Clustering Assisted Co-location Pattern Mining for Spatial Data

Naveen Kumar
Department of Computer Science, Pondicherry University, Kalapet - 605014, Puducherry, India.
E-mail: nav.bharti@gmail.com

S. Siva Sathya
Department of Computer Science, Pondicherry University, Kalapet - 605014, Puducherry, India.
E-mail: ssivasathya@gmail.com

Abstract
The importance of spatial data mining is growing with the increasing incidence and importance of large spatial datasets repositories of remote-sensing images, location based mobile app data, satellite imagery, medical data and crime data with location information, three dimensional maps, traffic data and many more. However, as classical data mining techniques are often inadequate for spatial data mining, different techniques for spatial data mining are being exclusively developed. Co-location pattern mining is one of the techniques to discover the set of spatial features frequently located together in the geographic proximity. In this paper we propose a model for finding the frequently occurring co-location patterns of objects in spatial datasets using a co-location mining algorithm which utilizes clustering technique before mining the data for rules. The proposed algorithm namely, SigCPM overcomes the limitations of the existing grid-based approaches. The proposed algorithm is compared with the existing methods of co-location pattern mining and evaluated.

Keywords: Spatial Data Mining; Co-location pattern; Transaction-based Co-location; Grid-based Co-location; Co-location rules; Prevalence Threshold

Introduction
We are living in a digital era where information systems are being flooded with terabytes of information every day. Amongst many other types of data, accumulation of spatial data has increased in tremendous speed in the last decade with the advent of advanced data collection systems like NASA’s EOS (Earth Observatory System), GPS (Global Positioning System) etc., termed as spatial big data [4]. In order to understand, collect and predict useful information from these huge repositories of data, these spatial data needed to be organized. This gave rise to specialized databases to handle such ‘data related to space’ called spatial databases [12] [14] [17] with advanced capabilities like spatial querying and spatial joins. The development of spatial databases and spatial database analysis (SDA) system necessitated automated discovery of spatial knowledge. Spatial data mining (SDM) [18] [14] [12] is the process of discovering interesting and previously unknown, but potentially useful patterns from large spatial datasets.
Spatial data [12] [14] [17] [13] is information about a physical object that can be represented by numerical values in a geographic coordinate system. The physical object can be represented as a point, line or polygon [14]. In case of a point it can be represented as either a co-ordinate like (5, 2) where the first number represents that point’s position on the horizontal (x) axis and the second number represents the point’s position on the vertical (y) axis or it might be represented as a point on a map like Point (46.521076, 21.972656) by its latitude and longitude.
It is more difficult to extract relevant and interesting patterns that are useful from spatial datasets than to extract the corresponding patterns from traditional numeric and categorical data because of the inherent complexity [14] [20] of spatial data types, relationships between them, spatial structure of errors, presence of mixed distribution and very importantly, spatial autocorrelation. Spatial autocorrelation [20] means the measure of the degree to which a set of spatial features and their associated data values tend to be clustered together in space (positive spatial autocorrelation) or dispersed (negative spatial autocorrelation). An example of co-location is the presence of different species of cicadas in a rainforest or symbiotic species of plants and fungi [28] depending on the environmental conditions.
The spatial dataset consists of a collection of spatial features [15] [1] [19] [25]. These features describe the presence of geographic object types at different locations in geographic space. Co-location patterns are the subsets of features usually located together in geographic proximity. Spatial proximity [1] is the important concept to discover co-location patterns from the huge spatial datasets. Co-location pattern mining is the task of discovering co-location patterns in the geographic space. This model infers the presence of spatial features in the neighborhood of instances of other spatial features.
Co-location pattern mining [24] [16] finds application in several real life scenario like crime related analysis, socio-economic studies, weather monitoring, predicting habitat of birds. For instance in urban crime analysis two or more types of crimes may occur together like robbery and assault, violence and theft etc. In the medical scenario chemical industry’s presence and the cancer cases in a geographical or spatial region is found. Emergence or breakthrough of co-location pattern poses some challenges [15] as spatial objects are embedded in a continuous space, whereas classical data is often discrete. A major portion of the computation time [1] is consumed in the identification of the instances of co-location patterns that plays a vital role to the organizations which make decision on spatial data. These organizations are concerned in
discovering frequent co-locations among spatial features that may yield important insights for many applications. Several works have been carried out to resolve co-location pattern mining problem. Spatial co-location pattern discovery and rule mining approaches are grouped into two broad categories: spatial statistics approaches [18] and spatial data mining approaches [15] [1] [18] [19]. Spatial statistics [1] approaches use statistical measures like cross k-function with monte-karlo, mean-nearest neighbor distance and spatial autocorrelation. The spatial data mining approach materializes the data for further processing. Association Rule Mining [7] based approaches for spatial data focus on the creation of the transactions over the geographic space and then apriori algorithm [5] is used for rules generation. Transaction based approach is one of the most important method for co-location pattern mining. In transaction based approach the datasets are converted into transactions and co-location rules mining is used to identify co-location rules. The previous work of co-location pattern mining is based on grid-based [1] [23] co-location pattern mining. The limitations of grid-based approach have been identified that data instances may be divided by grid boundaries, and another limitation is the grid granularity, very large grids or very short grids may result in meaningless co-location patterns. Defining the transactions, over geographic space, use reference-feature centric techniques [7] or data-partitioning techniques [26]. The reference-feature centric technique is based on the choice of reference spatial feature and is relevant to application domains more focusing on a specific spatial feature. The data-partitioning approach deals with the transaction creation by making use of prevalence measure that is based on frequency of co-location pattern by grouping the spatial instances into disjoint partition. Spatial Association Rules (SAR) [7] mining is one of the spatial data mining techniques to achieve the co-location pattern mining. Even though spatial features are similar to the items in Market-basket problem, it is very different from classical association rule mining because of the lack of natural notion of transactions. The spatial datasets are continuous in the geographic space (i.e. Spatially Auto correlated). The decomposition of spatial data using the predicates formation [22] may mislead and alter patterns. So, usage of co-location pattern mining increases the efficiency of finding the interesting patterns from the huge spatial datasets. The reference-feature centric model [15] [25] materializes a set of transaction around instance of reference feature. The transaction is created by giving focus to a user-specified spatial feature. The association rules are generated using the apriori algorithm, which is related to reference feature. The approach is non-trivial where no reference frame is defined. Forming the transactions around the location of instances of features may have duplicates of many candidate co-location. In this paper, we have investigated and designed a clustering-based data-partition approach that attempts to measure the frequency of a co-location pattern by grouping the spatial instances into the disjoint partitions. Each disjoint partition is treated as transaction around location of instances of features. The data-partitioning is achieved using k-means clustering [2] algorithm and then finding the convex-hull [3] for each cluster. The prevalence measure “\text{min}\_\text{conf}” [1] is used for discovering significant co-location rules. The rest of the paper is organized as follows. Related literature is reviewed in Section 2. The problem formation and basic concepts are defined in Section 3. The algorithm is described in Section 4. Section 5 describes the experimental results and analysis. Section 6 concludes the paper.

