CIT. Journal of Computing and Information Technology, Vol. 27, No. 3, September 2019, 29–43
doi: 10.20532/cit.2019.1004651

C4.5 Decision Tree Algorithm for Spatial Data, Alternatives and Performances

Sihem Oujdi[1], Hafida Belbachir[1] and Faouzi Boufares[2]

[1]Oran University of Science and Technology – Mohamed Boudiaf, Oran, Algeria
[2]University Sorbonne Paris Nord (Paris 13), Paris, France

Using data mining techniques on spatial data is more complex than on classical data. To be able to extract useful patterns, the spatial data mining algorithms must deal with the representation of data as stack of thematic layers and consider, in addition to the object of interest itself, its neighbors linked through implicit spatial relations. The application of the classification by decision trees combined with the visualization tools represents a convenient decision support tool for spatial data analysis. The purpose of this paper is to provide and evaluate an alternative spatial classification algorithm that supports the thematic-layered data organization, by the adaptation of the C4.5 decision tree algorithm to spatial data, named S-C4.5, inspired by the SCART and spatial ID3 algorithms and the adoption of the Spatial Join Index. Our work concerns both data organization and the algorithm adaptation. Decision tree construction was experimented on traffic accident dataset and benchmarked on both computation time and memory consumption according to different experimentations: study of phenomenon by a single and then by multiple other phenomena, including one or more spatial relations. Different approaches used show compromised and balanced results between memory usage and computation time.

ACM CCS (2012) Classification: Information systems → Information systems applications → Data mining

Information systems → Information systems applications → Spatial-temporal systems → Geographic information systems

Computing methodologies → Machine learning → Learning paradigms → Supervised learning → Supervised learning by classification

Keywords: spatial data mining, classification, decision tree, C4.5 algorithm, experimentation

1. Introduction

Thanks to technological advances, today we are living in the digital era where we can assist an explosion of the data quantity of which a major part is geo-referenced, and so, with a spatial nature that obeys the Tobler's first geography law [1] "*Everything is related to everything else but nearby things are more related than distant things*", introducing the neighborhood notion and connecting these data to each other. Analyzing spatial data disregarding this property is definitely incorrect [2]. Such a mass of data has a value only if we can extract from it useful knowledge, particularly in the fields of Business Intelligence (BI), Decision Support System (DSS), large-scale targeting and marketing operations and political campaigns.

The spatial dimension of geo-referenced data adds a significant complexity to the data mining tasks. Spatial objects are characterized by a geometrical representation and relative positioning which implicitly define both spatial relations and properties. In addition, spatial data are arranged in a set of spatially linked thematic layers representing discrete or continuous features.

Traditional data mining techniques cannot handle spatial data, hence the need to adapt new techniques in the area of spatial data mining. Enabling the extraction of interesting patterns from large datasets, spatial data mining fulfills the analysis needs of many geomatic applications and allows taking advantage of the grow-

ing availability of spatial data. However, spatial data mining is widely derived from conventional data mining techniques for spatial classification, which could be used to explain or to predict a phenomenon by analyzing the properties of the geographical environment, *e.g.* explaining the occurrences of accidents according to road conditions or the urban environment. The use of spatial classification by decision trees represents a helpful tool for decision support and analysis, operations research and phenomenon prediction.

This paper proposes a C4.5 [3] based spatial decision tree algorithm to construct a classification model from a spatial dataset that can be organized in multi-thematic layers and may contain both discrete and continuous features. Our contributions concern the adaptation of information gain at the C4.5 algorithm level, based on works [4], [5] and the data structure preparation using the Spatial Join Index (SJI) introduced in [6] (not to be confused with the classical spatial join algorithms). We have performed different classification experimentations on a road safety dataset in order to evaluate and compare the performances of the proposed algorithm alternatives.

Including introduction and conclusion, we structured this paper into five sections. Presentation of spatial data mining related works is in Section 2. Development of the proposed approaches, including their experimentations and results, is shown in Section 3. A comparative study and discussion of results is presented in Section 4.

2. Related Work

In order to extract useful knowledge, spatial data mining algorithms have to consider the neighbors of objects, which makes the discovery processes such as classification for spatial data more complex than those for non-spatial data [7].

Spatial relations, both topological, direction and metric are commonly used in spatial queries and analyses. They are usually stored implicitly in spatial databases and, therefore, require to be computed. These relations translate the influence of the neighbors of objects, which can be classified into two types: intra-theme, *e.g.* the spatial autocorrelation of geographical phenomena measurements (the temperature of two nearby places is close) and inter-theme, *e.g.* the influence of road traffic on the phenomenon of pollution.

We adopted the same categorization of spatial data mining techniques as presented in [8]. Hence, according to the consideration way of spatial relations (intra-theme or inter-theme), we distinguish mono-thematic and multi-thematic spatial data mining approaches families.

2.1. Mono-Thematic Approaches

These approaches are often related to data analysis and statistics. In the case of a single theme, the data are described with the same variables and so are comparable. This allows to include a contiguity parameter into a weight or model variables according to the values of the neighborhood. Below, we summarize the three most common mono-thematic approaches.

- Analysis of localizations without attributes: based only on localizations, these approaches tend to reveal the concentrations or trends by exploring a set of localizations (points set). Among the major works, we cite: "trend analysis by the method of density" [9], and "clustering" [10].
- Analysis of localizations provided with numerical measures: these analyses aim to characterize the spatial variation of measures taken on a spatial domain, often covering space by a surface cutting. It is frequently a single numeric attribute. Among the major works, we cite: "overall and local spatial autocorrelation" [11] and "trend analysis by linear regression" [7].
- Analysis of localizations with provided categories: these analyses focus on the characteristic properties extending the neighborhood or on the simultaneous presence of categories in space. In this case, localizations are assumed to be described by categorical attributes. Among the major works, we cite: "co-localization" [11]–[13], and "characterization" [14].

In the mono-thematic spatial data mining, the space is the object of the analysis with few attributes, usually a measure or a single category, while the spatial databases and the majority

of Geographic Information Systems (GIS) organize the data in thematic layers, each with a description or its own scheme. Mono-thematic methods are not compatible with this data organization, and therefore are unable to reveal the hidden inter-thematic relations.

2.2. Multi-Thematic Approaches

Unlike the mono-thematic approaches, the purpose of the multi-thematic data mining is to consider, in addition to the description of the object by its own attributes, its neighborhood relation as well as the description of neighboring objects. These approaches are based on spatial predicates which are interpreted as properties to be considered in the model to induce. These methods distinguish the main theme of the analysis (named the target theme) and then explore the other themes (named phenomena) that may influence it. Supervised classification and association rules are often used in multi-thematic approaches.

Among the pioneers of spatial data mining, Koperski and Han [15], [16], based on the works in [17], have defined association rules and classification methods involving explicitly thematic layers. Malerba and Lisi [18], [19] have adapted the works in [15] by the application of the Inductive Logic Programming (ILP) based multi-relational data mining on multi-thematic spatial data.

Ester *et al.* [7] have proposed a classification method based on ID3 algorithm [20] that considers the links and neighbors types, but the notion of the theme is not explicit. Comparatively, the classification method of Koperski described in [21] considers reference themes and precise neighbors relations.

