Discovery of General Knowledge in Large Spatial Databases

Wei Lu & Jiawei Han†
School of Computing Science
Simon Fraser University
Burnaby, British Columbia, Canada V5A 1S6

and

Beng Chin Ooi
Department of Information Systems and Computer Science
National University of Singapore
Lower Kent Ridge, Singapore 0511

Abstract

Extraction of interesting and general knowledge from large spatial databases is an important task in the development of spatial data- and knowledge-base systems. In this paper, we investigate knowledge discovery in spatial databases and develop a generalization-based knowledge discovery mechanism which integrates attribute-oriented induction on nonspatial data and spatial merge and generalization on spatial data. The study shows that knowledge discovery has wide applications in spatial databases, and relatively efficient algorithms can be developed for discovery of general knowledge in large spatial databases.

1. Introduction

Spatial reasoning using data and knowledge stored in large spatial databases is a crucial task in the development of geographical information systems, medical imaging and robotics systems. Because of the huge amount (usually, tera-bytes) of spatial data obtained from satellites, video cameras, medical equipments, etc., it is costly and often unrealistic for users to examine the spatial data in detail and extract interesting knowledge or general characteristics from spatial databases. This motivates the study and development of knowledge discovery mechanisms for large spatial databases.

Knowledge discovery in spatial databases is the extraction of interesting spatial patterns and features, general relationships between spatial and nonspatial data, and other general data characteristics not explicitly stored in spatial databases. Such discovery may play an important role at understanding spatial data, capturing intrinsic relationships between spatial and nonspatial data, presenting data regularity in a concise manner, and reorganizing spatial databases to accommodate data semantics and achieve high performance.

† The work was supported in part by the Natural Sciences and Engineering Research Council of Canada under Grant A-3723 and a research grant from Centre for System Science of Simon Fraser University.

There are different philosophical considerations on knowledge discovery in databases [7, 16], which may lead to different methodologies in the development of knowledge discovery techniques. First, we assume that A spatial DB stores a large amount, information-rich, relatively reliable and stable data. Furthermore, the following assumptions are made as the first step in the development of mechanisms for knowledge discovery in spatial DBs.

Assumption 1. A knowledge discovery process is initiated by a user's learning request.

Idealistically, one may expect that a knowledge discovery system will perform interesting discovery autonomously without human interaction. However, since learning can be performed in many different ways on any subset of data in the database, huge amount of knowledge may be generated from even a medium size database by unguided, autonomous discovery, whereas much of the discovered knowledge could be out of user's interests. In contrast, a command-driven discovery may lead to the discovery of what one wants to discover and therefore represents relatively constrained search for the desired knowledge. Thus, command-driven discovery is adopted in this study.

Assumption 2. Background knowledge is available for knowledge discovery process.

Discovery may be performed with the assistance of relatively strong background knowledge (such as conceptual hierarchy information, etc.) or with little support of background knowledge. Obviously, the discovery of conceptual hierarchy information itself can be treated as a part of knowledge discovery process. However, the availability of relatively strong background knowledge not only improves the efficiency of the discovery process but also expresses user's preference for guided generalization, which may lead to efficient and desirable generalization process.

Following these assumptions, our mechanism for knowledge discovery in spatial DB adopts a *learning-from-examples* approach which treats the task-relevant data as examples for learning processes and relies mainly on the generalization process.

There have been many studies on machine learning [5, 6] and some recent studies on knowledge discovery in large databases [3, 7, 9, 10, 12, 16]. These studies set up the foundation for knowledge discovery in spatial databases. Recently, an attribute-oriented approach has been developed for discovery of different kinds of knowledge rules in relational databases [9]. Moreover, a multi-resolution relational data model has been developed [13] for performance improvement in image database applications. Studies on data abstraction in spatial databases, such as spatial data abstraction using picture indexing and feature clustering [4] and geometric abstraction [2], are closely related to knowledge discovery in spatial databases.

In this study, the attribute-oriented induction technique is extended to knowledge discovery in spatial databases. Two kinds of concept hierarchies, thematic concept hierarchies and spatial hierarchies, are constructed for the learning process. Induction can be performed by ascending these hierarchies and summarizing general relationships between spatial and nonspatial attributes at a high concept level. The method can discover interesting interrelationships between spatial and nonspatial data and can be applied to analyzing correlations between different spatial features based on different thematic maps.