### Literature Review

The body of literature for co-location rule mining methods and techniques existing till now can be broadly divided into two categories: spatial statistical methods and spatial data mining methods. This section reviews these methods in brief.

#### A. Spatial Statistics Approach

Spatial statistics approach [18] uses some dedicated techniques like cross k-functions with Monte Carlo simulation [8], mean nearest-neighbor distance [9] and spatial regression [27]. These techniques are used to test the co-location between pairs of spatial features. These techniques may be expensive, therefore, pairwise co-location technique [8] is introduced which uses arbitrary partition of space into lattices [6]. Spatial correlation is found using Chi-square or association rule mining [7] [6] algorithm treating each lattices as transaction. Again, arbitrary partition of space into lattices may lose the neighboring instances across border of cells. The disadvantage of this approach is that it is computationally very expensive to compute all possible co-location patterns because the numbers of candidate subsets are exponential given a large collection of spatial features. Another major drawback of the above discussed techniques is that they cannot be easily extended for more than two spatial features.

#### B. Spatial Data Mining Approach

In the spatial data mining approaches, various efficient algorithms are proposed to discover the co-location patterns from spatial datasets. The spatial data mining approaches for co-location pattern mining can be further divided as transaction-based, distance-based and transaction-free approaches.

- A transaction-based approach partitions the data by defining transactions over spatial datasets and once the transactions are formed then conventional apriori-like algorithm [11] is used.

