Generalized chronicles for temporal sequence classification

Yann Dauxais[1] and Thomas Guyet[2]

[1] KU Leuven, Celestijnenlaan 200a, Leuven, Belgium
yann.dauxais@kuleuven.be
[2] Institut Agro/IRISA UMR6074
thomas.guyet@irisa.fr

Abstract. Discriminant chronicle mining (DCM) [6] tackles temporal sequence classification by combining machine learning and chronicle mining algorithms. A chronicle is a set of events related by temporal boundaries on the delay between event occurrences. Such temporal constraints are poorly expressive and discriminant chronicles may lack of accuracy. This article generalizes discriminant chronicle mining by modeling complex temporal constraints. We present the generalized model and we instantiate different generalized chronicle models. The accuracy of these models are compared with each other on simulated and real datasets.

Keywords: Temporal patterns, discriminant patterns, sequence classification

1 Introduction

Temporal sequences, *i.e.*, sequences of timestamped events, are broadly encountered in various applications. They may represent customer purchases, logs of monitoring systems, or patient care pathways and their analysis is highly valuable to support experts 1) to better understand underlying processes and 2) to decide future actions. Face to the large amount of such data, sequence mining techniques have been proposed to extract interesting behaviors. While most sequence mining approaches are dedicated to the extraction of frequent behaviors, few pattern mining approaches deal with discriminant patterns. Discriminant patterns address the task of sequence classification. In a set of labeled sequences, a discriminant pattern associated to a label L occurs more likely in sequence labeled with L than in the other sequences. Discriminant patterns describe the classes of sequences but they can also be used to predict labels of new sequences.

In this work, we assume that temporal information is an important feature to accurately discriminate behaviors. For instance, knowing the delay between two successive visits on a commercial web site may distinguish loyal customers from the others. The sequence of visited pages may be the same, it is the delay between the visit that witnesses the customer loyalty.

Dauxais et *al.* [6] introduced chronicles to discriminate temporal behaviors in temporal sequences. A chronicle is a set of events linked by temporal relations imposing numerical bounds on delays between events. We showed that the temporal information captured by chronicles improves the accuracy of sequence labeling.

Mining discriminant chronicles is similar to a regular classification problem. It consists in finding suitable boundaries on the temporal delay to accurately discriminate classes. But, chronicles express very simple boundaries, *i.e.*, delays belonging to an interval.

In this article, we extend the expressiveness of the temporal constraints discovered in chronicles to study the discriminatory power of different types of temporal constraints. The main contribution is the proposal of the generalized discriminant chronicles (GDC). GDC is a meta-model that enables to represent different types of patterns, characterized by their modeling of temporal relations between events. Our framework includes a unified GDC mining procedure inspired by the DCM algorithm [6] and a unified decision procedure to label new sequences. The experiments compare the accuracy of four instances of GDC on simulated and real datasets.

2 Related Works

Sequential patterns have been studied since the early stage of the field of pattern mining [18]. Mabroukeh et *al.* [13] review the most efficient sequential pattern approaches. All of them are based on the anti-monotonicity property of the pattern support which states that larger patterns occur fewer times in sequences.

In temporal sequences, events are timestamped and our assumption is that the temporal dimension is a key dimension to accurately characterize interesting behaviors. While sequential patterns capture only information about the order of occurrences of events, temporal patterns capture a more expressive temporal information. Different proposals have been made to enrich sequential patterns with more complex temporal information. Mannila et *al.* proposed episodes [14] as a pattern type which could combine parallel or serial events. Hoeppner et *al.* [11] introduced Allen's temporal logic to specify the temporal relations between interval events. Two events are not necessarily sequentially ordered, they could "overlap" or "be covered". The temporal relations that are discovered are qualitative. In temporally annotated sequences (TAS) [10] the successive events are constrained by numerical duration extracted by combining a density clustering technique. The chronicle model [5] is at a crossroad between episode and TAS. It is a partial temporal order applied on pattern events constrained by numerical temporal intervals. This pattern model is more general than sequential patterns, TAS and episodes.

Finally, quantitative episodes [15] are tree-based patterns that are graphically similar to chronicles but formally more similar to sets of TAS. Indeed, a quantitative episode represents a set of TAS that are all specifying the same

sequential pattern. This set of TAS is represented by a tree rooted on the first event of the sequential pattern for which each path leading to a leaf is a TAS.

Sequence classification has been addressed with statistical approaches such as HMM but also with machine-learning approaches such as recurrent neural networks (LSTM) [12].

Bringmann et *al.* [3] reviewed "pattern-based classification" that combines pattern mining algorithms and machine learning algorithms to classify structured data, such as sequences. This problem is quite similar to the subgroup discovery task [2]. The main difference between both approaches is that subgroup discovery is meant in a descriptive way whereas pattern-based classification is meant in a predictive way.

The main steps of pattern-based classification are the following (1) a pattern mining step building a vector representation of sequences based on the presence or absence of some extracted patterns; and (2) a machine learning algorithm building a classifier based on the vector representations of labeled sequences. The use of a final classifier makes the results difficult to interpret. For this reason, we focus our interest on the extraction of discriminant patterns, *i.e.*, patterns that can be interpreted by their own as a discriminant behavior.

The proposed approaches are based on interestingness measures different from frequency and capturing the differences between occurrences with subsets of sequences. The most-used measures are growth rate [7] and disproportionality [1]. The BIDE-D algorithm [9] extracts discriminant sequential patterns instead of frequent ones. This technique allows to use a smaller pattern set than the frequent one with a similar accuracy. Recently, the DCM algorithm [6] extended the discriminant sequential pattern mining with chronicles. But temporal constraints of chronicles (*i.e.*, inter-event duration, so-called time gap, within an interval) is maybe too simple to capture complex temporal relationships, and mining patterns with more complex temporal constraints may improve classification accuracy.

3 Discriminant Chronicle Mining

Let $\mathbb{E}$ be a set of event types totally ordered by $\leq_{\mathbb{E}}$. An *event* is a pair (e, t) such that $e \in \mathbb{E}$ and $t \in \mathbb{R}$. A *sequence* is a tuple $\langle SID, \langle (e_1, t_1), (e_2, t_2), \ldots, (e_n, t_n)\rangle, L\rangle$ where SID is the sequence index, $\langle (e_1, t_1), (e_2, t_2), \ldots, (e_n, t_n)\rangle$ a finite sequence of events and $L \in \mathbb{L}$ where $\mathbb{L}$ is a label set. Sequence events are ordered by timestamps and by labels if equality.

