Hierarchies of Weighted Closed Partially-Ordered Patterns for Enhancing Sequential Data Analysis

Cristina Nica, Agnès Braud, Florence Le Ber

Hierarchies of Weighted Closed Partially-Ordered Patterns for Enhancing Sequential Data Analysis

Cristina Nica(✉), Agnès Braud, Florence Le Ber

ICube, University of Strasbourg, CNRS, ENGEES
cristina.nica@engees.unistra.fr, agnes.braud@unistra.fr,
florence.leber@engees.unistra.fr
http://icube-sdc.unistra.fr

Abstract. Discovering sequential patterns in sequence databases is an important data mining task. Recently, hierarchies of closed partially-ordered patterns (cpo-patterns), built directly using Relational Concept Analysis (RCA), have been proposed to simplify the interpretation step by highlighting how cpo-patterns relate to each other. However, there are practical cases (e.g. choosing interesting navigation paths in the obtained hierarchies) when these hierarchies are still insufficient for the expert. To address these cases, we propose to extract hierarchies of more informative cpo-patterns, namely weighted cpo-patterns (wcpo-patterns), by extending the RCA-based approach. These wcpo-patterns capture and explicitly show not only the order on itemsets but also their different influence on the analysed sequences. We illustrate how the proposed wcpo-patterns can enhance sequential data analysis on a toy example.

1 Introduction

Searching for sequential patterns [1] is a well-known data mining task whose aim is to find regularities and tendencies in sequential data that can be interpreted and assessed by experts. Various algorithms have therefore been proposed [9] and many of them focus on extracting efficiently concise representations of sequential patterns (e.g. closed sequential patterns [15]). To obtain a more compact set of such sequential patterns, efficient algorithms for directly mining closed partially-ordered patterns (cpo-patterns, [2]) were proposed in [12,5]. Precisely, a cpo-pattern summarises a set of closed sequential patterns, which coexist in the same sequences, and it has a graphical representation that facilitates the interpretation step. However, regardless of the fewer number of obtained cpo-patterns, the interpretation step remains difficult since these cpo-patterns are unorganized.

In [10], Relational Concept Analysis (RCA, [13]) is used to directly extract hierarchies of cpo-patterns that help the interpretation step by highlighting the relationships between cpo-patterns. Indeed, RCA classifies sets of objects described by attributes and relations, allowing the discovery of hierarchies of patterns. Nica et al. have proposed to extract cpo-patterns by navigating only the

intents of the interrelated concepts from the RCA result, i.e. a family of concept lattices, beginning with concept intents from the *main lattice*. The extracted hierarchies of cpo-patterns help in understanding the obtained knowledge and provide a quick way to navigate to interesting cpo-patterns.

Nevertheless, cpo-patterns still do not capture all the particularities hidden in the analysed sequential data. A cpo-pattern considers only the order on itemsets in its supporting sequences, and, besides, the itemsets are treated uniformly even if they can have different roles in these sequences. In fact, previous studies showed that exploiting the time information from the analysed sequences, such as capturing time-intervals between adjacent itemsets [4] in the mined sequential patterns, leads to more valuable knowledge. In addition, there are practical cases (e.g. choosing among cpo-patterns that have the same frequency in the analysed data) when the hierarchical order on the extracted cpo-patterns given by the lattice is still insufficient for the expert. In contrast to existing works, here, we propose to study and measure the repetitive occurrences of *preceded itemsets* in a cpo-pattern, i.e. itemsets with specific predecessors; this measure may show the non-accidental occurrence of such itemsets in the considered sequences.

To address the aforementioned limitations, this paper focuses on extracting hierarchies of more informative cpo-patterns, namely *weighted cpo-patterns* (wcpo-patterns), that capture and explicitly show the different weightiness of itemsets. These hierarchies can be directly obtained by extending the RCA-based extraction method presented in [10]. Briefly, we suggest as well to navigate the extents of the interrelated concept that reveal the different weightiness of preceded itemsets in the analysed sequential data. Accordingly, by exploiting the RCA result, we extract hierarchies of wcpo-patterns that better characterise the analysed sequential data.

Our paper is structured as follows. In Section 2 we give the theoretical background of our work. Section 3 introduces a running medical example and details how to explore it using RCA. Section 4 formally defines our proposal for mining directly wcpo-patterns. Then, we illustrate how the proposed wcpo-patterns can enhance the sequential data analysis in Section 5. Finally, we present an overview of the related work in Section 6, and conclude the paper in Section 7.

2 Preliminaries

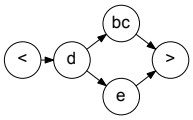
Our approach relies both on sequential patterns and formal concept analysis domains.

2.1 Sequences, Sequential Patterns and PO-patterns

Let $\mathcal{I} = \{I_1, I_2, ..., I_m\}$ be a set of *items*. An *itemset* IS is a non empty, unordered, set of items, $IS = (I_{j_1}...I_{j_k})$ where $I_{j_i} \in \mathcal{I}$. Let $\mathcal{IS}$ be the set of all itemsets built from $\mathcal{I}$. A *sequence* S is a non empty ordered list of itemsets, $S = \langle IS_1 IS_2 ... IS_p \rangle$ where $IS_j \in \mathcal{IS}$. The sequence S is a *subsequence* of another sequence $S' = \langle IS'_1 IS'_2 ... IS'_q \rangle$, denoted as $S \preceq_s S'$, if $p \leq q$ and if there are

Sequence Id	Sequence
S_1	$\langle(d)(bce)\rangle$
S_2	$\langle(ad)(bc)(e)\rangle$

(a) $\mathcal{D}_\mathcal{S}$



(b) $\mathcal{G}_1$

Fig. 1: (a) $\mathcal{D}_\mathcal{S}$ a sequence database; (b) $\mathcal{G}_1$ cpo-pattern

integers $j_1 < j_2 < ... < j_k < ... < j_p$ such that $IS_1 \subseteq IS'_{j_1}, IS_2 \subseteq IS'_{j_2}, ..., IS_p \subseteq IS'_{j_p}$. An item can occur only once in an itemset, but can occur several times in different itemsets of the same sequence.

Sequential patterns have been defined by [1] as frequent subsequences found in a sequential dataset. A sequential pattern is associated to a support, i.e. the number of sequences containing the pattern, that has to be greater than or equal to a minimum support, denoted by θ. Formally, the support of a sequential pattern M extracted from a sequential dataset $\mathcal{D}_\mathcal{S}$ is defined as $Support(M) = |\{S \in \mathcal{D}_\mathcal{S} | M \preceq_s S\}|$. For instance, $M_1 = \langle(d)(bc)\rangle$ and $M_2 = \langle(d)(e)\rangle$ are two sequential patterns found in Fig. 1(a) sequence database for $\theta = 2$.

