Healthcare trajectory mining by combining multidimensional component and itemsets

Elias Egho, Chedy Raïssi, Dino Ienco, Nicolas Jay, Amedeo Napoli, Pascal Poncelet, Catherine Quantin, Maguelonne Teisseire

Healthcare Trajectory Mining by Combining Multidimensional Component and Itemsets

Elias Egho[1], Chedy Raïssi[4], Dino Ienco[2,3], Nicolas Jay[1], Amedeo Napoli[1], Pascal Poncelet[2,3], Catherine Quantin[5] and Maguelonne Teisseire[2,3]

[1] Orpailleur Team, LORIA, Vandoeuvre-les-Nancy, France
{firstname.lastname}@loria.fr
[2] Irstea, UMR TETIS, F-34093 Montpellier, France
{firstname.lastname}@teledetection.fr
[3] LIRMM, Univ. Montpellier 2, Montpellier, France
{firstname.lastname}@lirmm.fr
[4] INRIA, Nancy Grand Est, France
{firstname.lastname}@inria.fr
[5] Department of Biostatistics and Medical Information
CHU of Dijon, Dijon, France

Abstract. Sequential pattern mining is aimed at extracting correlations among temporal data. Many different methods were proposed to either enumerate sequences of set valued data (i.e., itemsets) or sequences containing multidimensional items. However, in real-world scenarios, data sequences are described as events of both multidimensional items and set valued information. These rich heterogeneous descriptions cannot be exploited by traditional approaches. For example, in healthcare domain, hospitalizations are defined as sequences of multi-dimensional attributes (e.g. Hospital or Diagnosis) associated with two sets, set of medical procedures (e.g. { Radiography, Appendectomy }) and set of medical drugs (e.g. { Aspirin, Paracetamol }) . In this paper we propose a new approach called MMISP (*Mining Multidimensional Itemset Sequential Patterns*) to extract patterns from a complex sequences including both dimensional items and itemsets. The novelties of the proposal lies in: (i) the way in which the data can be efficiently compressed; (ii) the ability to reuse and adopt sequential pattern mining algorithms and (iii) the extraction of new kind of patterns. We introduce as a case-study, experimented on real data aggregated from a regional healthcare system and we point out the usefulness of the extracted patterns. Additional experiments on synthetic data highlights the efficiency and scalability of our approach.

Keywords: Sequential Patterns, Multi-dimensional Sequential Patterns, Data Mining

1 Introduction

Data warehouses are constituting a large source of data that can be used to extract information for expert analysis and decision makers [5]. In temporal data

warehouses, every bit of information is associated with a timeline describing a total order over events. This total ordering introduces complexity in the extraction process. Many efficient approaches were developed to mine these patterns (i.e., sequential patterns) like PrefixSpan [9], SPADE [17], ClosSpan [14],...etc. However, all these techniques and algorithms, without any exception, focus solely on sequences of set valued data (i.e., *itemsets*) and do not pay attention to real-world data that is described over multiple dimensions. To overcome this problem, Pinto et al. [10] introduced the notion of multi-dimensionality in sequences and proposed an efficient algorithm. Later works, like Zhang et al. [18] or Yu et al. [16] extended the initial Pinto's approach for different scenarios and use-cases. While in set valued approaches the events are represented by itemsets, in multi-dimensional temporal databases the events are defined over a fixed schema where all attributes appear in the extracted patterns. Furthermore, and this is particularly true in the data warehouse environment, background knowledge is usually available and can be represented as a hierarchy over the values of the attributes. Taking advantage of this observation, Plantevit et al. introduced M^3SP [11], an efficient algorithm that is able to incorporate different dimensions and their taxonomies in the sequential pattern mining process. The benefit of this approach is to extract patterns with the most appropriate level of granularity. Still, this ideal representation of data is uncommon in real-world applications where heterogeneity is usually elevated to a foundational concept. In this study, we focus on extracting knowledge from medical data warehouse representing information about patients in different hospitals. The successive hospitalizations of a patient can be expressed as a sequence of multidimensional attributes associated with a set of medical procedures and a set of medical drugs. Our goal is to be able to extract patterns that express patients stays along with combinations of procedures over time. This type of pattern is very useful to healthcare professionals to better understand the global behavior of patients over time. Unfortunately this kind of complex data cannot be mined by any traditional sequential pattern approach. In this paper, we propose a new method to extract patterns from sequences which include multidimensional items and itemsets at the same time. In addition, the proposed approach incorporates background knowledge in the form of hierarchies over attributes.

The remainder of this paper is organized as follows, Section 2 describes the existing work in the classical and multidimensional sequential patterns. Section 3 introduces the problem statement as well as a running example. The method for extracting multidimensional itemset frequent patterns is described in Section 4. Section 5 presents experimental results from both quantitative and qualitative point of views and Section 6 concludes the paper.

2 Related Work

Let $\mathcal{I}$ be a finite set of *items*. An *itemset* X is a non-empty subset of $\mathcal{I}$. A *sequence* S over $\mathcal{I}$ is an ordered list $\langle X_1 \cdots X_n \rangle$, where X_i $(1 \leq i \leq n,\, n \in \mathbb{N})$ is an itemset. A sequence $T = \langle Y_1 \cdots Y_m \rangle$ is a **subsequence** of $S = \langle X_1 \ldots X_n \rangle$,

denoted by $T \preceq S$, if there exist indices $1 \leq i_1 < i_2 < \cdots < i_m \leq n$ such that $Y_j \subseteq X_{i_j}$ for all $j = 1 \ldots m$ and $m \leq n$. S is said to be a **supersequence** of T. Let $\mathtt{S_{DB}} = \{\mathtt{S_1}, \mathtt{S_2} \ldots \mathtt{S_n}\}$ be a database of sequences. The support of a sequence `s` in `D` is the proportion of sequences of `D` containing `s`. Given a `minsup` threshold, the problem of frequent sequential pattern mining consists in finding the set `FS` of sequences whose support is not less than `minsup`. Following the first work of Agrawal and Srikant [1] and the Apriori algorithm, many studies have contributed to the efficient mining of sequential patterns. The main algorithms are PrefixSpan [9], SPADE [17], SPAM [3], PSP [8], DISC [4], PAID [15], FAST [12]. All of these algorithms aim to discover sequential patterns from a set of sequences of itemsets.

