International Journal of Engineering Science Invention
ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726
www.ijesi.org ||Volume 5 Issue 12|| December 2016 || PP. 16-23

Clustering of Structured Spatial Object

Murugesan.R[1], Rajiv Gandhi.K[2]
[1](*Research Scholar,PRIST University, Thanjavur, India)*
[2](*CS Department, Alagapa University, Karaikudi, India*

ABSTRACT*: Spatial data mining is the process of discovering interesting and previously unknown, but potentially useful, patterns from large spatial datasets. Clustering is one of the most valuable methods in spatial data mining. As there exist a number of methods for clustering. We analyses different approaches to clustering spatial data, we present some solutions devised to manipulate such data and we find out the main strengths and drawbacks of each method. Then, we present a new method for clustering complex training observations arranged in a graph (called discrete spatial structure). The method, named COSSO (Clustering Of Structured Spatial Objects), clusters observations on the basis of two criteria: i) each cluster should correspond to a connected sub graph of the original discrete spatial structure, and ii) observations in a cluster should be similar according to a similarity measure defined for structured objects (i.e., objects described by multiple database relations). Finally, we present specific issues of the proposed method, namely the selection of the seed observations and the evaluation of the homogeneity of a cluster.*

KEYWORDS - *Graph-based, Multi-relation, Spatial object, Spatial Structure, Seed Selection.*

I. INTRODUCTION

In this paper, we focus our attention on spatial clustering [3] that is, clustering data characterized by a spatial dimension. Spatial data have a specific shape and position in a given reference frame which implicitly define spatial relationships (e.g., intersection, adjacency, and so on). Moreover spatial data represent objects of different types (e.g., towns and rivers) which are naturally described by different relations of a relational database. For this reason, the most natural approach to spatial clustering seems to be the multi-relational one. We consider training observations as complex units of analysis described by a number of database relations, moreover we see training observations as nodes of a graph and we view at clustering as a graph-based partitioning [11]. The result of the clustering is a logical theory which describes each partition (or cluster) as in the case of conceptual clustering.

We analyses different approaches to clustering spatial data, then, we present a new method for clustering complex training observations arranged in a graph (called discrete spatial structure). The method, named COSSO(Clustering Of Structured Spatial Objects)[7], a spatial clustering method that combines the spatial and multi-relational approach in order to take advantage from the strengths of both of them and be able to group together structured (or multi-relational) spatial objects into meaningful classes. Its main characteristic is the construction of clusters where objects are arranged according to both their spatial relations and their structural resemblances. The output is not only the list of resulting clusters with the membership of each input object into a cluster, but also the model associated with each obtained cluster, that is, the logical theory (expressed in a first-order logic formalism) describing the common substructure of objects belonging it. In this way, COSSO provides users with the reason according to which each cluster has been created other than the arrangement of spatial objects into groups.

II. BACKGROUND AND MOTIVATIONS

The motivation that has given rise to COSSO is to build a spatial clustering approach that, differently from the available ones, is able to consider spatial entities as structured objects by overcoming the single-table assumption that characterizes the spatial clustering methods. In this way, it is possible to group objects not only on the basis of spatial features and relations (distance or connectivity, for instance), but also according to their structural similarity, leading to a more meaningful set of obtained clusters, where objects are grouped together not only because they are sufficiently close each other's, but also because they are sufficiently similar.

In spatial framework, gathered data are usually associated with areas (expressed as either irregular partitions of the available space or regular grid) rather than points in the space.

