SPoID : Extraction de motifs séquentiels pour les bases de données incomplètes

Céline Fiot, Anne Laurent, Maguelonne Teisseire

Laboratoire d'Informatique de Robotique et de Microélectronique de Montpellier
{fiot, laurent, teisseire}@lirmm.fr

Résumé. Les bases de données issues du monde réel contiennent souvent de nombreuses informations non renseignées. Durant le processus d'extraction de connaissances dans les bases de données, une phase de traitement spécifique de ces données est souvent nécessaire, permettant de les supprimer ou de les compléter. Lors de l'extraction de séquences fréquentes, ces données incomplètes sont la plupart du temps occultées. Ceci conduit parfois à l'élimination de plus de la moitié de la base et l'information extraite n'est plus représentative. Nous proposons donc de ne plus éliminer les enregistrements incomplets, mais d'utiliser l'information partielle qu'ils contiennent. La méthode proposée ignore en fait temporairement certaines données incomplètes pour les séquences recherchées. Les expérimentations sur jeux de données synthétiques montrent la validité de notre proposition aussi bien en terme de qualité des motifs extraits que de robustesse aux valeurs manquantes.

1 Introduction

Les données issues du monde réel sont souvent entâchées d'imperfections. En particulier, il est très courant de disposer de nombreuses données incomplètes (pannes, erreur de format, oubli humain, ...). Or la présence de valeurs manquantes induit de très sérieux problèmes, les données contenant des valeurs manquantes étant souvent éliminées lors du processus de fouille de données. C'est notamment le cas pour l'extraction de motifs séquentiels. Cette technique de fouille de données, présentée comme une extension des règles d'association prenant en compte l'information temporelle des bases de données historisées, ne permet en effet que l'analyse des données complètes, sans tenir compte des enregistrements incomplets, ce qui constitue une grande perte d'information. Par ailleurs, les solutions de remplacement des valeurs manquantes sont souvent soit trop simplistes pour produire des résultats intéressants, soit trop coûteuses pour être mises en oeuvre sur de gros volumes de données.

Or, s'il existe à ce jour des techniques robustes aux valeurs manquantes pour l'extraction de règles d'association, il n'existe aucune méthode générique pour l'extraction de motifs séquentiels. En effet, dans le contexte de la recherche de motifs séquentiels, les valeurs manquantes n'ont pas été considérées jusqu'ici, l'application principale, les bases de données de supermarchés, n'en comportant quasiment jamais. Désormais, les motifs séquentiels sont utilisés afin d'extraire des connaissances d'applications industrielles (analyse de processus, web access logs, ...) qui contiennent inévitablement des données incomplètes.

Nous proposons donc ici une extension des principes décrits dans (Agrawal et Srikant, 1995) afin d'extraire des séquences fréquentes en présence de valeurs manquantes réparties aléatoirement dans une base de données. Cette approche a été inspirée par une méthode d'extraction de règles d'association sur des bases de données incomplètes (Ragel et Cremilleux, 1998) et sur une technique couramment utilisée en apprentissage (Liu et al., 1997) : ignorer les valeurs manquantes sans ignorer tout l'enregistrement associé. Le principe consiste à utiliser seulement l'information disponible (i.e. les attributs renseignés) et à ignorer les informations manquantes. Ainsi, seules des bases partielles complètes servent à l'extraction de chaque schéma fréquent et l'ensemble de la base est utilisée pour trouver tous les schémas. Nous transposons ici ce principe afin d'extraire des motifs séquentiels, des séquences fréquentes maximales sur des bases de données historisées incomplètes. Pour cela, nous avons adapté les notions utilisées pour l'extraction de séquences fréquentes. Puis, nous avons implémenté l'algorithme SPoID (Sequential Patterns over Incomplete Database) sur la base d'un algorithme d'extraction de motifs séquentiels, PSP (Masseglia et al., 1998). Cet algorithme a été testé sur des jeux de données synthétiques afin de montrer la validité de notre approche.

Dans la suite de cet article, section 2, nous présentons les méthodes qui permettent d'extraire des règles d'association en présence de valeurs manquantes, ainsi que les concepts liés à la découverte de motifs séquentiels. Dans la section 3, nous développons notre approche pour l'extraction de ces motifs à partir de bases de données incomplètes, puis nous présentons notre algorithme dans la section 4. La section 5 est ensuite consacrée aux expérimentations qui montrent la validité de notre approche. Enfin, nous concluons dans la section 6 par les perspectives qu'ouvrent ce travail.

