MINING TIMED SEQUENCES WITH TOM4L FRAMEWORK

Nabil Benayadi and Marc Le Goc
LSIS Laboratory, University Saint Jerome, Marseille, France

Keywords: Sequential patterns, Information-theory, Temporal knowledge Discovering, Chronicles models, Markov processes.

Abstract: We introduce the problem of mining sequential patterns in large database of sequences using a Stochastic Approach. An example of patterns we are interested in is : 50% of cases of engine stops in the car are happened between 0 and 2 minutes after observing a lack of the gas in the engine, produced between 0 and 1 minutes after the fuel tank is empty. We call this patterns "**signatures**". Previous research have considered some equivalent patterns, but such work have three mains problems : (1) the sensibility of their algorithms with the value of their parameters, (2) too large number of discovered patterns, and (3) their discovered patterns consider only "after" relation (succession in time) and omit temporal constraints between elements in patterns. To address this issue, we present TOM4L process (Timed Observations Mining for Learning process) which uses a stochastic representation of a given set of sequences on which an inductive reasoning coupled with an abductive reasoning is applied to reduce the space search. The results obtained with an application on very complex real world system are also presented to show the operational character of the TOM4L process.

1 INTRODUCTION

The aim of Timed Data Mining techniques is to discover temporal knowledge from a set of timed messages sequences.

The general context is given in the Figure 1: a dynamic process is monitored with a Monitoring Cognitive Agent (MCA) that writes timed messages in a database. The dynamic process can be a manufacturing process, a telecommunication network or web servers for example. The timed messages are concerned with alarms or warnings, or with the starting or the stopping of tasks. The "learning process" aims at discovering the temporal knowledge that characterize the behavior of the monitored dynamic process to improve its management. This problematic is nowadays crucial in most of the industrial and the service sectors.

In this paper, we introduce the problems of mining such a pattern : 50% of cases of engine stops in the car are happened between 0 and 2 minutes after observing a lack of the gas in the engine, produced between 0 and 1 minutes after the fuel tank is empty. We call this patterns "**signatures**". Finding signatures are valuable in many fields, for example, when targeting markets using DM (Direct Mail), market analysts can use signatures to learn what actions they should take and when they should act to inform their customers to buy. In the industrial domain, operators can use signatures to control and supervise the process variables before maintaining the process in an equilibrium state. Other applications include predicting disease, forecasting weather, if we find signature : 60% of storms go through area B between 1 and 3 days after they strike area A, we can take steps to cope a disaster in the area B. We propose in this paper the basis of the TOM4L process (Timed Observations Mining for Learning process) defined to discover signatures among timed messages in large database of sequences. TOM4L process avoids also the two remains problems of Timed Data Mining techniques: the sensibility of the Timed Data Mining algorithms with the value of their parameters and the too large number of generated patterns. TOM4L avoids these two problems with the use of a stochastic representation of a given set of sequences on which an inductive reasoning coupled with an abductive reasoning is applied to reduce the space search. The next section recalls the basis of the main Timed Data Mining techniques and presents a (very) simple illustrative example to show the main problems of previous approaches. Next, section 3 introduces the basis of the TOM4L process and the section 4 describes the results obtained with an application of the TOM4L process on very complex

Benayadi N. and Le Goc M. (2010).
MINING TIMED SEQUENCES WITH TOM4L FRAMEWORK.
In *Proceedings of the 12th International Conference on Enterprise Information Systems - Artificial Intelligence and Decision Support Systems*, pages 111-120
DOI: 10.5220/0002958401110120

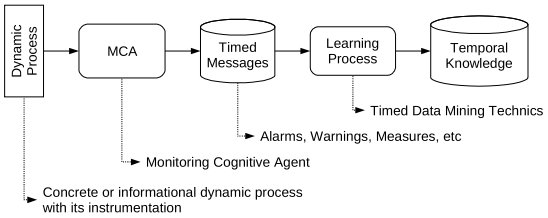

Figure 1: Temporal knowledge discovery context.

real world system monitored with a large scale knowledge based system, the Sachem system of the Arcelor-Mittal Steel group. The section 5 makes a synthesis of the paper and introduces our current works.

2 RELATED WORKS

Discovering temporal knowledge from the timed messages is a problem that can be studied from multiple points of view and a lot of scientific domains are concerned with this problem, specifically Machine Learning and Data Mining (cf. (Roddick and Spiliopoulou, 2002) for a complete state of the art).

The Timed Data Mining approaches aims at avoiding this problem. The basic principle consists in using a representativeness criteria, typically the support of a sequential pattern, to build the minimal set of sequential patterns that describes the given set of sequences. The support $s(p_i)$ of a pattern p_i is the number of sequences in the set of sequences where the pattern p_i is observed. A frequent pattern is a pattern p_i with a support $s(p_i)$ greater than a user defined thresholds $s(p_i) \geq S$. A frequent pattern is interpreted as a regularity or a condensed representation of the given set of sequences.

The Timed Data Mining techniques differs depending on whether the initial set of sequences is a singleton or not. The second case is the simpler because the decision criteria based on the support is directly applicable to a set of sequences. One of the first application can be found in the market basket analysis (Agrawal and Psaila, 1995) with the AprioriAll algorithm that has been improved with the SPAM (Ayres et al., 2002) or the SPADE (Zaki, 2001) algorithms.

When the initial set of sequences contains a unique sequence, the notion of windows has been introduced to define an adapted notion of support. The first way consists in defining a fixed size of windows that an algorithm like Winepi (Mannila et al., 1995) shifts along the sequence: the sequence becomes then a set of equal length sub-sequences and the support $s(p_i)$ of a pattern can be computed (Vilalta and Ma, 2002; Weiss and Hirsh, 1998). The second way consists in building a window for an a priori given pattern p_i. With the Minepi algorithm for example (Mannila and Toivonen, 1996), a window $W = [t_s, t_e[$ is a minimal occurrence of p_i if p_i occurs in W and not in any sub-window of W. In practice, a maximal window size parameter *maxwin* must be defined to bound the search space of patterns. A similar approach is proposed in (Dousson and Duong, 1999) to discover chronicle models, an abstract representation of patterns.

The Timed Data Mining approaches presents two main problems. The first is that the algorithms require the setting of a set of parameters: the discovered patterns depends therefore of the tuning of the algorithms (Mannila, 2002). The second problem is the number of generated patterns that is not linear with threshold value S of the decision criteria $s(p_i) \geq S$. In practice, to obtain an interesting set of frequent pattern, S must be small, and the number of frequent is huge ((Han and Kamber, 2006)). But generally, only a very small fraction of the discovered patterns are interesting. This leads to use interestingness measures to build a minimal set of frequent patterns having some potential to be significative. The mostly used interestingness measures are based on the Information theory (Shannon, 1949) like the j-measure (Smyth and Goodman, 1992) and the mutual information (Cover and Thomas, 1991). Let us take a simple example to illustrate these two basic problems of the Timed Data Mining approaches.

