Découverte de motifs séquentiels et de règles inattendus

H. Li*, A. Laurent**, P. Poncelet*

*LGI2P - EMA, SITE EERIE
Parc scientifique Georges Besse
30035 Nîmes Cedex 1
{haoyuan.li,pascal.poncelet}@ema.fr

**LIRMM - CNRS - Université Montpellier II
161 rue Ada. 34392 Montpellier Cedex 5
laurent@lirmm.fr

Résumé. Jusqu'à présent, les travaux autour de l'extraction de motifs séquentiels se sont particulièrement focalisés sur la définition d'approches efficaces pour extraire, en fonction d'une fréquence d'apparition, des corrélations entre des éléments dans des séquences. Même si ce critère de fréquence est déterminant, le décideur est également de plus en plus intéressé par des connaissances qui sont représentatives d'un comportement inattendu dans ces données (erreurs dans les données, fraudes, nouvelles niches, ...). Dans cet article, nous introduisons le problème de la détection de motifs séquentiels inattendus par rapport aux croyances du domaine. Nous proposons l'approche USER dont l'objectif est d'extraire les motifs séquentiels et les règles inattendues dans une base de séquence. Les expérimentations menées sur des jeux de données réelles et synthétiques montrent la validité de notre approche aussi bien en terme de qualité des motifs extraits que de robustesse par rapport à la taille des bases de données.

1 Introduction

Pour faire face aux besoins des nouvelles applications (médicales, suivi de consommation, suivi des navigations sur un serveur Web, etc), de plus en plus de données sont stockées sous la forme de séquences. Pour traiter ces bases et en extraire des connaissances pertinentes, les motifs séquentiels ont été proposées Agrawal et Srikant (1995). Ils permettent, étant donnée une base de donnée de séquences, de trouver toutes les séquences maximales fréquentes (au sens d'un support minimal défini par l'utilisateur). Ces dernières décrivent alors des tendances qui se dégagent des données et peuvent, par exemple, être utilisées pour optimiser l'architecture d'un site internet afin d'en faciliter la navigation. Il est important de noter que, contrairement aux règles d'association, les motifs séquentiels ne décrivent pas des règles du type "antécédent-conséquent", mais plutôt des corrélations entre les différentes éléments de la base. Si la découverte de telles corrélations dans les données séquentielles est primordiale pour le décideur, il n'en reste pourtant pas moins que certains problèmes ne peuvent être résolus par la recherche de tendances. De nouveaux motifs intéressent le décideur : les motifs inattendus qui

contredisent les croyances acquises sur le domaine. Avec la découverte de tels motifs, il serait, par exemple, possible de faire émerger des tentatives d'attaques sur un réseau, de trouver des données aberrantes dans le système ou encore de trouver de nouvelles niches commerciales.

Par exemple, en considérant une base de données traçant toutes les ventes de produits informatiques d'un grand site marchand, nous pourrions trouver via la détection de tendances avec les motifs séquentiels que, généralement, les acheteurs acquièrent un ordinateur Mac puis un iPod, et ensuite un iPhone. Cependant ce motif, si véridique qu'il soit, ne permet pas d'affiner le positionnement d'une stratégie commerciale, puisqu'il retrace une connaissance déjà acquise par les experts du domaine. En revanche, le fait de trouver que quelques clients ont acheté un ordinateur Mac puis un Mobile fonctionnant sous Windows, est une information importante car elle contredit la connaissance générale que nous avons du domaine.

Rappelons que notre objectif n'est pas de trouver les motifs rares, mais bien les motifs contredisant une connaissance. Si des méthodes d'extraction de contradictions existent dans le contexte des règles d'association, aucune méthode ne permet de mettre en évidence des séquences inattendues par rapport à une connaissance. Dans cet article, nous définissons donc la notion de base de croyance et de contradiction dans le contexte des séquences. Nous parlons alors de séquence inattendue, et nous introduisons les méthodes de découverte de telles séquences. Etant donné que les motifs séquentiels traditionnels ne mettent pas en évidence des règles du type "antécédent-conséquent", nous étendons la notion de séquences inattendues à celles de règles inattendues. Pour extraire ces règles à partir d'une base de données de séquences et d'une base de croyances, nous proposons l'approche USER (Unexpected Sequence Extracted Rules).

L'article est organisé de la manière suivante : la section 2 présente les travaux antérieurs associés à notre problématique. La section 3 introduit l'approche USER pour la découverte automatique de motifs séquentiels et règles inattendus. Pour cela, nous définissons formellement la notion de base de croyances et de séquence inattendue. Nous introduisons ensuite la notion de motif séquentiel inattendu, et de règle séquentielle inattendue décrivant des comportements inattendus au sein d'une base de séquences. Enfin, nous présentons également les algorithmes USE et USR qui composent l'approche USER. La section 4 rapporte les expérimentations que nous avons menées sur des jeux de données synthétiques et réels, avant de conclure et de présenter les perspectives de notre approche dans la section 5.

2 Travaux antérieurs et positionnement

Dans cette section, nous présentons les principaux travaux existants se rapportant aux mesures d'intérêt pour la fouille de données et à la recherche de connaissance inattendue.

Dans McGarry (2005), l'auteur dresse un panorama des principales mesures d'intérêt, qui peuvent, de manière générale, être distinguées selon qu'elles soient objectives ou subjectives. Les mesures objectives s'appuient principalement sur la structure des motifs extraits, et les critères sont alors basés sur les approches probabilistes et de fréquences (e.g. support, confiance, lift). Par contre, les mesures subjectives utilisent généralement une connaissance experte et les critères sont alors l'actionnabilité ou encore l'aspect inattendu.

La recherche de connaissance inattendue à partir d'une base de croyance a été introduite dans Silberschatz et Tuzhilin (1995) comme une mesure subjective pour laquelle les croyances pouvaient être catégorisées entre croyances fortes (ne pouvant être révisées, même face à de

nouveaux faits) et croyances faibles (pouvant être révisées face à de nouvelles données, en modifiant éventuellement le degré de croyance). A partir de cette proposition, Padmanabhan et Tuzhilin (2006) présentent une approche de découverte de règles d'association inattendues. Dans ce cadre, une croyance est représentée par une règle $X \rightarrow Y$, et une règle $A \rightarrow B$ est inattendue par rapport à $X \rightarrow Y$ si : (i) B et Y sont sémantiquement opposés (on notera $B\ et\ Y \models FAUX$) ; (ii) le support et la confiance de la règle $A \cup X \rightarrow B$ sont suffisants ; (iii) le support et la confiance de la règle $A \cup X \rightarrow Y$ ne sont pas suffisants. Le processus d'extraction de connaissance est effectué avec une approche de type *APriori* retournant l'ensemble minimal de règles d'association inattendues par rapport à un système de croyances.