The paper is organized as follows. Section 2 presents spatial learning primitives, which include spatial data representations, spatial hierarchies, and expected representation of learning results. Section 3 presents an algorithm for nonspatial-data-dominated spatial learning. Section 4 presents an algorithm for spatial-data-dominated spatial learning. Section 5 discusses the extension of the two algorithms to interleaved generalization and other related issues, and section 6 summarizes the study.

2. Primitives for Knowledge Discovery in Spatial Databases

There are different philosophies for knowledge discovery in spatial databases based on different kinds of databases and different kinds of rules to be extracted from databases. To confine our study to a well-defined domain, the following assumptions are made in this study. First, we assume that the rules to be extracted are general data characteristics and/or relationships, and the learning process is triggered by a learning request (or query) explicitly. Secondly, we assume that a spatial database consists of both spatial and nonspatial data, while the former is relational and stored in a relational database; and the latter is two-dimensional and is stored in spatial data structures. Spatial objects and their associated nonspatial information are linked to each other as in the SAND architecture [1]. There are different representations for spatial data. In many applications, spatial information is stored as thematic maps. Each map contains specific features of spatial objects, e.g., forest type and coverage. There are two representations of thematic maps: *raster* and *vector*.

(1) In a raster image, an attribute value is associated with each pixel. For example, a geomorphological map may have its height coded in color (or grey level).

(2) In a vector representation, an object is specified by its geometry, such as the boundary representation, and its associated thematic attributes. For example, a lake is specified by a sequence of points sampled at the boundary and the elevation value.

These two types of data can be represented as a set of spatially ordered objects, each of which has its spatial and nonspatial components. In the following discussion, a spatial object obj_i is assumed to be denoted as $< geo_i, attri_i >$. For example, each pixel in raster data is represented by $<(x, y), intensity>$, where (x, y) is the spatial location and $intensity$ the nonspatial attribute value.

An important aspect of learning from data is to cluster data into groups with similar characteristics. Different from relational data clustering, which is usually based on the concept hierarchy of each single attribute, spatial data clustering is two dimensional. Spatial aggregation may be obtained by constructing spatial hierarchy or consolidating neighboring spatial objects. Quad-tree and R-tree are typical spatial hierarchical structures [8, 14], where the former is frequently used for raster data, whereas the latter for vector data. Some spatial functions, such as *adjacent_to*, are useful for clustering neighboring spatial objects. In a stable environment, spatial hierarchies and adjacency relationships can often be computed and stored for efficient data retrieval and knowledge discovery [8, 11, 14].

In order to represent general characteristics at a high concept level, attribute concept hierarchies should be provided by domain experts or constructed automatically or semi-automatically by data statistical analysis [15]. In our algorithms, an attribute hierarchy is represented by a function *c_parent*(*attri_val*) which returns a parent (high-level) concept for a given attribute value. A spatial hierarchy may be represented by two functions, *s_parent*(*obj*) which returns the parent node of the object *obj*, and *s_children*(*obj*) which returns the set of all of the children nodes of *obj*.

Semantic concepts in a concept hierarchy satisfy upward consistency. A high-level concept represents information which is more general than but consistent with the lower-level concepts. For example, Figure 1 represents an agriculture hierarchy. The region which grows *corn, wheat, rice*, etc. can be generalized to a *grain-production* area according to the hierarchy. Many high-level concepts for numerical values can be represented by their summary data and served as generalized concepts. For instance, precipitation measurement between 2.0 and 5.0 inches can be either represented by its rainfall range or generalized to "wet", etc.

Discovered knowledge should be concise, informative, and be represented by high-level concepts with a small number of disjuncts (with each representing one case in the generalized rule). A generalization threshold, which represents the expected maximum number of disjuncts in the generalized rule, or a desired concept level can be used in the generalization process.

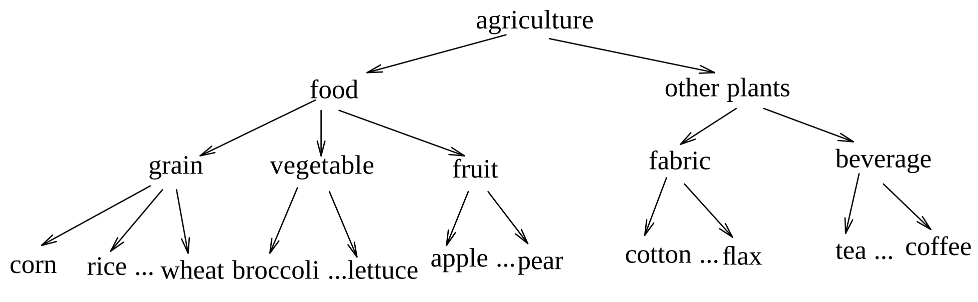

Figure 1. An agriculture hierarchy.

