COPPER - Constraint OPtimized Prefixspan for Epidemiological Research

Agustin Guevara-Cogorno\textsuperscript{a}, Claude Flamand\textsuperscript{b}, Hugo Alatrista-Salas\textsuperscript{a,∗}

\textsuperscript{a}Pontificia Universidad Católica del Perú, Department of Engineering, GRPIAA Labs., Lima, Peru
\textsuperscript{b}Institut Pasteur in French Guiana, Epidemiology unit, 23 avenue Louis Pasteur BP 6010, 97306 Cayenne, French Guiana

Abstract
Sequential pattern mining, is a data mining technique used to study the temporal evolution of events describing a complex phenomenon. This technique has a limited application due to the high number of common sequences generated by dense datasets. To tackle this problem, we propose COP, an extension of the PrefixSpan algorithm oriented towards optimizing the relevance of the results obtained in the sequential patterns mining process. Indeed, we use multiple and simultaneous constraints that represent the expertise of researchers in a specific domain. Experiments conducted on datasets associated to dengue epidemic monitoring show an improve in result relevance from an expert’s point of view, as well as, a considerable speed gains for mining dense datasets.

Keywords: Healthcare Risk Management, Epidemiological Databases, Constraints, Sequential Patterns Mining

1. Introduction
Certain epidemiological databases can be described as spatiotemporal data, in which, spatial information (e.g., the location of an event) and temporal information (e.g., the date of the event) are included. Data mining methods must be able to fulfill efficiently the new needs generated by this kind of data. In this context, sequential patterns mining provide very relevant solutions through identification without \textit{a priori} hypothesis of relationships between variables and events characterized in time. For instance, in the context of dengue fever epidemics, public health experts know that, the evolution of dengue depends on the interactions between environmental factors, humans and transmission vector (e.g., the mosquito that carries the disease). These characteristics can be represented by a sequence of events and can be studied globally using sequential pattern mining to extract temporal correlations.

Two main strategies have been proposed to extract sequential patterns. One of them, is based on the construction of a prefix-tree and is performed using an outperforming successive projection of the original database. Nevertheless, there are datasets with particular characteristics that are difficult to mine due to their statistical properties. In addition, real world events result in an exponential number of common patterns, in some cases, obfuscating interesting patterns

\footnotesize{∗} Corresponding author. Tel.: +51-1-62-62000 ext. 4829
E-mail address: halatrista@pucp.pe
to the experts. Thus, we propose an extension of PrefixSpan algorithm designed to the extraction of restricted sequential patterns from spatiotemporal datasets. Our proposal called COP (Constraint Optimized PrefixSpan), was born to extract sequential patterns including expert knowledge in form of constraints like the inclusion of interesting events, among others. Additionally, to highlight the importance of our approach, we have tested COP on a spatiotemporal database associated to the epidemiological monitoring of dengue fever.

This manuscript is organized as follows: in Section 2, we review existing algorithms of data mining and we show that these methods are not suitable for our problem. Next, in Section 3, we detail the theoretical framework around COP approach. In Section 4 we present the COP algorithm and suggested implementation details. Toward the end of this article, in Section 5, we present our experiments on dengue dataset. The paper ends with conclusions and future works.

2. Related works

The problem of sequential patterns mining was introduced by Agrawal et al. in the context of the basket market problem and applied with success in many fields such as biology, Web mining, anomaly detection or the description of behavior into group. Furthermore, authors as Pei et al. or Liu have studied temporal sequences, which only take into account the temporal dimension. Later, Tsoukatos et al. have extended these works in order to represent sets of environmental features evolving in time. They extract sequences of characteristics that appear frequently in areas. Recently, in the context of epidemics, Alatrista et al. have been extended the PrefixSpan algorithm in order to capture the spatial relationships between studied areas. They have proposed a new spatiotemporal patterns called spatio-sequential patterns.

If these methods can be very relevant for epidemiological surveillance, they do not capture the interaction of characteristics describing an epidemic including relevant events as constraints, limiting the minimum and/or the maximum size of sequences and restricting the minimum and/or the maximum number of events treated by time stamp.