Spiliopoulou (1999) propose un cadre basé sur la connaissance du domaine et des croyances pour trouver des règles séquentielles inattendues à partir de séquences fréquentes. Cette approche est basée sur la définition de séquences généralisés, notées $g_1 * g_2 * \ldots * g_n$ et nommées "g-séquence" pour lesquelles $g_1, g_2, \ldots, g_n$ sont des éléments de la séquence et $*$ une valeur joker. L'auteur définit alors une règle séquentielle comme une séparation de la g-séquence en deux parties adjacentes : une prémisse LHS et une conclusion RHS, notée $LHS \hookrightarrow RHS$. Une croyance définie à partir d'une g-séquence est un n-uplet $\langle LHS, RHS, CL, C\rangle$ où CL est une conjonction de contraintes sur la fréquence de la partie prémisse LHS et C est une conjonction de contraintes sur les éléments LHS et RHS. L'exemple suivant est proposé dans Spiliopoulou (1999) : soit la croyance $\langle a * b, c, CL, C\rangle$ avec $CL = (support(a * b) \geq 0.4 \wedge confidence(a, b) \geq 0.8)$ et $C = (confidence(a * b, c) \geq 0.9)$. Cette croyance signifie que la partie LHS d'une règle séquentielle $a * b \hookrightarrow c$ doit apparaître dans au moins 40% des séquences, la confiance d'une croyance donnée a doit être au minimum de 80% tandis que la confiance de RHS doit être d'au moins 90%. Ainsi une règle séquentielle est attendue si elle confirme une croyance en terme de fréquence. Enfin, les règles inattendues sont groupées par leur partie inattendue et peuvent être utilisées pour créer de nouvelles règles. Même si ces travaux considèrent des séquences inattendues, ils sont différents de notre problématique dans la mesure où la notion d'inattendue concerne des séquences fréquentes sur la base afin de trier les résultats obtenus. Notre objectif est d'extraire, à partir d'une base, toutes les séquences inattendues et d'obtenir des règles elles mêmes inattendues.

IL FAUT REPRENDRE CETTE EXPLICATION DE LA CROYANCE ET MIEUX EXPLIQUER NOTRE APPORT

Nous proposons donc ici d'étendre l'approche proposée par Spiliopoulou. Nous considérons à la fois l'aspect inattendu par rapport à une connaissance du domaine (mesure subjective) et l'aspect valide au sens classique du support et de la confiance (mesures objectives).

3 USER : Extraction de motifs séquentiels et de règles inattendus

Dans cette section, nous présentons notre approche USER. Nous définissons tout d'abord les concepts fondamentaux. A partir de ces définitions, nous définissons de manière formelle ce que nous entendons par base de croyances, et par séquence inattendue. Un motif séquentiel inattendu, dans ce contexte, est alors défini pour permettre la découverte de règles séquentielles inattendues. Par règle, nous entendons qu'il existe un antécédent et un conséquent pour décrire

une causalité. Enfin, nous présentons les algorithmes USE et USR pour découvrir les motifs et les règles séquentiels inattendus.

3.1 Définitions préliminaires

Soit un ensemble d'attributs distincts, on nomme **item** i un attribut de cet ensemble. Un **itemset** $\mathcal{I}$ est une collection non ordonnée d'items, notée $(i_1 i_2 \dots i_m)$. Une **séquence** s est une liste ordonnée d'itemsets, notée $\langle \mathcal{I}_1 \mathcal{I}_2 \dots \mathcal{I}_k \rangle$. Une **base de séquences** $\mathcal{D}$ est un ensemble de séquences (de taille potentiellement très grande).

Soient deux séquences $s = \langle \mathcal{I}_1 \mathcal{I}_2 \dots \mathcal{I}_m \rangle$ et $s' = \langle \mathcal{I}'_1 \mathcal{I}'_2 \dots \mathcal{I}'_n \rangle$, on dit que s est une **sous-séquence** de s', notée $s \sqsubseteq s'$ (s *est contenu dans* s') s'il existe des entiers $1 \leq i_1 < i_2 < \dots < i_m \leq n$ tels que $\mathcal{I}_1 \subseteq \mathcal{I}'_{i_1}, \mathcal{I}_2 \subseteq \mathcal{I}'_{i_2}, \dots, \mathcal{I}_m \subseteq \mathcal{I}'_{i_m}$. Dans un ensemble de séquences, si une séquence s n'est une sous-séquence d'aucune autre, elle est dite **maximale** ; sinon, si s est contenue dans s', on dit que s' **supporte** la séquence s. Soit une séquence $s = \langle \mathcal{I}_1 \mathcal{I}_2 \dots \mathcal{I}_k \rangle$, un **segment** $g \sqsubseteq s$ est une sous-séquence qui contient des itemsets contigus $\langle \mathcal{I}_i \mathcal{I}_{i+1} \dots \mathcal{I}_{i+n} \rangle$ avec $i \leq 1$ et $i + n \leq k$. Le **support** d'une séquence est défini comme la proportion de séquences dans $\mathcal{D}$ qui supportent cette séquence.

Nous définissons, dans cet article, la *longueur* d'une séquence comme le nombre d'itemsets qu'elle contient, noté $|s|$. Nous considérons également la séquence vide et la concaténation de séquences. Une *séquence vide* est notée $\emptyset$, avec $s = \emptyset \iff |s| = 0$. Soient deux séquences s_1 et s_2, la *concaténation* $s_1 \cdot s_2$ de ces deux séquences correspond à s_1 complétée par s_2 en fin de séquence. Nous avons alors $|s_1 \cdot s_2| = |s_1| + |s_2|$.

Pour simplifier les notations et la lecture, dans la suite de cet article, nous utilisons les majuscules $A, B, C \dots$ pour décrire des items, et la notation (ABC) pour désigner des itemsets. La notation $\langle (A)(AC)(BC) \rangle$ désigne une séquence (A puis A et C puis B et C).

