Intelligent Data Analysis 7 (2003) 541–566
IOS Press

Discovery of spatial association rules in geo-referenced census data: A relational mining approach

Annalisa Appice, Michelangelo Ceci, Antonietta Lanza, Francesca A. Lisi and Donato Malerba
Dipartimento di Informatica, Università degli Studi di Bari, via Orabona 4, 70126 Bari, Italy
E-mail: {appice, ceci, lanza, lisi, malerba}@di.uniba.it

Received 1 November 2002
Revised 10 January 2003
Accepted 25 February 2003

Abstract. Census data mining has great potential both in business development and in good public policy, but still must be solved in this field a number of research issues. In this paper, problems related to the geo-referenciation of census data are considered. In particular, the accommodation of the spatial dimension in census data mining is investigated for the task of discovering spatial association rules, that is, association rules involving spatial relations among (spatial) objects. The formulation of a new method based on a multi-relational data mining approach is proposed. It takes advantage of the representation and inference techniques developed in the field of Inductive Logic Programming (ILP). In particular, the expressive power of predicate logic is profitably used to represent both spatial relations and background knowledge, such as spatial hierarchies and rules for spatial qualitative reasoning. The logical notions of generality order and of the downward refinement operator on the space of patterns are profitably used to define both the search space and the search strategy. The proposed method has been implemented in the ILP system SPADA (Spatial Pattern Discovery Algorithm). SPADA has been interfaced both to a module for the extraction of spatial features from a spatial database and to a module for numerical attribute discretization. The three modules have been used in an application to urban accessibility of a hospital in Stockport, Greater Manchester. Results obtained through a spatial analysis of geo-referenced census data are illustrated.

1. Introduction

Most countries of the world conduct population and economic censuses at regular intervals. Population census information is of great value in planning public services for governments at all levels, such as cities, counties, provinces, and states. Both population and economic census data are also used by private companies or community organizations for various purposes, such as marketing studies, situating new factories or shopping malls, developing social service programs. Therefore, the application of data mining techniques to census data has great potential both in underpinning good public policy and in supporting business developments. However, mining census data is not straightforward and requires challenging methodological research.

In this work, we are mainly concerned with one of the research issues, namely the geo-referenciation of census data. The practice of attaching socio-economic data to specific locations has increasingly spread over the last few decades. In the UK, for instance, population census data are provided for each

enumeration district (ED), the smallest areal unit for which census data are published. At the same time, vectorized boundaries of the 1991 census EDs enable the investigation of socio-economic phenomena in association with the geographical location of EDs. These advances cause a growing demand for more powerful data analysis techniques that can link population data to their spatial, or, more precisely, geographical distribution.

Advances in spatial data structures [16], spatial reasoning [10], and computational geometry [38] have paved the way for the study of knowledge discovery in spatial data, and, more specifically, in geo-referenced data. *Spatial data mining* methods have been proposed for *the extraction of implicit knowledge, spatial relations, or other patterns not explicitly stored in spatial databases* [23].

Knowledge discovered from spatial data can be in various forms including classification rules, which describe the partition of the database into a given set of classes [22], clusters of spatial objects [35,39], patterns describing spatial trends, that is, regular changes of one or more non-spatial attributes when moving away from a given start object [15], and subgroup patterns, which identify subgroups of spatial objects with an unusual, an unexpected, or a deviating distribution of a target variable [20].

In this paper, we focus our attention on the specific task of discovering *spatial association rules*, that is, association rules involving spatial objects and relations. Association rules are a class of regularities introduced by Agrawal et al. [1] that can be expressed by an implication of the form:

$$P \rightarrow Q(s, c),$$

where P and Q are a set of literals, called *items*, such that $P \cap Q = \emptyset$, the parameter s, called *support*, estimates the probability $\mathrm{p}(P \cup Q)$, and the parameter c, called confidence, estimates the probability $\mathrm{p}(Q|P)$. We call an association rule $P \rightarrow Q$ *spatial*, if $P \cup Q$ is a *spatial pattern*, that is, it expresses a spatial relationship among spatial objects.

The problem of mining spatial association rules has already been tackled by Koperski and Han [21], who implemented the module Geo-associator of the spatial data mining system GeoMiner [18]. However, the method implemented in Geo-associator suffers from severe limitations due to the restrictive data representation formalism, known as *single-table assumption* [41]. More specifically, it is assumed that data to be mined are represented in a single table (or relation) of a relational database, such that each row (or tuple) represents an independent unit of the sample population and columns correspond to properties of units. In spatial data mining applications this assumption turns out to be a great limitation. Indeed, different geographical objects may have different properties, which can be properly modeled by as many data tables as the number of object types. In addition, attributes of the neighbors of some spatial object of interest may influence the object itself, hence the need for representing object interactions.

The recently promoted (multi-)*relational* approach to data mining [9] looks for patterns that involve multiple relations of a relational database. Thus data taken as input by these approaches typically consists of several tables and not just a single one, as is the case in most existing data mining approaches. Patterns found by these approaches are called *relational* and are typically stated in a more expressive language than patterns defined in a single data table. Typically, subsets of *first-order logic*, which is also called predicate calculus or relational logic, are used to express relational patterns.

Considering this strong link with logics, it is not surprising that many algorithms for multi-relational data mining originate from the field of *inductive logic programming* (ILP) [8,25,34,36]. Extending a single table data mining algorithm to a relational one is not trivial. Efficiency is also very important, as even testing a given relational pattern for validity is often computationally expensive. Moreover, for relational pattern languages, the number of possible patterns can be very large and it becomes necessary to limit their space by providing explicit constraints (*declarative bias*).

However, mining *spatial* association rules is a more complex task than mining *relational* association rules, whose solutions have already been reported in the literature [7]. Two further degrees of complexity are:

1. the implicit definition of spatial relations and
2. the granularity of the spatial objects.

The former is due to the fact that the location and the extension of spatial objects *implicitly* define spatial relations such as directional and topological relations. Therefore, complex data transformation processes are required to make spatial relations explicit, as in the application of machine learning techniques to topographic map interpretation in [28].

The latter refers to the fact that spatial objects can be described at multiple levels of granularity. For instance, if spatial objects are regions with some administrative autonomy, such as wards, districts and counties, they can be organized hierarchically as follows:

$$\text{Ward} \rightarrow \text{District} \rightarrow \text{County}$$

based upon the *inside* relationship between locations. Interesting association rules are more likely to be discovered at the lowest granularity level (ward) than at the county level. On the other hand, large support is more likely to exist at higher granularity levels (District and County) rather than at a low level.

In the next section, a new algorithm for mining spatial association rules is reported. The algorithm, named SPADA (Spatial Pattern Discovery Algorithm), is based on an ILP approach to relational data mining and permits the extraction of multi-level spatial association rules, that is, association rules involving spatial objects at different granularity levels. Details on both the interface of SPADA with a spatial database and the feature extraction process are reported in Section 3. Finally, the application of SPADA to a data mining task involving UK census data is reported in Section 4.

2. Mining spatial association rules

The discovery of spatial association rules is a descriptive mining task aiming to detect associations between *reference objects* and some *task-relevant objects*. The former are the main subject of the description, that is, the observation units, while the latter are spatial objects that are relevant for the task in hand and are spatially related to the former. For instance, if we are interested in investigating the socio-economic phenomenon of deprivation of some urban areas, we can look for spatial association rules that relate properties of some selected ED (reference objects) with properties of other spatial objects, such as public transport stops (task relevant objects). A spatial association rule which states that (only) "in 10% of reference EDs where the percentage of households with no car is high, there is a public transport stop" can lead to the conclusion that there are some seriously deprived areas in the examined territory. An indication of the gravity of this socio-economic phenomenon is given by the support of the rule: if the percentage is high (e.g., 80%), then there are many reference EDs with a high percentage of no-car-owning households which are not served by public transportation.

First-order logic is a useful tool for the formal representation of a spatial association rule. For instance, the association rule in the above example can be easily expressed as follows:

$$is\_a(X, ed)\ no\text{-}car\text{-}owning\text{-}households\%(X, high) \rightarrow$$
$$contains(X, Y)\ is\_a(Y, public\_transp\_stop) \quad (80\%, 10\%).$$

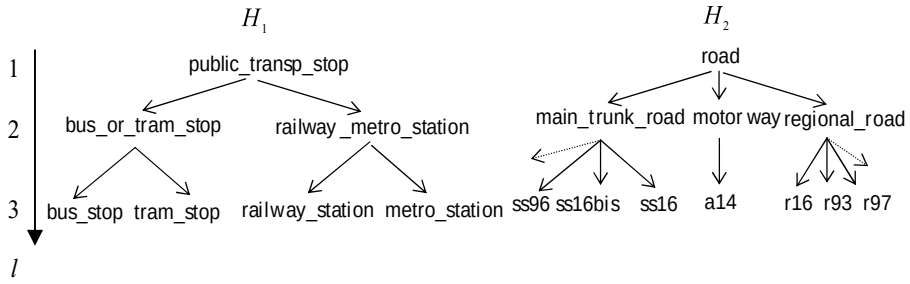

Fig. 1. Two spatial hierarchies and their association to three granularity levels (l).

This is a relational association rules, since the relational pattern

$$is\_a(X, ed)\ no\text{-}car\text{-}owning\text{-}households\%(X, high)\ contains(X, Y)$$
$$is\_a(Y, public\_transp\_stop)$$

expresses some form of relationship, namely spatial containment, between any reference ED and some spatial object classified as public transport stop. Since the relationship and the objects involved in this pattern are of a spatial nature, the above implication is also a *spatial* association rule.

In spatial association rules, the items are first-order logic *atoms*, that is, n*-ary predicates* applied to n *terms*. In the above example terms can be either *variables*, such as X and Y, or *constants*, such as *high* or *bus_ or_tram_stop*.

