An ILP Method for Spatial Association Rule Mining

Donato Malerba and Francesca A. Lisi

Dipartimento di Informatica, Università degli Studi
via Orabona, 4 - 70126 Bari - Italy
{malerba|lisi}@di.uniba.it

Abstract. Knowledge discovery in spatial databases raises challenging multi-relational data mining problems. A promising solution approach comes from the field of inductive logic programming (ILP). In this paper, an ILP method for spatial association rule mining is presented. It benefits from the available prior knowledge on the spatial domain, systematically explores the hierarchical structure of geographic layers, and deals with numerical aspatial properties of spatial objects. The method has been implemented into the ILP system SPADA which operates on a deductive relational database set up by an initial step of feature extraction from a spatial database. Advantages and limits of the method are illustrated by means of examples taken from an application of SPADA to the spatial data of an Italian province.

1 Introduction

Most of the currently available data mining algorithms are based on the so-called attribute-value (AV) approach which requires the reduction of a multi-relational database to the single table format. The issue of mining from multiple relations has been extensively studied in the field of ILP [8]. Besides the elegant solution to multi-relational data mining, the strenght of the ILP approach is the common background with *deductive relational databases* (DDB) which, e.g., can be fully exploited to implement the notion of inductive database [20] as pointed out by Flach [11]. In recent times, a database (DB) approach to multi-relational data mining has been presented [15]. It exploits the semantic information in the database schema to prune the search space and defines database primitives to ensure efficiency.

The growing interest in multi-relational data mining is boosting the development of data mining methods that are more suitable for inherently multi-relational data such as geo-referenced data. Knowledge discovery in spatial databases raises challenging multi-relational data mining problems because of the high number of object classes and relationships between each pair of object classes [14]. In fact, the explicit location and extension of spatial objects define implicit relations of spatial neighborhood [10] that should be materialized to enable the DB approach. In this work we claim that ILP provides a promising solution approach to spatial data mining. Indeed, the underlying theory of computational logic supplies representation

and reasoning means that are particularly appropriate for the spatial domain. In particular, spatial relations may be inferred by qualitative reasoning. This augmented expressive power is a relevant novelty with respect to the research in the field to date which has generally just bolted spatial constructs on top of well-established statistical techniques in order to accommodate the spatial dimension [24]. To the best of our knowledge, very few contributions from ILP to spatial data mining have been reported in the literature. GwiM [23] has been presented as a general-purpose ILP system that can solve several spatial data mining tasks, though no insight into the algorithmic issues has been provided. INGENS [18] is an inductive geographic information system with learning capabilities that currently support the classification task. This is performed by an embedded ILP system, named ATRE [17], which has been applied to the problem of topographic map interpretation.

In this paper, we focus our attention on the task of mining spatial association rules. This descriptive task aims at the detection of associations between *reference objects* and some *task-relevant objects*, the former being the main subject of the description while the latter being spatial objects that are relevant for the task at hand and spatially related to the former. For instance, we may be interested in describing a given area by finding associations among large towns (reference objects) and spatial objects in the road network, hydrography, and administrative boundaries layers (task-relevant objects). Some kind of taxonomic knowledge on task-relevant geographic layers may also be taken into account to get descriptions at different concept levels (*multiple-level association rules*). As usual in the problem setting of association rule mining, we search for associations with large support and high confidence (*strong rules*) like

is_a(A, large_town) , intersects(A,B) , adjacent_to(A,C) →
is_a(B,motorway), C\=B, is_a(C,sea) (36%, 80%)

"GIVEN THAT *36% of large towns intersect a motorway and are adjacent to the sea*, IF *a large town intersects a spatial object B and is adjacent to a spatial object C* THEN WITH CONFIDENCE *80% B is a motorway and C is the sea*".

The problem has already been tackled by Koperski and Han according to the AV approach [16]. They propose a top-down, progressive refinement method which exploits taxonomies both on spatial predicates and spatial objects.

In this paper, we present an ILP method for spatial association rule mining which can be considered the first-order counterpart of Koperski's method inspired by the work on mining association rules from multiple relations [5]. The method benefits from the available background knowledge on the spatial domain, systematically explores the hierarchical structure of task-relevant geographic layers and deals with numerical aspatial properties of spatial objects. It has been implemented into an ILP system, called SPADA (Spatial Pattern Discovery Algorithm), which operates on a DDB set up by an initial step of feature extraction from a spatial database.