Nous notons $\langle \mathbf{op}, n \rangle$ une contrainte sur la longueur de séquences, avec $\mathbf{op} \in \{\neq, =, <, \leq, >, \geq\}$ et $n \in \mathbb{N}$. La notation $|s'| \models \langle \mathbf{op}, n \rangle$ signifie que la longueur de la séquence satisfait $\langle \mathbf{op}, n \rangle$. Par exemple, $|\langle (A)(B)(C) \rangle| \models \langle >, 2 \rangle$ et $|\langle (A)(B) \rangle| \not\models \langle >, 2 \rangle$.

Dans le cas où l'on a $\langle \mathbf{op}, n \rangle = \langle \geq, 0 \rangle$, on note $*$.

Nous introduisons, à présent des notions étendues d'occurrence afin de raffiner la manière de prendre en compte les inclusions de séquence. Rappelons que la notion d' *occurrence* entre sous-séquences permet de décrire la manière dont apparaissent les séquences les unes dans les autres. Soit une séquence s, avec s_1 et s_2 deux sous-séquences de s, i.e. $s_1, s_2 \sqsubseteq s$, telles que s_1 apparaisse avant s_2 dans s. On a donc $s = s_1 \cdot g \cdot s_2$.

L'expression $s_1 \mapsto^{\langle \mathbf{op}, n \rangle} s_2$ indique que s_1 et s_2 apparaissent dans la séquence $s = s_1 \cdot g \cdot s_2$, avec g vérifie $\langle \mathbf{op}, n \rangle$. Dans le cas où $\langle \mathbf{op}, n \rangle = \langle =, 0 \rangle$ ($g = \emptyset$), nous notons $s_1 \mapsto s_2$ le fait que s_1 soit directement suivi de s_2 dans la séquence s. Nous avons alors : $\left\langle s_1 \mapsto^{\langle =, 0 \rangle} s_2 \right\rangle \equiv \langle s_1 \mapsto s_2 \rangle$.

Dans le cas le moins contraint, l'écriture $s_1 \mapsto^* s_2$ désigne le fait que s_2 apparaît après s_1 dans s ($s = s_1 \cdot s' \cdot s_2$ sans contrainte sur s').

3.2 Base de croyances et séquences inattendues

Une croyance représente une connaissance décrite sous la forme d'une relation de causalité temporelle entre des occurrences d'éléments dans une séquence.

Dans notre approche, nous utilisons des *règles* pour décrire les relations de causalité entre séquences. De manière similaire à Spiliopoulou (1999), nous notons s_α la prémisse et s_β la conclusion. Ceci signifie que $s_\alpha \Rightarrow s_\beta$ est satisfaite dans s, si l'occurrence de $s_\alpha \sqsubseteq s$ implique l'occurrence d'une sous-séquence $s_\beta \sqsubseteq s$ telle que $s_\alpha \cdot s_\beta \sqsubseteq s$.

Par exemple la règle $\langle(Mac)\rangle \Rightarrow \langle(iPhone)\rangle$ décrit que l'achat d'un ordinateur Mac implique l'achat d'un iPhone plus tard.

Dans notre approche, ce sont les experts qui définissent les règles à prendre en compte, en donnant la base de croyances définie comme un ensemble de croyances.

Définition 1 (Croyance)**.** *Une croyance c sur une séquence est un couple $(p, \mathcal{C})$ tel que :*

$c : (p, \mathcal{C})$
$p : s_\alpha \Rightarrow s_\beta$
$\mathcal{C} : \{\tau, \eta\}$
$\tau : \langle \mathbf{op}\ n\rangle, \mathbf{op} \in \{\neq, =, <, \leq, >, \geq\}, n \in \mathbb{N}$
$\eta : s_\beta \not\sim s_\gamma$

où p et τ forment une règle $s_\alpha \mapsto^{\langle \mathbf{op}, n\rangle} s_\beta$ et où η spécifie que l'occurrence de s_β ne peut pas être remplacée par une occurrence de s_γ. La croyance c est alors notée $[s_\alpha; s_\beta; s_\gamma; \tau]$. Dans le cas où $\tau = \langle \geq, 0\rangle$, on note $*$.

Soit une croyance b, une séquence s est *inattendue* par rapport à b si s viole l'une des contraintes introduites par b.

Exemple 1. Soit une croyance $b = [\langle(A)(B)\rangle ; \langle(C)(D)\rangle ; \langle(E)(F)\rangle ; < 2]$, la séquence $s_1 = \langle(A)(AB)(E)(C)(DE)\rangle$ est attendue par rapport à b puisqu'entre l'occurrence de $\langle(A)(B)\rangle$ et $\langle(C)(D)\rangle$ on a $\langle(E)\rangle$ avec $|\langle(E)\rangle| = 2 \not< 2$. La séquence $s_2 = \langle(A)(B)(E)(D)(C)(D)\rangle$ est en revanche inattendue par rapport à b puisqu'elle contredit la contrainte de fenêtre temporelle. La séquence $s_3 = \langle(A)(B)(C)(CE)(F)\rangle$ est inattendue par rapport à b puisqu'elle contredit la contrainte sur s_γ. La séquence $s_4 = \langle(A)(B)(E)(F)(C)(D)\rangle$ est inattendue par rapport à b puisqu'elle contredit à la fois τ et η. La séquence $s_5 = \langle(A)(C)(B)(E)\rangle$ n'est pas concernée par la croyance b. □

En nous appuyant sur ces contradictions possibles de contraintes, nous distinguons trois types de violation (caractères inattendus) : α-inattendu, β-inattendu, γ-inattendu.

Définition 2 (séquence α-inattendue)**.** *Soit une croyance $b = [s_\alpha; s_\beta; s_\gamma; *]$ et une séquence s. s est dite α-inattendue par rapport à b si $s_\alpha \sqsubseteq s$ et il n'existe pas s_β, s_γ tels que $\langle s_\alpha \mapsto^* s_\beta\rangle \sqsubseteq s$ ou $\langle s_\alpha \mapsto^* s_\gamma\rangle \sqsubseteq s$.*

Définition 3 (séquence β-inattendue)**.** *Soit une croyance $b = [s_\alpha; s_\beta; s_\gamma; \tau]$ où la contrainte τ est différente de $*$, et une séquence s. s est dite β-inattendue par rapport à b si $\langle s_\alpha \mapsto^* s_\beta\rangle \sqsubseteq s$ et s'il n'existe pas s' tel que $|s'| \models \tau$ et $\langle s_\alpha \mapsto s' \mapsto s_\beta\rangle \sqsubseteq s$,*

Dans l'exemple 1, les séquences s_2 et s_4 correspondent à uneβ-violation de la croyance b.