Since some kind of taxonomic knowledge on task-relevant objects may also be taken into account to obtain descriptions at different granularity levels (*multiple-level association rules*), finer-grained answers to the above query are also expected, such as:

$$is\_a(X, ed)\ no\text{-}car\text{-}owning\text{-}households\%(X, high) \rightarrow$$
$$contains(X, Y)\ is\_a(Y, bus\_stop)\quad (70\%, 11\%).$$

which provides more insight into the nature of the task relevant object Y, according to the spatial hierarchy reported in Fig. 1. It is noteworthy that the support and the confidence of the last rule have changed. Generally, the lower the granularity level, the lower the support of association rules. Therefore, we follow Han and Fu's [17] proposal to use different thresholds of support and confidence for different granularity levels.

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

- a spatial database (SDB),
- a set of reference objects S,
- some sets $R_k, 1 \leqslant k \leqslant m$, of task-relevant objects,
- a background knowledge BK including some spatial hierarchies H_k on objects in R_k,
- M granularity levels in the descriptions (1 is the highest while M is the lowest),
- a set of granularity assignments Ψ_k which associate each object in H_k with a granularity level,
- a couple of thresholds *minsup[l]* and *minconf[l]* for each granularity level,
- a declarative bias DB that constrains the search space,

Find strong multi-level spatial association rules.

An ILP approach to mining spatial association rules has already been reported in [31]. Representation problems and algorithmic issues related to the application of our logic-based computational method are discussed in the next two sub-sections.

2.1. Representing spatial data and background knowledge

The basic idea in our proposal is that a spatial database boils down to a deductive relational database (DDB) once the spatial relationships between reference objects and task-relevant objects have been extracted. More precisely, we pre-process data stored in a spatial database to represent it in a deductive database. For instance, spatial intersection between objects is represented in a derived relation *intersects*(*X*,*Y*).

As observed by Klösgen and May [20] this approach has some disadvantages, such as high computational burden, redundant data storage and loose integration between a GIS and the data mining method. However, there are at least two equally important advantages. First, once the data is pre-processed the calculation has not to be repeated, e.g. by constructing join indices. This is important in explorative data analysis, where the main effort is in analyzing data along several dimensions and according to a progressive refinement approach. The case in which an expert decides to throw away selected data and to consider a new data set occurs less frequently than the case in which the expert works on the same data by tuning the parameters of the data mining tool. Second, the expressive power of first-order logic in databases also allows us to specify a background knowledge BK, such as spatial hierarchies and a *domain specific knowledge* expressed as sets of rules. In particular, the specification of a domain specific knowledge permits the search for patterns which could not be otherwise found in the spatial database. The rules defining the domain specific knowledge are stored in the intensional part of the DDB and can support, amongst other things, qualitative spatial reasoning.

Henceforth, we denote the DDB in hand $D(S)$ to mean that it is obtained by adding the data extracted from SDB, regarding the set of reference objects S, to the previously supplied BK. The ground facts[1] 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 $O[s]$. We define the set:

$$R(s) = \{r_i | \exists k : r_i \in R_k \text{ and a ground fact } \alpha(s, r_i) \text{ exists in } D(S)\}$$

as the set of task-relevant objects spatially related to s. The set $O[s]$ is given by

$$O[s] = O[s|R(s)] \cup \mathbf{U}_{r_i \in R[s]} O[r_i|S],$$

where:

- $O[s|R(s)]$ contains properties of s and spatial relations between s and r_i
- $O[r_i|S]$ contains properties of r_i and spatial relations between r_i and some $s' \in S$.

In an extreme case, $O[s]$ can coincide with $D(S)$. This is the case in which s is spatially related to all task-relevant objects. The unique reference object associated to a spatial observation allows us to define the support and the confidence of a spatial association rule (see the definition of spatial association rule

[1] In this work we assume that ground facts concern either taxonomic "is_a" relationships or binary spatial relationships $\alpha(s, r)$ or object properties.

below). Note that the notion of spatial observation in SPADA adapts the notion of *interpretation*, which is common to many relational data mining systems [9], to the case of spatial databases.

Let $A = \{a_1, a_2, \ldots, a_t\}$ be a set of Datalog atoms whose terms are either variables or constants [4]. Predicate symbols used for A are all those permitted by the user-specified declarative bias, while the constants are only those defined in $D(S)$. The atom denoting the reference objects is called *key atom*. Conjunctions of atoms on A are called *atomsets* [6] like the itemsets in classical association rules. In our framework, a language of patterns $L[l]$ at the granularity level l is a set of well-formed atomsets generated on A. Necessary conditions for an atomset P to be in $L[l]$ are the presence of the key atom, the presence of taxonomic "is_a" atoms exclusively at the granularity level l, the linkedness [19], and safety. In particular, the last property guarantees the correct evaluation of patterns when the handling of negation is required. 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 $O[s] \cup BK$.*

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) at the granularity level l is an implication of the form*

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

where $P \cup Q \in L[l], P \cap Q = \emptyset$, P includes the key atom and at least one spatial relationship is in $P \cup Q$. The percentages s% and c% are respectively called the *support* and the *confidence* of the rule, meaning that s% of spatial observations in $D(S)$ is covered by $P \cup Q$ and c% of spatial observations in $D(S)$ that is covered by P is also covered by $P \cup Q$. The support and the confidence of a spatial association rule $P \rightarrow Q$ are given by $s = \sigma(P \cup Q)$ and $c = \varphi(Q|P) = \sigma(P \cup Q)/\sigma(P)$.

In multi-level association rule mining, an *ancestor* relation between two patterns at different granularity levels $P \in L[l]$ and $P' \in L[l']$, $l < l'$, exists if and only if P' can be obtained from P by replacing each spatial object $h \in H_k$ at granularity level $l = \Psi_k(h)$ with a spatial object $h' < h$ in H_k, which is associated with the granularity level $l' = \Psi_k(h')$.

The frequency of a pattern depends on the granularity level of task-relevant spatial objects.

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

The definition of the strong spatial association rule given above suggests that the generation of association rules at different granularity levels should proceed from the most general towards the most specific granularity levels. This is the approach followed in the ILP system SPADA, which has been developed for mining multi-level association rules in spatial databases. In the following section we explain how SPADA performs its search in the space of patterns at a given granularity level l, that is,

Find large 1-atomsets at level *l*
Cycle on the depth (k > 1) of search in the pattern space
1. Generate candidate *k*-atomsets at level *l* from large (*k*-1)-atomsets by applying the refinement operator ρ
2. Prune candidates that θ-subsume infrequent patterns
3. Prune candidates equivalent under θ-subsumption
4. Evaluate candidates and generate large *k*-atomsets at level *l* from candidate *k*-atomsets

Until the user-defined maximum depth

Fig. 2. Intra-level search implemented in SPADA.

in the space of patterns defined by the language $L[l]$ (*intra-level search*). Section 2.3 illustrates how SPADA takes advantage of statistics computed at a level l when it searches in the 'more specific' space at level $l+1$ (*inter-level search*).

2.2. Intra-level search of the pattern space

Given a granularity level l and a pattern language $L[l]$, the task of mining spatial association rules can be split into two sub-subtasks:

1. Find *large* (or *frequent*) spatial patterns in the space defined by $L[l]$;
2. Generate highly-confident spatial association rules at level l.

Algorithm design for frequent pattern discovery (step 1) has turned out to be a popular topic in data mining. The blueprint for most algorithms proposed in the literature is the levelwise method [32], which is based on a breadth-first search in the lattice spanned by a generality order $\geqslant$ between patterns. Given two patterns P_1 and P_2, we write $P_1 \geqslant P_2$ to denote that P_1 is more general than P_2 or equivalently that P_2 is more specific than P_1. 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. The intra-level search algorithm of SPADA implements the afore-mentioned levelwise method (see Fig. 2).

The pattern space is structured according to the θ-subsumption [37]. Many ILP systems adopt θ-subsumption as the generality order for clause spaces. In this context we need to adapt the framework to the case of atomsets. More precisely, the restriction of θ-subsumption to *Datalog queries* (i.e. existentially quantified conjunctions of Datalog atoms) is of particular 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 $Q_2\theta \subseteq Q_1$.*

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 $P_1 \geqslant P_2$, if and only if P_2 θ-subsumes P_1.*

θ-subsumption is a quasi-ordering, since it satisfies the reflexivity and transitivity property but not the anti-symmetric property. The quasi-ordered set spanned by $\geqslant_\theta$ can be searched by a *refinement operator*, namely a function that computes a set of refinements of a pattern.

Definition. *Let $\langle G, \geqslant_\theta \rangle$ be a pattern space ordered according to $\geqslant_\theta$. A downward refinement operator under θ-subsumption is a function ρ such that $\rho(P) \subseteq \{Q | P \geqslant_\theta Q\}$.*

In SPADA, the following operator ρ' is used.

Definition. *Let P be a pattern in $L[l]$. Then $\rho'(P) = \{P \wedge a_i | a_i$ is an atom in $L[l]\}$.*

It can be easily proven that $\rho'(P)$ is a downward refinement operator under θ-subsumption, that is $P \geqslant_\theta Q$ for all $Q \in \rho'(P)$. Indeed, $Q = P \wedge a_i$ for an atom a_i in $L[l]$. By adopting the set notation we can also write $Q = P \cup \{a_i\}$. The inequality $P \geqslant_\theta P \cup \{a_i\}$ holds if $P \cup \{a_i\}$ θ-subsumes P, that is, a substitution θ exists such that $P\theta \subseteq P \cup \{a_i\}$. Obviously, θ is the empty substitution. The refinement operator $\rho'(P)$ allows the generation of k-atomsets, that is atomsets of k literals, from (k-1)-atomsets.

