Including spatial relations and scales within sequential pattern extraction

Mickaël Fabrègue, Agnès Braud, Sandra Bringay, Florence Le Ber, Maguelonne Teisseire

Including spatial relations and scales within sequential pattern extraction

Mickaël Fabrègue[1,4], Agnès Braud[2], Sandra Bringay[3], Florence Le Ber[4], and Maguelonne Teisseire[1]

[1] Cemagref, UMR TETIS, Montpellier, France
E-mail: {mickael.fabregue,maguelonne.teisseire}@teledetection.fr
[2] LSIIT, UMR 7005, University of Strasbourg, France
E-mail: agnes.braud@unistra.fr
[3] LIRMM, CNRS, Montpellier, France
E-mail: bringay@lirmm.fr
[4] LHYGES, CNRS-ENGEES, Strasbourg, France E-mail:
florence.leber@engees.unistra.fr

Abstract. Georeferenced databases contain a huge volume of temporal and spatial data. They are notably used in environmental analysis. Several works address the problem of mining those data, but none are able to take into account the richness of the data and especially their spatial and temporal dimensions. In this paper, we focus on the extraction of a new kind of spatio-temporal pattern, considering the relationship between spatial objects and geographical scales. We propose an algorithm, STR_PrefixGrowth, which can be applied to a huge amount of data. The proposed method is evaluated on hydrological data collected on the Saône watershed during the last 19 years. Our experiments emphasize the contribution of our approach toward the existing methods.

Keywords: Data Mining, Sequential patterns, Spatio-temporal, Aquatic ecosystem

1 Introduction

Due to the recent explosion of mobile technologies and georeferenced data, a new kind of data has emerged: spatio-temporal data. Each data is associated with a given spatial reference (i.e. a localisation) and a temporal information (i.e. a timestamp). New needs for the monitoring of these data in time and space have appeared, for example to study the spread of information in social networks [1], in epidemic surveys [2] or for hydrological monitoring as presented in this article. In these domains the volume of data is huge, and commonly contains heterogeneous information. Often several levels of spatial division describe the geographical aspect, based on an inclusion property and relationships between geographical objects. An area can be included in another area (e.g. Europe is divided into countries Spain, France, Germany, etc.). Moreover, the geographical objects are linked by spatial relations. For example, an area is close to another

area, is located to the north or to the east of another area (e.g. USA and Canada are two adjacent areas and USA is south of Canada). In this article we will focus on data mining methods which consider the temporal dimension and also spatial relationship between spatial objects. The objective is to provide a method for extracting spatio-temporal patterns to highlight common behavior in large volumes of data. The method is applied to the environmental domain, and more specifically for studying aquatic ecosystems. The dataset is a collection of samples on the hydrological catchment of the Saône, from part of the Fresqueau project. This project aims to provide operational tools to study the state of aquatic systems. It falls within the scope of the Water European Framework Directive which aims to correct the state of the aquatic systems and catchments in 2015.

2 Related work

Pattern extraction has been the subject of lot of research in the field of data mining. Pattern discovery highlights a recurring information in data, characterizing a frequent behavior. This knowledge can be represented by various types of patterns. Several authors have proposed new methods that consider both time and space.

In [3], data is represented in a set of spatial grids where items (events) appear at different coordinates. Each grid describes the state of the problem at a specific timestamps t. For each date and absolute position, an itemset (a set of events), is generated. For each absolute position, a sequence of itemset is built by considering all the timestamps. Then, sequential patterns are extracted from such sequences by considering an absolute position as the reference point. An example of a pattern obtained by such a method is $\langle$(Rain(0,0))(Humidity(0,1))$\rangle$ meaning that it frequently rains at coordinates (0,0) and, later in time, humidity exists at coordinates (0,1). This kind of pattern has the disadvantage of being sensitive to the choice of the reference point. Furthermore the space is reduced to a grid representation.

In [4], the authors proposed the concept of close events in time and space. A spatio-temporal window is defined by both a temporal and a spatial interval. Patterns are association rules such as $\langle$Rain$\Rightarrow$Humidity$\rangle$, meaning that in close areas at close timestamps, the rain is frequently followed by humidity. These rules do not take in consideration potential relationships between spatial objects nor different geographical scales.

The extraction of spatio-temporals patterns with neighborhood relationships between geographical objects is proposed in [5]. Patterns have the shape $\langle$(Humidity .[Rain Wind])(Humidity Rain)$\rangle$. Neighborhood relationships are denoted by a neighborhood operator **.** and a grouping operator []. Consider, for example, a city in which the previous pattern is found. This pattern means that humidity appeared at a timestamp and at the same time rain and wind appeared in a nearby town (according to an euclidean distance or defined by user). Later, humidity and rain appeared in the city. This spatial relationship is simple and

it is not possible to specialize it, nor to have several levels of granularities (i.e. several geographical scales). Actually, it is limited to one kind of relationship: spatial proximity.

