhibits genuine spatial proximity coupled with a mathematically sound weight boost. This optimally filters out frequency-induced noise while recovering moderately rare patterns that the traditional WPI discards.

3.3 The Designed Mining Algorithm

After establishing neighbor relationships, data points are represented as an undirected graph $G(V, E)$ with vertices $V \equiv \Psi$ and edges E. Each edge is formed between any two vertices whose instances meet the neighborhood condition.

Definition 8 (Maximal Clique, MC). *A subset of instances $cl = \{o_1, \ldots, o_l\}$ is called a clique if all instances have a neighboring relationship. Furthermore, a clique cl is defined as a maximal clique if it cannot be included by any larger clique cl' ($cl \subset cl'$).*

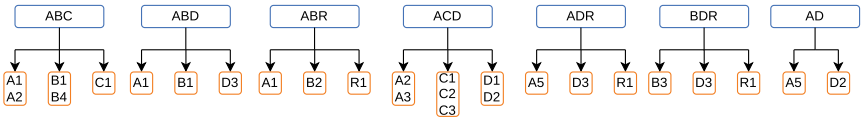
This study focuses on using MCs to optimize the SCP mining process rather than creating a new MC enumeration. Although enumerating MCs is an NP-hard problem, it still is an appeal domain with numerous established solutions have been proposed. In particular, hybrid strategies that adapt clique enumeration to the structural properties of real-world graphs have been shown to improve efficiency in dense neighborhoods. For the purposes of this work, we employ the MCEHybrid algorithm, which is the hybrid approach of BK_{rcd} and BK_{pivot}, to

identify maximal cliques. Detailed mechanics of this algorithm are omitted due to space constraints. However, comprehensive explanations can be found in [8].

To demonstrate Definition 8, we utilize the toy spatial dataset depicted in Fig. 1. Once the neighborhood relationships among the instances are materialized, the underlying structure is captured as a neighbor graph. By executing the MCEHybrid algorithm on this graph, we can extract all maximal cliques as shown in Fig. 2(a).

$\{A_1, B_1, C_1\}$	$\{A_2, B_4, C_1\}$	$\{A_1, B_1, D_3\}$	$\{A_1, B_2, R_1\}$	$\{A_2, C_1, D_1\}$
$\{A_3, C_3, D_2\}$	$\{A_5, D_3, R_1\}$	$\{B_3, D_3, R_1\}$	$\{A_5, D_2\}$	$\{A_3, C_2, D_2\}$

(a)



(b)

Fig. 2. The constructed participating instance hash table.

Definition 9 (Participating Instance Hash Table). *A participating instance hash table is defined as a hash-map structure which the keys represent feature sets extracted from instances within maximal cliques, and the corresponding values hold the actual instances of those cliques.*

Figure 2(b) illustrates the generated participating instance hash table from all the MCs extracted in Fig. 1. Suppose we are evaluating a candidate pattern $p = \{A, B\}$. To calculate the participating instances for the features in p, we identify all keys in the hash table that are supersets of {A, B}. By accessing the pattern {A, B, C}, {A, B, D} and {A, B, R} associated with these matching keys, we can extract the valid co-location instances $\{A_1, B_1\}$, $\{A_2, B_4\}$ and $\{A_1, B_2\}$. Aggregating these instances return the complete set of participating instances for candidate p.

Algorithm 1 is the pseudo-code of the designed mining SCP algorithm, named MCR. The detail of MCR is described as below.

Initialization (Steps 1–7). Initially, the features within the dataset are counted and sorted to compute their dispersion (Step 1–2). Following this, the neighbor graph among instances are constructed based on a user-specified threshold ϵ (Step 3). Subsequently, the MCEHybrid algorithm, is applied to enumerate the MCs (Step 4). Finally, a PriorityQueue extracts all keys from the hash table to form the initial candidate set for the mining process (Step 5).

