Relational Learning of Disjunctive Patterns in Spatial Networks

Corrado Loglisci, Michelangelo Ceci, and Donato Malerba

Department of Computer Science, University of Bari "A.Moro", Italy
{loglisci,ceci,malerba}@di.uniba.it

Abstract. In spatial domains, objects present high heterogeneity and are connected by several relationships to form complex networks. Mining spatial networks can provide information on both the objects and their interactions. In this work we propose a descriptive data mining approach to discover relational disjunctive patterns in spatial networks. Relational disjunctive patterns permit to represent spatial relationships that occur simultaneously with or alternatively to other relationships. Pruning of the search space is based on the anti-monotonicity property of support. The application to the problem of urban accessibility proves the viability of the proposal.

1 Introduction

A spatial network is a network of spatial objects, that is, objects characterized by both a spatial localization (e.g. in a geo-referenced system) and a geometry (e.g. an area). Nodes of spatial networks correspond to spatial objects, while links express spatial relationships (e.g. adjacency). In some cases, links are defined on the basis of other spatial objects (e.g. roads, railways, flights, rivers, etc.). A link might be labeled with a numerical weight which denotes the distance between two nodes. Although spatial networks are of great interest in the study of spatial phenomena, such as urban accessibility, they have not yet received the attention in data mining that they deserve. Yiu and Mamoulis [16] propose the extension of some traditional clustering techniques to face the problem of grouping objects in a large spatial network. In particular, the notion of shortest path between networked nodes is used in partitioning, density-based and hierarchical clustering methods. The same notion of shortest path is exploited in [8] for a problem of outlier detection in a dynamic network, where node insertion/deletion is allowed.

In these, as well as in other related works, a spatial network is modeled as a graph, which simplifies the network by removing the geometry. However, this representation is sometimes oversimplified, since it considers neither the heterogeneity of spatial objects (e.g. public services and private houses should be described by different feature sets) nor the heterogeneity of the spatial relationships expressed in links (e.g. connection by bus, railway or road). This heterogeneity of spatial objects and relationships demands for different representation formalisms and, consequently, a different class of data mining methods which are able to handle this further complexity in the data.

It has been recently argued that the (multi-)relational setting [6] is the most suitable for spatial data mining problems, since it can deal with the heterogeneity of spatial objects, it can distinguish their different role (reference or task-relevant), it can naturally represent a large variety of spatial relationships among objects and it can accommodate different forms of spatial autocorrelation [11]. Several spatial data mining methods have been developed according to the multi-relational setting. They concern descriptive and predictive tasks such as subgroup discovery[9], regression[12] and emerging patterns discovery[2].

In this paper, we extend our previous work on the task of spatial association analysis thorough an inductive logic programming (ILP) approach [1, 10]. Both spatial relationships and properties of spatial objects are represented as predicates, while discovered patterns are defined as conjunctions of atomic formulas built using these predicates. For instance, the following are two examples of spatial patterns which are discovered by the ILP system SPADA [10]:

$\langle district(A),\ road(B),\ intersects(B, A)\rangle$
$\langle district(A),\ road(B),\ crosses(B, A)\rangle$

where the semantics of the two predicates *intersects* and *crosses* is defined by means of the 9-intersection model defined for topological relationships [7]. The support of these patterns is computed by means of a θ-subsumption test[14] against the descriptions of the spatial networks. This is a crisp test which fails when, all other things being equal, two descriptions differ only in the name of a predicate. This brittleness is critical in spatial domains, where the computation of spatial relationships, though supported by a formal semantics, depends on the levels of abstraction granularity. For instance, for slightly different resolutions we may observe either an *intersects* relationship or a *crosses* relationship.

To improve the robustness of the spatial association rule mining method there are two alternatives. First, defining a hierarchy among spatial predicates, which could be used to generalize over spatial relationships. Second, enabling the generation of disjunctive patterns, that is patterns where two or more atoms may be OR-ed to express the variance on the spatial relationship existing between two objects. In this work we follow the second approach, since the definition of a hierarchy among spatial predicates can be cumbersome in many applications. Moreover, in order to prevent the generation of meaningless disjunctions, we exploit a user-defined dissimilarity measure between spatial relationships, which could be used to prune the search space.

The paper is organized as follows. In the next section, related works and contribution of the proposed approach are presented. In Section 3, the approach is presented in detail. In Section 4 the application to a case study is reported. Finally, conclusions are drawn and future works are presented.