Table 1 represents a set of six sequences containing five event types (A, B, C, D and E) and labeled with two different labels $\mathbb{L} = \{+, -\}$. In such case $\leq_{\mathbb{E}}$ is the lexicographic order.

A *chronicle* is a couple $(\mathcal{E}, \mathcal{T})$ such that: $\mathcal{E} = \{\!\{e_1 \ldots e_n\}\!\}$, $e_i \in \mathbb{E}$ and $e_i \leq_{\mathbb{E}} e_j$ for all $1 \leq i < j \leq n$. $\mathcal{E}$ is a *multiset*, *i.e.* $\mathcal{E}$ can contain several occurrences of a same event type. $\mathcal{T}$ is a set of *temporal constraints*, *i.e.* expressions of the form $(e_i, i)[t^-, t^+](e_j, j)$ such that $i, j \in [n]$, $i < j$ and $t^-, t^+ \in \mathbb{R} \cup \{-\infty, +\infty\}\}$. A

SID	Sequence	Label
$\boldsymbol{s_1}$	$(\mathsf{A},1)$, $(\mathsf{B},3)$, $(\mathsf{A},4)$, $(\mathsf{C},5)$, $(\mathsf{C},6)$, $(\mathsf{D},7)$	$+$
$\boldsymbol{s_2}$	$(\mathsf{B},2)$, $(\mathsf{D},4)$, $(\mathsf{A},5)$, $(\mathsf{C},7)$	$+$
$\boldsymbol{s_3}$	$(\mathsf{A},1)$, $(\mathsf{B},4)$, $(\mathsf{C},5)$, $(\mathsf{B},6)$, $(\mathsf{C},8)$,$(\mathsf{D},9)$	$+$
$\boldsymbol{s_4}$	$(\mathsf{B},4)$, $(\mathsf{A},6)$, $(\mathsf{E},8)$, $(\mathsf{C},9)$	$-$
$\boldsymbol{s_5}$	$(\mathsf{B},1)$, $(\mathsf{A},3)$, $(\mathsf{C},4)$	$-$
$\boldsymbol{s_6}$	$(\mathsf{C},4)$, $(\mathsf{B},5)$, $(\mathsf{A},6)$, $(\mathsf{C},7)$, $(\mathsf{D},10)$	$-$

Table 1. Sequences labeled with two classes $\{+,-\}$.

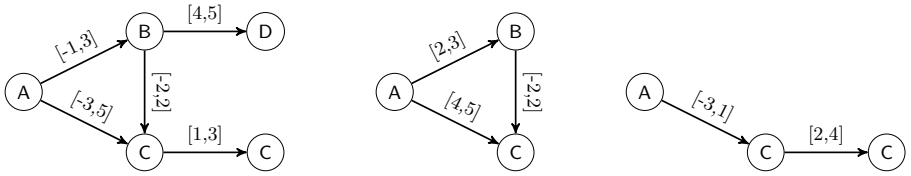

Fig. 1. Examples of three chronicles occurring in Table 1 (detailed in the text). From left to right, the chronicles $\mathcal{C}$, $\mathcal{C}_1$ and $\mathcal{C}_2$.

temporal constraint specifies acceptable delays between the occurrences of the multiset events.

A chronicle $\mathcal{C} = (\mathcal{E} = \{\!\{e'_1, \ldots, e'_m\}\!\}, \mathcal{T})$ *occurs* in a sequence $\boldsymbol{s} = \langle (e_1, t_1), \ldots, (e_n, t_n)\rangle$, denoted $\mathcal{C} \in \boldsymbol{s}$, iff there exists an injective function $f : [m] \mapsto [n]$ such that 1) $\tilde{\boldsymbol{s}} = \langle (e_{f(1)}, t_{f(1)}), \ldots, (e_{f(m)}, t_{f(m)})\rangle$ is a subsequence of $\boldsymbol{s}$, 2) $\forall i,\ e'_i = e_{f(i)}$ and 3) $\forall i, j,\ t_{f(j)} - t_{f(i)} \in [t^-, t^+]$ where $e_{f(i)}[t^-, t^+]e_{f(j)} \in \mathcal{T}$. An *occurrence* of $\mathcal{C}$ in $\boldsymbol{s}$ is a list of timestamps $\mathcal{O} = \langle o_1, \ldots, o_m\rangle$ where $\forall i \in [m]$, $o_i = t_{f(i)} \in \mathbb{R}$.

The *support* of a chronicle $\mathcal{C}$ in a set of sequences $\mathcal{S}$ is the number of sequences in which $\mathcal{C}$ occurs:

$$supp(\mathcal{C}, \mathcal{S}) = |\{\boldsymbol{s} \in \mathcal{S} \mid \mathcal{C} \in \boldsymbol{s}\}|.$$

Fig. 1 illustrates three chronicles represented by graphs. Chronicle $\mathcal{C} = (\mathcal{E}, \mathcal{T})$ where $\mathcal{E} = \{\!\{\mathsf{A}, \mathsf{B}, \mathsf{C}, \mathsf{C}, \mathsf{D}\}\!\}$ and $\mathcal{T} = \{(\mathsf{A},1)[-1,3](\mathsf{B},2), (\mathsf{A},1)[-3,5](\mathsf{C},3), (\mathsf{B},2)[-2,2](\mathsf{C},3), (\mathsf{B},2)[4,5](\mathsf{D},5), (\mathsf{C},3)[1,3](\mathsf{C},4)\}$ is illustrated at the top left. Chronicle $\mathcal{C}$ (see Fig. 1 on the left), occurs in sequences $\boldsymbol{s_1}$, $\boldsymbol{s_3}$ and $\boldsymbol{s_6}$ of Table 1. We notice there are two occurrences of $\mathcal{C}$ in sequence $\boldsymbol{s_1}$. Nonetheless, its support is $supp(\mathcal{C}, \mathcal{S}) = 3$. The two other chronicles, denoted $\mathcal{C}_1$ and $\mathcal{C}_2$, occur respectively in sequences $\boldsymbol{s_1}$ and $\boldsymbol{s_3}$; and in sequence $\boldsymbol{s_6}$. Their supports are $supp(\mathcal{C}_1, \mathcal{S}) = 2$ and $supp(\mathcal{C}_2, \mathcal{S}) = 1$.