- A distance-based approach in co-location pattern mining is proposed by Morimoto [26]. In Morimoto work, k-neighboring class [26] set is defined as distance-based co-location pattern, and a prevalence measure is taken as the total instance count for each pattern. A non-overlapping-instance constraint is used to get the anti-monotone property for the prevalence measure.

- A transaction-free co-location pattern mining algorithm on an event centric model is proposed in [25] as an extended work of distance-based co-location mining [19]. The distance-based is refined by providing an interest measure called participation index (PI) [25]. This interest measure possesses not only anti-monotone property for efficiently identifying co-location but have relationship to the cross-k function.

1387
Many algorithms are introduced under transaction approach such as reference-centric approaches, spatial join-based approaches and window centric approaches. The important tasks of transaction approaches are conversion of spatial datasets into transactions, identify co-locations and co-location rules. In reference-feature centric model \[7\] \[15\], a user-specified spatial feature is defined and then *apriori*-like algorithm is used for identifying the co-location rules. The identified co-location rules in reference-centric model are all related to one specified spatial feature.

The spatial join-based \[21\] model includes cluster-and-overlay and instance-join methods. In the first method cluster-and-overlay, a map layer is constructed for each feature on the basis of instance clustering boundary for generating transactions. A relation table is constructed using map layer and then association rule mining algorithm is used on the relational table. Second method is instance-based method for co-location pattern mining in which co-locations are identified by a framework similar to classical association rules mining \[15\]. In this framework, the basic concept is similar to the concept of association rule mining which is based on neighborhood relationship and participation index. Transaction based approaches focus on defining transactions over space so that an *Apriori* \[5\] like algorithm can be used for mining rules which look like association rules. Different approaches are followed including ad-hoc windowing \[7\] to create transactions. One of the approaches is to create a reference centric model that creates transactions around a reference feature. The shortcoming of this approach is that it might consider the same instance set more than once. Another approach, the window centric model divides the space into several regions (or cells) and considers instances in each cell as a transaction.

The window-centric approach \[1\], partitions spatial datasets into cells using grid-based transaction and considers spatial objects in each cell as a transaction. Once the transactions are created, the co-location mining algorithm is used to identify significant co-location. The window centric model enumerates all possible windows as transactions in the space discretized by grids or windows. Each transaction contains a subset of spatial features of which at least one instance occurs in the corresponding window. The support and confidence \[7\] \[11\] \[5\] of the traditional association rule problem can be used as prevalence and conditional probability measures. The major drawback of this approach is that some of the instances may be divided by boundary and some significant co-location may be partitioned into different cell that can mislead the prevalent co-location. In some cases, an arbitrary partition is treated as a transaction to derive co-location patterns without considering any patterns across partition boundaries. This is the approach of creating a local model. All the three models reference-feature centric, spatial-join based and window centric transforms the data into transactions in different ways from each other. This Paper proposes the approach of transaction-based co-location pattern discovery.

The above mentioned approaches of co-location pattern mining have limitations of constructing transaction and neighborhood relationship. Specially, the grid-based transaction causes meaningless co-location pattern identification. The proposed approach of transaction creation using KMeans clustering solves the above discussed limitations and identifies significant co-location rules.

### Basic concepts and Problem Formation

Given a set of spatial features \( F = \{ f_1, f_2, \ldots, f_n \} \), a set of spatial instance objects \( S = \{ s_1, s_2, \ldots, s_n \} \) and a neighborhood relationship over set \( S \), a co-location pattern \( p \) is a subset of spatial features, \( p \subseteq F \), whose instances are in a cluster with KNN relationship. The neighborhood relationship is formed using the K-means clustering algorithm on spatial datasets \( S \). Two spatial instances are neighbors of each other in the Euclidean space if they are in the same cluster. A co-location instance \( I \) is a set of objects, \( I \subseteq S \), which includes instance objects of all features in the co-location and form a clique. Note that here in our case the clique is a cluster in the set of clusters that we get from the k-means algorithm.