Usually, the information in a sequence is based on several dimensions. Pinto et al [10] propose the first work by including for mining multidimensional sequential patterns, by including dimensions in the first or the last itemset of the sequence. But this works only for dimensions that remain constant over time, such as gender of the patient. Among other proposals addressed in this area, Yu et al [16] consider multidimensional sequential pattern mining in the web domain. Here, dimensions are pages, sessions and days. They present two algorithms: AprioriMD and PrefixMDSpan.

in real world applications, each dimension can be represented at different levels of granularity, by using a taxonomy. The interest lies in the capacity of extracting more or less general/specific sequential patterns and overcome problems of excessive granularity and low support. Although Srikant and Agrawal [13] combined the use of hierarchy of values in the extraction of association rules and sequential patterns, their approach is not scalable in a multidimensional context. Han et al [7] proposed a method for mining multiple level association rules in large databases. But their approach could not extract patterns containing items from different levels in the taxonomy. Appice et al [2] proposed SPADA, an algorithm for discovering multi-level spatial association rules. Plantevit et al [11] proposed M^3SP, an algorithm taking both multilevel and multidimensional aspects into account. M^3SP is able to find sequential patterns with the most appropriate level of granularity. Egho et al [6] proposed an extension for M^3SP for extracting both general and specific sequences, they iteratively applied M^3SP, decreasing threshold by one objects at each step. Their proposition allows the extraction of more interesting sequences than using a single `minsup` threshold.

3 Problem Statement

In this section we list some preliminary definitions needed to formalize the problem. First of all, we introduce a motivating example from a real data set related to the PMSI (Program of medical information systems). This French nationwide information system describes hospital activities from both economical and medical points of view. In this system, each hospitalization is related to the recording of administrative, demographical and medical data. Let S_{DB} be a database of multidimensional itemsets data sequences. Figure 1 illustrates such a database.

Patients	Trajectories
P_1	$\langle(UH_{Paris}, C_1, \{p_1, p_2\}, \{drug_1, drug_2\}), (UH_{Paris}, C_1, \{p_1\}, \{drug_2\}), (GH_{Lyon}, R_1, \{p_2\}, \{drug_2\})\rangle$
P_2	$\langle(UH_{Paris}, C_1, \{p_1\}, \{drug_2\}), (UH_{Paris}, C_1, \{p_1, p_2\}, \{drug_1, drug_2\}), (GH_{Lyon}, R_1, \{p_2\}, \{drug_2\})\rangle$
P_3	$\langle(UH_{Paris}, C_1, \{p_1, p_2\}, \{drug_1, drug_2, drug_3\}), (GH_{Lyon}, R_1, \{p_2\}, \{drug_2, drug_4\})\rangle$
P_4	$\langle(UH_{Paris}, C_1, \{p_2\}, \{drug_1, drug_2\}), (UH_{Paris}, R_2, \{p_3\}, \{drug_2\}), (GH_{Lyon}, R_2, \{p_2\}, \{drug_3\})\rangle$

Fig. 1. An example of a database of patient trajectories

Definition 1. *(Dimensions and specialization down(d)) A dimension $(D, \leqslant)$ is a partially ordered set where D is the set of all items of dimension. For a given $d \in D$, down(d) (resp. up(d)) denotes the set of all specializations $\{x \in D | x \leqslant d\}$ (resp. generalizations $\{x \in D | d \leqslant x\}$) of d.*

Example 1. Figure 2 shows two dimensions (hospital and diagnosis). For hospital dimension, $D_{hospital} = \{T_{hospital}, UH, GH, UH_{Paris}, UH_{Nancy}, GH_{Paris}, GH_{Lyon}\}$ and $UH_{Paris} \in down(UH)$ as UH_{Paris} is a direct descendant of UH.

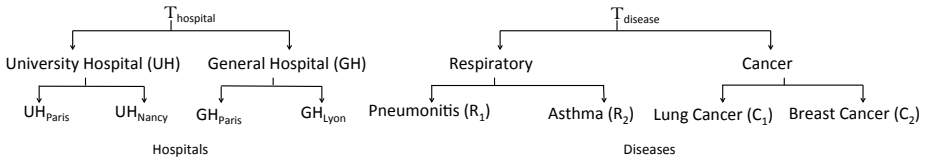


Fig. 2. Hospital and diagnoses taxonomies

By taking into account the multidimensional items and the sets of items, we define an event as follows.

Definition 2. *(Event) An event $e = (d_1, ..., d_n, itemset_{n+1}, ..., itemset_{n+m})$ is a vector of n multidimensional items and m sets of items where $d_i \in D_i, i = 1, \cdots, n$. Given two events $e = (d_1, ..., d_n, itemset_{n+1}, ..., itemset_{n+m})$ and $e' = (d'_1, ..., d'_n, itemset'_{n+1}, ..., itemset'_{n+m})$, e is more general than e', denoted by $e' \leq_e e$, if and only if:*

- $\forall\ i\ ;\ 1 \leqslant i \leqslant n\ ;\ d'_i \in down(d_i)$.
- $\forall\ j\ ;\ 1 \leqslant j \leqslant m\ ;\ itemset_{n+j} \subseteq itemset'_{n+j}$.