2 Des données incomplètes aux motifs séquentiels

Les motifs séquentiels sont souvent présentés comme une extension des règles d'association, initialement proposées dans (Agrawal et al., 1993). Ils mettent en évidence des corrélations entre des enregistrements d'une base de données, ainsi que la relation temporelle qui existe entre eux. Toutefois ces algorithmes ne prennent pas en compte les enregistrements incomplets contenus dans une base de données. Afin de diminuer le prétraitement lié à la présence de valeurs manquantes et d'améliorer la qualité des motifs séquentiels extraits, nous proposons une technique d'extraction de motifs séquentiels dans des bases de données incomplètes, inspirées des méthodes existantes pour les règles d'association. Nous présentons dans cette section les méthodes qui permettent d'extraire des règles d'association en présence de valeurs manquantes ainsi que les notions liées à la découverte de motifs séquentiels.

2.1 Règles d'association et valeurs manquantes

Des travaux ont été proposés pour la recherche de règles d'association dans des bases de données incomplètes. Notamment, (Nayak et Cook, 2001; Ng et Lee, 1998), mettent en œuvre un système d'approximation probabiliste dans lequel une valeur manquante peut prendre plusieurs valeurs lors de la découverte des règles. Ces méthodes sont particulièrement adaptées aux bases de données relationnelles, mais ne sont pas facilement transposables au format spécifique des données dont on extrait les motifs séquentiels, détaillé section 3.

Nous avons donc choisi de nous inspirer de l'algorithme RAR (Robust Association Rules), proposé dans (Ragel et Cremilleux, 1998). Cette méthode, complètement compatible avec la méthode originelle (Agrawal et al., 1993), permet la prise en compte des données incomplètes lors de l'extraction de règles dans des bases de données relationnelles incomplètes, par omission partielle et temporaire de ces enregistrements. Le principe consiste à ne prendre en compte que les attributs renseignés pour les enregistrements incomplets. La base de données entière n'est pas utilisée pour chaque règle mais pour générer l'ensemble des règles. Cette technique repose sur la définition de bases de données valides, complètes pour un ensemble d'items données, le reste de la base étant momentanément ignoré.
Afin de prendre en compte ce partitionnement de la base, les concepts de support (pourcentage des enregistrements de la base qui contiennent tous les items de la règle) et de confiance (la probabilité qu'un enregistrement qui contient la partie gauche de la règle contienne également la partie droite) ont été redéfinis. Par ailleurs, une nouvelle notion est introduite afin de tenir compte de la taille de l'échantillon complet considéré pour déterminer le support de la règle. Cette mesure de représentativité permet ainsi d'éliminer de la liste des règles celles trouvées sur une base peu significative par rapport à la base initiale.

La recherche de motifs séquentiels consiste à extraire des ensembles d'items couramment associés sur une période de temps bien spécifiée. Elle permet de mettre en évidence des associations inter-enregistrement, par rapport à celle de règles d'association qui extrait des combinaisons intra-enregistrement. L'identification des individus ou objets est alors indispensable afin de pouvoir suivre leur comportement au cours du temps.

2.2 Motifs séquentiels

Les motifs séquentiels ont initialement été proposés par (Agrawal et Srikant, 1995) et reposent sur la notion de *séquence fréquente maximale*.

Considérons une base de données DB d'achats pour un ensemble $\mathcal{O}$ d'objets o. Chaque enregistrement R correspond à un triplet *(id-objet, id-date, itemset)* qui caractérise l'objet auquel est rattaché l'enregistrement, ainsi que la date et les *items* correspondants.
Soit $\mathcal{I} = \{i_1, i_2, \cdots, i_m\}$ l'ensemble des *items* de la base. Un *itemset* est un ensemble non vide et non ordonné d'items, noté $(i_1, i_2, \ldots, i_k)$, où i_j est un *item*. Une *séquence* s se définit alors comme une liste ordonnée non vide d'itemsets qui sera notée $< s_1 s_2 \cdots s_p >$, où s_j est un itemset. Une n-séquence est une séquence de taille n, c'est-à-dire composée de n items.