A technique for granularity management of space is provided in [6]. As in [3], a grid of events represents spatiality and a set of spatial grids represent temporality. The user has to choose a level of granularity that will merge a set of adjacent cases in the grid. The higher the level of granularity, the bigger the set of merged cases. This technique aims to generalize data in a spatial way. To extract patterns, it is necessary to choose a granularity value and furthermore a grid representation to describe spatiality. Extracted patterns have the shape of classical sequential patterns such as $\langle$(Sun)(Wind)(Sun,Humidity)$\rangle$, meaning that frequently *Sun* is followed by the event $Wind$, followed itself by the events *Sun* and $Humidity$ according to a specific level of granularity.

All these methods do not effectively consider complex data with geographic objects linked together and at different scales. The approach in this paper aims to take into account all these notions: 1) by considering the temporal and spatial dimensions 2) by generalizing the problem to a more complex spatial relationships between geographical objects 3) by including all possible granularities during the extraction process.

In section 3.1, we introduce some preliminary definitions on which our method is based. Then sections 3.2 and 3.3 present a formal framework to take into account relationships between objects and different spatial granularities. The developed algorithm is presented in section 4. In section 5, the method is applied to a real dataset and the obtained results are presented. We discuss on the prospects of the scope of this proposal in section 6.

3 Spatio-temporal patterns

Our approach extends the notion of sequential patterns introduced in [7] and takes into account the temporal and spatial sequential patterns defined in [5].

3.1 Preliminaries

Sequential patterns are extracted from a set of data sequences. For each coordinate or geographical object, a sequence of events is built. First, we consider the database $\mathcal{DB}$ presented in the table 1 which shows the set of events which appeared in three different cities in the south of France. For each city, a sequence is generated (see. the table 2)

Definition 1 ***(Sequence)*** *Let $\mathcal{I} = \{I_1, I_2, \ldots, I_m\}$ be the set of items (events). An itemset is a non empty set of items denoted $(I_1, I_2, \ldots, I_k)$ where $I_j \in \mathcal{I}$ (it is a non ordered representation). A sequence s is a non empty ordered list of itemsets noted $\langle IS_1 IS_2 \ldots IS_p \rangle$ where $IS_j \in \mathcal{IS}$, with $\mathcal{IS}$ the set of itemsets.*

Each sequence being composed of itemset, an item can appears several times in a same sequence.

City	Month	Items
Nîmes	2011/01	Humidity=Low, Sun
Montpellier	2011/02	Sun
Nîmes	2011/03	Heat=High
Montpellier	2011/03	Humidity=Low, Heat=High
Nîmes	2011/04	Heat=Low, Wind
Orange	2011/04	Rain
Orange	2011/06	Rain, Wind

Table 1. Database

City	Sequence
Nîmes	$\langle$(Humidity=Low Sun)(Heat=High)(Heat=Low Wind)$\rangle$
Montpellier	$\langle$(Sun)(Humidity=Low Heat=High)$\rangle$
Orange	$\langle$(Rain)(Rain Wind)$\rangle$

Table 2. Sequences of city

Extracting knowledge from sequences search frequent sub-sequences, named as sequential patterns. Several algorithms have been proposed for sequential pattern mining [7–12].

Definition 2 ***(Sub-sequence)*** *A sequence $A = \langle IS_1 IS_2 \dots IS_p \rangle$ is a sub-sequence of another sequence $B = \langle IS'_1 IS'_2 \dots IS'_m \rangle$ ($A \preceq B$) if $p \leqslant m$ and if there exists integers $j_1 < j_2 < \dots < j_k < \dots < j_p$ such as $IS_1 \subseteq IS_{j_1}, IS_2 \subseteq IS_{j_2}, \dots, IS_p \subseteq IS_{j_p}$.*

Example 1 *Consider the sequences presented in table 2 where each represents an event sequence for a city, we note that the sequence $S = \langle (Sun)\ (Heat{=}High) \rangle$ is supported by sequences $S_{N\hat{i}mes}$ and $S_{Montpellier}$. Then $S \preceq S_{N\hat{i}mes}$ and $S \preceq S_{Montpellier}$.*

A sequential pattern is a frequent sub-sequence characterized by a support which is the number of occurrences of a pattern in $\mathcal{S}$, a set of sequences. The extraction of those patterns is determined by a minimum support parameter denoted θ. This means that only patterns with a support value greater than θ will be extracted. Let $\mathcal{M}$ be the set of extracted sequential patterns, then $\forall M \in \mathcal{M}$ and $Support(M) \geqslant \theta$.

Definition 3 ***(Sequential pattern support)*** *A sequence $S \in \mathcal{S}$ supports a sequential pattern M when $M \preceq S$. The support of a pattern M is the number of sequences in S in which M is included (supported). Let $\mathcal{S}'$ be the set of sequences that support m, then $\mathcal{S}' = \{S_i \in \mathcal{S}$ such as $M \preceq S_i\}$ and $Support(M) = |\mathcal{S}'|$.*

Example 2 *Consider table 2, we note that sequence $S = \langle (Sun)\ (Heat{=}High) \rangle$ is supported by sequences $S_{N\hat{i}mes}$ and $S_{Montpellier}$. Therefore $Support(S) = 2$.*

Although sequential patterns fit the temporal aspect well, they are not able to consider the spatiality nor potential relationships between geographical objects. To take into consideration these two aspects, we now present a hierarchical approach based on dimensions.