A co-location rule is of the form of \( X \Rightarrow Y \) with conf \((X \Rightarrow Y)\) as the prevalence measure, where \( X \subseteq F \) and \( Y \subseteq F \) are co-location patterns and \( X \cap Y = \emptyset \). The interest measurement of the co-location rules can be calculated by the prevalent measures: Support (sup) and Confidence (conf). We use co-location prevalence measure defined by \[15\]. First, the participation ratio \( PR(f, t) \) of a feature \( f \in F \) in a transaction \( t \in T \) is given by Equation 1. The participation ratio of a feature in a transaction is the fraction of the feature \( f \) in a transaction \( t \). The participation ratio \( PR(p, t) \) of a co-location pattern \( p \) in a transaction \( t \) is the product of the participation ratio \( PR(f, t) \) of all the features of the co-location pattern \( p \) in transaction \( t \), defined as \( PR(p, t) \) is given in Equation 2, where \( \Pi \) is the product of all the features \( f \in p \) in co-location \( p \). The support \( sup(p) \) of co-location pattern \( p \) is the sum of participation ratio \( PR(p, t) \) of co-location pattern \( p \) in all the transactions \( T \). The support of co-location pattern is defined in Equation 3, where \( \Sigma \) represents summation of participation ratio of co-location patterns in transaction \( t \). A high support of a co-location indicates that the spatial features in that co-location are highly co-located together. The confidence \( \text{conf}(X | Y) \) of a co-location rules \( X \Rightarrow Y \) is the fraction of instances of \( Y \) in the neighborhood of \( X \). The confidence of a co-location rules \( X \Rightarrow Y \) is defined in Equation 4.

\[
PR_f(f, t) = \frac{\text{Total instances of } f \text{ in transaction } t}{\text{Total instances of all feature in } t} \quad (1)
\]

\[
PR_p(p, t) = \prod_{f \in p} PR_f(f, t) \quad (2)
\]

\[
sup(P) = \sum_{p \subseteq F} PR_p(p, t) \quad (3)
\]

\[
\text{conf}(X | Y) = \frac{\text{sup}(X \cup Y)}{\text{sup}(X)} \quad (4)
\]

#### Proposed SIGCPM Algorithm

This section presents the proposed co-location mining algorithm called SigCPM which is meant to generate all the co-location rules with significant prevalence and confidence above or equal to the user defined \( \text{min}_\text{sup} \) and \( \text{min}_\text{conf} \).
The algorithm comprises of three major phases:
1. Transaction generation using clustering over datasets
2. Co-location pattern mining using interest measure calculation
3. Co-location rule generation using co-location patterns

**SigCPM Algorithm:**

**Input:** Set of spatial data S; significant threshold value \( \text{min\_sup}, \text{min\_conf} \);

**Output:** Set of significant co-locations and co-location rules

1. **Step 1:** Apply clustering (Kmeans) on spatial datasets S with clustering threshold.
   Average number of data in each cluster should be at least \(|F|\), Where F is a set of features.
2. **Step 2:** \( T = \text{Set of clusters as Transactions} \)
3. **Step 3:** \( CP = \text{Generate set of candidate Colocation Patterns} \)
4. **Step 4:** Calculate participation ratio of features \( f \in F \) in \( t \in T \)
   
   For each \( t \in T \) Do
   
   For each \( f \in F \) Do
   
   \( \text{PR}(f, t) \)
   
   End for

   End for

5. **Step 5:** Calculate Participation-Ratio and Support of co-locations in \( CP \)

   For each \( p \in CP \) Do
   
   For each \( t \in T \) Do
   
   \( \text{p.PR} = \text{PR}(p, t) \)
   
   End for

   If \( \text{Support}(p) \geq \text{min\_sup} \)
   
   \( \text{p.support} = \text{Support}(p) \)

   End If

   End for

6. **Step 6:** Calculate Co-location rule generation

   For each rule \( (i) \in R \) Do
   
   //R is a set of rules and each rule \( (i) = (X \Rightarrow Y) \), where X and Y are co-locations and \( X \cap Y = \emptyset \) disjoint
   
   If rule \( (i) \) \( \text{conf} \geq \text{min\_conf} \) Do
   
   \( \text{CR} = \text{rule}(i) \)

   End if

   End for

7. **Step 7:** Return CR sets

   End

The steps of the algorithm could be explained as follows:

**Step 1:** Applies KMeans clustering algorithm on the spatial dataset S with the K threshold (e.g. KMeans threshold \( K \geq |S|/|F| \)), where F is a set of features in the spatial dataset set S. The threshold constrained K make sure to get all clusters with maximum number of feature’s instances present in it. Also it is ensured that each cluster should have atleast \(|F|\) number of points. KMeans clustering algorithm attempts to determine K partitions and it is relatively scalable and efficient in processing large datasets. It is well known centroid based technique that takes the parameters K and datasets S, where K is the user defined threshold for the minimum number of objects in a cluster and S is set of spatial dataset.