Frequent chronicle mining consists in extracting all chronicles $\mathcal{C}$ in a dataset $\mathcal{S}$ such that $supp(\mathcal{C}, \mathcal{S}) \geq \sigma_{min}$ [5]. The *DCM* algorithm extracts *discriminant chronicle* [6]. A discriminant chronicle occurs at least g_{min} times more in the set of positive sequences, *i.e.* sequences labeled with $+$, than in the set of negative sequences (labeled with $-$). Then, it can be represented as a classification rule $\mathcal{C} \Rightarrow +$ specifying that sequences in which $\mathcal{C}$ occurs more likely belongs to class

+. In this mining task, the user has to specify two thresholds: the minimum frequency threshold σ_{min} and the minimal growth rate $g_{min} \geq 1$.

4 Generalized Discriminant Chronicles (GDC)

We now introduce the generalized discriminant chronicle (GDC) meta-model. The GDC meta-model defines an abstract pattern model of temporal behaviors. Next section instantiates different concrete approaches within a unified framework of generalized discriminant chronicle, *i.e.* a GDC model and a mining algorithm (see Sect. 4.2).

Let $\mathbb{L}$ be a set of labels and $\mathbb{E}$ be a set of event types, a *generalized discriminant chronicle* (GDC) is a couple $(\mathcal{E}, \mu)$, where $\mathcal{E}$ is a multiset of event types and $\mu : \mathbb{R}^{|\mathcal{E}|} \mapsto [0,1]^{|\mathbb{L}|}$ is an *occurrence assessment function*.

The *occurrence assessment function* intuitively gives the confidence measure that a multiset witnesses each label. For some occurrence $\mathcal{O} \in \mathbb{R}^{|\mathcal{E}|}$ of multiset $\mathcal{E}$ in a sequence, $\mu(\mathcal{O}) = [p_1, p_2, \ldots, p_{|\mathbb{L}|}]$ where $\forall i,\ \mu^i(\mathcal{O}) = p_i \in [0,1]$ gives the confidence measure that $\mathcal{O}$ belongs to the i-th class. In case it sum to 1, this vector can be interpreted as a probability distribution.

GDC generalizes the previous definition of chronicles in two manners: 1) the *occurrence assessment function* is a generalization of the temporal constraints, 2) the *weighted vector of decisions* $[0,1]^{|\mathbb{L}|}$ is the generalization of the association of a chronicle to a label ($\mathcal{C} \Rightarrow L$, $L \in \mathbb{L}$).

In particular, it is possible to encode the discriminant chronicle $(\mathcal{E}, \mathcal{T}) \Rightarrow L_l$, where $L_l \in \mathbb{L}$ is the l-th sequence class, as a GDC using $\mu_{\mathcal{T}}$ defined such that for some occurrence $\mathcal{O} = \{o_i\}_{i \in [|\mathcal{E}|]}$ of $\mathcal{E}$:

$$\mu_{\mathcal{T}}(\mathcal{O}) = \begin{cases} \mathbb{1}_l \text{ if } \forall e_i[a,b]e_j \in \mathcal{T}, a \leq o_j - o_i \leq b \\ \mathbf{0} \text{ otherwise} \end{cases}$$

where $\mathbb{1}_l$ is a vector of zeros except at position l (value 1), and $\mathbf{0}$ is a vector of zeros. The size of these two vectors is $|\mathbb{L}|$.

4.1 Taking Decisions with Generalized Discriminant Chronicles

This section describes how generalized discriminant chronicles are used to automatically classify new sequences. Let $C = (\mathcal{E}, \mu)$ be a GDC and $\boldsymbol{s}$ be a sequence to classify such that there exists at least one occurrence $\mathcal{O} \in \mathbb{R}^{|\mathcal{E}|}$ of multiset $\mathcal{E}$. Then, decision vector is given by $\mu(\mathcal{O})$. But the multiset $\mathcal{E}$ may occur several times in $\boldsymbol{s}$. All decisions have to be combined and the final classification decision for sequence $\boldsymbol{s}$, denoted $d_C(\boldsymbol{s}) \in \mathbb{L}$, is the class label with the largest confidence value:

$$d_C(\boldsymbol{s}) = \underset{l \in \mathbb{L}}{\operatorname{argmax}} \left(\max_{\mathcal{O}} \mu^l(\mathcal{O}) \right) \tag{1}$$

where $\max_{\mathcal{O}} \mu^l(\mathcal{O})$ denotes the maximum confidence value obtained for label $l \in \mathbb{L}$ for all occurrences $\mathcal{O}$ of the multiset. The function $d_C(\boldsymbol{s})$ enables to use a GDC as a decision rule. It decides which class a sequence belongs to.

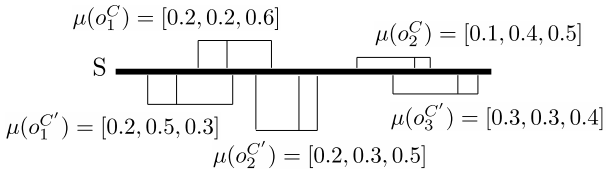

Fig. 2. Examples of multiple occurrences of two chronicles C and C' to be combined to make the final decision. A "rake" illustrates item positions in the sequence of one occurrence of the multiset.

The class label can also be decided from a set of chronicles $\mathcal{C} = \{C_i\}_{1 \leq i \leq n}$. In this case, each chronicle yields its own decision, and they are merged into a final decision. The decision procedure we propose, denoted $d_{\mathcal{C}}(\boldsymbol{s})$ – with a collection of chronicles as subscript, is a linear combination of the number of occurrences of a chronicle C_i in $\boldsymbol{s}$ labeled with $l_j \in \mathbb{L}$, more formally:

$$d_{\mathcal{C}}(\boldsymbol{s}) = \underset{j \in \mathbb{L}}{\operatorname{argmax}} \left(\sum_{i=1}^{n} \alpha_{i,j} \nu_{C_i}^j(\boldsymbol{s}) + \beta_j \right) \tag{2}$$

where $\alpha_{i,j} \in \mathbb{R}$ and $\beta_{i,j} \in \mathbb{R}$ are parameters, and $\nu_{C_i}^j(\boldsymbol{s})$ is the number of occurrences of chronicle C_i in sequence $\boldsymbol{s}$ that suggests classifying the sequence in class j (*i.e.* $d_{C_i}(\boldsymbol{s}) = j$):

$$\nu_{C_i}^j(\boldsymbol{s}) = \left| \left\{ \mathcal{O} \in \mathbb{R}^{|\mathcal{E}|} \,\middle|\, \underset{l \in \mathbb{L}}{\operatorname{argmax}}(\mu^l(\mathcal{O})) = l_j \right\} \right| \tag{3}$$

Fig. 2 illustrates a sequence $\boldsymbol{s}$ classified with a set of two chronicles C and C', and $|\mathbb{L}| = 3$. Chronicle C occurs twice in $\boldsymbol{s}$ and C' occurs thrice, $\mathcal{O} = \{o_1^C, o_2^C, o_1^{C'}, o_2^{C'}, o_3^{C'}\}$. The figure illustrates respective decision vectors.