3.2 Relationships between spatial objects

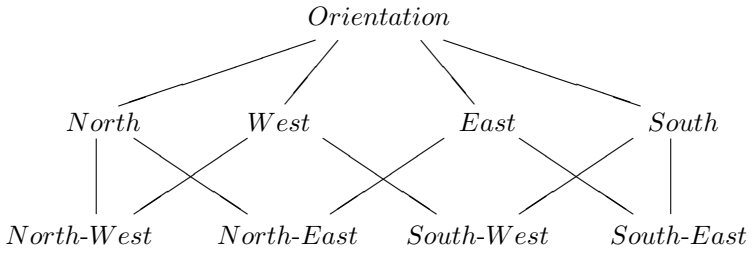
Relationships between spatial objects are potential links that exist between geographic points or objects. For instance, in an epidemiological context, spatial objects are cities or areas. Therefore, several links should be considered, as migration flows or obstructions presence like a forest or a mountain. These links can also be specialized, e.g a forest can be a fir forest or an oak forest. Therefore important to take into account these links but also their potential specializations. To achieve this goal, we use a set of spatio-relational dimensions $\mathcal{D}_{\mathcal{R}}$ described by sets of values with an associated hierarchy for each dimension.

Definition 4 ***(Spatio-relational dimension)*** *A dimension $D \in \mathcal{D}_{\mathcal{R}}$ is defined by a domain of values X_j such as $dom(D) = \{X_1, X_2, ..., X_n\}$.*

Example 3 *Let $D_{Orientation}$ be the spatio-relational dimension representing orientation according to the points of the compass with the addition of North-West, North-East, South-West and South-East: $dom(D_{Orientation}) = \{North, West, South, East, North\text{-}West, North\text{-}East, South\text{-}West, South\text{-}East\}$. Then the following hierarchy is constructed:*

After defining the concept of a spatio-relational dimension, we introduce the notion of a hierarchy on a dimension. The ojective is to easily consider the more specific relations of an existing relation.

Definition 5 ***(Hierarchical representation of a spatio-relational dimension)*** *Let $D \in \mathcal{D}_{\mathcal{R}}$ be a spatio-relational dimension with $dom(D) = \{X_1, X_2, ..., X_n\}$ and let $H \in \mathcal{H}_{\mathcal{R}}$ be the hierarchy representation associated to this dimension, then H is a semi-lattice or an oriented tree and for all node $N \in H$, $label(N) \in dom(D)$.*



To navigate in this hierarchy, we have to define some navigation operations as defined in [13]. These operations represent the notions of direct and global generalization and specialization.

Definition 6 ***(Direct and global specialization)*** *Let $down_R(X_i)$ be the operation which allows access to direct specializations of the relation X_i and $downAll_R$ (X_i) the operation which allows access to all specializations of relation X_i. The direct specializations of X_i are X_j such that there is a descending edge from X_i to X_j in the hierarchy, and the global specializations of X_i are X_k such that there is a descending path X_i to X_k.*

Example 4 *Lets consider the dimension $D_{Orientation}$:*
- $down_R(West) = \{North\text{-}West, South\text{-}West\}$,
- $downAll_R(Orientation) = dom(d_{Orientation})$.

Definition 7 ***(Direct and global generalization)*** *Let $up_R(X_i)$ be the operation which allows access to direct generalizations of the relation X_i and $upAll_R(X_i)$ the operation which allows access to all generalizations of the relation X_i. The direct generalizations of X_i are X_j such as there is an ascending edge from X_i to X_j in the hierarchy, and X_i global generalizations are all X_k such as there is an ascending path from X_i to X_k.*

Example 5 *Using the same example:*
- $up_R(North\text{-}East) = \{North, East\}$,
- $upAll_R(North\text{-}West) = \{North, West, Orientation\}$.

This hierarchy offers the possibility to extract information at different levels. For instance, it can take into consideration the presence of an event at the north, but also drill down the hierarchy to find more specific relations. From the definition of the hierarchy on spatio-relational dimensions and their operations, patterns are extracted by considering relations between spatial objects. To add this new information to patterns, we use the link operator **.** when an item is found in a zone linked by a relation, as in [5]. When multiple items are considered by the operator **.**, the *n-ary* group operator [] is used.

Definition 8 ***(Related itemset)***
Let IS and IS' be two itemsets which describe two differents zones Z and Z' at the same timestamps, if there exists a link δ in the spatio-relational hierarchy between Z and Z', then they constitute a related itemset noted $IS_R = (IS\ .\delta[IS'])$ which means the itemset IS is found in Z and at the same time the itemset IS' appears in a closed zone in δ relation with the first zone.