Partially-ordered patterns, *po-patterns*, have been introduced by [2], to synthesise sets of sequential patterns. Formally, a *po-pattern* is a directed acyclic graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, l)$. $\mathcal{V}$ is the set of vertices, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of directed edges, and l is the labelling function mapping each vertex to an itemset. With such a structure, we can determine a strict partial order on vertices u and v such that $u \neq v : u < v$ if there is a directed path from u to v. However, if there is no directed path from u to v, these elements are not comparable. Each path of the graph represents a sequential pattern, and the set of paths in $\mathcal{G}$ is denoted by $\mathcal{P}_\mathcal{G}$. A po-pattern is associated to the set of sequences $\mathcal{S}_\mathcal{G}$ that contain all paths of $\mathcal{P}_\mathcal{G}$. The support of a po-pattern is defined as $Support(\mathcal{G}) = |\mathcal{S}_\mathcal{G}| = |\{S \in \mathcal{D}_\mathcal{S} | \forall M \in \mathcal{P}_\mathcal{G}, M \preceq_s S\}|$. Furthermore, let $\mathcal{G}$ and $\mathcal{G}'$ be two po-patterns with $\mathcal{P}_\mathcal{G}$ and $\mathcal{P}_{\mathcal{G}'}$ their sets of paths. $\mathcal{G}'$ is a sub po-pattern of $\mathcal{G}$, denoted by $\mathcal{G}' \preceq_g \mathcal{G}$, if $\forall M' \in \mathcal{P}_{\mathcal{G}'}, \exists M \in \mathcal{P}_\mathcal{G}$ such that $M' \preceq_s M$. A po-pattern $\mathcal{G}$ is *closed*, denoted *cpo-pattern*, if there exists no po-pattern $\mathcal{G}'$ such that $\mathcal{G} \prec_g \mathcal{G}'$ with $\mathcal{S}_\mathcal{G} = \mathcal{S}_{\mathcal{G}'}$. For example, Fig. 1(b) shows $\mathcal{G}_1$ cpo-pattern that synthesises M_1 and M_2 sequential patterns that coexist exactly in the same sequences S_1 and S_2.

2.2 FCA and RCA

Formal Concept Analysis (FCA, [6]) considers an object-attribute context which is a set of objects described by attributes, and builds from it a concept lattice used to analyse the objects. Concisely, an object-attribute context K is a 3-tuple (G, M, I), where G is a set of objects, M a set of attributes, and $I \subseteq G \times M$ an incidence relation. $C = (X, Y)$ where $X = \{g \in G | \forall m \in Y, (g, m) \in I\}$ and $Y = \{m \in M | \forall g \in X, (g, m) \in I\}$ is a formal concept built from K. X and Y

are respectively the extent and the intent of the concept. Let $\mathcal{C}_K$ be the set of all formal concepts that can be built on K. Let $C_1 = (X_1, Y_1)$ and $C_2 = (X_2, Y_2)$ be two concepts from $\mathcal{C}_K$, the concept generalisation order $\preceq_K$ is here defined by $C_1 \preceq_K C_2$ iff $X_1 \subseteq X_2$ ($\Leftrightarrow Y_2 \subseteq Y_1$). $\mathcal{L}_K$=($\mathcal{C}_K$, $\preceq_K$) is the concept lattice built from K. We denote by $\top(\mathcal{L}_K)$ the concept from $\mathcal{L}_K$ whose extent has all the objects, and by $\bot(\mathcal{L}_K)$ the concept from $\mathcal{L}_K$ whose intent has all the attributes.

RCA extends the purpose of FCA to relational data. RCA applies iteratively FCA on a Relational Context Family (RCF). An RCF is a pair $(\mathcal{K}, \mathcal{R})$, where $\mathcal{K}$ is a set of object-attribute contexts and $\mathcal{R}$ is a set of object-object contexts. $\mathcal{K}$ contains n object-attribute contexts $K_i = (G_i, M_i, I_i)\,, i \in \{1, ..., n\}$. $\mathcal{R}$ contains m object-object contexts $R_j = (G_k, G_l, r_j)\,, j \in \{1, ..., m\}$, where $r_j \subseteq G_k \times G_l$ is a binary relation with $k, l \in \{1, ..., n\}$, $G_k = dom(r_j)$ the domain of the relation and $G_l = ran(r_j)$ the range of the relation. G_k and G_l are the sets of objects of the object-attribute contexts K_k and K_l, respectively. RCA relies on a relational scaling mechanism that is used to transform a relation r_j into a set of *relational attributes* that extends the object-attribute context describing the set of objects $dom(r_j)$. A relational attribute $\exists r_j(C)$, where $\exists$ is the existential quantifier, and $C = (X, Y)$ is a concept whose extent contains objects from the $ran(r_j)$, describes an object $g \in dom(r_j)$ if $r_j(g) \cap X \neq \emptyset$. Other quantifiers can be found in [13]. RCA process consists in applying FCA first on each object-attribute context of an RCF, and then iteratively on each object-attribute context extended by the relational attributes created using the concepts from the previous step. The RCA result is obtained when the families of lattices of two consecutive steps are isomorphic and the object-attribute contexts are unchanged.

3 Relational Analysis of Sequential Data

3.1 Running Example

Patterns hidden in sequential medical data about patients and their medical histories can provide valuable medical knowledge for physicians. Here, we propose to study the symptoms (e.g. fever and cough) that indicate the presence of viruses (e.g. influenza) in patients. The symptoms and viruses are detected by medical examinations and viral tests, respectively. In Fig. 2 is a medical sequence, i.e. a chronologically ordered set of medical examinations with a viral test at the end, all undergone by the same patient. The medical examinations are itemsets of symptoms, while the viral test is the *target 1-itemset* (set of only one item) that contains the studied *object of interest* (here, the influenza virus).

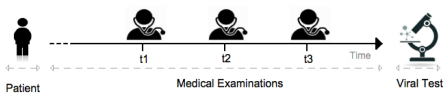


Fig. 2: Medical sequence

Table 1: Medical toy sequential dataset.

Sequence Id	Sequence
S1	$\langle$(FEVER$_{\text{moderate}}$)(FEVER$_{\text{moderate}}$ COUGH$_{\text{high}}$)(FEVER$_{\text{high}}$ COUGH$_{\text{high}}$)(FEVER$_{\text{moderate}}$)(Influenza$_{\text{A}}$)$\rangle$
S2	$\langle$(FEVER$_{\text{moderate}}$)(FEVER$_{\text{high}}$ COUGH$_{\text{high}}$)(Influenza$_{\text{A}}$)$\rangle$
S3	$\langle$(COUGH$_{\text{moderate}}$ FEVER$_{\text{moderate}}$)(Influenza$_{\text{A}}$)$\rangle$
S4	$\langle$(FEVER$_{\text{moderate}}$)(FEVER$_{\text{moderate}}$ COUGH$_{\text{high}}$)(FEVER$_{\text{high}}$ COUGH$_{\text{high}}$)(Influenza$_{\text{A}}$)$\rangle$
S5	$\langle$(FEVER$_{\text{moderate}}$)(FEVER$_{\text{high}}$)(FEVER$_{\text{high}}$)(COUGH$_{\text{high}}$)(Influenza$_{\text{A}}$)$\rangle$
S6	$\langle$(FEVER$_{\text{moderate}}$)(COUGH$_{\text{high}}$)(FEVER$_{\text{high}}$)(FEVER$_{\text{high}}$)(Influenza$_{\text{B}}$)$\rangle$
S7	$\langle$(FEVER$_{\text{moderate}}$)(COUGH$_{\text{high}}$ FEVER$_{\text{high}}$)(COUGH$_{\text{high}}$)(Influenza$_{\text{B}}$)$\rangle$
S8	$\langle$(COUGH$_{\text{moderate}}$ FEVER$_{\text{moderate}}$)(Influenza$_{\text{B}}$)$\rangle$
S9	$\langle$(FEVER$_{\text{moderate}}$)(FEVER$_{\text{high}}$)(COUGH$_{\text{high}}$)(Influenza$_{\text{B}}$)$\rangle$
S10	$\langle$(FEVER$_{\text{moderate}}$)(FEVER$_{\text{high}}$ COUGH$_{\text{high}}$)(Influenza$_{\text{B}}$)$\rangle$

Table 1 illustrates a medical toy example of sequential data. We consider only pertinent medical sequences to recognize influenza outbreaks (e.g. only medical examinations undergone by patients within two weeks before a viral test). There are three items as follows: two symptoms `FEVER` and `COUGH`, and one virus `Influenza`. The symptoms can be `moderate` or `high`, while the influenza virus can be of type `A` or `B`. Thus, in this example we deal with *qualitative sequential data*. For instance, *S3* sequence consists in one medical examination undergone by a patient who experienced moderate cough and moderate fever before being diagnosed with influenza A virus.