The authors in [4] developed a spatial decision tree algorithm named SCART (Spatial Classification and Regression Trees) by the extension of the CART method, one of the commonly used systems for induction of decision trees proposed in [22]. SCART considers both the organization of geographical data in thematic layers, and their spatial relations. SCART uses Spatial Join Index table (SJI) [6] to calculate the spatial relations between objects.

The study [23] extended the ID3 algorithm [20] to support spatial datasets, taking into account spatial objects and their relations to their neighbors. Generation of the tree with this algorithm is made by selecting the best layer for the separation of a dataset into as pure as possible small partitions, meaning that all objects in partitions belong to the same class. This algorithm uses an adapted version of the information gain enabling the choice of a layer as a splitting layer.

The authors in [5] discuss another decision tree from spatial data for the discrete characteristics represented with points, lines and polygons. The proposed method was based on non-spatial properties of the classified objects, predicates and functions that describe the spatial relation between the objects, in addition to other features located in the spatial proximity of the classified objects.

2.3. Limitations of Existing Research

In the first category, named mono-thematic approaches, the analysis of localizations without attributes is insufficient in the analysis of spatial databases. The analysis of localizations provided with numerical measures or categories is limited because it is mono-attribute. Generally, mono-thematic approaches consider only intra-theme relations between objects, excluding the spatial relations that may exist between objects of different themes.

In the second category, named multi-thematic approaches, [7], [15] use (binary) spatial predicates rather than weighted spatial relations such as distance. Methods in [15], [16] begin by generalizing the data before applying the classification algorithm. Although this is seen as an optimization technique, it can lead to the loss of information. In addition, this method involves transformation of the data into predicates that prevents the use of existing classification algorithms and this rewriting of the base as predicates has a cost to be added in the building model costs. In [18], [19], PLI based methods have the same disadvantages as the previous ones. Moreover, unlike the previous ones, they do not consider scaling and do not propose any indexing or optimization technique in this sense.

The SCART algorithm defined in [4] constructs, like CART, binary decision trees where branches are generated on a single pair of attri-

bute-value rather than on all values of the selected attribute. However, the drawback of this restriction is that the generated tree may be less interpretable, because of multiple splits occurring on the same attribute at adjacent levels, so, it may be a bad binary division on an attribute that has a good multi-way split during the generation of decision trees using CART [24]. As SCART is based on CART, it may be subject to the same drawback.

By using the ID3 algorithm, the works in [23] are limited to discrete characteristics.

3. Proposed Approach

In order to provide an efficient and convenient spatial data analysis tool, our contribution to spatial data mining fits the context of classification, a commonly used method to find mining rules from large databases, by decision trees, a frequent method used for the classification because of its simple hierarchical structure for the user understanding and decision making. Our contribution complies with the following considerations:

1. taking into account the organization of spatial data as stack of thematic layers;
2. support of large spatial datasets with acceptable performance;
3. flexible solution, allowing the application of different models.

In addition to the consideration of spatial objects themselves, taking into account their relations to neighbors and also the neighbors objects description, places our vision of spatial data mining in the multi-thematic approaches category, as described in the above related works section. This allows us to comply with the first consideration.

To satisfy the second consideration about performance, based on the conclusion of a decision trees comparative study presented in [25], which clearly demonstrates the enhancements in execution time and accuracy of C4.5, we propose to use this algorithm for the decision trees construction. In addition to the limits presented in the end of the related works section concerning the choice of CART an ID3 algorithms, the use of C4.5 allows us to the following:

- generation of *n*-ary trees;
- ability to handle continuous data;
- ability to manage attributes with missing values;
- possibility to post-prune the generated tree.

In order to develop a flexible solution, we adopted the use of the Spatial Join Index (SJI) presented in [6]. The use of the SJI allows also the optimization of performance through the application of different models.

The main aspect of spatial data mining is the consideration of spatial relations between the objects. The SJI was proposed in [6] as an extension of the Join Indices developed in [26] to exactly calculate these spatial relations between the collections of objects from different thematic layers and to improve performance of the complex operations in a Database Management System (DBMS). The result of the SJI is a new table that contains the pre-calculated results of spatial relations between different objects. Figure 1 shows the basic scheme of SJI (ID_X, SR, ID_Y), where ID_X and ID_Y reference respectively their matching object and SR their computed spatial relation, which can be topologic, metric or direction, *e.g.*, in the case of metric relation, SR will contain the exact distance value. SJI can be handled in the same way as other tables and manipulated using the standardized SQL query language.

Introducing this new intermediate table will increase the required space according to the number of included spatial relations. To reduce additional memory usage generated by the SJI, the concerned objects will be referenced through their IDs. The pre-computed spatial relations can be quantitative (discrete/continuous) variables and will be stored as-is or qualitative variables and will be stored using data coding techniques. Memory consumption monitoring in different experiments is investigated below:

In this paper we focus on the construction of decision trees based on the adapted C4.5 algorithm using different approaches and conducting performance measures. The ensembles of decision trees, *e.g.* bagging, boosting, *etc.* are out of the scope of this work.

In order to be able to handle spatial data, we propose the S-C4.5 alternative, which is an adaptation of C4.5 according to two modifica-

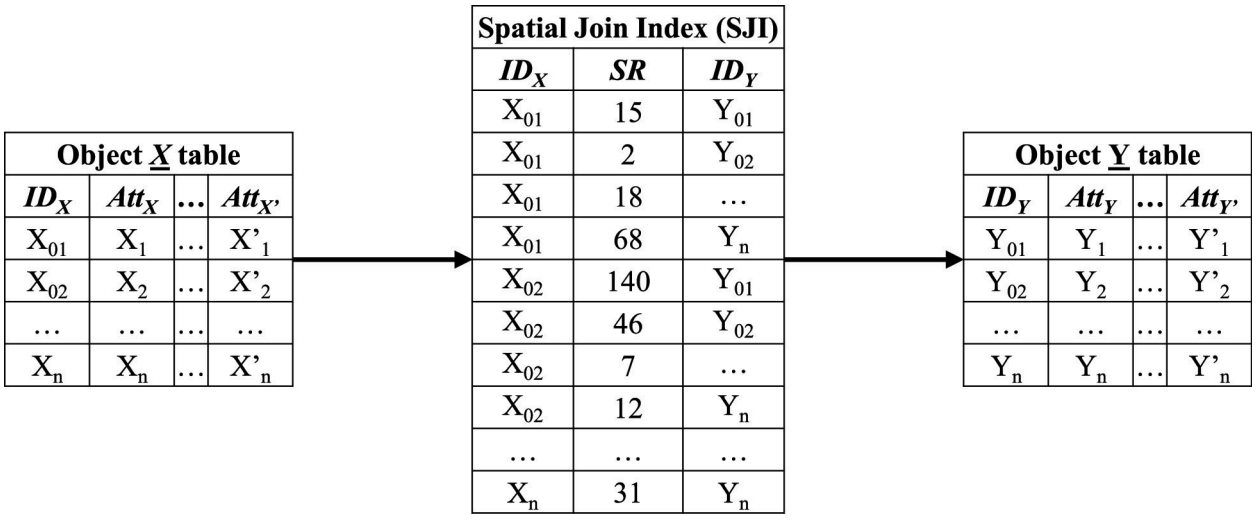


Figure 1. General structure of Spatial Join Index (SJI).

tions, the first is at the data organization level and the second is at the algorithm level.