Example 6 *Taking two cities C_1 and C_2, humidity appears in C_1, rain and wind in C_2 at the same timestamps t. Furthermore the hierarchy highlights the fact that C_2 is at south of C_1 then the related itemset $IS_R = (Humidity\ ._{South}[Rain\ Wind])$ is found in C_1.*

We have now to define the inclusion of a related pattern in another related sequential pattern. This inclusion is very close to the classic sequential pattern inclusion, the difference concerns the inclusion between itemsets.

Definition 9 ***(Inclusion of a related itemset)***
A related itemset $IS_R = IS_i.\delta[IS_j]$ *is included in another related itemset* $IS'_R = IS'_i.\delta'[IS'_j]$*, if and only if,* $IS_i \subseteq IS'_i$*,* $IS_j \subseteq IS'_j$ *and* $\delta = \delta'$ *or* $\delta \in upAll(\delta')$ *(i.e.* δ' *is equal to* δ *or* δ' *is a specialization of* δ*).*

Example 7 *Let* $D\{Orientation\}$ *be the spatio-relational dimension of two itemsets* IS_1 *and* IS_2 *such that* $IS_1 = .{}_{South}[Humidity, Wind]$ *and* $IS_2 = .{}_{South-East}[$ *Humidity, Rain, Wind]. We can note that all items in* IS_1 *are included in* IS_2 *and the relationship* IS_1 *is more general than* IS_2 *in the hierarchy. Therefore* $IS_1 \preceq IS_2$*.*

The obtained sequential patterns are composed of related itemsets and form a new kind of pattern, i.e. related sequential patterns.

Definition 10 ***(Related sequential pattern)***
Let $\mathcal{IS}$ *be the set of itemsets and* $\mathcal{IS}_\mathcal{R}$ *the set of related itemsets, a related sequential pattern* M_R *is a non-empty ordered list of itemsets and related itemsets denoted* $\langle IS_1, IS_2, \ldots, IS_p \rangle$ *where* $IS_j \in \mathcal{IS} \cup \mathcal{IS}_\mathcal{R}$ *with a support value* $Support(M_R)$*.*

In this section, we introduced a new kind a sequential pattern which considers existing links between geographical objects. These relations are organized in a hierarchy to efficiently consider specializations and generalizations. But in the context of spatial segmentation, it is also important to take into account the spatial granularity which exist between the zones in patterns in order to provide the experts with more precise patterns. The next section presents this new feature.

3.3 Geographical granularities in patterns

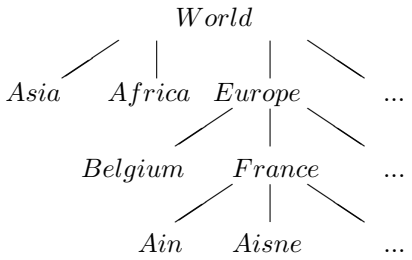
Different geographical granularities describe a division of space, itself divided into sub-divisions. This segmentation can have different shapes according to the context of the problem to solve. For example, let us consider a division of the Earth with respect to a geopolitical point of view. Space is divided according to continental frontiers or country boundaries. With respect to a climatic point of view, this division is different: hot climate areas, temperate areas, etc. In addition, areas are further divided into smaller regions. It is therefore necessary to not only take into consideration areas, but also their sub-divisions.

Definition 11 ***(Area dimension)*** *An area dimension* $D \in \mathcal{D}_\mathcal{S}$ *is defined by a domain of values* X_j *such as* $dom(D) = \{X_1, X_2, ..., X_n\}$*.*

Example 8 *Let* $D_{Country} \in \mathcal{D}_\mathcal{S}$ *be an area dimension describing Europe's division into countries.*
$dom(d_{Country}) = \{Austria, Belgium, Bulgaria, ..., Sweden, United\ Kingdom\}$*.*

Considering granularity relies on the construction of a hierarchy based on an inclusion relation on such geographical divisions. To illustrate this, let us take as an example a division of the Earth by considering multiple granularities, continents and countries. The following hierarchy describes this division.

Example 9 *Earth geopolitical division hierarchy*



This hierarchical representation of spatial granularities is close to hierarchies representing links between spatial objects (previous section). However, this hierarchy is not based on a generalization/specialization notion but on an inclusion order. For instance, a country is not a continent specialization but a sub-division. We have to redefine navigation operations in this type of hierarchy $H \in \mathcal{H}_{\mathcal{S}}$, with $\mathcal{H}_{\mathcal{S}}$ the set of hierarchies on granularity dimension.

Definition 12 ***(Direct and global content)*** *Let $down_S(X_i)$ be an operation that allows access to the direct content of granularity X_i and $downAll_R(X_i)$ the operation which allows access to all the content of granularity X_i. The direct content of X_i is X_j such that there is a descending edge from X_i to X_j in the hierarchy, the global content of X_i is X_k such that there is a descending path from X_i to X_k.*

Example 10 *Let us take the example of Earth division:*
- $down_S(Europe) = \{Belgium, France...\}$,
- $downAll_S(Europe) = \{Belgium, France..., Ain, Aisne...\}$.