In this case, $\nu_C = [0, 0, 2]$ because the majority class in the two occurrences of chronicle C is the third one, and $\nu_{C'} = [0, 1, 2]$. Assuming $\beta_j = 0$ and $\alpha_{i,j} = 1$, $\forall i, j$, then the predicted class is $d_{\mathcal{C}}(\boldsymbol{s}) = 3$ because $\sum_{i=1}^{n} \alpha_{i,3} \nu_{C_i}^3(\boldsymbol{s}) + \beta_3 = 4$ is the largest predicted value among possible classes.

The intuition is that the contribution to the final decision of chronicle C_i is more important if this chronicle appears several times in the sequence. Combining the numbers of occurrences is preferred to the combination of confidence measures to prevent from bias due to chronicles with low recall (*i.e.* poorly informative) but with possible high confidence.

In practice, the $\alpha_{i,j}$ and β_j parameters are not set up manually but learned from data as explain in next section.

4.2 Learning Generalized Discriminant Chronicles Classifiers

The overall procedure dedicated to learn the sequence classifier is given in Fig. 3. This procedure extracts both a set of γ discriminant chronicles, denoted $\overline{\mathbb{C}}$, and

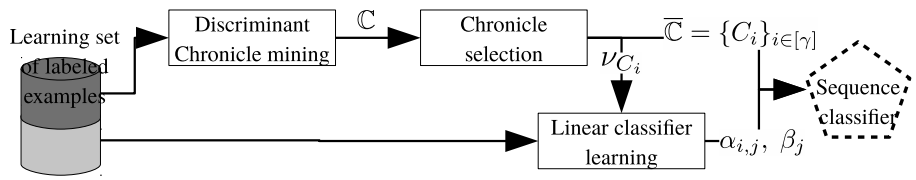

Fig. 3. Overall procedure of sequence classifier learning.

parameters values of decision function (see Equation 2). First of all, the learning dataset is split into two separated bunches of sequences.

One dataset is used to extract a set of discriminant chronicles $\mathbb{C} = \{C_i\}$. A subset of the γ most discriminant chronicles, denoted $\overline{\mathbb{C}}$, is selected from $\mathbb{C}$. According to BIDE-D [9], reducing the set of chronicles prevents from overfitting. The second dataset is used to learn the decision procedure. In case of a dataset with two classes ($\mathbb{L} = \{+, -\}$), equation 2 can be seen as a linear classification problem. Then, a linear-SVM classifier learns the $\alpha_{i,j}$ and β_j parameters. In practice, a linear-SVM classifier is also used for a multi-class setting parameters and its model serves as decision function that takes the final classification decision.

We now come back to the mining of generalized discriminant chronicles. This algorithm is based on the original DCM algorithm [6]. Algorithm 1 gives the general principle of GDC mining from a dataset of labeled sequences $\mathcal{S}$. The two main parameters are σ_{min}, a minimal support threshold used to prevent from generating too much poorly-representative chronicles and g_{min}, a minimal growth rate threshold. The overall principle from learning multiple-class chronicles is the one class against all. For some class L, the minimal growth rate g_{min} indicates that a GDC occurs at least g_{min} times more in sequences of class L than in all other sequences.

Algorithm 1 extracts GDC for each class L in two main steps. It firstly extracts $\mathbb{M}$, the set of frequent multisets in the sequences of class L. Then, ExtractDTC learns a μ function from the list of occurrences of a frequent multiset. There is a unique μ per multiset. Its principle is first to build a time-gap table [19] from all occurrences of a multiset and, second, to learn a temporal model from the time-gap table. Each time-gap occurrence is labeled by the label of its sequence and any standard machine learning algorithm can learn the μ function.

The DCM algorithm [6] is based on rule induction (*e.g. Ripper* [4]) to learn temporal constraints of a chronicle ($\mathcal{T}$). It is a specific case of a $\mu_{\mathcal{T}}$ function that fits the requirements of the original model of chronicle. Next section introduces alternative classes of occurrence assessment functions.

Algorithm 1 Generalized discriminant chronicle mining

Require: $\mathcal{S}$: labeled sequence sets, $\mathbb{L}$: set of labels, σ_{min}: minimal support threshold, g_{min}: minimal growth threshold

$\mathbb{C} \leftarrow \emptyset$ ▷ $\mathbb{C}$ is the discriminant chronicle set
for all $L \in \mathbb{L}$ **do**
  $\mathbb{M} \leftarrow$ ExtractMultiSet($\mathcal{S}^L$, σ_{min})
  for all $ms \in \mathbb{M}$ **do**
    for all $\mu \in$ ExtractDTC($\mathcal{S}$, L, ms, g_{min}, σ_{min}) **do**
      $\mathbb{C} \leftarrow \mathbb{C} \cup \{(ms, \mu)\}$ ▷ Add a new GDC
return $\mathbb{C}$

5 Examples of GDC Instances

This section illustrates several types of patterns that can be represented by GDC: discriminant sequential patterns, discriminant episodes, SVM-DC and DT-DC. The first two types of patterns illustrate the ability of GDC to model existing patterns (less expressive than the original discriminant chronicles) and the last two models illustrate meaningful generalizations of temporal constraints. In the remaining of this section, we briefly present each of these models as instances of the GDC.