The paper is organized as follows. Section 2 introduces the logical framework in our ILP approach to the task of mining spatial association rules. Section 3 is devoted to the presentation of the method with the help of illustrative examples taken from an application of SPADA to spatial data of the Province of Bari, Italy. Conclusions and future work are given in Section 4.

2 The logical framework

The problem of mining spatial association rules can be formally stated as follows:

Given

- a spatial database (SDB),
- a set of reference objects S,
- some task-relevant geographic layers R_k, $1 \le k \le m$, together with spatial hierarchies defined on them,
- two thresholds for each level l in the spatial hierarchies, *minsup*[l] and *minconf*[l]

Find strong multiple-level spatial association rules.

The basic idea in our ILP approach is that a *spatial database* can be boiled down to a DDB once that reference objects and task-relevant objects, their aspatial properties and the spatial relationships among them have been extracted according to a predefined semantics (*feature extraction*). As for topological relations, we have adopted the 9-intersection model [9]. Thus our approach requires that spatial data are transformed into ground facts of a logical language for relational databases. In particular, we resort to Datalog [4] whose expressive power allows us to specify background knowledge (BK) such as spatial hierarchies, spatial constraints and rules for spatial qualitative reasoning. Formal details follow.

From now on, we denote the DDB at hand $D(S)$ to mean that it is obtained by adding spatial relations extracted from SDB as concerns the set of reference objects S to the previously supplied *BK*. The tuples in $D(S)$ can be grouped into distinct subsets: Each group, uniquely identified by the corresponding reference object $s \in S$, is called *spatial observation* and denoted $O[s]$. It is given by

$$O[s] = O[s|s] \cup \{O[r_i|s] \mid \exists \text{ tuple } \theta \in D(S): \theta(s, r_i)\}_{1 \le i \le n}$$

where $O[s|s]$ contains spatial relations between s and some task-relevant object $r_i \in R_k$ and each $O[r_i|s]$ contains spatial relations between r_i and some $s' \in S$.

Example 1 Suppose the mining task is to discover associations relating large towns (S) with water bodies (R_1), roads (R_2) and province boundaries (R_3) in the Province of Bari, Italy. We are also given a BK including the spatial hierarchies of interest (see Figure 1 for a graphical representation of the layer of roads).

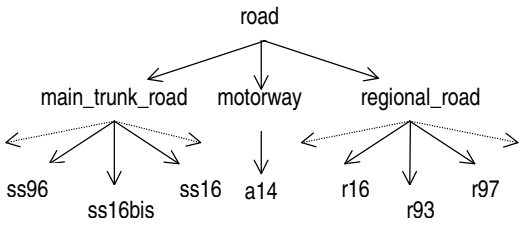

Fig. 1. A spatial hierarchy for the layer of roads

Is-a assertions are deducted by means of rules from BK. Here, the *is-a* relationship is overloaded, namely it may stand for *kind-of* as well as for *instance_of* depending on the context. Spatial relations between objects in S and objects in any of R_1, R_2 and R_3,

are extracted by means of spatial computation and transformed into facts of the kind <spatial relation>(RefObj, TaskRelevantObj) to be added to $D(S)$. Spatial observations are portions of $D(S)$, each concerning a reference object. In our case, there are eleven distinct spatial observations, one for each large town. For instance, O[barletta] is given by the union of the sets of ground facts listed in Table 1.

Table 1. The spatial observation O[barletta].

***O*[barletta | barletta]**	***O*[ss16 | barletta]**
is_a(barletta, large_town).	is_a(ss16, road).
adjacent_to(barletta, adriatico).	intersects(bari, ss16).
intersects(barletta, a14).	intersects(trani, ss16).
intersects(barletta, ss16).	intersects(monopoli, ss16).
intersects(barletta, ss16bis).	intersects(molfetta, ss16).
intersects(barletta, r170).	…
intersects(barletta, r193).	***O*[r170 | barletta]**
close_to(barletta, fg_boundary).	is_a(r170, road).
…	intersects(andria, r170).
***O*[adriatico | barletta]**	…
is_a(adriatico, water).	***O*[r193 | barletta]**
adjacent_to(bari, adriatico).	is_a(r193, road).
adjacent_to(trani, adriatico).	…
adjacent_to(molfetta, adriatico).	***O*[fg_boundary | barletta]**
adjacent_to(monopoli, adriatico).	is_a(fg_boundary, boundary).
…	adjacent_to(trani, fg_boundary).
***O*[a14 | barletta]**	…
is_a(a14, road).	***O*[ss16bis | barletta]**
intersects(bari, a14).	is_a(ss16bis, road).
intersects(trani, a14).	intersects(bari, ss16bis).
intersects(bitonto, a14).	intersects(trani, ss16bis).
intersects(gioia_del_colle, a14).	intersects(molfetta, ss16bis).
intersects(molfetta, a14).	…
…	