It is noteworthy that $\geqslant_\theta$ on patterns represented as Datalog queries is monotone with respect to support.

Property of θ-subsumption monotony. *Let $\langle G, \geqslant_\theta \rangle$ be a pattern space ordered according to $\geqslant_\theta$. For any two patterns P_1 and P_2 such that $P_1 \geqslant_\theta P_2$ we have that $\sigma(P_1) \geqslant \sigma(P_2)$.*

Therefore, the refinement operator ρ drives the search towards patterns with decreasing support. If a pattern P is infrequent, all its refinements in $\rho'(P)$ are also infrequent. This is the first-order counterpart of one of the properties holding in the family of the Apriori-like algorithms [1], on which the pruning criterion is based. Indeed, the generation of patterns obtained as refinements of infrequent patterns can be avoided, since those patterns have certainly a support lower than the user-defined threshold. This is what happens at step 1) in the algorithm of Fig. 2.

Given a frequent pattern P of k-1 atoms, it may happen that some pattern Q $\in \rho'(P)$ θ-subsumes another infrequent pattern P' of k' atoms, with $k' < k$. This means that Q is certainly infrequent because of the above monotony property, and its evaluation can be avoided (step 2 in Fig. 2). Additional candidates not worth being evaluated are those equivalent under θ-subsumption to some other candidate (step 3 in Fig. 2).

Finally, unpruned candidates are evaluated to check whether they are large (i.e., frequent) or not (*candidate evaluation* phase, step 4). The evaluation of each generated pattern P requires a θ-subsumption test against some spatial observations $O[s]$. Indeed, if $O[s] \cup BK$ θ-subsumes P, then $eqc(P)$ is true in $O[s] \cup BK$, that is P covers $O[s]$, according to the definition given in the previous section. Actually, in SPADA the test of a pattern Q $\in \rho'(P)$ is performed only against those spatial observations covered by P, since, if a spatial observation $O[s]$ is not covered by P, it cannot be covered by Q without violating the transitive property of θ-subsumption.

2.3. Inter-level search of the pattern space

As specified in Section 2.1, to be able to define a pattern P as *large* (or *frequent*) at level l two conditions must be satisfied, namely

i) $\sigma(P) \geqslant minsup[l]$ and
ii) all ancestors of P with respect to the hierarchies H_k are large at their corresponding levels.

The second condition suggests an additional pruning strategy. Let P and Q be two frequent patterns at levels l and $l+1$ respectively, such that P is an ancestor of Q. Suppose that P has been refined into the infrequent pattern P' while searching in the pattern space at level l. When the space of patterns at level $l+1$ is explored and Q is refined, it is possible to generate a candidate pattern Q' whose ancestor is P'. In this case, Q' can be safely pruned, since it cannot be a large pattern without violating condition ii). In order to support this additional pruning strategy, the refinement operator implemented in SPADA

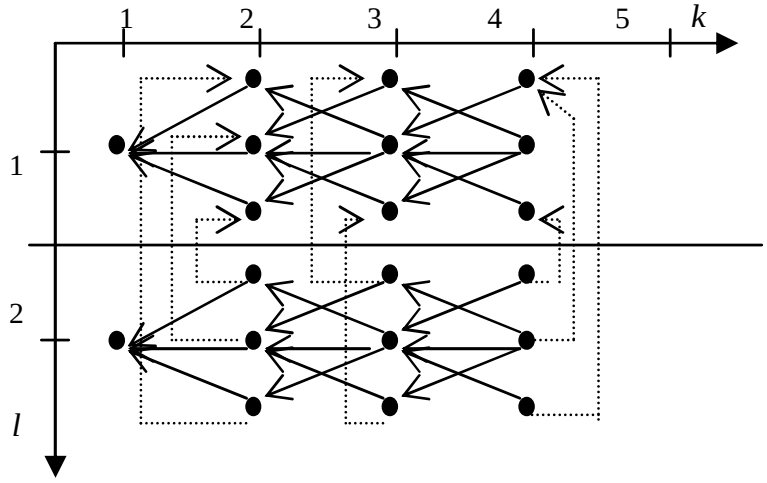

Fig. 3. Graph of intra-space and inter-space backward pointers.

uses a graph of backward pointers to be updated while searching. Backward pointers keep track of both intra-space and inter-space search stages. Figure 3 gives an example of such a graph, where nodes, dotted edges and dashed edges represent patterns, intra-space generality and inter-space parenthood, respectively. The effectiveness of this computational solution is illustrated in [26].

2.4. From patterns to association rules

Once large patterns have been generated, it is possible to generate strong spatial association rules. For each pattern P, SPADA generates antecedents suitable for rules being derived from P. The consequent corresponding to an antecedent is simply obtained as a complement of atoms in P and not in the antecedent. It is noteworthy that the generation of "good" rule antecedents is crucial. A naïve implementation would consist of a combinatorial computation step followed by a pruning step. The former would output combinations of atoms occurring in P, while the latter would discard those that are not well-formed, e.g. without the key atom in the antecedent or not respecting the constraints of linkedness and safety. Backward pointers can also be exploited to speed up the generation of association rules instead. In particular, SPADA recursively retrieves the predecessors of a frequent pattern and returns only those yielding strong rules. This eliminates the need for evaluating rules a posteriori.

2.5. Filtering patterns and association rules

In many applications, not all large patterns or strong association rules are deemed interesting by the user. The presentation of thousands of rules can discourage users from interpreting them in order to find 'nuggets' of knowledge. SPADA provides users with two filtering mechanisms, one for patterns and one for association rules. They are part of the declarative bias that constrains the search space. More precisely, the user can define the following pattern constraint:

$$\texttt{pattern\_constraint}(AtomList, Min\_occur),$$

where *AtomList* is a list of atoms (for atomic constraints) or a list of atom lists (for conjuctive constraints), while *Min_occur* is a positive number which specifies the minimum number of constraints in the list that must be satisfied. Patterns that do not satisfy this constraint are filtered out.

In addition, the user can define a constraint either on the antecedent or on the consequent of a spatial association rule, by specifying one of the following types of declarative bias:

$$\texttt{body\_constraint}(AtomList, Min\_occur)$$

$$\texttt{head\_constraint}(AtomList, Min\_occur),$$

where *AtomList* and *Min_occur* have the same meaning as in the pattern constraint. The application of this pattern/rule filtering approach proved very useful in the application to census data mining.

3. Interfacing SPADA to the spatial database

The application of the ILP approach to spatial databases is made possible by a middle-layer module for feature extraction. This layer is essential to cope with one of the main issues of spatial data mining, namely the requirement of complex data transformation processes to make spatial relations explicit. This function is partially supported by the spatial database (SDB), which offers spatial data types in its data model and query language and supports them in its implementation, providing at least spatial indexing and efficient algorithms for spatial join [16]. Thus spatial databases supply an adequate representation of both single objects and spatially related collections of objects. In particular, the abstraction primitives for spatial objects are point, line and region. Among the operations defined on spatial objects, spatial relationships are the most important because they make it possible, for example, to ask for all objects in a given spatial relationship with a query object.

3.1. Spatial features

Many spatial features (relations and attributes) can be extracted from spatial objects stored in SDB. According to their *nature*, features can be categorized as follows:

1. *locational* features, when they concern the location of objects;
2. *geometrical* features, when they are based on the principles of Euclidean geometry;
3. *directional* features, when they regard (relative) spatial orientation in 2D or 3D;
4. *topological* features, when they are relations preserving themselves under topological transformations such as translation, rotation, and scaling;
5. *hybrid* features, when they merge properties of two or more of the previous categories.

Locational features

Locational features are the simplest and more intrinsic features of spatial objects, since they are the attributes concerning the location. Depending on the abstraction primitives, meaningful locational features are: in the case of an areal object, the coordinates of the centroid, in the case of a linear object, the coordinates of the extremes and the coordinates of the point itself.

Geometrical features

Geometrical features are the most classical ones, since they are based on principles of Euclidean geometry. Generally, they depend on some distance/metric computations. Region area, perimeter, length of axes and shape properties are typical examples of attributive geometrical features, while distance and angle of incidence are typical examples of geometrical relations.

$$R(A,B) = \begin{pmatrix} A^0 \cap B^0 & A^0 \cap \partial B & A^0 \cap B^- \\ \partial A \cap B^0 & \partial A \cap \partial B & \partial A \cap B^- \\ A^- \cap B^0 & A^- \cap \partial B & A^- \cap B^- \end{pmatrix}$$

Fig. 4. The 9-intersection model represented as a matrix.

Directional features

Directional features concern information about relative spatial orientation in 2D or 3D. They are fundamental in image processing and are 'natural' relations to be considered when assuming as a reference system a regular gridding system. Directional features are defined on the basis of the *neighborhood* concept. The most common metrics adopted to define neighborhoods are the *city block* distance (d_4), and the *chess board* distance (d_8). Given two points, p and q, with co-ordinates (i, j) and (h, k), respectively, such distance functions are defined as follows:

$$d_4(p,q) = |i-h| + |j-k|$$
$$d_8(p,q) = \max\{|i-h|, |j-k|\}$$

The *neighbors* of a point p are the points having unitary distance from p. The neighborhood of a point p is constituted by all its neighbors and the point p itself. When the 4-distance is adopted, there are only 4 neighbors that are all adjacent to p along the main directions: north, east, south, and west. When the more general 8-distance is used, there are 8 neighbors which are adjacent to p along all the possible directions: north, north-east, east, south-east, south, south-west, west, and north-west.

