A Framework for Discovering Co-location Patterns in Data Sets with Extended Spatial Objects

Hui Xiong *† Shashi Shekhar† Yan Huang‡ Vipin Kumar†
Xiaobin Ma† Jin Soung Yoo†

Abstract

Co-location patterns are subsets of spatial features (e.g. freeways, frontage roads) usually located together in geographic space. Recent literature has provided a transaction-free approach to discover co-location patterns over spatial point data sets to avoid potential loss of proximity relationship information in partitioning continuous geographic space into transactions. This paper provides a more general transaction-free approach to mining data sets with extended spatial objects, e.g. line-strings and polygons. Key challenges include modeling of neighborhood and relationships among extended spatial objects as well as control of related geometric computation costs. The approach we propose is based on a new buffer-based definition of neighborhoods. Furthermore, we introduce and compare two pruning approaches, namely a prevalence-based pruning approach and a geometric filter-and-refine approach. Experimental evaluation with a real data set (a digital roadmap of the Minneapolis and St. Paul metropolitan area) shows that the geometric filter-and-refine approach can speed up the prevalence-based pruning approach by a factor of 30 to 40. Finally, we show how the extended co-location mining algorithm proposed in this paper has been used to find line-string co-location patterns, which can help with decision-makings on selecting most challenging field test routes. These field test routes are important for evaluating a GPS-based approach to accessing road user charges.

Keywords

Spatial Data Mining, Co-location Patterns, GIS Buffer Operation, Spatial Association Rules

1 Introduction

Co-location patterns represent subsets of Boolean spatial features whose instances are often located in close geographic proximity. For example, E-services are growing along with mobile computing infrastructures such as PDAs and cellular phones. Finding E-services frequently located together is of interest to providing location-awareness market promotions. In ecology, scientists are interested in finding frequent co-occurrences among Boolean spatial features, e.g., drought, El Nino, substantial increase in vegetation, substantial drop in vegetation, extremely high precipitation, etc. Effective tools for extracting information from geo-spatial data, the focus of this work, are crucial to organizations which make decisions based on large spatial datasets. These organizations are spread across many domains including ecology and environmental management, public safety, transportation, public health, business, and tourism [3, 14, 16, 10, 23, 27].

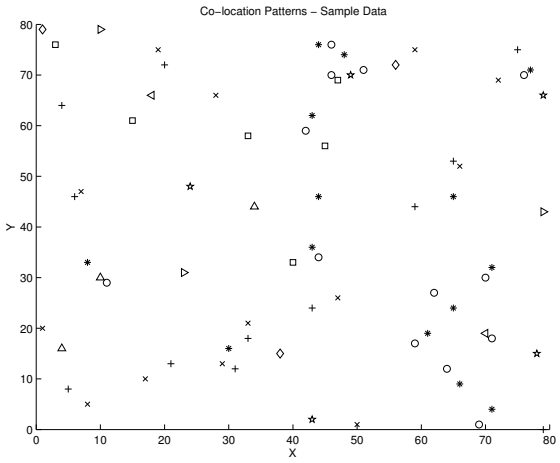

Figure 1: Point Spatial Co-location Patterns Illustration. Shapes represent different spatial feature types. Spatial features in sets {'+', '×'} and {'o', '*'} tend to be located together.

In the real world, many spatial datasets consist of a collection of instances of Boolean spatial features (e.g., drought, needle leaf vegetation). Figure 1 shows the frequent co-occurrences of some point spatial feature types represented by different shapes. As can be

*Contact Author
†Department of Computer Science and Engineering
University of Minnesota - Twin Cities
{huix, shekhar, kumar, xiaobin, jyoo}@cs.umn.edu
‡Department of Computer Science
University of North Texas, huangyan@cs.unt.edu

Figure 2: Line String Co-location Patterns Illustration

seen, instances of spatial features in sets {'+', '×'} and {'o', '*'} tend to be located together. Figure 2 shows an instance of co-location patterns among extended spatial features, namely road-types, on an urban roadmap. Highways in large metropolitan area often have frontage roads nearby. Identification of such co-locations is useful in selecting test-sites for evaluating in-vehicle navigation technology [28]. While Boolean spatial features can be thought of as item types, there may not be an explicit finite set of transactions due to the continuity of the underlying space. As a result, it is difficult to apply classic association rule mining [1, 2, 12, 18, 21, 22, 25, 26] directly to spatial contexts.

Related Work: Approaches to discovering co-location rules in the literature can be categorized into two classes, namely spatial statistics and data mining approaches. Spatial statistics-based approaches use measures of spatial correlation to characterize the relationship between different types of spatial features. Measures of spatial correlation include the cross-K function with Monte Carlo simulation [5], mean nearest-neighbor distance, and spatial regression models [4]. Computing spatial correlation measures for all possible co-location patterns can be computationally expensive due to the exponential number of candidate subsets given a large collection of spatial Boolean features.

Data mining approaches can be further divided into a clustering-based map overlay approach and association rule-based approaches. A clustering-based map overlay approach [9, 8] treats every spatial attribute as a map layer and considers spatial clusters (regions) of point-data in each layer as candidates for mining associations. Given X and Y as sets of layers, a clustered spatial association rule is defined as $X \Rightarrow Y(CS, CC\%)$, for $X \bigcap Y = \emptyset$, where CS is the clustered support, defined as the ratio of the area of the cluster (region) that satisfies both X and Y to the total area of the study region S, and $CC\%$ is the clustered confidence, which can be interpreted as $CC\%$ of areas of clusters (regions) of X intersect with areas of clusters(regions) of Y. There are several disadvantages in clustering-based approaches. First, these approaches assume that instances of each spatial feature are clustered. However, many spatial features can be completely spatially random or declustered (i.e. negative spatial autoregression) at many scales. Second, these approaches are sensitive to the choices of clustering algorithms from among a large number of candidates [11]. Finally, these approaches are quite sensitive to the presence of noise.

Association rule-based approaches fall into two categories: The first category focuses on the creation of transactions over space so that an *Apriori*-like algorithm [2] can be used. Transactions over space can be defined by a reference-feature centric model [15] or a data-partition [17] approach.

The **reference feature centric model** [15] is relevant to application domains focusing on a specific Boolean spatial feature, e.g. cancer. Domain scientists are interested in finding the co-locations of other task relevant features (e.g. asbestos, other substances) to the reference feature. This model enumerates proximity neighborhoods to "materialize" a set of transactions around instances of the reference spatial feature. A specific example is provided by the spatial association rule [15]. Transactions are created around instances of one user-specified spatial feature. The association rules are derived using the *Apriori* [2] algorithm. The rules found are all related to the reference feature. Generalizing this paradigm to the case where no reference feature is specified is non-trivial. Defining transactions around locations of instances of all features may yield duplicate counts for many candidate associations.

Defining transactions by a **data-partition approach** [17] attempts to measure the frequency of a co-location pattern by grouping the spatial instances into disjoint partitions. This approach may be useful in data exploration when one is interested in exploring the sets of partitions and identifying regions that maximize co-location. Occasionally, imposing artificial disjoint transactions via space partitioning may undercount instances of tuples intersecting the boundaries of artificial transactions or double-count instances of tuples co-located together. In addition, there may be multiple partitions yielding distinct sets of transactions, which in turn yields different values of prevalence for co-

location patterns.

