Mining Frequent Geographic Patterns with Knowledge Constraints

Vania Bogorny
II – UFRGS
Caixa Postal 15.064
Porto Alegre/RS – Brazil
+ 55 51 3316 7017
vbogorny@inf.ufrgs.br

Sandro Camargo
II – UFRGS
Caixa Postal 15.064
Porto Alegre/RS – Brazil
+ 55 51 3316 7017
scamargo@inf.ufrgs.br

Paulo Martins Engel
II – UFRGS
Caixa Postal 15.064
Porto Alegre/RS – Brazil
+ 55 51 3316 6829
engel@inf.ufrgs.br

Luis Otavio Alvares
II – UFRGS
Caixa Postal 15.064
Porto Alegre/RS – Brazil
+ 55 51 3316 6843
alvares@inf.ufrgs.br

ABSTRACT

The large amount of patterns generated by frequent pattern mining algorithms has been extensively addressed in the last few years. In geographic pattern mining, besides the large amount of patterns, many are well known geographic domain associations. Existing algorithms do not warrant the elimination of all well known geographic dependences since no prior knowledge is used for this purpose. This paper presents a two step method for mining frequent geographic patterns without associations that are previously known as non-interesting. In the first step the input space is reduced as much as possible. This is as far as we know still the most efficient method to reduce frequent patterns. In the second step, all remaining geographic dependences that can only be eliminated during the frequent set generation are removed in an efficient way. Experiments show an elimination of more than 50% of the total number of frequent patterns, and which are exactly the less interesting.

Categories and Subject Descriptors

H.2.8 Database Applications: *Data mining, Spatial databases, and GIS*

General Terms

Design, algorithms

Keywords

Geographic databases, spatial association rules, frequent pattern mining, semantic knowledge constraints

1. INTRODUCTION

Association rules is a data mining technique that has been largely used for knowledge discovery in databases (KDD). As most discovery techniques it has the objective of identifying non-trivial, valid, novel, potentially useful, and ultimately understandable patterns from data [10]. Their main drawback, however, is the generation of a large number of patterns. In geographic databases (GDB) this problem increases significantly. Besides the large amount of patterns, many are well known natural geographic dependences intrinsic to the data. Different thresholds and syntactic constraints have been proposed to reduce the number of patterns, but they do not warrant the elimination of well known geographic dependences.

Figure 1 shows an example of a *well known* geographic dependence, where every gas station intersects at least one street. Figure 2 shows an example of non-standard spatial relationships, where gas stations and industrial residues repositories may either have a spatial relationship with water bodies or may not. There is no explicit pattern among these data. Considering, for example, that water analyses showed high chemical pollution, the extraction of spatial relationships among water resources, gas stations, and industrial residues repositories will be interesting for knowledge discovery.

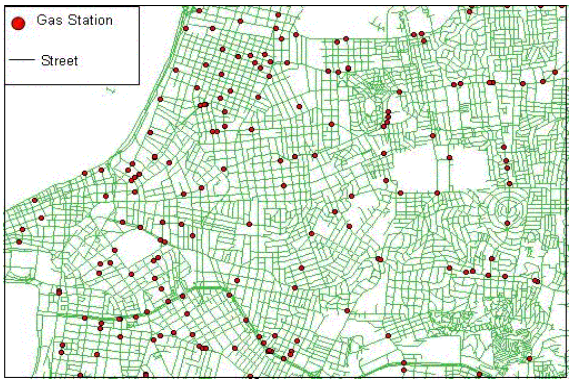

Figure 1. Examples of spatial relationships which produce well known patterns

Relationships such as *gas stations intersect streets* or i*slands within water bodies*, under rare exceptions or some geographic location inconsistency, hold for practical purposes in a 100% of the cases. The result is the generation of non-interesting rules, such as *is_a* (*x*,*island*) → *intersects* (*x*,waterBody) (100%) or *is_a* (*x*,*GasStation*) → *intersects* (*x,Street*) (100%).

Although users might be interested in high confidence patterns or rules, not all strong patterns necessarily hold considerable information. Moreover, the mixed presentation of thousands of interesting and uninteresting rules can discourage users from interpreting them in order to find 'patterns' of either novel or unexpected knowledge.

ACM-GIS'06, November 10–11, 2006 Arlington, Virginia, USA.

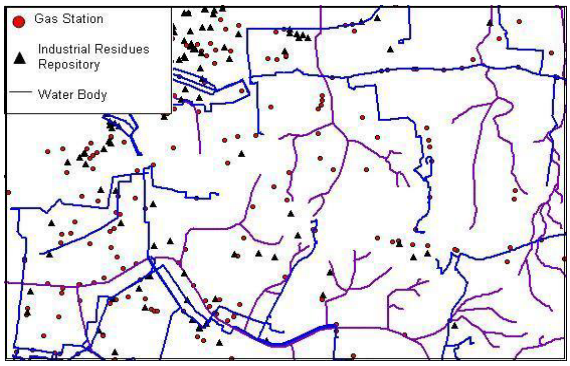

Figure 2. Examples of non-standard spatial relationships

Well known geographic dependences are mandatory spatial relationships which represent spatial integrity constraints. Such constraints must hold in order to warrant the consistency of spatial data in geographic databases. They are normally explicitly represented in geographic database schemas, as will be shown with two case studies. Existing algorithms for frequent pattern mining, however, have considered only data by themselves, while the schema, which is a rich knowledge repository, has not been considered so far in the discovery process. The result is that the same associations explicitly represented by the database designer are unnecessarily extracted by frequent pattern mining (FPM) algorithms and presented to the data mining user.