To make a knowledge discovery process focus on a set of interested data and extract desired knowledge, a learning request should be used to trigger the discovery process. Similar to DBLEARN [9], a learning request can be specified in the syntax similar to SQL. One such example is presented here.

Example-1. Given a large set of climate data (monthly mean temperature and monthly precipitation) obtained from over 500 weather stations scattered in British Columbia (B.C.), our task is to find general weather pattern related to different areas in B.C. in the summer of 1990. There are over 18,000 pieces of data records per year. It is impossible to find general weather pattern by simple data retrieval. A generalization process can be initiated by the following query.

extract characteristic rule

from precipitation-map, temperature-map

where province = "B.C." **and** period = "summer" **and** year = 1990

in relevance to region **and** precipitation **and** temperature

Notice that precipitation and temperature are thematic data related to thematic maps, and *summer* is a general concept (higher than month) in a concept hierarchy *period*.

In general, learning requests provide the following primitives for knowledge discovery: the set of relevant data, concept hierarchies, desired rule forms and the learning request. Two learning algorithms are introduced based on the availability of those primitives: one is (nonspatial) attribute-oriented induction, which performs the generalization on nonspatial data first; whereas the other is spatial hierarchy directed induction, which performs generalization on spatial data first.

3. Nonspatial-Data-Dominated Generalization

A spatial database stores both spatial data and their associated nonspatial data. Spatial data is often obtained by preprocessing image data and is stored in high-resolution with large volumes. To extract general knowledge from spatial databases, generalization usually needs to be performed on both spatial and nonspatial data. When one of the components is generalized, the other component will be adjusted accordingly. Based on which component, spatial or nonspatial, to be generalized first, different algorithms, nonspatial-data-dominated generalization vs. spatial-data-dominated generalization, can be derived for different applications. The high-level precipitation concepts and the concept hierarchy for season periods are provided in Table 1 and Figure 2 respectively.

Table 1. High-level precipitation concepts

very dry (v.d.)	dry (d.)	moderately dry (m.d.)	fair(f.)	moderately wet (m.w.)	wet (w)	very wet (v.w.)
[0, 0.1]	(0.1, 0.3]	(0.3, 1.0]	(1.0, 1.2]	(1.2, 2.0]	(2.0, 5.0]	5.0 & up

Suppose that the spatial database stores a map of British Columbia with a set of weather stations scattered around the provinces as shown in Figure 3, where the sample stations: D.C., Kam., Nan., Pen., P.G., P.R., Van. and Vic., are abbreviations for Dawson Creek, Kamloops, Nanaimo, Penticton, Prince George, Prince Rupert, Vancouver and Victoria, respectively. Climate data is collected from these weather stations. The data contains average monthly precipitation and minimum, maximum, and average temperatures for each regional station. Table 2 shows sample precipitation data.

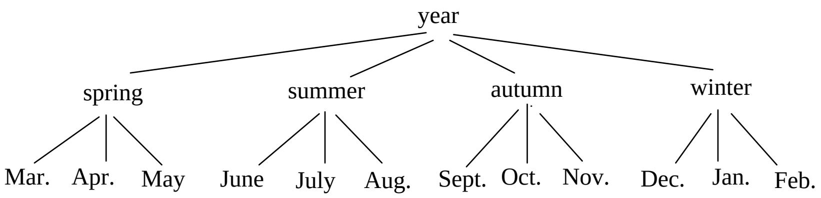

Figure 2. A year-season-month hierarchy.

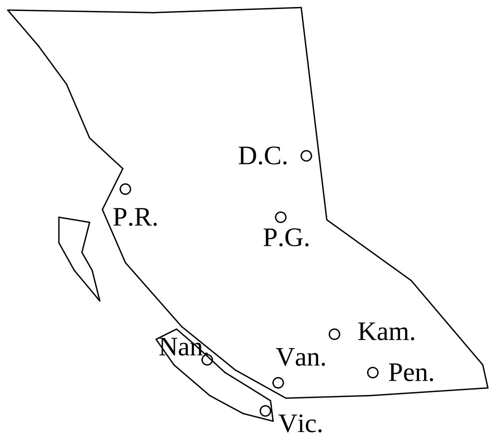

Figure 3. A map of British Columbia.