Example 2. $e' = (UH_{Paris}, C_1, \{p_1, p_2, p_3\}, \{drug_2, drug_3, drug_4\})$ is an event, where:

- UH_{Paris}, C_1 are two multidimensional items representing the two dimensions (hospital and diagnosis).
- $\{p_1, p_2, p_3\}, \{drug_2, drug_3, drug_4\}$ are two sets of items representing the medical procedures and the medical drugs.

The event $e = (UH, T_{disease}, \{p_1, p_2\}, \{drug_2, drug_3\})$ is more general than e', $e' \leq_e e$, because of:

– $UH_{Paris} \in down(UH)$ *and* $C_1 \in down(T_{disease})$.
– $\{p_1, p_2\} \subseteq \{p_1, p_2, p_3\}$ *and* $\{drug_3, drug_4\} \subseteq \{drug_2, drug_3, drug_4\}$.

A multidimensional itemsets data sequence is composed of events.

Definition 3. *(Multidimensional Itemsets Sequence) A multidimensional itemsets sequence* $s =< e_1, e_2, ..., e_l >$ *is an ordered list of events* e_i. *Given two multidimensional itemsets sequences* $s =< e_1, e_2, ..., e_l >$ *and* $s' =< e'_1, e'_2, ..., e'_{l'} >$, s *is more general than* s', *denoted by* $s \leq_s s'$, *if there exist indices* $1 \leq i_1 < i_2 < ... < i_l \leq l'$ *such that* $e_j \leq_e e'_{i_j}$ *for all* $j = 1 \ldots l$ *and* $l \leqslant l'$.

Example 3. The multidimensional itemsets sequence $s = \langle (UH_{Paris}, C_1, \{p_1, p_2\}, \{drug_1, drug_2, drug_3\}), (GH_{Lyon}, R_1, \{p_2\}, \{drug_2, drug_4\})\rangle$ is a sequence of two events. It expresses the fact that a patient was admitted to the University Hospital of Paris UH_{Paris} for a lung cancer C_1, underwent procedures p_1 and p_2 and was treated with $\{drug_1, drug_2, drug_3\}$, then he went to the General Hospital of Lyon GH_{Lyon} for pneumonitis R_1 where he underwent procedure p_2 and received $\{drug_2, drug_4\}$.

The sequence $s' = \langle (UH_{Paris}, Cancer, \{p_1\}, \{drug_1, drug_2\})\rangle$ is more general than s, $s \leq_s s'$, because $(UH_{Paris}, C_1, \{p_1, p_2\}, \{drug_1, drug_2, drug_3\}) \leq_e (UH_{Paris}, Cancer, \{p_1\}, \{drug_1, drug_2\})$.

Definition 4. *(Patient Trajectory) A patient trajectory is defined as* a multidimensional itemsets sequence.

Example 4. In Table 1, the multidimensional itemsets sequence $s = \langle (UH_{Paris}, C_1, \{p_1, p_2\}, \{drug_1, drug_2\}), (UH_{Paris}, C_1, \{p_1\}, \{drug_2\}), (GH_{Lyon}, R_1, \{p_2\}, \{drug_2\})\rangle$ represents the trajectory for the patient P_1.

Let $\texttt{supp(s)}$ be the number of sequences that includes s in S_{DB}. Furthermore σ be a minimum support threshold specified by the end-user.

Definition 5. *(Most Specific Frequent Multidimensional Itemsets Sequence) Let* s *be multidimensional itemsets sequence, we say that* s *is the most specific frequent multidimensional itemsets sequence in* S_{DB}, *if and only if:* $supp(s) \geq \sigma$ *and* $\nexists s' \in S_{DB}$, *where* $supp(s) = supp(s')$ *and* $s \leq_s s'$.

The problem of mining multidimensional itemsets sequences is to extract the set of all most specific frequent multidimensional itemsets sequence in S_{DB} such as $supp(s) \geq \sigma$. By using the dimensions we can extract general or specific patterns and overcome problems of excessive granularities and low supports.

Example 5. Let $\sigma = 0.75$ (i.e. a sequence is frequent if it appears at least three times in S_{DB}). The sequence $s_1 = \langle (UH_{Paris}, C_1, \{p_1, p_2\}, \{drug_1, drug_2\}), (GH_{Lyon}, R_1, \{p_2\}, \{drug_2\})\rangle$ is frequent. $s_2 = \langle (UH, Cancer, \{p_1, p_2\}$, $\{drug_1, drug_2\}), (GH, Respiratory, \{p_2\}, \{drug_2\})\rangle$ is also frequent. Nevertheless, s_2 is not kept since it is too general compared to s_1.

4 Mining multidimensional itemsets sequential patterns

In this section, we present the MMISP (*Mining Multidimensional Itemsets Sequential Patterns*) algorithm for extracting multidimensional itemsets sequential patterns with different levels of granularity over each dimension. MMISP follows a bottom-up approach by first focusing on extracting frequent multidimensional items that can exist at different level of granularity, then it considers the itemsets part of the events and compute the support of every item is S_{DB} for each itemset. After these two steps, frequent multidimensional items and frequent itemsets are combined to generate events. In the final step, the frequent events are mapped to a new representation and a standard sequential mining algorithm is applied to enumerate multidimensional itemsets sequential patterns.

In the next subections, we provide the details of each step of our work and discuss the different challenges.

4.1 Generating frequent multidimensional items

MMISP starts by processing the n multidimensional items of the events in the sequences. Basically it considers three types of dimensions: a temporal dimension D_t, a set of analysis dimension D_A and a set of reference dimension D_R. MMISP splits S_{DB} into blocks according to dimension D_R. Then, MMISP sorts each block according to the temporal dimension D_t. This is a classic way of partitioning the database and was introduced in [11]. The tuples of n multidimensional items appearing in an event are defined w.r.t. analysis dimensions D_A. The support of n multidimensional items is computed according to dimension of D_R. It is the ratio of the number of blocks supporting the n multidimensional items over the total number of blocks.

