International Journal of Digital Information and Wireless Communications (IJDIWC) 5(3): 165-170
The Society of Digital Information and Wireless Communications, 2015 (ISSN: 2225-658X)

Patterns Mining in Sequences Using Constraints

Rajeb Akram
University of Sfax, MIRACL Laboratory
Sfax, Tunisia
akram.isimg@gmail.com

Ben Hamadou Abdelmajid / Loukil Zied
University of Sfax, MIRACL Laboratory
Sfax, Tunisia
{abdelmajid.benhamadou; zied.loukil}@gmail.com

ABSTRACT

Data mining is a set of methods used in the process of KDD(Knowledge Discovery in Data) in order to distinguish relationships and unknown patterns in the data. Mining patterns is an interesting technique and is widely used in data mining; its objective is to find the patterns that appear frequently in a database. The sequence mining is the pattern discovery problem in the sequences. A declarative approach has been proposed to solve this problem in order to transform this problem to an SAT model.
In this paper, we propose a CSP-based encoding for the problem of discovering frequents and closed patterns in a sequence. We show that is possible to employee constraint programming techniques for modeling and solving a wide variety of constraint-based item-set mining tasks, such as frequent, closed and maximal. Preliminary experiments show that the new formulation is competitive and can outperform the SAT based approach on the considered sequences.

KEYWORDS

Frequent pattern, closed pattern, sequence mining, constraint programming.

1 INTRODUCTION

Data mining is the overall analysis of data whose goal is to seek relationships and generalizations of the data in a new, understandable and useable representation. These data relationships and generalizations can be grouped into two major parts: patterns and models. The difference between these two generalizations is that a pattern is local (used to describe the properties of a subset of terms) whereas the models are global (they characterize the entire set of data).
Data mining applies to the majority of data. For each type of data, an algorithm of search is assigned, among which we can list: transactional databases, multimedia databases, World Wilde Web, data warehouses and files. Mining frequent patterns is an interesting technique and is widely used in data mining. Its goal is to find the patterns that appear frequently in a database. The most famous example of application of the common reasons is the analysis of frequent purchases (frequent patterns in till receipts) in a supermarket. This latter problem is much studied; several algorithms were proposed since 1993 to solve it. In this work we take the problem of finding patterns in a specific class with jokers in sequences. The sequence can be considered as a sub-sequence that contains jokers who could connect any element [6, 3, 9]. We are interested here in enumerating all the patterns in a sequence which occurs at least λ time. The number of extracted patterns is exponential. This exploration is discussed by using several approaches.
The use of constraints in the enumeration of patterns is one of the approaches to solve this latter problem, which has two objectives. The first one is the reduction of the number of patterns to be analyzed, by focusing on the needs, then the elimination of patterns which does not satisfy the needs. The difficulty of deploying patterns, explains the integration of constraints in the process of enumerating of frequent patterns. This use has succeeded to limit the set satisfying the needs. In this work we follow the enumeration approach proposed by S. Jabbour and al [1]. In this work, we follow the constraint programming (CP)

based data mining framework proposed L. Sais and al. in [5] for sequence mining. This recent work offers an SAT-Based approach for discovering frequent, closed and maximal patterns with wildcards in a sequence of items [2, 3, 4, 5, 7]. We propose a CSP encoding of the problem of enumerating frequent and closed patterns with wildcards in a sequence of items using the frequency and closed constraint. The contribution of this paper is that we formulate the encoding of [8, 9] into CSP encoding. Secondly, we present our approach to find frequent and closed patterns in a sequence.

This paper is organized as follows. Section 2 provides an introduction to the main principles of constraint satisfaction problem. Section 3 introduces the problem of frequent pattern mining in a sequence. Section 4 studies closed pattern mining. We describe in the section 4 our CSP approach for frequent and closed pattern in a sequence. Finally, experimental results are conducted and discussed before concluding.