We claim that well known associations explicitly represented in geographic database schemas should be eliminated in frequent pattern mining. For this purpose, this paper presents a two step method for mining frequent geographic patterns without well known dependences.

1.1 Related Works and Contribution

For extracting frequent patterns from geographic databases there are basically two approaches in the literature. One is based on quantitative reasoning, which mainly computes distance relationships during the frequent set generation. Quantitative reasoning approaches [22][23] deal with geographic data (coordinates *x,y*) directly. Although they have the advantage of not requiring the definition of a reference object, they have some general drawbacks: usually deal only with points, consider only quantitative relationships, and do not consider non-spatial attributes of geographic data, which may be of fundamental importance for knowledge discovery. For spatial objects/features represented by lines or polygons, their centroid is extracted. This process, however, might lose significant information and generate non-real patterns (e.g. the Mississippi River *intersects* many states considering its real geometry, but is *far from* the same states if only the centroid is extracted).

The other is based on qualitative reasoning [2][8][9][13] and usually considers distance and topological relationships between a reference geographic object type and a set of relevant feature types represented by any geometric primitive (e.g. points, lines, and polygons). Relationships are normally extracted in a first step in data *preprocessing* tasks, while frequent patterns are generated in another step.

Both qualitative and quantitative reasoning approaches, however, have not focused on interesting geographic aspects to be considered in FPM. Neither do they make use of prior knowledge to specify which spatial relationships should be computed nor reduce the number of well known semantic patterns. [13], for example, presented a top-down, progressive refinement method to extract spatial predicates where patterns and rules are reduced using only minimum support. [8] presented a similar method for mining association rules from geographic objects with broad boundaries.

[2] proposed a method to extract all spatial features and spatial relationships to a deductive relational database. This process is computationally expensive since all spatial relationships are computed a priori. Although patterns and association rules can be reduced, the user has to specify a different pattern constraint for all different spatial relationships or possible association rules. Besides requiring a lot of background knowledge, pruning is performed in post processing steps, i.e., after both frequent sets and association rules have already been generated.

In order to reduce the number of spatial joins in geographic data preprocessing [4] we proposed to use geo-ontologies [3]. In [5] we proposed to remove well known geographic dependences in spatial association rule mining using geographic database schemas as prior knowledge.

In this paper we propose a general framework to completely eliminate geographic dependences in frequent geographic pattern mining. In a first step the input problem is reduced, and dependences are eliminated before computing any frequency. According to [6] this is still the most efficient way to prune frequent patterns. Additionally, in an efficient way, pairs of geographic objects with dependences are eliminated further and completely, during the frequent set generation.

Our framework can be either partially or totally applied to any algorithm that generates frequent sets, being it an Apriori-like approach or not. Its main advantage is its simplicity. In two single steps all dependences are removed, and more interesting patterns and rules will be generated. While dozens of approaches define syntactic constraints and different thresholds to reduce the number of patterns and association rules, we consider *semantic knowledge constraints,* and eliminate the exact pairs of geographic objects that produce well known patterns.

1.2 Scope and Outline

This paper proposes an approach for mining non-standard frequent patterns from geographic databases. The proposed approach uses qualitative spatial reasoning. The results show an improvement on the existing techniques for knowledge discovery in geographic databases.

The remainder of the paper is organized as follows: Section 2 describes the problem of geographic dependences in FPM. Section 3 presents two case studies of GDB schemas to show the large amount of well known geographic dependences. Section 4 presents a framework to eliminate well known geographic dependences in FPM. Section 5 presents experiments that show the significant frequent set reduction, while Section 6 concludes the paper and suggests some directions of future work.

2. THE PROBLEM OF GEOGRAPHIC DEPENDENCES IN FPM

At least two steps are required to extract patterns from GDB: the computation of spatial neighborhood relationships and the generation of frequent sets and association rules. Well known geographic dependences appear in both steps, and in different

ways, producing different amounts of well known patterns. In the following sections we show how geographic dependences appear in these two steps.

2.1 Geographic Dependences in Spatial Predicate Extraction

In transactional data mining, every row in the dataset to be mined is usually a transaction and columns are items, while in qualitative spatial data mining, every row is an instance of a reference object type (e.g. city), called *target feature type*, and columns are predicates. Every predicate is related to a non-spatial attribute (e.g. population) of the target feature type, or a *relevant feature type* that is spatially related to the target feature type (e.g. *intersects(gasStation)*). Table 1 shows an example of a spatial dataset where every row is a city (target feature type) and the predicates are different geographic object types (port, water body, hospital, treated water network, factory) spatially related to city. Spatial predicates can be represented at different granularity levels [12], according to the objective of the discovery. In Table 1, data are represented at a general granularity level, but lower levels as, for example, chemical, metallurgical, and textile factories could be used instead of factory.