Discriminant episodes and sequences An episode is a set of events ordered temporally by a partial order $\leq_{\mathcal{E}}$. If $\leq_{\mathcal{E}}$ is a total order, the episode is a sequential pattern. Such classical temporal patterns have been used for mining discriminant behaviors respectively by Fabrègue et *al.* [8] and by Fradkin et *al.* [9]. A discriminant episode is an episode associated to a label $L \in \mathbb{L}$. Such discriminant patterns could be represented by a GDC model instance by the following occurrence assessment function:

$$\mu_{\mathcal{T}}(o) = \begin{cases} \mathbb{1}_L & \text{if } \forall (i,j),\ i \leq_{\mathcal{E}} j \Rightarrow o_i \leq o_j \\ \mathbf{0} & \text{otherwise} \end{cases} \tag{4}$$

For example, a multiset $\mathcal{E} = \{\!\{\mathsf{A}, \mathsf{B}, \mathsf{C}\}\!\}$ ordered by $\leq_{\mathcal{E}}$ such that $\mathsf{B} \leq_{\mathcal{E}} \mathsf{A}$ and $\mathsf{B} \leq_{\mathcal{E}} \mathsf{C}$ specifies an episode representing sequences where B occurs before events A and C, no matter the order between A and C. While associated to a label, it becomes a discriminant episode. Expressed with chronicle temporal constraints, we have $\mathcal{T} = \{(\mathsf{A},1)[-\infty,0](\mathsf{B},2), (\mathsf{A},1)[-\infty,\infty](\mathsf{C},3), (\mathsf{B},2)[0,\infty](\mathsf{C},3)\}$.

Decision Tree Discriminant chronicles (DT-DC) A discriminant chronicle is characterized by temporal constraints on the time gaps ($\mathcal{T}$). A constraint (e,i) $[t^-,t^+](e',j)$ enforces the time gap δ between some occurrences of events e and e' to belong to the interval $[t^-,t^+]$. But chronicle does not allow disjunctions of constraints. For instance, it is not possible to specify that δ may belong to $[t^-,t^+] \cup [t'^-,t'^+]$.

The DT-DC model replaces the conjunctive rule learning algorithm by a decision tree, such as C4.5 [17]. For example, let's consider a dataset of positive sequences matching temporal constraints $(\mathsf{A},1)[2,3](\mathsf{B},2)$ and $(\mathsf{A},1)[7,9](\mathsf{B},2)$ and

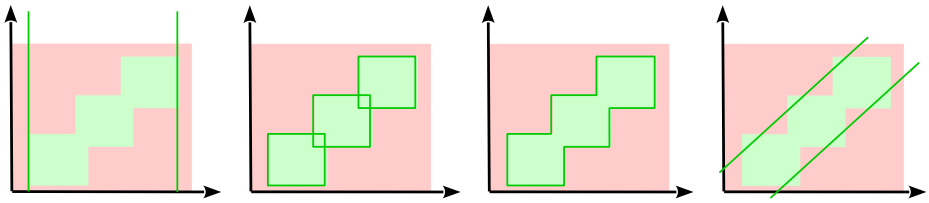

Fig. 4. Illustration of temporal discrimination power of the different instances of GDC. Planes (x, y) represent a pair of temporal constraints for some sequences $(\mathsf{A}, t_0)(\mathsf{B}, t_0 + x)(\mathsf{C}, t_0 + x + y)$. Positive sequences are those with (x, y) values in the green region, negative sequences have (x, y) values in the red region. The bold-green lines represent the separation boundaries learned by a GDC depending on the type of occurrence assessment function, μ. From left to right: discriminant episodes (temporal constraints with shape $[0, +\infty]$), chronicles (three chronicles with temporal constraints represented by rectangles), DT-DC (a single shape combining several rectangles), linear-SVM (a single chronicle, with generalized linear boundaries).

a dataset of negative sequences matching the temporal constraint $(\mathsf{A}, 1)[2, 9](\mathsf{B}, 2)$. In this case, two chronicles would be discriminant (one per interval, $[2, 3]$ and $[7, 9]$). On the opposite, a single DT-DC will capture the disjunction of intervals in the same model. The expected benefit of this model is a better generalization power.

SVM Discriminant chronicles (SVM-DC) SVM Discriminant chronicles illustrate the case of a complex learnable occurrence assessment function μ, *i.e.* a μ modeled by a multi-class SVM classifier. Compared to the previous types of patterns, SVM-DC is not limited to linear boundaries to separate examples (time gaps of multiset occurrences) and is a good candidate for yielding accurate patterns.

It is worth noticing that any machine learning model yields a new type of discriminant temporal patterns based on the GDC. The above GDC instances show the potential variety of temporal constraints that GDC can model. Fig. 4 illustrates the shape of boundaries defined by occurrence assessment functions of a chronicle learned from a synthetic dataset.

6 Experiments

In this part, we compare different results in pattern-based classification using discriminant episodes, discriminant chronicles, DT-DC and SVM-DC. The goal of these experiments is to highlight the impact of the GDC model choice on the accuracy of decision functions presented in Sect. 4.1: $d_C(\boldsymbol{s})$ and $d_{\mathcal{C}}(\boldsymbol{s})$. In the experiments we analyze the classification power of individual GDC (*i.e.* $d_C(\boldsymbol{s})$) and of a set of GDC (*i.e.* $d_{\mathcal{C}}(\boldsymbol{s})$).

The DT-DC and SVM-DC mining algorithms are implemented in Python using scikit-learn library [16]. The algorithm dedicated to discriminant chronicle mining is implemented in C++.[3]

6.1 Experimental Setup

The different experiments compare mean accuracy of different GDC models obtains by cross-validation on synthetic and real datasets.

A 5-cross-validation is performed on each dataset for the parameters σ_{min} and g_{min} of the mining step described in Sect. 3 and the parameter γ described in Sect. 4.2. The domains used for σ_{min}, g_{min} and γ are respectively $\{0.2, 0.3, 0.4, 0.5, 0.6\}$, $\{1.4, 1.6, 1.8, 2, 3\}$ and $\{90, +\infty\}$. $\gamma = +\infty$ means that all discriminant chronicles are kept. To improve the computation time, a fourth parameter is introduced for the mining step: the maximal size of extracted chronicles max_size. This parameter constrains the maximal number of events that a GDC, *i.e.* its multiset, can contain. The domain of this parameter is $\{3, 4, 5, 6\}$.