The illustrative example is a simple dynamic SISO system $y(t) = F \cdot x(t)$ where F is a convolution operator. This example is used trough this paper to illustrate the claims.

Let us defining two thresholds ψ_x and ψ_y for the input variable $x(t)$ and the output variable $y(t)$. These two thresholds respectively defines two ranges for each of the variables: $rx_0 =]-\infty, \psi_x]$, $rx_1 =]\psi_x, +\infty]$, $ry_0 =]-\infty, \psi_y]$ and $ry_1 =]\psi_y, +\infty]$). Let us suppose that there exists a (very simple) program that writes a constant when a signal enter in a range. Such a program writes the constant 1 (resp. H) when $x(t)$ (resp. $y(t)$) enters in the range rx_1 (resp. ry_1) and 0 (resp. L) when $x(t)$ (resp. $y(t)$) enters in the range rx_0 (resp. ry_0). The evolution of the $x(t)$ in the figure 2 leads to the following sequence: $\omega = \{(1,t_1), (H,t_2), (0,t_3), (L,t_4), (1,t_5), (H,t_6), (0,t_7), (L,t_8), (1,t_9), (H,t_{10}), (0,t_{11}), (L,t_{12}), (1,t_{13}), (H,t_{14}), (0,t_{15}), (L,t_{16}), (1,t_{17}), (H,t_{18}), (0,t_{19}), (L,t_{20}), (1,t_{21}), (H,t_{22}), (0,t_{23}), (L,t_{24})\}$.

To illustrate the sensibility of the Winepi and the Minepi algorithms with the parameters, we defines two sets of parameters and apply the algorithms to the sequence ω. In the first set of parameters, the window

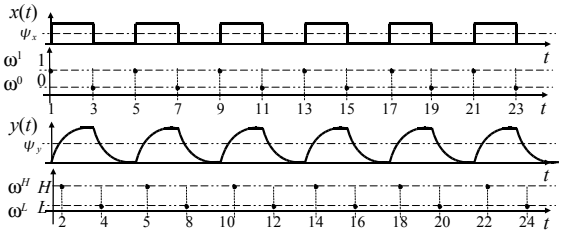

Figure 2: Temporal evolution of variables x and y.

width w and window movement v for Winepi are both set to 4 (this is the ideal tuning) and for Minepi, the max window is set to 4 and the minimal frequency is fixed to 6 (this is also the ideal tuning). In the second set of parameters, the window width and window movement of Winepi are equal to 8 and the support is equal to 3. The minimal frequency for Minepi is set to 8. The table 1 provides the number of patterns discovered by each algorithm with the two sets of parameters.

These two experimentations show the sensibility of the Winepi and the Minepi algorithms with the parameters: from the first set to the second, the number of patterns increase of more than 626% for Winepi, and more than 18666% from Minepi. The main problem is the too large number of discovered patterns. The paradox is then the following: to find the ideal set of parameters that minimizes the number of discovered patterns, the user must know the system while this is precisely the global aim of the Data Mining techniques. There is then a crucial need for another type of approach that is able to provide a good solution for such a simple system and provide operational solutions for real world systems. The aim of this paper is to propose such an approach: the TOM4L process (i.e. Timed Observation Mining for Learning) which find only 4 relations with the example without any parameters.

Table 1: Number of discovered patterns.

	Winepi	Minepi
First parameter set	15	15
Second parameter set	94	2800

3 FINDING SIGNATURES

The TOM4L process is based on the Theory of Timed Observations of (Le Goc, 2006) that defines an inductive reasoning and an abductive reasoning on a stochastic representation of a set of sequences $\Omega = \{\omega_i\}$, this set being or not a singleton.
This theory provides the mathematical foundations of the four steps Timed Data Mining process of Figure 3 that reverses the usual Data Mining process in order to minimize the size of the set of the discovered patterns:

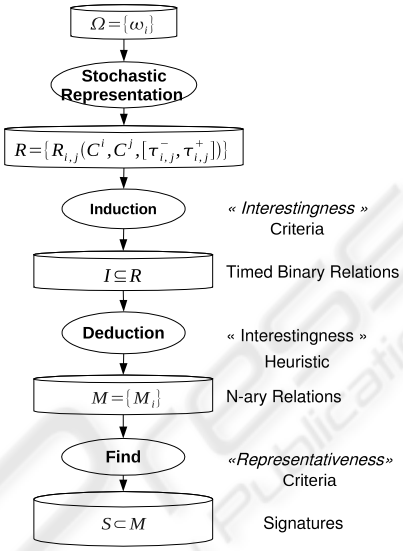

Figure 3: The four steps of TOM4L approch.

1. Stochastic Representation of a set of sequences $\Omega = \{\omega_i\}$. This step produces a set of timed binary relations of the form $R_{i,j}(C^i, C^j, [\tau^-_{i,j}, \tau^+_{i,j}])$.
2. Induction of a minimal set of timed binary relations. This step uses an interestingness criteria based on the BJ-measure describes in the following section.
3. Deduction of a minimal set of n-ary relations. This step uses an abductive reasoning to build a set of n-ary relations that have some interest according to a particular problem.
4. Find the minimal set of n-ary relations being representatives according to the problem. This step corresponds to the usual search step of sequential patterns in a set of sequences in Minepi or Winepi.

The discovered n-ary relations discovered in the last step are called signatures. The next section provides the basic definitions of the Timed Observations Theory.

3.1 Basic Definitions

A discrete event e_i is a couple (x_i, δ_i) where x_i is the name of a variable and δ_i is a constant. The constant δ_i denotes an abstract value that can be assigned to the variable x_i. The illustrative example allows the definition of a set E of four discrete events: $E = \{e_1 \equiv (x,1), e_2 \equiv (x,0), e_3 \equiv (y,H), e_4 \equiv (y,L)\}$.

A discrete event class $C^i = \{e_i\}$ is an arbitrary set of discrete event $e_i = (x_i, \delta_i)$. Generally, and this will be true in the suite of the paper, the discrete event classes are defined as singletons because when the constants δ_i are independent, two discrete event classes $C^i = \{(x_i, \delta_i)\}$ and $C^j = \{(x_j, \delta_j)\}$ are only linked with the variables x_i and x_j ((Le Goc, 2006)). The illustrative example allows the definition of a set Cl of 4 discrete event classes: $Cl = \{C^1 = \{e_1\}, C^0 = \{e_0\}, C^L = \{e_L\}, C^H = \{e_H\}\}$.