In this paper, we propose COP, a method for extracting sequential patterns (i.e., sequences of sets of events) under constraints, applied to the study of epidemiological monitoring. For instance, thanks to our proposal, we can refine the extraction process to produce only patterns where there is a combination of temperature variables involved and a high number of dengue cases. This improvement, reduces execution time greatly, as well as, increases the relevance of the resulting patterns.

3. Theoretical Framework

This section is divided in two parts. First, we slightly introduce certain concepts associated to the general problem of sequential patterns mining.

**Item and Itemset:** Let I be an item, a literal value that represents an event, e.g., the high temperature. An itemset, \( IS = \{I_1, I_2, ..., I_p\} \), is a non empty set of items, which represents a set of events occurred in a timestamp in an specific studied area. For instance, in Lima city, on September 12, a high temperature, a high humidity and four cases of dengue have been measured. All items in an itemset are associated with attributes in a database. An itemset with k items is called k-itemset.

**Sequence:** A sequence \( S \) is an ordered list of itemsets, denoted \( S = \langle IS_1IS_2...IS_n \rangle \) where \( IS_1, IS_{i+1} \) satisfy the constraint of temporal sequentiality. A sequence \( S = \langle IS_1IS_2...IS_n \rangle \) is included, denoted \( \subseteq \), in another sequence \( S' = \langle IS'_1IS'_2...IS'_m \rangle \) iff \( n \leq m \wedge \exists i_1 < i_2 < ... < i_n \in \mathbb{Z} \) such that \( IS_1 \subseteq IS'_{i_1}, IS_2 \subseteq IS'_{i_2}, ..., IS_n \subseteq IS'_{i_n} \).

**Prefix and Suffix of a Sequence:** We define the function \( \text{prefix}(S, N) \rightarrow S \) where \( S \) is a set of sequences, \( N \) is a set of positive integers, and \( \text{prefix}(s, k) = s[1 : k] \). In other words, \( \text{prefix}(s, k) \) returns the first \( k \) items, i.e., the prefix of \( s \). In the same way, we define the function \( \text{suffix}(S, N) \rightarrow S \) where \( S \) is a set of sequences. Let \( s \in S \) with \( n \) items and \( s' \in S \) with \( m \) items, if \( s' \) is a prefix of \( s \) then the \( \text{suffix}(s, s') = s[m+1 : n] \) returns the sequence of \( n - m \) last items of the sequence \( s \) prefixed by \( s' \).

**Minimum Support Threshold:** We define the support of a sequence \( S \) as the number of sequences in the database that include \( S \). Given a positive integer \( \sigma \), a sequence can be considered frequent if its support is greater than or equal to \( \sigma \). We call \( \sigma \) the minimum support threshold.

Now, we introduce new definitions associated to our proposal.
Indicator Function over Sequences: We define an indicator function over sequences, denoted $\mathbb{I}_S$.

$$\mathbb{I}_S(x) = \begin{cases} 1 & \text{if } \exists k \text{ such that } x \in IS_k \land \langle IS_k \rangle \subseteq S \\ 0 & \text{otherwise} \end{cases}$$

Constraint: A constraint is a function $f : S \rightarrow B$, where $S$ is a set of sequences and $B$ is the boolean domain. A sequence $S_i$ is said to satisfy a constraint if $f(S_i) = 1$. It follows that a constraint can be constructed from multiple constraints through the application of multiplication to the results. $g : S \rightarrow B$ where:

$$g(S_i) = \prod_{n=1}^{n} f_n(S_i)$$

and $f_i$ denotes a constraint $\forall i$.

Soft Inclusion Constraint: A soft inclusion constraint $C_e$, where $e = e(x_1, x_2, \ldots, x_i)$ is a propositional formula, $\forall k \in [1..i]$, $\exists k'$ such that $k \in \text{dom}(D_{A_e})$.

$$C_e(S) = \begin{cases} 1 & \text{if } e(\mathbb{I}_S(x_1), \mathbb{I}_S(x_2) \ldots \mathbb{I}_S(x_n)) \text{ evaluates to true} \\ 0 & \text{otherwise} \end{cases}$$

Additionally, consider a reduced subset of propositional formulas characterized by being a number of finite conjunctions and disjunctions. If a soft inclusion constraint is defined over this subset we define it as a Positive Soft Inclusion Constraint.