Areal data can be represented as point data by identifying each area with its centroid, but this is restrictive when observations for an area are descriptive of one or more (spatial) primary units, possibly of different types, collected within the same area boundary. In this case, data includes both attributes that relate to primary units or areas and attributes that refer to relations between primary units (e.g., contact frequencies between households)

and between areal units (e.g., migration rates). Moreover, spatial-referencing poses a further degree of complexity due to the fact that the geometrical representation (point, line or polygon) and the relative positioning of primary units or areal units implicitly define spatial features (properties and relations) of different nature, that is, geometrical (e.g. area, distance), directional (e.g. north, south) and topological (e.g. crosses, on top) features. This relational information may be responsible for the spatial variation among areal units and it is extremely useful in descriptive modeling of different distributions holding for spatial subsets of data. An extra consequence is that observations across space cannot be independent due to the spatial continuity of events occurring in the space. Continuity of events over neighbor areas is a consequence of social patterns and environmental constraints that deal with space in terms of regions and allow to identify a mosaic of nearly homogeneous areas in which each patch of the mosaic is demarcated from its neighbors in terms of attributes levels. For instance, the spatial continuity of an environmental phenomenon such as air pollution may depend on the geographical arrangements of pollution sources. As a model for this spatial continuity, the regional concept encourages the analyst to exploit spatial correlation following from the first Law of Geography, according to which "**everything is related to everything else, but near things are more related than distant things**". This means that primary units forming areal units of analysis will tend to be essentially identical members of same populations in nearby locations. In this spatial framework, relations among areal units of analysis are expressed in form of relational constraints that represent a **discrete spatial structure** arising in spatial data, while relations among primary units within an area model the spatial structure of each single areal unit of analysis.

The spatial structure above mentioned is said to be **discrete** to emphasize the fact that most of spatial phenomena are continuous (air pollution, for instance) but we are interested in the discretization of such phenomena in order to generate models of them. Data structures employed to represent discrete phenomena are **tessellation** and **vector**. The former partitions the space into a number of cells each of which is associated with a value of a given attribute. No variation is assumed within a cell and values correspond to some aggregate function (e.g., average) computed on original values in the cell. A grid of square cells is a special tessellation model called **raster**. This model is simple but the geometry of a spatial object is imprecise and requires large storage capabilities. In the **vector** model the geometry is represented by a vector of coordinates. This is a concise and precise representation but involved data structures are complex and the computation of spatial operations, such as intersection, is computationally demanding.

We propose to represent the discrete spatial structure as a graph, where nodes are associated with relational descriptions of areal units to be clustered, while links express relational constraints which typically reflect spatial relations such as adjacency. In this way, discontinuity in the graph represents some obstacles in the space.

Considering the clustering [8] strategy adopted and the formalism in which data and models are expressed, COSSO takes advantage from the contribution of different approaches, as illustrated in Figure 1.

From the spatial clustering approach, COSSO takes the basic requirement for the objects aggregation: the existence of spatial relations among the considered objects. As the main goal is to group together spatial objects, the first requirement of a cluster is that each object belonging to it must be related to some other object in the same cluster. As a consequence, two objects not linked by any of the defined spatial relations cannot belong to the same cluster.

From the graph-based clustering[6] approach, the method gets the graph-based representation as the structure adopted to easily and effectively represent spatial objects and their relations (discrete spatial structure). Consequently, the problem of detecting clusters of objects can be reformulated as a graph-partitioning problem, where the goal is to find the best partitioning of the graph such that each partition includes objects resulting to be connected and similar according to a given measure. However, differently from the graph-partitioning approaches where partitions are calculated by only taking into account links among nodes, our purpose is consider structural similarity of objects as well.

The contribution of conceptual clustering approach is that each detected cluster in COSSO is associated with an implicit definition (or model), expressed in first-order logic formalism, representing the objects belonging to it. Such models are used by the method to evaluate the homogeneity of the set of objects candidate to become a cluster. However, they are also provided as the output of the method since they can be considered as an intentional description of the clusters (in addition to the extensional description consisting of the list of objects belonging to the clusters).