Definition 13 ***(Direct and global containers)*** *Let $up_S(X_i)$ be an operation that allows access to the granularity that directly contains X_i and $upAll_S(X_i)$ the operation which allows access to all granularities that contain X_i. The direct containers of X_i are all X_j such as there is a ascending edge from X_i to X_j in the hierarchy, and the global containers of X_i are all X_k such that there is an ascending path from X_i to X_k.*

Example 11 *Let us take the previous example:*
- $up_S(Belgium) = \{Europe\}$,
- $upAll_S(Ain) = \{France, Europe, World\}$.

We use this hierarchy to add the notion of spatial inclusion into patterns To extract these patterns, the algorithm navigates throughout granularity hierarchies and checks if a pattern is frequent at a more specific level of the hierarchy. If it is indeed the case, the pattern becomes spatio-temporal because its frequency depends on a specific spatial area.

Definition 14 ***(Spatio-temporal pattern)***
Let $[\![\ ]\!]$ be an operator of spatiality and M a classic or related sequential pattern, $X_k \in D$ the value of a granularity dimension D, a minimal support θ and $\mathcal{S}'$ the set of sequences S_i such that $|M \preceq S_i|$ at the granularity value X_k. If $|\mathcal{S}'| > \theta$ then a spatio-temporal pattern M' is created, such that $M' = [\![X_k]\!]M$.

Example 12 *Let $M = \langle (Humidity._{North}[RainWind])(HumidityRain)\rangle$ be the relational pattern, with $\theta = 10\%$ and $Support(M) = 50\%$. The pattern M has a frequency equal to 50% over the Earth but has a frequency equal to 15% if we just consider European cities. A spatio-temporal pattern M' is created such that $M' = [\![Europe]\!]\langle (Humidity._{North}[RainWind])(HumidityRain)\rangle$ and $Support(M') = 15\%$*

The previously presented definitions allow for taking into account spatial relationships and also geographical granularities. An adapted algorithm has been implemented to extract related spatio-temporal patterns at differents scales. This algorithm is presented in the next section.

4 STR_PrefixGrowth algorithm

To extract patterns, we used the PrefixSpan [14] extraction algorithm, as was also used in [5]. This is currently one of the most efficient algorithms for extracting sequential patterns, both in terms of computation time and in terms of memory consumption. Sequential patterns are extracted from common prefixes. For instance, $\langle (a)\rangle$, $\langle (a)(a)\rangle$, $\langle (a)(ab)\rangle$ and $\langle (a)(abc)\rangle$ are prefixes of sequence $\langle (a)(abc)(ac)(d)(cf)\rangle$.

If a prefix is present in a number of sequences greater than a minimum support value θ, then this prefix is considered as frequent. When a frequent prefix is found, the database is divided recursively. When we look for frequent patterns, it is not necessary to keep the entire database and therefore data (i.e sequences) that do not support the current pattern are not preserved in the projected database. The reason is that these sequences will not support patterns of greater length because of the antimonotonic property of support. The efficiency of this algorithm is due to (1) the non-generation of candidate patterns thanks to research of frequent prefixes, and (2) the projection of the database into smaller databases to accelerate the exploration by removing sequences no longer needed.

Algorithm 1: $STR\_PrefixGrowth(\alpha, \theta, \mathcal{DB}|_\alpha, \mathcal{D}_\mathcal{R}, \mathcal{D}_\mathcal{S})$

input : α a pattern, θ a support minimum, $\mathcal{DB}|_\alpha$ projected database according to pattern α , $\mathcal{D}_\mathcal{R}$ a set of spatio-relational hierarchies, $\mathcal{D}_\mathcal{S}$ a set of granularity hierarchies

output: $\mathcal{SP}$ set of patterns extracted in this function call (i.e current recursion)

$I_\theta \leftarrow$ getListOccurences$(\theta, \mathcal{DB}|_\alpha, \mathcal{D}_\mathcal{R})$;
$\mathcal{SP} \leftarrow \varnothing$;
foreach i *in* I_θ **do**
  β=append(α,i);
  $\mathcal{SP} \leftarrow \mathcal{SP} \cup \beta$;
  $\mathcal{SP} \leftarrow \mathcal{SP} \cup$ prefixGrowth$_{STM}(\beta, \theta, \mathcal{DB}|_\beta, \mathcal{D}_\mathcal{R}, \mathcal{D}_\mathcal{S})$;
  $\mathcal{SP} \leftarrow \mathcal{SP} \cup$ exploreSpatialHierarchy$(\beta, \theta, \mathcal{DB}|_\beta, \mathcal{D}_\mathcal{S})$;
end