Table 2. Sample precipitation data (in inch) of 1990.

city	Jan	Feb	Mar	Apr	May	June	July	Aug	Sept	Oct	Nov	Dec	year total
Nanaimo	6.37	4.36	3.99	2.50	1.47	1.55	0.91	1.01	1.73	4.19	6.06	7.11	41.25
Vancouver	8.6	6.1	5.3	3.3	3.0	2.7	1.3	1.7	4.1	5.9	10.0	7.8	59.8
Victoria	11.12	9.74	5.15	2.68	2.51	1.07	0.42	2.42	0.95	2.69	2.64	4.36	45.75
Prince Rupert	9.8	7.6	8.4	6.7	5.3	4.1	4.7	5.2	7.7	12.2	12.3	11.3	95.16
. . .	. . .												

Example-2. Given the above information, the query is to report general precipitation pattern zones in spring 1990, which is represented as below.

extract region

from precipitation-map

where province = "B.C." **and** period = "spring" **and** year = 1990

in relevance to precipitation **and** region.

The learning process can be provided by generalization on nonspatial attribute *precipitation* first, which consists of the following steps.

(1) *Collect related nonspatial data*. The execution of the SQL query on nonspatial data extracts the precipitation records relevant to the province, months and year. Notice

that "period = spring" is a piece of generalized data, which is decomposed into "month = March **or** month = April **or** month = May" by consulting the generalization hierarchy.

(2) *Perform attribute-oriented generalization on collected nonspatial data*. This merges the three months into "spring" and generalizes precipitation attribute values by averaging the precipitation values of the three months. A portion of the table is shown in Table 3. Since the average precipitation value in the nonspatial table contains many distinct values and do not reach a desired concept level, further generalization needs to be performed on the nonspatial data. In this case, the average precipitation value is generalized to an even higher level, such as "wet", "very wet", etc. by consulting the concept hierarchy of precipitation. During the generalization and merge of identical nonspatial tuples, spatial object pointers are collected in the generalized nonspatial data entry.

(3) *Perform spatial generalization*. When the nonspatial data is generalized to the desired level or to a small number of disjuncts, neighboring areas with the same high-level attribute values can be merged together based on a spatial function *adjacent_to*. In order to generalize and merge spatial objects into a small number of regions, it is often necessary to perform approximation. Within a spatial region, if there is only a small portion of the area carrying some attribute values different from that of the majority portion of the area, the small portion can be omitted in the high level description. For example, if poplars occupy only 3% of the area in a pine forest, the generalized description may ignore this small portion of poplars and generalize the area to *pine forest* (with 97% certainty).

In general, the learning process described above can be summarized in the following nonspatial-data-dominated generalization algorithm which generalizes nonspatial data using concept hierarchies and then merges corresponding spatial objects accordingly. The judgement of whether the current generalization on nonspatial data is sufficient can be based either on the number of generalized tuples in the generalized relation (which can be specified as a *generalization threshold*) or on an *appropriate concept level*, which can be specified by users or experts explicitly.

To demonstrate the learning process, our example data is gathered from three Georgia Strait regions, *Vancouver, Victoria* and *Nanamio*. The relevant precipitation data are collected in the columns 2 to 4 in Table 3. Average monthly rainfall for each region is computed in column 5. It is then generalized to the last column using the high-level precipitation concept provided in Table 3.

Since the three neighboring regions carry the same generalized precipitation attribute value "*wet*", the three regions are merged into one, which can be assigned to a meaningful geographic name, such as "*Georgia Strait*" by a user or an expert.

The remaining generalization processes are similar to the above. The learning result is reported in a tabular form as shown in Table 4. Figure 4 shows the learning result of precipitation in spring 1990 for the whole province.