Définition 4 (séquence γ-inattendue)**.** *Soit une croyance $b = [s_\alpha; s_\beta; s_\gamma; \tau]$ et une séquence s. s est dite γ-inattendue par rapport à b si $\langle s_\alpha \mapsto^* s_\gamma\rangle \sqsubseteq s$ et il existe s' tel que $|s'| \models \tau$ et $\langle s_\alpha \mapsto s' \mapsto s_\gamma\rangle \sqsubseteq s$.*

Notons qu'une séquence inattendue peut être liée à deux contraintes violées. Ainsi une séquence peut être à la fois α- et γ-inattendue, ou être à la fois β- et γ-inattendue.

Les exemples ci-dessous montrent de telles séquences.

Exemple 2. Nous reprenons ici l'exemple de la section 1 où M désigne un "ordinateur Mac", P un "iPhone", et W un "téléphone Windows Mobile". La croyance considérée est $b = [\langle M \rangle ; \langle P \rangle ; \langle W \rangle ; *]$. Une séquence s d'un client ayant acheté un ordinateur Mac est inattendue si s montre que le client n'a acheté ni iPhone ni téléphone Windows Mobile (α-contradiction de b), ou si le client a ensuite acheté un téléphone Windows Mobile (γ-contradiction de b). □

Exemple 3. Si nous considérons maintenant un serveur WebMail, où un utilisateur valide se connectant doit être redirigé vers la page du courrier entrant. Une croyance serait alors $[\langle L \rangle ; \langle M \rangle ; \langle O \rangle ; = 0]$ où O désigne la page de déconnexion. Une séquence β-inattendue correspondrait alors à un utilisateur qui n'aurait pas été redirigé vers la page de courrier entrant. Une séquence γ-inattendue serait que l'utilisateur est redirigé vers la page de déconnexion (on pourrait alors soupçonner des pannes de service). □

3.3 Motifs séquentiels inattendus et règles séquentielles inattendues

Une fois la notion de séquence inattendue définie, il est intéressant de découvrir les tendances au sein de ces séquences afin de faire émerger des tendances générales liées à ces problèmes et découvrir ainsi les motifs associés (par exemple pour caractériser une attaque ou une fraude). Nous utilisons pour cela le cadre des motifs séquentiels afin d'extraire les séquences fréquentes maximales à partir des séquences contenant des violations de contraintes. À partir d'une base de données de séquences $\mathcal{D}$ et d'une base de croyances $\mathcal{B}$, nous utilisons le support pour déterminer à quel point une séquence inattendue s_u issue d'une séquence $s \in \mathcal{D}$ est fréquente au sens de la croyance $b \in \mathcal{B}$ et du type de violation $u \in \{\alpha, \beta, \gamma\}$. On note D_u le sous-ensemble de D composé des séquences violant b pour le type de contradiction u. On a alors :

$$supp(s_u) = \frac{|\{s \in \mathcal{D}_u | s_u \sqsubseteq s\}|}{|\mathcal{D}_u|}.$$

Définition 5 (Motif séquentiel inattendu)**.** *Soit une base de données $\mathcal{D}$, u un type de violation de contrainte, et $\mathcal{D}_u$ le sous-ensemble de $\mathcal{D}$ contenant les séquences $s \sqsubseteq \mathcal{D}$ telles que s est une u-violation de croyance. Un motif séquentiel inattendu est une séquence maximale fréquente, c'est-à-dire dont le support est supérieur à un seuil fixé par l'utilisateur.*

Les motifs séquentiels inattendus permettent donc de mettre en valeur les dépendances entre séquences inattendues pour une contrainte de type $u \in \alpha, \beta, \gamma$. L'exemple 4 illustre ce type de motif.

Exemple 4. Soit une croyance $b = [\langle A \rangle ; \langle B \rangle ; \langle C \rangle ; = 1]$, nous considérons trois séquences β-inattendues par rapport à b.

$$s_1 = \langle (D)(AB)(CD)(D)(BC)(E) \rangle ,$$
$$s_2 = \langle (D)(AB)(D)(E)(BD)(E) \rangle ,$$
$$s_3 = \langle (C)(A)(E)(E)(B)(C) \rangle .$$

On a donc $supp(\langle(D)(AB)(D)(B)(E)\rangle) = 2/3$. Avec une valeur de support minimum 0.5, $\langle(D)(AB)(D)(B)(E)\rangle$ est un motif séquentiel β-inattendu pour b. □

Afin de mieux identifier les segments de séquence correspondant à la partie "souche" et aux parties de contradiction, nous introduisons la notion de *séquence inattendue bornée*.

Définition 6 (séquence α-inattendue bornée)**.** *Soit une croyance $b = [s_\alpha; s_\beta; s_\gamma; *]$ et une séquence s telle que $s = g' \cdot s_\alpha \cdot g$ avec $s_\alpha \not\sqsubseteq g'$, $s_\beta \not\sqsubseteq g$, $s_\gamma \not\sqsubseteq g$ (s est α-inattendue pour b). La séquence α-inattendue bornée est définie comme le segment $s_b = s_\alpha \cdot g$.*

Définition 7 (séquence β-inattendue bornée)**.** *Soit une croyance $b = [s_\alpha; s_\beta; s_\gamma; \tau]$ et une séquence s telle que $s = g' \cdot s_\alpha \cdot g \cdot s_\beta \cdot g''$ où le segment g ne satisfait pas τ (s est β-inattendue pour b). La séquence β-inattendue bornée est définie comme le segment $s_b = s_\alpha \cdot g \cdot s_\beta$.*

Définition 8 (séquence γ-inattendue bornée)**.** *Soit une croyance $b = [s_\alpha; s_\beta; s_\gamma; \tau]$ et une séquence $s = g' \cdot s_\alpha \cdot g \cdot s_\gamma \cdot g''$ où (s est γ-inattendue pour b). La séquence γ-inattendue bornée est définie comme le segment $s_b = s_\alpha \cdot g \cdot s_\gamma$.*