Algorithm 2: $getListOccurences(\theta, \mathcal{DB}|_\alpha, \mathcal{D}_\mathcal{R})$

input : θ a minimum support, $\mathcal{DB}|_\alpha$ projected database according to pattern α, $\mathcal{D}_\mathcal{R}$ a set of spatio-relationnal hierarchies

output: I_θ the list of frequent occurences in $\mathcal{DB}|_\alpha$

$I_\theta \leftarrow I_\theta \cup$ searchIExtend$(\theta,\mathcal{DB}|_\alpha)$;
$I_\theta \leftarrow I_\theta \cup$ searchSExtend$(\theta,\mathcal{DB}|_\alpha)$;
foreach dim_i *in* $\mathcal{D}_\mathcal{R}$ **do** `/* For each dimension in` $\mathcal{D}_\mathcal{R}$ `*/`
  $I_\theta \leftarrow I_\theta \cup$ searchIntend$(\theta,\mathcal{DB}|_\alpha,dim_i)$;
  $I_\theta \leftarrow I_\theta \cup$ searchExtend$(\theta,\mathcal{DB}|_\alpha,dim_i)$;
end

Algorithm 3: $exploreSpatialHierarchy(\alpha, \theta, \mathcal{DB}|_\alpha, \mathcal{D}_\mathcal{S})$

input : α a pattern , θ a minimum support, $\mathcal{DB}|_\alpha$ projected database according to pattern α , $\mathcal{D}_\mathcal{S}$ a set of granularity hierarchies

output: $\mathcal{SP}$ the set of extracted patterns

$\mathcal{SP} \leftarrow \varnothing$;
foreach dim_i *in* $\mathcal{D}_\mathcal{S}$ **do**
  foreach s *in* dim_i **do**
    if *isFrequent*$(\alpha, \theta, \mathcal{DB}|_\alpha,s)$ **then** `/* check if a pattern is frequent in the current granularity */`
      $\mathcal{SP} \leftarrow \mathcal{SP} \cup$ spatialPattern(α, s);
  end
end

Our general approach is described by the recursive algorithm 1, called $STRPrefixGrowth$ for **S**patio **T**emporal and **R**elational **PrefixGrowth** . This method first determines the list of frequent occurrences in the database projected according to α and depending on the minimum support θ. A frequent occurrence (e.g. a frequent item) means that a pattern of greater length is found. In the function $getListOccurences()$, we explore the relationship hierarchies. Two operations are used, the $searchIExtend()$ and $searchSExtend()$, representing the two ways to extend a pattern, the I-Extension and the S-Extension. The I-extension adds an item to the last itemset of a sequence and the S-Extension adds a new item to a new itemset at the end of a sequence, at a further timestamp. For example let us take the pattern $m = \langle(a)(b)\rangle$ and a frequent occurrence representing the item c. If c is an I-extension and m' an extended pattern, then $m = \langle(a)(bc)\rangle$. If c is an S-extension and m'' an extended pattern, then $m'' = \langle(a)(b)(c)\rangle$. For each relationship hierarchy, $searchIExtend()$ and $searchSExtend()$ operations are used to find occurrences of relations on every level of hierarchies. Frequent relations are then considered as occurrences. Relations between sequences are managed as individual items, they are returned along with occurrences of classics items. This function is provided by algorithm 2.

Occurrences, or frequent items, will be used to extend the pattern α with the function $append()$, which considers that an item is an intension or an extension. Then, for each extended pattern β, we project the database according to this pattern and we call $PrefixGrowth_{STM}$ to continue the recursive search of patterns. Finally, each pattern is given as a parameter of the function $exploreSpatialHierarchy()$ that explores the spatial dimensions at all levels of granularity (algorithm 3) to find new patterns (section 3.3). For each spatial dimension, it checks if a pattern is frequent at each granularity of the hierarchies. If it does, we add the spatial pattern to the set of patterns.

The PrefixSpan complexity in the worst case is $\Theta((2 \cdot I)^L)$ with I the number of items and L the length of the longest sequence in the database $\mathcal{DB}$. Let H_R be the number of hierarchies of spatial-relations, let R be the maximal number of relations per hierarchy of spatial relations, let H_S be the number of spatial hierarchies and let S the maximal number of spatial areas per spatial hierarchy, the complexity of the STR_PrefixGrowt algorithm is $\Theta(H_S \cdot S \cdot (2 \cdot N \cdot H_R \cdot R)^L)$. This algorithm is pseudo-polynomial, i.e. is linear according to the number of extracted patterns. The worst case corresponds to the maximal number of patterns which could be extracted in a specific dataset.

To test and validate our method, we have applied this algorithm to a real dataset and we have compared it to existing methods. These results are presented in the following section.

5 Mining hydrological data

The dataset has been supplied by the RMC agency in the context of the Fresqueau project. It describes the biological and physicochemical information of streams in the Saône watershed, in the east of France. The data have been collected at

different timestamps on 771 sites. The information contains different kinds of characteristics as biological indicators, pH, levels of nitrates or phosphates... For each site, a set of collected data for a specific timestamp is an itemset and those itemsets are ordered according to the time to generate a sequence.