Table 3. The relevant precipitation data of the regions and its generalization.

city	Mar	Apr	May	Avg	high-level concept
Nanaimo	3.99	2.50	1.47	2.85	wet
Vancouver	5.3	3.3	3.0	4.1	wet
Victoria	5.15	2.68	2.51	3.43	wet

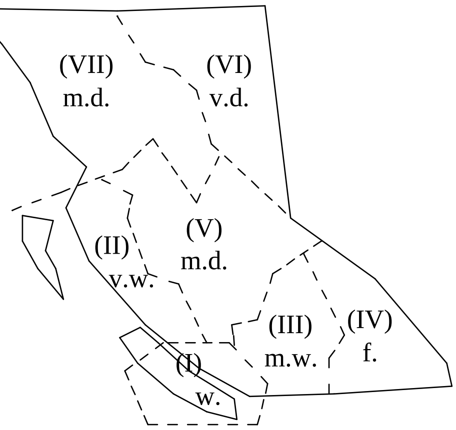

Figure 4. A sample B.C. spring precipitation diagram.

The learning process is summarized into the following algorithm, which generalizes nonspatial data attributes by concept hierarchies ascension and consolidation of adjacent spatial objects with similar attribute values. Generalization terminates when the generalized concept level reaches the desired concept level, or when the number of disjuncts is within a prespecified threshold.

Table 4. General Precipitation Information.

Region	Rainfall
Georgia Strait (I)	wet
Coastal (II)	very wet
Okanagan-Thompson (III)	moderately wet
Columbia-Kootenay (IV)	fair
Central Interior (V)	moderately dry
Peace-Liard (VI)	very dry
Northern Interior (VII)	moderately dry

Algorithm-1 Nonspatial-data-dominated generalization.

Input. (i) A spatial database consisting of a set of nonspatial data and a spatial map, (ii) a learning request which indicates particular interested set of data and the desired threshold (or concept level), and (iii) a set of concept hierarchies.

Output. A rule which characterizes the general properties and/or relationships of spatial objects.

Method.

(1) Collect the set of task-relevant nonspatial data by an SQL query.

(2) Perform attribute-oriented induction repeatedly on the collected nonspatial data by (i) concept hierarchy ascension, (ii) attribute removal, and (iii) merge of identical tuples until the number of tuples is within the generalization threshold, or until every attribute has been generalized to a desired concept level. The spatial object pointers are collected as a set of pointers and put into the generalized nonspatial data entry during the merge of identical tuples.

(3) Generalize the spatial data: for every generalized nonspatial tuple, follow their spatial pointers to retrieve the spatial objects, and perform spatial merge and approximation until the resulting set of generalized spatial objects are reduced to a small set .

(4) Output the generalized rule or the relationship between the generalized nonspatial and spatial data. □

Theorem-1. The complexity of Algorithm-1 is $O(N \log N)$.

Proof. Given a database with nonspatial components of N spatial objects, the retrieval of one component takes $O(\log N)$. The worst case for Step 1, the retrieval of relevant data, may take $O(N\ logN)$. Step 2, nonspatial attribute generalization takes $O(N \log N)$ [9]. The retrieval of relevant spatial objects using the set of pointers obtained from nonspatial data generalization takes at most $O(N)$. With the availability of spatial indices, the adjacent objects of a given object can be found in time $O(\log N)$. Since the maximum number of spatial merge is N, Step 3 takes $O(N\ logN)$. Thus, the overall complexity of the algorithm is $O(N\ logN)$. □

4. Spatial-Data-Dominated Generalization

In some applications, generalization may also be performed first on spatial data based on spatial hierarchical information, which involves partitioning regions stored in spatial data structures, generalizing spatial data to a certain level, then generalizing their corresponding nonspatial components, and merging/grouping the generalized concepts to derive general and concise relationships between nonspatial and spatial data at a high level.

Spatial-data-dominated generalization relies on spatial generalization hierarchies which can be obtained based on (i) the semantics of spatial data, e.g. hierarchical administration regions: county, city, province, etc.; (ii) clustering of spatial objects, e.g. based on densely clustered spatial objects; and (iii) spatial indexing structures, such as R-trees, Quad-trees, etc. We examine one example.

Example-3. Given regional temperature data and high-level concept of temperature (Table 5), the learning task is to find general temperature information in prespecified administration regions for summer 1990. The learning request can be written in an

SQL-like query as follows.

extract characteristic rule
from temperature-map
where province = "B.C." **and** period = "summer" **and** year = 1990
in relevance to region **and** temperature.