Date	Hospital	Diagnosis
1	UH_{Paris}	C_1
2	UH_{Paris}	C_1
3	GH_{Lyon}	R_1

Block: $Patient_1$

Date	Hospital	Diagnosis
1	UH_{Paris}	C_1
2	GH_{Lyon}	R_1

Block: $Patient_3$

Date	Hospital	Diagnosis
1	UH_{Paris}	C_1
2	UH_{Paris}	C_1
3	GH_{Lyon}	R_1

Block: $Patient_2$

Date	Hospital	Diagnosis
1	UH_{Paris}	C_1
2	UH_{Paris}	R_2
3	GH_{Lyon}	R_2

Block: $Patient_4$

Fig. 3. Block partition of the database according to D_R={Patient}

Example 6. In the running example, H (hospitals) and D (diseases) are the analysis dimensions, $Date$ is the temporal dimension, and P (patients) is the reference dimension. By using P (patients) to split the dataset, we obtain four blocks defined by $Patient_1$, $Patient_2$, $Patient_3$ and $Patient_4$ as shown in Figure 3.

To simplify our works we will represent the n multidimensional items of the event as follows:

Definition 6. *(multidimensional component) Given a dimension $(D, \leqslant)$, a multidimensional component over D, denoted $(mdc, \leqslant_{mdc})$, is a tuple $(d_1, ..., d_n)$ where $d_i \in D$, $i = 1, \cdots, n$. For two given multidimensional components $mdc = (d_1, ..., d_n)$ and $mdc' = (d'_1, ..., d'_n)$, $mdc' \leqslant_{mdc} mdc$ denotes that mdc is more general than mdc', if for every $i = 1, ..., n$, $d'_i \in down(d_i)$.*

Example 7. Let (UH_{Paris}, *Lung Cancer*) and (UH, *Cancer*) be two multidimensional components. (UH_{Paris}, *Lung Cancer*) $\leqslant_{mdc}$ (*UH, Cancer*) because $UH_{Paris} \in down(UH)$ and *Lung Cancer* $\in$down(*Cancer*).

The first steps in MMISP is generation all the frequent multidimensional components. This generation is given by the product of all partially ordered sets of the dimensions. The result of this product is a semilattice which has a top element $(T_1, ..., T_m)$ and each node in this semilattice is a multidimensional component. Extracting only the frequent multidimensional components can be done by choosing `minsup` and building the iceberg semi-lattice. The iceberg semi-lattice is a semi-lattice where its elements have a support greater than `minsup`. Figure 4 shows iceberg semi-lattice generated by the product of the two partially ordered sets (hospital and diagnosis) in Figure 2 with minsup= $\frac{3}{4}$ patients.

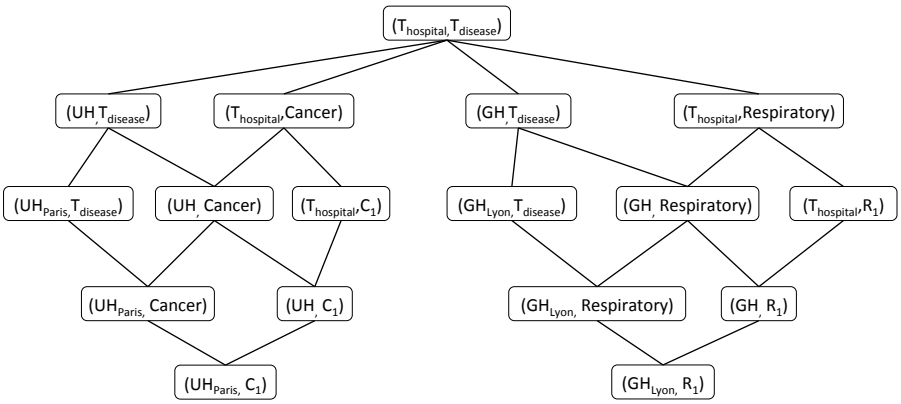


Fig. 4. Iceberg semilattice generated by the product of the two partially ordered sets (hospital and diagnosis) in Figure 2 with minsup= $\frac{3}{4}$ patient

Handling the product of several partial order sets is a cumbersome process. The result of a product is exponential in the number of partial order sets and the cardinality of each set. So, we present a simple and efficient algorithm to generate all frequent multidimensional components.

Following the previous partitioning, algorithm generates all the frequent multidimensional components as follows: firstly, we generate the most general multidimensional component, that is $(T_1, ..., T_n)$. In our running example, we have two

dimensions (hospital and disease), so the most general multidimensional component is $(T_{hospital}, T_{disease})$. Then, the algorithm generates all multidimensional components of the form (T_1, ...T_{i-1}, d_i, T_{i+1}, ..., T_n) where $d_i \in down(T_i)$. We take only the frequent multidimensional component which has support greater than σ. In the running example and for $\sigma = 75\%$ (3 blocks from 4), there are four new frequent multidimensional components: $(UH, T_{disease})$, $(GH, T_{disease})$, $(T_{hospital}, Respiratory)$ and $(T_{hospital}, Cancer)$.

The recursive generation of the new multidimensional components continues by using each previously generated frequent multidimensional component (a). This is done with an indexing method that identifies an integer z which is the position of the last dimension in a and is not top T. For example if a=$(UH, T_{Disease})$, z is equal to one, which is the first dimension (hospital) because the value for the hospital dimension (UH) and the second dimension (disease) has the value $T_{disease}$.