An occurrence $o(k)$ of a discrete event class $C^i = \{e_i\}$, $e_i = (x_i, \delta_i)$, is a triple (x_i, δ_i, t_k) where t_k is the time of the occurrence. An occurrence $o(k) \equiv (x_i, \delta_i, t_k)$ is called a timed observation in (Le Goc, 2006) because it can always be interpreted as the assignation of the abstract value δ_j to the variable x at time t_k (i.e. $o(k) \Leftrightarrow x(t_k) = \delta_i$). The idea is that a timed observation is supposed to be written by a program that implements the following specification:

$$\begin{array}{l} \exists t_{k-1}, t_k \in \Re, t_{k-1} < t_k, \\ x_i(t_{k-1}) \leq \psi_i \wedge x_i(t_k) > \psi_i \Rightarrow o(k) \equiv (x_i, \delta_i, t_k) \end{array} \quad (1)$$

When useful, the rewriting rule $o(k) \equiv (x_i, \delta_i, t_k) \equiv C^i(k)$ will be used in the following.

A sequence $\Omega = \{o(k)\}_{k=1\ldots n}$, is an ordered set of n occurrences $C^i(k) \equiv (x_i, \delta_i, t_k)$. The illustrative example defines the following sequence: $\Omega = \{(C^1(1), C^H(2), C^0(3), C^L(4), C^1(5), C^H(6), C^0(7), C^L(8), C^1(9), C^H(10), C^0(11), C^L(12), C^1(13), C^H(14), C^0(15), C^L(16), C^1(17), C^H(18), C^0(19), C^L(20), C^1(21), C^H(22), C^0(23), C^L(24)\}$. As a consequence, a sequence $\Omega = \{o(k)\}_{k=1\ldots n}$ defines:

- A set $K = \{k\}, k \in \aleph$, of time index.
- A set $\Gamma = \{t_k\}$, $t_k \in \Re$ of times generated by a continuous clock structure ($t_{k-2} - t_{k-1} \neq t_{k-1} - t_k$).
- A set $\Delta = \{\delta_i\}$ of constants.
- A set $X = \{x_i\}$ of variables.
- A set $E = \{e_i\}$ of discrete event $e_i = (x_i, \delta_i)$ defined on $X \times \Delta$.
- A set $Cl = \{C^i\}$ of discrete event classes (also called timed observation classes).

Le Goc (Le Goc, 2006) shows that when the constants $\delta_i \in \Delta$ are independent, a sequence $\Omega = \{o(k)\}$ defining a set $Cl = \{C^i\}$ of m classes is the superposition of m sequences $\omega^i = \{C^i(k)\}$:

$$\Omega = \{o(k)\} = \bigcup_{i=1\ldots m} \omega^i = \{C^i(k)\} \quad (2)$$

The Ω sequence of the illustrative example is then the superposition of four sequences $\omega^i = \{C^i(k)\}$:

$\omega^1 = \{C^1(1), C^1(5), C^1(9), C^1(13), C^1(17), C^1(21)\}$
$\omega^0 = \{C^0(3), C^0(7), C^0(11), C^0(15), C^0(19), C^0(23)\}$
$\omega^L = \{C^L(4), C^L(8), C^L(12), C^L(16), C^L(20), C^L(24)\}$
$\omega^H = \{C^H(2), C^H(6), C^H(10), C^H(14), C^H(18), C^H(22)\}$

3.2 Stochastic Representation

The stochastic representation transforms a set of sequences $\omega_i = \{o(k)\}$ in a Markov chain $X = (X(t_k); k > 0)$ where the state space $Q = \{q_i\}$, $i = 1\ldots m$, of X is confused with the set of m classes $Cl = \{C^i\}$ of $\Omega = \bigcup_i \omega_i$.

Consequently, two successive occurrences $(C^i(k-1), C^j(k))$ correspond to a state transition in X: $X(t_{k-1}) = q_i \longrightarrow X(t_k) = q_j$. The conditional probability $P[X(t_k) = q_j | X(t_{k-1}) = q_i]$ of the transition from a state q_i to a state q_j in X corresponds then to the conditional probability $P\left[C^j(k) \in \Omega | C^i(k-1) \in \Omega\right]$ of observing an occurrence of the class C^j at time t_k knowing that an occurrence of a class C^i at time t_{k-1} has been observed:

$$\begin{array}{l} \forall i, j, \forall k \in K, \\ P[X(t_k) = q_j | X(t_{k-1}) = q_i] = P\left[C^j(k) \in \Omega | C^i(k-1) \in \Omega\right] \\ \equiv p_{ij} = \frac{N_{ij}}{\sum_{l, l \neq i}^{m} N_{il}} \end{array}$$

The transition probability matrix $P = [p_{i,j}]$ of X is computed from the contingency table $N = [n_{i,j}]$, where $n_{i,j} \in N$ is the number of couples $(C^i(k), C^j(k+1))$ in Ω. The table 2 is the contingency table N of the sequence Ω of the illustrative example.

Table 2: Contingency table $N = [n_{i,j}]$ of Ω.

	C^1	C^0	C^H	C^L	Total
C^1	0	0	6	0	6
C^0	0	0	0	6	6
C^H	0	6	0	0	6
C^L	5	0	0	0	5
Total	5	6	6	6	23

The stochastic representation of a given set Ω of sequences is then the definition of a set $R = \{R_{i,j}(C^i, C^j, [\tau_{ij}^-, \tau_{ij}^+])\}$ where each the conditional probability $p_{i,j} = P\left[C^j(k) \in \Omega | C^i(k-1) \in \Omega\right]$ of each binary relation $R_{i,j}(C^i, C^j, [\tau_{ij}^-, \tau_{ij}^+])$ is not null. The timed constrains $[\tau_{ij}^-, \tau_{ij}^+]$ is provided by a function of the set D of delays $D = \{d_{ij}\} = \{(t_{k_j} - t_{k_j})\}$ computed from the binary superposition of the sequences $\omega^{i,j} = \omega^i \cup \omega^j$: $\tau_{ij}^- = f^-(D), \tau_{ij}^+ = f^+(D)$. For example, the authors of (Bouché, 2005) use the properties of the Poisson law to compute the timed constraints: $\tau_{ij}^- = 0, \tau_{ij}^+ = \frac{1}{\lambda_{i,j}}$ where $\lambda_{i,j}$ is the Poisson rate (i.e.

the exponential intensity) of the exponential law that is the average delay $d^{ij}_{moy} = \frac{\sum(d_{ij})}{Card(D)}$. But more frequently, a min-max approach is used (Dousson and Duong, 1999) : $\tau^-_{ij} = min(D_{ij}), \tau^+_{ij} = max(D_{ij})$. The set R of the illustrative example is the following: $R = \{R_{1,H}(C^1, C^H, [\tau^-_{1,H}, \tau^+_{1,H}]), R_{0,L}(C^0, C^L, [\tau^-_{0,L}, \tau^+_{0,L}]), R_{H,0}(C^H, C^0, [\tau^-_{H,0}, \tau^+_{H,0}]), R_{L,1}(C^L, C^1, [\tau^-_{L,1}, \tau^+_{L,1}])\}$.