Table 5. High-level temperature concepts.

very cold	cold	moderately cold	mild	moderately hot	hot	very hot
−5 & below	[−5, 10)	[10, 32)	[32, 50)	[50, 70)	[70, 90)	90 & up

The major learning steps are as follows.

(1) Collect task-relevant data by an SQL query (on nonspatial data) and corresponding spatial data retrieval.

(2) Generalize spatial database by clustering spatial data objects according to their regions and merge the corresponding nonspatial pointers until it reaches the desired concept level or the number of generalized spatial objects is within the threshold.

(3) For each region, perform generalization on non-spatial objects (e.g. taking average or mean or numerical values) until a small number of concepts which subsume all of the concepts existing in each subregion.

To illustrate the spatial-oriented learning process, we examine Figure 5, the south-central region of the province.

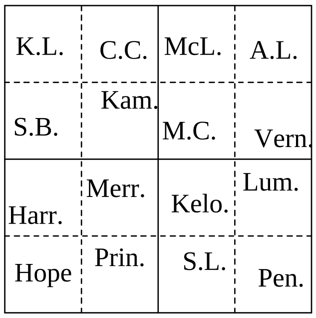

Figure 5. South-Central region of British Columbia.

Table 6 shows the relevant temperature data for that region in summer 1990. The spatial hierarchy is shown by the grid in Figure 5.

In the spatial generalization, objects in each quadrant are first merged according to the first level of the spatial hierarchy, which are in turn merged according to higher level spatial hierarchies. The average of the temperatures in these regions can be computed, which can be in turn generalized to its corresponding high-level concept, such as *moderately hot*, etc. The learning result is shown in Figure 6 and is mapped to Table 7.

Table 6. Relevant temperature data for the south-central region.

city	June	July	Aug	Avg
Adams Lake (A.L.)	62	67	64	64.3
Criss Creek (C.C.)	68	65	64	65.7
Harrison (Harr.)	60	64	63	62.3
Hope	61	65	65	63.3
Kamloops (Kam.)	64	70	68	67.4
Kelowna (Kelo.)	61	66	64	63.3
Kelly Lake (K.L.)	57	61	61	59.7
Lumby (Lum.)	55	62	61	59.3
McLure (McL.)	57	63	62	60.7
Merritt (Merr.)	57	62	61	60
Mont Creek (M.C.)	55	63	58	58.7
Penticton (Pen.)	63	74	67	68
Princeton (Prin.)	58	64	62	61.3
Spences Bridge (S.B.)	57	61	60	59.3
Summerland (Sum.)	64	70	68	67.3
Vernon (Vern.)	57	65	62	61.3

Notice that averaging nonspatial data may not be the most desirable way to generalizing nonspatial data in many cases since averaging may hide exceptions or smooth data excessively. Actually, when the generalized values (such as temperatures) present significantly different values, generalization could return a small number of disjuncts (possibly associated with statistical information in each disjunct). In this case, generalization on the nonspatial data can be performed by clustering and generalizing only those nonspatial components which carry similar data values, as that has been done in attribute-oriented induction of nonspatial data.

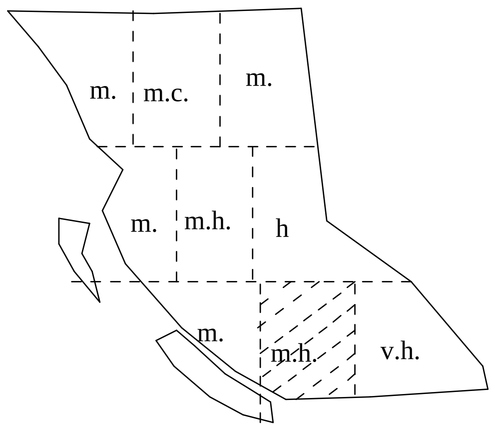

Figure 6. A sample of B.C. summer temperature diagram.

The example can be summarized into a spatial-oriented learning algorithm which utilizes the spatial hierarchy to obtain generalized objects. The generalized attribute value of the new object is obtained by climbing up the attribute concept hierarchy to find a minimal concept which subsumes the attribute values of the corresponding sub-objects.

In the case when the map attribute is numeric, the new attribute value can be determined by weighted average. For example, when the attribute is *precipitation*, the precipitation of a large region can be computed from the precipitation of its sub-regions weighted by the areas of the region. This method can also be applied to generating multi-resolution images.