2 PRELIMINAIRES

2.1 Constraint satisfaction problems

In this section we suggest a brief synopsis of constraint programming. Constraint programming is a declarative programming paradigm: the user specifies a problem in terms of constraints, and the system is charged with finding solutions that respect the constraints. The class of problems that focuses on constraint programming systems is the constraint satisfaction problem.
The search can be made easier in cases where the solution instead of corresponding to an optimal path, is only required to satisfy local consistency condition. We call such problems, constraint satisfaction problems (CSP).
A CSP is specified by a set of variables, a domain, which maps by variable to a set of values and a set of constraint.
A CSP consist in deciding if it admits an assignment of values to its variables satisfying all the constraints. Solving a CSP is considered an NP-complete problem.
Algorithms for solving CSPs are called solvers. Some of these solvers have been integrated into a programming language, thus defining a new programming paradigm called constraint programming: to solve a CSP with constraint programming language, it is sufficient to specify constraints; their resolution is supported automatically (without needing a program) by constraint solvers integrated language.
Formally, we can define a CSP by a triple (X, D, C), where:
X = {X1, X2,…,Xn} is the set of variables of the problem
D is the domains of the variables
C is a set of constraint {C1, C2, …., Cn}
For example, we can define the following CSP:
X = {a, b, c, d}
D (a) = D(b)= D(c)= D(d)={0, 1}
C= {a ≠ b, c ≠ d, a + c < b}
This CSP has four variables a, b, c and d each one can take two values (0 or 1).
These variables must satisfy the following constraints: a must be different from b; c must be different from d; the sum of a and c must be less than b.

2.2 Frequent and closed patterns mining in a sequence (F-CPS)

In this section, we introduce the problem of frequent and closed patterns mining in a sequence. Let us first give some preliminary definitions and notations.

Sequence of items

Let Σ be an alphabet built on a finite set of symbols. A sequence S is a succession of characters S1 ... Sn such that Si Є Σ {1 ... m} represents the character at position i in S. The length of the string S is denoted by | S | = n. We denote θ = {1 ... m} as the set of positions of characters in S. A wildcard is an additional character noted o not belonging to Σ (o ∉ Σ) that can match any symbols of the alphabet.

Pattern

A pattern over Σ is a string $p = p_1 \dots p_m$ where $p_1 \in \sum$, $p_m \in \sum$ and $p_i \in \sum \cup \{o\}$ for i=2…..m-1 (Started and ends with a solid character).

Occurrence ($\mathcal{L}$)

We consider the location list $\mathcal{L}_x \subseteq \{1 \dots n\}$ as the set of all the position on s at which x occurs.

Frequent pattern

Let S be a sequence and a pattern p, λ is a positive number called quorum and p is a frequent pattern

in S when $|L_S(p)| \geq \lambda$ .The set of all patterns of S for the quorum λ is denoted: M_S^{λ}.

Closed pattern

In a sequence of items, a frequent pattern p is considered closed if for any frequent pattern q satisfying q ⊃ p, there is no integer d such that $\mathcal{L}_s \subseteq s(q) = \mathcal{L}_s \subseteq s(p) + d$

Where

$\mathcal{L}_s \subseteq s(p) + d = \{l + d \mid l \in \mathcal{L}_s \subseteq s(p)\}$.

EPS (Enumerating patterns in a sequence)

Enumerating all motifs in a sequence can be defined as follows. Let a sequence S and a quorum $\lambda \geq 1$, enumerates all patterns $p \in M_S^{\lambda}$.

3 CONSTRAINT PROGRAMMING MODEL FOR ENNUMERATION PATTERNS IN A SEQUENCE (EPS)

In this section, we describe the CP model for EPS. Our idea consists in expressing the equations defined in [1] to have a Constraints Programming representation.

A sequence of items S over an alphabet ∑ is defined as a sequence s1,…,sn where si ∈ ∑ for I = 1 ,..., n. A pattern p = p1…, pm over ∑ is defined as a sequence of items where the first and the last character are different from empty item. We denote also by |S| = n is size of S and m is the maximal size of the motif, i.e., m= n- λ+1.

Example:

S= XYXYYZZZXYX, λ = 3

n = 11, m = 9

The pattern XY is a frequent pattern in S.

We now present the CP model for enumerate all motifs in a sequence S, given a quorum λ.