For each dimension d_k in a, where $k \in [z, m]$, we replace d_k with each of its specialization from the set $down(d_k)$. For example, if a=$(UH, T_{Disease})$, we have z=1 and we can generate four new mdc_s: $\{(UH_{Paris}, T_{Disease}), (UH_{Nancy}, T_{Disease})$, $(UH, Respiratory)$, $(UH, Cancer)\}$. The first and the second multidimensional components are generated by replacing UH by $down(UH) = \{UH_{Paris}, UH_{Nancy}\}$, the third and the forth multidimensional components are generated by replacing $T_{Disease}$ by $down(T_{Disease}) = \{Respiratory, Cancer\}$.

At each step, we select only the frequent multidimensional components. For our previously example with $\sigma = 75\%$, $\{(UH_{Paris}, T_{Disease}), (UH, Cancer)\}$ are the new frequent multidimensional components generated by $(UH, T_{Disease})$.

Finally, from all frequent multidimensional components generated, we select only the most specific multidimensional component.

Definition 7. *(Most specific multidimensional component) Let a be multidimensional component, we can say that, a is the most specific multidimensional component, if and only if* $\nexists$ *a′ multidimensional component, where supp*$(a) =$ *supp*(a') *and* $a' \leqslant_{mdc} a$.

Frequent multidimensional component
(UH_{Paris}, C_1)
(GH_{Lyon}, R_1)

Table 1. The most specific frequent multidimensional components

Example 8. Figure 5 illustrates the generation of all frequent multidimensional components on the running example with $\sigma = \frac{3}{4}$. The most specific components are (UH_{Paris}, C_1) and (GH_{Lyon}, R_1).

4.2 Generating Frequent Itemsets

In this step, MMISP focuses on m itemsets part of the events, $(d_1, ..., d_n, \texttt{itemset}_{n+1}, ..., \texttt{itemset}_{n+m})$. We will study separately each itemset in this part. Basically,

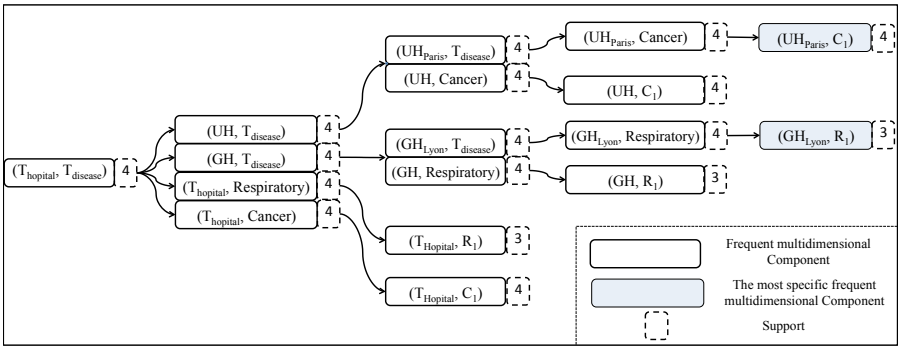


Fig. 5. Frequent multidimensional components generation

this step aims at extracting the set of all items that are frequent in a sequence of length 1. Recall that, in level-wise approaches, either itemset-extension or sequence-extension can be considered. For example, if we have a sequence $s_1 = \langle\{1,2,3\}\rangle$, then $s_2 = \langle\{1,2,3\}\{4\}\rangle$ is an extended sequence of s_1 and $s_3 = \langle\{1,2,3,4\}\rangle$ is an itemset-extended sequence of s_1. In our context we only consider itemset-extension. This task can be easily done by adapting any standard sequential pattern algorithm to extract only the sequence of length 1.

Patients	Sequences of medical procedures
P_1	$\langle\{p_1,p_2\}\{p_1\}\{p_2\}\rangle$
P_2	$\langle\{p_1\}\{p_1,p_2\}\{p_2\}\rangle$
P_3	$\langle\{p_1,p_2\}\{p_2\}\rangle$
P_4	$\langle\{p_2\}\{p_3\}\{p_2\}\rangle$

Sequences of medical procedures

Frequent medical procedures
$\{p_1\}$
$\{p_2\}$
$\{p_1,p_2\}$

Frequent medical procedures candidates

Patients	Sequences of procedures
P_1	$\langle\{drug_1,drug_2\}\{drug_2\}\{drug_2\}\rangle$
P_2	$\langle\{drug_2\}\{drug_1,drug_2\}\{drug_2\}\rangle$
P_3	$\langle\{drug_1,drug_2,drug_3\}\{drug_2,drug_4\}\rangle$
P_4	$\langle\{drug_2,drug_3\}\{drug_2\}\{drug_3\}\rangle$

Sequences of medical drugs

Frequent medical drugs
$\{drug_1\}$
$\{drug_2\}$
$\{drug_1,drug_2\}$

Frequent medical drugs candidates

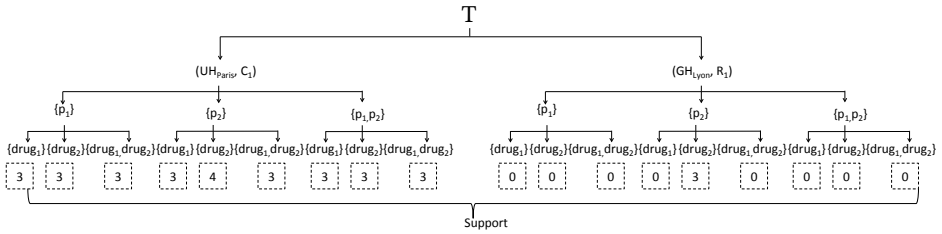
Fig. 6. The frequent itemset generated

Example 9. Figure 6 shows the sequences of medical procedures and medical drugs for patients, and also the frequent medical procedures and medical drugs candidates for $\sigma = \frac{3}{4}$.