By definition, the observation encompasses not only spatial relations between the reference object barletta$\in S$ and task-relevant objects in R_1 (e.g. adriatico), R_2 (e.g. a14), R_3 (e.g. fg_boundary), but also spatial relations between each of these task-relevant objects and some other $s' \in S$ (e.g. bari) like in adjacent_to(bari, adriatico). •

Let $A=\{a_1, a_2, \ldots, a_l\}$ a set of Datalog atoms. Conjunctions of atoms on A are called *atomsets* [5]. In our framework, the language of patterns L is the set of well-formed atomsets generated on A. Necessary conditions for an atomset P to be in L are the presence of the key atom, the linkedness [13], and the safety. In particular, the last property guarantees the correct evaluation of patterns when they require the handling of negation (see Example 2). To a pattern P we assign an existentially quantified conjunctive formula $eqc(P)$ obtained by turning P into a Datalog query.

Definition A pattern P covers an observation $O[s]$ if $eqc(P)$ is true in $O[s] \cup BK$.

Example 2 The pattern

$P \equiv$ is_a(X, large_town), intersects(X,R), intersects(Y,R), Y\=X, is_a(R, road)

covers the spatial observation O[barletta] shown in Table 1 because the corresponding

$$\text{eqc}(P) \equiv \exists\ \text{is\_a}(X, \text{large\_town}) \wedge \text{intersects}(X,R) \wedge \text{intersects}(Y,R) \wedge Y \backslash= X \wedge \text{is\_a}(R, \text{road})$$

is satisfied by O[barletta]$\cup$BK. Here the predicate \= is the ISO Prolog Standard built-in predicate for non-unifiability of two variables. Note that it hides a negation. •

Definition Let O be the set of spatial observations in $D(S)$ and O_p denote the subset of O containing the spatial observations covered by the pattern P. The support of P is defined as $\sigma(P) = |O_p| / |O|$.

Definition A spatial association rule in $D(S)$ is an implication of the form

$$P \rightarrow R\ (s\%, c\%),$$

where $P \in L$, $R \subseteq A$, $P \cap R = \varnothing$, and at least one atom in $P \cup R$ represents a spatial relationship. The percentages s% and c% are respectively called the support and the confidence of the rule, meaning that s% of spatial observations in $D(S)$ are covered by $P \cup R$ and c% of spatial observations in $D(S)$ that are covered by P are also covered by $P \cup R$.

Definition The support and the confidence of a spatial association rule $P \rightarrow R$ are given by $s = \sigma(P \cup R)$ and $c = \varphi(R|P) = \sigma(P \cup R) / \sigma(P)$.

The frequency of a pattern depends on the level currently explored in the hierarchical structure of task-relevant geographic layers.

Definition Let *minsup*[l] and *minconf*[l] be two thresholds setting respectively the minimum support and the minimum confidence at level l in the spatial hierarchies. A pattern P is *large* (or frequent) at level l if $\sigma(P) \geq$ *minsup*[l] and all ancestors of P with respect to the taxonomies are large at their corresponding levels. The confidence of a spatial association rule $P \rightarrow R$ is high at level l if $\varphi(R|P) \geq$ *minconf*[l]. A spatial association rule $P \rightarrow R$ is strong at level l if $P \cup R$ is large and the confidence is high at level l.

3 The method

In the ILP method being proposed, the problem of mining spatial association rules is decomposed into two sub-problems: 1) Find large (or frequent) spatial patterns; 2) Generate strong spatial association rules. The reason for this decomposition is that frequent patterns are commonly not considered useful for presentation to the user as such. They are usually post-processed into rules that exceed given threshold values. In the case of association rules the threshold values of support and confidence offer a natural way of pruning weak and rare rules. It is noteworthy that SPADA can tackle applications which cannot be handled by either Geo-Associator [12] or WARMR [6]. In fact, as shown in Section 2, our system differs from the former for the expressive power of the representation language and from the latter for the semantics of patterns. In particular, although it has been presented as a system able to use is-a hierarchies, WARMR is *not* a system for mining *multiple-level* association rules, namely it does not exploit ontological information while searching and pruning the pattern space.