The real datasets are the UCI datasets presented in the BIDE-D experiments [9]: *asl-bu*, *asl-gt* and *blocks*. These datasets are part of the standard benchmark for pattern-based classification approaches.

We generated two collections of synthetic datasets:

- A first collection of datasets is based on the principle illustrated by Fig. 4. Random sequences $\langle(\mathsf{A}, 0)(\mathsf{B}, x)(\mathsf{C}, x + y)\rangle$ have been generated: the event A occurs at time 0 in each sequence and the time gaps between A and B and between B and C are randomly generated in the interval $[0, 15]$. The label of the sequence is generated depending on the temporal constraints. According to Fig. 4, positive examples having time gaps located in one of the three green squares. Coordinates of the square corners are $(1, 1), (6, 6)$; $(5, 5), (10, 10)$ and $(9, 9), (14, 14)$. Each dataset contains 150 positive and 150 negative sequences.
- A second collection of datasets is based on random sequences with shape $\langle(\mathsf{A}, t_\mathsf{A})\ (\mathsf{B}, t_\mathsf{A} + x)(\mathsf{C}, t_\mathsf{C})(\mathsf{D}, t_\mathsf{C} + k \times x)\rangle$ where $x \in [1, 9]$, $t_\mathsf{A} = 15$ and $t_\mathsf{C} \in [1, 29]$. The two sequence classes are distinguished by the k factor: $k = 2$ for positive sequences while $k = 1$ for negative ones. Each dataset contains 100 positive and 100 negative sequences.

To ease the comparison between DT-DC and discriminant chronicles as individual patterns, we choose to use each node of the extracted trees as discriminant temporal constraint. This prevents from comparing the classification power of the decision-tree algorithm and the rule learning algorithm (*Ripper*). Furthermore, decision trees produce more discriminant chronicles than *Ripper* because tree nodes are more redundant.

[3] All software sources and datasets are available at https://gitlab.inria.fr/ydauxais/GDC-PBC.

	σ_{min}	g_{min}	accuracy	support	number
discriminant episodes	0.3	2.0	**0.84**(±**0.17**)	5.67(±2.50)	12
	0.2	2.0	0.76(±0.16)	6.78(±3.12)	27
	0.2	1.6	0.76(±0.13)	6.74(±3.22)	23
discriminant chronicles	0.2	3.0	**0.89**(±**0.15**)	9.13(±3.02)	69
	0.3	1.8	0.78(±0.17)	13.2(±7.32)	56
	0.5	2.0	0.63(±0.08)	20.7(±7.13)	10

Table 2. Three most accurate parameter sets for discriminant episodes and discriminant chronicles on the first synthetic dataset. The *number* attribute is the total number of chronicles extracted in the 5 runs.

6.2 Results

Let us first present results obtained by the GDC instances on synthetic datasets. For this experiment, we only consider the extracted chronicles with multiset $\{\!\{\mathsf{A}, \mathsf{B}, \mathsf{C}\}\!\}$. Thus, only one DT-DC and one SVM-DC is extracted for each run, but the number of discriminant chronicles or episodes depends on the setting (see Table 2). The unique DT-DC represents almost perfectly the discriminant behavior used to generate the dataset with a mean accuracy of 0.99(±0.02). This result was expected because of the dataset structure (squares with boundaries orthogonal to the axis) fits the discrimination capabilities of decision trees.

No SVM-DC are extracted for the default parameters of g_{min}. Our explanation is that concavities in the shape containing positive occurrences disadvantage linear SVM. Relaxing the constraint of g_{min}, SVM-DC reaches a mean accuracy of 0.48(±0.05). This shows experimentally that DT-DC can be more accurate than SVM-DC for some datasets.

An overview of the results for the discriminant episodes and the discriminant chronicles on the latter dataset is given in Table 2. The best mean accuracy presented in this table is 0.89 for discriminant chronicles and 0.84 for discriminant episodes. Even if these accuracy results are very good, the standard deviations of 0.15 and 0.17 illustrate that some extracted patterns are much less accurate. The three squares defining the positive occurrences may be represented by three chronicles but it is more difficult to represent the negative occurrences because some of them are not included in a frequent rectangle containing only negative occurrences. It is not possible to represent the whole dataset with discriminant episodes and it can be seen in Table 2. Fewer episodes are extracted than chronicles and their mean support is lower. Furthermore, only a small part of the dataset can be represented by episodes. On the other hand, the extraction parameters influence less their accuracy and support. Thus, discriminant episodes overfit less the data.

We compared these results with the discriminant chronicles obtained using DCM. Among the discriminant chronicles extracted using DCM, discriminating positive occurrences from negative ones generates three perfectly discriminant chronicles representing the three squares used to generate the data with parameters $\sigma_{min} = 0.2$, $g_{min} = 3$ and considering only the multiset $\{\!\{\mathsf{A}, \mathsf{B}, \mathsf{C}\}\!\}$.

σ_{min}	g_{min}	accuracy	support	number
0.3	3.0	1	10.5(±3.46)	16
0.2	1.8	0.95(±0.13)	8.19(±3.64)	37
0.2	1.6	0.90(±0.17)	12.9(±8.86)	31
0.6	1.6	0.79(±0.17)	18.8(±10.79)	11
0.6	1.4	0.72(±0.17)	20.4(±10.48)	10

Table 3. Five most accurate parameter sets for regular discriminant chronicles on the second synthetic dataset. The attribute *number* is the total number of chronicles extracted in the 5 runs.

Discriminating the negative occurrences from the positive ones with DCM generates two perfectly discriminant chronicles representing the largest rectangles of negative occurrences on the top left and on the bottom right of the Fig. 4. Then, the mean accuracy of discriminant chronicles is 1 and, contrary to DT-DC, some negative occurrences are not covered by these chronicles. This perfect accuracy is correlated to the strategy of *Ripper* that does not reuse covered occurrences to build a new temporal constraint. The remaining occurrences are so considered too few to be used for building a new constraint. The accuracy is better due to the partial coverage of the dataset made by discriminant chronicles.

We present the same experiment on the second collection of datasets. These datasets are generated to favor the SVM-DC model with boundaries that correlates linearly the time gaps. Again, we only considered the extracted chronicles with multiset $\{\!\{\mathsf{A}, \mathsf{B}, \mathsf{C}, \mathsf{D}\}\!\}$. For the simplest dataset, the single extracted SVM-DC obtained the accuracy of 1 for the 5 runs. The extracted DT-DC obtained an mean accuracy of 0.99(±0.02). Thus, SVM-DC accuracy is not better than DT-DC, but, DT-DC builds a very large decision tree that overfits the boundaries, which is not suitable in real applications.