Topological features

The 9-*intersection model* was proposed by Egenhofer and Franzosa [11] and Egenhofer and Herring [12] to categorize binary topological relations in geographic databases. It is independent of the concepts of distance and direction and is based upon purely topological properties. Currently, it can be considered the only comprehensive framework for qualitative spatial reasoning

The 9-intersection model applies to spatial objects represented by regions, lines, and points. It is based on the consideration that for each spatial object A it is possible to distinguish three parts: its interior (A^0), its boundary (∂A) and its exterior (A^-). In the case of spatial objects described by the Cartesian space R^2, regions have non-empty interiors, both lines and points have empty interiors, lines have non-empty boundaries (coincide with them), while points have empty boundaries.

Binary topological relations between two objects can be described in terms of part intersections. There are nine possible intersections between two parts, hence the name of the model. In Fig. 4 the 9-intersection model is concisely represented as a 3×3 matrix.

Different topological relations have different 9-intersections, each being intersection empty ($\emptyset$) or non-empty ($\neg\emptyset$). By considering the possible combinations of the two values $\emptyset$ and $\neg\emptyset$ one can distinguish $2^9 (= 512)$ binary topological relations. Not all the configurations correspond to physically feasible relations between two spatial objects. Indeed, there are restrictions on the combinations of values that could occur in the intersection matrix for the topological components of objects and only those combinations that comply with consistency constraints for topological object relations can occur.

Relations between two regions

In the bidimensional Cartesian space 8 relations can hold between two regions with connected boundaries. The existence of topological relations corresponding to the 9-intersections has been verified by

producing prototypical geometric configurations in R^2. The relations correspond to *disjoint*, *contains*, *inside*, *equal*, *meet*, *covers*, *covered by*, and *overlap*. This set is mutually exclusive and closed for regions, that is one and only one of the eight relations holds between any two regions.

Relations between two lines

Globally, there are 57 relations between two lines. In particular, 33 relations can be realized between simple lines, that is, lines composed of only one segment. Twenty-four other relations exist specifically for complex lines.

Relations between a region and a line

The topological relations between a region and a line involve two objects of different dimensions, therefore conditions that hold between a region and a line do not necessarily hold between a line and a region. There are 20 meaningful 9-intersections between a region and a line. One of them can be realized only if the line is a non-simple line. In Fig. 5 twelve feasible 9-intersections between a region and a line are reported. They are used later in the application to urban accessibility.

Relations between a point and a non-point

Since the boundary of a point is empty, it is irrelevant to analyze its three boundary intersections. Consequently, there are only 6 significant intersections to describe the topological relations between a non-point (region or line) B and a point A, and there are only 2^6 possible relations. Only 3 combinations of intersections between a point and a non-point are physically feasible for point-region and point-line configurations, respectively: They represent the three situations in which the point A is *external* to the region or to the line B, the point A is *internal* to the region or is on the line B, and, finally, the point A is *on the boundary* of the region B or *on one of the two extremes* of the line B.

Relations between two points

In the case of two points, we observe that both boundaries are empty. Then, there are only 4 relevant intersections: the intersections between interiors and exteriors. There are only 2 combinations of intersections for which the corresponding topological relations, *disjoint* and *equal*, can be realized between two points.

Hybrid features

Hybrid spatial features merge properties of two or more spatial feature categories. For instance, the features that express the conditions of *parallelism* and *perpendicularity* of two lines are both topological and geometrical. They are topological since they are invariant with respect to translation, rotation and stretching, while they are geometrical since their semantics is based on the angle of incidence. Another example of a hybrid spatial feature is represented by the relation of "*faraway-west*", whose semantics mixes both directional and geometric concepts. Moreover, we call hybrid those features that mix spatial relations with aspatial properties, such as the feature that describes coplanar roads by combining the condition of parallelism with information on the type of spatial objects (road).

3.2. Spatial features for census data mining

Three different kinds of spatial analysis can be performed on geo-referenced census data:

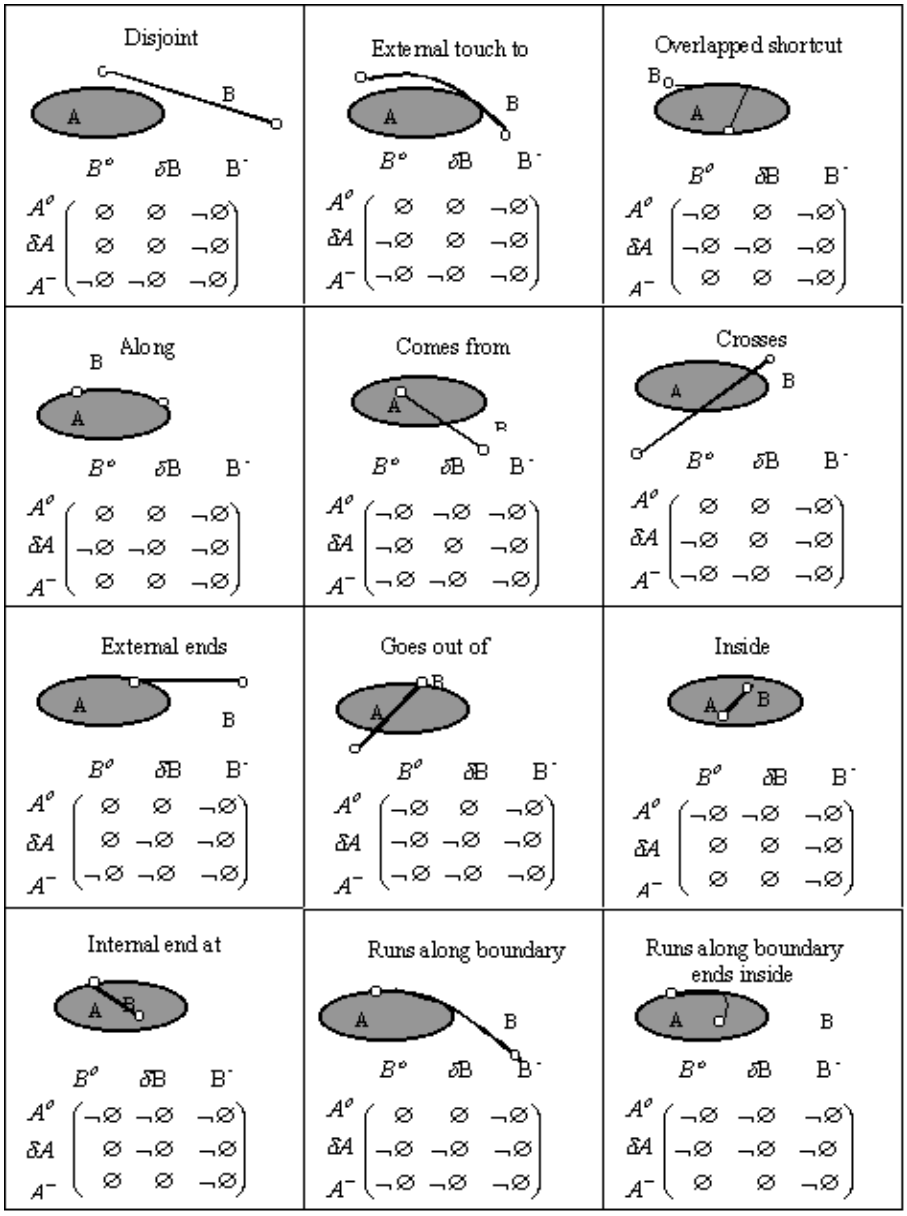

Fig. 5. Twelve feasible relations between a region and a line according to the 9-intersection model.

– Intra-ED analysis: spatial features concern geographic objects enclosed within the boundaries of an ED, while aspatial features are aggregated census data concerning a single ED. An example of intra-ED analysis is the characterisation of a site (residential, industrial area, and so on) for land allocation purposes.
– Inter-ED analysis: spatial features concern relations between EDs, while aspatial features are mainly extracted from census data for all EDs. An example of inter-ED analysis is the evaluation of the accessibility of a site for transport planning.

– Mixed intra-inter ED analysis: it is the most complex situation in which both spatial relations between geographic objects in an ED and spatial relations between EDs are required. An example of mixed analysis is the study of the impact that land allocations might have on the transportation system.

Consequently, the choice of spatial features that are *meaningful* for census data mining depends on the kind of analysis to be performed as well as on the specific task to be tackled.

For intra-ED analysis, man-made and natural features, ranging from houses and factories to roads and rivers, seem to be the kind of data to be considered. First, the relations "*contains*", "*inside*" and "*internal to*" can be used to define which different objects are present in a specific ED, since *contains* gives information on the existence of areal objects (such as the factories), *inside* gives information on linear objects (such as the roads), and *internal to* gives information on punctual objects (such as the bus stops). Then, the topological relations existing between the specific ED and its different objects and between the objects themselves might be used to obtain useful information. For example, if we are interested in discovering whether an ED is strongly urbanized, we might define a hybrid relation that checks the conditions of parallelism and perpendicularity between two roads, in order to detect a regular grid system of roads, which is typical of an urban area. Actually, similar relations have been defined and used for map interpretation tasks within planning contexts. In particular, they are described in Esposito and Lanza [13] for the application of machine learning techniques to technical charts of the Apulian region to identify the two land morphologies: cliffs and ravines. Descriptions of other features that have been adopted for the applications of machine learning to topographic maps can be found in Esposito et al. [14] and Malerba et al. [29]. These features were applied to localize some environmental categories important for the environmental protection (i.e., fluvial landscapes, royal cattle tracks, systems of cliffs and regular grid systems of farms) in the territory around the Ofanto river, Southern Italy.

For inter-ED analysis tasks, most of the spatial relations listed in the previous sub-section can be useful. In particular, topological relations between a region and a line might be used to investigate the accessibility of an ED from a road or a railway or a bus line. For example, if the *disjoint* relation (see Fig. 5) holds between a specific ED and a bus line of interest, we can conclude that the ED is not reachable directly by means of that bus line, so we have to search for other bus lines. Moreover, topological relations between two regions might be used to express spatial dependencies between EDs. In fact, only two cases might occur between two EDs: they are adjacent, i.e. the *meet* relation holds, or they are disjoint, i.e. the *disjoint* relation holds. In addition, the geometrical relation *distance* might be useful to evaluate the closeness of bus stops to public services (hospitals, railway stations, and so on). Finally, all the simple and hybrid features described in Section 3.2 might be useful for more complex mixed analysis tasks.