The main procedure of SPADA is reported in Figure 2. The algorithm for frequent pattern discovery is illustrated in the following Section while Section 3.2 gives an insight into the algorithm for generating strong rules from frequent patterns.

```
Procedure mineMultipleLevelAssociations (A, keyAtom, maxLevel, maxDepth)
FP: set of frequent patterns
IP: set of infrequent patterns
SR: set of strong rules
l: level in task-relevant spatial hierarchies (≤ maxLevel)
k: depth in the pattern space (≤ maxDepth)

begin
FP←∅; SR ← ∅; l←1;
        foreach level l do
        IP(l) ← ∅; k←1; FP(l, k)←{keyAtom}
        while k>maxDepth and FP(l, k) ≠ ∅ do
                k← k +1;
                [FP(l, k), IP(l)]← generateFrequentPatterns(FP(l, k-1), IP(l), A);
                SR(l, k) ← generateStrongRules(FP(l, k));
                FP(l) ← FP(l) ∪FP(l, k); SR(l) ← SR(l) ∪SR(l, k)
        endwhile
        FP ← FP∪FP(l); SR ← SR∪SR(l); l← l +1
endforeach
return [FP, RS]
```

Fig. 2. Main procedure of SPADA

3.1 From spatial data to frequent spatial patterns

The procedure *generateFrequentPatterns* implements the levelwise method [21], which is based on a breadth-first search in the lattice spanned by a generality order ≥ between patterns. The space is searched one level at a time, starting from the most general patterns and iterating between the candidate generation and candidate evaluation phases. In SPADA, the pattern space is structured according to θ-subsumption [22]. Many ILP systems adopt θ-subsumption as generality order for clause spaces. In this context we need to adapt the framework to the case of atomsets. To be more precise, the restriction of θ-subsumption to *Datalog queries* is of interest.

Definition Let Q_1 and Q_2 be two queries. Then Q_1 *θ-subsumes* Q_2 if and only if there exists a substitution θ such that $Q_1 \supseteq Q_2\theta$.

Example 3 Consider the queries

$$Q_1 \equiv \exists\ \text{is\_a(X, large\_town)} \wedge \text{intersects(X, R)} \wedge \text{is\_a(R, road)}$$
$$Q_2 \equiv \exists\ \text{is\_a(X, large\_town)} \wedge \text{intersects(X,Y)}$$
$$Q_3 \equiv \exists\ \text{is\_a(X, large\_town)}$$

We say that Q_1 θ-subsumes Q_2 and Q_2 θ-subsumes Q_3 with substitutions $\theta_1=\{Y \setminus R\}$ and $\theta_2=\varnothing$ respectively. •

We can now introduce the generality order adopted in SPADA.

Definition Let P_1 and P_2 be two patterns. Then P_1 is more general than P_2 under θ-subsumption, denoted as $P_1 \geq_\theta P_2$, if and only if P_2 θ-subsumes P_1.

Example 4 With reference to the queries reported in Example 3, we can say that true $\geq_\theta Q_3 \geq_\theta Q_2 \geq_\theta Q_1 \geq_\theta$ false where true and false are the bottom and the top of the $\geq_\theta$-ordered pattern space respectively. •

It is noteworthy that $\geq_\theta$ on patterns represented as Datalog queries is monotone with respect to support which is the criterion for candidate evaluation in SPADA. Furthermore, θ-subsumption is a *quasi-ordering* because it does not verify the anti-symmetric property. It follows that, given two queries such that $P_1 \geq_\theta P_2$ and $P_2 \geq_\theta P_1$, we can not conclude that P_1 and P_2 are equal up to renaming, i.e. P_1 and P_2 are not alphabetic variants.[1] As shown below, this feature has to be taken into account in search. A quasi-ordered set of patterns can be searched by a *refinement operator*, namely a function which computes a set of refinements of a pattern. In particular, we need a refinement operator under θ-subsumption that enables the bottom-up search of the pattern space from the most specific to the most general patterns.

Definition Let $<L, \geq_\theta>$ be a pattern space ordered according to $\geq_\theta$. An *upward refinement operator under θ-subsumption* is a function ρ such that $\rho(P) \subseteq \{Q \in L \mid P \geq_\theta Q\}$.