Sequence Size Constraint: A sequence size constraint $C_{n,m}$, $n \leq m, \forall n, m \in \mathbb{N}$, over a sequence $S = \langle IS_1 IS_2 \ldots IS_k \rangle$ is defined as the function:

$$C_{n,m}(S) = \begin{cases} 1 & \text{if } n \leq k \leq m \\ 0 & \text{otherwise} \end{cases}$$

Itemset Size Constraint: An itemset size constraint $C_{n,m}$, $n \leq m, \forall n, m \in \mathbb{N}$, over a sequence $S = \langle IS_1 IS_2 \ldots IS_i \rangle$ is defined as the function:

$$C_{n,m}(S) = \begin{cases} 1 & \text{if } n \leq \min(IS_{s1,i}) \land \max(IS_{s1,i}) \leq m \\ 0 & \text{otherwise} \end{cases}$$

4. PrefixSpan Under Constraints (COP)

PrefixSpan is a pattern-growth method for mining sequential patterns. The major idea is to perform successive projections of the database, by considering all the possible occurrences of frequent subsequences. This projection is based only on frequent prefixes because any frequent subsequence can always be found by growing a frequent prefix. In this article, we propose COP, an extension of the PrefixSpan algorithm for sequential pattern mining, which is detailed in this section. The main idea is to generate an interesting (w.r.t., the expert point of view) subset of sequential patterns, which satisfy a set of constraints, reducing the search space both, in preprocessing and at each step of execution. This allows us to reduce the running time and increasing the result relevance. We focus on the satisfaction of constraints composed of any combination of positive soft inclusion, pattern size and itemset size constraints. These constraints, have been applied on two steps of the general process.

1. Preprocessing: In order to optimize running time, we consider the possibility of pruning the database before the algorithm execution by eliminating all sequences which cannot support the constraint-satisfying frequent sequences. Then, we apply the positive soft inclusion constraint and the minimum sequence size constraint to the database to remove all sequences that do not satisfy the constraints from it. During this process, we also generate the set of unique elements in sequences, which we denote $E$ for reasons which will be seen later. Preprocessing is advantageous when dealing with small number of large sequences, barring highly uniform sequences. The number of removed sequences is small since the odds of any one sequence lacking all the required conditions is small.
2. **Mining process:** PrefixSpan generates candidate elements, which we denote as $b$, for construction of new sequences $\alpha'$ through a scanning of the database $S$. Later PrefixSpan produces the projected database $S|_{\alpha'}$ and executes the recursive call. A recommended modification of PrefixSpan by the original authors is through the use of pointers for the generation of projections so as to reduce the memory consumption. In this context, we propose a way of consolidating the scanning for unique elements with the projection on them through the use of a set of unique elements $E$. It can be trivially shown that this set is non-increasing in each subsequent call. Once an element from the set fails to meet the threshold it can be removed from all further projections. At every step the set is updated by generating $S|_{\alpha+b}$, removing the empty sequences and comparing it to the support threshold. If it does not meet it, $b$ is removed from the set for all further calls, otherwise, it fixes the support threshold we store $b$ with its associated projected database. This set can be generated at preprocessing while the sequences are loaded into memory requiring no additional time during the execution of the algorithm itself.

The application of constraints is done through post-processing and pruning. For pruning step, we consider the pattern size constraint and the sequence size constraint. Given a candidate sequence $S'$, if $S'$ fails to satisfy the minimum itemset size constraint, then, we only consider $b$ which can be assembled. If $S'$ would fail to satisfy the maximum itemset size constraint after an assembly, we only consider $b$ which can be appended. If $S'$ would fail to meet the maximum sequence size constraint after an appending, we only consider $b$ which can be assembled.

Finally we only consider the remaining constraints to decide whether to output a candidate sequence or not. The resulting algorithm associated to COP is detailed in Algorithm 1.