Candidate Pattern Evaluation and Generation (Step 8–17). The algorithm iteratively processes the Pattern Queue until it is empty. In each iteration,

it extracts the highest-priority candidate pattern p (Step 9). To prevent redundant computations, the algorithm checks if p is already in the processed set or not. If so, it skips to the next iteration (Step 10–12). Otherwise, p is added to the processed set. After that, the participating instances of p is retrieved by querying the hashmap (Step 14). Following this, it calculates the rare intensity (RI_c) and uses it to compute the weighted participating instances (WPI_c) for the candidate (Step 15–16). Finally, the algorithm derives a new set of sub-patterns from p to be evaluated in subsequent steps (Step 17).

Algorithm 1: MCR Colocation Mining

Input: Spatial instance set $\mathcal{O}$, Feature set $\mathcal{T}$, Distance threshold ϵ, Prevalence threshold min_prev

Output: Set of spatial co-location patterns with rare features($SCPs$)

1 $feat_freq \leftarrow$ `CountAndSortFeatures`$(\mathcal{O}, \mathcal{T})$;
2 $\kappa \leftarrow$ `CalculateDispersion`$(feat_freq)$;
3 $G(V, E) \leftarrow$ `BuildNeighborGraph`$(\mathcal{O}, \epsilon)$;
4 $clique_map \leftarrow$ `ExecuteMCEHybrid`$(G(V, E))$;
5 $PatternQueue \leftarrow$ `PriorityQueue`(`ExtractCandidates`$(clique_map)$);
6 $PCPs \leftarrow \emptyset$;
7 $SCPs \leftarrow \emptyset$;
8 $processed \leftarrow \emptyset$;
9 **while** *PatternQueue is not empty* **do**
10 $p \leftarrow PatternQueue$.`pop_max()`;
11 **if** $p \in processed$ **then**
12 **continue**
13 **end**
14 $processed \leftarrow processed \cup \{p\}$;
15 $PartI \leftarrow$ `QueryInstances`$(p, clique_map)$;
16 $RI_c \leftarrow$ `CalcRareIntensity`$(p, feat_freq, \kappa)$;
17 $WPI_c \leftarrow$ `ComputeWeightedPI`$(PartI, P, RI_c, feat_freq)$;
18 $newCs \leftarrow$ `DeriveSubPatterns`(p);
19 **if** $WPI_c \geq min_prev$ **then**
20 $PCPs \leftarrow PCPs \cup \{p\}$;
21 $InferredPCPs \leftarrow$ `InferPrevalentSubsets`$(newCs, p, feat_freq)$;
22 $SCPs \leftarrow SCPs \cup \{p\} \cup InferredPCPs$;
23 $newCs \leftarrow newCs \setminus InferredPCPs$;
24 **end**
25 **foreach** $subset \in newCs$ **do**
26 **if** $subset \notin processed$ **then**
27 $PatternQueue$.`push`$(subset)$;
28 **end**
29 **end**
30 **end**
31 **return** $SCPs$;

Select Prevalent Co-locations (Lines 18–30). The algorithm evaluates whether the WPI_c of candidate meets the minimum prevalence threshold (Step 18). If the condition is satisfied, the candidate p is added to PCPs. The algorithm then infers prevalent subsets via InferPrevalentSubsets, and accumulates all confirmed patterns—both directly evaluated and inferred—into SCPs (Steps 20–23). The final output SCPs thus contains the complete set of discovered rare-feature co-location patterns. Finally, the algorithm iterates through these remaining subsets. Any subset that has not yet been processed is pushed into the candidate queue for subsequent processing (Step 24–30).

4 Experiments and Analysis

This section presents a comprehensive evaluation of the proposed algorithm. The evaluation is conducted between three baseline methods: PI [14], MaxPR [6], and the original WPI method with the ordered NR-tree structure [13].

The PI-based baseline used the prevalence-driven approach which evaluated the pattern significance through the participation index and leverages its anti-monotone property for candidate pruning. MaxPR focused on co-location patterns with rare features but has limited effectiveness due to the weak constraints of anti-monotonic property. The original WPI method improved rare-feature discovery by using a Gaussian weighting method and utilizes ordered NR-tree approach for retrieving instances faster. Generally, these three methods highlighted varying strategies for mining spatial co-location, particularly rare features, and established robust frameworks for comparison.

All algorithms are implemented and executed on a computer with the configuration of an Intel(R) Core i7-12700H CPU, 16 GB of main memory, and running in Windows 11 operating system.