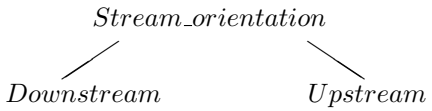
Moreover, to apply our approach, we have selected some characteristics 1) to explicit the links between river sites and 2) to consider different geographical scales.

5.1 Hierarchies

Those data are described by several dimensions with their associated hierarchies to consider granularities and links between stations. They are presented as follows:

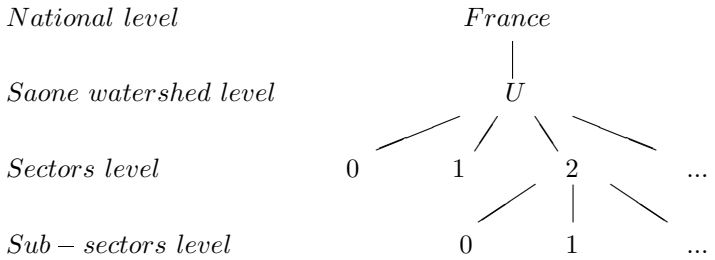
Stream orientation: this allows us to know whether a site is located downstream or upstream from another site. This is a simple hierarchy, one level deep.

Example 13 *Stream orientation hierarchy*



Hydrographic zones: France is divided into general watersheds and into three more specific partitions. Each level is a sub-division of the previous level. Watersheds are the most general level, itself divided into hydrographic areas. Then, there are sectors divided into sub-sectors. This hierarchy therefore has 4 levels.

Example 14 *Hydrographic zone hierarchy*



Each site is upstream or downstream to a neighboring site and is associated with a hydrographic zone. Stream's orientation is used as a spatio-related dimension between site (section 3.2). Hydrographic zones are used to take into account geographical granularity (section 3.3).

5.2 Experimentation

Before extracting patterns, we have to discretize data. An arbitrary discretization with 5 intervals is selected for each type of information. A description of the information that appears in patterns given in table 4 is presented in the following:

ibgn: Normalized global biological index (IBGN) is a tool used to evaluate biological quality in a watershed. This biological index has a value between 0 and 20 depending on the presence of some bioindicators (invertebrates).

ibgn_note: is a score ranging from 0 to 5 and is based on the IBGN value.

var_taxo: this data describes taxonomic variety. This is a metric corresponding to number of taxa (freshwater macroinvertebrates) collected during a sampling and is used in the IBGN computation.

We compare our approach to classical sequential pattern extraction methods (**MS**) and spatio-temporal patterns obtained with the approach in [5] ($\mathbf{M}_{ST}$). Both methods are close to ours, called $\mathbf{M}_{STR}$ for **S**patio-**T**emporal and **R**elated. In table 3, we vary support minimum values to observe the evolution of the number of sequential patterns according to the differents methods. Table 4 presents an example of patterns extracted for each method.

	MS	$\mathbf{M}_{ST}$	$\mathbf{M}_{STR}$
0.5	1	4	4
0.4	4	12	12
0.3	22	60	64
0.2	75	186	233
0.1	180	445	1882

Table 3. Number of extracted patterns according to minimum support

Exploring hierarchical granularities and spatial relations allows the extraction of more specific and expressive patterns, not obtainable with existing methods. For instance, the pattern $p = \langle(.Orient[ibgn\_11\text{-}15])(var\_taxo\_31\text{-}40)\rangle$ 4) means that frequently an IBGN value between 11 and 15 is frequently found in a neighboring site (i.e upstream **or** downstream) associated with a later taxonomic variety between 31 and 40. The pattern $p' = \langle(.Downstream[ibgn\_11\text{-}15])(var\_taxo\_31\text{-}40)\rangle$ is a specialization of p and frequently finds the IBGN value between 11 and 15 in a downstream site. The pattern $p'' = [\![U2]\!]\langle(.Orient[ibgn\_11\text{-}15])(var\_taxo\_31\text{-}40)\rangle$ means that the pattern p is frequent in the sector U2, a more specific geographic area. These patterns cannot be obtained with classical sequential patterns, e.g. $\langle(var\_taxo\_31\text{-}40)\rangle$, nor the method presented in [5], e.g. $\langle(.[ibgn\_11\text{-}15])(var\_taxo\_31\text{-}40)\rangle$. Experts often have difficulties to determine the best scale to obtain the best observations, and for each parameter, the

Method	Sequence	Support
MS	$\langle(var\_taxo\_31\text{-}40)\rangle$	0.404
$\mathbf{M}_{ST}$	$\langle(.[ibgn\_11\text{-}15])(var\_taxo\_31\text{-}40)\rangle$	0.089
	$\langle(.[ibgn\_note\_3])(var\_taxo\_31\text{-}40)\rangle$	0.056
$\mathbf{M}_{STR}$	$\langle(.Orient[ibgn\_11\text{-}15])(var\_taxo\_31\text{-}40)\rangle$	0.089
	$\langle(.Downstream[ibgn\_11\text{-}15])(var\_taxo\_31\text{-}40)\rangle$	0.051
	$[\![U1]\!]\langle(.Orient[ibgn\_11\text{-}15])(var\_taxo\_31\text{-}40)\rangle$	0.054
	$[\![U2]\!]\langle(.Orient[ibgn\_11\text{-}15])(var\_taxo\_31\text{-}40)\rangle$	0.073
	$\langle(.Orient[ibgn\_note\_3])(var\_taxo\_31\text{-}40)\rangle$	0.056
	$[\![U1]\!]\langle(.Orient[ibgn\_note\_3])(var\_taxo\_31\text{-}40)\rangle$	0.051