Table 3 shows the results of regular discriminant chronicles. We observe that the most accurate parameter sets extract chronicles with a small support and the least accurate parameter sets extract chronicles with a bigger support. We do not present results for discriminant episodes because not any discriminant episodes are extracted. This shows the limit of a too simple model.

Let us now present the results obtained by the three GDC models as individual patterns and as pattern sets on real datasets. An overview of the classification power of the individual patterns of DT-DC, discriminant chronicles and discriminant episodes is given by Table 4. It shows that DT-DC patterns are individually less accurate than discriminant chronicles obtained from the same decision trees. Discriminant episodes are also individually more discriminant than discriminant chronicles. The intuition behind these results is that decision trees overfit more the datasets than temporal constraints or sequential orders. Temporal constraints and sequential orders gather only dense sets of occurrences, represented as squares on Fig. 4, but decision trees generalize examples and gather too dissimilar occurrences.

	σ_{min}	g_{min}	max_size	accuracy	support
discriminant episodes	0.3	2.0	3	0.92(±0.14)	5.62(±8.60)
	0.3	1.8	4	0.86(±0.19)	5.24(±7.65)
	0.6	3.0	3	0.86(±0.20)	2.96(±5.54)
	0.6	1.8	4	0.83(±0.18)	11.41(±9.71)
	0.3	1.6	3	0.83(±0.22)	4.41(±5.94)
discriminant chronicles	0.3	2.0	3	0.64(±0.37)	3.16(±2.91)
	0.3	1.8	4	0.60(±0.40)	2.69(±2.68)
	0.6	3.0	3	0.59(±0.33)	5.74(±7.01)
	0.3	1.6	3	0.58(±0.38)	3.67(±3.28)
	0.6	1.8	4	0.57(±0.34)	3.83(±3.37)
DT-DC	0.5	2.0	5	0.38(±0.28)	8.12(±6.57)
	0.6	3.0	5	0.37(±0.25)	8.61(±6.34)
	0.2	1.6	3	0.36(±0.34)	5.99(±5.68)
	0.5	1.8	3	0.34(±0.21)	13.2(±8.19)
	0.5	1.8	4	0.33(±0.23)	10.0(±7.40)

Table 4. Comparison table of DT-DC, discriminant chronicles and discriminant episodes for the 5 best parameter sets in terms of mean accuracy on *asl-bu*.

Conversely to the accuracy, the mean support is higher for DT-DC than for discriminant chronicles and discriminant episodes. Each DT-DC covers more examples than the two other types of patterns. Furthermore, the coverage of discriminant episode is worse than DT-DC due to their poor expressiveness of temporal behaviors.

These accuracy results are extreme because we did not use the decision tree parameter constraining a leaf to have a minimal support. For example, if this parameter is set to f_{min}, a decision tree can be seen as a set of discriminant chronicles for a unique multiset.

To illustrate the importance of this parameter, we can compare the two accuracy distributions. Fig. 5 at top left shows the accuracy distribution of DT-DC for the most accurate parameter set. The distribution at bottom right is the accuracy distribution of discriminant chronicles for the most accurate parameter set. The distribution at bottom left is the accuracy distribution of DT-DC for the best discriminant chronicle parameter set with a mean accuracy of 0.23(±0.25). The distribution at top right is the accuracy distribution of discriminant chronicles for the best DT-DC parameter set with a mean accuracy of 0.32(±0.39).

We first notice that the accuracy distributions of discriminant chronicles and DT-DC are almost similar. These histograms show three main peaks: patterns that obtained the accuracy of 0, 0.5 and 1. It makes sense considering that both types of patterns were extracted by the same algorithm. The differences between the mean accuracy are mainly in the proportion of patterns with accuracy equals to 0 and equals to 1. Proportionally to the number of 1-accuracy patterns, the peak of 0-accuracy is higher for DT-DC than for discriminant chronicles. This means that the proportion of patterns that always make wrong decisions is higher

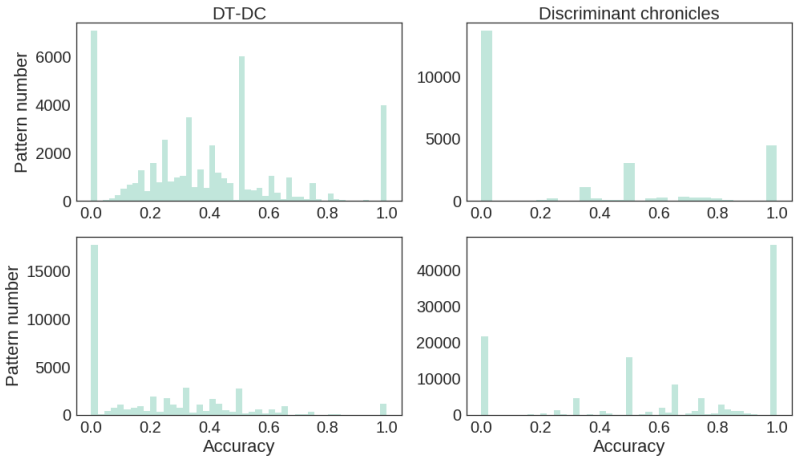

Fig. 5. Accuracy distribution for DT-DC and discriminant chronicles with parameters $\sigma_{min} = 0.5$, $g_{min} = 2$ and $max\_size = 5$ for the first row and $\sigma_{min} = 0.3$, $g_{min} = 2$ and $max\_size = 3$ for the second one.

	SVM-DC		**discriminant chronicles**	
dataset	accuracy	CPU time (s)	accuracy	CPU time (s)
asl-bu	**0.73**(±**0.05**)	**15.2**(±**0.34**)	0.68(±0.06)	16.4(±0.37)
asl-gt	**0.42**(±**0.02**)	386(±8.37)	0.32(±0.01)	**7.70**(±**0.22**)
blocks	0.98(±0.05)	25.4(±2.58)	**1.00**(±**0.00**)	**11.0**(±**0.13**)

Table 5. Best accuracy results in SVM-DC-based and discriminant chronicle-based classification and computation times with $\sigma_{min} = 0.4$, $max\_size = 3$ and $g_{min} = 2$.

for DT-DC than for discriminant chronicles and, thus, that DT-DC overfit more the datasets than discriminant chronicles. The same behavior is observed in most of the experiments.