4.1 Datasets

Our approach is evaluated on four real world datasets namely Las Vegas, Toronto, Beijing, and Shanghai. The Las Vegas dataset is extracted from the Yelp Open Dataset[1], including varying venue categories in the United States. The remaining three datasets are constructed from public Point-of-Interest (POI) sources [1,10,11] and undergo preprocessing processes to eliminate invalid records.

Table 1 illustrates the key features of each dataset. Beyond basic statistics which are number of feature types, total instances, and spatial coverage, we have the global dispersion κ as defined in Definition 5. This feature captures the difference in instance counts across all feature pairs in the dataset. A higher κ value indicates the larger imbalance in feature frequencies, reflecting more pronounced rare-feature characteristics in the dataset. In contrast, a lower κ implies a relatively uniform distribution. The four datasets provide a wide range

[1] https://business.yelp.com/data/resources/open-dataset.

Table 1. Datasets characteristics

Name	# Features	# Instances	Spatial Coverage	κ
Las Vegas	17	22,725	$38{,}649 \times 62{,}985$ km^2	2.90
Toronto	20	17,129	$23{,}417 \times 56{,}036$ km^2	7.86
Shanghai	14	41,454	$65{,}557 \times 113{,}690$ km^2	13.22
Beijing	19	104,909	138.7×227 km^2	16.74

of κ values (from 2.90 to 16.74), which providing a comprehensive element for assessing the behavior of rare-feature discovery strategies under varying degrees of skewness.

4.2 Effectiveness in Pattern Discovery

To evaluate the effectiveness of the proposed method in practical scenarios, we analyze a subset of representative patterns extracted from the Toronto dataset (represented in Table 1) under the distance threshold $\epsilon = 120$ m and minimum prevalence threshold $min_prev = 0.2$. Table 2 shows a detailed comparison of these patterns to illustrate the advantages of our Cauchy-based approach over WPI approach. We analyze the patterns in two different categories; those new patterns (*Added*) and those correctly pruned (*Removed*) by our algorithm.

Table 2. The differences between patterns discovered by WPI and the proposed method.

Group	Pattern	WPI		Ours	
		WPRs	Weights	WPRs	Weights
Added	{R, E}	[0.4661, 0.1894]	[1.0, 2.08]	[0.4661, **0.2245**]	[1.0, **2.46**]
	{S, F}	[0.6000, 0.1902]	[1.0, 1.46]	[0.6000, **0.2289**]	[1.0, **1.76**]
Removed	{T, G, C}	[0.2857, 0.5986, 4.60×10^{10}]	[1.0, 27.35, $\mathbf{7.19}\times10^{12}$]	[0.2857, 0.1667, 0.3848]	[1.0, 7.62, **60.21**]
	{T, G, D}	[0.2500, 0.6447, 2.10×10^{6}]	[1.0, 27.35, $\mathbf{1.82}\times\mathbf{10^8}$]	[0.2500, 0.1795, 0.4510]	[1.0, 7.62, **39.04**]

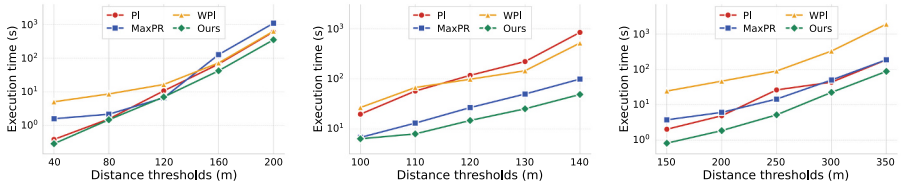
The *Added* group highlights the additional rare-feature co-location pattern that are not captured by the traditional WPI under the frequency imbalance. For example, in pattern {S, F}, the traditional WPI assigns a smaller weight of 1.46 to the rare feature, leading to a minimum WPR of 0.1902. Because this prevalence score falls below the minimum threshold ($min_prev = 0.2$), this co-location is missed. By contrast, our proposed method illustrates early sensitivity by assigning a significantly higher weight of 1.76. This decisive boost the pattern's final prevalence score to 0.2289, which is higher than the required minimum threshold, resulting in the recovered of the pattern {S, F}.

