MINING TIMED SEQUENCES TO FIND SIGNATURES

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

Keywords: Information-theory, Temporal knowledge discovering, Chronicles models, Markov processes.

Abstract: We introduce the problem of mining sequential patterns among timed messages 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. A very simple example is used to show the efficiency of the TOM4L process against others literature approaches.

1 INTRODUCTION

A ”Monitoring Cognitive Agent” (MCA) is a software system that aims at monitoring, diagnosing and controlling dynamic processes like manufacturing processes, telecommunication networks or web servers. The main task of an MCA is to analyze the sensor data provided by the instrumentation to inform about the observed behavior of the process with timed messages. Huge amounts of timed messages so collected in temporal databases (so-called ”event log”). There is an increasing interest in mining these timed messages to discover patterns that describe relations between the variables that govern the dynamic of the process and so improving its management.

In this paper, we introduce the problems of mining such a pattern: 50% of cases of engine stops in the car are happen 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. 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 sensitivity 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. In the literature, the common characteristic of techniques that mine sequences is the discovery of patterns that are frequents (Agrawal and Srikant, 1995), (Mannila et al., 1997): the more frequently a pattern occurs, the more likely a pattern is important. Mining sequential patterns was originally proposed for market analysis (Agrawal and Srikant, 1995) where the temporal relations between retail transactions are mined with the *AprioriAll* algorithm. This algorithm is based on a interestingness criteria called the ”support” of a sequential pattern, defined as the number of time a pattern is observed at least one time in a sequence. A pattern is then frequent when its support is greater than a given arbitrary threshold. Because this approach fails when there is only one sequence, two principal solutions have been proposed to gets around of this problem: the *Maximal window size constraint* solution and the *minimal occurrence* solution (Mannila et al., 1997). The Maximal win-

Benayadi N. and Le Goc M. (2010).
MINING TIMED SEQUENCES TO FIND SIGNATURES.
In *Proceedings of the 5th International Conference on Software and Data Technologies*, pages 450-455
DOI: 10.5220/0003007604500455

dow size constraint solution devises the sequence in set of sub-sequences so that a support can be computed (Winepi algorithm). Because the cutting of the sequence is arbitrary, Minepi algorithm is proposed that uses the *minimal occurrences* solution to define the windows. The problem with these "Frequential Approaches" is that the support allows to discover a lot of frequently observed patterns that are not representative of the relations between the process variables. So "informativeness" criteria are required to reduce the set of frequent patterns. The Stochastic Approach proposes to reverse this sequence mining process to first identify the potential interesting patterns before looking for frequently observed patterns.

The next section presents a simple illustrative example to show the main problems of previous approaches. Section 3 introduces the basis of the TOM4L process and the section 4 discusses and compares the results obtained by TOM4L process and others literature approaches on the illustrative example. The section 5 makes a synthesis of the paper and introduces our current works.

2 ILLUSTRATIVE EXAMPLE

Consider a system that monitors the stopping problem of a car. Figure 1 shows the structure of the monitored variables that might affect the stopping of a car. There are 6 variables ($x1$,$x2$,$x3$,$x7$,$x8$,$x9$) in the car system that can be assigned to following constants: $\Delta = \{x1 = \{Blown\}, x2 = \{Low\}, x3 = \{Empty\}, x7 = \{Off\}, x8 = \{False\}, x9 = \{Does\_Not\_Start\}\}$.

Let suppose that the car system was monitored for 30 minutes, this leads to the following sequence of 100 observations : $\omega = (Low, t_1)$, $(Empty, t_2)$,$(Empty, t_3)$,$(False, t_4)$,$(Does\_Not\_Start, t_5)$, $\cdots$, (Off, t_{98}), $(Empty, t_{99})$, $(Low, t_{100})\}$.