The physicians try to assess the cough and fever symptoms felt by patients to better understand how to early recognize outbreaks of influenza A/B virus, and, besides, to distinguish between influenza A and influenza B outbreaks. To this end, we build qualitative sequential sub-datasets from Tab. 1, based on the diagnosed type of influenza. Hence, there are two sub-datasets referred to as $\mathcal{D}_{\mathcal{S}fluA}$ (sequences from *S1* to *S5*) and $\mathcal{D}_{\mathcal{S}fluB}$ (sequences from *S6* to *S10*).

3.2 Preprocessing Qualitative Sequential Data

In this section and in Section 3.3, we briefly recall and improve the temporal relational analysis step proposed in [10], and we exemplify it with only $\mathcal{D}_{\mathcal{S}fluA}$. Exploiting the relational character of our medical qualitative sequential data given in Tab. 1, we define a *temporal model* for $\mathcal{D}_{\mathcal{S}fluA}$ composed of four sets of objects, as follows: viruses (V), symptoms (S), viral tests (VT), and medical examinations (ME). The viral tests are linked to viruses by a qualitative binary relation *has virus A*. Similarly, medical examinations are linked to symptoms by qualitative relations *has symptom* (mS or hS) differentiated by the type of identified symptoms, e.g. *moderate* or *high*. Viral tests/medical examinations and medical examinations are linked by a temporal binary relation *is preceded by* (ipb) that associates a viral test/medical examination to a medical examination if the viral test/medical examination is preceded in time by the medical examination. There is no temporal binary relation between viral tests since our aim is to study the symptoms that prognosticate influenza A virus.

As explained in Section 3.1, the set of viruses contains only one object (item) `Influenza` and the set of symptoms contains two objects `COUGH` and `FEVER`. In

Table 2: $\mathcal{D}'_{\mathcal{S}fluA}$ of unique identifiers.

Sequence
⟨IS1_Seq1 IS2_Seq1 IS3_Seq1 IS4_Seq1 Seq1⟩
⟨IS1_Seq2 IS2_Seq2 Seq2⟩
⟨IS1_Seq3 Seq3⟩
⟨IS1_Seq4 IS2_Seq4 IS3_Seq4 Seq4⟩
⟨IS1_Seq5 IS2_Seq5 IS3_Seq5 IS4_Seq5 Seq5⟩

order to build the set of viral tests and the one of medical examinations (since an itemset can correspond to several viral tests or medical examinations in the analysed sequences, each occurrence of the itemset should be uniquely identified), we re-modelled our medical sequences of itemsets as the medical sequences of unique identifiers (UIDs) given in Tab. 2.

Formally, let $\mathcal{D}_{\mathcal{S}}$ be a sequential dataset and $S \in \mathcal{D}_{\mathcal{S}}$ a sequence of itemsets. We model S as ⟨IS1_Seq IS2_Seq...ISp_Seq Seq⟩, that is, a sequence of UIDs. Let $\mathcal{D}'_{\mathcal{S}}$ be the set of all such sequences of UIDs derived from $\mathcal{D}_{\mathcal{S}}$ sequences. Seq$i$ is the UID of the *target 1-itemset* and it uniquely identifies the sequence S. We define $G_m = \{\text{Seq}i\}_{i\in[1,n]}$, where $n = |\mathcal{D}'_{\mathcal{S}}|$, as the set of all UIDs of the target 1-itemsets in $\mathcal{D}'_{\mathcal{S}}$. The function $getS : G_m \rightarrow \mathcal{D}_{\mathcal{S}}$ maps a target 1-itemset UID to the corresponding sequence of itemsets. ISj_Seq$i$ is the UID of an itemset and specifies Seq$i$ sequence that owns the itemset. We define $G_t = \{\text{IS}j\text{\_Seq}i\}_{i\in[1,n];j\in[1,l_i]}$, where l_i is the number of itemsets (except the target 1-itemset) in Seq$i$ sequence, as the set of all itemset UIDs, excluding G_m, in $\mathcal{D}'_{\mathcal{S}}$. The function $getSeq : G_t \rightarrow G_m$ maps an itemset UID to the sequence that owns it. The function $getIS : G_m \cup G_t \rightarrow \mathcal{IS}$ maps an itemset/target 1-itemset UID to the corresponding itemset.

For instance, $\mathcal{D}'_{\mathcal{S}fluA}$ (Tab. 2) is a sequence database of UIDs, where G_m is the set of all viral test UIDs, while G_t is the set of all medical examination UIDs. The third sequence ⟨IS1_Seq3 Seq3⟩ is derived from $getS$(Seq3) = *S3* (Tab. 1). Seq3 uniquely identifies the viral test $getIS$(Seq3) = ($\text{Influenza}_{\text{A}}$) in *S3*. IS1_Seq3 is owned by $getSeq$(IS1_Seq3) = Seq3 and it uniquely identifies the medical examination $getIS$(IS1_Seq3) = ($\text{COUGH}_{\text{moderate}}$ $\text{FEVER}_{\text{moderate}}$) in *S3*.

3.3 Exploring Qualitative Sequential Data Using RCA

Firstly, the RCA input (RCF) – an excerpt is depicted in Tab. 3 – is built following the temporal modelling described in Section 3.2. Tables KS (symptoms), KVT (viral tests) and KME (medical examinations) represent object-attribute contexts. Let us note that, KME has no column since a medical examination is described only using *has symptom* qualitative relations, and the rows represent the UIDs of medical examinations from Tab. 2. There is no object-attribute context of viruses since we focus on a specific virus and thus all viral tests detect influenza A. Tables RVT-ipb-ME (viral test *ipb* medical examination), RME-ipb-ME (medical examination *ipb* medical examination), RmS (medical examination detects a moderate symptom) and RhS (medical examination detects a high symptom) represent object-object contexts. For example, RVT-ipb-ME has viral tests as

Table 3: RCF excerpt composed of object-attribute contexts (KS, KVT and KME), and object-object contexts (RVT-ipb-ME and RhS).