- Variables

We introduce two types of variables:

- P = {p1, p2,…,pm} represents the candidate pattern.
- B = {b1, b2,…,bn } represents the support $L_S(p)$, it's a integer variable: bk =0 if the pattern is not located in S at the position k, 1 otherwise.

- Domains
  - Dom (pi) = ∑ U { o }
  - Dom (bk) = { 0,1 }
- Constraints

We first need to enforce the first character to be a solid character (not a joker symbol), the flowing constraint express this propriety:

$$\mathbf{p_1} \neq \mathbf{o} \qquad \text{(C1)}$$

The following propriety: $b_k = 0$ if the pattern is not located in S at the position k, 1 otherwise is expressed by the following constraint:

For all $1 \leq k \leq n$

For all $1 \leq i \leq m$

$$\mathbf{p_i} \neq \mathbf{o} \rightarrow (\mathbf{p_i} \neq \mathbf{S_{i+k-1}} \rightarrow \mathbf{b_k}) \qquad \text{(C2)}$$

In the problem of enumerating all the frequent patterns in a sequence S, we need to express that the candidate pattern occurs at least λ times. This following constraint expresses this propriety:

$$\sum_{\mathbf{k=1}}^{\mathbf{n}} \mathbf{b_k} \leq \mathbf{n} - \boldsymbol{\lambda} \qquad \text{(C3)}$$

The problem of enumerating all frequent patterns is expressed by the constraints C1, C2 and C3.

Example:

S= aba and λ= 2

n= |S|=3

m = n – λ+1= 2

Our model corresponds to the following constraints:

1) P1≠ o
2) B1 =1⇔ [(P1 =0 ∨ P1 = S0) ∧ [(P2 =0 ∨ P2 = S1)]
3) B2 =1⇔ [(P1 =0 ∨ P1 = S1) ∧ [(P2 =0 ∨ P2= S2)]
4) B3 =1⇔ [(P1 =0 ∨ P1 = S2) ∧ [(P2 =0)]
5) B1+ B2 + B3 ≥ λ

Although the frequency constraint can be used to limit the patterns of numbers, it is often rather restrictive to find the necessary patterns. To solve this problem several methods were introduced, the most famous is the condensed representation of the patterns.

In the following section we are going to study the formalization of other types of constraint based on the condensed representations of patterns, to

realize the pruning of the search space and avoid the redundancy of patterns in the process of mining.

3.1 Condensed representation of patterns

Condensed representations aim at avoiding redundant patterns, based on the equivalence class properties. Patterns can be classified according to their supports forming classes called equivalence classes; patterns of the same class have the same supports and frequency.

There are several condensed representations by patterns among which we can quote: represented by closed patterns [10], the representations by maximal patterns [11] and the representations by free patterns [12].

3.1.1 Representations by closed patterns

Patterns can be grouped according to their supports: Patterns which have the same support and thus have the same frequency belonging to the same equivalence class. In an equivalence class the maximal patterns are named closed patterns and the minimal are free patterns. Figure 1 explains all these terms.

Consider the following database (table1):