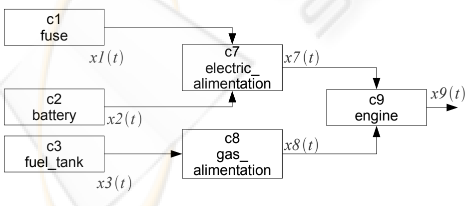

Figure 1: Temporal evolution of variables.

To illustrate the sensibility of the ApprioriAll, Winepi and the Minepi algorithms with the parameters, we defines a set of parameters and apply the algorithms to the sequence ω. The window widths W are set from 2 to 12, and for every window width W, the window movement v is set to $W/3$.

The table 1 provides the number of patterns discovered by each algorithm with the set of parameters.

Table 1: Number of discovered patterns.

W	Number of the discovered patterns		
	Winepi	AprioriAll	Minepi
2	16	16	27
3	28	28	41
4	51	51	57
5	79	79	74
6	133	133	111
7	211	211	145
8	293	293	197
9	282	282	256
10	381	381	329
11	494	494	464
12	825	825	593

These experimentations show the sensibility of the Winepi, AprioriAll and the Minepi algorithms with the parameters: from the first to the end experimentation, the number of patterns increase of more than 5156% for Winepi and AprioriAll, and more than 21961% 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 which find only 3 relations with the example without any parameters.

3 STOCHASTIC APPROACH FRAMEWORK

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 not necessarily a singleton. This theory provides the mathematical foundations of four steps that reverses the usual Data Mining process in order to minimize the size of the set of the discovered patterns.

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 six discrete events: $E = \{e_1 \equiv (x1, Blown)$, $e_2 \equiv (x2, Low)$, $e_3 \equiv (x3, Empty)$, $e_7 \equiv (x7, Off)$, $e_8 \equiv (x8, False)$, $e_9 \equiv (x9, Does\_Not\_Start)\}$. A discrete event class $C^i = \{e_i\}$ is an arbitrary set of discrete event $e_i = (x_i, \delta_i)$. Generally, 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 6 discrete event classes: $Cl = \{C^1 = \{e_1\}$, $C^2 = \{e_2\}$, $C^3 = \{e_3\}$, $C^7 = \{e_7\}$, $C^8 = \{e_8\}$, $C^9 = \{e_9\}\}$.

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. 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 $C^i(k) \equiv (x_i, \delta_i, t_k)$. For example, the illustrative example defines the following sequence: $\Omega = \{(C^2(1), C^3(2), C^3(3), C^8(4), C^9(5), \cdots, C^7(98), C^3(99), C^2(100)\}$. 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 $\omega^i = \{C^i(k)\}$ (Le Goc, 2006):

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

Where each sequence $\omega^i = \{C^i(k)\}$ contains only the observations of the same class C^i. For example, the Ω sequence of the illustrative example is then the superposition of six sequences $\omega^i = \{C^i(k)\}$.

3.1 Step 1: Stochastic Representation

The stochastic representation transforms a set of sequences Ω 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 Ω.

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:

$$\begin{aligned} &\forall i, j, \forall k \in K, \\ &P\left[X(t_k) = q_j | X(t_{k-1}) = q_i\right] = \\ &P\left[C^j(k) \in \Omega | C^i(k-1) \in \Omega\right] \end{aligned}$$

$$\equiv p_{ij} = \frac{N_{ij}}{\sum\limits_{l, l \neq i}^{m} N_{il}}$$

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 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 $D = \{d_{ij}\} = \{(t_{k_j} - t_{k_i})\}$ 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 (Le Goc, 2006) use the properties of the Poisson law to compute the timed constraints: $\tau^-_{ij} = 0, \tau^+_{ij} = \dfrac{1}{\lambda_{i,j}}$ where $\lambda_{i,j}$ is the Poisson rate (i.e. the exponential intensity) of the exponential law that is the average delay $d^{ij}_{moy} = \dfrac{\sum(d_{ij})}{Card(D)}$. The set R of the illustrative example is a set of 26 binary relations : $R = \{R_{i,j}(C^i, C^j, [\tau^-_{i,j}, \tau^+_{i,j}])\}$ where $p_{i,j} = \frac{n_{i,j}}{n_i} > 0$.