KS	FEVER	COUGH
FEVER	×	
COUGH		×

KVT
Seq1
Seq2
Seq3
Seq4
Seq5

KME
IS1_Seq1
IS2_Seq1
IS3_Seq1
IS4_Seq1
IS1_Seq2
IS2_Seq2
IS1_Seq3
IS1_Seq4
IS2_Seq4
IS3_Seq4
IS1_Seq5
IS2_Seq5
IS3_Seq5
IS4_Seq5

RVT-ipb-ME	IS1_Seq1	IS2_Seq1	IS3_Seq1	IS4_Seq1	IS1_Seq2	IS2_Seq2	IS1_Seq3	IS1_Seq4	IS2_Seq4	IS3_Seq4	IS1_Seq5	IS2_Seq5	IS3_Seq5	IS4_Seq5
Seq1	×	×	×	×										
Seq2					×	×								
Seq3							×							
Seq4								×	×	×				
Seq5											×	×	×	×

RhS	FEVER	COUGH
IS1_Seq1		
IS2_Seq1		×
IS3_Seq1	×	×
IS4_Seq1		
IS1_Seq2		
IS2_Seq2	×	×
IS1_Seq3		
IS1_Seq4		
IS2_Seq4		×
IS3_Seq4	×	×
IS1_Seq5		
IS2_Seq5	×	
IS3_Seq5	×	
IS4_Seq5		×

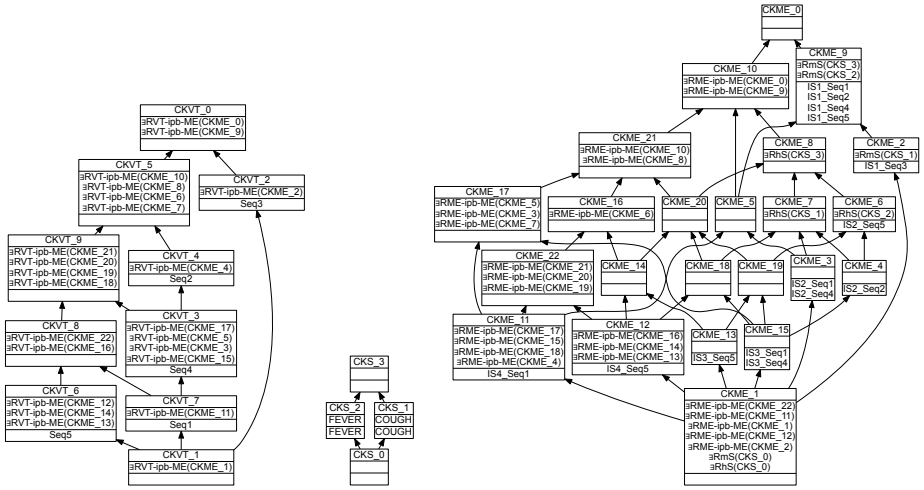
rows and medical examinations as columns. A cross indicates a link between objects, e.g. the cell identified by the viral test Seq3 and the medical examination IS1_Seq3 contains a cross since both are undergone by the same patient and the medical examination precedes the viral test, as shown in Tab. 2.

Secondly, RCA is applied[1] on the aforementioned RCF and the family of concept lattices given in Fig. 3 is obtained after four iterations. There is a concept lattice for each object-attribute context as follows: $\mathcal{L}_{\texttt{KVT}}$ (viral tests), $\mathcal{L}_{\texttt{KS}}$ (symptoms) and $\mathcal{L}_{\texttt{KME}}$ (medical examinations). $\mathcal{L}_{\texttt{KVT}}$ is considered as the *main lattice* since it contains the target 1-itemsets. $\mathcal{L}_{\texttt{KME}}$ is considered as the *temporal lattice* since it describes the temporal links between the itemsets. $\mathcal{L}_{\texttt{KME}}$ and $\mathcal{L}_{\texttt{KVT}}$ are modified during the iterative steps due to the qualitative and temporal relations that have respectively as domain the set of objects of KME and KVT. Each concept is represented by a box structured from top to bottom as follows: concept name, simplified intent, simplified extent. The representation of each lattice is simplified as every attribute/object is top-down/bottom-up inherited. The navigation amongst these lattices follows the concepts used to build relational attributes, e.g. the relational attribute $\exists$RVT-ipb-ME(CKME_6), which is a temporal one since it introduces the temporal relation *is preceded by*, of the concept intent CKVT_5 in $\mathcal{L}_{\texttt{KVT}}$ lattice allows us to navigate from CKVT_5 to CKME_6 concept in $\mathcal{L}_{\texttt{KME}}$ lattice.

4 Extracting WCPO-patterns from the RCA Result

In [10], Nica et al. have shown how to directly extract hierarchies of cpo-patterns by navigating interrelated concept intents from the RCA result (which is built as explained in Section 3) beginning with concept intents from the main lattice. For each navigated concept intent, a vertex (itemset) is derived from all qualitative relational attributes, while an edge is derived from each temporal relational attribute. When a cpo-pattern is extracted the order on itemsets in the analysed sequences is exploited, while the itemsets themselves are considered uniformly.

[1] using RCAExplore tool (http://dolques.free.fr/rcaexplore)



(a) $\mathcal{L}_{\texttt{KVT}}$ (main lattice) (b) $\mathcal{L}_{\texttt{KS}}$ (c) $\mathcal{L}_{\texttt{KME}}$ (temporal lattice)

Fig. 3: $\mathcal{L}_{\texttt{KVT}}$ lattice of viral tests, $\mathcal{L}_{\texttt{KS}}$ lattice of symptoms and $\mathcal{L}_{\texttt{KME}}$ lattice of medical examinations obtained by applying RCA on the RCF given in Tab. 3

Figure 4, on the left hand side, illustrates a set of concepts whose intents are navigated starting with CKVT_4 from the main lattice $\mathcal{L}_{\texttt{KVT}}$ (Fig. 3(a)); on the right hand side, is depicted the extracted cpo-pattern, which is contained in each sequence of CKVT_4 extent (*S1*, *S2* and *S4*). The last vertex of the cpo-pattern is derived from the first navigated concept intent. It is noted that the cpo-pattern preserves the order on itemsets in these sequences. However, the cpo-pattern can be misleading if the itemsets have different numbers of occurrences in these sequences. For instance, the cpo-pattern given in Fig. 4 does not encapsulate that in our sequences there are 3 occurrences of ($\texttt{FEVER}_{\texttt{high}}$ $\texttt{COUGH}_{\texttt{high}}$) itemset when each occurrence is preceded by ($\texttt{FEVER}_{\texttt{moderate}}$) itemset, and 5 occurrences of ($\texttt{FEVER}_{\texttt{moderate}}$) with no constraint on its order.

Formally, let $\mathcal{G} = (\mathcal{V}, \mathcal{E}, l)$ be a cpo-pattern, and $\mathcal{S}_{\mathcal{G}}$ the set of sequences that support $\mathcal{G}$. Let $v_i \in \mathcal{V}$ be a vertex of $\mathcal{G}$, and $\mathcal{V}_i = \{v \in \mathcal{V} | v \leq v_i\}$ the set of predecessors of v_i in $\mathcal{G}$ (including v_i). Furthermore, $\mathcal{E}_i = \{(v_k, v_l) \in \mathcal{E} | v_k \in \mathcal{V}_i \text{ and } v_l \in \mathcal{V}_i\}$. $\mathcal{G}_i = (\mathcal{V}_i, \mathcal{E}_i, l)$ is a sub-graph of $\mathcal{G}$, associated to vertex v_i. Let

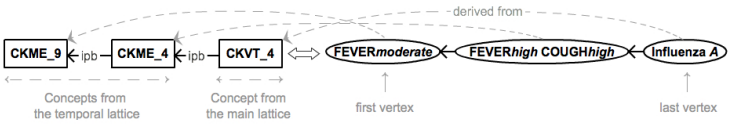


Fig. 4: From a set of navigated concepts to a cpo-pattern

introduce $IS_i \supseteq l(v_i)$ an itemset in a sequence $S \in \mathcal{S}_\mathcal{G}$. IS_i is a *preceded itemset* w.r.t. $v_i \in \mathcal{V}$, iff $\exists S_i \preceq_s S, S_i = \langle IS_1 IS_2 ... IS_p IS_i \rangle$ and $\forall M \in \mathcal{P}_{\mathcal{G}_i}, M \preceq_s S_i$ (i.e. there exists a subsequence of S, ending with IS_i, that supports $\mathcal{G}_i$).