Exemple 5. Comme décrit dans l'exemple 4, les séquences s_1, s_2 et s_3 sont inattendues pour la croyance $b = [\langle A\rangle ; \langle B\rangle ; \langle C\rangle ; = 1]$. Les séquences $\langle(AB)(CD)(D)(BC)\rangle, \langle(AB)(D)(E)(BD)\rangle$ et $\langle(A)(E)(E)(B)\rangle$ sont alors les séquences bornées inattendues correspondant à s_1, s_2 et s_3. □

Une séquence inattendue s peut donc être représentée comme $s = g_a \cdot s_b \cdot g_c$ où s_b est une séquence bornée inattendue correspondant à la violation de contrainte de croyance, et g_a, g_c sont deux segments de s. Nous avons $|s_b| > 0$, $|g_a| \geq 0$ et $|g_c| \geq 0$. Le segment $g_a \sqsubseteq s$ est appelé *antécédent* et le segment $g_c \sqsubseteq s$ est appelé *conséquent*. Soit un ensemble de séquences inattendues supportant le type de violation u, on note l'ensemble de tous les antécédents (y compris l'ensemble vide) $\mathcal{D}_u^a$, et l'ensemble de tous les conséquents $\mathcal{D}_u^c$. Le support d'une séquence s_a contenu dans $\mathcal{D}_u^a$ et s_c contenu $\mathcal{D}_u^c$ est la fraction de séquences de $\mathcal{D}_u^a$ ou de $\mathcal{D}_u^c$ qui supportent s_a ou s_c, c'est-à-dire

$$supp(s_a) = \frac{|\{s \in \mathcal{D}_u^a | s_a \sqsubseteq s\}|}{|\mathcal{D}_u^a|}$$

et

$$supp(s_c) = \frac{|\{s \in \mathcal{D}_u^c | s_c \sqsubseteq s\}|}{|\mathcal{D}_u^c|}.$$

Une séquence maximale fréquente contenue dans $\mathcal{D}_u^a$ est une *séquence antécédent fréquente* et une séquence fréquente maximale contenue dans $\mathcal{D}_u^c$ est une *séquence fréquente conséquent*.

Définition 9 (Règle Antécédent)**.** *Soit un ensemble $\mathcal{D}_u$ de séquences inattendues de type $u \in \alpha, \beta, \gamma$. Soit $\mathcal{D}_u^a$ l'ensemble de toutes les séquences antécédent contenues dans $\mathcal{D}_u$ et s_a une séquence antécédent contenue dans $\mathcal{D}_u^a$ fréquente par rapport à un support défini par l'utilisateur σ_a. Une règle antécédent est une règle de la forme $s_a \Rightarrow u$.*

Les règles antécédent reflètent les éléments d'une séquence qui sont en amont d'une violation de contrainte d'une croyance donnée. Donc, dans une règle antécédent, nous pouvons décrire le lien de causalité de comportements inattendus au sein d'une séquence.

Définition 10 (Règle Conséquent)**.** *Soit un ensemble $\mathcal{D}_u$ de séquences inattendues de type u Soit $\mathcal{D}_u^c$ l'ensemble de toutes les séquences conséquent contenues dans $\mathcal{D}_u$ et soit s_c la séquence conséquente contenue dans $\mathcal{D}_u^c$ fréquente par rapport au seuil σ_c fixé par l'utilisateur. Une règle conséquent est une règle de la forme $u \Rightarrow s_c$.*

Les règles conséquent reflètent les éléments d'une séquence qui correspondent au caractère inattendu par rapport à une croyance.

Décrites par leur support, les règles antécédent et conséquent sont également décrites par leur *confiance*.

Nous avons donc $supp(s_a \Rightarrow u) = supp(s_a)$ et $supp(u \Rightarrow s_c) = supp(s_c)$.

Définition 11 (Support et Confiance de règles)**.** *Soit une base de données de séquences $\mathcal{D}$ et un type de violation u. Soit l'ensemble d'antécédents $\mathcal{D}_u^a$ et l'ensemble des conséquents $\mathcal{D}_u^c$. La valeur du support d'une règle $s_a \Rightarrow u$ équivaut à la valeur de support de s_a de même que la valeur de support de $u \Rightarrow s_c$ est égale à la valeur de support de s_c. La confiance d'une règle antécédent est donnée par :*

$$conf(s_a \Rightarrow u) = \frac{|\{s \in \mathcal{D}_u^a | s_a \sqsubseteq s\}|}{|\{s \in \mathcal{D} | s_a \sqsubseteq s\}|},$$

et la confiance d'une règle conséquent est donnée par :

$$conf(u \Rightarrow s_c) = \frac{|\{s \in \mathcal{D}_u^c | s_c \sqsubseteq s\}|}{|\{s \in \mathcal{D} | s \models u\}|}.$$

Exemple 6. Nous reprenons l'exemple 3 et la base de croyance associée. Supposons que nous disposons d'un fichier contenant 10 000 sessions utilisateurs comprenant des informations $(Temps, IP, Requete)$ où $Temps$ désigne une plage temporelle, IP identifie une plage d'adresses IP, et $Requete$ désigne les ressources requises avec $Requete \in \{$ *Begin, End, Help, Login, Logout, Recall, ...*$\}$, où $Recall$ est la page de rappel du mot de passe et $Help$ décrit la page d'aide en ligne. Dans un tel fichier log, chaque session utilisateur est une séquence.

Supposons, à présent, que 100 séquences soient β-inattendues. Nous avons alors $supp(u_\beta) = 0.01$. Soit $\sigma_a = \sigma_c = 0.1$. Supposons que 80 séquences supportent la séquence antécédent fréquente $\langle(t1, ip1, Begin)\rangle$ (support 0.8), que 10 séquences supportent la séquence antécédent fréquente $\langle(ip2, Begin)\rangle$ (support 0.1), que 80 séquences supportent la séquence conséquent fréquente $\langle(t1, ip1, End)\rangle$, que 15 séquences supportent la séquence conséquent fréquente $\langle(ip2, Recall)(ip2, End)\rangle$ (support 0.15), et que 10 séquences supportent la séquence conséquent $\langle(ip2, Help)(ip2, End)\rangle$. Si le nombre total de séquences supportant $\langle(t1, ip1, Begin)\rangle$ et $\langle(t1, ip1, End)\rangle$ est de 80, alors la confiance de $\langle(t1, ip1, Begin)\rangle \models u_\beta$ est de 1, la confiance de $u_\beta \models \langle(t1, ip1, End)\rangle$ est de 0.8. Si le nombre total de séquences supportant $\langle(ip2, Begin)\rangle$ est de 9000, alors la confiance de $\langle(ip2, Begin)\rangle \models u_\beta$ est de $1/900$. Dans cet exemple, il est donc clair que les connexions depuis l'intervalle 1 d'IP au temps 1 peuvent être considérées comme des attaques. □

Une fois ces définitions posées, nous étudions comment rechercher ces connaissances au sein de bases de données de séquences. Pour ce faire, nous proposons l'approche USER.