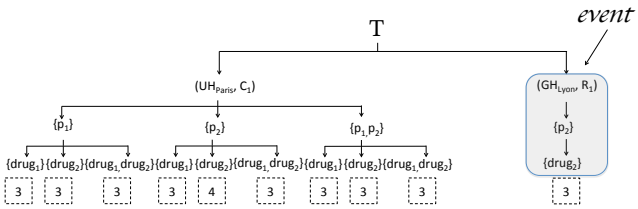
4.3 Generating Frequent Events

Generating frequent events is achieved by combining frequent multidimensional components with frequent itemsets. This task has be done by building a prefix

tree such that the first level in this tree is composed of the frequent multidimensional components and from the second level to leafs, each level is composed the frequent itemset candidates for each itemset part in the vector of itemsets. More precisely, each branch in the tree represents an event. Then a scan is performed over the database to prune irrelevant events from the tree. For example, Figure 7 illustrates the tree before and after pruning infrequent events for $\sigma = \frac{3}{4}$.



Combination tree before pruning



Combination tree after pruning

Fig. 7. An example of the tree for generating frequent events before and after the pruning

4.4 Extracting frequent multidimensional itemsets pattern

Frequent sequences can then be mined by using any standard sequential pattern mining algorithm. As these algorithms require that the dataset to be mined is composed of pairs in the form (id, seq), where id is a sequence identifier and seq is a sequence of itemsets, we transform the initial dataset as follows:

- Each branch in the prefix tree after pruning is assigned a unique id which will be used during the mining operation. This is illustrated in Table 2 .
- Each block (patient) is assigned a unique id of the form P_i.
- Every block b is transformed into a pair $(P_i, \mathbb{S}(p_i))$, where $\mathbb{S}(P_i)$ is built according to the date and the content of the blocks. The final result is reported in Table 3.

A standard sequence mining algorithm can be applied on the transformed database.

Then, the extraction of frequent sequences can be carried out. With $\sigma = 0.75$, the pattern $\langle\{e_9\}\{e_{10}\}\rangle$ is frequent. This sequence corresponds to $\langle(UH_{Paris}, C_1$

event-id	Frequent Event
e_1	$(UH_{Paris}$,C_1,$\{p_1\},\{drug_1\})$
e_2	$(UH_{Paris}$,C_1 $\{p_1\},\{drug_2\})$
e_3	$(UH_{Paris}$,C_1,$\{p_1\},\{drug_1, drug_2\})$
e_4	$(UH_{Paris}$,C_1,$\{p_2\},\{drug_1\})$
e_5	$(UH_{Paris}$,C_1,$\{p_2\},\{drug_2\})$
e_6	$(UH_{Paris}$,C_1,$\{p_2\},\{drug_1, drug_2\})$
e_7	$(UH_{Paris}$,C_1, $\{p_1, p_2\},\{drug_1\})$
e_8	$(UH_{Paris}$,C_1,$\{p_1, p_2\},\{drug_2\})$
e_9	$(UH_{Paris}$,C_1, $\{p_1, p_2\},\{drug_1, drug_2\})$
e_{10}	$(GH_{Lyon}$,R_1, $\{p_2\},\{drug_2\})$

Table 2. Identification each branch (Event) in T

id	Sequence data
P_1	$\langle\{e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8, e_9\}\{e_2\}\{e_{10}\}\rangle$
P_2	$\langle\{e_2\}\{e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8, e_9\}\{e_10\}\rangle$
P_3	$\langle\{e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8, e_9\}\{e_{10}\}\rangle$
P_4	$\langle\{e_5\}\rangle$

Table 3. Transformed database

$\{p_1, p_2\},\{drug_1, drug_2\}),(GH_{Lyon}, R_1,\{p_2\},\{drug_2\})\rangle$ by using the identification in Table 2.

5 Experiments

We conduct experiments on both real and synthetic datasets. The algorithm is implemented in Java and the experiments are carried out on a MacBook Pro with a 2.5GHz Intel Core i5, 4GB of RAM Memory running OS X 10.6.8. The extraction of sequential patterns is based on the public implementation of CloSpan algorithm [14]. We use the implementation supplied by the IlliMine[6] toolkit.

In order to assess the effectiveness of our approach, we run several experiments on the PMSI dataset. This database includes the following informations for each stay: patient id and gender, hospital id, principal diagnosis and date of the stay, a set of associated diagnosis and a set of medical procedures. Our dataset contains 486 patients suffering from lung cancer and living in the East of France. The average length of data sequences is 27. The data is encoded using controlled vocabularies. In particular, diagnoses are encoded with the International Classification of Diseases (ICD10)[7]. This classification is used as an input taxonomy for MMISP. The ICD10 can be seen as a tree with two levels. As illustrated in Figure 8, 3-characters codes such as C34 (Lung cancer) have specializations: C340 is cancer of the main bronchus, C341 is cancer of upper lobe etc.

Figure 9 shows an example of care trajectories described over two dimensions (diagnosis, hospital ID) coupled with two sets of medical procedures and associ-

[6] http://illimine.cs.uiuc.edu/
[7] http://apps.who.int/classifications/apps/icd/icd10online/

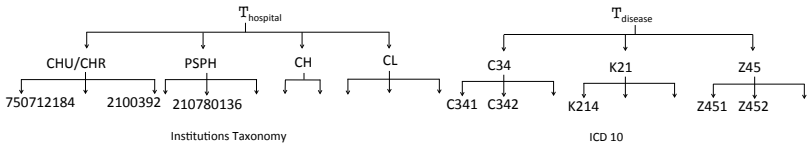


Fig. 8. Examples of taxonomies used in multilevel sequential pattern mining

Patients	Trajectories
P_1	$\langle(C341, 750712184, \{ZBQK002\}, \{D123, K573, C780\}), (Z452, 580780138, \{ZZQK002\}, \{C189\}), \ldots\rangle$
P_2	$\langle(C770, 100000017, \{ZBQK002\}, \{C189\}), (C770, 210780581, \{ZZQK002, YYYY030\}, \{D123, T573\}), \ldots\rangle$
P_3	$\langle(H259, 210780110, \{YYYY030\}, \{D123, T573\}), (H259, 210780110, \{ZZQK002\}, \{D123, T573\}), \ldots\rangle$
P_4	$\langle(R91, 210780136, \{YYYY030\}, \{D123, C780\}), (C07, 210780136, \{ZBQK002\}, \{C780\}), \ldots\rangle$

Fig. 9. Care trajectories of 4 patients

ated diagnosis. For example $(C341, 750712184, \{ZBQK002\}, \{D123, K573, C780\})$ represents the stay of a patient in the University Hospital of Dijon (coded as 750712184) treated for a lung cancer (C341), where the patient underwent chest radiography (coded as ZBQK002) and during his treatment, he has the set of associated diagnosis $\{D123, K573, C780\}$.