2 Related Works and Contribution

Discovering patterns in spatial networks has attracted great interest in the area of in geographical information sciences (GIS). In his seminal work, Zhang [17] introduces a categorization of spatial patterns into grid-like, star-like and irreg-

ular categories of patterns and outlines the differences based on the parallelism relationship among the roads. Less attention has been rather paid in knowledge discovery, although the research in spatial data mining is become mature. A representative work is reported in [4] where the authors propose a framework which extracts snippets from Web, recognizes the locations and, finally, discovers patterns in the form of access points to the recognized locations.

The aforementioned works have common characteristic: spatial objects and networks are represented as vectors or graphs, which suffer from several limitations when heterogeneous spatial objects and relationships have to be represented. This motivates our interest in relational approaches to spatial pattern discovery. Moreover, to cope with the problem of brittleness of subsumption tests for relational patterns, we extend relational mining algorithms in order to discover disjunctive patterns, where alternative relationships between two spatial objects are allowed.

In the literature on frequent pattern mining, we found two noticeable contributions to the problem of discovering disjunctive patterns. In [13], association rules with inclusive or exclusive logical disjunction are discovered, while in [15] traditional algorithms are extended to mine association rules with item groups, where an item group is a disjunction of items created by considering the conceptual distance between items. Both methods work on propositional representations, which are too restrictive for spatial domains.

This paper makes a contribution to the literature on spatial network mining by considering disjunctive patterns in relational formalisms, which are more appropriate to represent spatial networks with heterogeneous spatial objects and relationships. In particular, we extend a method for spatial association rule discovery in order to represent:

1. Disjunctions (e.g. $(intersects(B, A)\ OR\ crosses(B, A))$). They are created by exploiting a user-defined background knowledge in the form of a semantic graph, where vertices correspond to spatial relationships (e.g. *intersects*), while edges denote the semantic relatedness among them and are labelled with numerical weights which quantify the dissimilarity among the relationships (e.g., $intersects \overset{0.9}{\leftrightarrow} crosses$);
2. Disjunctive patterns (e.g., $\langle district(A), transport\_line(B), (intersects(B, A)\ OR\ crosses(B, A)), is\_a(A, market\_square), is\_a(B, road)\rangle$). They are extracted from a graph of patterns which is refined until user-defined input criteria are met.

The proposed approach follows a three-stepped procedure. First, it extracts the infrequent conjunctive patterns which can be upgraded to the disjunctive form. For instance, given $P_1 : \langle district(A), contained\_in(A, B), marketplace(B)\rangle$ and the similarity between *contained_in* and *overlaps*, we can upgrade it to $P_1' : \langle district(A), (contained\_in(A, B)\ OR\ overlaps(A, B)),\ marketplace(B)\rangle$. Second, background knowledge is accommodated to exploit the information on the (dis)similarity among the spatial relationships in the process of generation of disjunctive patterns. Third, disjunctive patterns are produced by iteratively integrating disjunctions into the patterns by means of a pair-wise joining. For instance, given the patterns $P_1 : \langle district(A), contained\_in(A, B), marketplace(B)\rangle$,

$P_2 : \langle district(A), overlaps(A, B), marketplace(B)\rangle$ and assumed that *contained_in* and *overlaps* are two "similar" atoms according to background knowledge, P_1 and P_2 are merged to form the pattern:

$\langle district(A), (contained\_in(A, B)\ OR\ overlaps(A, B)), marketplace(B)\rangle$.

Finally, only the disjunctive patterns whose frequency exceeds the traditional minimum threshold are considered.

3 Learning Disjunctive Relational Patterns

Before formally stating the data mining problem, we introduce some basic notions. In the relational setting, when handling spatial objects, different roles can be played by different *sorts* of data. In a spatial network, objects can be distinguished into target objects of analysis (TO) and non-target objects of analysis (NTO). By introducing this distinction we follow the usual practice in statistics of distinguishing between units of analysis and units of observation. Generalization concerns the units of analysis, while the units of observation are typically secondary data considered potentially useful to explain a phenomenon.

In this work, the target objects (units of analysis) are data on which patterns are enumerated and contribute to compute the frequency of a pattern, while the non-target objects (units of observation) contribute to define the former and can be involved in a pattern. We denote the set of TO as S and the sets of NTO by means of the sets R_k $(1 \leq k \leq M)$, where M is the number of sorts of data that are not considered to be TO. NTOs, belonging to a set R_k, can be organized hierarchically according to a user defined taxonomy. Target objects and non-target objects are represented in Datalog language [3] as ground atoms and populate the extensional part D_E of a deductive database D. A ground atom is an n-ary logic predicate symbol applied to n constants.