3.3. Extracting features with FEATEX

In order to generate features of spatial objects (points, lines, or regions) a feature extractor module, named FEATEX, has been devised. It enables a loose coupling of SPADA with the SDB. FEATEX is implemented as an Oracle package of procedures and functions, each of which implements a different feature. FEATEX functions can be used in SQL queries. For example, the query:

SELECT FEATEX.DIRECTION $(x.geom)$

FROM $river\_va\_polyline\ x$;

returns the geographic *direction* for each river in the table *river_va_polyline*. In this way, it is possible to formulate complex SQL queries involving both spatial and aspatial data (e.g., census data). The geographic direction of a spatial object is the only directional feature generated by means of FEATEX for the application. The corresponding FEATEX function returns the geographic direction ("north_east", "north_west", "east", "north") of the object identified by *geom* if the object is a line, "error" otherwise. Its specification is the following:

$$\text{FEATEX.DIRECTION}(geom)\ \text{RETURN VARCHAR2}.$$

Three hybrid features extractable by means of FEATEX are *almost-parallel*, *almost-perpendicular* and *density*. The first two concern the parallelism and perpendicularity conditions respectively between two spatial objects. The adverb 'almost' is justified by the fact that in cartography, as well as in nature, it is almost impossible to find two exactly parallel or perpendicular objects. For instance, two contour slopes are rarely exactly parallel, while two incidental roads do not always form a perfect angle of 90°. The feature concerning parallelism is extracted by means of the function:

$$\text{FEATEX.ALMOST\_PARALLEL}(geom1, geom2, tolerance)$$
$$\text{RETURN VARCHAR2}.$$

The first two parameters specify the geometries of the objects of interest. The *tolerance* parameter is used in distance computations. The function returns "true" if the condition holds, "false" otherwise.

The perpendicularity condition is extracted by means of the function:

$$\text{FEATEX.ALMOST\_PERPENDICULAR}(geom1, geom2, tolerance, angularTolerance)$$
$$\text{RETURN VARCHAR2}.$$

The definition of the function is based on the computation of the incidence angle between linear objects. The last parameter is the tolerance on the incidence angle.

The density is computed by means of the function:

$$\text{FEATEX.DENSITY}(geom1, dim1, geom2, dim2)\ \text{RETURN NUMBER},$$

where $dim1$ and $dim2$ are the dimensional information arrays corresponding to the first and second object. This function determines the relation between two objects according to the 9-intersection model, and then it computes their areas by means of the Gauss method. If the object identified by *geom1* is contained in the object identified by *geom2*, this function returns the ratio between the two areas, otherwise it returns zero.

Two geometrical features can be generated by FEATEX. One is the *distance* between two objects, which can be computed by means of the function:

$$\text{FEATEX.DISTANCE}(geom1, dim1, geom2, dim2)\ \text{RETURN NUMBER}.$$

It returns the distance between two objects identified by the geometries *geom1* and *geom2* and can be used even when objects have different geometries. The other geometrical feature extracted is *line_shape*. Its corresponding function is:

$$\text{FEATEX.LINE\_SHAPE}(geom, tolerance)\ \text{RETURN VARCHAR2}.$$

The *tolerance* parameter specifies the angular tolerance in radians for the computation. The function returns the values "straight" or "curvilinear", if the shape of the line identified by *geom* is linear or curvilinear, respectively. It returns "error" if the object identified by *geom* is not a line.

A detailed description of the algorithms underlying the computation of the above features can be found in [24], where an application to topographic map interpretation is also reported.

The topological features are computed by means of the following function:

$$\text{FEATEX.RELATE}(geom1, dim1, geom2, dim2) \text{ RETURN VARCHAR2.}$$

It analyses the geometries of the two objects of interest, which are identified by *geom1* and *geom2*, to determine the spatial relationship that holds between them, based on the 9-intersection model. The *relate* function returns the name of the topological binary relation (e.g. Disjoint, Along). In the case of topological relations involving lines, only those defined for simple lines are computed.

The result of a FEATEX function is an object-relational table. To transform it into a set of atoms, a function call is passed to the GENERATE function. The syntax of the GENERATE function is:

$$\text{FEATEX.GENERATE}(queryStr, predicateName, path, fileName, fileType)$$
$$\text{RETURN INTEGER}$$

The function executes a generic SQL query, given as a string parameter (*queryStr*) and outputs a text file where an atom is reported for each tuple. The predicate name used for the generated atoms is one of the arguments of the GENERATE function. More specifically, the result of a query:

$$\texttt{SELECT attr}_1, \texttt{attr}_2, \ldots, \texttt{attr}_n$$
$$\texttt{FROM table}_1, \texttt{table}_2, \texttt{table}_m$$
$$\texttt{WHERE condition},$$

where $attr_i (i = 1..n)$ can also be a FEATEX function, is translated into a set of the atoms:

$$predicateName(arg_1, \ldots, arg_n),$$

where $(arg_1, \ldots, arg_n)$ is a tuple in the query set.

The parameters *path* and *filename* in the GENERATE function identify the output file, while *filetype* specifies whether the output file is opened in append or write mode. The GENERATE function returns 1 if no errors occur, 0 otherwise.

3.4. Discretizing numerical attributes

Since SPADA, like many other association rule mining algorithms, cannot process numerical data properly, it is necessary to perform a discretization of numerical features with a relatively large domain. For this purpose we have implemented the relative unsupervised discretization algorithm RUDE [27], which discretizes an attribute of a relational database in the context defined by other attributes. Formally, the problem can be stated as follows:

Given

- a database table T consisting of m tuples,
- a continuous attribute in T to be discretized (*target attribute*),

– a set of continuous attributes (*source attributes*) in T that define the context for the discretization of the target attribute,
– a relative tolerance between split points (minimal difference) s

Find a set of split points that minimize loss of correlation between attributes.

The algorithm RUDE is based on two general procedures: a *prediscretization* procedure, used to pre-process the *source attributes*, and a *clustering* procedure, used to group target attribute values corresponding to some source attribute value or interval. Therefore, several different "specializations" of the RUDE algorithm can be generated by varying the two procedures. RUDE is implemented as a Java package. Two prediscretization algorithms are implemented, namely equal width and equal frequency, as well as two clustering algorithms, *EM* [40] and *AutoClass* [5]. RUDE proves to be suitable for dealing with numerical data in the context of association rule mining. An experimental study not reported in this paper showed that better performance can be obtained by using the equal width prediscretization procedure and the Autoclass algorithm. This combination has been used in the application to mining UK census data.

SPADA, FEATEX and RUDE have been integrated in the client-server system ARES, which is available at the URL: http://www.di.uniba.it/∼malezba/software/ARES/.

4. Mining UK census data: An application to urban accessibility

In this section we describe a practical example that shows how it is possible to perform a spatial analysis on UK 1991 census data. The application has been developed in the context of the European project SPIN! (Spatial Mining for Data of Public Interest) [33]. Census data concern Stockport, one of the ten districts in Greater Manchester, UK. In total 89 tables, each having 120 attributes on average, have been made available for policy analysis. Census attributes provide statistics on the population (resident at the census time, ethnic group, age, marital status, economic position, and so on), on the households in each ED (number of households with n children, number of households with n economically inactive people, number of households with two cars, and so on) as well as on some services available in each ED (e.g., number of schools).

Stockport is divided into twenty-two wards for a total of 589 EDs. Spatial analysis is enabled by the availability of vectorized boundaries of the 1991 census EDs as well as by other Ordnance Survey digital maps of the district, where several interesting layers are available, namely roads, bus priority lines, and so on. By joining UK 1991 census data available at the ED summarization level with some spatial objects (e.g., EDs, roads, and railways) it is possible to investigate socio-economic issues from a spatial viewpoint.

For the application of our spatial association rule mining method we have focused our attention on transportation planning, more specifically, on an important issue reported in the Unitary development Plan of Stockport, the accessibility of the Stepping Hill Hospital. The concept of "accessibility" appears initially in the context of geographical science and was progressively introduced in transport planning in the 1960's and 1970's. Many different definitions of accessibility and many ways to measure it can be found in the literature. In this work we are interested in urban accessibility, which refers to local (inner city) daily transport opportunities. A great effort has been made to define urban *accessibility indices*, which can be used to assess/compare transportation facilities within different regions of an urban area or between urban regions [2]. Accessibility is usually measured with respect to key activity locations for individuals (e.g., home, workplace) and evaluates the transportation services provided in these key

Fig. 6. Stockport map around Stepping-Hill Hospital. Spatial objects are the following: one-hundred and fifty-two task relevant EDs (white regions), five task-relevant EDs (yellow regions), the only bus priority line (thick light-blue line), crossing roads (blue lines), crossing railways (green lines).

locations to assess their relative advantages [3]. In this work, we are interested in the accessibility "to" the Stepping Hill Hospital "from" the actual residence of people living within in the area served by the hospital. Since (micro) data on the actual residence of each involved household are not available, we study the accessibility at the ED level. Moreover, our study does not aim to synthesize a new accessibility index, but to discover human interpretable patterns that can also contribute to directing resources for facility improvement in areas with poor transport accessibility.

We decided to mine association rules relating five EDs close to the Stepping Hill Hospital (task relevant objects) with one-hundred and fifty-two EDs within a distance of 10 Km of the hospital (reference objects). The goal is to understand which reference EDs have access to the task relevant EDs. To define the accessibility we used the Ordnance Survey data on transport network, namely the layers of roads, railways and bus priority lines (see Fig. 6).