The experiments extract multidimensional sequential patterns for describing and analyzing patient trajectories. For this experiment the support value is set to 15 (i.e. $\sigma = 0.03$). MMISP generates 156 different frequent trajectories. Figure 10 shows some results of the experiment. *Pattern 2* can be interpreted as follows: 40% of patients had a hospitalization in the University Hospital of Dijon (750712184) for any diagnosis (ALL), where they underwent a chest radiography (coded as ZBQK002) and an Electrocardiography (coded as DEQP003), with supplementary billing (coded as YYYY030); they had a malignant tumor of the lung as associated diagnosis. Then, the same patients had another stay for acute respiratory failure (J960), and they underwent tests with supplementary billing (coded as YYYY030). This second stay could occur in any hospital (ALL) and had the same associated diagnosis(C349).

id	Support	Trajectory Patterns
1	53%	$\langle(710780263, All, \{DEQP003\}, \{C349\})\rangle$
2	40%	$\langle(750712184, All, \{ZBQK002, YYYY030, DEQP003\}, \{C349\})(All, J960, \{YYYY030\}, \{C349\})\rangle$
3	34%	$\langle(710780263, All, \{ZBQK002, YYYY030, DEQP003\}, \{C349\})(710780263, All, \{ZBQK002, YYYY030, DEQP003\}, \{C349\})\rangle$

Fig. 10. Some healthcare patients trajectories obtained by MMISP

In the second experiment, we study the scalability of the approach. We consider the number of extracted patterns and the running time with respect to two different parameters, the number of dimensions and the average length of itemsets in the event. The first batch of synthetic data generated contains 10000 sequences defined over (2, 3, 4 and 5) analysis dimensions. Each sequence contains 30 events and each event is described, in average, by 15 items in the itemset. Each dimension is defined over 5 levels of granularity between elements of each

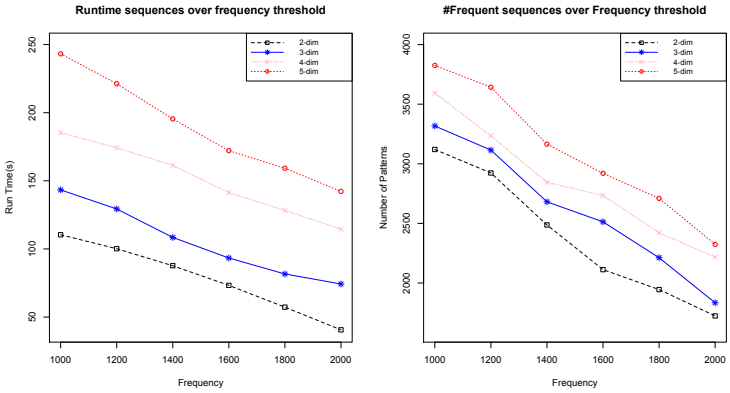


Fig. 11. Running Time (left) and Number of extracted pattern (right) obtained by MMISP with varying in the number of dimension

analysis dimension. Figure 11 reports the results according to different values of support threshold for different number of dimension in event. The running time increases for each newly added dimension. The second batch of generated synthetic data contains 10000 sequences with varying number of items 5, 10, 15 and 20. The sequences in the four generated data sets have an average cardinality of 30 events, by 3 dimensions. The dimensions are defined over 5 levels of granularity between elements of each dimension. Figure 12 reports the results according to different values of support threshold for different lengths of itemsets.

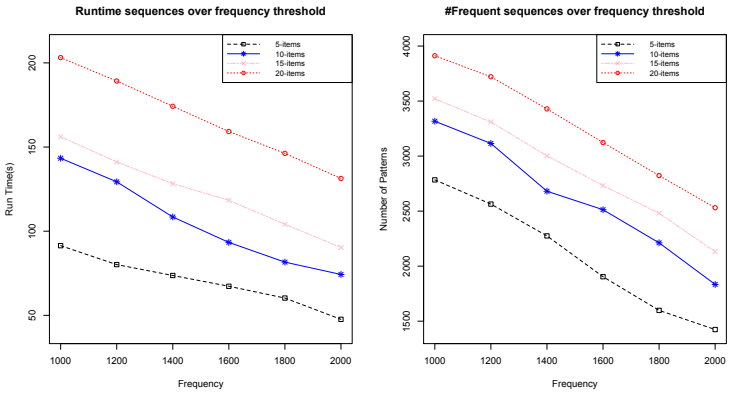


Fig. 12. Number of extracted pattern (right) and Running Time (left) obtained by MMISP with varying itemsets' cardinalities

Another experiment is aimed at comparing the performance of MMISP with M^3SP on a synthetic dataset. In comparison we consider both the number of extracted patterns and the running time. The synthetic data generated contains

10000 sequences defined over two dimensions with one itemsets described by 5 items. Figure 13 reports the results according to different values of support threshold for both M^3SP and MMISP. MMISP is able to extract less patterns than M^3SP while from the point of view of time execution the two approaches show comparable performances. The reduced size of the MMISP results is related to its ability in extracting a multidimensional itemsets sequential patterns.