The second category of association-rule based approaches are transaction-free. In other words, no explicit transactions are generated for the purpose of mining co-location patterns. The **event centric model** [13, 19] falls into this category and is relevant to applications like ecology, where many types of Boolean spatial features exist. Ecologists are interested in finding subsets of features likely to occur in a neighborhood around instances of given subsets of event types. The event centric model yields a definition of one prevalence measure without the need for generating transactions. However, the event centric model is only for spatial point objects; there is no natural extension of this model to extended spatial objects (e.g. polygons and line strings).

In this paper, we generalize the concept of co-location patterns to extended spatial data objects and provide a more general transaction-free co-location mining model by using the notion of buffer, a zone of specified distance around spatial objects. This buffer-based model integrates the best features of the event centric model and can identify co-location patterns over extended spatial objects. Furthermore, this paper presents two pruning approaches, namely a prevalence-based pruning approach and a geometric filter-and-refine approach. The geometric filter can reduce a large number of expensive geometric intersection operations, thus saving a lot of computation costs. As demonstrated by our experiments on a real data set (the roadmap for Minneapolis and St. Paul metropolitan area), the geometric filter-and-refine approach can speed up the prevalence-based pruning approach by a factor of 30 to 40. Finally, we introduce an application of the proposed extended co-location mining algorithm for discovering line-string co-location patterns which can help with decision making regarding the selection of most challenging field test routes. These field test routes can then be used to evaluate the performance of a GPS-based approach to accessing road user charges [20].

Outline: The remainder of this paper is organized as follows. Section 2 describes the buffer-based model and its associated measures of prevalence and conditional probability. Section 3 presents a coarse-level co-location mining framework and the geometric challenge. Co-location mining algorithms and design decisions are described in section 4. We provide the experimental results in section 5. Finally, section 6 gives conclusions and suggests future work.

2 A Buffer-based Model for Co-location Pattern Discovery

In this section, we propose a buffer-based model for mining co-location patterns. This model can deal with point objects as well as extended spatial objects, such as line strings and polygons.

2.1 Basic Concepts of the Buffer-based Model

To facilitate our discussion, we first present some basic concepts of the buffer-based model.

DEFINITION 2.1. *A* **co-location pattern** *is a set of spatial features with the prevalence measure of this set greater than a user-specified minimum prevalence threshold. A* **co-location rule** *is of the form:* $C_1 \rightarrow C_2(s, cp)$ *where* C_1 *and* C_2 *are co-locations,* s *is a number representing the prevalence measure and* cp *is a number measuring the interestingness of the rule.*

A prevalence measure describes statistical significance of a co-location pattern while interestingness measures how useful or actionable a co-location pattern is.

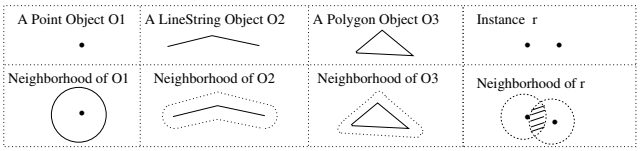

Figure 3: A Buffer-based Neighborhood Illustration.

DEFINITION 2.2. $N(p)$, *the size-d Euclidean neighborhood of a point location p, is a circle of radius d with p as its center.*

DEFINITION 2.3. $N(o)$, *the size-d neighborhood of an extended spatial object (e.g. polygon, line-string), is defined by the buffer operation as shown in Figure 3.*

In GIS or geographic information systems, a buffer is a zone of specified distance around spatial objects. The boundary of the buffer is the isoline of equal distance to the edge of the objects. Figure 3 shows a buffer operation on spatial objects. As can be seen, the buffer operation results in new boundaries around points, lines, or polygons. Although the buffer operations are computationally expensive, there are many advantages to using buffers in GIS. First of all, objects in space frequently have some sort of impact on the objects and areas around them. For example, Freeways create "noise pollution" that can be heard blocks away. Also, factories emit fumes that can affect people for miles around. Buffers can be used in these instances to depict a sphere of influence in which the people and places within this "sphere" are more significantly impacted by a given phenomenon than those on the outside. Another advantage of using buffers

in GIS is to protect places that are less significantly impacted by a given phenomenon. Examples include areas around school where liquor stores are prohibited.

A common concern among people using buffers is to figure out what buffer size to use in their analysis. Since buffer size designation can vary substantially between projects, various factors could be considered. These factors may include: 1) Input from source: An attribute of the object being buffered is used to decide the buffer size. 2) Internal factors within a buffer: Variables affecting the area inside the buffer boundaries such as topography within the buffer. 3) Outcomes: What is probably going to happen because of variables inside and outside the buffered area.

DEFINITION 2.4. *The Euclidean neighborhood $N(f_j)$ of a feature f_j is the union of $N(i_l)$ for every instance i_l of the feature f_j.*

DEFINITION 2.5. *The Euclidean neighborhood $N(f_1 f_2 \dots f_k)$ for a feature set $C = \{f_1, \dots, f_k\}$ is the intersection of $N(f_i)$ for every feature f_i in C.*

DEFINITION 2.6. *$I = \{i_1, i_2, \dots, i_k, B\}$ is a* **row instance** *of a feature set $C = \{f_1, \dots, f_k\}$ if the feature set of I contains C and no proper subset of I does so; and $B > 0$ where B represents $\bigcap_{i_j \in I} N(i_j)$. The* **table instance** *of a feature set $C = \{f_1, \dots, f_k\}$ is the collection of all row instance of the set C.*

DEFINITION 2.7. *The* **coverage ratio** *$Pr(f_1 f_2 \dots f_k)$ for a feature set $C = \{f_1, \dots, f_k\}$ is $\frac{N(f_1 f_2 \dots f_k)}{The\ total\ area\ of\ the\ plane}$, where $N(f_1 f_2 \dots f_k)$ is the Euclidean neighborhood of the set C.*

The coverage ratio serves as the prevalence measure in our buffer-based model. In other words, for a spatial feature set F, if the coverage ratio $Pr(F)$ is greater than a user-specified minimum prevalence threshold, the feature set F is a co-location pattern. Intuitively, the coverage ratio measures fraction of the total area of the spatial framework influenced or covered by the instances of given spatial features.