1. *Data Organization Modification*. This modification is necessary in order to consider the spatial nature of data, which consists of including among the analysis data, the pre-calculated spatial relations in the Spatial Join Index (SJI).
2. *Algorithm Modification*. This modification is necessary given the multi-relational nature of the spatial data, which are generally organized in several tables representing individually a spatial phenomenon. The change is made at the informational gain calculation level. For this purpose, depending on the choice of the target phenomenon to be studied, if the attribute for which we want to calculate the informational gain is in the same table as the target attribute, then the gain is calculated directly, otherwise a join is necessary before it can be calculated.

In Algorithm, 1 we describe the recursive main function of the S-C4.5 algorithm.

Steps 2–7 check the stop criteria: if there are no neighbors attributes or if all training tuples are in the same class, then a leaf labeled with the majority class is returned.

Steps 8–20 evaluate the measures of all candidate features and splitting thresholds to generate intermediate nodes. Function *bestGainCalculation* in step 9 returns the attribute maximizing the informational gain after calculation and operates according to our modification described above. Steps 10 to 20 are the same as the classical C4.5 algorithm with only one difference: inclusion of the spatial relations *SR* pre-calculated in the SJI.

Dataset and test environment. For the purpose of all our experiments, we used a detailed database of road safety, of the circumstances and injuries of road accidents in Great Britain from 1979 to 2001 [27], the types (including model) of the vehicles involved and the consequences. The statistics refer only to the accidents on public roads involving bodily injuries, which were reported to the police and then recorded. Our dataset is detailed in Table 1.

Our entire tests environment is on a machine with Intel Quad Core Q9400 2.4 GHz CPU and 12GB DDR3 memory under Microsoft Windows 10 Pro 64-bit edition using Oracle 10g DBMS.

Table 1. Description of the experimentations database.

Table Name	Attributes Number	Tuples Number
Accidents	13	1 048 576
Vehicles	24	1 048 576
Climate	04	1 048 576
Routes	12	1 048 576
Victims	14	1 048 576
Areas	03	1 048 576

Algorithm 1. Description of the main function of the proposed S-C4.5 algorithm.

Input:

- A spatial dataset of training tuples *SD* and their associated class labels constructed from a set of layers using spatial relations.
- A target layer $T_L \subset SD$ including a target attribute $T_A \in T_L$.
- A set of neighbors (explanatory) layers $N_L \subseteq SD$.
- A set of neighbors (explanatory) attributes N_A, where $\exists\, N_A \subseteq N_L$ and $\exists\, N_A \subset T_L$.
- Set of spatial relations *SR* pre-calculated using the SJI method.

Output:

- A spatial decision tree.

Method:

```
1.  function S-C4.5(SD, T_A, N_A, SR)
2.      for each table T of SD do
3.          if N_A = NULL then → (Terminal node)
4.              return a node N with the most represented value for T_A
5.          else
6.              if all examples have the same value for T_A then → (Terminal node)
7.                  return a node N with this value of T_A
8.              else → (Intermediate node)
9.                  Selected attribute S_A = bestGainCalculation(N_A, SR, T_A)
10.                 Remaining neighbors attributes R_NA = deleteFromList(N_A, S_A)
11.                 New node N_N = node labeled with S_A
12.                 for each value V_SA of S_A do
13.                     Filtered sample F_S = filterSamplesWithValueForAttribute(T, S_A, V_SA)
14.                     N_N.son(V_SA) = S-C4.5(F_S, T_A, R_NA, SR)
15.                 end for
16.                 return N_N
17.             end if
18.         end if
19.     end for
20. end function
```

We applied our S-C4.5 algorithm on two major study cases:

- case 1: study of a phenomenon by a single other phenomenon;
- case 2: study of a phenomenon by multiple other phenomena.

3.1. Study of a Phenomenon by a Single other Phenomenon

This study consists of taking two tables as inputs: a target table (target phenomenon), and a neighbor table (neighbor phenomenon), without missing a Spatial Join Index that links the first phenomenon with the second one by including one or more spatial relations.

The target table is automatically identified at the selection of the target attribute that represents a phenomenon for which the spatial study will be performed. We can simply define it as the table that contains the target attribute and, therefore, it can be only a single and unique target table.

The neighbor table defines the considered phenomenon to explain the target phenomenon.

The Spatial Join Index contains the pre-calculated results of spatial relations between different objects (tables). The SJI is necessary to include the spatial character of these data, so,

we must consider it as an intermediary table to enable the join between the target and the neighbor table.

We divide our study into two cases: the first case considers only one spatial relation between the two studied phenomena (the target table, and the neighbor table), in the second case, we consider multiple spatial relations between the two phenomena.

3.1.1. Study of a Phenomenon by a Single other Phenomenon Including a Single Spatial Relation

In this study case, the data structure is as in Figure 2:

N.B.: *T_xSRN_y represents the pre-calculated spatial relation SR between the target tuple with ID=T_x and the neighbor tuple with ID=N_y from their spatial attributes (latitude, longitude). These relations have discretized values, for example, for the distance, according to the situation; the values are discretized to 3 values of near, medium and far.*

For real world data example, you can refer to Figure 5.

In this study case, we adapted the C4.5 algorithm according to two approaches:

- join on the fly;
- joins materialization.

The join on the fly method consists of joining the target and the neighbor tables through the SJI when it is necessary, that means, when the attribute for which we want to calculate the informational gain is outside the target table.

On the other side, the joins materialization consists of materializing in advance the join of the tables involved: target table, neighbor table, Spatial Join Index, and considering the result of the join as a single table, then applying the classical algorithm of data mining to this table.

3.1.2. Study of a Phenomenon by a Single other Phenomenon including multiple Spatial Relations

Unlike the first study, this time, the SJI contains more than one spatial relation. This allows considering several spatial relations between the objects, which enhances the obtained results. The structure is described in Figure 3.

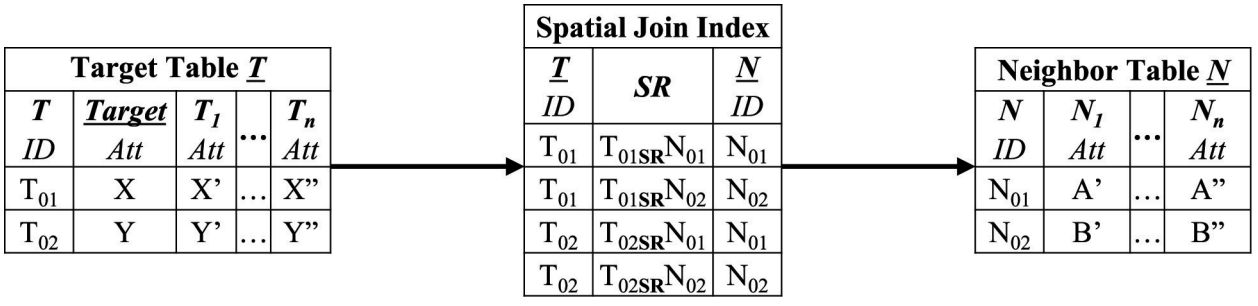


Figure 2. Structure of the study of a phenomenon by a single other phenomenon including a single spatial relation.