3.3 Discrete Binary Memoryless Channel Model

Considering a binary relation $R_{i,j}(C^i, C^j, [\tau^-_{ij}, \tau^+_{ij}])$, a sequence Ω defining the set Cl of m classes with n occurrences contains $n-1$ couples $(o(k), o(k+1))$. Each of them is one of the four following types: $(C^i(k), C^j(k+1))$, $(C^i(k), \overline{C^j}(k+1))$, $(\overline{C^i}(k), C^j(k+1))$, and $(\overline{C^i}(k), \overline{C^j}(k+1))$, where $\overline{C^i}$ (resp. $\overline{C^j}$) is an abstract class denoting any classes of Cl but C^i (resp. C^j).

The $n-1$ couples $(o(k), o(k+1))$ can then be seen as $n-1$ realizations of one of the four relations linking two abstract binary variables X and Y of a discrete binary memoryless channel in a communication system according to the information theory (Shannon, 1949), where $X(t_k) \in \{C^i, \overline{C^i}\}$ and $Y(t_{k+1}) \in \{C^j, \overline{C^j}\}$ (Figure 4).

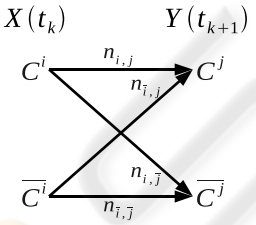

Figure 4: Two abstract binary variables connected by a discrete memoryless channel.

To use this model, the number of occurrences of the abstract classes $\overline{C^i}$ and $\overline{C^j}$ can not be the number of the occurrences of the classes $Cl - C^i$ and $Cl - C^j$ but an average value:

- $n_{i,j}$ is the number of couples $(C^i(k), C^j(k+1))$ in Ω.
- $n_{i,\overline{j}}$ is the average number of couples $(C^i(k), \overline{C^j}(k+1))$ in Ω:
  - $n_{i,\overline{j}} = \frac{1}{m-1} \sum_{\forall C^l \in \overline{C^i}} n_{i,l}$
- $n_{\overline{i},j}$ is the average number of couples $(\overline{C^i}(k), C^j(k+1))$ in Ω:
  - $n_{\overline{i},j} = \frac{1}{m-1} \sum_{\forall C^l \in \overline{C^i}} n_{l,j}$
- $n_{\overline{i},\overline{j}}$ is the average number of couples $(\overline{C^i}(k), \overline{C^j}(k+1))$ in Ω:
  - $n_{\overline{i},\overline{j}} = \frac{1}{(m-1)^2} \sum_{\forall C^l \in \overline{C^i}, \forall C^f \in \overline{C^j}} n_{l,f}$

This leads to $m \cdot (m-1)$ binary contingency tables of the form of the Table 3.

Table 3: Contingency table for X and Y.

X \ Y	C^j	$\overline{C^j}$	Σ
C^i	$n_{i,j}$	$n_{i,\overline{j}}$	$n_i = \sum_{y \in \{j,\overline{j}\}} n_{i,y}$
$\overline{C^i}$	$n_{\overline{i},j}$	$n_{\overline{i},\overline{j}}$	$n_{\overline{i}} = \sum_{y \in \{j,\overline{j}\}} n_{\overline{i},y}$
Σ	$n_j = \sum_{x \in \{i,\overline{i}\}} n_{x,j}$	$n_{\overline{j}} = \sum_{x \in \{i,\overline{i}\}} n_{x,\overline{j}}$	$N = \sum_{x \in \{i,\overline{i}\}, y \in \{j,\overline{j}\}} n_{x,y}$

These contingency tables allow computing two conditional probabilities matrix P^s (i.e. $P(Y(t_{k+1})|X(t_k))$) and P^p (i.e. $P(X(t_k)|Y(t_{k+1}))$(Table 4). These two matrix allow the definition of the BJ-measure to build a criteria to evaluate the interest of a binary relation $R_{i,j}(C^i, C^j, [\tau^-_{ij}, \tau^+_{ij}])$.

Table 4: P^s and P^p matrix.

P^s	C^j	$\overline{C^j}$
C^i	$p(C^j\|C^i) = \frac{n_{i,j}}{n_i}$	$p(\overline{C^j}\|C^i) = \frac{n_{i,\overline{j}}}{n_i}$
$\overline{C^i}$	$p(C^j\|\overline{C^i}) = \frac{n_{\overline{i},j}}{n_{\overline{i}}}$	$p(\overline{C^j}\|\overline{C^i}) = \frac{n_{\overline{i},\overline{j}}}{n_{\overline{i}}}$
P^p	C^j	$\overline{C^j}$
C^i	$p(C^i\|C^j) = \frac{n_{i,j}}{n_j}$	$p(C^i\|\overline{C^j}) = \frac{n_{i,\overline{j}}}{n_{\overline{j}}}$
$\overline{C^i}$	$p(\overline{C^i}\|C^j) = \frac{n_{\overline{i},j}}{n_j}$	$p(\overline{C^i}\|\overline{C^j}) = \frac{n_{\overline{i},\overline{j}}}{n_{\overline{j}}}$

3.4 Evaluating the Interestingness of Binary Relations

The idea for defining an efficient interestingness criteria to induce binary relations is that if knowing $C^i(k)$ increases the probability of observing $C^j(k+1)$ (i.e. $p(C^j|C^i) > p(C^j)$), then the observation $C^i(k)$ provides some information about an observation $C^j(k+1)$ (Blachman, 1968).

We propose then to use the distance of Kullback-Leibler $D(p(Y|X = C^i)\|p(Y))$ to evaluate the relation between the *a priori* distribution $p(C^j)$ of an observation $C^j(k)$ and the conditional distribution $p(C^j|C^i)$:

$$D(p(Y|X=C^i)\|p(Y)) = p(Y=C^j|X=C^i) \times log_2\left(\frac{p(Y=C^j|X=C^i)}{p(Y=C^j)}\right) + p(Y=C^{\overline{j}}|X=C^i) \times log_2\left(\frac{p(Y=C^{\overline{j}}|X=C^i)}{p(Y=C^{\overline{j}})}\right) \quad (3)$$

One of the property of this distance is that $D(p(Y|X=C^i)\|p(Y))=0$ when $p(Y=C^j|X=C^i)=p(Y=C^j)$. This property means that when the distributions $p(Y=C^j)$ and $p(X=C^i)$ are independent, the Kullback-Leibler distance is null. This allows to decompose the Kullback-Leibler distance in two terms.