Some predicate symbols are introduced in order to express both properties and relationships of TO and NTO. The predicate symbols represent spatial relationships and can be categorized into four classes: 1) *key predicate* identifies the TO in D_E (e.g., in the examples above, *district(·)*); 2) property predicates are binary predicates which define the values taken by an attribute of a TO or of an NTO; 3) structural predicates are binary predicates which relate NTO as well as TO with others NTO (e.g., in the examples above, *contained_in*$(\cdot,\cdot)$); 4) *is_a* predicate is a binary taxonomic predicate which associates NTO with a symbol contained in the user defined taxonomy.

The intensional part D_I of the deductive database D includes the definition of the semantic graph (background knowledge) that permits us to express the dissimilarity among spatial relationships in the form of *Datalog* weighted edges of a graph. An example of the Datalog weighted edge is the following:

external_touch_to - (crosses - 0.88)

It states that the dissimilarity between the relationships *external_touch_to(·,·)* and *crosses(·,·)* is 0.88. More generally, it represents an undirected edge e be-

tween two vertices v_i, v_j (e.g., *external touch to*, *crosses*) with weight w_{ij} (e.g., 0.88) and it is denoted as $e(v_i, v_j, w)$. A finite sequence of undirected links $e_1, e_2, \ldots, e_m$ which connects two vertices v_i, v_j is called *path* and denoted as $\rho(n_i, n_j)$. The complete list of such undirected edges represents the background information on the dissimilarity among relationships and allows to join patterns by introducing disjunctions *(externa_touch_to(A,B) OR crosses(A,B))*.

Discovered patterns are conjunctions of Datalog non-ground atoms and disjunctions of non-ground atoms, which can be expressed by means of a set notation. A Datalog non-ground atom is an n-ary predicate symbol applied to n terms (either constants or variables), at least one of which is a variable. A formal definition of pattern of our interest is reported in the following:

Definition 1. *A* disjunctive pattern P *is a set of atoms and disjunctions of atoms* $p_0(t_0^1)$, $(p_1(t_1^1, t_1^2)|p_2(t_2^1, t_2^2)|\ldots), \ldots, (p_k(t_k^1, t_k^2)|\ldots|p_{k+h}(t_{k+h}^1, t_{k+h}^2))$ *where* p_0 *is the key predicate, while* p_i, $i = 1, \ldots, k+h$, *is either a structural predicate or a property predicate or an* is_a *predicate. Symbol " | " indicates disjunctions.*

Terms t_i^j are either constants, which correspond to values of property predicates, or variables, which identify target objects or non-target objects. Each p_i is a predicate occurring in D_E (extensionally defined predicate). Some examples of disjunctive patterns are the following:

$P_1 \equiv district(A), (comes\_from(A, B)|external\_ends\_at(A, B)), shape(A, rectangle)$

$P_2 \equiv district(A), (external\_ends\_at(A, B)|runs\_along\_boundary\_and\_goes\_in(A, B)),$
$transport\_net(A, roads)$

where the variables A denote target objects, and variables B denote some non-target objects, while the predicates $district(A)$ identify the key predicate in P_1 and P_2, $shape(A, rectangle)$ and $transport\_net(A, roads)$ are property predicates and the others are structural predicates. All variables are implicitly existentially quantified.

We now can give a formal statement of the problem of discovering relational frequent patterns with disjunctions:

1. *Given:* the extensional part D_E of a deductive database D, and two thresholds $minSup \in [0; 1]$, $nSup \in [0; 1]$ ($minSup$ represents a minimum frequency value while $nSup$ represents a maximum frequency value, $nSup < minSup$), *Find:* the collection I_R of the relational infrequent patterns whose support is included in $[nSup; minSup)$.

2. *Given:* the collection I_R, the intensional part D_I of a deductive database D, and two thresholds $minSup$ and $\gamma \in [0; 1]$ (γ defines the maximum dissimilarity value of relationships involved in the disjunctions), *Find:* relational disjunctive patterns whose frequency exceeds $minSup$ and whose dissimilarity of relationships involved in the disjunctions does not exceed γ.