Exemple 1. *Si un client achète les produits a, b, c, d et e selon la séquence S=<(a)(b c)(d)(e)>, cela signifie qu'il a d'abord acheté le produit a, puis les produits b et c ensemble, ensuite seulement le produit d et finalement le produit e. S est une 5-séquence.*

Une séquence $S' =< s'_1 s'_2 \cdots s'_m >$ est une *sous-séquence* de $S =< s_1 s_2 \cdots s_p >$ s'il existe des entiers $a_1 < a_2 < \cdots < a_m$ tels que $s'_1 \subseteq s_{a_1}, s'_2 \subseteq s_{a_2}, \cdots, s'_m \subseteq s_{a_m}$, S' est *incluse* dans S.

Exemple 2. *La séquence $S' =< (b)(e) >$ est une sous-séquence de S car $(b) \subseteq (b\ c)$ et $(e) \subseteq (e)$. Par contre $< (b)(c) >$ n'est pas une sous-séquence de $< (b\ c) >$, ni l'inverse.*

Les enregistrements de la base sont regroupés par objet et ordonnés chronologiquement, définissant ainsi des *séquences de données*. Un objet o *supporte* une séquence S si elle est incluse dans la séquence de données o. Le *support* (ou fréquence) d'une séquence est alors défini

comme le pourcentage d'objets de la base DB qui supporte S. Une séquence est dite *fréquente* si son support est au moins égal à une valeur minimale $minSup$ spécifiée par l'utilisateur. Une *séquence candidate* est une séquence potentiellement fréquente.

La recherche de motifs séquentiels dans une base de séquences telle que DB consiste alors à trouver toutes les séquences maximales (non incluses dans d'autres) dont le support est supérieur à $minSup$. Chacune de ces séquences fréquentes maximales est un *motif séquentiel*.

Des extensions ont été proposées pour prendre en compte la recherche incrémentale de motifs séquentiels (Masseglia et al., 2000), la gestion de valeurs numériques associées aux items (Hong et al., 2001; Chen et al., 2001; Fiot et al., 2005) ou encore la généralisation des motifs séquentiels pour différents paramètres temporels (espacement des différents évènements d'une séquence, rapprochement d'évènements proches en une même date...) (Srikant et Agrawal, 1996; Masseglia et al., 1999; Fiot et al., 2006). Toutefois, il n'existe pas, à notre connaissance, de techniques permettant de gérer les valeurs manquantes lors de la découverte de motifs séquentiels. C'est pourquoi nous proposons ici une approche permettant d'extraire des séquences fréquentes maximales dans une base de séquences incomplètes.

3 SPoID : une nouvelle approche du traitement des données incomplètes

3.1 Motivations

Nous souhaitons extraire les motifs séquentiels contenus dans la base de données TAB. 1.

Obj.	Séquence
O1	$(a\ b)\ (b\ c\ d)\ (b\ c\ e)$
O2	$(a)\ (b\ c)\ (b\ d)$
O3	$(a\ b)\ (b\ c)\ (b\ c\ d)$

TAB. 1 – *Base de données complète.*

Obj.	Séquence
O1	$(a\ b)\ (?\ ?\ c)\ (?\ b\ c)$
O2	$(a)\ (?\ c)\ (b\ d)$
O3	$(a\ b)\ (?\ c)\ (?\ ?\ c)$

TAB. 2 – *Base de données incomplète.*

Obj.	Séquence
O1	$(a\ b)\ (c)\ (b\ c)$
O2	$(a)\ (c)\ (b\ d)$
O3	$(a\ b)\ (c)\ (c)$

TAB. 3 – *Après suppression des valeurs manquantes.*

Avec un support de 50%, les motifs obtenus sont : <(a b)(b c)(b c)>, <(a b)(b c d)> et <(a)(b c)(b d)>.

Considérons maintenant la même base, mais incomplète, TAB. 2. Pour certaines séquences de données, les informations n'ont pas été transmises et des valeurs sont manquantes. On considère que ces valeurs sont identifiées comme des attributs de valeur indéterminée, contrairement à des attributs inexistants où la valeur sera identifiée comme non-renseignée. Afin de pouvoir extraire les motifs, les méthodes classiques requièrent une élimination des valeurs manquantes. La base sur laquelle s'effectue la fouille de données est alors la base TAB. 3 et les motifs obtenus pour $minSup$ = 50% sont : <(a b)(c)(c)> et <(a)(c)(b)>.