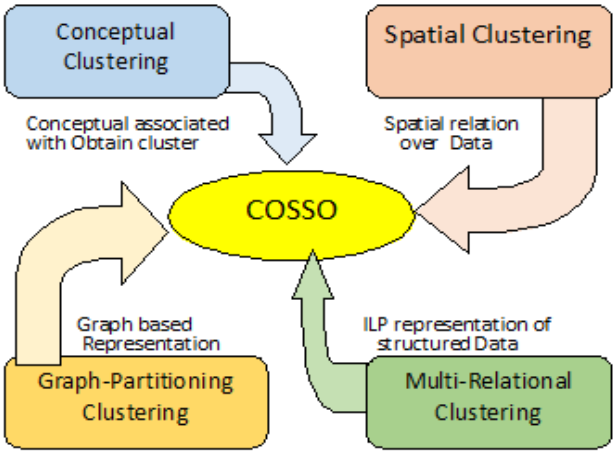


Figure 1. COSSO: clustering approaches contributions

Finally, the methods takes from the MRDM (Multi-Relation Data Mining) approach the capability of dealing with relational (or structured) objects and using the ILP framework as the basic mechanism to reason about them in order to induce their models (or generalizations).

III. THE CLUSTERING OF STRUCTURED SPATIAL OBJECT

In a quite general formulation, the problem of clustering structured objects (e.g., complex areal units), which are related by links representing persistent relations between objects (e.g., spatial correlation), can be defined as follows:

Given:

- a set of structured objects O
- a background knowledge BK and
- a binary relation R expressing links among objects in O;

Find

a set of homogeneous clusters $C \subseteq \wp(O)$ that is coherent with R.

Each structured object $o_i \in O$ can be described by means of a conjunctive ground formula (conjunction of ground selectors) in a first-order formalism (see Example 1), while background knowledge (BK) is expressed with first-order clauses that support some qualitative reasoning on O (see Example 2). In both cases, each basic component (i.e., selector) is a relational statement in the form $f(t_1, \ldots, t_n) = v$, where f is a function symbol or descriptor, t_i are terms (constant or variables) and v is a value taken from the categorical or numerical range of f. Structured objects are related by R that is a binary relation $R \subseteq O \times O$ imposing a discrete structure on O. In spatial domains, this relation may be either purely spatial, such as topological relations (e.g. adjacency of regions), distance relations (e.g. two regions are within a given distance), and directional relations (e.g. a region is on south of another region), or hybrid, which mixes both spatial and non-spatial properties (e.g. two regions are connected by a road).

Example 1. Let us consider data consisting of observations for a site (e.g., areal units) descriptive of one or more (spatial) primary units, possibly of different type, collected within the same site boundary. The areal units are the (structured) objects to be clustered, while the discrete data structure is naturally imposed by the spatial adjacency relation among areal units. Areal units are described in terms of both spatial and a spatial properties, such as:

arealunit(apulia) ← contain(apulia, bari), is_a(bari) = town,
inhabitants(bari) = 342129, contain(apulia, taranto), is_a(taranto) = town, distance(bari, taranto) = 98, . . .
arealunit (basilicata) ← contain(apulia, potenza), is_a(potenza) = town,
inhabitants (potenza) = 68141, contain(apulia, matera), is_a(matera) = town,
. . .

In this case adjacent (apulia, basilicata) and adjacent (basilicata, apulia) are two instances of the relation R (adjacency relation).

Example 2. Background knowledge is a source of domain independent knowledge. For example the definite clause:

accessibility(X, Y) ← town(X) = XName, town(Y) = Y Name,
cross(X, Z) = true, cross(Y, Z) = true, road(Z) = ZName.

expresses accessibility of a town from another town by means of one road.

The relation R can be described by the graph G = (N_O , A_R) where N_O is the set of nodes n_i representing each structured object o_i and A_R is the set of arcs $a_{i,j}$ describing links between each pair of nodes (n_i, n_j) according to the discrete structure imposed by R. This means that there is an arc from n_i to n_j only if R(o_i, o_j) holds.
Let $N_R(n_i)$ be the R-neighborhood of a node n_i such that $NR(n_i) = \{n_j \mid$ there is an arc linking n_i to n_j in G$\}$. Then, a node n_j is R-reachable from ni if $n_j \in N_R(n_i)$, or $\exists n_h \in N_R(n_i)$ such that n_j is R-reachable from n_h.
According to this graph-based formalization, a clustering $C \subseteq \wp(O)$ is coherent with the discrete structure imposed by R (or, shortly, coherent with R) when two objects o_1 and o_2 can belong to the same cluster C_i only if a linking path exists from o_1 to o_2 (or vice-versa) according to R.
Moreover, the cluster C is homogeneous when it groups structured objects of O sharing a similar relational description according to some similarity criterion.