DEFINITION 2.8. *The* **conditional probability** *$Pr(C_2|C_1)$ of a co-location rule $C_1 \rightarrow C_2$ is the probability of finding the neighborhood of C_2 in the neighborhood of C_1. It can be computed as $\frac{N(C_1 \bigcup C_2)}{N(C_1)}$ using the neighborhoods of co-locations C_1 and $C_1 \bigcup C_2$.*

LEMMA 2.1. *The coverage ratio for co-location patterns is monotonically non-increasing with the size of the co-location pattern increasing.*

Proof. According to Definition 2.7, the **coverage ratio** $Pr(f_1 f_2 \dots f_k)$ for a co-location $C = \{f_1, \dots, f_k\}$ is $\frac{N(f_1 f_2 \dots f_k)}{The\ total\ area\ of\ the\ plane}$, where $N(f_1 f_2 \dots f_k)$ is the Euclidean neighborhood of the co-location C. For any co-location $C' = C \cup \{f'\}$, where $f' \notin C$, we need to prove that $Pr(f_1 f_2 \dots f_k) \geq Pr(f_1 f_2 \dots f_k f')$. Also, consider that $Pr(f_1 f_2 \dots f_k f') = \frac{N(f_1 f_2 \dots f_k f')}{The\ total\ area\ of\ the\ plane}$, we only need to prove $N(f_1 f_2 \dots f_k) \geq N(f_1 f_2 \dots f_k f')$. Since the Euclidean neighborhood N(C) for a co-location C is the intersection of $N(f_i)$ ($\forall\ f_i \in C$) and adding one more feature can only reduce the intersection area, we obtain $N(f_1 f_2 \dots f_k) \geq N(f_1 f_2 \dots f_k f')$.

Lemma 2.1 ensures that the coverage ratio can be used to efficiently discover co-location patterns with high prevalence. The coverage ratio pruning in co-location pattern mining is similar to the support-based pruning in association-rule mining [1].

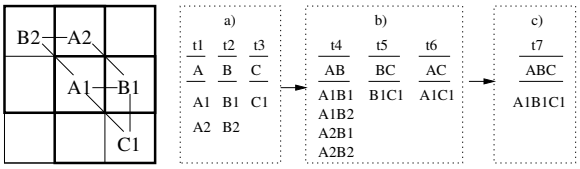

Figure 4: An Illustration to show the inconsistency of the definition of the conditional probability measure in the event centric model with the multiplication rule. (a) Table instances of co-locations $\{A\}$, $\{B\}$, and $\{C\}$. (b) Table instances of co-locations $\{A, B\}$, $\{B, C\}$, and $\{A, C\}$. (c) Table instance of co-location $\{A, B, C\}$.

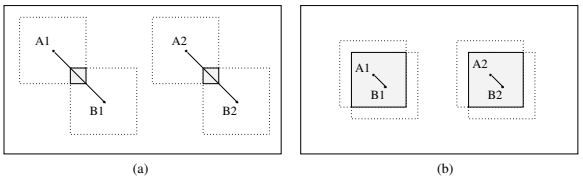

Figure 5: An Illustration Example to show that the event centric model is not good at incorporating spatial context.

2.2 Advantages of the Buffer-based Model

The buffer-based model has three advantages over the event centric model [19] as follows.

- First, the event centric model is only for point objects, while the buffer-based model can deal with point objects as well as extended spatial objects.
- Second, the conditional probability measure used in the event centric model does not satisfy the multiplication rule [7] in statistics.

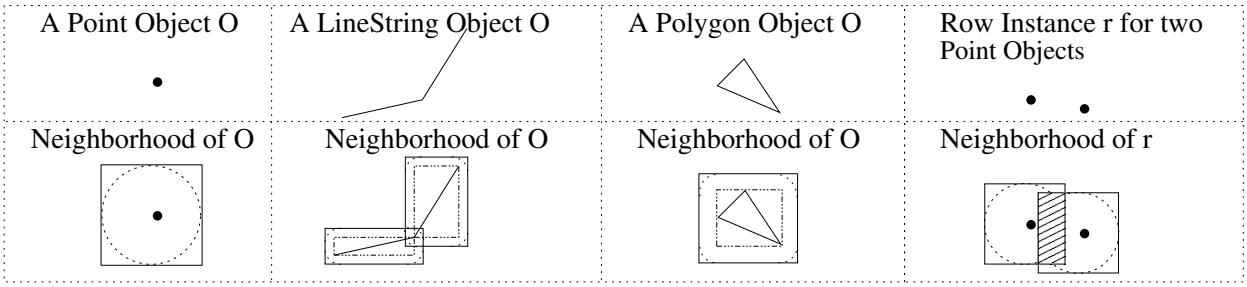

Figure 6: Neighborhood Illustration for Various Spatial Objects

To show this, we first recall the definition of the conditional probability in the event centric model. A set of spatial instances I is a row instance of a subset of spatial features C, if any pair of elements from I are neighbors and the spatial feature set formed by spatial features of elements of I contains C and no proper subset of I does so. The conditional probability of a co-location rule $C_1 \rightarrow C_2$ is $\frac{|distinct(\pi_{C_1}(row\ instances\ of\ C_1 \cup C_2))|}{|row\ instances\ of\ C_1|}$ where π is a relational projection operation. For the illustration spatial dataset shown in Figure 4, the table t4 in Figure 4 (b) contains four row instances: $A_1B_1, A_1B_2, A_2B_1\ and\ A_2B_2$ of the co-location $\{A, B\}$ and the table t7 in Figure 4 (c) contains one row instance $A_1B_1C_1$ of the co-location $\{A, B, C\}$. The conditional probability $Pr(C|AB)$ of the co-location rule $AB \rightarrow C$ is $\frac{|distinct(\pi_{\{A,B\}}(row\ instances\ of\ \{A,B,C\}))|}{|row\ instances\ of\ \{A,B\}|} = \frac{1}{4}$. Also, we get $Pr(BC|A) = 1/2$ and $Pr(B|A) = 1$ (Please note that, after projecting on feature A, there are only two different instances of A although there are four row instance of the co-location $\{A, B\}$. That is the reason why $Pr(B|A) = 1$). The above results in $Pr(BC|A) \neq Pr(C|AB)Pr(B|A)$. However, by the multiplication rule for the conditional probability, we know $Pr(C|AB)Pr(B|A) = \frac{Pr(ABC)}{Pr(AB)} . \frac{Pr(AB)}{Pr(A)} = Pr(BC|A)$;

While the definition of the conditional probability measure proposed in the event centric model does not satisfy with the multiplication rule in statistics, our new conditional probability definition does as shown below in Theorem 2.1.

- Third, the event centric model is not good at incorporating spatial context. To illustrate this, let us look at the example dataset shown in Figure 5. Assume that the size of square neighborhood is fixed, under the event centric model, we will identify the same co-location pattern $\{A, B\}$ from two different illustration datasets (a) and (b) with the same significance. However, as we can see, the distance between instances of A and B in dataset (b) is closer than the distance between instances of A and B in dataset (a). According to Tobler's first law of geography: everything is related to everything else but nearby things are more related than distant things [24], we can infer that the co-location pattern $\{A, B\}$ in dataset (b) should be more significant. In spatial statistics, an area within statistics devoted to the analysis of spatial data, this called spatial autocorrelation [5]. Knowledge discovery techniques which ignore spatial autocorrelation typically perform poorly in the presence of spatial data.

THEOREM 2.1. *Suppose that $f_1, f_2, \ldots, f_n$ are n spatial events and $Pr(f_1 f_2 \ldots f_n)$ is the coverage ratio of the co-location $C = \{f_1, f_2, \ldots, f_n\}$. Then*

$$Pr(f_1 f_2 \ldots f_n) = Pr(f_1)Pr(f_2|f_1) \ldots Pr(f_n|f_1 f_2 \ldots f_{n-1}). \quad (2.1)$$

where $Pr(f_n|f_1 f_2 \ldots f_{n-1})$ is the conditional probability of the co-location rule $\{f_1, f_2, \ldots, f_{n-1}\} \rightarrow \{f_n\}$.