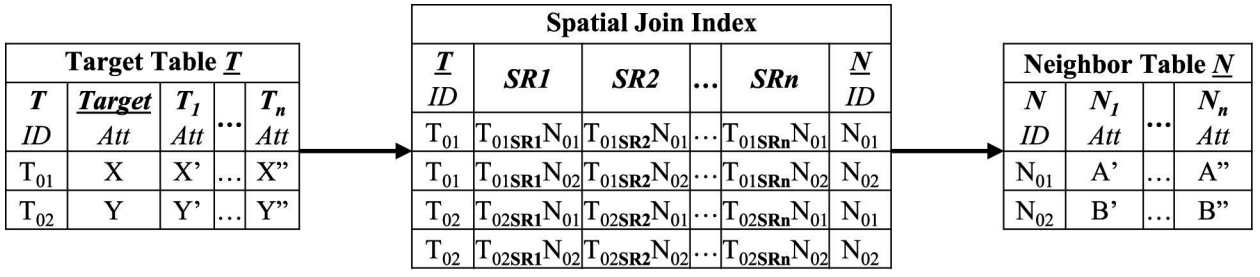


Figure 3. Structure of the study of a phenomenon by a single other phenomenon including multiple spatial relations.

We also applied this study to the two methods, namely, the method of the join on the fly and that of the joins materialization.

Experimentations. In the study case of a phenomenon by a single other phenomenon, including one or more spatial relations, we define the table *Accident* as a target table, and the table *Area* as a neighbor table. The purpose of this study on this dataset is to link the consequences (represented by the severity attribute) of accidents to the areas, which could help to make right decisions and intervene effectively in a certain area, thanks to predictions of the kinds of accidents that can occur depending on the location.

Results. For the purpose of comparison with other works, we have tested the proposed S-C4.5 algorithm with multiple spatial relations (distance and direction in our case) according to two different methods: join on the fly and joins materialization, on a spatial dataset.

We compared our works with the existing SCART algorithm [4].

Comparison of performances concerns the duration of analysis and the memory consumption during the treatments. The results obtained for this study case are detailed in Table 2.

N.B.: *In the joins materialization, 'Phase I' corresponds to the materialization of joins; 'Phase II' corresponds to the execution of the algorithm*.

At first, according to the results, we can deduce the same conclusions as those made for the case with a single included spatial relation:

- the joins materialization method gives better results in terms of execution time compared to the join on the fly method;
- the join on the fly method consumes less memory space compared to the joins materialization method.

Secondly, we note that the S-C4.5 algorithm uses a little bit less memory than the SCART algorithm, for both approaches, the join on the fly and the joins materialization; furthermore, it requires significantly less execution time than SCART.

The consideration of an additional spatial relation is the equivalent of including additional attributes in the analysis. We can see that in all methods of the study of a phenomenon by a single other phenomenon including multiple spatial relations, the execution time and memory consumption are slightly higher, compared to the study with a single spatial relation. So, the results confirm that the inclusion of multiple spatial relations will marginally decrease the performances. On the other hand, it will enhance the analysis content, because spatial relations are the information that translates an essential property and characteristic of real world, which is the influence of the neighborhood. This information is usually exploited in the spatial requests and analyses, the more spatial relations are included, the more data mining is spatial.

To get closer to reality and to be able to make the analysis more relevant, making spatial data mining by studying a phenomenon by only a single other phenomenon is a limitation in itself, despite the consideration of multiple spatial relations. However, in real world, a phenomenon is

Table 2. Results of the first contribution including multiple spatial relations.

Algorithm	Joins materialization			Querying on the fly different tables	
	Duration		Memory space	Duration	Memory space
S-C45	Step1 20.460 s	Step2 54 min 01.020 s	6 438 MB	96 min 26.680 s	1 392 MB
	Total 13 min 37.075 s				
SCART	Step1 20 min 05.496 s	Step2 71 min 44.912 s	6 534 MB	126 min 15.660 s	1 668 MB
	Total 18 min 4.310 s				

determined by several other phenomena, hence the need to be able to make a study for a certain phenomenon by multiple other phenomena.

3.2. Study of a Phenomenon by Multiple other Phenomena

Figure 4 illustrates general structure of the study of a phenomenon by multiple other phenomena.

Based on this structure of the study of a phenomenon by multiple other phenomena, we adapted the S-C4.5 algorithm to include more than one neighbor according to four methods: join on the fly, joins materialization, joins semi-materialization and imbricated materialization.

Experimentations were conducted on the same database presented in the previous experimentations whose details are illustrated in Table 1. The *Accident* table is considered as the target table, the *Climate* and *Road* tables are considered as neighbors tables. For spatial relations, we used the distance and direction. Considering these tables (phenomena or themes), we will be able to determine and predict at certain percentage the risk and consequences of accidents that can occur depending on the road type and weather conditions.

Figure 5 illustrates a real-world sample of the structure of this study case.

3.2.1. Method 1: Join on the Fly

This method consists of performing the join when the attribute for which we want to calculate the gain is out of the target table, going through the Spatial Join Index. Based on the structure described in Figure 4, we obtain pairs of attributes following this form: [Target attribute, X attribute], where X is the attribute for which we want to calculate the informational gain, and which is situated out of the target table.