**Step 2:** Generates a set of transactions using the above clustered spatial dataset using convex-hull to make the neighbor relationship. A transaction is defined as a set of features corresponding to spatial objects instances.

**Step 3:** Generates a set of candidate co-locations of all possible sizes, the candidate co-location generation is achieved by using combinations of features. The computational cost of candidate generation is \( O(|F|^2) \) where F set of features. Here all possible combinations are generated as given below and they are pruned later based on interest measurements. Sample combinations of different colocation pattern sizes are given below:

\[ F = \{f_1, f_2, f_3, f_4, f_5\} \]

\[ \text{Size-2} = \{(f_1, f_2), (f_1, f_3), (f_1, f_4), (f_1, f_5), (f_2, f_3), (f_2, f_4), (f_2, f_5), (f_3, f_4), (f_3, f_5), (f_4, f_5)\} \]

\[ \text{Size-3} = \{(f_1, f_2, f_3), (f_1, f_2, f_4), (f_1, f_2, f_5), (f_1, f_3, f_4), (f_1, f_3, f_5), (f_1, f_4, f_5)\} \]

\[ \text{Size-4} = \{(f_1, f_2, f_3, f_4), (f_1, f_2, f_3, f_5)\} \]

**Step 4:** Calculates the participation ratio \( \text{pr}(f, t) \) of a feature \( f \in F \) in a transaction \( t \in T \). The participation ratio gives probability of a feature in a transaction. The probability shows the contribution of a feature in a transaction.

**Step 5:** Calculates the participation ratio \( \text{pr}(P, t) \) of a co-location pattern \( P \) in a transaction \( t \). The participation ratio is the probability of a co-location pattern in a transaction, the value that shows the occurrence of co-location pattern in transaction. Also calculates the support of a co-location that is the prevalence value of co-location pattern. Here, we apply pruning techniques to minimize computation by excluding unnecessary candidate co-location where participation value is less than the user defined prevalence value.

**Step 6:** Calculates the confidence value of co-location and generates the significant rules that has confidence value greater or equal to the user-defined \( \text{min\_conf} \) value. The above steps are shown in Figure-1.

![Figure 1: Co-location Algorithm Flow](image)

To illustrate the proposed algorithm, consider the figure-2 which shows a spatial dataset with five features represented by different symbols (circle, square, triangle, star, diamond) each corresponding to the following features of a sample dataset: vehicle accident (VA), vehicle lost (VL), missing male (MM), missing female (MF) and unidentified body (UB) from a defined polygon region of 32X34 square kilometer. Figure-3 shows the result of clustering the dataset shown in figure-2. The green boundary shows the clusters and is treated as a transaction. Within each transaction, feature-instances that coexist is determined in the next step. For example, in each cluster, one can see that vehicle accident (VA) and unidentified body (UB) coexists together.
Experimental Results and Analysis
This section presents the experiments carried out to evaluate the proposed algorithm in terms of the following parameters: complexity, number of co-location patterns identified, and scalability. The results are also compared with the existing grid based approach and tabulated.

A. Datasets Generation
The experiments are performed on synthetic dataset. The datasets are generated using Random Point Generation algorithm and then geocoded on the map of defined region using QGIS [10] tool to add spatial nature to the datasets. Two synthetic datasets are generated in 32 x 34 square kilometers bounding region, the first datasets with 5 features and the second datasets with 10 features. The first dataset is resampled into four different samples (628, 1000, 5000, 10,000 points in each sample) all with 5 features as in Figure 1 and Figure 2, another dataset has a total of 10,000 points with 10 features.

B. Results and Analysis
Two experiments on the different datasets mentioned above have been carried out. As no tools are present to analyze the grid-based co-location patterns the algorithm has been implemented in Java and compared with the proposed algorithm.