Proof. Since $Pr(f_1) = \frac{N(f_1)}{The\ total\ area\ of\ the\ spatial\ framework}$ and we know $Pr(f_2|f_1) = \frac{N(f_1 f_2)}{N(f_1)}$, the product of probabilities on the right side of Equation (2.1) is equal to

$$\frac{N(f_1)}{The\ total\ area\ of\ the\ spatial\ framework} \frac{N(f_1 f_2)}{N(f_1)} \cdots \frac{N(f_1 f_2 \ldots f_n)}{N(f_1 f_2 \ldots f_{n-1})}.$$

Because $Pr(f_1 f_2 \ldots f_{n-1}) > 0$, each of the denominator in the above product must be positive. All of the terms in the product cancel each other except the final numerator $N(f_1 f_2 \ldots f_n)$ and the first denominator $The\ total\ area\ of\ the\ plane$, which is $\frac{N(f_1 f_2 \ldots f_n)}{The\ total\ area\ of\ the\ plane}$. Also, the left side of Equation (2.1) is equal to $\frac{N(f_1 f_2 \ldots f_k)}{The\ total\ area\ of\ the\ plane}$, which is the right side of Equation (2.1).

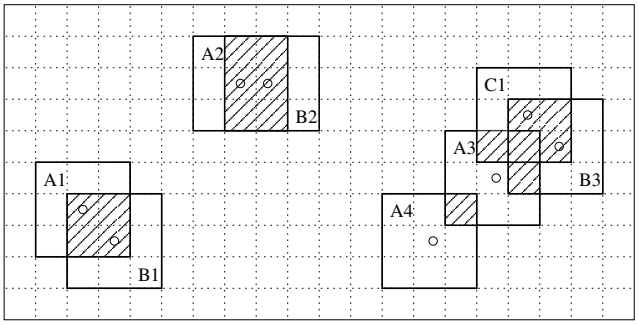

Figure 7: A sample spatial dataset to illustrate the process of mining coarse-level co-location patterns

3 A Coarse-level Co-location Pattern Mining Framework

The buffer-based model has a major challenge in dealing with a large number of overlay operations, which find intersection areas among buffers of spatial objects through geometric intersections. Overlay operations on objects with irregular shapes are very expensive. To cope with this computation challenge, in this section, we present a coarse-level co-location pattern mining framework. This approach follows a filter-and-refine paradigm and is motivated by the observation that spatial objects have unique spatial characteristics, such as distance differences or density differences. In other words, we apply a geometric filter to eliminate a lot of feature sets which cannot form co-location patterns, thus greatly reducing the number of overlay operations and improving performance significantly.

3.1 Basic Concepts

DEFINITION 3.1. *$BN(o)$, the bounding neighborhood of a spatial object (e.g. point, polygon, line-string) o, is defined as MBBR(Buffer(MOBR(Spatial Object O), d)) as shown in Figure 6, where MOBR is the minimum object bounding box, Buffer is the buffer operation with a buffer size as d, and MBBR is the minimum buffer bounding box.*

For instance, for a line-string object O, we first get the minimum bounding box of the object O, MOBR(O). Then we construct a buffer for MOBR(O). Finally, the bounding neighborhood of the object O is the minimum bounding box for this buffer. This process is shown in the second column of Figure 6.

DEFINITION 3.2. *The Euclidean bounding neighborhood $BN(f_j)$ of a spatial feature f_j is the union of $BN(i_l)$ for every instance i_l of the spatial feature f_j.*

DEFINITION 3.3. *The Euclidean bounding neighborhood $BN(f_1 f_2 \dots f_k)$ for a feature set $CC = \{f_1, \dots, f_k\}$ is the intersection of $BN(f_i)$ for every feature f_i in CC.*

For example, Figure 7 shows eight objects with their bounding neighborhoods. In the figure, we can see four instances of feature A, A1, A2, A3, A4, and only the bounding neighborhood of A3 has one-cell overlapping with the bounding neighborhood of A4. If we set the area of a cell to be one unit, the Euclidean bounding neighborhood BN(A) of feature A is 4 x 9 - 1 = 35, which is the union of the bounding neighborhoods of these four instances. In the above calculation, the minus one is due to the fact that we do not want to double count the overlapping area. In addition, the bounding neighborhood of feature set {A, B}, BN(AB), is 4 + 6 + 2 = 12, which is the intersection area of the bounding neighborhood of feature A and feature B.

DEFINITION 3.4. *$CI = \{i_1, i_2, \dots, i_k, BB\}$ is a* **coarse-level row instance** *of a feature set $CC = \{f_1, \dots, f_k\}$ if the feature set of CI contains CC and no proper subset of CI does so; and $BB > 0$ where BB represents $\bigcap_{i_j \in I} BN(i_j)$. The* **coarse-level table instance** *of a feature set $CC = \{f_1, \dots, f_k\}$ is the collection of all coarse-level row instance of the set CC.*

In Figure 7, $CI = \{A1, B1, 4\}$ is a coarse-level row instance of the feature set $CC = \{A, B\}$ since the intersection area of the bounding neighborhoods of instances A1 and B1 is 4. In addition, the table instance of the coarse-level co-location pattern $CC = \{A, B\}$ is $\{\{A1, B1, 4\}, \{A2, B2, 6\}, \{A3, B3, 2\}\}$.

DEFINITION 3.5. *The* **coarse-level coverage ratio** *$CPr(f_1 f_2 \dots f_k)$ for a set $CC = \{f_1, \dots, f_k\}$ is $\frac{BN(f_1 f_2 \dots f_k)}{The\ total\ area\ of\ the\ plane}$, where $BN(f_1 f_2 \dots f_k)$ is the Euclidean bounding neighborhood of the set CC.*

The coarse-level coverage ratio serves as the prevalence measure in our coarse-level co-location mining framework. For the spatial dataset shown in Figure 7, the coarse-level coverage ratio $CPr(A)$ for feature A is $\frac{BN(A)}{The\ total\ area\ of\ the\ plane} = \frac{35}{200} = 0.175$. Furthermore, the coarse-level coverage ratio $CPr(AB)$ for the set $CC = \{A, B\}$ is $\frac{BN(AB)}{The\ total\ area\ of\ the\ plane} = \frac{12}{200} = 0.06$.

DEFINITION 3.6. *A* **coarse-level co-location pattern** *is a set of spatial features with a coarse-level coverage ratio greater than a user-specified minimum prevalence threshold.*

LEMMA 3.1. *The coarse-level coverage ratio for coarse-level co-location patterns is monotonically non-increasing when the size of the coarse-level co-location pattern is increasing.*

Since the proof of this lemma is similar to the proof of lemma 2.1, we omitted it here.