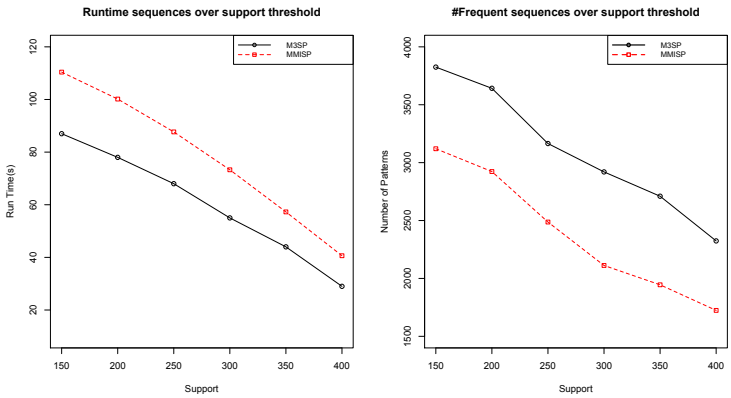


Fig. 13. Running Time (left) and Number of extracted pattern (right) obtained by MMISP and M^3SP over the synthetic dataset

6 Conclusion

In this paper, we propose a new approach to mine multidimensional itemset sequential patterns. Our approach is based on multidimensional items and the set of items. We provide formal definitions and propose a new algorithm MMISP to mine this new kind of pattern. We conduct experiments on both real and synthetic datasets. The method was applied on real-world data where the problem was to mine healthcare patients trajectories and gave potential interesting patterns for healthcare specialists.

References

1. Rakesh Agrawal and Ramakrishnan Srikant. Mining sequential patterns. In *Proceedings of the Eleventh International Conference on Data Engineering*, ICDE '95, pages 3–14, Washington, DC, USA, 1995. IEEE Computer Society.
2. Annalisa Appice, Margherita Berardi, Michelangelo Ceci, and Donato Malerba. *Mining and Filtering Multilevel Spatial Association Rules with ARES*. 2005.
3. Jay Ayres, Jason Flannick, Johannes Gehrke, and Tomi Yiu. Sequential pattern mining using a bitmap representation. In *KDD*, pages 429–435, 2002.

4. Ding-Ying Chiu, Yi-Hung Wu, and Arbee L. P. Chen. An efficient algorithm for mining frequent sequences by a new strategy without support counting. In *ICDE*, pages 375–386, 2004.
5. Jeffrey Cohen, John Eshleman, Brian Hagenbuch, Joy Kent, Christopher Pedrotti, Gavin Sherry, and Florian Waas. Online expansion of largescale data warehouses. *PVLDB*, 4(12):1249–1259, 2011.
6. Elias Egho, Nicolas Jay, Chedy Raïssi, and Amedeo Napoli. A FCA-based analysis of sequential care trajectories. In Amedeo Napoli and Vilem Vychodil, editors, *The Eighth International Conference on Concept Lattices and their Applications - CLA 2011*, Nancy, France, October 2011. INRIA Nancy Grand Est - LORIA.
7. Jiawei Han and Yongjian Fu. Mining multiple-level association rules in large databases. *Knowledge and Data Engineering, IEEE Transactions on*, 11(5):798 –805, sep/oct 1999.
8. Florent Masseglia, Fabienne Cathala, and Pascal Poncelet. The psp approach for mining sequential patterns. In *PKDD*, pages 176–184, 1998.
9. Jian Pei, Jiawei Han, Behzad Mortazavi-Asl, Helen Pinto, Qiming Chen, Umeshwar Dayal, and Meichun Hsu. Prefixspan: Mining sequential patterns by prefix-projected growth. In *ICDE*, pages 215–224, 2001.
10. Helen Pinto, Jiawei Han, Jian Pei, Ke Wang, Qiming Chen, and Umeshwar Dayal. Multi-dimensional sequential pattern mining. In *CIKM*, pages 81–88, 2001.
11. Marc Plantevit, Anne Laurent, Dominique Laurent, Maguelonne Teisseire, and Yeow Wei Choong. Mining multidimensional and multilevel sequential patterns. *TKDD*, 4(1):1–37, 2010.
12. Eliana Salvemini, Fabio Fumarola, Donato Malerba, and Jiawei Han. Fast sequence mining based on sparse id-lists. In *Proceedings of the 19th international conference on Foundations of intelligent systems*, ISMIS'11, pages 316–325, Berlin, Heidelberg, 2011. Springer-Verlag.
13. Ramakrishnan Srikant and Rakesh Agrawal. Mining sequential patterns: Generalizations and performance improvements. In *Proceedings of the 5th International Conference on Extending Database Technology: Advances in Database Technology*, EDBT '96, pages 3–17, London, UK, UK, 1996. Springer-Verlag.
14. Xifeng Yan, Jiawei Han, and Ramin Afshar. Clospan: Mining closed sequential patterns in large datasets. In *In SDM*, pages 166–177, 2003.
15. Zhenglu Yang, Masaru Kitsuregawa, and Yitong Wang. Paid: Mining sequential patterns by passed item deduction in large databases. In *IDEAS*, pages 113–120, 2006.
16. Chung-Ching Yu and Yen-Liang Chen. Mining sequential patterns from multidimensional sequence data. *IEEE Trans. Knowl. Data Eng.*, 17(1):136–140, 2005.
17. Mohammed J. Zaki. Spade: An efficient algorithm for mining frequent sequences. *Mach. Learn.*, 42(1-2):31–60, January 2001.
18. Changhai Zhang, Kongfa Hu, Zhuxi Chen, Ling Chen, and Yisheng Dong. Approxmgmsp: A scalable method of mining approximate multidimensional sequential patterns on distributed system. In *FSKD (2)*, pages 730–734, 2007.