Table 1. Dataset in a high granularity level for FPM

Tuple (city)	Spatial Predicates
1	contains(Port), contains(Hospital), contains(TreatedWaterNet), contains(Factory), crosses(WaterBody)
2	contains(Hospital),contains(TreatedWaterNet), crosses(WaterBody)
3	contains(Port), contains(TreatedWaterNet), contains(Factory), crosses(WaterBody)
4	contains(Port), contains(Hospital), contains(TreatedWaterNet), crosses(WaterBody)
5	contains(Port), contains(Hospital), contains(TreatedWaterNet), contains(Factory), crosses(WaterBody)
6	contains(Hospital), contains(TreatedWaterNet), contains(Factory)

Spatial predicates are materialized spatial relationships extracted with spatial joins between all instances t (e.g. NewYorkCity) of a target feature type T (e.g. city) and all instances r (e.g. Route68) of every relevant feature type R (e.g. road) in a set of relevant feature types S (e.g. road, hospital, factory). In this step, a cartesian product between T and S is performed. Being $T=\{t_1, t_2,...,t_n\}$, $S = \{ R_1, R_i,..., R_m\}$, and $R_i = \{ r_1, r_2,..., r_q\}$, the extraction of spatial predicates implies the comparison of every instance of T with all instances of R, for all R in S. This process is the bottleneck of computational time in spatial data mining.

Well known geographic dependences may exist among T and any R in S, or between pairs of R in S. For example, in the dataset shown in Table 1, there is a well known dependence between the target feature type (city) and treated water network, because every city has at least one treated water network. This means that the predicate *contains*(*TreatedWaterNet*) has a 100% support and a large number of both patterns and rules with this predicate will be generated, such as, for example, *contains(factory)* → *contains(TreatedWaterNet)*. Such a rule expresses that cities that contain factories do also contain treated water networks. Although the rule seems to be interesting, it can be considered obvious due the simple fact that *all* cities have treated water networks, having they factories, or not.

Predicates with 100% support appear in most generated patterns and rules. Table 2 shows the result of an experiment with the dataset in Table 1, using minimum support 20% and 50%. Considering 20% minimum support, 31 frequent sets and 180 rules were generated. Among the 31 frequent sets and the 180 rules, 16 frequent sets and 130 rules had the dependence *contains(TreatedWaterNet)*. Increasing minimum support to 50% does not warrant the elimination of the geographic dependence. Although the number of frequent sets is reduced to 25 and rules to 96, 13 frequent sets and 72 rules still had the dependence.

Table 2. Frequent Patterns and rules with dependences

MinSup %	All Frequent Sets/ Rules	Rules with Dependence / Rules without Dependence	FrequentSets with dependence / FrequentSets without dependence
20	31 / 180	130/ 50	16/15
50	25 / 96	72 / 24	13/12

In the previous example, the high number of patterns including the geographic dependence was generated because of a dependence between the target feature type and a relevant feature type. This kind of dependences can be eliminated by pruning the input space, because such dependences with a 100% support only hind the discovery process. However, dependences may also exist among relevant features. In the dataset shown in Table1, there is also a dependence between *Port* and *Water Body*, where all cities which have *Ports* do also have *Water Bodies,* because every *Port* must be related to at least one *Water Body*. In this case, however, we cannot prune the input space because either *Water Body* or *Port* may have an interesting association with any other relevant feature type (Hospital, Treated Water Network, Factory). In the following section we introduce the problem of geographic dependences among relevant feature types.

2.2 Geographic Dependences in Frequent Sets

According to [1], the general problem of mining frequent predicate sets and association rules can be decomposed in two steps:

- *Find all large/frequent predicates sets*: a set of predicates is large if its support is at least equal to a certain threshold, called *minsup*.
- *Generate strong rules*: a rule is strong if its support is at least equal to minimum support and the confidence is higher or equal to a certain threshold, called *minconf.*

Assertion 1. If a predicate set Z is frequent, then every subset of Z will also be frequent. If the set Z is infrequent, then every set that contains Z is infrequent too. All rules derived from Z satisfy the support constraint if Z satisfies the support constraint.

To find *frequent predicate sets* and *extract strong association rules,* relevant feature types R in S are combined with each other for the different instances of the target feature type T, and not among T and R as in the previous problem.

To illustrate the geographic dependence replication process in frequent pattern mining, let us consider the well known method for frequent set generation introduced by [1]. Table 3 shows the frequent sets extracted from the dataset in Table 1 with 50% minimum support, where k is the number of elements in the frequent sets.

Geographic dependences appear the first time in frequent sets with 2 elements, where k=2. Notice that since the dependence has minimum support, i.e., is a frequent predicate set, this dependence

is replicated to many frequent sets of size $k>2$ with predicates that reach minimum support, as shown in bold style in Table 3. Considering such a small example and high minimum support (50%), one single geographic dependence participates in 6 frequent sets, which represents 30% of the total number of frequent sets. Notice that the number of rules having a geographic dependence will be much larger than the frequent sets, mainly when the largest frequent set (with 4 elements) contains the dependence.

In Table 3 we can clearly observe that the technique of generating closed frequent sets [17] does not warrant the elimination of geographic dependences. Geographic dependences generate their own closed frequent set, which in the example, are the two largest sets.