LEMMA 3.2. *For any spatial feature set* $F = \{f_1, f_2, \ldots, f_k\}$, *the coarse-level coverage ratio CPr(F) is greater than the coverage ratio Pr(F).*

Proof. According to definition 2.7, the **coverage ratio** $Pr(F)$ for a feature set $F = \{f_1, \ldots, f_k\}$ is $\frac{N(f_1 f_2 \ldots f_k)}{The\ total\ area\ of\ the\ plane}$, where $N(f_1 f_2 \ldots f_k)$ is the Euclidean neighborhood of the feature set F. Also, by definition 3.5, the **coarse-level coverage ratio** $CPr(F)$ is $\frac{BN(f_1 f_2 \ldots f_k)}{The\ total\ area\ of\ the\ plane}$, where $BN(f_1 f_2 \ldots f_k)$ is the Euclidean bounding neighborhood of the feature set F. Since $BN(f_1 f_2 \ldots f_k)$ is greater than $N(f_1 f_2 \ldots f_k)$ due to the way that the bounding neighborhood is constructed, we know $CPr(F) > Pr(F)$. Hence, this lemma holds.

Lemma 3.2 allows us to design a filter-and-refine approach to finding co-location patterns, since, for a user specified minimum coverage ratio threshold θ, we can first use the coarse-level co-location mining framework as a filter to find coarse-level co-location patterns. All co-location patterns should be within the set of coarse-level co-location patterns by Lemma 3.2. Then, we use overlay operations to find co-location patterns from the set of coarse-level co-location patterns.

3.2 Geometric Challenges and Solutions

In this subsection, we present geometric challenges arising in the coarse-level co-location mining framework and provide the corresponding solutions.

For spatial data sets, it is common that the bounding neighborhoods of instances can overlap with each other. In order to correctly compute the bounding neighborhoods for features or feature sets, we need to build a mechanism to prevent the overlapping area from double counting. Otherwise, we may overestimate the coarse-level coverage ratio of features or feature sets. For this purpose, an innovative and effective geometric mechanism is provided as follows.

LEMMA 3.3. *For any n spatial events* $A_1, \ldots, A_n$,

$$\bigcup_{i=1}^{n} BN(A_i) = \sum_{i=1}^{n} BN(A_i) - \sum_{i<j} BN(A_i A_j) + \sum_{i<j<k} BN(A_i A_j A_k) - \sum_{i<j<k<l} BN(A_i A_j A_k A_l) + \ldots + (-1)^{n+1} BN(A_1 A_2 \ldots A_n). \tag{3.2}$$

Proof. In probability theory, the probability of the union $\bigcup_{i=1}^{n} A_i$ of n events $A_1, A_2, \ldots, A_n$ can be computed as the following:

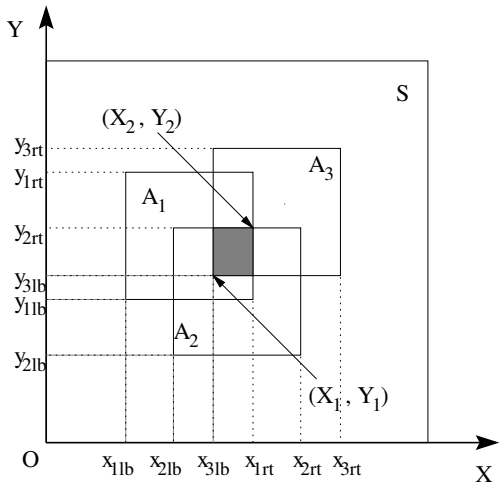

Figure 9: An overlapping example

$$Pr(\bigcup_{i=1}^{n} A_i) = \sum_{i=1}^{n} Pr(A_i) - \sum_{i<j} Pr(A_i A_j) + \sum_{i<j<k} Pr(A_i A_j A_k) - \sum_{i<j<k<l} Pr(A_i A_j A_k A_l) + \ldots + (-1)^{n+1} Pr(A_1 A_2 \ldots A_n). \tag{3.3}$$

where $Pr(A)$ indicates the probability that A will occur. One detailed proof of this equation can be found in [7]. Instead, in our coarse-level co-location mining framework, $CPr(A)$ is defined as the coarse-level coverage ratio of the spatial event A. This definition is similar to the conventional probability definition. As a result, the coarse-level coverage ratio of the union of a finite number of spatial events can be computed in the same way. Since $CPr(A_i) = \frac{BN(A_i)}{the\ total\ area\ of\ the\ plane}$, $CPr(A_i A_j) = \frac{BN(A_i A_j)}{the\ total\ area\ of\ the\ plane}$, and $CPr(A_1 A_2 \ldots A_n) = \frac{BN(A_1 A_2 \ldots A_n)}{the\ total\ area\ of\ the\ plane}$, the right side of the above equation is equal to

$$\sum_{i=1}^{n} \frac{BN(A_i)}{the\ total\ area\ of\ the\ plane} - \sum_{i<j} \frac{BN(A_i A_j)}{the\ total\ area\ of\ the\ plane} + \ldots + (-1)^{n+1} \frac{BN(A_1 A_2 \ldots A_n)}{the\ total\ area\ of\ the\ plane}.$$

Also, the left side of the equation is equal to

$$\frac{\bigcup_{i=1}^{n} BN(A_i)}{the\ total\ area\ of\ the\ plane}.$$

The same denominator, *the total area of the plane*, can be cancelled from both sides, so giving us Equation 3.2.

THEOREM 3.1. *Given any n spatial events* $A_1, A_2, \ldots, A_n$ *and the corresponding bounding neighborhoods* $((x_{1lb}, y_{1lb}), (x_{1rt}, y_{1rt}))$,

EXCOM ALGORITHM

Input: (a) A $D_1 \times D_2$ Spatial Framework $\mathcal{R}$
(b) FT = {A Set of Spatial Features, which can be represented as points, line strings, and polygons.}
(c) I = {Instance-ID, Feature-Type, Location in Space} representing a set of instances of features
(d) A buffer size d.
(e) A minimum coverage ratio threshold θ
(f) A conditional probability threshold α for generating co-location rules.

Output: (1) A set of co-location patterns with coverage ratios greater than a user-specified minimum threshold θ.
(2) A set of co-location rules with a conditional probability greater than α

Variables: k: the co-location size
CC_2: a set of candidate size-2 coarse level co-location patterns.
CP_2: a set of size-2 coarse-level co-location patterns having coverage ratios $> \theta$.
C_k: a set of candidate size-k co-location patterns.
P_k: a set of size-k co-location patterns.
R_k: a set of co-location rules derived from size-k co-location patterns

The Geometric Filter

1. Initialization;
2. CC_2= geometric_search(FT, I, d);
3. CP_2= prevalence_prune(CC_2, θ);

The Refinement and Combinatorial Search

4. Initialization;
5. P_2 = overlay(CP_2, d); k=2;
6. while(not empty P_k) do {
7. C_{k+1} = generate_candidate_colocation(P_k);
8. P_{k+1} = prevalence_prune(C_{k+1}, θ);
9. R_{k+1} = generate_colocation_rule(α);
10. $k = k + 1$;
11. }
12. SAVE: union($P_2, \ldots, P_{k+1}$);
13. SAVE: union($R_2, \ldots, R_{k+1}$);