Such refinement operator drives the search towards patterns with decreasing support, therefore all refinements $\rho(P)$ of an infrequent pattern P are infrequent. This is the first-order counterpart of one of the properties holding in the family of the Apriori-like algorithms [2] on which the pruning criterion is based.

For each level (l) in the is-a taxonomies associated to the task-relevant geographic layers, SPADA generates and evaluates candidates by searching the pattern space. The *candidate generation* phase consists of a refinement step followed by a pruning step. The former applies the refinement operator under θ-subsumption to patterns previously found frequent by preserving the properties of linkedness and safety. The latter mainly involves verifying that candidate patterns do not θ-subsume any infrequent pattern. Further pruning criterions have been implemented in SPADA. In particular, the system checks that candidates are not alphabetic variants of previously discovered patterns. The complexity of this test is $O(n^2)$, where n is the number of atoms in the two patterns to be compared. The *candidate evaluation* phase is performed by verifying the largeness of the candidate pattern. If the pattern turns out not to be a large one, it is rejected. As for the support count, the candidate is transformed into an existential query whose answer set supplies all the substitutions that make the pattern true in $D(S)$. In particular, the number of different bindings for the variable which is the placeholder for reference objects is assumed as absolute frequency of the pattern in $D(S)$. It is noteworthy that SPADA "virtually" implements the notion of spatial observation, thus the property of linkedness guarantees the equivalence between the absolute frequency of a pattern and the number of observations covered by the pattern. The support is obtained as relative frequency of the pattern in $D(S)$.

As for the computational complexity, SPADA does not escape the notorious trade-off between expressiveness and efficiency in first-order representations. Studies on

[1] Let E and F be two expressions. Then E and F are *variants*, denoted E ≈ F, if and only if there exist substitutions θ and σ such that E=Fθ and F=Eσ. We also say that E is an *alphabetic variant* of F. For instance, f(X) and f(Y) are alphabetic variants.

learnability theory have suggested the use of *prior knowledge* and *declarative bias* to improve scalability [8, 25]. SPADA benefits from the available prior knowledge on the spatial domain and relies on a language bias specification to constrain the search for patterns. In particular, a refinement step consists of adding to the pattern to be refined one or more Datalog atoms in the language of patterns. Recently, the ILP setting of *learning from interpretations* has been proposed as a promising way of scaling up ILP algorithms in KDD applications [3]. The notion of spatial observation in SPADA, though only virtually supported in the current version of the system, adapts the notion of *interpretation* to the case of spatial databases.

Example 5 The system SPADA has been run on the mining task in Example 1 with thresholds *minsup*[1]=0.3 and *minconf*[1]=0.8 at level 1, *minsup*[2]=0.25 and *minconf*[2]=0.7 at level 2, and *minsup*[3]=0.2 and *minconf*[3]=0.6 at level 3. The specification of the language bias and other settings is given in input to the system together with *D*(large_town). The specification defines the alphabet *A* by listing all the "valid" atoms as facts like lb_atom(close_to(old ro, diff tro)). It also supplies syntactical rules for generating patterns in *L*. They serve as directives to the candidate generation phase (e.g. the atom in *A* to be used as key of spatial observations, modes of generating variables, conditions under which variables can be unified, etc.). The whole discovery process has taken 397.94 sec on a PC Pentium III with 128 Mb RAM (36.03 sec for level 1, 162.14 sec for level 2, and 199.77 sec for level 3). As for the frequent pattern discovery, it has returned 576 frequent patterns out of 8092 candidate patterns. Some interesting patterns have been discovered. For instance, at level l=2 in the spatial hierarchies, the following candidate *P*:

is_a(A, large_town), intersects(A,B), is_a(B, main_trunk_road), intersects(A,C), C\=B,
is_a(C, regional_road), intersects(D,C), D\=A, is_a(D,large_town)

has been generated after k=7 refinement steps and has been evaluated with respect to *D*(*S*) by means of the query:

?- is_a(A, large_town), intersects(A,B), is_a(B, main_trunk_road), intersects(A,C), C\=B,
is_a(C, regional_road), intersects(D,C), D\=A, is_a(D,large_town)

The answer set includes two substitutions, θ_1={A\barletta, B\ss16, C\r170, D\andria} and θ_2={A\barletta, B\ss16bis, C\r170, D\andria}. Therefore, the spatial observation *O*[barletta], reported in Table 1, is covered. However, while computing the support, the two substitutions count as only one because both refer to the same large town. Since six of eleven spatial observations are covered and all the ancestor patterns are large at their level ($l \leq 2$), the pattern is a large one at level l=2 with 54% support. For the sake of clarity, the following pattern

is_a(A, large_town), intersects(A,B), is_a(B,road), intersects(A,C), C\=B, is_a(C,road),
intersects(D,C), D\=A, is_a(D,large_town)

is one of the large ancestors for the pattern *P*. It has been generated after k=7 refinement steps at level l=1 and is supported by 82% large towns.