In contrast, the *Removed* group exposes the numerical instability of the exponential Gaussian kernel under extreme disparity. With the patterns like {T, G, C} and {T, G, D}, the traditional WPI generates an extremely large weights

for rare features C and D with 7.19×10^{12} and 1.82×10^{8}, respectively. These massive numbers completely hide the true spatial participation of the instances. When the raw PR is multiplied by such extreme weights, the true spatial proximity becomes completely irrelevant. The mathematical artifact makes the WPR extremely high (for example 4.60×10^{10}). As a result, random noise patterns go well above the 0.2 threshold although they have very little spatial importance. By proposing a heavy-tailed polynomial curve, our Cauchy based framework completely resolve this explosion. The extreme weight for feature C is reduced to 60.21 while the weight for feature D is reduced to 39.04. Consequently, the true prevalence of the pattern {T, G, C} is correctly evaluated with 0.1667, allowing our algorithm to identify and prune the patterns that the traditional WPI promotes falsely.

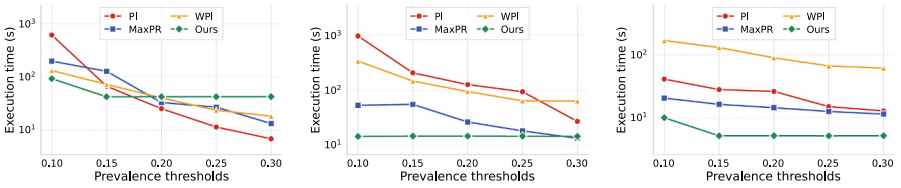
4.3 Computational Efficiency: Time and Memory

Efficient in the Execution Time. Figures 3 and 4 illustrates the execution time under varying distance thresholds ϵ and prevalence thresholds min_{prev}. As shown in Fig. 3, a large ϵ causes the runtime of all methods to rise significantly. The reason for this is that the more neighboring relationships are established, the larger number of co-location instances and an expanded candidate search space. PI and MaxPR, which are built based on the joinless approach, are heavily affected. Specifically, their candidate sets and row instances grow rapidly as the neighborhood expands, leading to considerably increases in runtime. Besides, WPI mitigates some redundant join operations through its ordered NR-tree structure. However, constructing and querying this index still introduces considerable time overhead. Under different min_{prev} (Fig. 4), a stricter the prevalence condition prunes the search space more effectively, decreasing the runtime for all algorithms. PI and MaxPR still generate a substantial number of intermediate candidates due to their exhaustive level-wise traversal, while WPI incurs additional computational cost from maintaining ordered neighborhood structures. In contrast, our method, which stores maximal cliques in a hash map, remains stable across both parameter settings. This structure avoids redundant instance enumeration and enables efficient candidate evaluation.



(a) Las Vegas (min_{prev} = 0.15) (b) Toronto (min_{prev} = 0.2) (c) Shanghai (min_{prev} = 0.2)

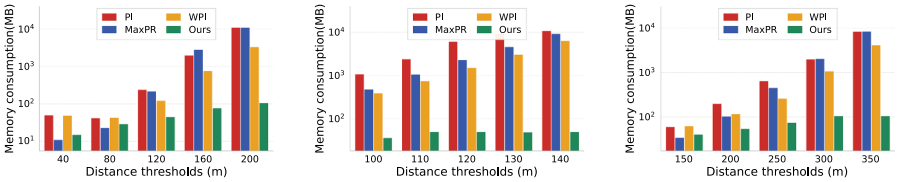
Fig. 3. Comparison of execution time under varying distance thresholds (ϵ).



(a) Las Vegas ($\epsilon = 160$m) (b) Toronto ($\epsilon = 120$m) (c) Shanghai ($\epsilon = 250$m)

Fig. 4. Comparison of execution time under varying prevalence thresholds (min_{prev}).