By using FEATEX we extracted facts concerning two topological relationships between EDs and the only bus priority line reported in the spatial database for that area. Two examples are the following:

$$\texttt{external\_touches\_to}(\texttt{ed\_03bsfq29}, \texttt{bus\_priority\_line\_1}).$$

$$\texttt{comes\_from}(\texttt{ed\_03bsfl27}, \texttt{bus\_priority\_line\_1}).$$

The constants `ed_03bsfq29 and ed_03bsfl27` denote two distinct EDs, while `bus_priority_line_1` is the only constant associated to a bus priority line. The topological relations `external_touches_to` and `comes_from` are schematized in Fig. 5.

The set of topological relationships between EDs and roads is more varied. Some facts extracted by FEATEX are the following:

$$\texttt{along}(\texttt{ed\_03bsfk28}, \texttt{road\_15329}).$$

comes_from(ed_03bsfb23, road_12212).

crosses(ed_03bsfc13, road_12245).

external_ends_at(ed_03bsfc01, road_11501).

external_touches_to(ed_03bsfb22, road_15260).

external_comes_from(ed_03bsfc01, road_11502).

goes_out_of(ed_03bsfh01, road_10884).

inside(ed_03bsfc01, road_11494).

internal_ends_at(ed_03bsfc01, road_11500).

runs_along_boundary_ends_inside(ed_03bsfg22, road_10884).

runs_along_boundary(ed_03bsfc23, road_12312).

In this case constants *road_#* refer to roads crossing the interested area. It is noteworthy that the topological relations above are mutually exclusive, that is, it is impossible for two of them to concern the same pair of constants (ED, road).

Finally, we used FEATEX to extract spatial relationships between EDs and railways. Some examples of facts generated by the Oracle package are:

along(ed_03bsfg25, rail_2453).

comes_from(ed_03bsfc05, rail_2355).

crosses(ed_03bsfc13, rail_2391).

external_ends_at(ed_03bsfc05, rail_2389).

external_touches_to(ed_03bsfc20, rail_2389).

inside(ed_03bsfc01, rail_2341).

internal_ends_at(ed_03bsfc23, rail_2418).

overlapped_shortcut(ed_03bsfh12, rail_2487).

runs_along_boundary_ends_inside(ed_03bsfg11, rail_2429).

runs_along_boundary(ed_03bsfc10, rail_2391).

The total number of facts is 1,147. Despite the complexity of the spatial computation performed by FEATEX to extract these facts, the results are still not appropriate for the goals of our data analysis tasks. Indeed, we are interested in relationships between EDs, such as those stating that two EDs are 'connected' by the same bus priority line or the same road or the same railway. To solve this problem we specified the following rules to the domain specific knowledge:

1. `crossed_by_bus_line(X) :- external_touches_to(X,bus_priority_line_1).`
2. `crossed_by_bus_line(X) :- comes_from(X,bus_priority_line_1).`
3. `connected_by_bus_line(X,Y) :- crossed_by_bus_line(X), crossed_by_bus_line(Y), X≠Y.`
4. `crossed_by_road(X,Z) :- along(X,Z), is_a(Z,road).`

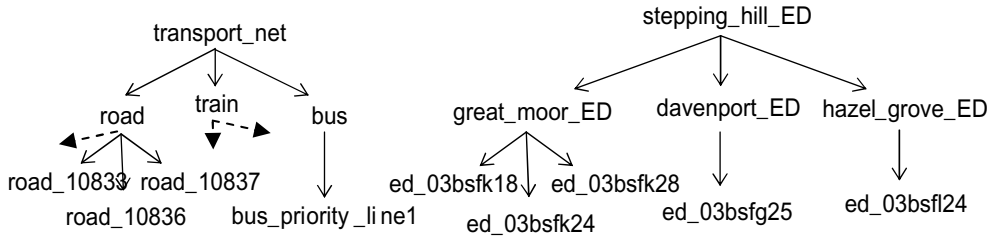

Fig. 7. Two spatial hierarchies defined for the mining task concerning the accessibility of the Stepping Hill Hospital. They are mapped into three granularity levels.

5. `crossed_by_road(X,Z) :- comes_from(X,Z), is_a(Z,road).`
6. `crossed_by_road(X,Z) :- crosses(X,Z), is_a(Z,road).`
7. `crossed_by_road(X,Z) :- external_ends_at(X,Z), is_a(Z,road).`
8. `crossed_by_road(X,Z) :- external_touches_to(X,Z), is_a(Z,road).`
9. `crossed_by_road(X,Z) :- goes_out_of(X,Z), is_a(Z,road).`
10. `crossed_by_road(X,Z) :- inside(X,Z), is_a(Z,road).`
11. `crossed_by_road(X,Z) :- internal_ends_at(X,Z), is_a(Z,road).`
12. `crossed_by_road(X,Z) :- runs_along_boundary_ends_inside(X,Z), is_a(Z, road).`
13. `crossed_by_road(X,Z) :- runs_along_boundary(X,Z), is_a(Z,road).`
14. `connected_by_road(X,Y) :- crossed_by_road(X,Z), crossed_by_road(Y,Z), X≠Y.`
15. `crossed_by_rail(X,Z) :- along(X,Z), is_a(Z,rail).`
16. `crossed_by_rail(X,Z) :- comes_from(X,Z),is_a(Z,rail).`
17. `crossed_by_rail(X,Z) :- crosses(X,Z),is_a(Z,rail).`
18. `crossed_by_rail(X,Z) :- external_ends_at(X,Z),is_a(Z,rail).`
19. `crossed_by_rail(X,Z) :- external_touches_to(X,Z),is_a(Z,rail).`
20. `crossed_by_rail(X,Z) :- inside(X,Z),is_a(Z,rail).`
21. `crossed_by_rail(X,Z) :- internal_ends_at(X,Z),is_a(Z,rail).`
22. `crossed_by_rail(X,Z) :- overlapped_shortcut(X,Z),is_a(Z,rail).`
23. `crossed_by_rail(X,Z) :- runs_along_boundary_ends_inside(X,Z),is_a(Z, rail).`
24. `crossed_by_rail(X,Z) :- runs_along_boundary(X,Z),is_a(Z,rail).`
25. `connected_by_rail(X,Y) :- crossed_by_rail(X,Z), crossed_by_rail(Y,Z), X≠Y.`

Here the use of the predicate is_a hides the fact that two hierarchies have been defined for spatial objects (see Fig. 7). Both hierarchies have depth three and are straightforwardly mapped into three granularity levels.

The "connection' predicates defined above express direct accessibility of an ED from another ED by means of only one road or railway or bus line. To express a more complex concept of accessibility, we added the following rules to the domain specific knowledge:

26. `can_reach_by_road(X,Y) :- connected_by_road(X,Y).`
27. `can_reach_by_road(X,Y) :- connected_by_road(X,Z), can_reach_by_road (Z,Y),X≠Y.`

28. `can_reach_by_rail(X,Y) :- connected_by_rail(X,Y).`
29. `can_reach_by_rail(X,Y) :- connected_by_rail(X,Z), can_reach_by_rail (Z,Y), X≠Y.`
30. `can_reach_by_bus(X,Y) :- connected_by_bus_line(X,Y).`
31. `can_reach_by_bus(X,Y) :- connected_by_bus_line(X,Z), can_reach_by_bus _line(Z,Y), X≠Y.`

These rules express a limited form of the transitivity property of 'connectedness'. Indeed, they state that an ED Y can be reached from another ED X if they are either directly connected by a road or a railway or a bus line, or if there is another "intermediate" ED Z, which is directly connected to both X and can be reached from Y (recursive definition).

To complete the domain specific knowledge, we added the following rules on the accessibility by means of public transport:

32. `can_reach_by_road_rail(X,Y) :- connected_by_road(X,Z), connected_by _rail(Z,Y), X≠Y.`
33. `can_reach_by_road_bus(X,Y) :- connected_by_road(X,Z), connected_by _bus_line(Z,Y), X≠Y.`
34. `can_reach_by_rail_bus(X,Y) :- connected_by_rail(X,Z), connected_by _bus_line(Z,Y), X≠Y.`
35. `can_reach_by_rail_bus(X,Y) :- connected_by_bus_line(X,Z), connected _by_rail(Z,Y), X≠Y.`
36. `can_reach_by_public_transport(X,Y) :- can_reach_by_bus(X,Y).`
37. `can_reach_by_public_transport(X,Y) :- can_reach_by_rail(X,Y).`
38. `can_reach_by_public_transport(X,Y) :- can_reach_by_rail_bus(X,Y).`
    and the complementary definition of accessibility by means of roads alone:[2]
39. `can_reach_only_by_road(X,Y) :- can_reach_by_road(X,Y), \+can_reach_by _public_transport(X,Y).`

Until now, census data have not been used to define the accessibility of the Stepping Hill Hospital. All extracted data and user-defined background knowledge are purely spatial. However, we can observe that the accessibility of an area cannot be defined on the basis of the transport network alone. Even though some roads connect a reference ED X with a task relevant ED Y, people living in X might have problems reaching Y because they do not drive. This means that sociological data available in the census data tables can be profitably used to give an improved definition of accessibility. We selected four attributes on the percentage of households with zero, one, two, and three or more cars, we discretized them with RUDE and generated the following four binary predicates for SPADA: *no_car, one_car, two_cars, three_more_cars*. The first argument of the predicate refers to an ED, while the second argument is an interval returned by RUDE.