**Algorithm 1** Constraint Optimised PrefixSpan ($\alpha, L, E, S|_{\alpha'}$)

1: if $\min(\text{IS}_s)_{s\in\alpha} < \min\text{ItemsetSize} \lor |\alpha| + 1 > \max\text{SequenceSize}$ then
2:   for $b \in E$ do
3:     Construct $S|_{\alpha'}$ of only $b$ which are assemblies not yet in the last itemset in $\alpha$
4:     if $|S|_{\alpha'}| < \text{SupportThreshold}$ then
5:       $G[b] = S|_{\alpha'}$
6:     end if
7:   end for
8: end if
9: if $\max(\text{IS}_s)_{s\in\alpha} + 1 > \max\text{ItemsetSize}$ then
10:   for $b \in E$ do
11:     Construct $S|_{\alpha'}$ of only $b$ which are appending
12:     if $|S|_{\alpha'}| < \text{SupportThreshold}$ then
13:       $G[b] = S|_{\alpha'}$
14:     else
15:       $E = E - b$
16:     end if
17:   end for
18: end if
19: for $b \in G$ do
20:   $\alpha' = \alpha + b$
21:   if $\alpha'.\text{valid}()$ then
22:     output $\alpha'$
23:   end if
24: Call PrefixSpan ($\alpha, L+1, E, G[b]$)
25: end for

5. **Experiments and Results**

In this section, we present the dataset at our disposal, the experiments conducted in this work and the results.

5.1. **Epidemic Dataset**

Dengue is a mosquito-borne infection (Aedes aegypti) that occurs in all tropical and subtropical regions of the planet. Currently, dengue viral infection has become an increasing global health concern with over two-fifths of the
Table 1. Sample Database Entries

<table>
<thead>
<tr>
<th>Territory</th>
<th>ID</th>
<th>Week</th>
<th>Rainfall mean</th>
<th>Temp.Mean</th>
<th>Mean</th>
<th>Wind Force</th>
<th>Confirmed Incidence x1000hab</th>
</tr>
</thead>
<tbody>
<tr>
<td>IDC</td>
<td>25</td>
<td>2006-25</td>
<td>18.2286</td>
<td>25.0857</td>
<td>5.85714</td>
<td>0.097909629</td>
<td></td>
</tr>
<tr>
<td>IDC</td>
<td>26</td>
<td>2006-26</td>
<td>10.3143</td>
<td>25.1</td>
<td>5.71429</td>
<td>0.146864444</td>
<td></td>
</tr>
<tr>
<td>Kourou</td>
<td>26</td>
<td>2006-26</td>
<td>14.8286</td>
<td>25.1714</td>
<td>5.14286</td>
<td>0.048954815</td>
<td></td>
</tr>
<tr>
<td>Kourou</td>
<td>27</td>
<td>2006-27</td>
<td>3.6</td>
<td>26.2571</td>
<td>5.28571</td>
<td>0.019581926</td>
<td></td>
</tr>
<tr>
<td>SLM</td>
<td>25</td>
<td>2006-25</td>
<td>8.88571</td>
<td>25.2857</td>
<td>NULL</td>
<td>0.048954815</td>
<td></td>
</tr>
<tr>
<td>SLM</td>
<td>26</td>
<td>2006-26</td>
<td>11.8571</td>
<td>25.2714</td>
<td>NULL</td>
<td>0.048954815</td>
<td></td>
</tr>
</tbody>
</table>

worlds population at risk of infection. It is the most rapidly spreading vector borne disease, attributed to demographics, urbanization and environment changing. The severe dengue (formerly known as dengue hemorrhagic fever) was first identified in the fifties of last century during an outbreak in the Philippines and Thailand.

Recent estimates indicate between 50 and 100 million infections per year worldwide. In French Guiana, France's overseas territory in South America with 230,000 inhabitants, the epidemiology of dengue evolved from an endemico-epidemic to a hyper-endemic state. Four major epidemics linked to the circulation of 1 or 2 predominant serotypes have occurred between 2006 and 2013. These outbreaks usually last for 6 – 12 months and may affect nearly 10% of the population.