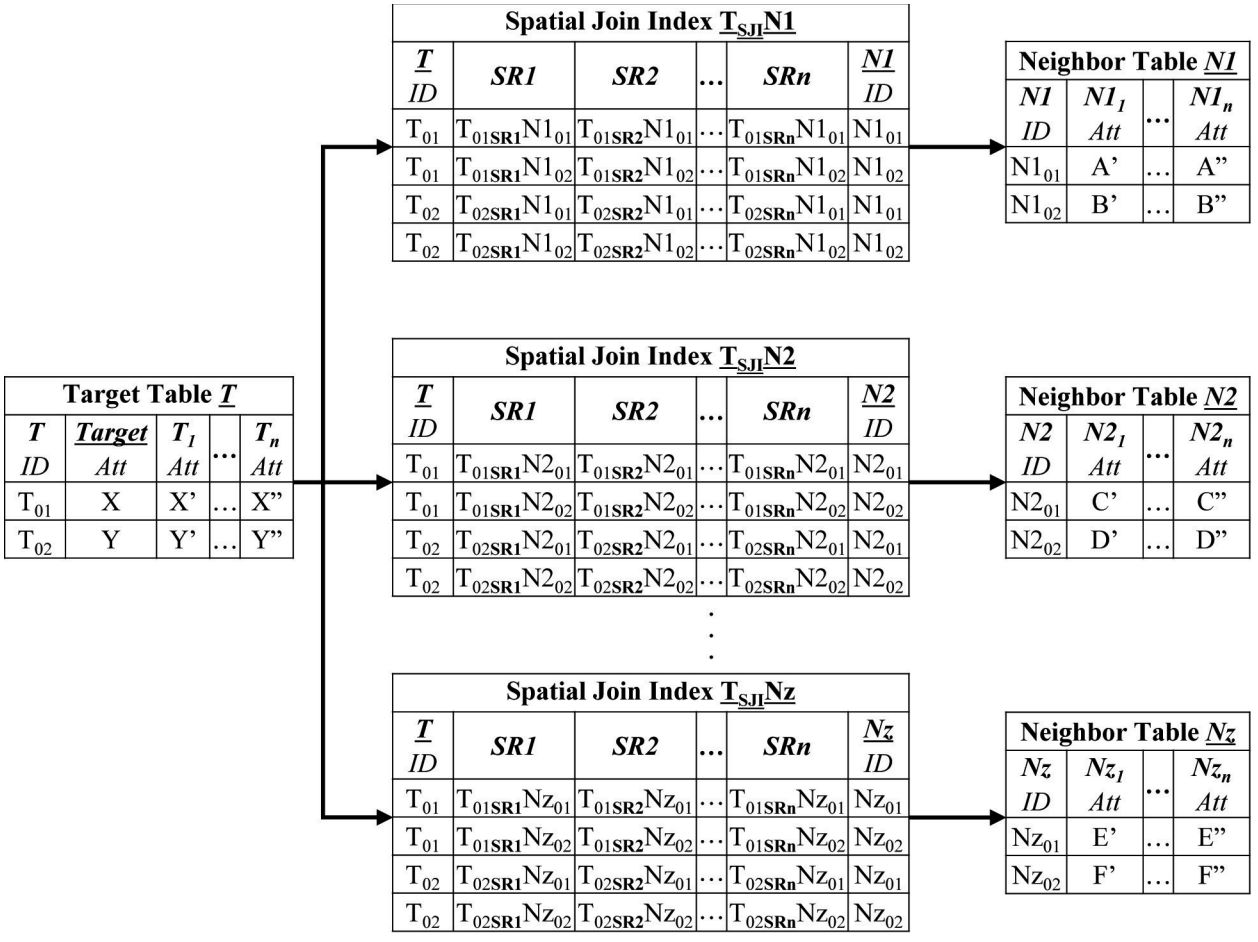
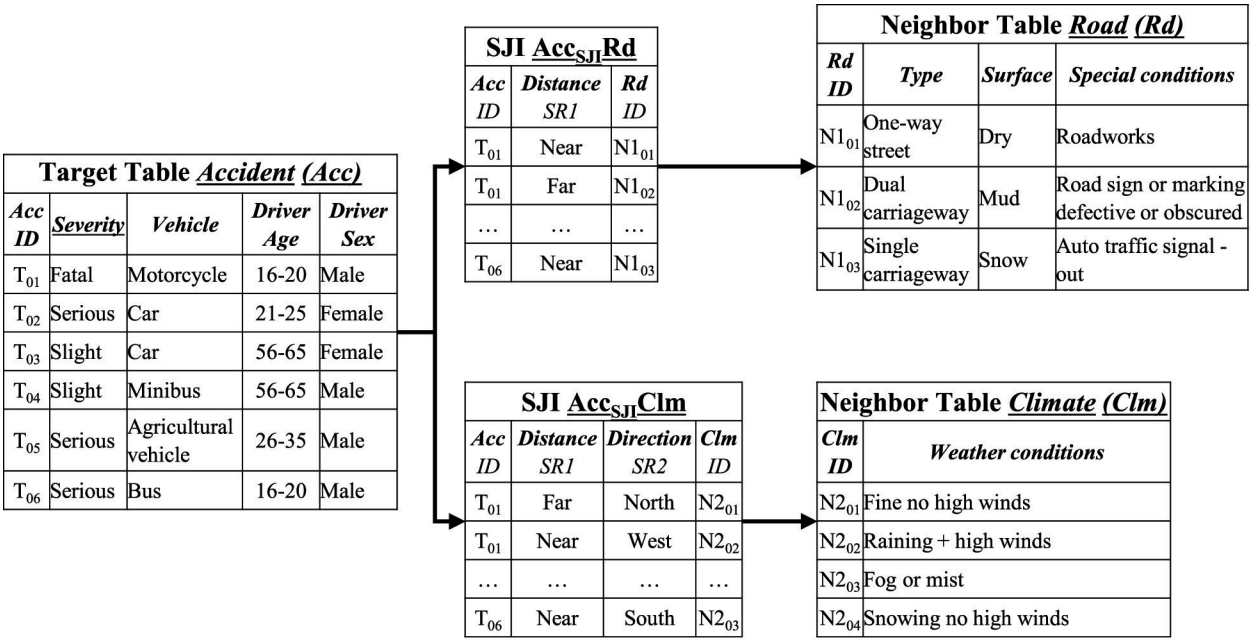


Figure 4. General structure of the study of a phenomenon by multiple other phenomena.



Target Table *Accident (Acc)*

Acc ID	*Severity*	*Vehicle*	*Driver Age*	*Driver Sex*
T_{01}	Fatal	Motorcycle	16-20	Male
T_{02}	Serious	Car	21-25	Female
T_{03}	Slight	Car	56-65	Female
T_{04}	Slight	Minibus	56-65	Male
T_{05}	Serious	Agricultural vehicle	26-35	Male
T_{06}	Serious	Bus	16-20	Male

SJI $Acc_{SJI}Rd$

Acc ID	*Distance SR1*	*Rd ID*
T_{01}	Near	$N1_{01}$
T_{01}	Far	$N1_{02}$
…	…	…
T_{06}	Near	$N1_{03}$

Neighbor Table *Road (Rd)*

Rd ID	*Type*	*Surface*	*Special conditions*
$N1_{01}$	One-way street	Dry	Roadworks
$N1_{02}$	Dual carriageway	Mud	Road sign or marking defective or obscured
$N1_{03}$	Single carriageway	Snow	Auto traffic signal - out

SJI $Acc_{SJI}Clm$

Acc ID	*Distance SR1*	*Direction SR2*	*Clm ID*
T_{01}	Far	North	$N2_{01}$
T_{01}	Near	West	$N2_{02}$
…	…	…	…
T_{06}	Near	South	$N2_{03}$

Neighbor Table *Climate (Clm)*

Clm ID	*Weather conditions*
$N2_{01}$	Fine no high winds
$N2_{02}$	Raining + high winds
$N2_{03}$	Fog or mist
$N2_{04}$	Snowing no high winds

Figure 5. Study of accidents consequences by the roads and the weather conditions.

Results. Table 3 illustrates the results obtained for the calculation of duration and consumed memory space.

Table 3. Results of method 1: join on the fly.

Duration	Memory Space
115 min 02.335 s	1 644 MB

Although this method simplifies the inclusion of more than one phenomenon, intensive use of the joins operations when the concerned attribute by the calculation of the informational gain is out of the target table, despite lower consumption of memory space, gives a weak execution time performance.

3.2.2. Method 2: Joins Materialization

Concerning the method of joins materialization, which has the advantage of bringing back the spatial data mining to classical data mining, by the materialization of all the joins operations of all tables to a single global table, called materialized table.

Implementation of the S-C4.5 algorithm with the joins materialization method generates a materialized table with incoherent data, even more when spatial relationships overlap between themselves, and this will cause incoherent data analysis. Figure 6 illustrates this problem.

The main problem in the structure exposed in Figure 6 occurs when we consider the same spatial relation with multiple neighbors. When calculating the informational gain of a specific spatial relation, it is impossible to define for which explaining phenomenon (neighbor) this gain is brought, which is incoherent and falsifies the results. The only possibility is to consider distinct spatial relations for each neighbor. For example, if we consider the distance as spatial relation for the first neighbor, it will not be used for the other neighbors, which will significantly limit the analysis.

However, we can find alternatives to overcome this difficulty; we proposed an alternative illustrated in Figure 7. This solution requires taking back the same spatial relation by precising with which neighbor it is calculated. This will generate a huge table, especially when we work on spatial databases.