Actually, each sequence of $\mathcal{S}_\mathcal{G}$ can repeatedly contain the same itemset having the same predecessors. In the following, we are going to show how to capture such information (that reveal the weightiness of vertices) by additionally navigating the interrelated concept extents to obtain more informative cpo-patterns.

4.1 From Uniform Vertices to Weighted Vertices

Our purpose is to formalise an approach for determining the weightiness of vertices (derived from concepts of the temporal lattice) that correspond to itemsets with specific predecessors, namely *preceded itemsets*. To this end, as explained in Section 2.1, let $\mathcal{D}_\mathcal{S}$ be a sequence dataset re-modelled as $\mathcal{D}'_\mathcal{S}$ of UIDs. G_m is the set of all target 1-itemset UIDs in $\mathcal{D}'_\mathcal{S}$, while G_t is the set of all other itemset UIDs.

Let $\mathcal{L}_{K_m} = (\mathcal{C}_{K_m}, \preceq_{K_m})$ be the main lattice (e.g. the lattice of viral tests $\mathcal{L}_{\texttt{KVT}}$) whose set of main concepts $\mathcal{C}_{K_m}$ is derived from the formal context $K_m = (G_m, M_m, I_m)$. G_m is the domain of a temporal relation *is preceded by*, denoted by $ipb_1 \subseteq G_m \times G_t$ (e.g. viral test ipb_1 medical examination). A main concept $C_m \in \mathcal{C}_{K_m}$ is a pair (X_m, Y_m) such that:

- **the intent** Y_m consists of temporal relational attributes that are navigated to reveal $\mathcal{G}_{C_m} = (\mathcal{V}_m, \mathcal{E}_m, l_m)$ cpo-pattern whose last vertex v_m is the one derived from C_m; v_m is labelled with the *target 1-itemset*;
- **the extent** X_m gathers all UIDs in G_m of the sequences that contain all paths in $\mathcal{G}_{C_m}$; $\mathcal{S}_{\mathcal{G}_{C_m}} = \{getS(\texttt{Seq}) \in \mathcal{D}_\mathcal{S} | \texttt{Seq} \in X_m\}$.

Note that the range of ipb_1 temporal relation is G_t, and thus the set of vertices $\mathcal{V}_m$ contains one or more vertices v_t derived from temporal concepts, and v_m vertex. Indeed, let $\mathcal{L}_{K_t} = (\mathcal{C}_{K_t}, \preceq_{K_t})$ be the temporal lattice (e.g. the lattice of medical examinations $\mathcal{L}_{\texttt{KME}}$) whose set of temporal concepts $\mathcal{C}_{K_t}$ is derived from $K_t = (G_t, M_t, I_t)$ context. G_t is both the domain and the range of a second temporal relation *is preceded by*, denoted by $ipb_2 \subseteq G_t \times G_t$ (e.g. medical examination ipb_2 medical examination). A temporal concept $C_t \in \mathcal{C}_{K_t}$ is a pair (X_t, Y_t) such that:

- **the intent** Y_t contains temporal relational attributes that are navigated to reveal $\mathcal{G}_{C_m} = (\mathcal{V}_m, \mathcal{E}_m, l_m)$ cpo-pattern whose vertex v_t is derived from C_t; v_t vertex is labelled with $l_m(v_t)$ itemset;
- **the extent** X_t gathers all UIDs in G_t that identify itemsets containing the itemset $l_m(v_t)$ and respect the temporal order with the UIDs pointed by temporal relational attributes of Y_t. We introduce $X_{t|m} = \{\texttt{IS\_Seq} \in X_t | getSeq(\texttt{IS\_Seq}) \in X_m\}$.

Proposition 1. *$X_{t|m}$ is the set of all UIDs that identify a preceded itemset w.r.t. $v_t \in \mathcal{V}_m$. Furthermore, X_t is the set of all UIDs that identify a preceded itemset w.r.t. $v_t^k \in \mathcal{V}_m^k$, with $k \in \{1, ..., |\mathcal{L}_{K_m}|\}$.*

Proof. Let IS_i be a preceded itemset w.r.t. $v_t \in \mathcal{V}_m$. Then $IS_i \supseteq l_m(v_t)$ and $\exists S \in \mathcal{D}_\mathcal{S}, \exists S_i \preceq_s S$ such that IS_i is the last itemset in S_i and S_i supports the sub-graph of v_t predecessors in $\mathcal{G}_{C_m}$ while S supports $\mathcal{G}_{C_m}$. Let `Seq` be the UID of S: $\texttt{Seq} \in X_m$. Furthermore, the UID of IS_i, i.e. $\texttt{IS}_i\texttt{\_Seq}$, owns all temporal relational attributes of Y_t and is thus included in $X_{t|m}$.

Let C_t be a temporal concept revealing a vertex $v_t \in \mathcal{V}_m$. Let $\texttt{IS}_i\texttt{\_Seq} \in X_{t|m}$ be the UID of $IS_i = getIS(\texttt{IS}_i\texttt{\_Seq}) \supseteq l_m(v_t)$, and $S \in \mathcal{D}_\mathcal{S}$ the sequence referred by $getSeq(\texttt{IS}_i\texttt{\_Seq}) \in X_m$. $IS_i \in S$, S supports the graph $\mathcal{G}_{C_m}$. We can define $S_i \preceq_s S$ the subsequence of S ending with IS_i. Let $\mathcal{G}_t$ be the sub-graph of v_t predecessors in $\mathcal{G}_{C_m}$: $\forall M \in \mathcal{P}_{\mathcal{G}_t}, M \preceq_s S_i$. Thus IS_i is a preceded itemset w.r.t. $v_t \in \mathcal{V}_m$.□

Definition 1 (Weighted CPO-pattern). *Given a main concept C_m, the vertex v_m derived from C_m, the associated cpo-pattern $\mathcal{G}_{C_m} = (\mathcal{V}, \mathcal{E}, l)$, and a function $w : (\mathcal{V} - \{v_m\}) \rightarrow \mathbb{R}^n_{\geq 0}$, where n is constant. A* weighted cpo-pattern *is a quadruple $(\mathcal{V}, \mathcal{E}, l, w)$, i.e. the cpo-pattern $\mathcal{G}_{C_m}$ with the function w that maps each vertex to a n-tuple of real positive numbers (vertex measures of weightiness).*

We propose three vertex measures of weightiness that represent: the *persistency* of the corresponding preceded itemset in the subset of sequences of $\mathcal{D}_\mathcal{S}$ (how many repetitions of it are in that subset); the *overall weight* of the preceded itemset (how often it occurs) in $\mathcal{D}_\mathcal{S}$; the *specificity* of the preceded itemset in the subset of sequences of $\mathcal{D}_\mathcal{S}$ (the extent to which it belongs only to that subset).

In the following, we consider a main concept $C_m = (X_m, Y_m)$, the associated cpo-pattern $\mathcal{G}_{C_m} = (\mathcal{V}, \mathcal{E}, l)$, and a vertex $v_t \in \mathcal{V}$ derived from a temporal concept $C_t = (X_t, Y_t)$.