In the context of a collaboration between the epidemiology unit of the Pasteur Institute in French Guiana, we have analyzed data associated with epidemiological monitoring of dengue. These data were collected from the Regional epidemiology unit for the public health surveillance in a territory divided into 3 territories located in the coastal area. This dataset contains information associated to: population data, meteorological data, epidemiological data among others. Overall, the dengue dataset contains 417 dates by zone corresponding to a week of monitoring, and a maximum of 22 items by date.

5.2. Experiments protocol

The original epidemiological database was reduced from 22 variables to 4 per date to preserve the problematic properties of the database while reducing the effect of constraints on non-relevant characteristics to prevent incidental variable effects on the results. The goal of these experimentations is to proof that our proposal and the constraints defined in Section 3. A sample reduced entry can be seen in Table 1.

The data was discretized into 5 equal depth partitions per variable through the use of the Weka suite and the simple equal-depth partitioning filter. All the experiments were performed on an Intel(R) Core(TM) i7-3779 CPU processor, with 3.40GHz clock speed, and 32.00 GB of RAM on a 64-bit Windows 7. All the methods were implemented on International Standard ISO/IEC 14882:2014(E) Programming Language C++.

The algorithm was first executed with each individual numerical constraint taking values from 2 to 8, it was then executed for all numerical constraints and an inclusion constraint for the presence of the highest partition of the number of dengue cases, finally all numerical constraints were tested in conjunction with the previous inclusion constraint and the previously best performing constraint of minimum itemset size of 4. Every test case was repeated 30 times and the average time taken is reported. All cases resulting in a time-out were omitted.

It is important to notice that, the traditional PrefixSpan algorithm did not terminate after 3 days of continuous exclusive execution at an 100% threshold, which made impossible a comparison with our proposal.

5.3. Results

The failure of the performance of the traditional PrefixSpan algorithm, highlights the importance of our proposal. Our experimental results showed that under an increased number of constraints, the convergence rate of the algorithm under the time-out limit increased. Furthermore, the number of results decreased resulted in a higher relevance for the resulting candidate space needing to be analyzed by specialists. The execution time cannot be conclusively said to be reduced by single composition of constraints, due to the algorithmic costs of checking the constraints with no benefit on the pruning process. The increase in convergence can be explained by the fact that the composition of a quickly
convergent constraint with any other given constraint will have an upper time bound given by the fastest convergent constraint (within error margin).

In regards to the epidemiological dataset used for experiments, only the minimum itemset size fixed to 4 and the maximum pattern size fixed to 2, provide results in the desired time frame. Fixing the maximum itemset size to 4 is equivalent to requiring whole itemset matches which effectively would make mining the database equivalent to mining a unidimensional itemset database, which simplifies the process greatly. Forcing desired patterns to be of size 2 greatly reduces the number of candidate to be tested as frequent. This effectively limits the resulting candidates to a form equivalent to a simple two part association rules.

6. Conclusion & Perspectives

In this paper, we propose an extension of the well-known PrefixSpan algorithm for the direct application of a priori domain knowledge to the pattern mining process for an increase in result relevance and a reduction in execution times. Experimental results on real dengue datasets prove the efficiency of our proposal in terms of increasing relevance and in terms of an increase of convergence under time windows. The question of the precise effect of constraint types remains open however.

Among possible future developments of our work, we consider the possibility of first trying to mine only a compact set of spatiotemporal patterns using, for instance, the notion of top- \( k \) patterns or closed patterns. Further considerations include filtering interesting patterns integrating a declarative patterns mining approach for spatiotemporal databases or exploiting temporal characteristics of data integrating a temporal gap in the patterns mining process. Moreover, it is important to analyze the patterns and to provide a visualization tool to help decision makers.

Acknowledgements

We would like to acknowledge and thank the Regional epidemiology unit of the French Institute for Public Health Surveillance for the data provided that was used in the experimental portion of this work. Additionally, this paper was written in the context of project financed by FONDECYT.

References