Such way of taking the taxonomies into account during the pattern discovery process implements what we refer to as the systematic exploration of the hierarchical structure of task-relevant geographic layers. Furthermore, it is noteworthy that the use of variables and the addition of atoms of the kind \= derived from the directive diff in

the language bias allow the algorithm to distinguish between multiple instances of the same class of spatial objects (e.g. the class large_town).

The results being presented in this paper are like snapshots of the Province of Bari and may give useful hints about the relevant features of a land area to GIS users. For instance, some knowledge about the spatial associations between large towns, water bodies and roads comes from the following frequent patterns at level l=2:

is_a(A,large_town), adjacent_to(A,B), is_a(B, sea), intersects(A,C), C\=B,
is_a(C, main_trunk_road) (45%)

"In 45% of cases, a large town is on the sea and intersects a main trunk road"

is_a(A,large_town), intersects(A,B), is_a(B,main_trunk_road), intersects(C,B), C\=A,
is_a(C,large_town), adjacent_to(C,D), D\=B, is_a(D, sea) (91%)

"In 91% of cases, a large town intersects a main trunk road which leads to a large town on the sea"

is_a(A,large_town), adjacent_to(A,B), is_a(B,sea), intersects(A,C), C\=B,
is_a(C,main_trunk_road), intersects(D,C), D\=A, is_a(D,large_town) (45%)

"In 45% of cases, a large town on the sea intersects a main trunk road that leads to another large town"

As for the novelty of the discovered knowledge, we believe that it strongly depends on the degree of knowledge of the territory under study. This is the case of GIS applications with large map repositories. •

3.2 From frequent spatial patterns to strong association rules

The procedure *generateStrongRules* implements the counterpart of the levelwise method for rules. The *candidate generation* phase consists of a combinatorial computation step followed by a pruning step. The former builds rules by putting together the left and right hand sides obtained as combinations of atoms occurring in a frequent pattern. The latter discards candidate rules whose left hand side is not well-formed because they cannot be evaluated. Indeed, the *candidate evaluation* phase is performed by verifying the strength of the candidate rules. Weak rules are rejected.

Example 5 As for the generation of spatial association rules in Example 4, the system has returned 27111 strong rules out of 51999 generated rules. The following strong rules have been derived from the frequent pattern P:

is_a(A,large_town), intersects(A,B), intersects(A,C), is_a(C,regional_road), intersects(D,C),
C\=B → is_a(B,main_trunk_road), D\=A , is_a(D,large_town) (54%, 75%)

"GIVEN THAT *54% of large towns intersect both a main trunk road and a regional road the latter intersecting a large town distinct from the previous one,* IF *a large town A intersects two spatial objects the former being an unknown B while the latter being a regional road which in turn intersects some spatial object D* THEN WITH CONFIDENCE 75% *B is a main trunk road and D is a large town distinct from A*"

is_a(A,large_town), intersects(A,B), intersects(A,C), is_a(C,regional_road), intersects(D,C),
D\=A, C\=B → is_a(B,main_trunk_road), is_a(D,large_town) (54%, 86%)

"GIVEN THAT *54% of large towns intersect both a main trunk road and a regional road the latter intersecting a large town distinct from the previous one,* IF *a large*

town A intersects two spatial objects the former being an unknown B while the latter being a regional road which in turn intersects some spatial object D distinct from A THEN WITH CONFIDENCE 86% *B is a main trunk road and D is a large town*".
They highlight that constraints of the kind \= impact the confidence count. Indeed, the atom D\=A in the left-hand side of the second rule reduces the number of spatial observations covered by the pattern.