Definition 2 (Vertex Persistency). *The* persistency *of v_t, denoted by ϖ_{v_t}, is the total number of repetitions (repetitive occurrences in the same sequence) of preceded itemsets w.r.t. v_t.*

$$\varpi_{v_t} = \frac{|X_{t|m}| - |X_m|}{|X_m|} \tag{1}$$

Persistency of a vertex measures the persistence of the corresponding preceded itemset in a subset of the analysed dataset. We consider that the preceded itemset characterizes the subset of the analysed data if it is not accidental, i.e. the preceded itemset occurs repeatedly in the subset.

Definition 3 (Vertex Overall Weight). *The* overall weight *of v_t, denoted by ω_{v_t}, is the total number of occurrences of preceded itemset w.r.t. $v_t^i \in \mathcal{G}^i_{C_m}$, $i \in \{1, ..., |\mathcal{L}_{K_m}|\}$.*

$$\omega_{v_t} = |X_t| \tag{2}$$

Overall Weight of a vertex measures how numerous is the corresponding preceded itemset in all analysed sequences. Therefore, the overall weight provides an overview of the number of occurrences of the preceded itemset in the analysed dataset and it can be a reference point used in decision-making by the expert. Using the overall weight of a vertex v_t, the *overall frequency* of v_t in $\mathcal{D}_\mathcal{S}$ can be computed as $\varphi_{v_t} = \frac{|X_t|}{|G_t|}$.

Definition 4 (Vertex Specificity). *The* specificity *of* v_t*, denoted by* ς_{v_t}*, is the relative number of preceded itemsets w.r.t.* v_t*.*

$$\varsigma_{v_t} = \frac{|X_{t|m}|}{|X_t|}100 \in (0\%, 100\%] \tag{3}$$

Specificity of a vertex measures the extent to which the corresponding preceded itemset is specific for a subset of the analysed data. We consider that the vertex is likely to be more interesting for low values of the specificity, that is, if the preceded itemset characterises the current subset and other sequences from the analysed dataset as well.

Using these three measures, a vertex derived from a temporal concept can be mapped to a 3-tuple such as $(\varpi_{v_t}, \omega_{v_t}, \varsigma_{v_t})$.

4.2 Application to the Running Example

To illustrate our method, let us examine the set of interrelated concepts navigated to extract $\mathcal{G}_{\texttt{CKVT\_5}}$ cpo-pattern associated to CKVT_5 main concept from $\mathcal{L}_{\texttt{KVT}}$ (Fig. 3(a)). More precisely, we propose to investigate the navigated concept extents. To this end, Fig. 5 illustrates $\mathcal{G}_{\texttt{CKVT\_5}}$ cpo-pattern, whose vertices are annotated with 3-tuples $(\varpi_{v_t}, \varphi_{v_t}, \varsigma_{v_t})$, and the navigated concept extents.

The vertex labelled with ($\texttt{Influenza}_\texttt{A}$) target 1-itemset, is derived from CKVT_5 intent. CKVT_5 extent comprises the sequences in $\mathcal{D}_{\mathcal{S}fluA}$ (Tab. 1) containing all the paths in $\mathcal{G}_{\texttt{CKVT\_5}}$, i.e. $\mathcal{S}_{\mathcal{G}_{\texttt{CKVT\_5}}} = \{S1, S2, S4, S5\}$. There are 4 distinct ($\texttt{Influenza}_\texttt{A}$) target 1-itemsets in $\mathcal{D}_{\mathcal{S}fluA}$ that are preceded by the itemsets ($\texttt{FEVER}_\texttt{high}$), ($\texttt{COUGH}_\texttt{high}$) and ($\texttt{FEVER}_\texttt{moderate}$) in the order they appear in $\mathcal{G}_{\texttt{CKVT\_5}}$. The vertex labelled with ($\texttt{FEVER}_\texttt{high}$) preceded itemset is derived from CKME_6 temporal concept intent and it is denoted by $v_{\texttt{CKME\_6}}$. The CKME_6 extent gathers the 5 itemsets (each UID represents an itemset) in $\mathcal{D}_{\mathcal{S}fluA}$ that contain $\texttt{FEVER}_\texttt{high}$ item and that are preceded by the ($\texttt{FEVER}_\texttt{moderate}$) itemset. Therefore, the overall weight of $v_{\texttt{CKME\_6}}$ is $\omega_{v_{\texttt{CKME\_6}}} = 5$. Since all the itemsets in the CKME_6 extent are owned by the sequences in $\mathcal{S}_{\mathcal{G}_{\texttt{CKVT\_5}}}$, the $v_{\texttt{CKME\_6}}$ specificity is $\varsigma_{v_{\texttt{CKME\_6}}} = \frac{5}{5}100 = 100\%$. In addition, we observe that CKME_6 extent contains the group of itemsets {IS2_Seq5, IS3_Seq5} that occur in the same sequence $getSeq(\texttt{IS2\_Seq5}) = getSeq(\texttt{IS3\_Seq5}) = \texttt{Seq5}$ s.t. $getS(\texttt{Seq5}) \in \mathcal{S}_{\mathcal{G}_{\texttt{CKVT\_5}}}$. Then, CKME_6 extent contains only one repetition identified by IS3_Seq5, and thus $v_{\texttt{CKME\_6}}$ persistency is $\varpi_{v_{\texttt{CKME\_6}}} = 0.25$. Similarly, the vertex $v_{\texttt{CKME\_7}}$ labelled with ($\texttt{COUGH}_\texttt{high}$) preceded itemset is derived from CKME_7 temporal concept intent and has the overall weight $\omega_{v_{\texttt{CKME\_7}}} = 6$ and the specificity $\varsigma_{v_{\texttt{CKME\_7}}} = \frac{6}{6}100 = 100\%$. Since there are two groups of two itemsets that occur in Seq1 and Seq4, respectively, the persistency of $v_{\texttt{CKME\_7}}$ is $\varpi_{v_{\texttt{CKME\_7}}} = 0.5$. The vertex $v_{\texttt{CKME\_9}}$ labelled with ($\texttt{FEVER}_\texttt{moderate}$) preceded itemset is derived from CKME_9 temporal concept intent. CKME_9 extent comprises the 8 itemsets in $\mathcal{D}_{\mathcal{S}fluA}$ (Tab. 1) that contain $\texttt{FEVER}_\texttt{moderate}$ item, i.e. the overall weight of $v_{\texttt{CKME\_9}}$ is $\omega_{v_{\texttt{CKME\_9}}} = 8$. The itemset IS1_Seq3 (gray colored in Fig. 5) is owned by $getS(\texttt{Seq3}) \notin \mathcal{S}_{\mathcal{G}_{\texttt{CKVT\_5}}}$ and thus, the $v_{\texttt{CKME\_9}}$ specificity is $\varsigma_{v_{\texttt{CKME\_9}}} = \frac{7}{8}100 = 87.5\%$. Since the CKME_9 extent contains three repetitions identified by IS2_Seq1, IS4_Seq1, and IS2_Seq4, the $v_{\texttt{CKME\_9}}$ persistency is $\varpi_{v_{\texttt{CKME\_9}}} = 0.75$.