3.1 Discovering Infrequent Conjunctive Patterns

The intuition underlying the discovery of pattern with disjunctions is that of extending infrequent conjunctive patterns with disjunctive forms until the thresh-

old *minSup* is exceeded. Each conjunctive pattern P is associated with a statistical parameter $sup(P, D)$ (support of P on D), which is the percentage of *units of analysis* in D *covered by* P. More precisely, a unit of analysis of a target object $s \in S$ is a subset of ground atoms in D_E defined as follows:

$$D[s] = is\_a(R(s)) \cup D[s|R(s)] \cup \bigcup_{r_i \in R(s)} D[r_i|R(s)], \quad (1)$$

where $R(s)$ is the set of NTO directly or indirectly related to s, $is\_a(R(s))$ is the set of is_a atoms which define the sorts of $r_i \in R(s)$, $D[s|R(s)]$ contains properties of s and relations between s and some $r_i \in R(s)$, $D[r_i|R(s)]$ contains properties of r_i and relations between r_i and some $r_j \in R(s)$. By assigning a pattern P with an existentially quantified conjunctive formula $eqc(P)$ obtained by transforming P into a Datalog query, the units of analysis $D[s]$ are *covered by* a pattern P if $D[s] \models eqc(P)$, namely $D[s]$ logically entails $eqc(P)$).

Conjunctive patterns are mined with SPADA[10] which however enables the discovery of relational patterns whose support exceeds *minSup* (frequent patterns). In this work we exploit the capabilities of SPADA to identify infrequent conjunctive patterns, but this does not exclude the possibility of using other methods for mining infrequent relational patterns in this initial processing step. SPADA performs a breadth-first search of the space of patterns, from the most general to the more specific ones, and prunes portions of the space which contain only infrequent patterns, which are the conjunctive patterns of our interest. The pruning strategy guarantees that all infrequent patterns are removed and, at this aim, uses a generality ordering based on the notion of θ-subsumption [14]:

Definition 2. *P_1 is more general than P_2 under θ-subsumption ($P_1 \succeq_\theta P_2$) if and only if P_1 θ-subsumes P_2, i.e. a substitution θ exists, such that $P_1\theta \subseteq P_2$.*

For instance, given $P1 \equiv district(A), crosses(A, B)$, $P2 \equiv district(A), crosses(A, B)$, $is\_a(B, transport\_net)$, $P3 \equiv district(A), crosses(A, B), is\_a(B, transport\_net), along(A, C)$ we observe that P_1 θ-subsumes P_2 ($P_1 \succeq_\theta P_2$) and P_2 θ-subsumes P_3 ($P_2 \succeq_\theta P_3$) with substitutions $\theta_1 = \theta_2 = \oslash$. The generality order is monotonic with respect to the pattern support, so whenever $P1$ will be infrequent the patterns more specific of it (e.g., $P2, P3$) will be infrequent too.

The search is based on the level-wise method and implements a two-stepped procedure: i) generation of candidate patterns with k atoms (k-th level) by considering the frequent patterns with $k-1$ atoms (*(k-1)*-th level); ii) evaluation of the frequency with k atoms. So, the patterns whose support does not exceeds *minSup* will be not considered for the next level: the patterns discarded (infrequent) at each level are rather considered for the generation of disjunctions. The collection I_R is thus composed of a subset of infrequent patterns, more precisely those with support greater than or equal to *nSup* (and less than *minSup*).

3.2 Upgrading Relational Patterns with Disjunctions

The generation of disjunctive patterns is performed by creating disjunctions among similar relationships (thus similar atoms in the patterns) in accordance

to the background semantic graph: two patterns which present similar atoms are joined to form only one. The implemented algorithm (see Algorithm 1) is composed of two sub-procedures: the first one (lines 2-12) creates a graph $\mathcal{G}_\mathcal{D}$ with the patterns of I_R by exploiting the knowledge defined in D_I, while the second one (lines 13-32) joins two patterns (vertices) on the basis of the information (weight) associated to their edge. The initial graph $\mathcal{G}_\mathcal{D}$ evolves due to joining of patterns on the vertices until the setting of $minSup$ and γ is met (Section 3.1).

In particular, for each pair of patterns which have the same length (namely, at the same level of the level-wise search method) it checks whether they differ in only one atom and share the remaining atoms up to a redenomination of variables (line 3). Let α and β be the two atoms differentiating P from Q (α in P, β in Q), a path ρ which connects α to β (or viceversa) is searched among the weighted edges according to D_I (semantic network): in the case the sum ω of the weights found in the path is lower than the maximum dissimilarity γ the vertices P and Q are inserted into $\mathcal{G}_\mathcal{D}$ and linked through an edge with weight ω (lines 4-9). Note that when there is more than one path between α and β, then the path with lowest weight is considered. Intuitively, at the end of the first sub-procedure, $\mathcal{G}_\mathcal{D}$ will contain, as vertices, the patterns which meet the condition at the line 3, and it will contain, as edges, the weights associated to the path linking the atoms differentiating the patterns.