3.2 Step 2: Induction of Binary Relations

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 except 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 2). To use this model, the

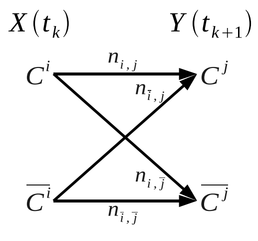

Figure 2: Two abstract binary variables connected by a discrete memoryless channel.

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

This leads to $m \cdot (m-1)$ binary contingency tables of the form of the Table 2.

Table 2: Contingency table for X and Y.

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

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. $P(X(t_k)|Y(t_{k+1}))$. These two matrix allow the definition of the BJ-measure to build a criterion to evaluate the interest of a binary relation $R_{i,j}(C^i, C^j, [\tau_{ij}^-, \tau_{ij}^+])$.

3.2.1 Interestingness of Binary Relations

The idea for defining an efficient interestingness criterion 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 $p(C^j|C^i)$:

$$D(p(Y|X=C^i) \| p(Y)) = p(Y=C^j|X=C^i) \times log_2\left(\frac{p(Y=C^j|X=C^i)}{p(Y=C^j)}\right) + p(Y=C^{\bar{j}}|X=C^i) \times log_2\left(\frac{p(Y=C^{\bar{j}}|X=C^i)}{p(Y=C^{\bar{j}})}\right) \quad (2)$$

In order to remove the symmetry introduced when evaluating the relation $R_{i,j}(C^i, C^j)$ and $R_{i,\bar{j}}(C^i, \overline{C^j})$, we propose to use an oriented Kullback-Leibler distance, called *BJL*.

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 $D(p(Y|X = C^i) \| p(Y))$:*

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

The $BJL(C^i, C^j)$ is the information brought by the occurrences of the class C^i about the occurrences of the class C^j. The Kullback-Leibler distance can be written as the sum of two BJL as follow:

$$D(p(Y|C^i) \| p(Y)) = BJL(C^i, C^j) + BJL(C^i, \overline{C^j}) \quad (3)$$

Contrary to Kullback-Leibler distance, $BJL(C^i, C^j)$ is an asymmetric measure which differently evaluates the binary relations $R_{i,j}(C^i, C^j)$ and $R_{i,\bar{j}}(C^i, \overline{C^j})$. 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 2. *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 $D(p(X|Y = C^j) \| p(X))$:*

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

Both the $BJL(C^i, C^j)$ and $BJW(C^i, C^j)$ measures are combined in a single measure called $BJM(C^i, C^j)$:

Definition 3. *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:

$$\begin{aligned} BJM(C^i,C^j) &= \sqrt{BJL(C^i,C^j)^2 + BJW(C^i,C^j)^2} \\ &- \sqrt{BJL(C^i,\overline{C^j})^2 + BJW(\overline{C^i},C^j)^2} \end{aligned}$$

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 $\theta_{i,j} = \dfrac{n_i}{n_j}$. 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 4.

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

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 26 binary relations of R of the illustrative example are given in table 3. The measure M can finally used as

Table 3: Matrix M.

M	C^1	C^2	C^3	C^7	C^8	C^9
C^1	0.56	0	0	0.8	0	0
C^2	0	0	0	0.64	0	0
C^3	0	0.52	0.49	0	0.54	0
C^7	0	0	0.501	0	0	0.59
C^8	0	0.501	0.51	0	0	0.59
C^9	0	0.51	0.54	0	0.51	0

interestingness criterion for inducing binary relations as follows :

$$M(C^i,C^j) > 0.5 \Rightarrow R_{i,j}(C^i,C^j) \in I \tag{4}$$

For example, the set I of binary relations that can be induced from R of the illustrative example contains 13 binary relations : $I = \{R(C^1,C^1,[\tau^-_{1,1},\tau^+_{1,1}]), R(C^1,C^7,[\tau^-_{1,7},\tau^+_{1,7}]),\cdots\}$.