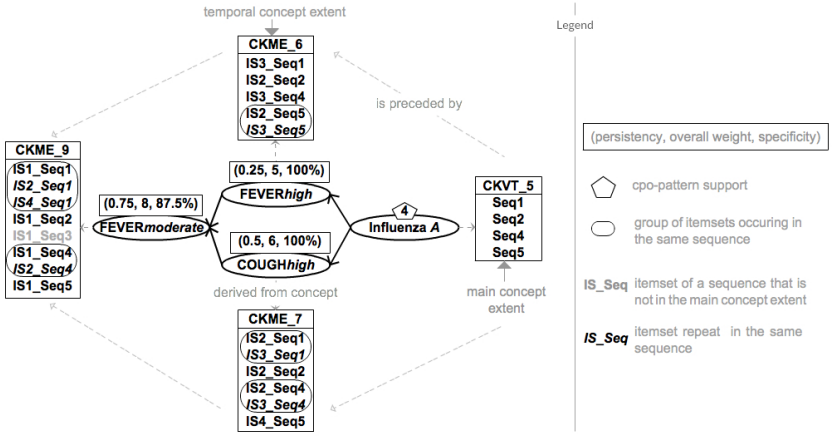


Fig. 5: Extraction of the wcpo-pattern associated to `CKVT_5` concept (Fig. 3(a))

5 Enhancing Sequential Data Analysis Using WCPO-patterns

Using RCA as explained in [10], hierarchies of cpo-patterns are obtained in order to improve the interpretation step by highlighting how the extracted cpo-patterns relate to each other. However, there are practical cases (three such cases are discussed in Section 5.1, 5.2 and 5.3) when the order between the extracted cpo-patterns is insufficient for the expert. To improve these practical cases, we propose to use hierarchies of wcpo-patterns and to exploit the vertex measures of weightiness introduced in Section 4.1.

Henceforth, we use our running example to illustrate three practical cases that take advantage of the proposed wcpo-patterns when a physician tries to interpret the extracted medical knowledge. As these examples demonstrate, for the sake of our approach illustration, the wcpo-patterns can lead to more informative medical knowledge since the different importance of vertices or paths are considered. It is worth mentioning that the persistency, overall weight and the specificity of a vertex can be considered simultaneously or not depending on the motivation behind the analysis step.

5.1 Practical Case: Ranking CPO-pattern Vertices and Paths

In a cpo-pattern, its vertices/paths are considered uniformly. The expert can easily be misled by this assumption into thinking that all vertices/paths in a cpo-pattern have the same impact on the object of interest. To illustrate that, let us suppose that a physician tries to interpret the cpo-pattern given in Fig. 5 by disregarding the weightiness of vertices. The physician finds that often before outbreaks of influenza A the patients feel high cough and high fever in any order, but after feeling moderate fever. Since the medical knowledge (cpo-pattern) was

mined with very high support (4 out of 5 analysed medical sequences), the physician can infer *with high confidence* that "moderate fever should always be considered as an early sign of a possible influenza A outbreak" and that "high fever and high cough should always be the first signs of influenza A outbreak".

However, let us assume that the physician analyses again the cpo-pattern given in Fig. 5 by paying attention to the weightiness of vertices. High fever and high cough symptoms, which are felt by patients after moderate fever, are $\varsigma_{v_{\texttt{CKME\_6}}} = \varsigma_{v_{\texttt{CKME\_7}}} = 100\%$ specific only to the four medical sequences. In contrast, moderate fever symptom is $\varsigma_{v_{\texttt{CKME\_9}}} = 87.5\%$ specific to the four medical sequences and besides, to other analysed medical sequences. Consequently, moderate fever felt by patients before influenza A outbreaks can be a global available tendency in the dataset and thus the physician can infer *with higher confidence* that "moderate fever should always be an early sign of a possible influenza A outbreak". Since the high fever and high cough felt by patients after moderate fever is a tendency available only in a subset of the analysed dataset, the physician can conclude *with less confidence* that "high fever and high cough should always be the first signs of influenza A outbreak".

In addition, the physician can deduce that high cough is more persistent than high fever in the four medical sequences, i.e. $\varpi_{v_{\texttt{CKME\_7}}} > \varpi_{v_{\texttt{CKME\_6}}}$. Therefore, the high cough is more likely to be the first sign of influenza A outbreak, while high fever, for example, can be caused by a bacterial infection. Similarly, this assumption holds if the overall weights are analysed. Relying on the persistency of high cough, the physician can rank the paths, i.e. $\texttt{FEVER}_{\texttt{moderate}} \leftarrow \texttt{COUGH}_{\texttt{high}} \leftarrow \texttt{Influenza}_{\texttt{A}}$ path is more pertinent to recognize influenza A outbreak.

5.2 Practical Case: Selecting Interesting Navigation Paths in CPO-pattern Hierarchies

Usually the extracted hierarchies of cpo-patterns are very large, and even if the relationships between cpo-patterns are highlighted, and the support measure can be considered, their navigation is still not an easy task for the expert. For instance, let us suppose that a physician tries to navigate the hierarchy of cpo-patterns shown in Fig. 6 while ignoring the weightiness of vertices. This figure depicts an excerpt (with five cpo-patterns from (a) to (e)) from the hierarchy of wcpo-patterns obtained adding new medical sequences to the RCF given in Tab. 3. The physician begins the navigation from the simple cpo-patterns (having only one vertex) (a) and (b). Thus, the physician has an overview of the common tendencies of the analysed $\mathcal{D}_{\mathcal{S}fluA}$ and minimises the chance of overlooking interesting cpo-patterns. It is noted that both cpo-patterns were mined with the same support, and apparently they mark out two interesting navigation paths in the hierarchy. Nevertheless, when the physician considers the persistency of vertices, their different importance are highlighted. The physician can easily infer that high cough is more probable to be a sign of influenza A outbreak, i.e. high cough is more persistent ($\varpi = 2$) than high fever ($\varpi = 1$). Accordingly, the physician selects the navigation path that consists in the descendant

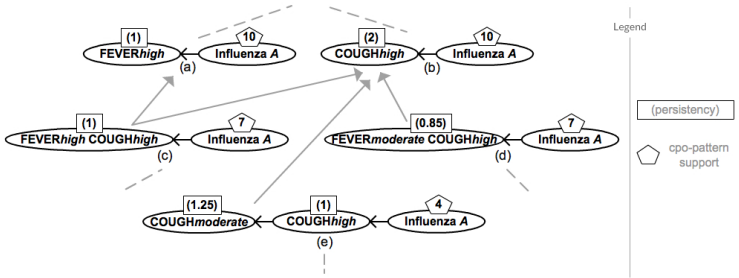


Fig. 6: Excerpt from the hierarchy of wcpo-patterns obtained by adding new medical examinations and viral tests to the RCF given in Tab. 3

wcpo-patterns of (b) and the analysis continues by applying the same ranking criterion.

5.3 Practical Case: Distinguishing the Best Represented Sub-Dataset by a CPO-Pattern

There are cases when it is useful to find out discriminant tendencies for different types of the studied object of interest. Here, in our running example, the physician is interested in distinguishing between outbreaks of influenza A and B by assessing the symptoms felt by patients. Usually, the physician determines that the same extracted cpo-pattern belongs rather to $\mathcal{D}_{\mathcal{S}fluA}$ or $\mathcal{D}_{\mathcal{S}fluB}$ (given in Tab. 1) by relying on support measure. However, there are cases when a cpo-pattern is found with equal support in both datasets.