Table 7. Generalized temperature information.

Region	Temperature
North-West	mild
North-Central	moderately cold
North-East	mild
Mid-West	mild
Central	moderately hot
Mid-East	hot
South-West	mild
South-Central	mild
South-East	very hot

Algorithm-2 Spatial-data-dominated generalization.

Input. (i) A spatial database consisting of a set of nonspatial data and a spatial map, (ii) a spatial hierarchy, (iii) a learning request which indicates the interested set of data and the desired threshold or concept level, and (iv) a set of concept hierarchies.

Output. A rule which characterizes the general properties and/or relationships of spatial objects.

Method.

(1) Collect the set of task-relevant spatial data by an SQL query.

(2) Perform spatial-oriented induction on the collected spatial data by spatial hierarchy ascension to create high-level spatial objects until either the number of spatial objects is within the generalization threshold or the generalized concepts reach the desired generalization level. Nonspatial data entry pointers of each generalized spatial object are collected during the generalization.

(3) Retrieve nonspatial data using the nonspatial data pointers and generalize nonspatial data for each spatial object using the attribute-oriented approach [9].

(4) Output the generalized rule or the discovered relationship. □

Theorem-2. The complexity of Algorithm-2 is $O(N \log N)$.

Proof. Given N objects in the database, Step 1, the retrieval of related spatial data objects using spatial hierarchy, takes $O(N\ logN)$. The maximum number of merges for N spatial objects is $O(N)$. Step 3 is also $O(N\ logN)$. Therefore, the overall complexity is $O(N \log N)$. □

Both algorithms may be invoked by different high-level queries. Two kinds of concept hierarchy ascension can be combined into a hybrid algorithm which selects different kinds of hierarchies (spatial or nonspatial) in hierarchy ascension by the evaluation of cost functions (e.g. data volume reduction ratio, etc.) in order to achieve efficient learning and elegant learning results.

5. Discussions

Besides the two primitive generalization techniques presented above, complex spatial environments may require the techniques to be extended in many ways. Because of the limited space, only two extensions: *interleaved generalization* and *generalization on multiple thematic maps*, are discussed here.

5.1. Interleaved Generalization between Spatial and Nonspatial Data

Algorithm 1 generalizes nonspatial data before generalization on spatial data, whereas algorithm 2 proceeds in reverse order. In some cases, one may interleave the generalization between spatial and nonspatial data to achieve satisfactory results with reasonable performance. For example, to *find ten regional climate zones in B.C.*, although the threshold or the concept level for generalization is defined on spatial data, spatial generalization, relying on spatial operations such as spatial merge or spatial join, is often more expensive than relational ones. Thus a spatial-data-dominated algorithm could be costly for evaluation. It is often preferable to perform non-spatial (relational) generalization to certain level and then a high-level spatial merge/join or approximation. Further generalization may depend on the number of distinct spatial objects or appropriate concept levels. The concrete algorithm integrating the above two algorithms to achieve interleaved generalization is left as an exercise to interested readers.

5.2. Generalization on Multiple Thematic Maps

The generalization in our previous examples involves only one thematic map. In some applications, a learning task may require generalization on more than one thematic map. For example, the classification of regions in an area based on both precipitation and temperature may need two thematic maps: *precipitation* and *temperature*. Generalization can be performed by first generalizing each map based on the generalized properties, such as *wet*, *dry*, of the two maps. As a result, the regions are classified as *wet & cold*, *dry & cold*, *very wet & mild*, etc. Similar spatial generalization techniques, such as spatial merge and approximation, can be applied on the overlay of the two maps to find the regions in each class. Generalization may also derive relationships between nonspatial attributes.

6. Conclusions

We studied the requirements and necessity of knowledge discovery in spatial databases and developed two knowledge discovery techniques: *nonspatial-data-dominated generalization* and *spatial-data-dominated generalization*. Our study shows that knowledge discovery can be performed efficiently in spatial databases by extension of the techniques for knowledge discovery in relational databases. Generalization over nonspatial data is similar to generalization in relational databases; whereas generalization over spatial data is often performed by spatial region merging and/or spatial approximation. Two generalizations can be integrated and interleaved with the consideration of both the cost and the semantics of generalization.