Once we have $\mathcal{G}_\mathcal{D}$, a list $\mathcal{L}_\mathcal{D}$ is populated with the vertices and edges of $\mathcal{G}_\mathcal{D}$: an element of $\mathcal{L}_\mathcal{D}$ is a triple $\langle P, Q, \omega \rangle$ composed of a pair of vertices-patterns (P,Q) with their relative weight. Elements in $\mathcal{L}_\mathcal{D}$ are ranked in ascending order with respect to the values of ω so that the pairs of patterns with lower dissimilarity will be joined for first. This guarantees that disjunctions with very similar atoms will be preferred to the others (line 13). For each element of $\mathcal{L}_\mathcal{D}$ whose weight ω is lower than γ the two patterns P, Q are joined to generate a pattern J composed by the conjunction of the same atoms in common to the two patterns P, Q and of the disjunction formed by the two different (but similar) atoms (lines 14-15). This joining procedure permits to have patterns with the same length of the original ones and which occur when at least one of original patterns occurs. Therefore, if a pattern J is obtained by joining P and Q, it covers a set of units of analysis equal to the union of those of P and Q: the support of J is determined as in line 16 and, generally, it is higher than the support of P and Q. In the case the support of J exceeds $minSup$ then it can be considered statistically interesting and no further processing is necessary (lines 16-17). Otherwise, J is again considered and inserted into $\mathcal{G}_\mathcal{D}$ as follows. The edges which linked another pattern R of $\mathcal{G}_\mathcal{D}$ to P and Q are modified in order to keep the links from R to J: the weight of the edges between one pattern R and J will be set to the average value of the weights of all the edges which linked R to P and Q (lines 19-27). The modified graph $\mathcal{G}_\mathcal{D}$ contains conjunctive patterns (those of I_R) and pattern with disjunctions (those produced by joining). Thus, $\mathcal{G}_\mathcal{D}$ is re-evaluated for further joins and the algorithm proceeds iteratively (line 29-30) until no additional disjunctions can be done (namely, when $\mathcal{L}_\mathcal{D}$ is empty or the weights ω are higher than γ). At each iteration, the patterns P and Q are removed from $\mathcal{G}_\mathcal{D}$ (line 32).

Algorithm 1 Upgrading Relational Pattern with Disjunctions.

input: $I_R, D_I, \gamma, minSup$ **output:** $\mathcal{J}$ // $\mathcal{J}$ set of disjunctive patterns
for all $(P, Q) \in I_R \times I_R$, $Q \neq P$ **do**
  if $P.length = Q.length$ and $check\_atoms(P, Q)$ **then**
    $(\alpha, \beta) := atoms\_diff(P, Q)$ //α, β atoms differentiating P,Q
    if $\rho(\alpha, \beta) \neq \oslash$ **then**
      $\omega := \sum_{e(v_i, v_j, w_{ij}) \ in \ \rho(\alpha,\beta)} w_{ij}$
      if $\omega \leq \gamma$ **then**
        $addNode(P, \mathcal{G}_\mathcal{D})$; $addNode(Q, \mathcal{G}_\mathcal{D})$; $addEdge(P, Q, \omega, \mathcal{G}_\mathcal{D})$
      end if
    end if
  end if
end for
$\mathcal{L}_\mathcal{D} \leftarrow$ edges of $\mathcal{G}_\mathcal{D}$ // list of edges of $\mathcal{G}_\mathcal{D}$ ordered in ascending mode w.r.t. ω
while $\mathcal{L}_\mathcal{D} \neq \oslash$ and $\forall e(P, Q, \omega) \in \mathcal{G}_\mathcal{D}$ $\omega \leq \gamma$ **do**
  $J \leftarrow join(P, Q)$; $J.support := P.support + Q.support - (P \cap Q).support$;
  if $J.support \geq minSup$ **then**
    $\mathcal{J} := \mathcal{J} \cup \{J\}$
  else
    for all R *such that* $\exists\ e(P, R, \omega_1) \in \mathcal{G}_\mathcal{D}$ and $\exists\ e(Q, R, \omega_2) \in \mathcal{G}_\mathcal{D}$ **do**
      $addEdge(R, J, (\omega_1 + \omega_2)/2, \mathcal{G}_\mathcal{D})$
    end for
    for all R *such that* $\exists\ e(P, R, \omega_1) \in \mathcal{G}_\mathcal{D}$ and $\nexists\ e(Q, R, \omega_2) \in \mathcal{G}_\mathcal{D}$ **do**
      $addEdge(R, J, \omega_1, \mathcal{G}_\mathcal{D})$
    end for
    for all R *such that* $\exists\ e(Q, R, \omega_2) \in \mathcal{G}_\mathcal{D}$ and $\nexists\ e(P, R, \omega_1) \in \mathcal{G}_\mathcal{D}$ **do**
      $addEdge(R, J, \omega_2, \mathcal{G}_\mathcal{D})$
    end for
    $\mathcal{L}_\mathcal{D} \leftarrow$ edges of $\mathcal{G}_\mathcal{D}$
    update $\mathcal{L}_\mathcal{D}$
  end if
  $removeNode(P, \mathcal{G}_\mathcal{D})$; $removeNode(Q, \mathcal{G}_\mathcal{D})$
end while