Results. Table 4 illustrates the duration of calculation and the memory space needed by method 2: joins materialization, according to the alternative structure detailed in Figure 7 and tested with the dataset presented above.

Target table *T*				**Spatial Join Index**				**Neighbor *N1***				**Neighbor *Nz***		
Target *Att*	T_1 *Att*	…	T_n *Att*	*SR1*	*SR2*	…	*SRn*	$N1_1$ *Att*	…	$N1_n$ *Att*	…	Nz_1 *Att*	…	Nz_n *Att*
X	X'	…	X"	$X_{SR1}A$	NULL	…	$X_{SRn}A$	A'	…	A"	…	NULL	NULL	NULL
X	X'	…	X"	$X_{SR1}B$	NULL	…	$X_{SRn}B$	B'	…	B"	…	NULL	NULL	NULL
Y	Y'	…	Y"	$Y_{SR1}A$	NULL	…	$Y_{SRn}A$	A'	…	A"	…	NULL	NULL	NULL
Y	Y'	…	Y"	$Y_{SR1}B$	NULL	…	$Y_{SRn}B$	B'	…	B"	…	NULL	NULL	NULL
X	X'	…	X"	NULL	$X_{SR2}C$	…	$X_{SRn}C$	NULL	NULL	NULL	…	C'	…	C"
X	X'	…	X"	NULL	$X_{SR2}D$	…	$X_{SRn}D$	NULL	NULL	NULL	…	D'	…	D"
X	X'	…	X"	NULL	$X_{SR2}E$	…	$X_{SRn}E$	NULL	NULL	NULL	…	E'	…	E"
Y	Y'	…	Y"	NULL	$Y_{SR2}C$	…	$Y_{SRn}C$	NULL	NULL	NULL	…	C'	…	C"
Y	Y'	…	Y"	NULL	$Y_{SR2}D$	…	$Y_{SRn}D$	NULL	NULL	NULL	…	D'	…	D"
Y	Y'	…	Y"	NULL	$Y_{SR2}E$	…	$Y_{SRn}E$	NULL	NULL	NULL	…	E'	…	E"

Figure 6. First structure of multi-thematic joins materialization.

Target table *T*				**Neighbor *N1***							**Neighbor *Nz***					
Target *Att*	T_1 *Att*	…	T_n *Att*	***Spatial Relations***			***Attributes***			…	***Spatial Relations***			***Attributes***		
				SR_1N1	…	SR_nN1	$N1_1$ *Att*	…	$N1_n$ *Att*	…	SR_2Nz	…	SR_nNz	Nz_1 *Att*	…	Nz_n *Att*
X	X'	…	X"	$X_{SR1}A$	…	$X_{SRn}A$	A'	…	A"	…	NULL	…	NULL	NULL	…	NULL
X	X'	…	X"	$X_{SR1}B$	…	$X_{SRn}B$	B'	…	B"	…	NULL	…	NULL	NULL	…	NULL
Y	Y'	…	Y"	$Y_{SR1}A$	…	$Y_{SRn}A$	A'	…	A"	…	NULL	…	NULL	NULL	…	NULL
Y	Y'	…	Y"	$Y_{SR1}B$	…	$Y_{SRn}B$	B'	…	B"	…	NULL	…	NULL	NULL	…	NULL
X	X'	…	X"	NULL	…	NULL	NULL	…	NULL	…	$X_{SR2}C$	…	$X_{SRn}C$	C'	…	C"
X	X'	…	X"	NULL	…	NULL	NULL	…	NULL	…	$X_{SR2}D$	…	$X_{SRn}D$	D'	…	D"
X	X'	…	X"	NULL	…	NULL	NULL	…	NULL	…	$X_{SR2}E$	…	$X_{SRn}E$	E'	…	E"
Y	Y'	…	Y"	NULL	…	NULL	NULL	…	NULL	…	$Y_{SR2}C$	…	$Y_{SRn}C$	C'	…	C"
Y	Y'	…	Y"	NULL	…	NULL	NULL	…	NULL	…	$Y_{SR2}D$	…	$Y_{SRn}D$	D'	…	D"
Y	Y'	…	Y"	NULL	…	NULL	NULL	…	NULL	…	$Y_{SR2}E$	…	$Y_{SRn}E$	E'	…	E"

Figure 7. Alternative structure of multi-thematic joins materialization.

Table 4. Results of method 2: joins materialization.

Duration	**Memory Space**
71 min 15.320 s	7 280 MB

The materialization of joins considerably reduces the needed duration of execution, compared to method 1: join on the fly, but it requires much larger memory space.

Given the implementation constraints related to the joins materialization, especially those of the management of data duplication, we have proposed an alternative approach, which is a hybrid method, situated between joins materialization and join on the fly methods. We call it joins semi-materialization.

3.2.3. Method 3: Joins Semi-Materialization

This method consists of separately joining the target table with each neighbor table through its Spatial Join Index. Then, the best attribute (with the best informational gain) is determined for each triplet (Target Table, Spatial Join Index, and Neighbor Table) which is the dividing element to generate the decision tree. We precise that, in this method, the calculation of the informational gain for the included spatial relations is immediately identified referring to the appropriate neighbor. This allows getting distinct and common spatial relations for all neighbors' tables at the same time. Figure 8 illustrates the joins semi-materialization method.