Finally, Table 5 shows the accuracy of SVM-DC and discriminant chronicles for real datasets: *asl-bu*, *asl-gt* and *blocks*. The parameters used for these results were obtained through a grid search. The involved parameters are σ_{min}, g_{min} and γ but also the C parameter of the global linear SVM classifier. Table 5 shows that SVM-DC produces patterns with better accuracy than discriminant chronicles on *asl-bu* and *asl-gt*. For *blocks*, discriminant chronicles are not more accurate than SVM-DC but the discriminant chronicles are discriminant enough to describe such a simple dataset. A classifier based on chronicles is perfect to classify the *blocks* sequences.

These results show that combining decisions of discriminant chronicles makes discriminant less competitive than SVM-DC, even if discriminant chronicles are individually accurate. Thereby, we cannot conclude from previous results that discriminant chronicles are the most accurate GDC. Indeed, chronicles do not

involve all the occurrences of a multiset and represent very specific discriminant behaviors. But in a pattern-based classification context, a set of very discriminant chronicles is not sufficient to cover the whole dataset and so to obtain a good accuracy. This leads to a typical overfitting situation.

Finally, Table 5 also gives the mean computation times for both approaches. These times are strongly related to the computing times of machine algorithms which vary a lot depending on datasets. Discriminant chronicle mining (DCM) is faster in most cases thanks to a particular implementation effort for this approach.

7 Conclusion and Perspectives

This article presents a generalization of the model of discriminant chronicles. The model of generalized discriminant chronicles (GDC) proposes to combine a multiset pattern and a decision function learned from the temporal duration between occurrences of a multiset pattern. Initially, discriminant chronicles were extracted using a rule learner and their temporal boundaries were intervals. Such a representation may be too restrictive an assumption on how to discriminate temporal sequences and, thus, had to be generalized.

We demonstrate the expressiveness of the framework by showing that it can model classical patterns (episodes, sequential patterns and chronicles) and episodes, sequential patterns and new types of patterns. DT-DC are based on decision tree classifiers and SVM-DC are based on a SVM classifier.

The experiments show that individual chronicles have good accuracy but SVM-DC overtakes the combination of chronicles on real datasets. An interesting perspective of this work is to blend different types of chronicles within the same combination. Furthermore, comparison in terms of interpretability between several GDC instances would be interesting. Indeed, chronicles are attractive for its interpretability, thanks to its graphical representation. However, new temporal patterns like DT-DC or SVM-DC can not be graphically represented so simply. Then it would be possible to suggest GDC instances that would offer a tradeoff between prediction accuracy and interpretability.

Acknowledgements

This work has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement No [694980] SYNTH: Synthesising Inductive Data Models).

References

1. Asker, L., Boström, H., Karlsson, I., Papapetrou, P., Zhao, J.: Mining candidates for adverse drug interactions in electronic patient records. In: Proceedings of the International Conference on PErvasive Technologies Related to Assistive Environments (PETRA). pp. 22:1–22:4 (2014)

2. Atzmueller, M.: Subgroup discovery. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery **5**(1), 35–49 (2015)
3. Bringmann, B., Nijssen, S., Zimmermann, A.: Pattern-based classification: a unifying perspective. In: Proceedings of the LeGo Workshop "From Local Patterns to Global Models". p. 10 (2009)
4. Cohen, W.W.: Fast effective rule induction. In: Proceedings of the International Conference on Machine Learning. pp. 115–123 (1995)
5. Cram, D., Mathern, B., Mille, A.: A complete chronicle discovery approach: application to activity analysis. Expert Systems **29**(4), 321–346 (2012)
6. Dauxais, Y., Guyet, T., Gross-Amblard, D., Happe, A.: Discriminant chronicles mining - application to care pathways analytics. In: Proceedings of 16th Conference on Artificial Intelligence in Medicine (AIME). pp. 234–244 (2017)
7. Dong, G., Li, J.: Efficient mining of emerging patterns: Discovering trends and differences. In: Proceedings of the International conference on Knowledge discovery and data mining (KDD). pp. 43–52 (1999)
8. Fabrègue, M., Braud, A., Bringay, S., Grac, C., Le Ber, F., Levet, D., Teisseire, M.: Discriminant temporal patterns for linking physico-chemistry and biology in hydro-ecosystem assessment. Ecological informatics **24**, 210–221 (2014)
9. Fradkin, D., Mörchen, F.: Mining sequential patterns for classification. Knowledge and Information Systems **45**(3), 731–749 (2015)
10. Giannotti, F., Nanni, M., Pedreschi, D.: Efficient mining of temporally annotated sequences. In: Proceedings of the International Conference on Data Mining (ICDM). pp. 348–359 (2006)
11. Höppner, F.: Discovery of temporal patterns. In: European Conference on Principles of Data Mining and Knowledge Discovery. pp. 192–203 (2001)
12. Lipton, Z.C., Berkowitz, J., Elkan, C.: A critical review of recurrent neural networks for sequence learning. arXiv preprint arXiv:1506.00019 (2015)
13. Mabroukeh, N.R., Ezeife, C.I.: A taxonomy of sequential pattern mining algorithms. ACM Computing Surveys (CSUR) **43**(1), 3:1–3:41 (2010)
14. Mannila, H., Toivonen, H., Verkamo, A.I.: Discovery of frequent episodes in event sequences. Data mining and knowledge discovery **1**(3), 259–289 (1997)
15. Nanni, M., Rigotti, C.: Extracting trees of quantitative serial episodes. In: International Workshop on Knowledge Discovery in Inductive Databases. pp. 170–188. Springer (2006)
16. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research **12**, 2825–2830 (2011)
17. Quinlan, J.R.: Learning decision tree classifiers. ACM Computing Surveys (CSUR) **28**(1), 71–72 (1996)
18. Srikant, R., Agrawal, R.: Mining sequential patterns: Generalizations and performance improvements. Advances in Database Technology – EDBT pp. 1–17 (1996)
19. Yen, S.J., Lee, Y.S.: Mining non-redundant time-gap sequential patterns. Applied Intelligence **39**(4), 727–738 (2013)