To complete the problem statement we specified a declarative bias both to constrain the search space and to filter out some uninteresting spatial association rules. In particular, we asked for rules containing only the following predicates: *can_reach_by_public_transport*, *can_reach_only_by_road*, *no_car*, *one_car*, *two_cars*, and *three_more_cars*. In this way, we ruled out all spatial relations directly extracted by means of FEATEX and all intermediate spatial relations that helped to define the two interesting ones, namely

[2]We used Sicstus Prolog notation '$\backslash+$' to express the negation as failure.

the accessibility by public transport and the accessibility only by roads. Moreover, the specification of the following filter:

$$pattern\_constraint([no\_car(\_,\_), one\_car(\_,\_), two\_cars(\_,\_), three\_more\_cars(\_,\_)], 1).$$

prevents the generation of association rules with purely spatial patterns, that is, patterns showing only spatial relations between spatial objects. Purely spatial patterns are indeed of no interest to the expert in transport planning, since it is very likely that they convey no additional information to what he/she already knows.

After some tuning of the parameters *min_sup* and *min_conf* for each granularity level, we decided to run the system with the following parameter values:

$$\min\_\sup[1] = 0.2 \;\; \min\_\text{conf}[1] = 0.5$$
$$\min\_\sup[2] = 0.1 \;\; \min\_\text{conf}[2] = 0.4$$
$$\min\_\sup[3] = 0.1 \;\; \min\_\text{conf}[3] = 0.3$$

Despite the above constraints, SPADA generated 944 rules in 88 secs from a set of 39,830 extracted or inferred facts. More precisely, the system generated 28 rules in 38 secs at granularity level 1, 215 rules in 17 secs at level 2, and 701 rules in 33 secs at level 3. The output rules are stored in $M \times K$ XML files, where M is the number of granularity levels (3) and K is the maximum number of refinement steps (6). An additional index HTML file allows users to browse the output rules both by level and by refinement step (see Fig. 8).

Two of the rules returned by SPADA at the first level are the following:

$$ed\_around\_stepping\_hill(A), can\_reach\_only\_by\_road(A,B),$$
$$is\_a(B, stepping\_hill\_ED) \rightarrow no\_car(A, [0.228..0.653]) \quad (38.15\%, 56.31\%)$$
$$ed\_around\_stepping\_hill(A), can\_reach\_by\_public\_transport(A,B),$$
$$is\_a(B, stepping\_hill\_ED) \rightarrow no\_car(A, [0.266..0.653]) \quad (21.71\%, 61.11\%).$$

The spatial pattern of the first rule occurs in fifty-eight distinct EDs. This means that from fifty-eight distinct EDs within a distance of 10Km from Stepping Hill Hospital, it is possible to reach the hospital only by road and the percentage of households with no car is quite high (between 22.8% and 65.3%). Moreover, if from an ED A around Stepping Hill Hospital it is possible to reach one of the five task relevant EDs only by road, then the confidence that A has a high percentage of households with no car is 56.31%.

The spatial pattern of the second rule occurs in thirty-three distinct EDs. This means that from thirty-three reference EDs whose percentage of households with no car is quite high it is possible to reach the area of the Stepping Hill Hospital by public transport. The confidence in the second rule is a little higher than the first association rule.

At granularity level 2, SPADA specializes the task relevant object *stepping_hill_ED* considered at level 1. Only three specializations are possible for the five task relevant objects, namely *hazel_grove_ed, davenport_ed, great_moor_ed*, which correspond to three distinct wards (Hazel Grove, Davenport, Great Moor). The first rule above is specialized as follows:

$$ed\_around\_stepping\_hill(A), can\_reach\_only\_by\_road(A,B),$$

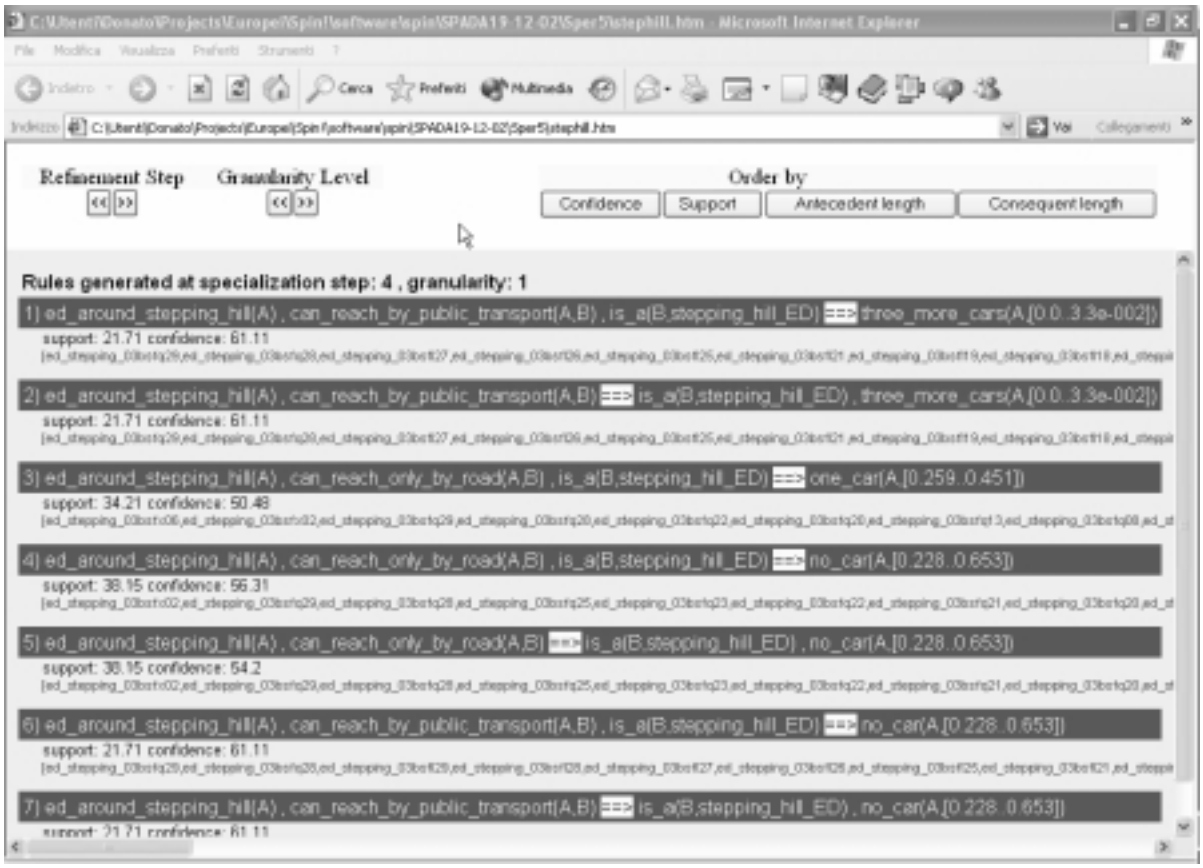

Fig. 8. Browsing results of the SPADA system. The user can select the refinement (or specialization) step and the granularity level. Rules are reported in the order in which they are generated, but they can be sorted in a decreasing order by confidence, support, antecedent length and consequence length. In addition to usual confidence and support values, the name of the reference EDs supporting the rule is shown. Results can be easily shown on a map.

$$is\_a(B, great\_moor\_ED) \rightarrow no\_car(A, [0.228..0.653]) \quad (38.15\%, 56.31\%)$$

$$ed\_around\_stepping\_hill(A), can\_reach\_only\_by\_road(A, B),$$

$$is\_a(B, davenport\_ED) \rightarrow no\_car(A, [0.228..0.653]) \quad (21.71\%, 50.76\%)$$

$$ed\_around\_stepping\_hill(A), can\_reach\_only\_by\_road(A, B),$$

$$is\_a(B, hazel\_grove\_ED) \rightarrow no\_car(A, [0.228..0.653]) \quad (21.71\%, 50.76\%).$$

As expected, the support of some rules have decreased. However, since both support and confidence are greater than the corresponding user-defined thresholds, all the three rules are output by SPADA. Similar considerations apply to granularity level 3, where specific task relevant EDs are reported.

Association rules found by SPADA in this application as well as in other related applications [30] are of interest to urban planners, since they relate data on the transport network with data on sociological factors. However, this study has three main limitations due to the nature of available data. First, we considered 1991 Census data, which are now obsolete. Second, the crossing of a railway does not necessarily mean that there is a station in an ED. Similar considerations can be made for bus priority

lines and roads. Third, digital maps made available by the Ordnance Survey are devised for cartographic reproduction purposes and not for data analysis. Hence, a road may appear to be 'blocked' in the digital map, because it runs under a bridge. A solution to these problems is planned for the near future.

5. Conclusions

In this paper, a multi-relational approach to the problem of mining spatial association rules has been illustrated in the context of an application to geo-referenced census data. This approach is justified by the need to consider relationships implicitly defined between spatial objects. Spatial relationships and spatial reasoning rules can be easily represented by means of first-order logic clauses, therefore, the definition of the spatial data mining method has naturally been based on several concepts developed in computational logics and, more specifically, in inductive logic programming. The interface to a spatial database is another crucial issue in spatial data mining. In this work, a form of loose coupling between the rule mining system SPADA and the SDB has been presented. It is based on the implementation of an Oracle package for the extraction of a number of spatial and aspatial features initially represented as tuples and then translated into atoms. Advantages and drawbacks of this approach have been briefly discussed in the paper. The future implementation of a tight coupling will permit an experimental comparison of the two solutions.

The specific census mining problem illustrated in this paper concerns the accessibility of an urban area. Unlike typical accessibility studies, no index has been developed in this application. Rather, we aimed at discovering human interpretable patterns that can also contribute to directing resources for facility improvement in areas with poor transport accessibility. Indeed, some of the discovered rules seem to convey new knowledge to urban planners, although the search for these "nuggets" requires a lot of tuning and effort on the part of the data analyst in order to constrain the search space properly and discard most of the obvious or totally useless patterns hidden in the data. One of the main limitations of our system, which is also a problem of many other relational data mining systems, is the requirement of some expertise in data and knowledge engineering. Indeed, the user should know how data are organized in the spatial database (e.g., layers and physical representation of objects), the semantics of spatial relations that can be extracted from digital maps, the meaning of some parameters used in the discretization process and in the generation of spatial association rules, as well as the correct and most efficient way to specify the domain knowledge and declarative bias. A solution to these usability problems will be investigated in the context of specific projects for which actual competences of end-users are known.