To implement the grid-based algorithm the granularity of the grid (the unit of the grid cells which affects the number of points per cell) is crucial for the accuracy of the result. The grid-based co-location depends on the size of the cell. Very large size of cell may lead to omission of some regions of space and very small size of cell may leads to a great number of transactions. As we decrease the size of cells by two it leads to increase in the transactions by four, that means increase in computation, this is controlled in [1] by using the average buffer size. Figure-4-5 shows the visualization of the cluster based and grid based approaches to find the co-location pattern.
In the proposed algorithm the number of clusters that should be used to find the co-location patterns is an important parameter which determines the quality of the results. Through experimentation, it is found that the number of clusters should be equal to the total no. of data instances divided by the number of features to give good results. i.e. No. of clusters = (No. of instance i.e. |S| / No. of features i.e. |F|). The Choice of number of clusters is done in this way so that each cluster will have proper participation of instances of each features. The experiments to analyze the algorithm is given below:

i. **Experiment 1**
In Experiment 1 four different tests are performed on the four samples of datasets with 5 features, each samples has an average number of 5 instances per transactions. Thus 100 transactions were generated. For each sample datasets confidence threshold is varied from 0.3 (30%) to 0.7 (70%). This experiment was done to analyze the performance of the proposed algorithm SigCPM (Significant co-location Pattern Mining) algorithm with grid-based co-location algorithm in terms of execution speed. The experiment is done for examining the time complexities of SigCPM and grid-based co-location algorithms. The results of the algorithm are shown in Figure-6 to Figure-9 which demonstrates that the total run time in all the cases for a threshold below 70%, the proposed algorithm SigCPM performs significantly faster than Grid-based algorithm. The experiment also shows that as the confidence threshold (min_conf) is increased the run time decreases. The pruning of lower threshold decreases the run time of phase 3.
ii. Experiment 2

The experiment 2 has been done to prove the performance of the algorithm in terms of the number of co-location patterns identified. It is performed on a dataset with 10,000 instances consisting of 10 features. The number of discovered co-locations in Figure-10 increases as the co-location size increases with co-location size less than or equal to 50% of the total features in the dataset. Both the algorithms are executed on user defined min_conf value 0.5 (50%). The total number of co-locations discovered by the proposed algorithm is more when compared to the Grid-based algorithm irrespective of co-location size. The x-axis shows the size of co-location which implies the number of features that coexist in a particular geographic size and y-axis shows how many such co-location exists.

Figure 9: 10,000 Data with 5 Features

Figure 10: Co-location Discovered on 50% Confidence Threshold

The proposed algorithm SigCPM was able to find 34 significant co-locations and 243 co-location rules with the 628 input datasets on prevalence threshold 50. The Grid-based co-location algorithm discovered 26 significant co-locations and 195 co-location rules with the same number of data instances and prevalence threshold. Among them 172 rules are matching in both the tests and 71 rules are significant and unique in the former execution. From the experiments carried out, it was found that the time complexity of the algorithm is significantly less compared to the grid based algorithm as seen in the figures. Also, the scalability of the algorithm was tested with different data sizes and proved to yield good results as shown in figure-6-9. With respect to the number of colocation patterns identified, it has outperformed the grid based approach and is shown in figure-10.

When sensor nodes distance becomes larger, the efficiency of charging increases. Because when sensor nodes are more distance they can better capture the energy in the air without interfering with each other.

We can see while NDN Based DCP takes less time to reach the sensor node when compared to the hierarchical network because in layers network to reach the information of battery low of nodes its take long time to reach the proxy nodes. From proxy nodes the information is passed to the sensor. Where in proposed this time to reach information to sensor is very much reduced.

In proposed system the delay to reach the energy information of sensor is lesser when compared to the hierarchical order of the nodes in the wireless sensor network, because in cluster there is no more no of intermediate nodes to receive and send the neighbor nodes energy message.

Conclusion

This paper has identified the limitations of the conventional co-location pattern mining algorithms which works on grid-based approaches and proposed a new approach to the co-location pattern mining problem. Due to the lack of real spatial data pertaining to collocated features, synthetic data have been generated and Geocoded to prove the effectiveness of the algorithm. The uses of clustering techniques to convert the spatial datasets into transactions minimizes the neighborhood complexity. In order to minimize the computation, the pruning techniques are presented. The algorithm was compared with the popular grid-based co-location mining algorithm and the experiment showed that our approach finds significant co-location rules. Scientists are interested in understanding the evolution of co-location patterns in spatial network activities. In future work, we plan to explore the methods of mining co-locations for the spatial network event activity like how different spatial network activities are co-located to each other along spatial network. Also the algorithm is to be tested on some real spatial datasets.

References