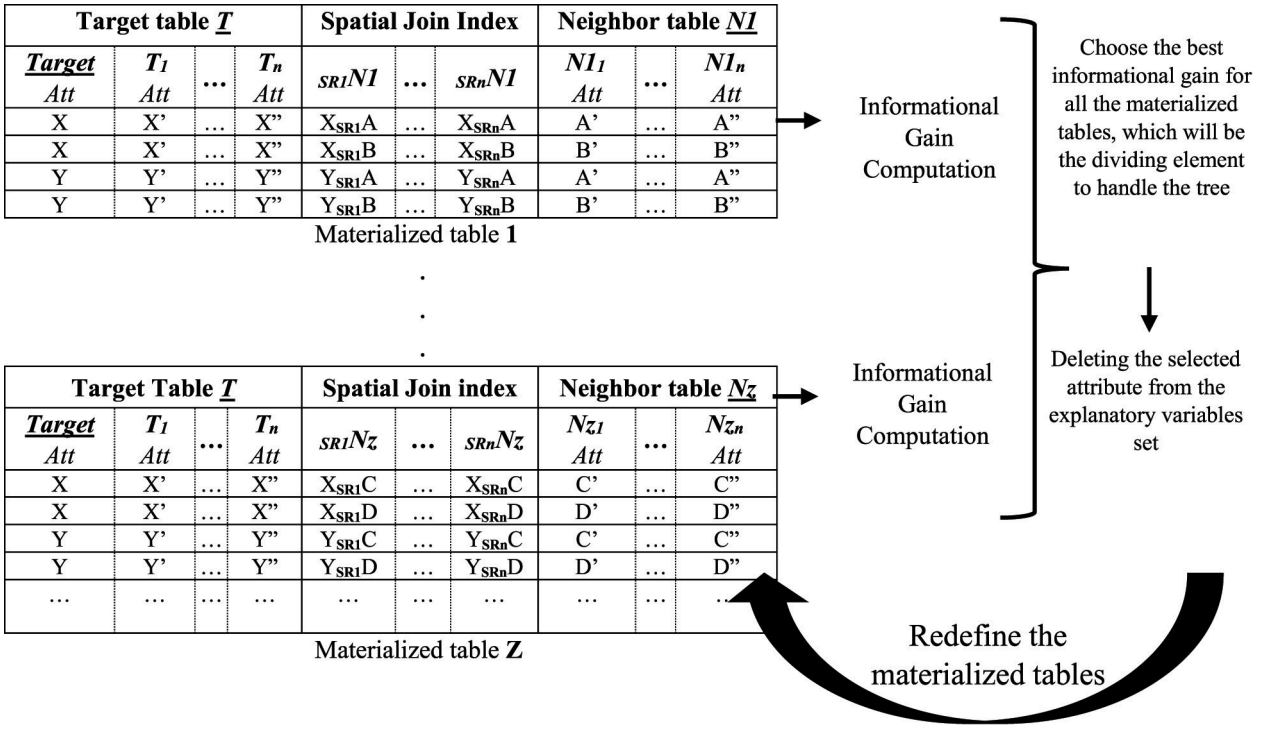


Figure 8. Structure of method 3: joins semi-materialization.

Results. Table 5 illustrates calculation of the duration and memory space needed by method 3: joins semi-materialization, tested with the dataset presented above.

The needed duration of execution is reduced compared to method 1: join on the fly, due to the lower number of join operations that are reduced. Paradoxically, it requires a larger memory space compared to method 1.

Table 5. Results of method 3: joins semi-materialization.

Duration	Memory Space
79 min 24.289 s	3 729 MB

3.2.4. Method 4: Imbricated Materialization

Despite the implementation of the joins semi-materialization method, we are still constrained by the balance between memory space and computing duration. To avoid these constraints, we have chosen to use imbricated data representation, which can also dispense us from other issues, like data duplication. We call this new method imbricated materialization.

This method is based on the same principle as the joins materialization, to bring all tables into a single table. However, this time, the proposed approach uses the object-oriented databases. It allows to avoid data duplication by including the attribute of 'table' type (which must be defined in advance) in the structure of the main table (materialized table).

In order to achieve this structure, this approach requires two steps:

- creation of the 'table' type for the imbricated tables;
- creation of the main table (materialized), containing imbricated tables.

The column of the materialized table represents a set of imbricated tables, and the line represents a set of lines in the appropriate imbricated table. This will generate a materialized table with a considerably reduced size, which meets all the criteria of avoiding data duplication and information loss.

We illustrate this proposed structure in Figure 9.

Results. Table 6 shows the computing time and memory space consumed by method 4: imbricated materialization, tested with the dataset presented above.

Target table *T*				**Neighbor *N1***						…	**Neighbor *Nz***					
Target *Att*	T_1 *Att*	…	T_n *Att*	***Spatial Relations***			***Attributes***			…	***Spatial Relations***			***Attributes***		
				SR_1N1	…	SR_nN1	$N1_1$ *Att*	…	$N1_n$ *Att*	…	SR_2Nz	…	SR_nNz	Nz_1 *Att*	…	Nz_n *Att*
X	X’	…	X”	$X_{SR1}A$	…	$X_{SRn}A$	A’	…	A”	…	$X_{SR2}C$	…	$X_{SRn}C$	C’	…	C”
				$X_{SR1}B$	…	$X_{SRn}B$	B’	…	B”	…	$X_{SR2}D$	…	$X_{SRn}D$	D’	…	D”
Y	Y’	…	Y”	$Y_{SR1}A$	…	$Y_{SRn}A$	A’	…	A”	…	$Y_{SR2}C$	…	$Y_{SRn}C$	C’	…	C”
				$Y_{SR1}B$	…	$Y_{SRn}B$	B’	…	B”	…	$Y_{SR2}D$	…	$Y_{SRn}D$	D’	…	D”

Figure 9. Structure of method 4: imbricated materialization.

Table 6. Results of method 4: imbricated materialization.

Duration	**Memory Space**
67 min 42.010 s	2 456 MB

The use of imbricated databases allowed us to optimize performance by reducing the size of the materialized table, represented as an object that contains imbricated tables, without any duplication (principal element for size reduction), which has automatically reduced the computing time. Consequently, this method establishes a certain balance between computing duration and memory space.

4. Comparison and Discussion

Table 7 summarizes the results of the study of phenomenon by multiple other phenomena, according to different approaches.

Table 7. Results of multiple methods of spatial data mining studying of a phenomenon by multiple other phenomena.

Method	**Duration**	**Memory Space**
Method 1: join on the fly	115 min 02.335s	1 644 MB
Method 2: joins materialization	71 min 15.320 s	7 280 MB
Method 3: joins semi-materialization	79 min 24.289 s	3 729 MB
Method 4: imbricated materialization	67 min 42.010 s	2 456 MB

For comparison purpose, we have used the same dataset throughout our works under the same tests environment. Depending on the applied method, we obtained different results; each one has its own advantages and disadvantages compared to the other methods. The first method applied consists of using the join operation when it is necessary, reducing the need of memory space at the expense of higher computing time, because of the important number of join operations.

Unlike the first method, the second one reduces the computing duration, but it uses more memory space by materializing all the tables into a single table once and for all. Considering the issues of this method, we have opted for another method that combines the first and the second methods, which are join on the fly and joins materialization. We have called this method joins semi-materialization. It reduces the number of joins operation, which gives better computing time compared to the first method.

The last method, imbricated materialization, allows a certain balance between computation time and required memory space, combining the advantages of previous methods and this is permitted through the object-oriented databases.

5. Conclusion and Perspectives

We have presented our work on spatial data mining by the adaptation of the classification C4.5 algorithm to spatial data according to two modifications, the first at the data organization level by the use of the Spatial Join Index (SJI) and the second at the algorithm level. We applied our S-C4.5 algorithm on the road safety spatial dataset organized in multi-thematic layers and conducted performance measures through different approaches divided into two study cases. The first case is a study of a phenomenon by a single other phenomenon, including one or multiple spatial relations. The second case is a study of a phenomenon by multiple other phenomena, including multiple spatial relations.

In the first case, we have used two methods, the join on the fly and joins materialization. The

first method fosters the consumption of memory space despite the necessary computing time, unlike the second one that reduces calculation time by consuming more memory space.

For the second case, we have included multiple phenomena and spatial relations, which had a negative impact on the performance in terms of computing time with the method 1, join on the fly, this leads us to experiment with the second method joins materialization, which presented data duplication issues.

To overcome this problem, we have proposed a new method, which we have named joins semi-materialization. It is situated between the two previous methods in terms of calculation time and memory space.

Finally, we have used object-oriented databases, in order to achieve a certain balance between the computing time and required memory space, by avoiding at the same time the previous methods' disadvantages, especially the data duplication.

Spatial data represent a huge complex data volume that negatively influences automatic learning algorithms, which makes the analysis of this data painful on a single machine due to limited memory and CPU resources. For these reasons, it would be interesting to bring our work to parallel and/or distributed computing architectures, where the method of join on the fly, for example, could present better results. It would be also interesting to experiment and compare our work with recent well-known technologies of Big Data, Data Warehouses and Data Lakes, with different databases management systems (column-oriented, kes-value, in-memory).

Future works will not be limited to performance issues only, it will also deal with other important classification aspects such as accuracy improvement.