The results also show the usefulness of spatial association rules for describing maps. For instance, the importance of the main trunk roads ss16 and ss16bis for the people who either live on the Adriatic sea or drive along the motorway a14 emerges from the following strong rules at level *l*=3:

is_a(A,large_town), intersects(A,B), is_a(B,ss16)
→ adjacent_to(A,C), is_a(C,adriatico), B\=C (45%, 100%)
is_a(A,large_town), intersects(A,B), is_a(B,ss16bis)
→ intersects(A,C), C\=B, is_a(C,ss16) (36%, 100%)
is_a(A,large_town), adjacent_to(A,B), is_a(B,adriatico)
→intersects(A,C), C\=B, is_a(C,ss16bis) (36%, 80%)
is_a(A,large_town), intersects(A,B), is_a(B,ss16bis)
→intersects(A,C), is_a(C,a14), B\=C (36%, 100%)

Some hints about the status of the road network in the neighborhood of the Provinces of Matera and Taranto are given by the following strong rules:

is_a(A,large_town), close_to(A,B), is_a(B,mt_boundary), intersects(A,C), intersects(D,C), is_a(D,large_town), D\=A , C\=B → is_a(C,regional_road) (27%, 100%)
is_a(A,large_town), close_to(A,B), is_a(B, ta_boundary), close_to(C,B), is_a(C,large_town), intersects(C,D), D\=B , C\=A →is_a(D,main_trunk_road) (27%, 100%)
that have been discovered at level *l*=2. •

4 Conclusions and future work

Knowledge discovery in spatial databases present a great challenge to the developers of multi-relational data mining methods. In this work we have proposed a special-purpose ILP method for spatial association rule mining which has been implemented in the ILP system SPADA. Illustrative examples taken from an application of SPADA to spatial data of the Province of Bari show that our method enables us to tackle applications that cannot be handled by either Geo-Associator or WARMR. Furthermore, deductive databases offer effective representation means for domain knowledge, constraints and qualitative reasoning. Of major interest is the possibility of embedding rules for the inference of implicit spatial relationships that are too numerous to be either stored in the spatial database or computed by computational geometry algorithms. Actually the use of relational vs. object-oriented vs. logic-based data models have been debated for a long time in the GIS world. Deductive object-oriented databases (DOOD) would be the best framework for reasoning on spatial data [1]. Extensions of ILP to DOOD are to be investigated.

We are currently working on the optimization of SPADA and its extension with unsupervised discretization techniques to deal with numerical attributes of spatial

objects. This will allow us to conduct experiments on large real-world data sets. Preliminary experimental results on geo-referenced census data can be found in [19]. For the future, we plan to improve the method by investigating proper measures of interestingness (e.g. pruning conditions for the rule space) and the issue of robustness (e.g. techniques for handling approximated spatial relations). On the application side, we intend to integrate SPADA into INGENS to supply GIS users with a powerful tool for the exploration of maps which can be considered preliminary and/or complementary to their interpretation.

The work presented in this paper is in partial fulfillment of the research objectives set by the IST project SPIN! (Spatial Mining for Data of Public Interest) funded by the European Union (`http://www.ccg.leeds.ac.uk/spin/`).

References

1. Abdelmoty, A.I., Paton, N.W., Williams, M.H., Fernandes, A.A.A., Barja, M., Dinn, A.: Geographic Data Handling in a Deductive Object-Oriented Database. In: Karagiannis, D. (ed.): Database and Expert Systems Applications. LNCS 856, Springer-Verlag, Berlin (1994) 445-454
2. Agrawal, R., Srikant, R.: Fast Algorithms for Mining Association Rules. In: Proc. of the Twentieth VLDB Conference, Santiago: Cile (1994).
3. Blockeel, H., De Raedt, L., Jacobs, N., Demoen, B.: Scaling Up Inductive Logic Programming by Learning from Interpretations. Data Mining and Knowledge Discovery 3(1) (1999) 59-93
4. Ceri, S., Gottlob, G., Tanca, L.: What you Always Wanted to Know About Datalog (And Never Dared to Ask). IEEE Transactions on Knowledge and Data Engineering 1(1) (1989) 146-166
5. Dehaspe, L., De Raedt, L.: Mining Association Rules in Multiple Relations. In: Lavrac, N., Dzeroski, S. (eds.): Inductive Logic Programming. LNCS 1297, Springer-Verlag, Berlin (1997) 125-132
6. Dehaspe, L., Toivonen, H.: Discovery of frequent Datalog patterns. Data Mining and Knowledge Discovery 3(1) (1999) 7-36
7. De Raedt, L., Dehaspe, L.: Clausal Discovery. Machine Learning 26 (1997) 99-146.
8. Dzeroski, S.: Inductive Logic Programming and Knowledge Discovery in Databases. In: Fayyad, U.M., Piatetsky-Shapiro, G., Smyth, P., Uthurusamy, R. (eds): Advances in Knowledge Discovery and Data Mining. AAAI Press/The MIT Press (1996) 117-152
9. Egenhofer, M.J., Herring, J.R.: Categorizing Binary Topological Relations Between Regions, Lines, and Points in Geographic Databases. In: Egenhofer, M.J., Mark, D.M., Herring, J.R. (eds): The 9-Intersection: Formalism and its Use for Natural-language Spatial Predicates. Technical Report 94-1, U.S. National Center for Geographic Information and Analysis (1994).
10. Ester, M., Kriegel, H.P., Sander, J.: Spatial Data Mining: A Database Approach. In Scholl, M., Voisard, A. (eds.): Advances in Spatial Databases. LNCS 1262, Springer-Verlag, Berlin (1997) 47-66
11. Flach, P.: From Extensional to Intensional Knowledge: Inductive Logic Programming Techniques and Their Application to Deductive Databases. In: B. Freitag et al. (eds.):