Id	Items
1	A C D
2	B C E
3	A B C E
4	B E
5	A B C E

Table1. Example of transactions

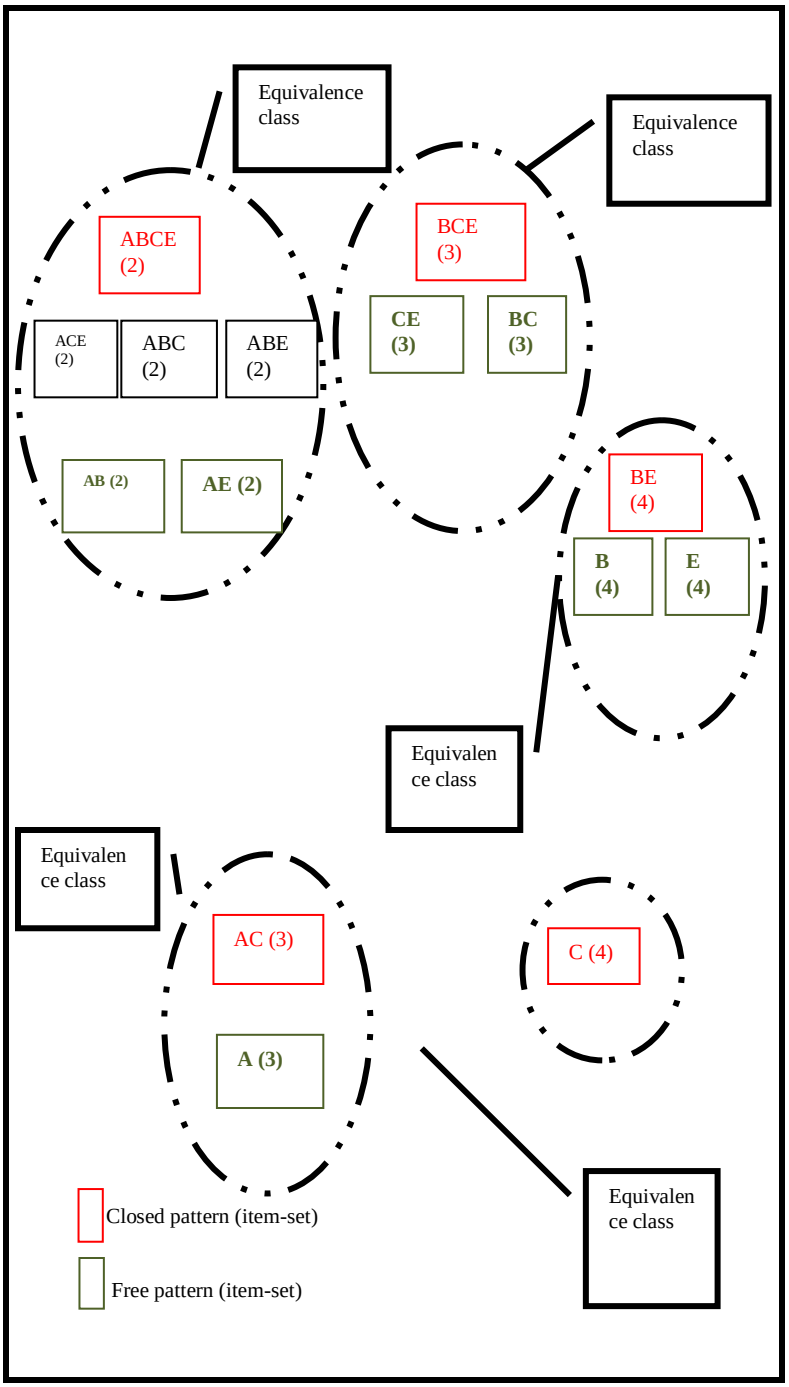

Figure1. Equivalence class

Example

SQ= XXYZXXY
$\lambda = 2$
The set of frequent patterns:
{Pt1=X, Pt2=Y, Pt3=X000X, Pt4=X0Y, Pt5=XX, Pt6=XXY and Pt7=XY}
$\mathcal{L}_{sq}$ (Pt1) = {1, 2, 5, 6}

$\mathcal{L}_{sq}$ (Pt2) = {3, 7}
$\mathcal{L}_{sq}$ (Pt3) = {1, 2}
$\mathcal{L}_{sq}$ (Pt4) = {1, 4}
$\mathcal{L}_{sq}$ (Pt5) = {1, 4}
$\mathcal{L}_{sq}$ (Pt6) = {1, 4}
$\mathcal{L}_{sq}$ (Pt7) = {2, 4}
Pt1 ⊂ Pt3, Pt1 ⊂ Pt4, Pt1 ⊂ Pt4, Pt1 ⊂ Pt5, Pt1 ⊂ Pt6 and Pt1 ⊂ Pt5, but the support of Pt1 is bigger in size, so even shifting can't fall on the support of Pt1.
So, the Pt1 is a frequent closed pattern.
In the same way, Pt3 and Pt4 are considered closed frequent patterns.