Efficient in the Memory Consumption. Figures 5 and 6 report the memory consumption under varying distance thresholds (ϵ) and prevalence thresholds (min_{prev}), respectively. Overall, a similar trend is observed between memory consumption and runtime. When ϵ increases (Fig. 5), PI and MaxPR must retain large rowsets to support their join operations, resulting in rapid memory growth and making them the most memory consumption methods. WPI generally uses less memory than PI and MaxPR but still incurs significant overhead. Under varying min_{prev} (Fig. 6), the memory footprint of all baselines decreases because fewer patterns survive under the stricter prevalence conditions. In contrast, our approach demands significantly less memory than the baselines across all parameter settings and datasets. This is because our method stores maximal cliques in a hash map and retrieves participating instances directly based on key lookup. This method avoids the materialization of large intermediate tables. This advantage allows our method to operate with an small and stable memory footprint, even when the search space expands significantly.

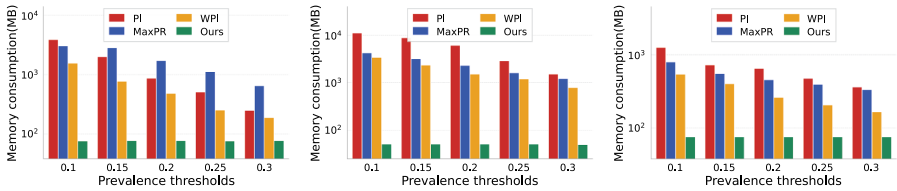


(a) Las Vegas (min_{prev}=0.15) (b) Toronto (min_{prev} = 0.2) (c) Shanghai (min_{prev} = 0.2)

Fig. 5. Memory consumption comparison under varying distance thresholds (ϵ).

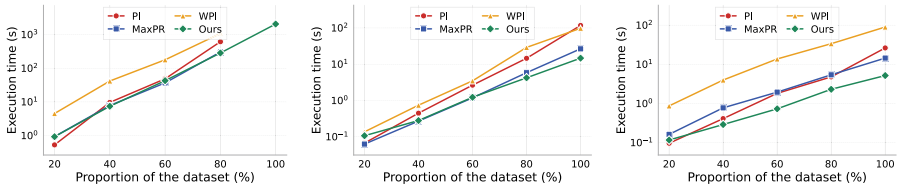
4.4 Scalability Analysis

Figures 7 and 8 report how execution time and memory scale as the dataset proportion grows from 20% to 100%. Across all algorithms, both metrics increase with data size, driven by the rapid expansion of neighboring relationships and



(a) Las Vegas ($\epsilon = 160$m) (b) Toronto ($\epsilon = 120$m) (c) Shanghai ($\epsilon = 250$m)

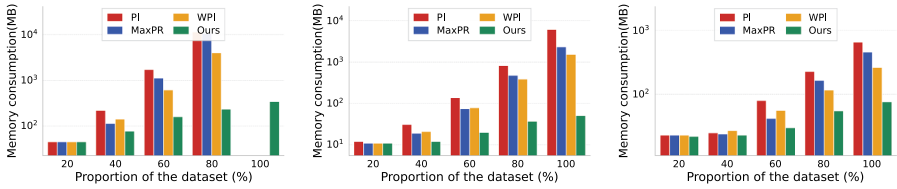
Fig. 6. Memory consumption comparison under varying prevalence thresholds (min_{prev}).



(a) Beijing ($\epsilon = 250$m) (b) Toronto ($\epsilon = 120$m) (c) Shanghai ($\epsilon = 250$m)

Fig. 7. Scalability of execution time across three datasets with varying proportions of instances (from 20% to 100%, all cases $min_{prev} = 0.2$).

the resulting growth in candidate patterns. PI and MaxPR are most adversely affected, as their joinless level-wise paradigm requires maintaining and extending large rowsets proportional to the number of instances. WPI reduces some of this overhead through its ordered NR-tree, but tree traversal and maintenance become increasingly costly at larger scales. Our method exhibits the most gradual growth in both time and memory, attributable to the hash-map-based maximal clique storage, which enables direct instance retrieval and avoids repeated traversal of neighborhood structures as data volume increases. Overall, these results confirm that the proposed approach achieves superior scalability relative to both joinless-based and tree-based competitors.