Table 3. Frequent predicate sets with minimum support 50%

k	Frequent sets with support 50%
1	{contains(Port)}, {contains(Hospital)}, {contains(TreatedWaterNet)}, {contains(Factory)}, {crosses(WaterBody)}
2	{Contains(Port),contains(Hospital)}, {Contains(Port),contains(TreatedWaterNet)}, {Contains(Port),contains(Factory)}, ***{Contains(Port),crosses(WaterBody)},*** {Contains(Hospital),contains(TreatedWaterNet)}, {Contains(Hospital),contains(Factory)}, {Contains(Hospital),crosses(WaterBody)}, {Contains(TreatedWaterNet),contains(Factory)}, {Contains(TreatedWaterNet),crosses(WaterBody)}, {Contains(Factory),crosses(WaterBody)},
3	{Contains(Port),contains(Hospital),contains(TreatedWaterNet)}, ***{Contains(Port),contains(Hospital),crosses(WaterBody)},*** ***{Contains(Port),contains(TreatedWaterNet),crosses(WaterBody)},*** ***{Contains(Port),contains(Factory),crosses(WaterBody)},*** {Contains(Port),contains(TreatedWaterNet),contains(Factory)}, {Contains(Hospital),contains(TreatedWaterNet),contains(Factory)} {Contains(Hospital),contains(TreatedWaterNet),crosses(WaterBody)}, {Contains(TreatedWaterNet),contains(Factory),crosses(WaterBody)}
4	***{Contains(Port),contains(Hospital),contains(TreatedWaterNet),crosses(WaterBody)}*** ***{Contains(Port),contains(TreatedWaterNet),contains(Factory),crosses(WaterBody)}***

After understanding the replication process of geographic dependences in FPM, the next section presents two case studies to evaluate the amount of well known dependences in real GDB.

3. GEOGRAPHIC DEPENDENCES: A CASE STUDY

Geographic dependences are mandatory spatial relationships among geographic objects, and are normally represented through associations with cardinality constraints one-one and one-many [18][19 pp.36-37]. Geographic dependences are *well known* because they are explicitly represented by database designers to warrant the spatial integrity [18] of geographic data. In geographic database schemas, geographic dependences are given by a spatial relationship or a single association, aggregation with cardinalities *one-one* or *one-many*.

In order to evaluate the amount of well known geographic dependences explicitly represented in real geographic databases, two real schemas were analyzed: the Brazilian Army data model, which has been the basis to construct geographic maps for the whole country, and the data warehouse developed in the project *iPara* for the Para state, in Brazil.

The geographic database schema developed by the Brazilian Army contains all geographic elements that are part of the Brazilian terrain model, which under a few variations, is similar to any terrain model represented in geographic databases. On account of the large number of objects and relationships to be represented, geographic data conceptual schemas are usually designed in different packages/superschemas. The geographic database schema developed by the Brazilian Army has 8 packages: edification, infra-structure, hydrography, vegetation, administrative regions, referential, relief, and toponymy. The package infra-structure, for example, is divided in six sub-schemes, including information about transportation, energy, economy, communication, etc. The hydrography package, for example, represents objects such as rivers, oceans, and lakes.

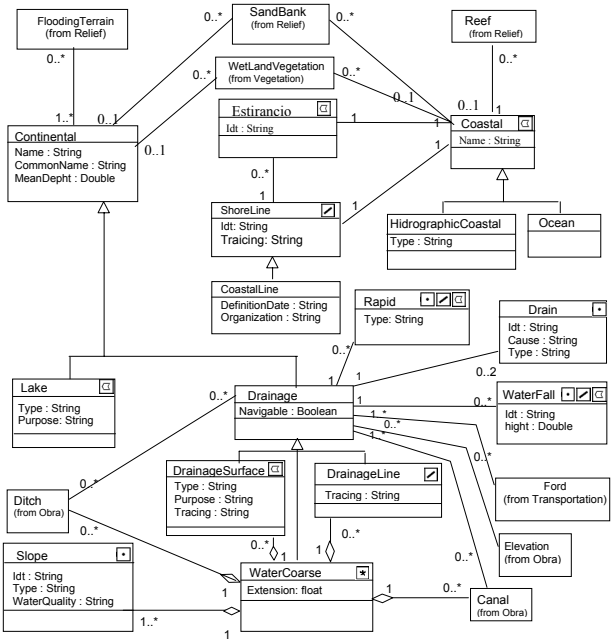

Figure 3. Part of the conceptual object-oriented schema of the Brazilian Geographic Territory (MCOO of EBG - Brazilian Army – STI – DSG - 1ºDL)

Information of different packages may be extracted for data mining, and the number of *one-one* and *one-many* relationships varies from one package to another. For example, the hydrography package, which is shown in Figure 3, has a total of 24 geographic objects (15 from its own package and 8 from others) which share 2 *one-many* relationships and 16 *one-one* relationships if super classes are concrete, and more that 20 if super classes are abstract.

The infra-structure level, for example, has 73 geographic objects in its own package and has relationships with 88 objects from other packages. Among the 88 relationships, 70 are mandatory *one-one* dependences.

The conceptual schema of the project *iPara* is a geographic data warehouse developed in cooperation with II/UFRGS, COHAB-PA, and SEIR-PA. It integrates general geographic data of the state of Para (SIGIEP) and the urban geographic database (SIME). The complete conceptual schema of *iPara* has more than 20 different packages such as Hidrography, Infra-Structure, and Transportation. The Transportation package, for example, has 29 objects with 19 *one-one* or *one-many* associations. Because of space limitations, a very small part of the Water Distribution schema is shown in Figure 4, where among 7 objects there are 2 *one-many* and 4 *one-one* associations.

The case study with two real schemas showed that a large number of mandatory well known geographic dependences is explicitly represented. If used as prior knowledge in frequent pattern mining, their extraction can be avoided and the generation of obvious patterns and rules significantly reduced.