On constate que seule une petite partie de la base est utilisée pour extraire l'information et on ne retrouve qu'une partie des schémas fréquents extraits de la base complète : des sous-séquences des motifs que l'on devrait extraire. C'est pourquoi il paraît nécessaire d'utiliser l'intégralité de la base lors de la fouille et non pas d'en supprimer une partie.

3.2 Sequential Patterns over Incomplete Database (SPoID)

L'élimination des enregistrements incomplets conduisant à une perte d'information, nous avons donc envisagé d'adapter une méthode d'extraction de règles d'association robuste aux valeurs manquantes pour extraire des motifs séquentiels. Nous présentons ici la méthode SPoID (Sequential Patterns over Incomplete Database), inspirée de l'algorithme RAR présenté dans (Ragel et Cremilleux, 1998).

Le principe général de notre méthode, comme de la méthode RAR, repose sur la désactivation des éléments incomplets, dans notre cas, des séquences. Alors que pour les règles d'association, l'algorithme RAR ne considère que les enregistrements complets, nous proposons ici de ne prendre en compte que les séquences de données complètes pour la séquence candidate recherchée. Autrement dit, on ne considèrera que les dates et attributs renseignés pour les séquences incomplètes. Ainsi pour chaque séquence on déterminera si elle est fréquente sur une base partielle, mais la totalité de la base sera utilisée pour l'ensemble des séquences fréquentes.

Considérant une séquence candidate S, l'ensemble $\mathcal{O}$ des objets de la base peut être divisé en trois sous-ensembles disjoints (FIG. 1) :

- l'ensemble des séquences de données qui supportent S, noté $\mathcal{O}_S$,
- l'ensemble des séquences de données qui ne supportent pas S, noté $\mathcal{O}_{\overline{S}}$,
- l'ensemble des séquences de données pour lesquelles on ne sait pas si elles supportent S ou non, noté $\mathcal{O}_S^*$.

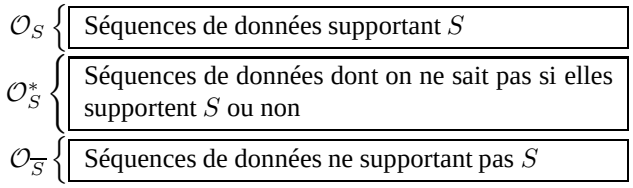

FIG. 1 – *Découpage de la base de données selon l'inclusion de S*

Pour chaque séquence candidate S, on ne considèrera donc que le sous-ensembles $\mathcal{O}_{\overline{S}} \cup \mathcal{O}_S$ afin de déterminer si la séquence S est fréquente ou non. Ce sous-ensemble de séquences de données forme la *base de données valide* pour S.

Définition 1. *Une* base de données valide *est une base de séquences complètes pour une séquence candidate donnée.*

La construction d'une base de données valide repose sur la *désactivation* temporaire des séquences de données qui contiennent des valeurs manquantes pour les items présents dans la séquence candidate. Cette désactivation implique une redéfinition de la notion de support, pour prendre en compte la désactivation d'une partie de la base.

Définition 2. *Une séquence de données est* désactivée *pour une séquence candidate S si elle est incomplète pour S (i.e. on ne sait pas si elle supporte S ou non). On note $Dis(S)$ l'ensemble des séquences désactivées pour la séquence candidate S.*

La notion de *support* définie dans la section 3 doit donc être modifiée afin de prendre en compte la notion de base de données valide et pour considérer le fait que seule une partie de la base est utilisée.

Définition 3. *Le* support *d'une séquence S est le taux d'apparition de cette séquence parmi les séquences de données qui peuvent la supporter. On le définit comme le ratio du nombre de séquences de données qui supportent cette séquence, par le nombre de séquences de données dont on est sûr qu'elles pourraient ou non inclure cette séquence (complètes pour cette séquence). Il est donné par la formule :*

$$Supp(S) = \frac{|\mathcal{O}_S|}{|\mathcal{O}| - |Dis(S)|}$$

Nous avons montré, sous certaines conditions peu restrictives, que cette définition respecte la propriété d'antimonotonie du support énoncée dans (Agrawal et Srikant, 1995).

La nouvelle définition du support étant antimonotone, on peut utiliser les différentes propriétés énoncées dans (Agrawal et Srikant, 1995) afin de réaliser l'extraction de motifs séquentiels sur base de données incomplètes. Toutefois, la notion de *fréquence* dépend du calcul du *support* et de la taille de la base de données valide utilisée pour le calculer. On définit un critère de *représentativité* minimale : une base de données valide doit être un échantillon significatif de la base de départ pour qu'on considère la séquence comme fréquente si elle valide le support minimal $minSup$.