Figure 8: Overview of the EXCOM Algorithm

$((x_{2lb}, y_{2lb}), (x_{2rt}, y_{2rt}))$, ..., $((x_{nlb}, y_{nlb}), (x_{nrt}, y_{nrt}))$, *where the bounding neighborhood of event* A_i, $1 \leq i \leq n$, *is represented by the left bottom point* (x_{ilb}, y_{ilb}) *and the right top point* (x_{irt}, y_{irt}), *if the bounding neighborhoods of these n spatial events have a common intersection area, then this intersection area can be computed by Equation 3.4.*

$$BN(A_1 A_2 \ldots A_n) = (X_2 - X_1) * (Y_2 - Y_1) \tag{3.4}$$

where

$$X_2 = min\{x_{1rt}, x_{2rt}, \ldots, x_{nrt}\},$$
$$X_1 = max\{x_{1lb}, x_{2lb}, \ldots, x_{nlb}\},$$
$$Y_2 = min\{y_{1rt}, y_{2rt}, \ldots, y_{nrt}\},$$
$$Y_1 = max\{y_{1lb}, y_{2lb}, \ldots, y_{nlb}\}.$$

Proof. Since the bounding neighborhoods of these n spatial events have the common intersection area, we can represent this intersection region as S, $S \subseteq BN(A_i)$, $for 1 \leq i \leq n$. For any point $(x, y) \in S$, we claim that $X_1 \leq x \leq X_2$ and $Y_1 \leq y \leq Y_2$. This claim can be proved by contradiction as follows.

Assume that $X_1 \leq x$ is not true; this assumption means that at least one value from the set $\{x_{1lb}, x_{2lb}, \ldots, x_{nlb}\}$ is greater than x. Without loss of generality, say $x_{ilb} > x$, since x_{ilb} is the left edge of the bounding neighborhood of the spatial event A_i, we can get $(x, y) \notin BN(A_i)$. Since $(x, y) \in S$, we get $S \not\subseteq BN(A_i)$, which contradicts the given condition that $S \subseteq BN(A_i)$. Hence $X_1 \leq x$ is true. Similarly, we can prove $x \leq X_2$ and $Y_1 \leq y \leq Y_2$ are true.

By Theorem 3.1 and Lemma 3.3, we can compute the bounding neighborhoods of features or feature sets without double counting the overlapping area.

For instance, in Figure 9, we can find three instances of feature A, so the bounding neighborhood of feature A is $\bigcup_{i=1}^{3} BN(A_i)$. According to Lemma 3.3, $\bigcup_{i=1}^{3} BN(A_i) = BN(A_1) + BN(A_2) + BN(A_3) - BN(A_1A_2) - BN(A_1A_3) - BN(A_2A_3) + BN(A_1A_2A_3)$. In addition, we can get $BN(A_1A_2)$, $BN(A_1A_3)$, $BN(A_2A_3)$, *and* $BN(A_1A_2A_3)$ by Theorem 3.1, so we can compute the correct value for $\bigcup_{i=1}^{3} BN(A_i)$ by Equation 3.2.

4 Algorithm Descriptions

Figure 10 presents an overview of algorithm designs for mining co-location patterns over extended spatial objects. In the figure, we show two pruning approaches. One is prevalence-based pruning using the anti-monotone property of the coverage ratio. This is similar to the support-based pruning in association-rule mining [2]. The second is a novel geometric filtering approach, which makes use of unique spatial characteristics of spatial objects and dramatically reduces the pattern search from a global space to local spaces.

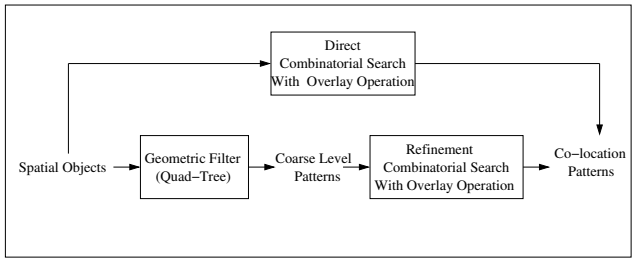

Figure 10: The Algorithm Design Illustration.

DCS: Direct Combinatorial Search Algorithm: One choice of co-location pattern mining is to use direct combinatorial search - an Apriori-like algorithm [2], in which we only apply prevalence-based pruning. All patterns satisfying the minimum prevalence threshold are candidate co-location patterns. Then, GIS overlay operations are applied to produce neighborhoods for these candidate co-location patterns. In GIS overlay operations, extensive geometric intersections are required. Hence, the computation cost of overlay operation is very high. Indeed, the computation cost for GIS overlay operations dominates in the DCS algorithm.

EXCOM: An Extended Co-location Mining Algorithm: We also design a more sophisticated algorithm, called an **EX**tended **CO**-location **M**ining algorithm (EXCOM) for mining co-location patterns over extended spatial objects. Figure 8 illustrates the pseudocode of the EXCOM algorithm, which follows a filter-and-refine paradigm and can prune the search space based on the following two criteria. 1) Pruning based on the anti-monotone property of the coverage ratio (Lemma 2.1). 2) Pruning based on a geometric filter - a quad-tree [6]. The difference between the EXCOM algorithm and Apriori-like approaches [2] is from the unique characteristics of spatial features. Specifically, in the EXCOM algorithm, we first apply the coarse-level co-location mining framework to find size-2 coarse-level co-location patterns and then conduct overlay operations to find size-2 co-location pattern. Finally, we generate co-location patterns with size greater than two using Apriori-like approaches.

In the EXCOM algorithm, the number of patterns required for GIS overlay operations is significantly reduced compared with the DCS algorithm. Since the computation cost of GIS overlay operations is extremely high, the computation cost of the EXCOM algorithm can be much cheaper than that of the DCS algorithm.

5 Experimental Evaluation

In this section, we present the results of extensive experiments on a real digital roadmap data set to evaluate the proposed buffer-based model and the EXCOM algorithm for mining co-location patterns over extended spatial objects. Specifically, we demonstrate: (1) the geometric filtering effect in the EXCOM algorithm. (2) the effectiveness of the buffer-based model for dealing with extended spatial data types, such as line strings. (3) the application of line-string co-location patterns for test route selection.

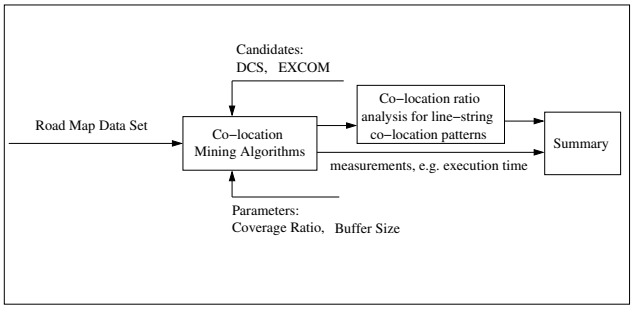

Figure 11: The Experimental Design

Experimental Data Sets. We conducted experiments on a real data set, namely a digital roadmap of the Minneapolis and St. Paul metropolitan area. The raw data is from the Minnesota Department of Transportation (MN/DOT) base map (http://rocky.dot.state.mn.us/basemap) and is stored in Shape File format that can be read and displayed by

GIS tools, such as Arc/View and Arc/Info. We transformed all the data into text format, including projected coordinates information and road type information for each road segment. There are a total of 511361 road segments in this dataset.