Definition 1. *The BJL-measure $BJL(C^i,C^j)$ of binary relation $R(C^i,C^j)$ is the right part of the Kullback-Leibler distance $D(p(Y|X=C^i)\|p(Y))$:*

- $p(Y=C^j|X=C^i) < p(Y=C^j) \Rightarrow BJL(C^i,C^j) = 0$
- $p(Y=C^j|X=C^i) \geq p(Y=C^j) \Rightarrow BJL(C^i,C^j) = D(p(Y|X=C^i)\|p(Y))$

Considering the discrete memoryless binary channel (Figure 4), the $BJL(C^i,C^j)$ is not null when the observation $C^i(k)$ provides some information about the observation $C^j(k)$. Symmetrically, when $BJL(C^i,C^j)=0$, the observation $C^i(k)$ provides some information about any observations but $C^j(k)$, that is to say about an observation $\overline{C^j}(k)$. This leads to define the BJL-measure $BJL(C^i,\overline{C^j})$ of a binary relation $R(C^i,\overline{C^j})$:

Definition 2. *The BJL-measure $BJL(C^i,\overline{C^j})$ of a binary relation $R(C^i,\overline{C^j})$ is the left part of the Kullback-Leibler distance $D(p(Y|X=C^i)\|p(Y))$:*

- $p(Y=C^j|X=C^i) < p(Y=C^j) \Rightarrow BJL(C^i,\overline{C^j}) = D(p(Y|X=C^i)\|p(Y))$
- $p(Y=C^j|X=C^i) \geq p(Y=C^j) \Rightarrow BJL(C^i,\overline{C^j}) = 0$

This leads to the decomposition of the Kullback-Leibler distance property:

$$D(p(Y|C^i)\|p(Y)) = BJL(C^i,C^j) + BJL(C^i,\overline{C^j}) \quad (4)$$

Looking at the Figure 4, the definition of the BJL-measure decomposes the information provided by the assignation $X(t_k)=C^i$ (i.e. an observation $C^i(k)$) between the assignation $Y(t_{k+1})=C^j$ (i.e. the observation $C^j(k+1)$) and the assignation $Y(t_{k+1})=\overline{C^j}$ (i.e. the observation $\overline{C^j}(k+1)$). In other words, the BJL-measure evaluates the information distribution between the next successor ($C^j(k+1)$ or $\overline{C^j}(k+1)$) of an observation $C^i(k)$ at time t_k. The same reasoning can be done when considering the information distribution between the predecessors $X(t_k)=C^i$ or $X(t_k)=\overline{C^i}$ of the assignation $Y(t_{k+1})=C^j$:

Definition 3. *The BJW-measure $BJW(C^i,C^j)$ of binary relation $R(C^i,C^j)$ is the right part of the Kullback-Leibler distance $D(p(X|Y=C^j)\|p(X))$:*

- $p(X=C^i|Y=C^j) < p(X=C^i) \Rightarrow BJW(C^i,C^j) = 0$
- $p(X=C^i|Y=C^j) \geq p(X=C^i) \Rightarrow BJW(C^i,C^j) = D(p(X|Y=C^j)\|p(X))$

Symmetrically:

Definition 4. *The BJW-measure $BJW(\overline{C^i},C^j)$ of binary relation $R(\overline{C^i},C^j)$ is the left part of the Kullback-Leibler distance $D(p(X|Y=C^j)\|p(X))$:*

- $p(X=C^i|Y=C^j) < p(X=C^i) \Rightarrow BJW(\overline{C^i},C^j) = D(p(X|Y=C^j)\|p(X))$
- $p(X=C^i|Y=C^j) \geq p(X=C^i) \Rightarrow BJW(\overline{C^i},C^j) = 0$

Again, the BJW-measure decomposes the Kullback-Leibler distance $D(p(X|Y=C^j)\|p(X))$ in two terms:

$$D(p(X|Y=C^j)\|p(X)) = BJW(C^i,C^j) + BJW(\overline{C^i},C^j)$$

The BJW-measure evaluates then the information distribution between the predecessors ($C^i(k)$ or $\overline{C^i}(k)$) of an observation $C^j(k+1)$ at time t_{k+1}.

The BJL-measure evaluates the information that flows in two successor relations of a discrete memoryless binary channel (i.e. from $X(t_k)=C^i$ to $Y(t_{k+1})$, Figure 4) and the BJW-measure evaluates the information that flows in two predecessor relations (i.e. from $X(t_k)$ to $Y(t_{k+1})=C^j$). Because $(p(C^j|C^i) < p(C^j)) \Leftrightarrow p(C^i|C^j) < p(C^i))$, these two measures are null at the same independence point. The information flowing trough these four relations can then be combined in a single measure called the BJM-measure.

Definition 5. *The BJM-measure $BJM(C^i,C^j)$ of a binary relation $R(C^i,C^j)$ is the norm of the vector* $\begin{pmatrix} BJL(C^i,C^j) \\ BJW(C^i,C^j) \end{pmatrix}$ *:*

- $(p(C^j|C^i) \geq p(C^j)) \vee (p(C^i|C^j) \geq p(C^i)) \Rightarrow BJM(C^i,C^j) = \sqrt{BJL(C^i,C^j)^2 + BJW(C^i,C^j)^2}$
- $(p(C^j|C^i) < p(C^j)) \vee (p(C^i|C^j) < p(C^i)) \Rightarrow BJM(C^i,C^j) = -\sqrt{BJL(C^i,\overline{C^j})^2 + BJW(\overline{C^i},C^j)^2}$

The minus sign is used to build a monotonous measure that distinguishes the position of a relation $R(C^i,C^j)$ around the independence point. The BJM-measure $BJM(C^i,C^j)$ of a relation $R(C^i,C^j)$ is then simply:

$$BJM(C^i,C^j) = \sqrt{BJL(C^i,C^j)^2 + BJW(C^i,C^j)^2} - \sqrt{BJL(C^i,\overline{C^j})^2 + BJW(\overline{C^i},C^j)^2} \quad (5)$$

The maximum value $BJM(C^i,C^j)_{max}$ (obtained when $n_{i,j} = min(n_i,nj)$) and the minimum value of $BJM(C^i,C^j)_{min}$ (obtained when $n_{i,j} = 0$) depend on the ratio $\theta_{i,j} = \frac{n_i}{n_j}$. The comparison of two BJM-measures is not possible. To avoid this problem, the BJM-measure $BJM(C^i,C^j)$ is made linear with a M-measure $M(C^i,C^j)$ defined as follows:

Definition 6.

$$M(C^i,C^j) = \begin{cases} \frac{1}{2} \cdot \frac{BJM(C^i,C^j)}{BJM(C^i,C^j)_{max}} + \frac{1}{2} & \text{if } p(C^j|C^i) > p(C^j) \\ -\frac{1}{2} \cdot \frac{BJM(C^i,C^j)}{BJM(C^i,C^j)_{min}} + \frac{1}{2} & \text{else} \end{cases}$$

Whatever is the ratio $\theta_{i,j}$, the M-measure $M(C^i,C^j)$ as the following properties:

- $M(C^i,C^j) = 1 \Leftrightarrow BJM(C^i,C^j) = BJM(C^i,C^j)_{max}$ (ideal crisscross)
- $M(C^i,C^j) = 0,5 \Leftrightarrow BJM(C^i,C^j) = 0$ (C^i and C^j are independent)
- $M(C^i,C^j) = 0 \Leftrightarrow BJM(C^i,C^j) = BJM(C^i,C^j)_{min}$ (C^i and C^j are not linked)

For example, the values of the M-measure of the set $R = \{R_{1,H}(C^1,C^H,[\tau^-_{1,H},\tau^+_{1,H}]), R_{0,L}(C^0,C^L,[\tau^-_{0,L},\tau^+_{0,L}]), R_{H,0}(C^H,C^0,[\tau^-_{H,0},\tau^+_{H,0}]), R_{L,1}(C^L,C^1,[\tau^-_{L,1},\tau^+_{L,1}])\}$ of the illustrative example are given in table 5. This table shows that all the relations of R are ideally mixed.

Table 5: Matrix M.

	C^1	C^0	C^H	C^L
C^1	0	0	1	0
C^0	0	0	0	1
C^H	0	1	0	0
C^L	1	0	0	0

3.5 Inducing Binary Relations

In this example, the relations $R_{1,H}(C^1,C^H,[\tau^-_{1,H},\tau^+_{1,H}])$ and $R_{0,L}(C^0,C^L,[\tau^-_{0,L},\tau^+_{0,L}])$ have not the same meaning than the relations $R_{H,0}(C^H,C^0,[\tau^-_{H,0},\tau^+_{H,0}])$ $R_{L,1}(C^L,C^1,[\tau^-_{L,1},\tau^+_{L,1}])$: only the two first are linked with the system $y(t) = Fx(t)$, the two latter being only sequential relation (i.e. the system computes the values of $y(t)$, not the values of $x(t)$).

To distinguish between these two kind of relations, the idea is to add noise in the initial set of sequences. To this aim, we defined the "noisy" observation class C^{err} the occurrences of which are randomly timed. If a relation $R_{i,j}(C^i,C^j,[\tau^-_{i,j},\tau^+_{i,j}])$ is a property of the system, then the time interval between the occurrences of the C^i and C^j classes will be more regular than if this relation is a purely sequential relation. The table 6 shows the values of the M-measures of the relations $R(C^1,C^H)$, $R(C^H,C^0)$, $R(C^0,C^L)$ and $R(C^L,C^1)$ with different rate $\lambda_{err} = \frac{n_{err}}{t_{24}-t_0}$ of noisy occurrences added in Ω. For example, the sequence ω with $\lambda_{err} = 18$ is the following: $\omega = \{C^1(1), C^H(2), C^{err}(3), C^0(4), C^{err}(5), C^L(6), C^{err}(7), C^{err}(8), C^{err}(9), C^1(10), C^H(11), C^0(12), C^L(13), C^1(14), C^{err}(15), C^H(16), C^{err}(17), C^0(18), C^L(19), C^{err}(20), C^1(21), C^H(22), C^{err}(23), C^0(24), C^L(25), C^{err}(26), C^1(27), C^{err}(28), C^H(29), C^{err}(30), C^0(31), C^L(32), C^{err}(33), C^1(34), C^{err}(35), C^H(36), C^{err}(37), C^{err}(38), C^{err}(39), C^0(40), C^L(41), C^{err}(42)\}$. The table 6 shows that when $\lambda_{err} \in \{12,24\}$, the binary relations $R(C^H,C^0)$ and $R(C^L,C^1)$ disappears. Naturally, when the noise is too strong ($\lambda_{err} = 30$), all the relations disappear: this means that at least one occurrence $C^{err}(k)$ is systematically inserted between two occurrences of the initial sequence Ω.

Table 6: The M values evolution with different λ_{err}.

λ_{err}	$R(C^1,C^H)$	$R(C^H,C^0)$	$R(C^0,C^L)$	$R(C^L,C^1)$
0	1	1	1	1
6	0.75	0.56	1	0.63
12	0.78	0	1	0
18	0.61	0	0.79	0
24	0.55	0	0.55	0
30	0	0	0	0

This example leads also to an operational property of the M-measure: when $\theta_{i,j} \gg 1$ or $\theta_{i,j} \ll 1$, one class plays the same role of a noisy class for the other. This situation arises in the two following cases:

- $n_{i,\overline{j}} \geq n_{i,j} \Rightarrow p(\overline{C^j}|C^i) \geq 0.5$. The $\overline{C^j}$ plays the role of a noisy class for the class C^i.
- $n_{\overline{i},j} \geq n_{i,j} \Rightarrow p(C^j|\overline{C^i}) \geq 0.5$. The $\overline{C^i}$ plays the role of a noisy class for the class C^j.

These two conditions are both evaluated when comparing the product $p(C^j|C^i) \cdot p(C^i|C^j)$ with $\frac{1}{2} \cdot \frac{1}{2}$: when $p(C^j|C^i) \cdot p(C^i|C^j) \leq \frac{1}{4}$, $M(C^i,C^j) \leq 0.5$ and the relation $R_{i,j}(C^i,C^j)$ can not be justified with the M-measure. Inversely, when $p(C^j|C^i) \cdot p(C^i|C^j) > \frac{1}{4}$, $M(C^i,C^j) > 0.5$ and the relation $R_{i,j}(C^i,C^j)$ has some interest from the point of view of the M-measure.