3.1.2 Enumerating closed motifs in a sequence

The idea of enumerating closed motifs in sequences consists in adding a new constraint "Closed", its key idea consists in replacing the maximum of wildcards in a pattern with solid characters, i.e. it's impossible to extend the pattern on the left. For this, it suffices to assume that (Sk-j-1 = a) is false: k-j-1 <= 0.
This amounts to complete the sequence on the left by m wildcards (o).
Example:
S= AABCAAB; λ = 2 and m= 6
S= 000000 AABCAAB
Let us now introduce the closeness constraint:

$$\bigwedge_{k=1}^{n}(\neg(\neg q_k \vee S_{k+i-1} = a)) \vee p_i = a \quad \forall\, 1 \le i \le m, a \in \sum \qquad (C4)$$

$$\neg\left(\bigwedge_{i=m-j}^{m} p_i = o\right) \vee \neg\left(\bigwedge_{k=1}^{n}(\neg q_k \vee S_{k-j-1} = a)\right) \quad for\ 0 \le j \le m-2, a \in \sum \qquad (C5)$$

The problem of enumerating all closed, frequent pattern in a sequence S is expressed by the constraints (C1), (C2), (C3), (C4) and (C5).

4 EXPERIMENTS

In this section, we present a preliminary experimental evaluation of our proposed approach. Then, we provide a comparison with the approach SAT for EMS proposed in [8].

We run experiments on PCs with Intel i7 processors and 6GB of RAM. To solve our CP instances, we use the solver CHOCO. CHOCO is a library that implements the basic tools for the constraint programming: domain management, constraint propagation, global process and local search, this library have been implemented in the project OCRE for the purpose is to offer a constraint tool for Research and education (OCRE). It is built in a propagation mechanism based on events with backtrack structures.

In our experiments, we used two data sets with different size, the first one is DS_sz-50[1] (size of sequence = 50) and the second is DS_sz_100[1] (size of sequence =100).

In the first experiment, we show the performance of our approach CSP used the first data set (DS_sz_50). The quorum is also varied linearly (λ_0 =2 and $\lambda_i = \lambda_{i-1} + 1$) Figure2.

In the second experiment, we show the performance of our approach CSP used the second dataset (DS_sz_100[1]). The quorum is also varied linearly (λ_0 =4 and $\lambda_i = \lambda_{i-1} + 2$) Figure3.

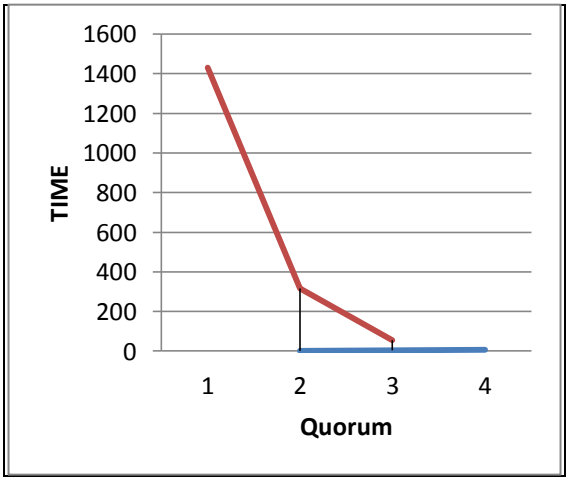

Figure 2. CSP-Closed frequent patterns on DS_sz_50 [1] sequence

[1]http://www.biomedcentral.com/14712105/11/175/additiona/

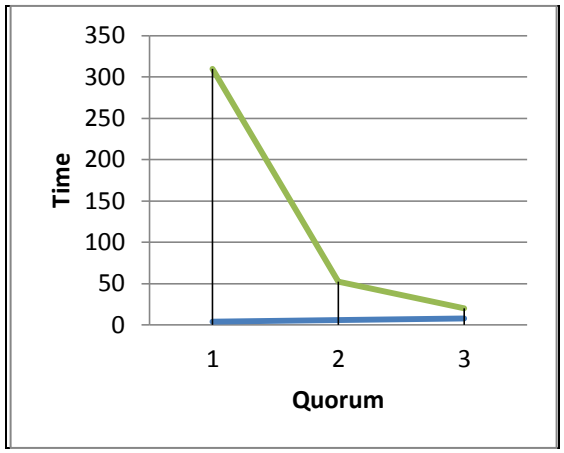

Figure 3. CSP-Closed frequent patterns on DS_sz_100 sequence

In the above experiment we have shown the feasibility of our approach. We have also seen when the quorum decreases, and the size of the sequence increases the CPU time in our approach increases but not very quickly.