Définition 4. *La* représentativité $Rep(S)$ *d'une séquence S est définie comme le ratio du nombre de fois où cette séquence apparaît complète ou n'apparaît pas avec certitude dans la base, par rapport au nombre total de séquences de données dans la base. Elle est donnée par :*

$$Rep(S) = \frac{|\mathcal{O}| - |Dis(S)|}{|\mathcal{O}|}$$

Définition 5. *Une séquence est* représentative *si sa représentativité est supérieure à une valeur minimale donnée.*

Pour être considérée comme fréquente, une séquence doit donc être trouvée fréquente sur une base de données valide et représentative, c'est-à-dire si sa représentativité est supérieure au seuil minimal de représentativité $minRep$ et si son support est supérieur au support minimum spécifié par l'utilisateur $minSup$.

3.3 Seuil de représentativité et marge d'erreur

Les statisticiens disposent d'outils d'échantillonnage permettant de considérer un sous-ensemble d'une population afin d'estimer une proportion, à un pourcentage d'erreur près, avec un niveau de confiance suffisant. Ces outils permettent de déterminer la taille optimale d'un échantillon en fonction de la distribution des données. Ainsi, dans le cas d'une distribution au hasard des données, (Toivonen, 1996) utilise les bornes de Chernoff pour déterminer la taille minimale d'un échantillon tiré au hasard pour l'extraction de règles d'association. Ce résultat est également démontré théoriquement et empiriquement dans (Zaki et al., 1996).

Nous proposons donc d'utiliser deux formes de représentativité selon le souhait de l'utilisateur : soit celle-ci sera définie comme une proportion de la base, soit elle sera absolue et calculée à partir de formules statistiques dépendant de la distribution des données, en respectant un pourcentage d'erreur et un niveau de confiance spécifiés par l'utilisateur. Les expérimentations montrent toutefois que le seuil de représentativité optimale n'est pas absolu mais dépend en fait du taux de valeurs manquantes contenues dans la base de données.

4 Mise en œuvre

4.1 Illustration

Reprenons la base incomplète présentée TAB. 2. On pose $minSup$=50%, puis on calcule le support et la représentativité de chacun des items de la base pour déterminer les items fréquents. L'item a est supporté de manière sûre par les trois objets son support est $supp(a) = 3/3 = 100\%$ et sa représentativité vaut 1. Il en est de même pour les items b et c.
Pour l'item d, on a $\mathcal{O}_{<d>}$ = $\{O2\}$ et $Dis(< d >)$ = $\{O1, O3\}$, donc $supp(d)$ = 1/(3-2) = 1 et $rep(d)$ = (3-2)/3 = 0.33. Si le seuil $minRep$ vaut 0.3, alors $rep(d) > minRep$ et d est un item fréquent. Par contre, si $minRep$ = 0.4, alors $rep(d) < minRep$ et d n'est pas un item fréquent car la base valide qui permet de calculer son support n'est pas suffisamment représentative.

On fixe $minRep$=0.3. Prenons maintenant la séquence $S =< (a\ b)(a\ b\ c\ d) >$. Cette séquence ne peut être supportée par aucune des trois séquences de données, car aucune ne comporte d'itemsets de 4 items, complet ou non, donc $\mathcal{O}_S$ = $\{\}$, $Dis(S)$ = $\{\}$ et $\mathcal{O}_{\overline{S}}$ = $\{O1, O2, O3\}$ et $supp(S)$=0. Cette séquence n'est pas fréquente.
La séquence $S' =< (a\ b)(b\ c) >$, quant à elle, est supportée par $O1$, ne peut l'être par $O2$ mais pourrait l'être par $O3$. On a donc $\mathcal{O}_{S'}$ = $\{O1\}$, $Dis(S')$ = $\{O3\}$ et $\mathcal{O}_{\overline{S'}}$ = $\{O2\}$, ce qui donne $supp(S')$=1/(3-1)=50% et rep(S')=(3-1)/3 = 0.67. S' est donc représentative et fréquente.