This leads to the following simple inducing rule that uses the M-measure as interestingness criteria:

$$M(C^i,C^j) > 0.5 \Rightarrow R_{i,j}(C^i,C^j) \in I \quad (6)$$

3.6 Deduction of N-ary Relations

The set I of binary relations contains then the minimal subset of R where each relation $R_{i,j}(C^i,C^j)$ presents a potential interest. From this set, the M-measure can be used to build n-ary relations having some potential to be observed in the initial set Ω of sequences. To this aim, the M-measure is used in an heuristic $h(m^{i,n})$ that guides an abductive reasoning to build a minimal set $M = \{m^{k,n}\}$ of n-ary relations of the form $m^{k,n} = \{R_{i,i+1}(C^i,C^{i+1})\}$, $i = k,\cdots,n-1$, that is to say paths leading to a particular final observation class C^n. The heuristic $h(m^{i,n})$ makes a compromise between the generality and the quality of a path $m^{i,n}$:

$$h(m^{i,n}) = card(m^{i,n}) \times BJL(m^{i,n}) \times P(m^{i,n}) \quad (7)$$

In this equation, $card(m^{i,n})$ is the number of relations in $m^{i,n}$, $BJL(m^{i,n})$ is the sum of the BJL-measures $BJL(C^{k-1},C^k)$ of each relation $R_{k-1,k}(C^{k-1},C^k)$ in $m^{i,n}$ and $P(m^{i,n})$ is the product of the probabilities associated with each relation in $m^{i,n}$:

- $BJL(m^{i,n}) = \sum_{k=card(m^{i,n}),\cdots,1} BJL\left(R(C^{k-1},C^k)\right)$
- $P(m^{i,n}) = \prod_{i=card(m^{i,n}),\cdots,1} p(C^k|C^{k-1})$

$P(m^{i,n})$ corresponds to the Chapmann-Kolmogorov probability of a path in the transition matrix $P = [p(k-1,k)]$ of the Stochastic Representation. The interestingness heuristic $h(m^{i,n})$ being of the form $\phi \cdot \ln(\phi)$, it can be used to build all the paths $m^{i,n}$ where $h(m^{i,n})$ is maximum (Benayadi and Le Goc, 2008). For the illustrative example, the deduction step found a set M of two binary relations $(M = I)$[1].

3.7 Find Representativeness N-ary Relations

Given a set $M = \{m^{k,n})\}$ of paths $m^{k,n} = \{R_{i,i+1}(C^i,C^{i+1})\}$, $i = k,\cdots,n-1$, the TOM4L process uses two representativeness criterion to build the subset $S \subseteq M$ containing the only paths $m^{k,n}$ being representative according the initial set Ω of sequences. These criterion are a timed version of support and confidence notions:

Definition 7. ***Anticipation Rate.***
The anticipation rate $Ta(m^{i,n})$ of a n-ary relation $m^{i,n}$ is the ratio between the number of instances of $m^{i,n}$ in Ω with the number of occurrences of the $m^{i,n-1}$ (i.e. the n-ary relation $m^{i,n}$ without the last binary relation $R_{n-1,n}(C^{n-1},C^n)$).

Definition 8. ***Cover Rate.***
The cover rate $Tc(m^{i,n})$ of a n-ary relation $m^{i,n}$ is the ratio between the number of occurrences of $m^{i,n}$ with the number of occurrences of the final class C^n of the n-ary relation $m^{i,n}$.

The anticipation rate $Ta(m^{i,n})$ and the cover rate $Tc(m^{i,n})$ are criterion that allow to define an interestingness criteria to find interesting n-ary relations $m^{i,n}$ that are called "Signatures":

Definition 9. ***Signature.***
An n-ary relation $m^{i,n}$ is a signature if and only if $Tc(m^{i,n}) \geq C$ and $Ta(m^{i,n}) \geq A$, where $C \in [0,1] \subset \Re$ and $A \in [0,1] \subset \Re$.

Given a set of sequences (typically Ω) and the values of A and C, the "BJT4S" algorithm computes all the anticipation rate $Ta(m^{i,n})$ and the cover rate $Tc(m^{i,n})$ of each sub-paths $m^{k,n}$, $k \geq i$, of each paths $m^{i,n}$ of M to build the set S of signatures that satisfy the conditions $Tc(m^{k,n}) \geq C$ and $Ta(m^{k,n}) \geq A$. To this aim, the "BJT4S" algorithm represents the sub-paths $m^{k,n}$ in DEVS models and uses an abstract chronicle recognition engine to compute the corresponding anticipation rate $Ta(m^{k,n})$ and the cover rate $Tc(m^{k,n})$ (Le Goc et al., 2006). The complexity of this algorithm is proportional with the number of sub-paths and the size of the sequence so the smallest the set $M = \{m^{k,n})\}$ is (i.e. the most efficient the interestingness heuristic $h(m^{i,n})$ is) , the faster the execution of the BJT4S algorithm is. For example, the values of the cover rate and the anticipation rate of both binary relations of M of the illustrative example are 100%. So, $S = M$, $S = \{R_{1,H}(C^1,C^H,[\tau^-_{1,H},\tau^+_{1,H}])$, $R_{0,L}(C^0,C^L,[\tau^-_{0,L},\tau^+_{0,L}])\}$. These signatures are the only relations (patterns) that are linked with the system $y(t) = Fx(t)$. Comparing with the set of patterns found by Apriori-like approaches, we can confirm from this illustrative example that TOM4L approach converges towards a minimal set of operational relations, which describe the dynamic of the process. In the next section, we present the application of TOM4L on a sequence generated by a very complex dynamic process, blast furnace process. Due to the process complexity, we can confirm, without experience, that Apriori-like approaches fail to mine this sequence.

[1] No paths containing more than one binary relation can be deduced from I.

4 APPLICATION

Sachem is the name of the very large scale knowledge-based system the Arcelor-Mittal Steel group has developed at the end the 20th century to help the operators to monitor, diagnose and control the blast furnace, a very complex production process (Le Goc, 2006).

With a Sachem system, the blast furnace behavior is described with a series of occurrences of phenomenon classes that corresponds to the observation classes of TOM4L. The application is concerned with the *omega* variable that reveals the management quality of the whole blast furnace. The *omega* is a very abstract variable corresponding to the ratio of the number of carbon atoms used to produce a ton of hot metal with the number of iron (f_e) atoms it contains (the studied blast furnace produces 6,000 tons of hot metal per day). The values of *omega* are provided by a mathematical model which is a set of 17 differential equations linking together 53 high level variables synthesizing the whole the blast furnace behavior. This model is used to compute the ideal value of *omega* corresponding to a perfectly adjusted blast furnace: any distance from this ideal value means that the blast furnace is not well managed. In a set of expertise documents of 1995, the experts defines the variable modifications that cause the main modifications of *omega* (Figure 5, a): the top gas speed (TGS), the flame temperature (TF), the burden permeability (BD) and the size of the sinter (SS) through the burden permeability. The studied sequence comes

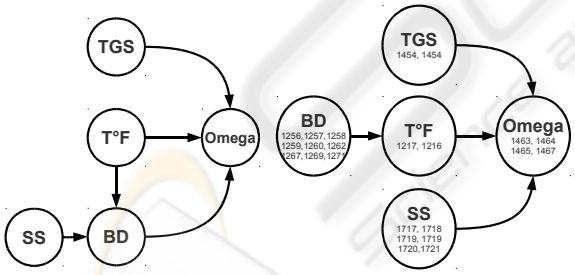

Figure 5: Expert's (1995, a) and discovered relations (2009, b).

from a blast furnace of Fos-Sur-Mer (France) from 08/01/2001 to 31/12/2001. It contains 7682 occurrences of 45 classes. For the 1463 class linked to the *omega* variable, the search space contains about $20^5 = 3,200,000$ binary relations. The inductive and the abductive reasoning steps of TOM4L produces a minimal set M of only 166 binary relations from which the set S of signatures of figure 6 have been discovered ($Ta = 50\%$ and $Tc = 10\%$). The set S is made with 50 binary relations.