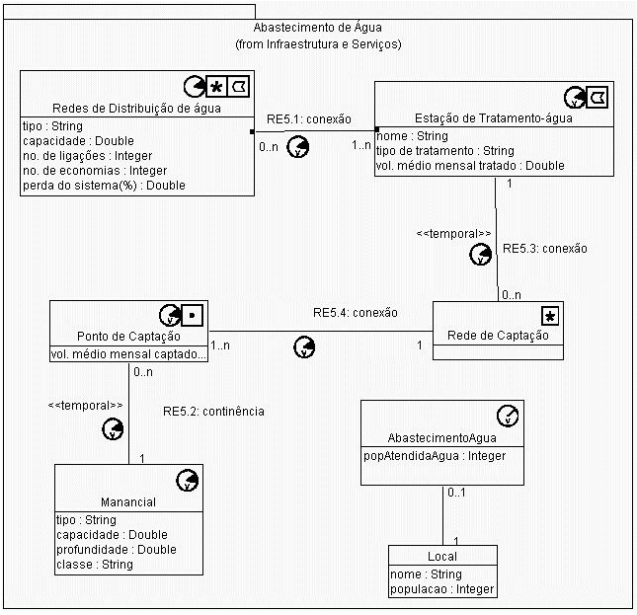

Figure 4. Part of the *iPara* conceptual schema

4. A FRAMEWORK FOR MINING FREQUENT GEOGRAPHIC PATTERNS WITH KNOWLEDGE CONSTRAINTS

Aiming to provide a complete and integrated framework for frequent geographic pattern mining without well known associations, Figure 5 shows an interoperable framework that supports the complete discovery process. To better understand the process, the framework can be viewed at three levels: data repository, data preprocessing, and pattern mining.

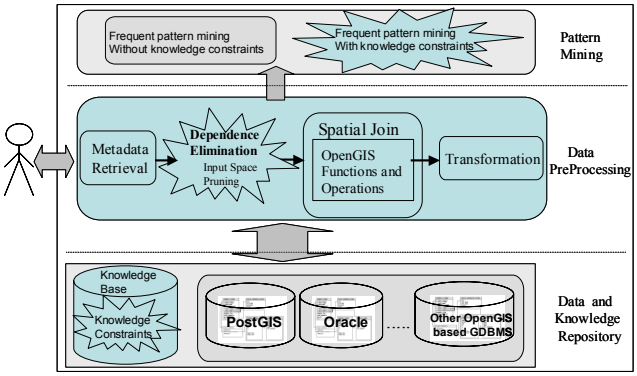

Figure 5. A Framework for FPM in geographic databases

At the bottom are the geographic data repositories, stored in GDBMS (geographic database management systems), constructed under OGC [16] specifications. There is also a knowledge repository which stores the pairs of geographic objects with dependences. These pairs can be either specified by the user or automatically retrieved with processes of reverse engineering [7] if the schema is not available. Different approaches to extract dependences from relational databases with reverse engineering are available in the literature. For knowledge discovery in non-geographic databases reverse engineering has been used to understand the data model [15] in legacy systems, or to automatically extract SQL queries [20], but not as prior knowledge to reduce well known patterns. In [5] we presented an algorithm to extract geographic dependences from database schemas. When provided by the user, a larger set of dependences can be specified; not only associations explicitly represented in the schema, but other application domain dependences which generate well known patterns.

In the center of the figure is the spatial data preprocessing level which covers the *gap* between data mining tools and geographic databases. At this level the data repositories are accessed through JDBC/ODBC connections and data are retrieved, preprocessed, and transformed into the single table format. At this level, dependences among the target feature and relevant features are removed. This step prunes the input space and reduces the number of spatial joins, as will be explained in the next section.

On the top are the algorithms for FPM. At this level a method for generating frequent sets for geographic data is proposed to eliminate all dependences between the relevant features types, i.e., those which cannot be removed from the input dataset. This step is explained in more detail in section 4.2.

4.1 Data Preprocessing: Eliminating Geographic Dependences among the Target Feature Type and the Relevant Feature Types

There are four main steps to implement the tasks of extracting spatial predicates for mining frequent geographic patterns: *metadata retrieval, dependence elimination, spatial join,* and *transformation.*

The *Metadata Retrieval* step retrieves all relevant information from the database, including the target feature type *T,* the target feature non-spatial attributes and the set of relevant feature types *S,* defined by the user, that may have some influence on *T*. The feature types are retrieved through the Open GIS database schema metadata.

In general words, the process described in this section can be summarized in the algorithm shown in Figure 6, where GDB is the geographic database, ϕ is the set of pairs of geographic objects with dependences, *T* is the target feature type*, S* is the set of relevant feature types *R,* and *x* is the spatial relationship (e.g. topology, distance).

The *Dependence Elimination* step verifies all associations between the target feature type and all relevant feature types. It searches the knowledge base and if *T* has a dependence with any *R* in *S*, then *R* is eliminated from *S*. For each relevant feature type removed from *S*, no spatial join is required to extract spatial relationships. By consequence, neither frequent sets nor spatial association rules will be generated with this relevant feature type.