Table 4. Patterns according to different methods

best scale can be different from another. Our approach allows the presence of different hierarchical levels in the results. Finally, our approach deals with several issues: 1) considering spatial and temporal dimensions, 2) managing relations between geographical objects, and 3) exploring all granularities.

6 Conclusion

The method proposed in this paper tackles on mining georeferenced data and is able to consider efficiently the spatial and temporal dimensions. Our approach differs from solutions proposed in the literature, by considering both spatial relationships and granularities in a new way. The obtained patterns are semantically richer nevertheless this type of extraction leads to the exploration of a huge search space with an important amount of patterns. In the future, we wish to adapt some interestingness measures [15, 16] to these kinds of patterns to 1) filter the patterns according to experts' needs and 2) push it in the pattern extraction process. We aim at improving the extraction time by reducing the search space, and also provide experts with the minimal and most interesting set of spatio-temporal and related patterns. An another prospect is to define some tools to help expert's navigation in results by considering ergonomic and visualization aspect.

7 Acknowledgments

Thomas Lampert is gratefully acknowledged for helpful comments on the manuscript. This work was partly funded by french contract ANR11_MONU14.

References

1. Lin, C.X., Mei, Q., Jiang, Y., Han, J., Qi, S.: Inferring the diffusion and evolution of topics in social communities. Evolution **3**(3) (2011) 1231–1240

2. Gubler, D.J.: Epidemic dengue/dengue hemorrhagic fever as a public health, social and economic problem in the 21st century. Trends in Microbiology **10**(2) (2002) 100–103
3. Wang, J., Hsu, W., Lee, M.L.: LNCS 3453 - Mining Generalized Spatio-Temporal Patterns. (2005) 649–661
4. Huang, Y., Zhang, L., Zhang, P.: A Framework for Mining Sequential Patterns from Spatio-Temporal Event Data Sets. IEEE Transactions on Knowledge and Data Engineering **20**(4) (April 2008) 433–448
5. Alatrista Salas, H., Bringay, S., Flouvat, F., Selmaoui-Folcher, N., Teisseire, M.: The pattern next door: Towards spatio-sequential pattern discovery. In Tan, P.N., Chawla, S., Ho, C., Bailey, J., eds.: Advances in Knowledge Discovery and Data Mining. Volume 7302 of Lecture Notes in Computer Science. (2012) 157–168
6. Tsoukatos, I., Gunopulos, D.: Efficient mining of spatiotemporal patterns. In: Proceedings of the 7th International Symposium on Advances in Spatial and Temporal Databases. SSTD '01, London, UK, UK, Springer-Verlag (2001) 425–442
7. Agrawal, R., Srikant, R.: Mining sequential patterns. (1995) 3–14
8. Ayres, J., Flannick, J., Gehrke, J., Yiu, T.: Sequential pattern mining using a bitmap representation. In: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. KDD '02, New York, NY, USA, ACM (2002) 429–435
9. Zaki, M.J.: Spade : An efficient algorithm for mining frequent sequences. Machine Learning **42** (2001) 31–60
10. Srikant, R., Agrawal, R.: Mining sequential patterns: Generalizations and performance improvements. (1996) 3–17
11. Masseglia, F., Cathala, F., Poncelet, P.: The psp approach for mining sequential patterns. (1998) 176–184
12. Han, J., Pei, J., Mortazavi-Asl, B., Chen, Q., Dayal, U., Hsu, M.C.: Freespan: frequent pattern-projected sequential pattern mining. In: Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining. KDD '00, New York, NY, USA, ACM (2000) 355–359
13. Plantevit, M., Laurent, A., Laurent, D., Teisseire, M., Choong, Y.W.: Mining multidimensional and multilevel sequential patterns. ACM Trans. Knowl. Discov. Data **4** (January 2010) 4:1–4:37
14. Pei, J., Han, J., Mortazavi-Asl, B., Wang, J., Pinto, H., Chen, Q., Dayal, U., Hsu, M.C.: Mining sequential patterns by pattern-growth: The prefixspan approach. IEEE Trans. on Knowl. and Data Eng. **16** (November 2004) 1424–1440
15. Tan, P.N., Kumar, V., Srivastava, J.: Selecting the right interestingness measure for association patterns. In: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. KDD '02, New York, NY, USA, ACM (2002) 32–41
16. Geng, L., Hamilton, H.J.: Interestingness measures for data mining: A survey. ACM Comput. Surv. **38**(3) (September 2006)