3.4 L'approche USER

Nous supposons connue une base de croyances et cherchons les comportements inattendus dans une base de données de séquences. L'approche décrite ici s'articule autour de deux phases. Dans la première phase, l'algorithme USE (Unexpected séquence Extraction) extrait toutes les séquences inattendues pour chaque type de violation de contrainte et pour chaque croyance. Dans une seconde phase, l'algorithme USR (Unexpected séquence Rules) trouve tous les motifs séquentiels inattendus et les règles associées à partir des séquences inattendues trouvées par USE, à partir de seuils de support/confidence définis a priori.

USE est décrit de manière détaillée dans l'algorithme 1. Il prend en entrée une base de séquences $\mathcal{D}$ et une base de croyances $\mathcal{B}$, et produit en sortie l'ensemble de séquences inattendues $\mathcal{D}_u$, ainsi que la séparation entre les séquences antécédent $\mathcal{D}_u^a$ et les séquences conséquent $\mathcal{D}_u^c$ pour chaque type de violation $u \in \alpha, \beta, \gamma$.

Nous rappelons que si $s' \sqsubseteq s$ alors c'est qu'il existe *au moins une* occurrence de s' dans s. Cependant, il est tout à fait possible qu'un itemset $\mathcal{I}_i \in s'$ soit *redondant*. Soit une séquence s et deux sous-séquences s', et $s'' \sqsubseteq s$ avec $s' = \langle \mathcal{I}'_1 \dots \mathcal{I}'_{n'} \rangle$ et $s'' = \langle \mathcal{I}''_1 \dots \mathcal{I}''_{n''} \rangle$. Si $s' \sqsubseteq s_{''}$ et $|s'| < |s''|$, alors s'' est une occurrence *redondante* de s' dans s ; sinon si nous avons $|s'| = |s''|$, alors s'' est une occurrence non redondante de s' dans s.

Dans notre approche actuelle, nous trouvons la première occurrence de s_{α} et toutes les occurrences de s_β ou de s_γ par rapport à la relation d'occurrence de s_{α}.

Pour les α-inattendus, la fonction $matchi$ retrouve la première occurrence de $s_{\alpha} \sqsubseteq s$ puis s'assure que $\langle s_\alpha \mapsto^* s_\beta \rangle \not\sqsubseteq s$ et $\langle s_\alpha \mapsto^* s_\gamma \rangle \not\sqsubseteq s$ par la fonction $match$. Pour les β-inattendus et les γ-inattendus, il faut trouver un segment $g_u \sqsubseteq s$ entre les occurrences de s_α et s_β, ou de s_α et s_γ. Il s'agit donc de trouver une séquence s' telel que $|s'| \not\models \tau$ et $\langle s_\alpha \mapsto s' \mapsto s_\beta \rangle \sqsubseteq s$, ou telle que $|s'| \models \tau$ et $\langle s_\alpha \mapsto s' \mapsto s_\gamma \rangle \sqsubseteq s$. La fonction $matchf$ quant à elle permet de retrouver la première occurrence de $s_\beta \sqsubseteq s$ ou $s_\gamma \sqsubseteq s$ à partir de l'occurrence de s_α et de la contrainte τ.

USR (Algorithme 2) prend en entrée une base de séquences $\mathcal{D}$, une base de croyances $\mathcal{B}$, les valeurs de support minimum σ_u, σ_a et σ_c, les valeurs de confiance minimum δ_a et δ_c, et les ensembles de séquences produits par l'algorithme USE. Pour chaque inattendu violant une croyance $b \in \mathcal{B}$, cet algorithme trouve d'abord les motifs séquentiels inattendus à partir de l'ensemble des séquences inattendues $\mathcal{D}_u$ par rapport à σ_u, puis trouve les motifs séquentiels à partir des ensembles de séquences $\mathcal{D}_u^a$ et $\mathcal{D}_u^c$ par rapport à σ_a et σ_c. Enfin, l'algorithme génère les règles antécédent et conséquent pour chaque type d'inattendu u par rapport à δ_{a} et δ_c.

La fonction $FindSequentialPatterns$ utilise les algorithmes classiques d'extraction de motifs séquentiels, plusieurs méthodes ayant été proposées, dont par exemple PSP Masseglia et al. (1998), SPADE Zaki (2001) ou encore PrefixSpan Pei et al. (2004).

4 Expérimentations

Deux séries d'expérimentations ont été menées pour valider notre approche. La première série consiste à extraire des motifs séquentiels et des règles inattendus à partir d'un fichier log d'un serveur Web existant, avec une base de croyance définis par des experts du domaine. Nous démontrons ainsi la pertinence de notre approche. Le second groupe d'experimentations est dédié au passage à l'échelle, étudié à partir de bases de données denses synthétisées grâce

Algorithm 1 Algorithme USE

Input: Base de séquences $\mathcal{D}$ et base de croyances $\mathcal{B}$

Output: Ensemble $\mathcal{D}_u$ de séquences inattendues, $\mathcal{D}_u^a$ de séquences antécédent et $\mathcal{D}_u^c$ de séquences conséquent pour chaque type de violation u