The *Spatial Join* step computes and materializes the user-specified spatial relationships between the *T* and all *R* in *S*, retrieved by the *Metadata Retrieval* step and filtered by *Dependence Elimination.*

Spatial joins to extract spatial predicates are performed on-the-fly with operations provided by the GIS. Following the OGC

specifications makes our framework interoperable with all GDBMS constructed under OGC specifications (e.g. Oracle, PostGIS). Before computing spatial joins, MBR (minimum boundary rectangle) is performed for accelerating the extraction of spatial relationships.

```
Given: GDB, ϕ, T, S, x;
Find: a dataset Ψ without geographic
      dependences between T and S;

Method:
     Ψ = T – geometric_abribute;
     Dependence_Elimination
     Begin
        For (i=1; i=#R in S, i++) do begin
           If (T has a dependence with Ri in ϕ)
               Remove Ri from S; //input pruning
           Else
               Ψ = Ψ ∪ spatial_join (x,T,Ri);
        End;
     End;
     Transformation (Ψ);
```

Figure 6. Pseudo-code of the preprocessing function to extract spatial predicates

The *Transformation* step transposes the *Spatial Join* step output into a single table Ψ, understandable by FPM algorithms.

4.2 Pattern Mining: Eliminating Geographic Dependences among Relevant Feature Types

Frequent pattern and association rule mining algorithms, under rare exceptions [11] generate candidates and frequent sets. In spatial data mining, the candidate generation is not a problem because the number of predicates is much smaller than the number of items in transactional databases [19 p.205]. The computational coast relies on the spatial predicate extraction (number of instances of both target and relevant feature types), which our method reduces by pruning the input space.

Approaches that generate closed frequent sets [17] and eliminate redundant rules [24] do previously compute the frequent sets, and than verify if they are closed. Although they reduce the number of frequent sets, they do not warrant the elimination of all well known geographic patterns.

Considering that Apriori [1] has been the basis for dozens of algorithms for mining spatial and non-spatial frequent sets we illustrate the method of geographic dependence elimination during the frequent set generation using Apriori-KC [5], as shown in Figure 7.

Given ϕ as the set of pairs of geographic objects with dependences (e.g. Island, Water) called *knowledge constraints*, Ψ as the input dataset generated in the previous step, and *minsup* as minimum support, well known geographic dependences are removed from the frequent sets with 2 elements, when the dependence appears the first time.

Similarly to [21], which eliminates in the second pass frequent sets that contain both parent and child specified in concept hierarchies, we propose to eliminate all frequent sets which contain geographic dependences, independently of any concept hierarchy.

The dependences are eliminated in an efficient way, in one step, in the second pass, when generating candidates with 2 elements, and when it appears at the first time. Through this elimination, no frequent sets with two or more predicates having the dependence will be generated. According to Assertion1, this step *warrants* that the pairs of geographic objects with dependences in ϕ will neither appear together in the frequent sets nor in the spatial association rules. This makes the method effective and independent of any threshold such as minimum support, minimum confidence, lift, etc.

```
Given: ϕ,Ψ, minsup;
Find: frequent geographic pattern without well
      known dependences

Method:
L1 = {large 1-predicate sets};
For ( k = 2; Lk-1 != ∅; k++ ) do  begin
    Ck = apriori_gen(Lk-1);// New  candidates
    If (k=2)
       C2  = C2 - ϕ; // frequent set prunning
    Forall  rows w ∈ Ψ  do begin
        Cw = subset(Ck, w); // Candidates in w
        Forall candidates c ∈ Cw do
            c.count++;
    End;
    Lk = {c ∈ Ck | c.count ≥ minsup};
End;
Answer = ∪kLk
```

Figure 7. Frequent set generation function with Apriori-KC

The main strength of this method in our framework is its simplicity. This single, but very effective and efficient step, removes all well known geographic dependences, and can be implemented by any algorithm that generates frequent sets. Considering the example of frequent sets shown in Table 3, the dependence is eliminated when it appears at the first time, in the second pass, such that no larger frequent sets or association rules with the dependence will be generated.

It is important to emphasize that no information is lost with our method, only well known patterns are eliminated. For instance, suppose that *AB* is a frequent set having a dependence. This pair is eliminated with the purpose to avoid the generation of larger frequent sets that contain the dependence, such as *ABC,* for example. If the set *ABC* has minimum support, then the pairs *AB, AC,* and *BC* reached minimum support too. As we eliminate only pairs with dependences, *AC* and *BC* which combine the attribute *C* with both *A* and *B* separately, are still generated, and no information is lost. Our method only eliminates patterns that are well known, and does not sacrifice the result quality.

5. EXPERIMENTS AND EVALUATION

The proposed framework was implemented in Weka, which we extended to support automatic spatial predicate extraction with intelligent input pruning. In order to evaluate the interoperability of the framework, experiments were performed with real geographic databases stored under Oracle 10g and PostGIS. Census sectors and districts, with non-spatial attributes such as population, sanitary condition, etc, were defined as the target feature types for different experiments. Datasets with different relevant feature types (e.g. bus routes, slums, water bodies,

factories, gas stations, cellular antennas) were preprocessed and mined, using prior knowledge and without using prior knowledge.

To precise the time reduction to compute spatial joins for mining frequent patterns is very difficult, since this step is totally data dependent. The computational time reduction to extract spatial joins depends on three mains aspects: the number of dependences (relevant feature types) eliminated in data preprocessing; the geometry type of the relevant feature (point, line or polygon); and the number of instances of the eliminated feature type (e.g. 60000 rows). For example, if a relevant feature type with 57580 polygons is eliminated, spatial join computation would significantly decrease. If the eliminated feature type has 3062 points, for instance, time reduction would be less significant. However, for every eliminated relevant feature type, no spatial join is necessary, and this warrants preprocessing time reduction.