En appliquant la méthode ci-dessus, les motifs extraits pour $minSup$=50% et $minRep$ = 0.3 sont : <(a b)(c)(b c)> et <(a)(c)(b d)>. Même si ces motifs ne sont pas exactement ceux extraits sur la base complète, on constate qu'ils sont plus proches des motifs qui devraient être extraits des motifs obtenus de la base après prétraitement.
Les expérimentations présentées dans la section 5 montrent qu'il existe une valeur de la représentativité à partir de laquelle l'algorithme SPoID extrait d'une base incomplète l'intégralité des motifs extraits sur la base complète.

4.2 Algorithme

Le principe de l'algorithme SPoID est similaire aux algorithmes d'extraction de motifs séquentiels de type générer-élaguer. Il consiste à générer toutes les séquences candidates de longueur k à partir des séquences fréquentes de longueur k-1, puis à scanner l'ensemble de la base pour compter le nombre de séquence de données qui supporte chacune des séquences candidates. La différence réside dans le dénombrement des séquences de données incomplètes pour la séquence candidate considérée. La phase de comptage est décrite par l'algorithme ALG. 1 : pour chaque séquence candidate, pour chaque objet,

– si on trouve la séquence candidate, on incrémente les valeurs absolues du support,

- si on ne trouve pas la séquence candidate ni de séquence dans laquelle des valeurs manquantes pourraient être remplacées par les items correspondant dans la séquence candidate, alors l'objet ne supporte pas la séquence candidate. On n'incrémente pas le support.
- on trouve une séquence incomplète dans laquelle des valeurs manquantes pourraient être remplacées par les items correspondant dans la séquence candidate. Dans ce cas, on ajoute cet objet à l'ensemble des objets désactivés.

Une fois l'ensemble de la base parcouru, on divise la valeur absolue du support par la différence entre le nombre total d'objets et le nombre d'objets désactivés, on calcule la représentativité. Puis on procède à la phase d'élagage en supprimant toutes les séquences candidates qui ne sont pas fréquentes puis celles qui ne sont pas représentatives.

SPoID - **Input :** $|\mathcal{O}|$, une base de séquences de données, *minSup*, support minimum spécifié par l'utilisateur
minRep, représentativité minimum, spécifiée ou calculée
Ouput : $SPList$, liste des séquences fréquente

$C \leftarrow \{i \in \mathcal{I}\}$; k = 1 ;
$F \leftarrow$ getFrepnRep(C,$minSup$,$minRep$) ;
$SPList$.add(F) ;
While ($C \neq \varnothing$) **do**
  k++ ; $C \leftarrow$ generate(F,k) ;
  For chaque séquence candidate $s \in C$ **do**
    For chaque objet $o \in \mathcal{O}$ **do**
      [On cherche s dans S_o]
      If ($s \in S_o$) **Then**
        $supp(s)$++ ; $Dis(s) \leftarrow Dis(s) \backslash o$;
      Else
        If ($\tilde{s} \in S_o / \tilde{s}$ pourrait être s) **Then**
          $Dis(s) \leftarrow Dis(s) \cup o$;
        End If
      End If
    End For
    $Support(s) \leftarrow supp(s)/(|\mathcal{O}| - |Dis(s)|)$; $Rep(s) \leftarrow |\mathcal{O}| - |Dis(s)|/|\mathcal{O}|$;
    If (($Support(s) < minSup$)||($Rep(s) < minRep$)) **Then**
      prune(s) ;
    End If
  End For
End While
return SPList ;

ALG. 1 – *SPoID - algorithme général.*

La complexité temporelle de cet algorithme est, dans le pire des cas, la même que celle de l'algorithme TOTALLYFUZZY présenté dans (Fiot et al., 2005). On utilise le même type d'optimisations afin de réduire le nombre de passes sur la base. Par contre, la complexité en mémoire est nettement moindre puisqu'elle est du même ordre que celle de PSP.

5 Expérimentations

Ces expérimentations ont été réalisées sur un PC équipé d'un processeur 2,8GHz et de 512Mo de mémoire DDR, sous système Linux, noyau 2.6. Nous utilisons un jeu de données synthétiques générés aléatoirement par une loi normale dans lequel nous remplaçons certains

items par des valeurs manquantes, réparties de maniére aléatoire. On extrait les motifs séquentiels sur la base complète ainsi que sur la base prétraitée (dont les itemsets incomplets ont été supprimés). Puis on compare ces motifs extraits par les méthodes existantes aux motifs extraits par notre algorithme SPoID. Les résultats présentés ici ont été obtenus à partir du traitement de plusieurs jeux de données synthétiques comportant environ 2000 séquences de 20 transactions en moyenne. Chacune de ces transactions comporte en moyenne 10 items choisis parmi 100.