COSSO integrates a neighborhood-based graph partitioning to obtain clusters which are coherent with the discrete structure defined by R and resorts to a multi-relational approach to evaluate similarity among structured objects and form homogeneous clusters. This faces with the spatial issue of modeling spatial continuity of a phenomenon over the space.
The top-level description of the method is presented in Algorithm 1.

Algorithm 1 Top-level description of COSSO algorithm.

```
1: function COSSO(O, BK, R, h − threshold) → CList;
2: CList ← ∅; O_BK ←saturate(O,BK); C ← newCluster( );
3: for each seed ∈ O_BK do
4: if seed is UNCLASSIFIED then
5:     N_seed ← neighborhood(seed,O_BK ,R);
6:   for each o ∈ N_seed do
7:       if o is assigned to a cluster different from C then
8:     Nseed = Nseed/o;
9:       end if
10:  end for
11:  T_seed ← neighborhoodModel(N_seed);
12:  if homogeneity(N_seed, T_seed) ≥ h − threshold then
13:      C.add(seed); seedList ← ∅;
14:      for each o ∈ N_seed do
15:          C.add(o); seedList.add(o);
16:      end for
17: (C,T_C)←expandCluster(C,seedList,O_BK,R,T_seed,h−threshold);
18: CLabel=clusterLabel(T_C ); CList.add(hC, CLabeli);
19: C ← newCluster( );
20: else
21: seed ← N OISE;
22: end if
23: end if
24: end for
25: return CList;
```

COSSO embeds a saturation step (function **saturate**) to make explicit information that is implicit in data according to the given BK. New information (in form of literals) is added to object descriptions by repeatedly applying BK rules to available descriptions of data until no additional literals can be derived from the application of the BK.
The key idea is to exploit the R-neighborhood construction and build clusters coherent with R-discrete structure by merging partially overlapping homogeneous neighborhood units. Cluster construction starts with an empty cluster (C ← newCluster()) and chooses an arbitrary node, called seed, from G.
The R-neighborhood N_{seed} of the node seed is then built according to G discrete structure (function **neighborhood**) and the first-order theory T_{seed} is associated to it. T_{seed} is built as a generalization of the objects falling in N_{seed} (function **neighborhood Model)**.

Algorithm 2 Expand current cluster by merging homogeneous neighborhood.

1: **function** expandCluster(C, seedList, O_{BK} , R, T_C , h − threshold) → (C, T_C **)**;
2: **while** (seedList is not empty) **do**
3: seed←seedList.first(); N_{seed} ← neighborhood(seed,O_{BK} ,R);
4: **for** each o ∈ N_{seed} **do**
5: **if** o is assigned to a cluster different from C **then**
6: N_{seed} = N_{seed}/o;
7**: end if**
8: **end for**
9: T_{seed} ← neighborhoodModel(N_{seed});
10: **if** homogeneity(N_{seed}, {T_C , T_{seed}})≥ h − threshold **then**
11: **for** each o ∈ N_{seed} **do**
12: C.add(o); seedList.add(o);
13: **end for**
14: seedList.remove(seed); T_C ← T_C ∪ T_{seed};
15: **end if**
16**: end while**
17**: return** (C, T_C**)**;

When the neighborhood is estimated to be a homogeneous set (function **homogeneity**), cluster C is grown with the structured objects enclosed in N_{seed} which are not yet assigned to any cluster. The cluster C is then iteratively expanded by merging the R-neighborhoods of each node of C (neighborhood expansion) when these neighborhoods result in homogeneous sets with respect to current cluster model T_C (function **expandCluster**, see Algorithm 2). T_C is obtained as the set of first-order theories generalizing the neighborhoods merged in C. It is noteworthy that when a new R-neighborhood is built to be merged in C, all the objects which are already classified into a cluster different from C are removed from the neighborhood. When the current cluster cannot be further expanded it is labeled with CLabel and an unclassified seed node for a new cluster is chosen from G until all objects are classified. CLabel is obtained by T_C (function **labelCluster**) to compactly describe C.