For example, let us consider that the physician tries to understand if the cpo-pattern given in Fig. 7 represents a discriminant tendency for influenza A or influenza B outbreak. The cpo-patterns given in Fig. 7(a) and Fig. 7(b) were extracted from $\mathcal{D}_{\mathcal{S}fluA}$ and from $\mathcal{D}_{\mathcal{S}fluB}$, respectively. Both cpo-patterns were mined with the same support and thus it is impossible to distinguish between them by disregarding the weightiness of vertices. In contrast, when the physician considers, for instance, the persistencies of vertices, it is easily noted that high cough and moderate fever are more persistent in the subset of $\mathcal{D}_{\mathcal{S}fluA}$, while high fever has the same persistence in both subsets (one of $\mathcal{D}_{\mathcal{S}fluA}$ and one of $\mathcal{D}_{\mathcal{S}fluB}$). Accordingly, the physician can conclude that the cpo-pattern is a distinguishing characteristic of influenza A outbreak since two out of three vertices are more significant in Fig. 7(a). Moreover, this inference is drawn with more confidence by additionally considering the overall weights of the vertices, i.e. mainly, high cough and moderate fever are more numerous in $\mathcal{D}_{\mathcal{S}fluA}$ than in $\mathcal{D}_{\mathcal{S}fluB}$.

6 Related Work

Traditional pattern mining algorithms in sequence databases, such as those surveyed in [9] for sequential pattern mining and closed sequential pattern mining,

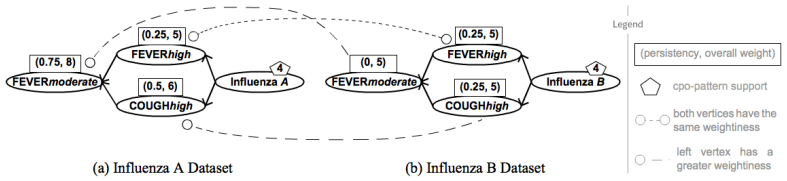


Fig. 7: Distinguishing between outbreaks of influenza A and B

or FreCPO [12] and OrderSpan [5] for cpo-pattern mining, consider only the order on itemsets in concerned sequences and treat all the itemsets uniformly. To capture more particularities hidden in the analysed data, Srikant et al. [14] propose to extract more informative sequential patterns by adding time constraints in advance, and thus a pattern is extracted only if it admits a max-gap and a min-gap between adjacent itemsets. Pei et al. [11] push various constraints, e.g. time-interval and gap information between items, into the mining process to limit the mining results. Chen et al. [4] propose to extract time-interval sequential patterns that reveal the time interval between successive items and besides these time-intervals are explicitly shown in the patterns. To capture the time interval between all pairs of items in the extracted patterns, Hu et al. [7] introduce the multi-time-interval sequential pattern. Chang et al. [3] propose to find weighted sequential patterns by pushing a time-interval weight measure (the weight of a sequence derived from the time-interval of the sequence itemsets) into the mining process. Besides, in [8,16] more informative sequential patterns are obtained pushing pre-assigned quantitative information, which are recorded in the analysed database, into the mining process. In contrast, our RCA-based approach focuses more on enhancing the interpretation step by extracting hierarchies of wcpo-patterns. We capture for each vertex (preceded itemset) in a cpo-pattern its weightiness (e.g. specificity), in the analysed sequences and we show them explicitly in the extracted wcpo-pattern. Consequently, the expert is guided by (i) the relationships between wcpo-patterns that are revealed by the obtained hierarchies, (ii) the weightiness of vertices and (iii) the more informative order (partial order) of itemsets.

7 Conclusion

This work presents an approach for enhancing sequential data analysis within the framework of RCA. To this end, we propose to extract more informative patterns, namely weighted cpo-patterns, that capture and explicitly show not only the order on itemsets (as do traditional cpo-patterns) but also their different influence on the analysed sequence database through measures such as persistency, specificity and overall weight. Moreover, thanks to the hierarchical RCA results, we directly obtain the relationships between these wcpo-patterns that guide the interpretation step and help in better understanding the extracted knowledge.

In this paper, we have formally defined our approach and we have illustrated it on a toy example.

In the future, we plan to study the properties of the proposed measures, and to make a comparison with existing measures of interest. In addition, a possible extension of our work is to consider time-intervals between itemsets, or to add quantitative information recorded in the analysed sequence database to obtain more valuable knowledge.

References

1. Agrawal, R., Srikant, R.: Mining sequential patterns. In: Int. Conference on Data Engineering. pp. 3–14 (1995)
2. Casas-Garriga, G.: Summarizing sequential data with closed partial orders. In: 2005 SIAM Int. Conference on Data Mining. pp. 380–391 (2005)
3. Chang, J.H.: Mining weighted sequential patterns in a sequence database with a time-interval weight. Know.-Based Syst. 24(1), 1 – 9 (2011)
4. Chen, Y.L., Chiang, M.C., Ko, M.T.: Discovering time-interval sequential patterns in sequence databases. Expert Systems with Applications 25(3), 343 – 354 (2003)
5. Fabrègue, M., Braud, A., Bringay, S., Le Ber, F., Teisseire, M.: Mining closed partially ordered patterns, a new optimized algorithm. Know.-Based Syst. 79, 68–79 (2015)
6. Ganter, B., Wille, R.: Formal Concept Analysis: Mathematical Foundations. Springer (1999)
7. Hu, Y.H., Huang, T.C.K., Yang, H.R., Chen, Y.L.: On mining multi-time-interval sequential patterns. Data & Knowledge Engineering 68(10), 1112 – 1127 (2009)
8. Kim, C., Lim, J.H., Ng, R.T., Shim, K.: Squire: Sequential pattern mining with quantities. Journal of Systems and Software 80(10), 1726 – 1745 (2007)
9. Mabroukeh, N.R., Ezeife, C.I.: A taxonomy of sequential pattern mining algorithms. ACM Comput. Surv. 43(1), 3:1–3:41 (2010)
10. Nica, C., Braud, A., Dolques, X., Huchard, M., Le Ber, F.: Extracting hierarchies of closed partially-ordered patterns using relational concept analysis. In: Proceedings of the 22nd International Conference on Conceptual Structures, ICCS 2016. pp. 17–30. Springer (2016)
11. Pei, J., Han, J., Wang, W.: Mining sequential patterns with constraints in large databases. In: Proceedings of the 11th International Conference on Information and Knowledge Management. pp. 18–25. CIKM '02, ACM (2002)
12. Pei, J., Wang, H., Liu, J., Wang, K., Wang, J., Yu, P.S.: Discovering frequent closed partial orders from strings. IEEE Transactions on Knowledge and Data Engineering 18(11), 1467–1481 (2006)
13. Rouane-Hacene, M., Huchard, M., Napoli, A., Valtchev, P.: Relational concept analysis: Mining concept lattices from multi-relational data. Annals of Mathematics and Artificial Intelligence 67(1), 81–108 (2013)
14. Srikant, R., Agrawal, R.: Mining sequential patterns: Generalizations and performance improvements. In: Proceedings of the 5th International Conference on Extending Database Technology. pp. 3–17. EDBT '96, Springer-Verlag (1996)
15. Yan, X., Han, J., Afshar, R.: Clospan: Mining closed sequential patterns in large datasets. In: In SDM. pp. 166–177 (2003)
16. Yun, U.: A new framework for detecting weighted sequential patterns in large sequence databases. Know.-Based Syst. 21(2), 110–122 (Mar 2008)