for all $s \in \mathcal{D}$ **do**
  for all $b \in \mathcal{B}$ **do**
    /* α-inattendus */
    for all $u_\alpha \vdash b$ **do**
      if $occu_\alpha \leftarrow matchi(s, b.s_\alpha)$ **then**
        if $matchf(s, b.s_\beta, occu_\alpha)$ **and** $matchf(s, b.s_\gamma, occu_\alpha)$ **then**
          ignore
        else
          $\mathcal{D}_{u_\alpha} \leftarrow \mathcal{D}_{u_\alpha} \cup s$
          $\mathcal{D}^a_{u_\alpha} \leftarrow \mathcal{D}^a_{u_\alpha} \cup subseq(s, s.begin, occu_\alpha.begin)$
          $\mathcal{D}^c_{u_\alpha} \leftarrow \mathcal{D}^c_{u_\alpha} \cup subseq(s, occu_\alpha.end, s.end)$
        end if
      end if
    end for
    /* β-inattendus */
    for all $u_\beta \vdash b$ **do**
      if $occu_\alpha \leftarrow matchi(s, b.s_\alpha)$ **then**
        if $occu_\beta \leftarrow matchf(s, b.s_\beta, occu_\alpha, b.\tau)$ **then**
          $\mathcal{D}_{u_\beta} \leftarrow \mathcal{D}_{u_\beta} \cup s$
          $\mathcal{D}^a_{u_\beta} \leftarrow \mathcal{D}^a_{u_\beta} \cup subseq(s, s.begin, occu_\alpha.begin)$
          $\mathcal{D}^c_{u_\beta} \leftarrow \mathcal{D}^c_{u_\beta} \cup subseq(s, occu_\beta.end, s.end)$
        end if
      end if
    end for
    /* γ-inattendus */
    for all $u_\gamma \vdash b$ **do**
      if $occu_\alpha \leftarrow matchi(s, b.s_\alpha)$ **then**
        if $occu_\gamma \leftarrow matchf(s, b.s_\gamma, occu_\alpha, b.\tau)$ **then**
          $\mathcal{D}_{u_\gamma} \leftarrow \mathcal{D}_{u_\gamma} \cup s$
          $\mathcal{D}^a_{u_\gamma} \leftarrow \mathcal{D}^a_{u_\gamma} \cup subseq(s, s.begin, occu_\alpha.begin)$
          $\mathcal{D}^c_{u_\gamma} \leftarrow \mathcal{D}^c_{u_\gamma} \cup subseq(s, occu_\gamma.end, s.end)$
        end if
      end if
    end for
  end for
end for
for all $u \vdash b$ **do**
  $\mathcal{D}_u, \mathcal{D}^a_u, \mathcal{D}^c_u$
end for

Algorithm 2 Algorithme USR

Input: Base de séquences $\mathcal{D}$, base de croyances $\mathcal{B}$, ensembles de séquences produits par USE, valeurs de support minimum $\sigma_u, \sigma_a, \sigma_c$, et valeurs de confiance minimale δ_a, δ_c

Output: Ensemble P^u de motifs séquentiels inattendus, ensemble R_u^a de règles antécédent inattendues et R_u^c de règles conséquent inattendues pour chaque type de violation u

for all $b \in \mathcal{B}$ **do**
  for all $u \vdash b$ **do**
    $\mathcal{P}_u \leftarrow FindSequentialPatterns(\mathcal{D}_u, \sigma_u)\ \mathcal{P}_u$
    $\mathcal{P}_u^a \leftarrow FindSequentialPatterns(\mathcal{D}_u^a, \sigma_a)$
    $\mathcal{P}_u^c \leftarrow FindSequentialPatterns(\mathcal{D}_u^c, \sigma_c)$
    for all $s_a \in \mathcal{P}_u^a$ **do**
      if $|s_a| \,/\, |\mathcal{D}| \geq \delta_a$ **then**
        $\mathcal{R}_u^a \leftarrow \mathcal{R}_u^a \cup \{s_a \Rightarrow u\}$
      end if
    end for$\mathcal{R}_u^a$
    for all $s_c \in \mathcal{P}_u^c$ **do**
      if $|s_c| \,/\, |\mathcal{D}_u| \geq \delta_a$ **then**
        $\mathcal{R}_u^c \leftarrow \mathcal{R}_u^c \cup \{u \Rightarrow s_c\}$
      end if
    end for$\mathcal{R}_u^c$
  end for
end for

au générateur IBM Quest Synthetic Data Generator[1], les bases de croyance étant alors générées aléatoirement.

Les expérimentations ont été menées sur un Sun Fire V880 (8 processeurs 1.2GHz UltraSPARC III et 32GB de mémoire centrale), sous système Solaris 10.

4.1 Analyse de fichier log

Le fichier analysé contient 2,271,955 d'enregistrements d'accès au site Web. L'approche USER a été utilisée, après pré-traitement de la base pour la décrire en 67,228 séquences intégrant les 27,552 items distincts.

Type de Croyance	Croyance	Sessions	Motifs séquentiels	Règles
Type de fichier et Statut	5	2586		
Mots clés	10	354		
Workflow	10	511		

TAB. 1 – *Résultats expérimentaux sur le jeu de données réel du serveur Web*

Le tableau 1 montre nos trois classes d'expérimentations sur ces données log, avec différentes compositions de bases de croyance. Avec 5 croyances sur le type de fichier et un statut

[1] *http : //www.almaden.ibm.com/cs/projects/iis/hdb/Projects/data_mining/datasets/syndata.html*

HTTP, le type de fichier ".cgi" et le code HTTP "404", nous avons trouvé 2586 sessions correspondant à des séquences inattendues. Avec 10 croyances associées aux URI et des mots clés tels que "etc" et "passwd" dans les requêtes HTTP, 354 sessions ont été détectées comme correspondant à des séquences inattendues. Enfin, avec 10 croyances sur les enchaînements tels que "/admin/" doit être suivi par "index.php" puis par "login.php" puis par "sql.php", 511 sessions ont été identifiées comme correspondant à des séquences inattendues.

4.2 Passage à l'échelle

Le passage à l'échelle de USER a été testé avec un nombre fixe de 20 croyances en augmentant la taille de la base de séquences de 10,000 séquences à 500,000 séquences, puis en considérant une base de 100,000 sequences en augmentant le nombre de croyance de 5 à 25.

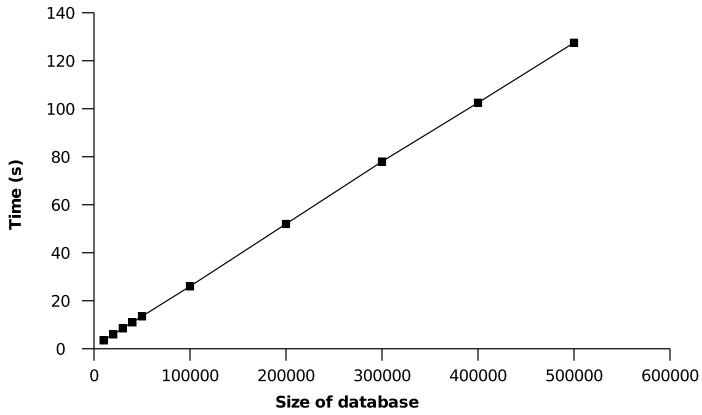

FIG. 1 – *Temps d'extraction des séquences inattendues avec 20 croyances*

La Figure 1 montre que, quand le nombre de croyances est fixe, le temps d'extration de toutes les séquences inattendues croît linéairement par rapport à la taille de la base.