This is different from spatial clustering performed by GDBSCAN (*Generalized Density Based Spatial Clustering of Application with Noise*) [10], although both methods share the neighborhood-based cluster construction. Indeed, GDBSCAN retrieves all objects density-reachable from an arbitrary core object by building successive neighborhoods and checks density within a neighborhood by ignoring the cluster. This yields a density-connected set, where density is efficiently estimated independently from the neighborhoods already merged in forming the current cluster. However, this approach may lead to merge connected neighborhoods sharing some objects but modeling different phenomena. Moreover, GDBSCAN computes density within each neighborhood according to a weighted cardinality function (e.g. aggregation of non-spatial values) that assumes single table data representation. COSSO overcomes these limitations by computing density within a neighborhood in terms of degree of similarity among all relationally structured objects falling in the neighborhood with respect to the model of the entire cluster currently built. In particular, following the suggestion given in [14], we evaluate homogeneity within a neighborhood N_{seed} to be added to the cluster C as the average degree of matching between objects of N_{seed} and the cluster model {T_C , T_{seed}}. Details on cluster model determination, neighborhood homogeneity estimation and cluster labeling are reported below.

3.1. Cluster Model Generation

Let C be the cluster currently built by merging w neighborhood sets N_1, . . . , N_w , we assume that the cluster model T_C is a set of first-order theories {T_1, . . . , T_w } for the concept C where T_i is a model for the neighborhood set N_i. More precisely, T_i is a set of first-order clauses: T_i : {cluster(X) = c ← H_{i1}, . . . , cluster(X) = c ← H_{iz} }, where each H_{ij} is a conjunctive formula describing a sub-structure shared by one or more objects in N_i and ∀o_i ∈ N_i, BK ∪ T_i |= o_i. Such model can be learned by resorting to the ILP (*Inductive Logic Programming*) system ATRE (*Apprendimento di Teorie Recursive da Esempi*) that adopts a separate-and-conquer search strategy to learn a model of structured objects from a set of training examples and eventually counter-examples. In this context, ATRE learns a model for each neighborhood set without considering any counter-examples. The search of a model starts with the most general clause, that is, cluster(X) = c ←, and proceeds top-down by adding selectors (literals) to the body according to some preference criteria (e.g. number of objects covered or number of literals).

Selectors involving both numerical and categorical descriptors are handled in the same way, that is, they have to comply with the property of likeness and are sorted according to preference criteria. The only difference is that selectors involving numerical descriptors are generalized by computing the closed interval that best covers positive examples and eventually discriminates from counter-examples, while selectors involving categorical descriptors with the same function value are generalized by simply turning all ground arguments into corresponding variables without changing the corresponding function value.

3.2. The Homogeneity Evaluation of Sets of Objects

In classical clustering methods, objects grouping is performed by exclusively relying on some measures of object similarity. Such measures of similarity are **context free**, in the sense that the similarity between any two objects depends solely on the properties of the two objects involved in the computation and is not influenced by any context. Consequently, methods that use such measures are unable to capture the properties that characterize a cluster as a whole and are not derivable from properties of individual entities.

In order to detect such properties, the system should be equipped with the ability to recognize configurations of objects representing certain global concepts, leading to the basic notion of conceptual clustering where the dominant aim is to find the concepts underlying the objects distribution.