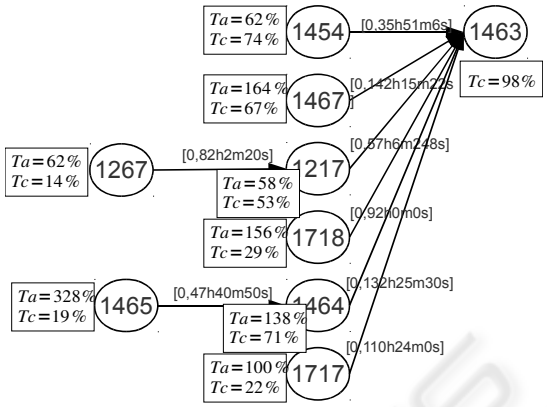

Figure 6: 1463 class signatures.

When substituting a class with its associated variable (the *omega* variable with the class 1464 for example) and the signatures of Figure 6 becomes the graph (b) of Figure 5 that contains the graph of the Expert's in 1995. The only difference is the direction of the relation between the variables TF and BD. This result shows that when pruning the branches bringing few information from a class to another, the BJ-measure allows to consider only the branches with a strong potentiality to be a signature: every signature Figure 6 have a strong credibility according to the laws governing the blast furnace. It is to note that the same result is observed on the Apache system, a clone of Sachem design to monitor and diagnose a galvanization bathe. As with the simple illustrative example of this paper, this result shows that the TOM4L process converges through a minimal set of binary relations with the elimination of the non interesting relations, despite of the complexity of the monitored process.

5 CONCLUSIONS

This paper presents the basis of the TOM4L process for discovering temporal knowledge from timed messages generated by monitored dynamic process. The TOM4L process is based on four steps: (1) a stochastic representation of a given set of sequences from which is induced (2) a minimal set of timed binary relations, and an abductive reasoning (3) is then used to build a minimal set of n-ary relations that is used to find (4) the most representative n-ary relations according to the given set of sequences. The induction and the abductive reasoning are based on an interestingness measure of the timed binary relations, that allows eliminating the relations having no meaning according to the given set of sequences. The results obtained

with an application on a very complex real world process (a blast furnace) are presented to show the operational character of the TOM4L process. These results provide new insights about the blast furnace behavior. So our current works are now focusing on the definition of a verity principle that is required to qualified the discovered relations.

REFERENCES

Agrawal, R. and Psaila, G. (1995). Active data mining. In Fayyad, Usama, M. and Uthurusamy, R., editors, *First International Conference on Knowledge Discovery and Data Mining (KDD-95)*, pages 3–8, Montreal, Quebec, Canada. AAAI Press, Menlo Park, CA, USA.

Ayres, J., Flannick, J., Gehrke, J., and Yiu, T. (2002). Sequential pattern mining using a bitmap representation. *KDD02: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 429–435.

Benayadi, N. and Le Goc, M. (2008). Using a measure of the crisscross of series of timed observations to discover timed knowledge. *Proceedings of the 19th International Workshop on Principles of Diagnosis (DX'08)*.

Blachman, N. M. (1968). The amount of information that y gives about x. *IEEE Transcations on Information Theory IT*, 14.

Bouché, P. (2005). *Une approche stochastique de modélisation de séquences d'événements discrets pour le diagnostic des systèmes dynamiques*. Thèse, Faculté des Sciences et Techniques de Saint Jérôme.

Cover, T. M. and Thomas, J. A. (August 12. 1991). *Elements of Information Theory*. Wiley-Interscience.

Dousson, C. and Duong, T. V. (1999). Discovering chronicles with numerical time constraints from alarm logs for monitoring dynamic systems. In *IJCAI : Proceedings of the 16th international joint conference on Artifical intelligence*, pages 620–626.

Han, J. and Kamber, M. (2006). *Data Mining: Concepts and Techniques*. Morgan Kaufmann.

Le Goc, M. (2006). *Notion d'observation pour le diagnostic des processus dynamiques: Application à Sachem et à la découverte de connaissances temporelles*. HDR, Faculté des Sciences et Techniques de Saint Jérôme.

Le Goc, M., Bouché, P., and Giambiasi, N. (2006). Devs, a formalism to operationnalize chronicle models in the elp laboratory, usa. In *DEVS'06, DEVS Integrative M&S Symposium*, pages 143–150.

Mannila, H. (2002). Local and global methods in data mining: Basic techniques and open problems. *29th International Colloquium on Automata, Languages and Programming*.

Mannila, H. and Toivonen, H. (1996). Discovering generalized episodes using minimal occurrences. In *Knowledge Discovery and Data Mining*, pages 146–151.

Mannila, H., Toivonen, H., and Verkamo, A. I. (1995). Discovering frequent episodes in sequences. In Fayyad, U. M. and Uthurusamy, R., editors, *Proceedings of the First International Conference on Knowledge Discovery and Data Mining (KDD-95)*, Montreal, Canada. AAAI Press.

Roddick, F. and Spiliopoulou, M. (2002). A survey of temporal knowledge discovery paradigms and methods. *IEEE Transactions on Knowledge and Data Engineering*, 14(4):750–767.

Shannon, C. E. (1949). Communication in the presence of noise. *Institute of Radio Engineers*, 37.

Smyth, P. and Goodman, R. M. (1992). An information theoretic approach to rule induction from databases. *IEEE Transactions on Knowledge and Data Engineering 4*, pages 301–316.

Vilalta, R. and Ma, S. (2002). Predicting rare events in temporal domains. In *ICDM02: Proceedings of the 2002 IEEE International Conference on Data Mining (ICDM02)*, page 474. IEEE Computer Society.

Weiss, G. M. and Hirsh, H. (1998). Learning to predict rare events in categorical time-series data. In *Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining, AAAI Press, Menlo Park, CA*.

Zaki, M. J. (2001). Spade: An efficient algorithm for mining frequent sequences. *Machine Learning*, 42:31–60.