An explanatory example is illustrated in Figure 1. Consider the background knowledge D_I on the dissimilarity among four spatial relationships and the set I_R containing four infrequent conjunctive patterns as illustrated in Figure 1a and γ equal to 0.7. The first sub-procedure of the algorithm 1 analyzes P_1, P_2, P_3, P_4 and discovers that they differ in only one atom, while the other atoms are in common. Then, it creates the graph $\mathcal{G}_\mathcal{D}$ by collocating P_1, P_2, P_3 in three different vertices and linking them through edges whose weights are taken from the paths ρ in D_I. P_4 is not considered because the vertex *overlaps* has dissimilarity with *internal_ends_at* higher than γ (row (1) in Figure 1b). The second sub-procedure starts by ordering the weights of the edges: the first disjunction is created by joining P_1 and P_3 given that the dissimilarity value is lower than γ and the lowest (row (2) in Figure 1b). Next, the pattern so created

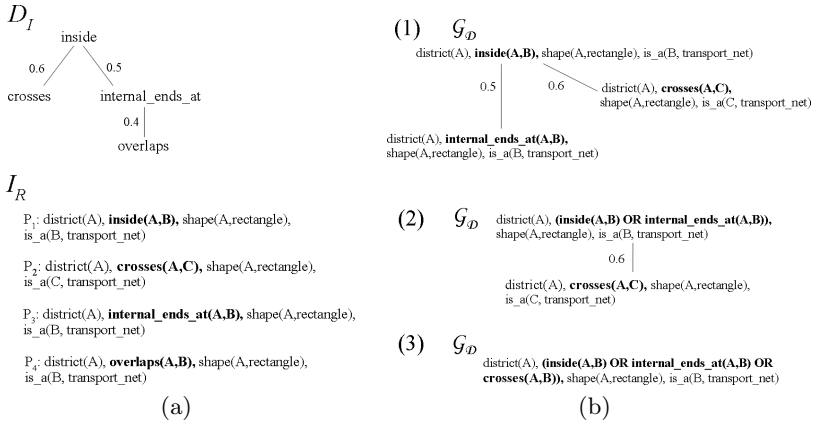

Fig. 1. Extending relational pattern with disjunctions: an example (γ=0.7).

and P_2 are checked for joining. Both have the same length and differ in only one atom. Although the first presents a disjunction and the second presents a "simple" atom, dissimilarity is lower than γ and a new disjunctive pattern is created (row (3) in Figure 1b).

4 Experiments

This approach has been implemented as the extension of the system SPADA aimed to discover relational patterns with disjunctions: the system (afterwards $jSPADA$) is now able to mine conjunctive patterns and disjunctive patterns as well. Here we present the application of both systems to mine spatial networks in a case study of urban accessibility. More precisely, the spatial network is obtained by analyzing both census and digital maps of Stockport, one of the ten districts in Greater Manchester and the analysis is aiming at investigating the accessibility *to* the Stepping Hill Hospital *from* the actual residence of people living far from the hospital. In this case study, transport network, namely the layers of roads, railways and bus priority lines, correspond to the links of the spatial network, while districts close to the hospital and districts distant from the hospital corresponds to the nodes of the network. In accordance with our setting defined in Section 3, districts close to the hospital are target objects while the transport network and districts distant from the hospital are non-target objects.

Property predicates define people with own cars and are *no_car()*, *one_car()*, *two_cars()*, *three_more_cars()*. Structural predicates represent binary topological relationships between districts and roads, railways or bus lines, and correspond to the twelve feasible relations between a region and a line according to the 9-intersection model [7]. Here, background knowledge D_I has been defined on the structural predicates and the dissimilarity values have been manually determined

by applying the Sokal-Michener dissimilarity measure on the matrix representation of the twelve relations[5]: for instance, the following $external\_ends\_at \overset{0.22}{\leftrightarrow} along$; $along \overset{0.277}{\leftrightarrow} comes\_from$ expresses the similarity among three spatial relationships quantified with 0.22 and 0.2777 respectively. Districts and transport network can be involved in more than one line-region spatial relationships and this advocates the usage of disjunctive patterns. D_E contains 1147 ground atoms for 152 target objects.