Nos analyses sont basées sur le calcul du nombre de bons motifs séquentiels trouvés par SPoID et du nombre de motifs différents extraits par SPoID, ces derniers regroupant les motifs extraits, qui n'existe pas dans la base complète et les motifs non trouvés, mais contenus dans la base complète. Le tableau 4 récapitule l'ensemble de ces notations.

β	Nombre de motifs extraits par SPoID, contenus dans la base complète
δ	Nombre de motifs différents
θ	Nombre de motifs extraits par SPoID sur la base incomplète
τ	Nombre de motifs extraits sur la base de données complète

TAB. 4 – *Notations pour les différentes catégories de motifs séquentiels extraits.*

La FIG. 2(a), tout d'abord, montre l'évolution du rapport β/θ, en fonction de la représentativité minimale. On constate que ce taux croît à mesure que $minRep$ augmente, ce qui signifie que parmi les motifs extraits par SPoID, la proportion des motifs également trouvés dans la base complète augmente avec la représentativité minimale.

On peut compléter cette observation par l'analyse de la FIG. 2(b), qui présente l'évolution du rapport β/τ (nombre de motifs extraits par rapport aux motifs à extraire) en fonction de la représentativité minimale. On constate que ce ratio diminue à mesure que $minRep$ augmente, ce qui signifie qu'il est nécessaire de choisir une représentativité suffisamment faible pour permettre l'extraction de l'intégralité des motifs présents dans la base complète.

Nous avons donc mis en évidence l'existence d'une valeur optimale du seuil de représentativité, pour laquelle les ratios β/θ (bons motifs/motifs extraits par SPoID) et β/τ sont les plus proches possibles de 1. Cette valeur correspond au seuil pour lequel le nombre de bons motifs extraits sur la base incomplète est le plus élevé possible par rapport au nombre de motifs différents. La FIG. 2(c) met en évidence l'existence de cette valeur optimale de $minRep$. Ce graphe montre l'évolution du rapport (β/δ), rapport du nombre de bons motifs extraits par SPoID et du nombre de motifs qui diffèrent des motifs extraits sur la base complète (manquants + supplémentaires). On constate qu'il n'existe pas de valeur absolue de la repésentativité minimale, commune à toutes les bases et dépendant d'une marge d'erreur donnée. D'après ces résultats, la représentativité minimale dépend uniquement du taux d'incomplétude de la base.

Quelle que soit la proportion de valeurs manquantes dans la base incomplète, l'allure générale de la courbe est la même : le ratio (β/δ) augmente avant d'atteindre un maximum puis de décroître. Ce point maximal correspond à la représentativité optimale, pour laquelle le nombre de bons motifs extraits par SPoID est le plus élevé et le nombre de motifs différents le plus faible. Le tableau TAB. 5 donne la valeur de la représentativité optimale trouvée expérimentalement pour chaque proportion de données incomplètes dans notre jeu de test.

La FIG. 2(c) met aussi en évidence l'évolution de comportement de l'algorithme SPoID selon le taux de valeurs manquantes dans la base. On constate une différence entre l'allure de la courbe et la valeur du ratio β/δ pour les bases incomplètes contenant 40% de valeurs man-