Experimental Design. To evaluate the filtering effect of the geometric component in the EXCOM algorithm, we compared the EXCOM algorithm with a direct combinatorial search (DCS) approach, as illustrated in Figure 11.

Experimental Implementation Platform. All experiments were performed on a Sun Ultra 10 workstation with a 440 MHz CPU and 128 Mbytes of memory running the SunOS 5.7 operating system.

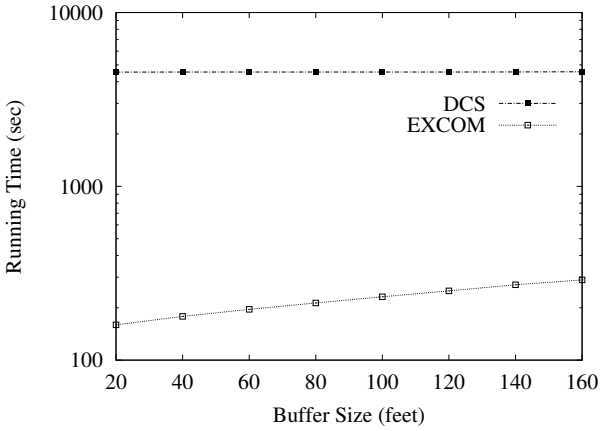

Figure 12: The Filtering Effect of the Geometric Component in the EXCOM algorithm.

5.1 The Filtering Effect of the Geometric Component in the EXCOM algorithm

In this experiment, we evaluated the filtering effect of the geometric component in the EXCOM algorithm using real digital roadmap data. For the purpose of comparison, the same prevalence threshold and buffer size were applied in both EXCOM and DCS algorithms. In other words, both algorithms were tested under the same experimental setting and produced the same set of co-location patterns.

Figure 12 shows the performance comparison between the direct combinatorial search algorithm (DCS) and the EXCOM algorithm. As can be seen, the execution time of the EXCOM algorithm is significantly less than that of the DCS algorithm. This can be explained by the fact that the DCS algorithm uses prevalence-based pruning only, while the EXCOM algorithm uses both prevalence-based pruning and geometric filtering. In this case, the geometric filter speeded up prevalence-based pruning by a factor of 30 - 40 as shown in the figure. This huge computation saving is due to the fact that GIS overlay operations dominate in both algorithms and a large number of GIS overlay operations were saved in the EXCOM algorithm.

We can also see that the computation performance of the DCS algorithm is not very sensitive to the buffer size. By contrast, the computation cost of the EXCOM algorithm is increased with the increase of the buffer size, since the performance of the geometric filter in this algorithm relies on the buffer size.

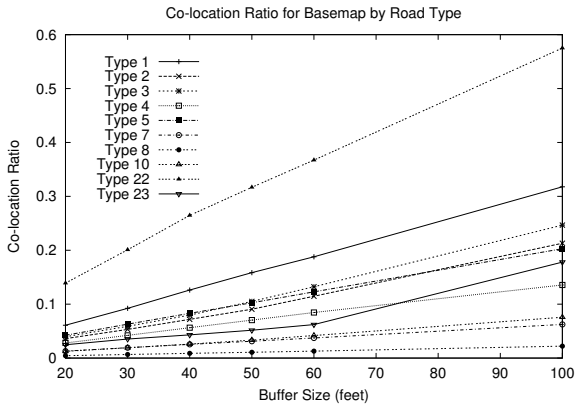

Figure 13: Illustration of Line-String Co-location Ratio for Different Road Types

5.2 Line-string Co-location Patterns

We also applied our buffer-based model to find line-string co-location patterns from the real digital roadmap data set. To the best of our knowledge, no techniques have been reported previously to discover such line-string co-location patterns in the literature. In this experiment, we present the line-string co-location ratio for each road type using different buffer sizes. The line-string co-location ratio is computed as

$$\frac{len(line-string\ co-locations\ within\ the\ neighborhood\ of\ the\ buffer)}{Total\ Length\ of\ the\ Corresponding\ Road\ Type}.$$

Figure 13 shows co-location ratios of several different road types in the MN/DOT base map. Here, we observed line-string co-location ratios with different buffer sizes including 20, 30, 40, 50, 60, and 100 feet. As can be seen, the co-location ratio goes up as the buffer sizes increase. Another interesting observation is that the co-location ratio for road type 22 is significantly higher than for other road types. In the MN/DOT base map definition, road type 22 is a ramp (please refer to ap-

pendix C). This finding indicates that the ramp is usually co-located with some other types of roads. Generally speaking, the co-location ratio provides a basic understanding of co-location pattern distributions over different road types.

5.3 The Application of Line-string Co-location Patterns for Test Route Selection

Here, we illustrate the application of line-string co-location patterns for selecting most challenging test routes, which are important for a novel GPS-based approach to accessing road user charges [20]. One common approach to evaluating digital roadmap accuracy is to measure the errors between a GPS track on a selected test route with a digital roadmap track. However, it is usually difficult to select a suitable test route for collecting GPS data. Consider that it is very often that errors happen near the dense road area among which the area that includes dense roads with different road types is the most important. Line-string co-location patterns from the digital roadmap provide a guide for identifying such error-prone areas.

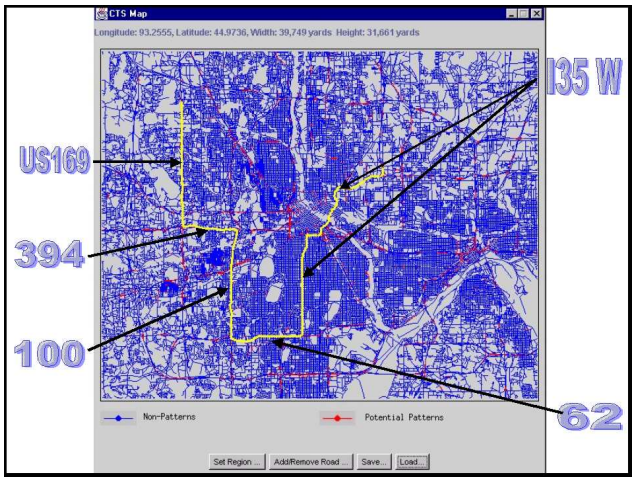

Figure 14: Field Test Route 1 in Twin Cities Area

In our project to evaluate the GPS-based approach to accessing road user charges [20], we were able to select five suitable test routes in the Minneapolis and St. Paul metropolitan area based on the line-string co-location patterns we identified using our extended co-location mining algorithm. These five routes were identified around areas having rich line string co-location patterns. For instance, one of these test routes is illustrated in Figure 14. For this test route, the highway part includes US 169, I-394, MN 100, MN 62, and I-35W in the Minneapolis and St. Paul metropolitan area.