Acknowledgments

The authors thank Jim Petch, Keith Cole and Mohammed Islam (MIMAS, University of Manchester, England) and Chrissie Gibson (Department of Environmental and Geographical Sciences, Manchester Metropolitan University, England) for providing access to census data and digital OS maps of Stockport, Manchester. The work presented in this paper is in partial fulfillment of the research objectives set by the IST European project SPIN! (Spatial Mining for Data of Public Interest) and by the MURST COFIN-2001 project on "Methods for the extraction, validation and representation of statistical information in a decision context". Thanks to Lynn Rudd for her help in reading the paper.

References

[1] R. Agrawal, T. Imielinski and A. Swami, *Mining association rules between sets of items in large databases*, Proceedings of the ACM SIGMOD Conference on Management of Data, 1993, 207–216.

[2] C. Bhat, S. Handy, K. Kockelman, H. Mahmassani, Q. Chen and L. Weston, *Urban accessibility index: literature review. Technical Report No TX-01/7-4938-1*, Texas Dept. of Transportation, University of Texas at Austin, 2000

[3] L.D. Burns, *Transportation, Temporal, and Spatial Components of Accessibility*, Lexington, MA: Lexington Books, 1979

[4] S. Ceri, G. Gottlob and L. Tanca, 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] P. Cheeseman and J. Stutz, Bayesian Classification (AutoClass): Theory and Results, in: *Advances in Knowledge Discovery and Data Mining*, U. Fayyad, G. Piatesky-Shapiro, P. Smyth and R. Uthurusamy, eds, AAAI/MIT Press, 1996, pp. 153–180.

[6] L. Dehaspe and L. De Raedt, Mining Association Rules in Multiple Relations, in: *Inductive Logic Programming*, N. Lavrač and S. Džeroski, eds, LNCS 1297, Springer-Verlag, Berlin, 1997, pp. 125–132.

[7] L. Dehaspe and H. Toivonen, Discovery of frequent Datalog patterns, *Data Mining and Knowledge Discovery* **3**(1) (1999), 7–36.

[8] L. De Raedt, *Interactive Theory Revision*, Academic Press, London, 1992.

[9] S. Džeroski and N. Lavrač, eds, *Relational Data Mining*, Springer-Verlag, Berlin, 2001.

[10] M.J. Egenhofer, *Reasoning about Binary Topological Relations*, Proceedings of the Second Symposium on Large Spatial Databases, Zurich, Switzerland, 1991, 143–160.

[11] M.J. Egenhofer and R. Franzosa, Point-Set Topological Spatial Relations, *International Journal of Geographical Information Systems* **5**(2) (1991), 161–174.

[12] M.J. Egenhofer and J.R. Herring, Categorizing Binary Topological Relations Between Regions, Lines, and Points in Geographic Databases, in: *The 9-intersection: Formalism and its Use for Natural-language Spatial Predicates*, M.J. Egenhofer, D.M. Mark and J.R. Herring, eds, 1994, 183–271.

[13] F. Esposito and A. Lanza, The application of Machine Learning Techniques to Map Interpretation, in: *Soft Computing in Remote Sensing Data Analysis*, E. Binaghi, P.A. Brivio and A. Rampini, eds, World Scientific Pu. Co., Singapore, 1996, pp. 101–112.

[14] F. Esposito, A. Lanza, D. Malerba and G. Semeraro, Machine Learning for Map Interpretation: An Intelligent Tool for Environmental Planning, *Applied Artificial Intelligence: an International Journal* **11**(7–8) (1997), 673–696.

[15] M. Ester, A. Frommelt, H.-P. Kriegel and J. Sander, *Algorithms for Characterization and Trend Detection in Spatial Databases*, Proc. 4th Int. Conf. on Knowledge Discovery and Data Mining, New York City, NY, 1998, pp. 44–50.

[16] R.H. Güting, An introduction to spatial database systems, *VLDB Journal* **4**(3) (1994), 357–399.

[17] J. Han and Y. Fu, Discovery of multiple-level association rules from large databases, in: *VLDB'95, Proceedings of the 21st International Conference on Very Large Data Bases*, U. Dayal, P.M.D. Gray, S. Nishio eds, Morgan-Kaufmann, 1995, pp. 420–431.

[18] J. Han, K. Koperski and N. Stefanovic, GeoMiner: A System Prototype for Spatial Data Mining, in SIGMOD 1997, J. Peckham, ed., Proceedings of the ACM-SIGMOD International Conference on Management of Data, *SIGMOD Record* **26**(2) (1997), 553–556.

[19] N. Helft, Inductive generalization: a logical framework. in: *Progress in Machine Learning*, I. Bratko and N. Lavrač, eds, Sigma Press, 1987, 149–157.

[20] W. Klösgen and M. May, Spatial Subgroup Mining Integrated in an Object-Relational Spatial Database, in: *Principles of Data Mining and Knowledge Discovery* (*PKDD*), T. Elomaa, H. Mannila and H. Toivonen, eds, 6th European Conference, LNAI 2431, Springer-Verlag, Berlin, 2002, pp. 275–286.

[21] K. Koperski and J. Han, Discovery of Spatial Association Rules in Geographic Information Databases, in: *Advances in Spatial Databases. LNCS 951, Springer-Verlag*, M.J. Egenhofer and J.R. Herring, eds, Berlin, 1995, pp. 47–66.

[22] K. Koperski, J. Han and N. Stefanovic, *An Efficient Two-Step Method for Classification of Spatial Data*, Proc. Symposium on Spatial Data Handling (SDH '98), Vancouver, Canada, 1998, 45–54.

[23] K. Koperski, J. Adhikary and J. Han, *Spatial Data Mining: Progress and Challenges*, Proc. ACM SIGMOD Workshop on Research Issues on a Mining and Knowledge Discovery, Montreal, Canada, 1996.

[24] A. Lanza, D. Malerba, F.A. Lisi, A. Apice and M. Ceci, Generating Logic Descriptions for the Automated Interpretation of Topographic Maps, in: *Graphics Recognition: Algorithms and Applications*, D. Blostein and Y.-B. Kwon eds, LNCS 2390, Springer-Verlag Berlin Heidelberg, 2002, pp. 200–210.

[25] N. Lavrač and S. Džeroski, *Inductive Logic Programming: techniques and applications*, Ellis Horwood, Chichester, 1994.

[26] F. Lisi and D. Malerba, *Efficient Discovery of Multiple-level Patterns*, Atti del Decimo Convegno Nazionale su Sistemi Evoluti per Basi di Dati SEBD'02, 2002, pp. 237–250.

[27] M.-C. Ludl and G. Widmer, Relative Unsupervised Discretization for Association Rule Mining, in: *Principles of Data Mining and Knowledge Discovery*, D.A. Zighed, H.J. Komorowski and J.M. Zytkow, eds, LNCS 1910, Springer-Verlag, 2000, pp. 148–158.
[28] D. Malerba, F. Esposito, A. Lanza and F.A. Lisi, Machine learning for information extraction from topographic maps, in: *Geographic Data Mining and Knowledge Discovery*, H.J. Miller and J. Han, eds, Taylor and Francis, London, UK, 2001, pp. 291–314.
[29] D. Malerba, F. Esposito, A. Lanza, F.A. Lisi and A. Appice, Empowering a GIS with Inductive Learning Capabilities: The Case of INGENS, *Computers, Environment and Urban Systems* **27**(3) (2003), 265–281.
[30] D. Malerba, F. Esposito, F.A. Lisi and A. Appice, Mining Spatial Association Rules in Census Data, *Research in Official Statistics* **5**(1) (2002), 19–44.
[31] D. Malerba and F.A. Lisi, *An ILP method for spatial association rule mining*, Working notes of the First Workshop on Multi-Relational Data Mining, Freiburg, Germany, 2001, 18–29.
[32] H. Mannila and H. Toivonen, Levelwise search and borders of theories in knowledge discovery, *Data Mining and Knowledge Discovery* **1**(3) (1997), 259–289.
[33] M. May, Spatial Knowledge Discovery: The SPIN! System. Proceedings of the 6th EC-GIS Workshop, Lyon, ed., Fullerton, K., JRC, Ispra, 2000.
[34] S. Muggleton, ed., *Inductive Logic Programming*, Academic Press, London, 1992.
[35] R. Ng and J. Han, *Efficient and effective clustering method for spatial data mining*, Proceedings of the International Conference VLDB 1994, Santiago, Chile, September, 1994, 124–155.
[36] S.-H. Nienhuys-Cheng and R. deWolf, *Foundations of inductive logic programming*, Springer, Heidelberg, Germany, 1997.
[37] G. Plotkin, A note on inductive generalization, *Machine Intelligence* **5** (1970), 153–163.
[38] F. Preparata and M. Shamos, *Computational Geometry: An Introduction*, Springer-Verlag, New York, 1985
[39] J. Sander, M. Ester, H.-P. Kriegel and X. Xu, Density-Based Clustering in Spatial Databases: A New Algorithm and its Applications, *Data Mining and Knowledge Discovery, an International Journal, Kluwer Academic Publishers* **2**(2) (1998), 169–194.
[40] I.H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*, Morgan Kaufmann, October 1999.
[41] S. Wrobel, Inductive logic programming for knowledge discovery in databases, in: *Relational Data Mining*, S. Džeroski and N. Lavrač, eds, Springer: Berlin, 2001, pp. 74–101.