Experiments were performed[1] by tuning the thresholds $minSup$, $nSup$, γ and the results are reported in Figure 2. A comparison between SPADA and jSPADA has been conducted by varying $minSup$, while, for jSPADA, the values of $nSup$ and γ are set to 0.005 and 0.6 respectively. As we can see the histogram values reported in Figure 2a, jSPADA discovers a number of patterns that is higher than that of SPADA. Indeed, jSPADA returns a set which includes those frequent conjunctive (generated by SPADA) and those disjunctive generated by re-evaluating the infrequent conjunctive ones. Thus, as $minSup$ increases, the range $[nSup; minSup)$ becomes larger and, generally, more disjunctive patterns are extracted while the number of conjunctive frequent patterns decreases. It is worthy that the set of only disjunctive patterns (the complement of the set of patterns of jSPADA relative to the set of SPADA) is actually much smaller than the set of only conjunctive patterns (patterns of SPADA). For instance, when $minSup$=0.007 the number of disjunctive patterns amounts to 5, while the number of conjunctive patterns is 898. Thus, the problem of huge amounts of disjunctive patterns is not so relevant as in the case of conjunctive patterns. This is a clear advantage of the proposed approach since the classical problem of manual analysis of patterns is mitigated.

As expected, also the threshold $nSup$ has influence on the patterns discovered by jSPADA. Indeed, from the figures 2c and 2d ($minSup = 0.025$ and $\gamma = 0.6$) we note that jSPADA is highly sensitive to $nSup$ since the number of disjunctive patterns is reduced of one order of magnitude (from 20 to 0) while $nSup$ is increased by factor of two (from 0.01 to 0.02). By comparing the plots (a), (c) and (d) we note that, by varying $minSup$, jSPADA has a limited capacity in unearthing infrequent patterns (but potentially interesting) than when varying $nSup$. This confirms the viability of the approach to discover new forms of interesting patterns. The sensitivity of the algorithm can be evaluated with respect to the dissimilarity of the disjunctions (Figure 2b). At high values of γ disjunctions can be created also between relationships whose similarity is small, so the patterns present disjunctions with several atoms and the final set is larger. On the contrary, lower values of γ permit to identify disjunctions only between very similar relationships, so the disjunctions present less atoms and the final set is smaller: when γ is set to 0.4, no disjunction is created since the minimum value of similarity between relationships amounted to 0.44.

A comparison between jSPADA and SPADA can also be done from a qualitative viewpoint. jSPADA enables the discovery of patterns which enrich the information extracted by SPADA. For instance, the pattern discovered by SPADA

[1] Data and results are accessible at http://www.di.uniba.it/∼loglisci/jSPADA/

$P_1 : district(A), comes\_from(A, B), is\_a(B, road), comes\_from(A, C), is\_a(C, road)$ $[support : 12\%]$

is enriched by P_2 discovered by jSPADA:

$P_2 : district(A), [comes\_from(A, C)|external\_ends\_at(A, C)], is\_a(C, road),$ $comes\_from(A, B), is\_a(B, rail)$ $[support : 16\%]$

which introduces the disjunctions $comes\_from(A, C)|external\_ends\_at(A, C)$ between two structural predicates. P_2 expresses the information that the road named as C can be connected to the district named as A through two possible simultaneous or alternative ways, $comes\_from(A, C)$ (C starts in A and terminates outside A) and $external\_ends\_at(A, C)$ (C starts outside A and terminates inside A). Remarkably, the support of P_2 is higher than that of P_1. jSPADA permits also the discovery of completely novel patterns that SPADA neglects. One of these is the following:

$P_3 : district(A), [external\_ends\_at(A, B)|along(A, B)|comes\_from(A, B)],$ $three\_more\_cars(A, [0.033; 0.114])$ $[support : 11.1\%]$

which introduces a property predicate (i.e., the percentage of households owing more three cars included in [0.033;0.114]) and expresses in the disjunction three possible forms of accessibility to the district A by the transport line B.