To evaluate the frequent pattern reduction by pruning the input space, Figure 8 describes an experiment where one dependence between the reference object and the relevant feature types was eliminated. Notice that input space pruning reduces frequent patterns for all different values of minimum support. Considering *minsup* 10%, 20%, and 30%, the elimination of one single dependence in data preprocessing pruned the frequent sets around 50%. The rule reduction is still more significant than the frequent set pruning, reaching 70% by eliminating one single dependence.

Algorithms that generate closed frequent sets [17], reduce the number of rules [14], and eliminate redundant rules [24] can reduce still further the number of both frequent sets and association rules if applied to the geographic domain using our method for pruning the input space.

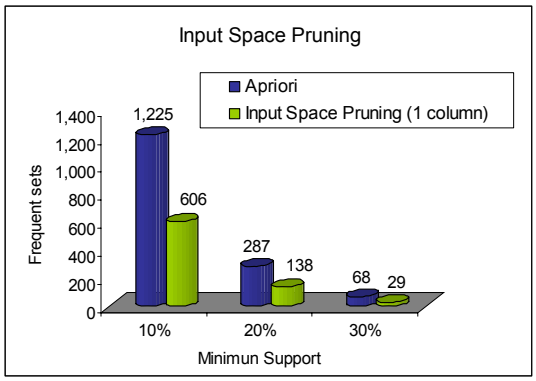

Figure 8. Input space pruning

Figure 9 shows the result of an experiment where two dependences among relevant feature types were eliminated during the frequent set generation without input pruning. Even pruning only the frequent sets, our method reduces the number of frequent sets for all different values of minimum support. Indeed, the higher the number of dependences, the more significant is the reduction.

Figure 10 shows an experiment where dependences were eliminated in both input space (between the target feature and relevant features) and during the frequent set generation (among relevant features). The total number of frequent sets is reduced in an average of 60% by removing one dependence, independently of minimum support. This experiment shows that in the geographic domain most frequent sets contain well known geographic dependences, and our method completely eliminates such dependences very fast. Because of space limitations, time reduction experiments are not presented.

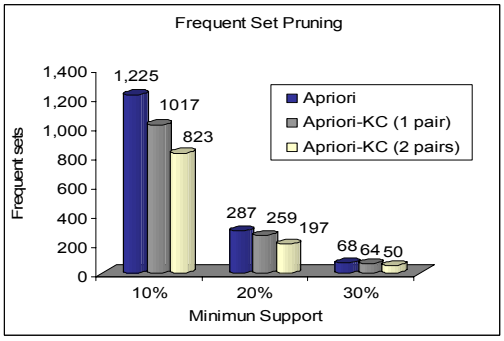

Figure 9. Frequent set pruning

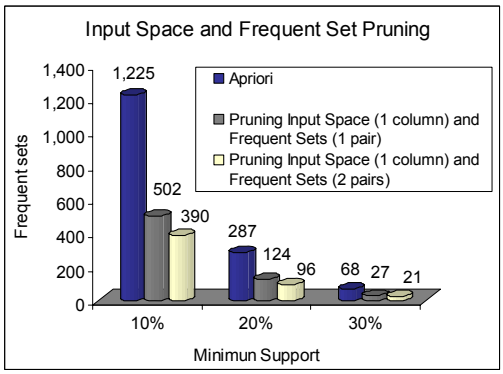

Figure 10. Input space and frequent set pruning

6. Conclusions and Future Works

This paper presented an intelligent framework for mining frequent geographic patterns without well known geographic dependences. Dependences are mandatory geographic associations which are explicitly represented in geographic database schemas. We showed that explicit mandatory relationships produce irrelevant patterns, while the non-standard spatial relationships may lead to more interesting knowledge.

Experiments showed that independent of the number of elements, geographic dependences generate a large number of frequent patterns without interesting knowledge. The elimination of one dependence is enough to prune a large number of patterns, but the higher the number of well known dependences to be eliminated, the larger is the pattern reduction. We showed that well known dependences can be partially eliminated by either pruning the input space or the frequent sets. Applying both steps eliminate known geographic dependences completely!

The main contribution is for the data mining user, which will analyze much less obvious patterns. The method is effective independently of other thresholds, and it warrants that known geographic domain associations will not appear among the discovered patterns.

The use of prior knowledge in geographic pattern mining has three main advantages: spatial relationships between feature types with dependences are not computed; the number of both frequent sets and association rules is significantly reduced; and the most important, the generated frequent sets and rules are free of associations that are previously known as non-interesting.

Next ongoing steps of this work include the evaluation of our methods with the closed frequent sets approach.

7. ACKNOWLEDGMENTS

Our thanks for both CAPES and CNPQ which partially provided the financial support for this research. To professor Cirano Iochpe and 1DL for the geographic database schemas. To Procempa, for the real geographic database.

8. REFERENCES

[1] Agrawal, R., and Srikant, R. Fast Algorithms for Mining Association Rules in Large Databases. In *Proceedings of the 20th International Conference on Very Large Data Bases (VLDB'94).* Morgan Kaufmann, Santiago,Chile, 1994, 487 – 499.