3.3 Step 3: 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, we can build a set of n-ary relations having some potential to be observed in the initial set Ω of sequences. To this aim, an heuristic $h(m^{i,n})$ can be used to guide 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 $m^{i,n}$:

$$h(m^{i,n}) = card(m^{i,n}) \times BJL(m^{i,n}) \times P(m^{i,n}) \tag{5}$$

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 $P(m^{i,n})$ 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, 2008a). For the illustrative example, let suppose that we are interested by explaining observations of the class C^9 ($C^9 = \{e_9 \equiv (x9, Does\_Not\_Start)\}$). So, the deduction step found three n-ary relations leading to the class C^9(Figure 3).

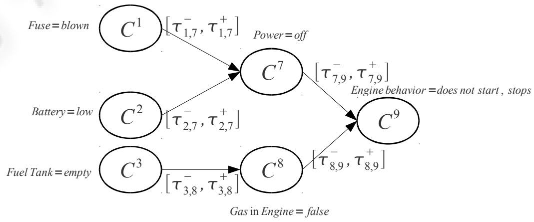

Figure 3: The discovered three n-ary relations.

3.4 Step 4: 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 criteria to build the subset $S \subseteq M$ containing the only paths $m^{k,n}$ being representative according the initial set Ω of sequences. These criteria are a timed version of support and confidence notions:

Definition 5. ***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 6. ***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* $m^{i,n}$*.*

When an n-ary relation $m^{i,n}$ satisfies these criteria, $m^{i,n}$ is called a **signature** (Benayadi and Le Goc, 2008b). For $Ta = 25\%$ and $Tc = 20\%$, all the n-ary relations of the set M of the illustrative example are signatures ($S = M$). These signatures are the only relations (patterns) that are linked with the car system.

4 DISCUSSION

To evaluate the performance of TOM4L process, we will report on the results obtained on the car example (section 2) by TOM4L process and the three popular timed data mining algorithms Winepi(Mannila et al., 1997), AprioriAll (Agrawal and Srikant, 1995) and Minepi (Mannila et al., 1997). It shows that the TOM4L process outperforms Winepi, AprioriAll and Minepi in terms of the number of discovered patterns and theirs accuracy. As we can see from the table 1 and the figure 3, TOM4L process outperforms the three algorithms Winepi, AprioriAll and Minepi in terms of number of the discovered patterns. Furthermore, TOM4L discovers patterns witch are consistent with the structural model of the car system, while most of the patterns discovered by Winepi, AprioriAll and Minepi contradict this structural model.

Also, the three algorithms Winepi, AprioriAll and Minepi require the setting of a set of parameters, so the discovered patterns depend therefore on the values of this parameters (Mannila, 2002). To obtain an interesting patterns, we must found the ideal set of parameters witch need to have some *a priori* knowledge about the car system while this is precisely the global aim of the Data Mining techniques.

Others experiments were made on sequences generated by complex dynamic process as blast furnace process where they show that TOM4L approach converges towards a minimal set of operational relations and outperforms Winepi, AprioriAll and Minepi.

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. Our experiment on a very simple illustrative process, the car system shows that TOM4L process outperforms literature approaches.

REFERENCES

Agrawal, R. and Srikant, R. (1995). Mining sequential patterns. *Proceedings of the 11th International Conference on Data Engineering (ICDE95)*, pages 3–14.

Benayadi, N. and Le Goc, M. (2008a). Discovering temporal knowledge from a crisscross of observations timed. *The proceedings of the 18th European Conference on Artificial Intelligence (ECAI'08). University of Patras, Patras, Greece*.

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

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

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.

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., Toivonen, H., and Verkamo, A. I. (1997). Discovery of frequent episodes in event sequences. *Data Mining and Knowledge Discovery*, 1(3):259–289.

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