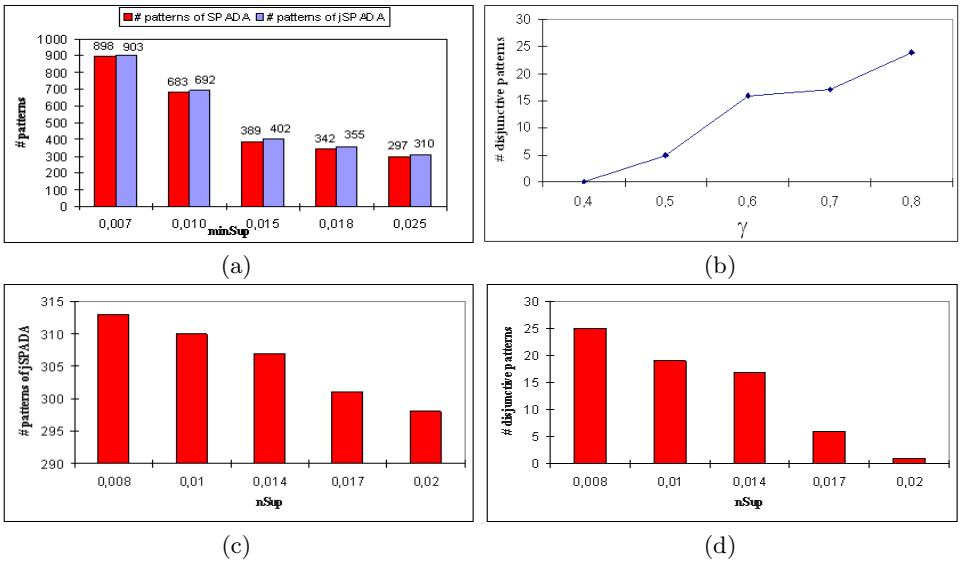

Fig. 2. Number of patterns discovered by tuning $minSup$, $nSup$, γ.

5 Conclusion

In this paper we present a relational data mining approach to discover disjunctive frequent patterns in spatial networks when considering a variance of spatial

relationships existing between two objects. The introduction of disjunctions into the patterns permits to represent spatial relationships which occur simultaneously with or alternatively to others. The application to the problem of urban accessibility points out some peculiarities of the proposal. As future work, we intend to extend experiments to evaluate scalability of the approach.

Acknowledgment. This work is supported by the Strategic Project PS121 "Telecommunication Facilities and Wireless Sensor Networks in Emergency Management" funded by Apulia Region.

References

1. A. Appice, M. Ceci, A. Lanza, F. A. Lisi, and D. Malerba. Discovery of spatial association rules in geo-referenced census data: A relational mining approach. *IDA*, 7(6):541–566, 2003.
2. M. Ceci, A. Appice, and D. Malerba. Discovering emerging patterns in spatial databases: A multi-relational approach. In *PKDD*, volume 4702 of *LNCS*, pages 390–397, 2007.
3. S. Ceri, G. Gottlob, and L. Tanca. *Logic Programming and Databases*. Springer, 1990.
4. D. Davidov and A. Rappoport. Geo-mining: discovery of road and transport networks using directional patterns. In *EMNLP '09*, pages 267–275, 2009.
5. E. Diday and F. Esposito. An introduction to symbolic data analysis and the sodas software. *Intell. Data Anal.*, 7(6):583–601, 2003.
6. S. Dzeroski and N. Lavrac. *Relational Data Mining*. Springer-Verlag, 2001.
7. M. J. Egenhofer and R. D. Franzosa. Point set topological relations. *International Journal of Geographical Information Systems*, 5:161–174, 1991.
8. W. Jin, Y. Jiang, W. Qian, and A. K. H. Tung. Mining outliers in spatial networks. In *DASFAA*, volume 3882 of *LNCS*, pages 156–170. Springer, 2006.
9. W. Klösgen and M. May. Spatial subgroup mining integrated in an object-relational spatial database. In *PKDD*, volume 2431 of *LNCS*, pages 275–286, 2002.
10. F. A. Lisi and D. Malerba. Inducing multi-level association rules from multiple relations. *Machine Learning*, 55(2):175–210, 2004.
11. D. Malerba. A relational perspective on spatial data mining. *IJDMMM*, 1(1):103–118, 2008.
12. D. Malerba, M. Ceci, and A. Appice. Mining model trees from spatial data. In *PKDD*, volume 3721 of *LNCS*, pages 169–180, 2005.
13. A. A. Nanavati, K. P. Chitrapura, S. Joshi, and R. Krishnapuram. Mining generalised disjunctive association rules. In *CIKM*, pages 482–489. ACM, 2001.
14. G. D. Plotkin. A note on inductive generalization. *Mach. Intell.*, 5:153–163, 1970.
15. J. F. Roddick and P. Fule. Semgram - integrating semantic graphs into association rule mining. In *AusDM*, volume 70 of *CRPIT*, pages 129–137, 2007.
16. M. L. Yiu and N. Mamoulis. Clustering objects on a spatial network. In *SIGMOD Conference*, pages 443–454. ACM, 2004.
17. Q. Zhang. Modeling structure and patterns in road network generalization. *7th ICA Workshop on Generalization*, 2004.