5 CONCLUSION AND FUTURE WORKS

In this paper, we presented a CSP approach for enumerating frequent pattern in a sequence. Our proposal reformulates the one proposed in [1] for SAT into a Constraint Programming Problem. First Evaluation shows the feasibility of CSP approach. As a future work, we plan to improve our encoding to enumerate maximal patterns in a sequence. We envisage also to consider the other CSP based solver and to consider some underlying problem as enumerating preferred patterns.

We studied in this paper the problem of finding patterns in a sequence using the frequency and closed constraints.

In the next work, we will study a comparison between our approach and another one such as SAT approach presented in [5].

REFERENCES

[1] E.Coquery, S.Jabbour, and L.Sais, "A Constraint Programming Approach for Enumerating Motifs in a Sequence", in proceedings of the International Workshop on Declarative Pattern Mining (DPM'2011 held in conjunction with IEEE-ICDM'2011). pp 1091-1097, December 11-14, 2011, Vancouver, Canada, 2011.

[2] L. Parida, I. Rigoutsos, A. Floratos and D. Platt, "An out put-sensitive flexible pattern discovery algorithm", in proceding of the 12th Annual Symposium on Combinatorial Pattern Maching (CPM'2001), Volume 2089 of Lecture Notes in computer Science, pp 131-142, Springer, 2001

[3] N. Pisanti, M. Crochemore, R. Grossi, and M.-F. Sagot, "A basis of tiling motifs for generating repeated patterns and its complexity for higher quorum", In Proc. MFCS'03, LNCS 2747, pp 622–631, 2003.

[4] H.Arimura and T.Uno, "An efficient polynomial space and polynomial delay algorithm for enumeration ofmaximal motifs in a sequence", in Journal of Combinatorial Optimization, 13, 2007.

[5] E.Coquery, S.Jabbour, L.Sais and Y.Salhi, "A SAT-Based Approch for Discovering Frequent, Closed and Maximal Pattern in a Sequence, in 20th European Conference on Artificial Intelligence ECAI, pp 258-263, 2012.

[6] L. Parida, I. Rigoutsos, A. Floratos, D. Platt, and Y. Gao, "Pattern discovery on character sets and real-valued data: linear bound on irredundant motifs and efficient polynomial time algorithm", in Proc. the 11th SIAM Symposium on Discrete Algorithms (SODA'00), pp 297–308, 2000

[7] J.Chen and R.Kumar, "Pattern Mining for Predicting Critical Events from Sequential Event Data Log," 2014 IFAC/IEEE International Workshop on Discrete Event Systems, Paris-Cachan, France, May 14-16, 2014.

[8] A.Rajeb, A.Ben Hamdou and Z.Loukil, "On the enumeration of frequent patterns in sequences" in the Inernational Conference on Artificial Intellience and Pattern Recongntion (AIPR'2014), pp 40–344, 2014.

[9] Agrawal, R., Imielinski, T. & Swami, A.N. (1993). "Mining association rules bet-ween sets of items in large databases". In P. Buneman & S. Jajodia, eds., Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, 207-216.

[10] Pasquier N, Bastide Y, Taouil R, Lakhal L (1999) "Discovering frequent closed itemsets for association rules". In Proceedings of the ICDT'99, pp 398–416.

[11] K.Gouda, M.J.Zaki GenMax: "An Efficient Algorithm for Mining Maximal Frequent Itemsets". In Proceedings of Data Mining and Knowledge Discovery'2005, pp 1–20.

[12] BOULICAUT J.-F., BYKOWSKI A. & RIGOTTI C. (2003). Free-sets: "a condensed representation of boolean data for the approximation of frequency queries". Data Mining and Knowledge Discovery, 7(1), pp 5–22.