As an emerging topic for integration of spatial database and machine learning technologies, knowledge discovery in spatial database will have wide applications in spatial knowledge discovery, spatial reasoning, spatial query optimization, the construction of multiple resolution spatial data models, etc. More investigation should be performed in this direction, especially on its integration with statistical methods, the development of customized spatial generalization operators, etc. This study demonstrates a promising direction towards knowledge discovery in spatial databases. Detailed performance studies and experiments on the algorithms presented here will be performed on relatively large spatial databases. Furthermore, this study is based on the assumptions that users have reasonably good knowledge on the spatial database structures and on what s/he wants to learn, and the system has reasonably good background knowledge (such as concept hierarchies) for generalization. More studies should be performed on knowledge discovery in spatial databases under different assumptions.

Reference References

1. W. G. Aref and H. Samet, Optimization Strategies for Spatial Query Processing, *Proc. of 17th Int. Conf. Very Large Data Bases*, Barcelona, Spain, Sept. 1991, 81–90.
2. B. P. Bruegger and J. Muller, Mechanisms of Geometric Abstraction, *Proceedings 5th Int’l Symp. on Spatial Data Handling*, Charleston, South Carolina, 1992, 123–133.
3. K. C. C. Chan and A. K. C. Wong, A Statistical Technique for Extracting Classificatory Knowledge from Databases, in G. Piatetsky-Shapiro and W. J. Frawley (eds.), *Knowledge Discovery in Databases, AAAI/MIT Press, 1991, 107–124.*
4. S. K. Chang and S. H. Liu, Picture Indexing and Abstraction Techniques for Picture Databases, *IEEE Trans. Pattern Analysis and Machine Intelligence*, **6**(4), July 1984, 475–483.
5. T. G. Dietterich and R. S. Michalski, A Comparative Review of Selected Methods for Learning from Examples, *in Michalski et. al. (eds.), Machine Learning: An Artificial Intelligence Approach, Vol. 1, Morgan Kaufmann, 1983, 41–82.*
6. D. Fisher and P. Langley, Approaches to Conceptual Clustering, *Proc. 9th Int. Joint Conf. AI*, Los Angeles, CA, Aug. 1985, 691–697.

7. W. J. Frawley, G. Piatetsky-Shapiro and C. J. Matheus, Knowledge Discovery in Databases: An Overview, in G. Piatetsky-Shapiro and W. J. Frawley (eds.), *Knowledge Discovery in Databases, AAAI/MIT Press, 1991, 1–27.*

8. A. Guttman, R-Tree: A Dynamic Index structure for Spatial Searching, *Proc. 1984 ACM-SIGMOD Conf. Management of Data*, Boston, MA, June 1984, 47–57.

9. J. Han, Y. Cai and N. Cercone, Knowledge Discovery in Databases: An Attribute-Oriented Approach, *Proc. 18th Int'l Conf. on Very Large Data Bases*, Vacouver, Canada, Aug. 1992, 547–559.

10. K. A. Kaufman, R. S. Michalski and L. Kerschberg, Mining for Knowledge in Databases: Goals and General Description of the INLEN System, in G. Piatetsky-Shapiro and W. J. Frawley (eds.), *Knowledge Discovery in Databases, AAAI/MIT Press, 1991, 449–462.*

11. W. Lu and J. Han, Distance-Associated Join Indice for Spatial Range Search, *Proc. 8th Int. Conf. Data Engineering*, Phoenix, AZ, Feb., 1992, 284-292.

12. M. V. Manago and Y. Kodratoff, Noise and Knowledge Acquisition, *Proc. 10th Int. Joint Conf. Artificial Intelligence*, Milan, Italy, 1987, 348–354.

13. R. L. Read, D. S. Fussell and A. Silberschatz, A Multi-Resolution Relational Data Model, *Proc. 18 Very Large Data Base*, Vancouver, Aug. 1992, 139-150.

14. H. Samet, *The Design and Analysis of Spatial Data Structures*, Addison-Wesley, 1990.

15. A. K. C. Wong, A Statistical Technique for Extracting Classificatory Knowledge from Databases, in G. Piatetsky-Shapiro and W. J. Frawley (eds.), *Knowledge Discovery in Databases, AAAI/MIT Press, 1991, 107–124.*

16. J. Zytkow and J. Baker, Interactive Mining of Regularities in Databases, in G. Piatetsky-Shapiro and W. J. Frawley (eds.), *Knowledge Discovery in Databases, AAAI/MIT Press, 1991, 31–54.*