La Figure 2 montre que, quand la taille de la base est fixe, le nombre de séquences inattendues extraites croît de manière non linéaire par rapport au nombre de croyances. Notons que dans ce test, les 10 dernières croyances sont beaucoup moins concernées que les autres.

La Figure 3 montre l'augmentation du temps d'extraction de toutes les séquences inattendues.

5 Conclusions

Dans cet article, nous avons introduit la problématique de la recherche de motifs séquentiels et règles séquentielles inattendus. Cette recherche s'effectue à partir d'une base de croyance que nous avons définie formellement. Nous pouvons ainsi mettre en évidence les comportements inattendus au sein d'une base de séquences, ce qui trouve de très nombreuses applications dans les bases de données réelles (détection de pannes, de fraudes, de niches commerciales, etc.) L'approche USER est proposée, décomposée en différentes étapes successives

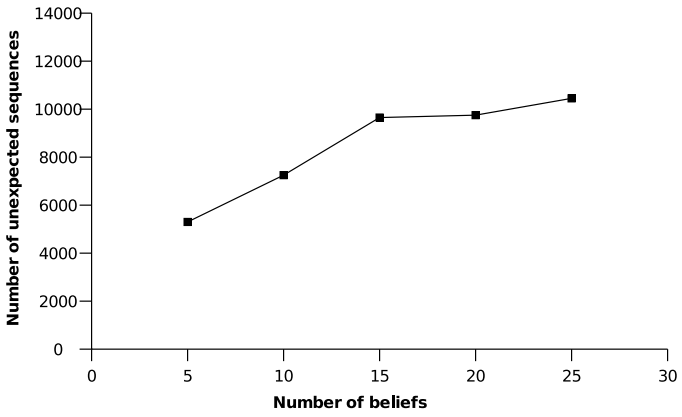

FIG. 2 – *Nombre de séquences inattendues extraites parmi 100,000 séquences.*

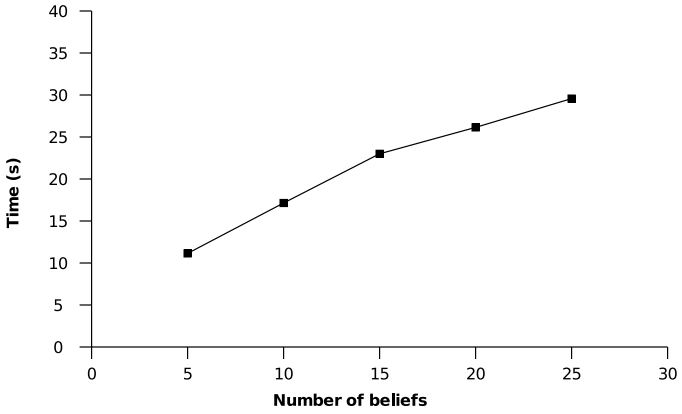

FIG. 3 – *Temps d'extraction de toutes les séquences inattendues pour 100,000 séquences*

(USE/USR). Des expérimentations sont menées sur des données réelles (Web logs pour la détection d'intrusions) et des données synthétiques, montrant l'intérêt de notre approche.

Les perspectives de ce travail sont très nombreuses. Nous pouvons notamment citer l'utilisation des expressions régulières (par exemple avec SPIRIT Garofalakis et al. (1999) pour la recherche des séquences inattendues. De plus, nous nous intéressons à la manière de générer automatiquement la base de croyance à partir de méthodes objectives. Enfin, nous souhaitons étendre la notion d'*inattendu* au processus général de fouille des séquences pour la mise à jour automatique de la base de croyance.

Références

Agrawal, R. et R. Srikant (1995). Mining sequential patterns. In *ICDE*, pp. 3–14.

Garofalakis, M. N., R. Rastogi, et K. Shim (1999). Spirit : Sequential pattern mining with regular expression constraints. In *VLDB*, pp. 223–234.

Masseglia, F., F. Cathala, et P. Poncelet (1998). The psp approach for mining sequential patterns. In *PKDD*, pp. 176–184.

McGarry, K. (2005). A survey of interestingness measures for knowledge discovery. *Knowl. Eng. Rev. 20*(1), 39–61.

Padmanabhan, B. et A. Tuzhilin (2006). On characterization and discovery of minimal unexpected patterns in rule discovery. *IEEE Trans. Knowl. Data Eng. 18*(2), 202–216.

Pei, J., J. Han, B. Mortazavi-Asl, J. Wang, H. Pinto, Q. Chen, U. Dayal, et M. Hsu (2004). Mining sequential patterns by pattern-growth : The prefixspan approach. *IEEE Trans. Knowl. Data Eng. 16*(11), 1424–1440.

Silberschatz, A. et A. Tuzhilin (1995). On subjective measures of interestingness in knowledge discovery. In *KDD*, pp. 275–281.

Spiliopoulou, M. (1999). Managing interesting rules in sequence mining. In *PKDD*, pp. 554–560.

Zaki, M. J. (2001). Spade : An efficient algorithm for mining frequent sequences. *Machine Learning 42*(1/2), 31–60.

Summary

Sequential pattern mining is the one most concentrated and applied in séquence mining research, and it gives a frequency based view of the correlations between elements contained in the séquences. However, when considering domain knowledge within the data mining process, the frequency based criterion become less interesting since most of the frequent séquences might have already been confirmed, and the most interesting séquences might not be the séquences corresponding to existing knowledge, but be the séquences contradicting existing knowledge that reflect unexpected behaviors.

In this paper we introduce the problem of finding unexpected behaviors within the context of séquence mining. We first give formal descriptions of belief system and unexpectedness of séquences, we then introduce unexpected sequential patterns and unexpectedness rules de-

picting unexpected behaviors within the séquences. We also propose the USER approach for mining unexpected sequential patterns and unexpectedness rules from a séquence database with respect to a belief system. Our experimental results show that both of the quantity and the quality of the unexpected séquences extracted by the USER approach are improved in comparison with the frequent séquences extracted by general sequential pattern mining approaches.