Transactions and Change in Logic Databases, LNCS 1472, Springer-Verlag, Berlin (1998) 356-387
12. Han, J., Koperski, K., Stefanovic, N.: GeoMiner: A System Prototype for Spatial Data Mining. In Peckham, J. (ed.): SIGMOD 1997, Proceedings of the ACM-SIGMOD International Conference on Management of Data. SIGMOD Record 26(2) (1997) 553-556.
13. Helft, N.: Inductive generalization: a logical framework. In: Bratko, I., Lavrac, N. (eds): Progress in Machine Learning. Sigma Press (1987) 149-157
14. Klösgen, W.: Challenges for Inductive Learning Approaches in Knowledge Discovery in Databases. AI*IA Notizie XIII(4) (2000) 17-25
15. Knobbe, A.J., Blockeel, H., Siebes, A.P.J.M., van der Wallen, D.M.G.: Multi-relational data mining. CWI, INS-R9908 (1999)
16. Koperski, K., Han, J.: Discovery of Spatial Association Rules in Geographic Information Databases. In: Egenhofer, M.J., Herring, J.R. (eds.): Advances in Spatial Databases, LNCS 951, Springer-Verlag, Berlin (1995) 47-66
17. Malerba, D., Esposito, F., Lisi, F.A.: Learning recursive theories with ATRE. In: Prade, H. (ed.): Proc. 13th European Conference on Artificial Intelligence, 435-439, John Wiley & Sons, Chichester, England. (1998)
18. Malerba, D., Esposito, F., Lanza, A., Lisi, F.A.: Discovering geographic knowledge: The INGENS system. In Ras, Z.W., Ohsuga, S. (Eds.): Foundations of Intelligent Systems, LNAI 1932, Springer-Verlag, Berlin (2000) 40-48
19. Malerba, D., Lisi, F.A.: Discovering Associations between Spatial Objects: An ILP Application. Accepted for presentation to the *11th International Conference on Inductive Logic Programming*, September 9-11, 2001, Strasbourg, France.
20. Mannila, H.: Inductive Databases and Condensed Representations for Data Mining. In: J. Maluszynski (ed.): Logic Programming. MIT Press (1997) 21-30.
21. Mannila, H., Toivonen, H.: Levelwise search and borders of theories in knowledge discovery. Data Mining and Knowledge Discovery 1(3) (1997) 259-289
22. Plotkin, G. 1970. A note on inductive generalization. Machine Intelligence 5: 153-163.
23. Popelinsky, L.: Knowledge Discovery in Spatial Data by means of ILP. In: Zytkow, J.M., Quafalou, M. (eds.): Principles of Data Mining and Knowledge Discovery. LNAI 1510, Springer-Verlag, Berlin (1998)185-193.
24. Roddick, J.F., Hornsby, K., Spiliopoulou, M.: An Updated Bibliography of Temporal, Spatial and Spatio-Temporal Data Mining Research. In Roddick, J.F., Hornsby, K. (eds.): Temporal, Spatial and Spatio-Temporal Data Mining. LNAI 2007, Springer-Verlag, Berlin (2001) 147-164
25. Weber, I.: A Declarative Language Bias for Levelwise Search of First-order Regularities. In Ras, Z.W., Skowron, A. (eds.): Foundations of Intelligent Systems. LNCS 1609, Springer-Verlag, Berlin (1999) 253-261