6 Conclusion and Future Work

In this paper, we proposed a buffer-based model for mining co-location patterns over extended spatial objects. This model integrates the best features of the event centric model and applies a statistically consistent definition for the conditional probability measure. Also, we provided an extended co-location mining algorithm (EXCOM), which follows a filter-and-refine paradigm and can efficiently find co-location patterns. Finally, experimental results indicate that the geometric filter can speed up the prevalence-based pruning approach by a factor of 30 - 40 and a case study of applying line-string co-location for test route selection shows the value of co-location patterns for real world applications.

As for future work, with the definition of time windows, it is possible to extend the concept of co-location events into co-incidence events. Co-incidence patterns contain the events that are frequently occurred during the same time period.

7 Acknowledgments

This work was partially supported by NASA grant # NCC 2 1231, Center for Transportation Studies, Minnesota Department of Transportation and by Army High Performance Computing Research Center under the auspices of the Department of the Army, Army Research Laboratory cooperative agreement number DAAD19-01-2-0014, the content of which does not necessarily reflect the position or policy of the government and no official endorsement should be inferred. Access to computing facilities was provided by the AHPCRC and the Minnesota Supercomputing Institute. Finally, we thank Kim Koffolt for helping improve the readability of this paper.

References

[1] R. Agarwal, T. Imielinski, and A. Swami. Mining Association Rules between Sets of Items in Large Databases. In *Proc. of the ACM SIGMOD Conference on Management of Data*, pages 207–216, May 1993.

[2] R. Agarwal and R. Srikant. Fast Algorithms for Mining Association Rules. In *Proc. of the 20th Int'l Conference on Very Large Data Bases*, 1994.

[3] P. Albert and L. McShane. A Generalized Estimating Equations Approach for Spatially Correlated Binary Data: Applications to the Analysis of Neuroimaging Data. *Biometrics*, 1, 1995.

[4] Y. Chou. *Exploring Spatial Analysis in Geographic Information System.* Onward Press, ISBN: 1566901197, 1997.

[5] N. Cressie. *Statistics for Spatial Data.* Wiley and Sons, ISBN:0471843369, 1991.

[6] M. De Berg, M. Van Kreveld, M. Overmars, and O. Schwarzkopf. *Computational Geometry Algorithms and Applications*. Springer Verlag; 2nd edition, ISBN: 3540656200, Feb. 2000.

[7] M. H. DeGroot. Probability and statistics (second edition). *ADDSON WESLEY, (ISBN:020111366X)*, 1986.

[8] V. Estivill-Castro and I. Lee. Data Mining Techniques for Autonomous Exploration of Large Volumes of Geo-referenced Crime Data. In *Proc. of the 6th International Conference on Geocomputation*, 2001.

[9] V. Estivill-Castro and A. Murray. Discovering Associations in Spatial Data - An Efficient Medoid Based Approach. In *Proc. of the Second Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 1998.

[10] R. Haining. *Spatial Data Analysis in the Social and Environmental Sciences*. Cambridge University Press, Cambfidge, U.K, 1989.

[11] J. Han, M. Kamber, and A. Tung. Spatial clustering methods in data mining: A survey. *Miller, H., and Han, J., eds., Geographic Data Mining and Knowledge Discovery*, 2001.

[12] J. Hipp, U. Guntzer, and G. Nakaeizadeh. Algorithms for Association Rule Mining - A General Survey and Comparison. In *Proc. of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2000.

[13] Y. Huang, H. Xiong, S. Shekhar, and J. Pei. Mining Confident Co-location Rules without A Support Threshold. In *Proc. 2003 ACM Symposium on Applied Computing (ACM SAC'03)*, 2003.

[14] Issaks, Edward, and M. Srivastava. *Applied Geostatistics*. Oxford University Press, Oxford, ISBN:0195050134, 1989.

[15] K. Koperski and J. Han. Discovery of Spatial Association Rules in Geographic Information Databases. In *Proc. of the 4th International Symposium on Spatial Databases*, 1995.

[16] P. Krugman. *Development, Geography, and Economic theory*. MIT Press, Camgridge, MA, 1995.

[17] Y. Morimoto. Mining Frequent Neighboring Class Sets in Spatial Databases. In *Proc. ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2001.

[18] J. Park, M. Chen, and P. Yu. Using a Hash-Based Method with Transaction Trimming for Mining Association Rules. In *IEEE Transactions on Knowledge and Data Engineering*, volume 9, September 1997.

[19] S. Shekhar and Y. Huang. Co-location Rules Mining: A Summary of Results. In *Proc. 7th Intl. Symposium on Spatio-temporal Databases*, 2001.

[20] S. Shekhar and X. Ma. GIS Subsystem for a new approach to accessing road user charges. *http://www.cs.umn.edu/research/shashi-group/Project/CTS/cts_report_03.ps*, 2003.

[21] R. Srikant and R. Agrawal. Mining Generalized Association Rules. In *Proc. of the 21st Int'l Conference on Very Large Databases*, 1997.

[22] R. Srikant, Q. Vu, and R. Agrawal. Mining Association Rules with Item Constraints. In *Proc. of the 3rd Int'l Conference on Knowledge Discovery and Data Mining*, Aug 1997.

[23] P. Stolorz, H. Nakamura, E. Mesrobian, R. Muntz, E. Shek, J. Santos, J. Yi, K. Ng, S. Chien, R. Mechoso, and J. Farrara. Fast Spatio-Temporal Data Mining of Large Geophysical Datasets. In *Proceedings of the First International Conference on Knowledge Discovery and Data Mining*, 1995.

[24] W. Tobler. Cellular Geography, Philosophy in Geography. *Gale and Olsson, Eds., Dordrecht, Reidel*, 1979.

[25] C. Tsur, J. Ullman, C. Clifton, S. Abiteboul, R. Motwani, S. Nestorov, and A. Rosenthal. Query Flocks: a Generalization of Association-Rule Mining. In *Proc. of ACM SIGMOD Conf. on Data Management*, 1998.

[26] H. Xiong, P. Tan, and V. Kumar. Mining strong affinity association patterns in data sets with skewed support distribution. In *Proc. of the third IEEE International Conference on Data Mining*, pages pp. 387–394, 2003.

[27] Y. Yasui and S. Lele. A Regression Method for Spatial Disease Rates: An Estimating Function Approach. *Journal of the American Statistical Association*, 1997.

[28] Y. Zhao. *Vehicle Location and Navigation Systems*. Artech House TS Series, ISBN: 0890068615, 1997.

Appendix

Type	Meaning
01	Interstate Trunk Highway
02	U. S. Trunk Highway
03	Minnesota Trunk Highway
04	County State-aid Highway
05	Municipal State-aid Street
07	County Road
08	Township Road
09	Unorganized Township Road
10	Municipal Street
11	National Park Road
12	National Forest Development Road
13	Indian Reservation Road
14	State Forest Road
15	State Park Road
16	Military Road
17	National Monument Road
18	National Wildlife Refuge Road
19	Frontage Road
20	State Game Preserve Road
22	Ramp
23	Private Jurisdiction Road

Table 1: Road Types For MN/DOT Digital Base Map