References

[1] W. R. Tobler, "Cellular Geography", in *Philosophy in Geography*, pp. 379–386, 1979. https://doi.org/10.1007/978-94-009-9394-5_18

[2] L. Anselin, "What is Special About Spatial Data? Alternative Perspectives on Spatial Data Analysis (89-4)", UC Santa Barbara: National Center for Geographic Information and Analysis, 1989.

[3] J. R. Quinlan, C4.5: Programs for Machine Learning. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993.

[4] N. Chelghoum *et al.*, "A Decision Tree for Multi-Layered Spatial Data", in *Advances in Spatial Data Handling*, Springer Berlin Heidelberg, 2002, pp. 1–10. https://doi.org/10.1007/978-3-642-56094-1_1

[5] I. S. Sitanggang *et al.*, "An Extended ID3 Decision Tree Algorithm for Spatial Data," in *Proc. of the IEEE International Conference on Spatial Data Mining and Geographical Knowledge Services*, 2011, pp. 48–53. https://doi.org/10.1109/ICSDM.2011.5969003

[6] K. Zeitouni *et al.*, "Join Indices as a Tool for Spatial Data Mining", in *Temporal, Spatial, and Spatio-Temporal Data Mining*, 2001, pp. 105–116. https://doi.org/10.1007/3-540-45244-3_9

[7] M. Ester *et al.*, "Spatial Data Mining: A Database Approach", in *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, 1997, vol. 1262 LNCS, pp. 47–66. https://doi.org/10.1007/3-540-63238-7_24

[8] K. Zeitouni, "Data Analysis and Knowledge Discovery in Spatio-Temporal Databases", Université de Versailles-Saint Quentin en Yvelines, 2006.

[9] A. C. Gatrell *et al.*, "Spatial Point Pattern Analysis and Its Application in Geographical Epidemiology", *Trans. Inst. Br. Geogr.*, vol. 21, no. 1, p. 256, 1996. https://doi.org/10.2307/622936

[10] M. Ester *et al.*, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise", pp. 226–231, 1996. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.121.9220

[11] A. Getis, "Cliff, A.D. and Ord, J.K. 1973: Spatial Autocorrelation. London: Pion", *Prog. Hum. Geogr.*, vol. 19, no. 2, pp. 245–249, 1995. https://doi.org/10.1177/030913259501900205

[12] S. Shekhar and Y. Huang, "Discovering Spatial Co-Location Patterns: A Summary of Results", in *Advances in Spatial and Temporal Databases*, 2001, pp. 236–256. https://doi.org/10.1007/3-540-47724-1_13

[13] G. Manikandan and S. Srinivasan, "Mining of Spatial Co-Location Pattern Implementation by FP Growth", *Ind. J. Comput. Sci. Eng*, vol. 3, pp. 344–348, 2012.

[14] M. Ester *et al.*, "Algorithms for Characterization and Trend Detection in Spatial Databases", in *Proc. of the 4th Int. Conf. on Knowledge Discovery and Data Mining (KDD-98)*, 1998, pp. 44–50.

[15] K. Koperski and J. Han, "Discovery of Spatial Association Rules in Geographic Information Databases", in *Advances in Spatial Databases*, 1995, pp. 47–66. https://doi.org/10.1007/3-540-60159-7_4

[16] K. Koperski *et al.*, "An Efficient Two-Step Method for Classification of Spatial Data", in *Proc. of the 8th Symp. Spatial Data Handling*, 1998, pp. 45–54.

[17] R. Agrawal *et al.*, "Mining Association Rules Between Sets of Items in Large Databases", in *Proc. of the 1993 ACM SIGMOD International Conference on Management of Data*, 1993, pp. 207–216. https://doi.org/10.1145/170035.170072

[18] D. Malerba and F. A. Lisi, "An ILP Method for Spatial Association Rule Mining", in the 1st *Workshop on Multi-Relational Data Mining*, 2001, pp. 18–29.

[19] M. Ceci *et al.*, "Spatial Associative Classification at Different Levels of Granularity: A Probabilistic Approach", *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 3202, pp. 99–111, 2004. https://doi.org/10.1007/978-3-540-30116-5_12

[20] J. R. Quinlan, "Induction of Decision Trees", *Mach. Learn.*, vol. 1, no. 1, pp. 81–106, 1986. https://doi.org/10.1007/BF00116251

[21] K. Koperski, "A Progressive Refinement Approach to Spatial Data Mining", Doctoral thesis, Simon Fraser University, 1999.

[22] L. Breiman *et al.*, Classification and Regression Trees, Taylor & Francis, 1984.

[23] S. Rinzivillo and F. Turini, "Classification in Geographical Information Systems", *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 3202, pp. 374–385, 2004. https://doi.org/10.1007/978-3-540-30116-5_35

[24] I. Kononenko, "A Counter Example to the Stronger Version of the Binary Tree Hypothesis", in *ECML-95 workshop on Statistics, Machine Learning, and Knowledge Discovery in Databases*, 1995, p. 31.

[25] B. Hssina *et al.*, "A Comparative Study of Decision Tree ID3 and C4.5", *Int. J. Adv. Comput. Sci. Appl.*, vol. 4, no. 2, 2014. https://doi.org/10.14569/SpecialIssue.2014.040203

[26] P. Valduriez, "Join Indices", *ACM Trans. Database Syst.*, vol. 12, no. 2, pp. 218–246, 1987. https://doi.org/10.1145/22952.22955

[27] "Road Safety Data." [Online]. Available: https://data.gov.uk/dataset/road-accidents-safety-data [Accessed: 19-Nov-2019].

Received: February 2019
Revised: February 2020
Accepted: March 2020

Contact address:
Sihem Oujdi
Oran University of Science and Technology – Mohamed Boudiaf
Oran
Algeria
e-mail: sihem.oujdi@univ-usto.dz

Hafida Belbachir
Oran University of Science and Technology – Mohamed Boudiaf
Oran
Algeria
e-mail: h_belbach@yahoo.fr

Faouzi Boufares
University Sorbonne Paris Nord (Paris 13)
Paris
France
e-mail: boufares@lipn.univ-paris13.fr

Sihem Oujdi received the MSc degree in computer science (option: information systems engineering) from the University of Science and Technology of Oran – Mohamed Boudiaf (USTO-MB), Oran, Algeria, in 2011. She is currently a PhD student at the same university and a member of the LSSD laboratory. Her current research topics are spatial data mining and big data.

Hafida Belbachir received the PhD degree in computer science from the University of Es-Senia, Oran, Algeria, in 1990. From 1992 to 2006, she was an Associate Professor at the University of Science and Technology of Oran – Mohamed Boudiaf (USTO-MB), Oran, Algeria. Since 2006, she is a Professor at the same university. Currently, Prof. Hafida Belbachir has been head of the Database group in the LSSD laboratory at the same university since 2007. Her research interests are advanced databases, data mining and data grid.

Faouzi Boufares received the PhD degree in fundamental computer science from the University of Paris 11 (Paris Sud, Orsay), Paris, France, in 1986. Currently, he is a Lecturer with an HdR habilitation at the University of Sorbonne Paris Nord (Paris 13), Paris, France. Dr. Fouazi Boufares is a member of the LIPN - UMR CNRS 7030 laboratory at the same university. His current research topics are data management, data science, big data, data quality and the future.