[2] Appice, A., Ceci, M., Lanza, A., Francesca, L., and Malerba, D. Discovery of Spatial Association Rules in Geo-Referenced Census Data: A Relational Mining Approach. *Intelligent Data Analysis* 7(6), (2003), 542-566.

[3] Bogorny, V., Engel, P., and Alvares, L.O. Towards the reduction of spatial joins for knowledge discovery in geographic databases using geo-ontologies and spatial integrity constraints. In *ECML/PKDD Second Workshop on Knowledge Discovery and Ontologies (KDO'05)* (October). Porto, Portugal, 2005, 51-58.

[4] Bogorny, V., Engel, P., and Alvares, L.O. GeoARM: an interoperable framework to improve geographic data preprocessing and spatial association rule mining. In *Proceedings of the 18th International Conference on Software Engineering and Knowledge Engineering (SEKE'06)* (July). San Francisco, 2006, 79-84.

[5] Bogorny, V., Camargo, S., Engel, P., and Alvares, L.O. Towards elimination of well known geographic domain patterns in spatial association rule mining. In *Proceedings Of The 3th IEEE International Conference On Intelligent Systems (IEEE IS'06)* (September 4-6). London, 2006 (To appear).

[6] Bonchi F., Giannotti F. Mazzanti A. and Pedreschi D. ExAMiner: Optimized level-wise frequent pattern mining with monotone constraints. In *Proceedings of the IEEE International Conference on Data Mining (ICDM'03*) (December 19-22). IEEE Computer Society, Melbourne, Florida, 2003, 11-18.

[7] Chifosky E.J., and Cross J. H. Reverse engineering and design recovery: a taxonomy. *IEEE Software*, Jan.1990.

[8] Clementini, E., Felice, Di, Koperski, K. Mining multiple-level spatial association rules for objects with a broad boundary. *Data & Knowledge Engineering.* 34 (3) (Sep. 2000). Elsevier Science Publishers, Amsterdan, 251–270.

[9] Ester, M., Frommelt, A., Kriegel, H.-P., and Sander, J. Spatial Data Mining: Database Primitives, Algorithms And Efficient DBMS Support. *Journal of Data Mining and Knowledge Discovery*, 4(2-3) (Jul. 2000), 193-216.

[10] Fayyad, U., Piatetsky-Shapiro, G., and Smyth, P. From data mining to discovery knowledge in databases**.** *AI Magazine*, 3(17), (1996) 37-54.

[11] Han J., Pei J., Yin Y., and Mao R. Mining frequent patterns without candidate generation: a frequent-pattern tree approach*. Data Mining and Knowledge Discovery.* 2004 8(1), 53-87.

[12] Han, J., and Fu, Y. Discovery of multiple-level association rules from large databases. In *Proceedings of the 21th International Conference on Very Large Data Bases* (VLDB'95), (U. Dayal, P.M.D. Gray, S. Nishio). Morgan-Kaufmann, Zurich, Switzerland, 1995, 420–431.

[13] Koperski, K., and Han, J. Discovery of Spatial Association Rules in Geographic Information Databases. In *Proceedings Of The 4th International Symposium On Large Geographical Databases* (SSD'95), Portland, Maine, 1995, 47-66.

[14] Liu B., Hsu W., and Ma Y. Pruning And Summarizing The Discovered Associations. In *Proceedings of the Fifth ACM-SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'99).* ACM. San Diego, California,1999, 125–134.

[15] MCKearney, S., and Roberts, H. Reverse engineering databases for knowledge discovery. In *Proceedings of the Second ACM-SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'96)*. AAAI Press. Oregon, 1996, 375-378.

[16] Open GIS Consortium (1999). OpenGIS simple features specification for SQL. In URL: http://www.opengeospatial.org/docs/99-054.pdf

[17] Pasquier, N. Bastide, Y., Taouil, R., and Lakhal, L . Discovering frequent closed itemsets for association rules. In Beeri, C., Buneman, P., eds.: *Proceedings of the 7th International Conference on Database Theory (ICDT '99).* Springer, Jerusalem, Israel, 1999, 398-416.

[18] Servigne, S. et al.: A Methodology For Spatial Consistency Improvement of Geographic Databases. *Geoinformatica*. 4(1), 2000 7-34.

[19] Shekhar, S., and Chawla, S. *Spatial Databases: A Tour*. Prentice Hall, Upper Saddle River, Nj, 2003.

[20] Shoval P., and Shreiber N. Database reverse engineering: from the relational to the binary relationship model. *Data and Knowledge Engineering.* (10), 1993, 293-315.

[21] Srikant, R., and Agrawal, R. Mining Generalized Association Rules, In *Proceedings of the 21st International Conference on Very Large Databases (VLDB'95).* Morgan Kaufmann, Zurich, Switzerland, 1995, 407-419.

[22] Yoo J.S., Shekhar S. and Celik M. A Join-less Approach for Co-location Pattern Mining: A Summary of Results, In *Proceedings of the IEEE International Conference on Data Mining(ICDM'05)*, Houston, 2005, 813-816.

[23] Yoo, J.S., and Shekhar S.. A partial join approach for mining co-location patterns. In *Proceedings of the 12th International Symposium on Geographic Information Systems (ACM-GIS'04)*, Washington, November, 2004, 241-249.

[24] Zaki. M. Generating Non-Redundant Association Rules. In *Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery And Data Mining (KDD'00)*. Boston, Massachussets, Usa, 2000, 34-41.