(a) Beijing ($\epsilon = 250$m) (b) Toronto ($\epsilon = 120$m) (c) Shanghai ($\epsilon = 250$m)

Fig. 8. Scalability of memory consumption across three datasets with varying proportions of instances (from 20% to 100%, all cases $min_{prev} = 0.2$).

5 Conclusion

This paper addressed spatial co-location pattern mining under severe feature-frequency imbalance. To overcome the weight explosion and insufficient compensation flaws of traditional exponential weighting (WPI), we proposed a robust Cauchy Distribution-based prevalence measure (WPI_C). By leveraging a heavy-tailed polynomial formulation, WPI_C prevents numerical instability under extreme disparity while preserving early sensitivity to moderately rare features. Furthermore, we integrated this measure into an efficient maximal-clique enumeration framework to improve scalability. Extensive experiments demonstrated that our method successfully discovers more informative rare-feature patterns than existing approaches while maintaining superior computational efficiency.

However, relying on a global dispersion measure limits the framework's adaptability to highly variable local skewness and dynamic environments. In future work, we will study locally adaptive parameterization, extend the framework to find spatio-temporal co-location patterns, and explore mining in streaming data where feature distributions change over time.

References

1. Bao, X., Wang, L.: A clique-based approach for co-location pattern mining. Inf. Sci. **490**, 244–264 (2019)
2. Bron, C., Kerbosch, J.: Algorithm 457: finding all cliques of an undirected graph. Commun. ACM **16**(9), 575–577 (1973)
3. Carrillo, R.E., Aysal, T.C., Barner, K.E.: A generalized Cauchy distribution framework for problems requiring robust behavior. EURASIP J. Adv. Sig. Process. **2010**(1), 312989 (2010)
4. Foss, S., Korshunov, D., Zachary, S.: An Introduction to Heavy-Tailed and Subexponential Distributions, vol. 6. Springer, Cham (2011). https://doi.org/10.1007/978-1-4614-7101-1
5. Hou, B., et al.: Efficient maximal clique enumeration over graph data. Data Sci. Eng. **1**(4), 219–230 (2016)
6. Huang, Y., Pei, J., Xiong, H.: Mining co-location patterns with rare events from spatial data sets. GeoInformatica **10**(3), 239–260 (2006)
7. Huang, Y., Shekhar, S., Xiong, H.: Discovering colocation patterns from spatial data sets: a general approach. IEEE Trans. Knowl. Data Eng. **16**(12), 1472–1485 (2004)
8. Li, Y., Shao, Z., Yu, D., Liao, X., Jin, H.: Fast maximal clique enumeration for real-world graphs. In: Li, G., Yang, J., Gama, J., Natwichai, J., Tong, Y. (eds.) DASFAA 2019. LNCS, vol. 11446, pp. 641–658. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-18576-3_38
9. Shekhar, S., Huang, Y.: Discovering spatial co-location patterns: a summary of results. In: Jensen, C.S., Schneider, M., Seeger, B., Tsotras, V.J. (eds.) SSTD 2001. LNCS, vol. 2121, pp. 236–256. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-47724-1_13

10. Tran, V., Wang, L., Chen, H., Xiao, Q.: MCHT: a maximal clique and hash table-based maximal prevalent co-location pattern mining algorithm. Expert Syst. Appl. **175**, 114830 (2021)
11. Tran, V., Wang, L., Zhang, J., Do, T.: Efficient mining of high utility co-location patterns based on a query strategy. In: Yang, X., et al. (eds.) ADMA 2023. LNCS, vol. 14176, pp. 401–416. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-46661-8_27
12. Tran, V., Wang, L., Zhou, L.: A spatial co-location pattern mining framework insensitive to prevalence thresholds based on overlapping cliques. Distrib. Parallel Databases **41**(4), 511–548 (2023)
13. Yang, P., Wang, L., Wang, X., Zhou, L.: Efficient discovery of co-location patterns from massive spatial datasets with or without rare features. Knowl. Inf. Syst. **63**(6), 1365–1395 (2021)
14. Yoo, J.S., Shekhar, S.: A joinless approach for mining spatial colocation patterns. IEEE Trans. Knowl. Data Eng. **18**(10), 1323–1337 (2006)