In COSSO we are interested in measuring how much objects arranged into a neighborhood are homogeneous, that is, similar each other's. This imposes that similarity evaluation must be performed among all the objects belonging the same group and not calculated between pairs of objects. In other words, we need a unique value of homogeneity associable with a group of objects rather than a number of homogeneity values concerning all the possible pairs of objects. To this aim, our similarity measure is evaluated by generating a model (in term of a logical theory) generalizing all the objects belonging to a given neighborhood and by calculating the average similarity between this model and each object in the set. In this way, when objects are very similar each other, their model will be very close to each of them since it will be described by common features by excluding the different ones and, consequently, the homogeneity value will be high. On the contrary, when objects are dissimilar each other, their model will tend to be quite general since the number of common features will be low, this leading to a low value of homogeneity evaluation.

IV. THE SEED SELECTION PROBLEM

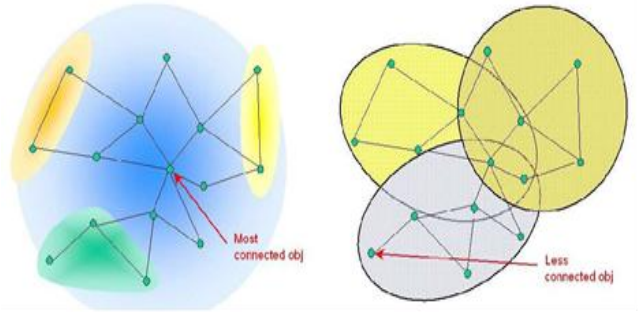
The cluster shapes depend on the object that COSSO chooses at each step as seed of the neighborhood to be considered. As basic approach, COSSO adopts a sequential strategy. In the case a new cluster has to be discovered, the seed is the first object accessed in O not yet assigned to any cluster. In the case an existing cluster has to be expanded, the seed is the object stored in the first position of a list collecting the cluster objects not yet considered for the expansion step.

Algorithm 3 Build a compact theory to describe a cluster C

```
1: function clusterLabel(T_C) → T_C';
2: T_C' ← ∅
3: merge ← false;
4: while T_C is not empty do
5: H is a first-order clause in T_C ;
6: T_C = T_C /H;
7: for each H' ∈ T_C do
8: if H and H' are generalizable without lost of information then
9: H = generalize(H,H'); T_C = T_C /H'; merge = true;
10: end if
11: end for
12: T_C' = T_C' ∪ H;
13: end while
14: if merge is true then
15: T_C' ←clusterLabel(T_C');
16: end if
17: return T_C';
```

The sequential seed selection (SEQ) is efficient, but quality of clustering clearly depends on the "order" of storing and accessing objects, which is not necessarily the best one. For this reason, two alternative strategies have been empirically investigated and implemented in COSSO that take advantage from the graph structure in seed selection step. In particular, we base the choice of the candidate seed on the concept of density (cardinality)

of a neighborhood.
In each seed selection, the "best" candidate seed is the object whose neighbor-hood in the graph has the highest density (or the first one according to the descending order of connectivity, **DESC**) or the lowest density (or the first one according to the ascending order of connectivity, **ASC**).
In the first case (DESC), we follow the intuition coming from the density-based framework where dense areas are labeled as clusters. Our suggestion is to estimate density in terms of the number of connections in the graph from the candidate seed to its neighboring objects. According to this strategy, the objects to prefer in the seed selection are the most connected ones because in general social and environmental phenomena affecting a spatial region (such as unemployment rates in population or pollution distribution over territory) are more prominent in the centre of the area and tend to soften in peripheric zones (see Figure 2.a).
In the second case (ASC), the basic assumption is that a dense area in the graph may correspond to an area covered by contiguous different clusters, hence, discovery should preferentially start from peripheral objects. In this strategy the less connected objects are preferred in the seed selection since the underlying idea is that objects with a high number of connections are also affected by a higher number of phenomena characterizing surrounding areas (see Figure 2.b).



(a) Spatial arrangement of phenomena (b) Phenomena overlapping in central areas

Figure 2: Motivations of descending (a) and ascending (b) order of connectivity in the seed selection

Currently, COSSO implements all the three strategies resented above for the seed selection: sequential selection (SEQ), ascending order of connectivity (ASC) and descending order of connectivity (DESC). However, as future work, the method could be extended by a further strategy that puts together all the above mentioned strategies. In particular, the idea is to calculate the clustering results coming from the three approaches and select that one that is best scored according to some heuristic that evaluates the clusters quality.

V. CONCLUSIONS

In this paper we have presented COSSO, a spatial clustering method that overcomes limitations of both spatial clustering and multi-relational approach. We have shown how COSSO detects clusters of objects by taking into account the spatial correlation of data and their structural resemblances. Spatial correlation of data is represented by links among nodes corresponding to spatial objects in a graph-like structure where data are arranged, while structural resemblance is detected by relying on a homogeneity evaluation function expressing how much objects in a neighborhood are similar each other. COSSO builds clusters by putting together objects that are both linked and similar each other.

In addition, issues concerning the seed selection are discussed and some solution have been proposed and implemented in the system.

References

[1] G. Vladimir Estivill-Castro and Ickjai Lee. "Fast spatial clustering with different metrics and in the presence of obstacles". In International Symposium on Advances in geographic information systems, pages 142–147. ACM Press, 2001.
[2] Margherita Berardi, Antonio Varlaro, and Donato Malerba. "On the effect of caching in recursive theory learning". In Inductive Logic Programming, 14th International Conference, ILP 2004, pages 44–62, 2004.
[3] Jiawei Han, Micheline Kamber, and Anthony K.H. Tung. "Spatial clustering methods in data mining: A survey". pages 188–217, 2001.
[4] Michelangelo Ceci and Annalisa Appice. "Spatial associative classification: propositional vs structural approach". J. Intell. Inf. Syst., 27(3):191–213, 2006.
[5] E. Hancock and M. Vento. "Graph Based Representations in Pattern Recognitions". Springer-Verlag, 2003.
[6] Istvan Jonyer, Diane J. Cook, and Lawrence B. Holder. "Graph-based hierarchical conceptual clustering". Journal of Machine Learning Research, 2:19–43, 2001.

[7] Donato Malerba, Annalisa Appice, Antonio Varlaro, and Antonietta Lanza. "Spatial clustering of structured objects". In Stefan Kramer and Bernhard Pfahringer, editors, Inductive Logic Programming, 15th International Conference, ILP 2005, volume 3625 of Lecture Notes in Computer Science, pages 227–245. Springer, 2005.
[8] Vladimir Batagelj and Anuška Ferligoj. "Clustering relational data". In Wolfgang Gaul, Otto Opitz, and Martin Schader, editors, Data Analysis, pages 3–15. Springer-Verlag, 2000.
[9] Martin Ester, Hans-Peter Kriegel, and Jörg Sander. "Algorithms and applications for spatial data mining". Geographic Data Mining and Knowledge Discovery, 5(6), 2001.
[10] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. "A densitybased algorithm for discovering clusters in large spatial databases with noise". In Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining, pages 226–231, 1996.
[11] Jesus A. Gonzalez, Lawrence B. Holder, and Diane J. Cook. "Graph-based concept learning". In FLAIRS Conference, pages 377–381, 2001.
[12] Lawrence B. Holder and Diane J. Cook. "Graph-based relational learning:current and future directions". SIGKDD Explorations, 5(1):90–93, 2003.
[13] Stefan Kramer, Nada Lavrač, and Peter Flach. "Relational Data Mining, chapter Propositionalization Approaches to Relational Data Mining", pages 262–291. LNAI. Springer-Verlag, 2001.
[14] D. Mavroeidis and P.A. Flach. "Improved distances for structured data". In T. Horváth and A. Yamamoto, editors, Inductive Logic Programming, 13th International Conference, volume 2835, pages 251–268. Springer-Verlag, 2003.