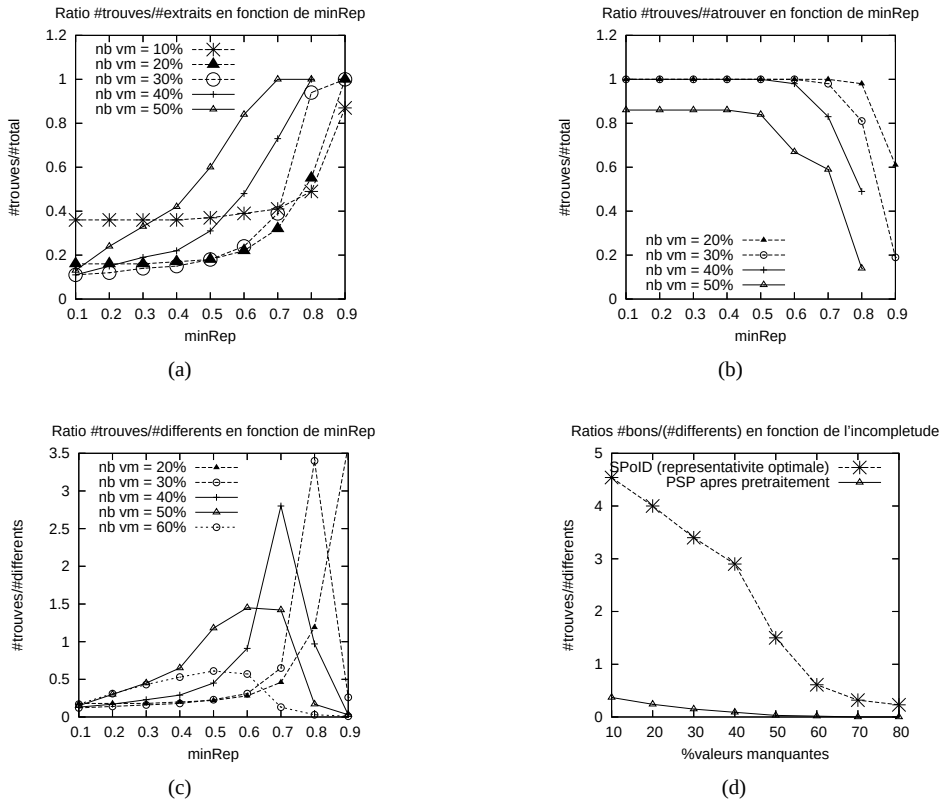

FIG. 2 – *(a) : Proportion (bons motifs)/(motifs extraits) (β/θ) en fonction de la représentativité minimale ; (b) : Proportion (bons motifs)/(motifs à trouver) (β/τ) en fonction de la représentativité minimale ; (c) : Proportion (bons motifs)/(motifs différents) (β/δ) en fonction de la représentativité minimale ; (d) : Proportion (bons motifs)/(motifs différents) (β/δ) en fonction du pourcentage de valeurs manquantes dans la base.*

% de valeurs manquantes	10%	20%	30%	40%	50%	60%	70%	80%
$minRep$ optimale	0.97	0.9	0.81	0.74	0.6	0.48	0.39	0.22

TAB. 5 – *Moyenne des représentativités optimales selon la proportion de valeurs manquantes dans la base de données.*

quantes et moins et celles qui comportent 50% d'enregistrements incomplets ou plus.
Ainsi, la FIG. 2(d) permet de comparer le taux de réussite de SPoID et d'une extraction sur données préparées (base incomplète dans laquelle les données ont été supprimées). Cette figure permet de mettre en évidence qu'un certain nombre de bons motifs, extraits par SPoID, ne sont pas trouvés par un traitement classique précédé d'une préparation des données, en montrant

l'évolution du ratio optimal de bons motifs trouvés par SPoID selon le pourcentage d'incomplétude de la base. On constate que ce taux chute rapidement lorsque l'on passe de 40 à 50% de valeurs manquantes : le nombre de motifs différents devient proportionnellement plus important par rapport au nombre de bons motifs.
On remarque également que ce ratio devient inférieur à 1, lorsque le pourcentage de valeurs manquantes dépasse 50%. SPoID permet donc d'extraire les motifs séquentiels d'une base incomplète tant que plus de la moitié de ses enregistrements sont complets alors qu'un algorithme classique ne permet pas de trouver tous les motifs fréquents dès 10% de valeurs manquantes. Lorsque le taux d'incomplétude atteint 40%, le nombre de bons motifs extraits est même quasi nul, alors qu'il reste encore relativement élevé pour SPoID.

6 Perspectives

La découverte de motifs séquentiels est une méthode de fouille de données intéressante lorsqu'il s'agit d'extraire des connaissances dans une base de données historisée, telle que des relevés de processus industriel ou de fonctionnement de machines. Or, dans ce type de bases de séquences, la présence de valeurs manquantes est inévitable. Pourtant il n'existe aucune technique permettant de découvrir des séquences fréquentes à partir de bases de données incomplètes. Nous avons donc proposé dans cet article une adaptation des définitions originales liées à l'extraction de motifs séquentiels afin de pouvoir traiter les informations incomplètes distribuées au hasard, directement pendant la fouille plutôt que de supprimer ces enregistrements, comme cela était le cas avec les algorithmes existants. Notre méthode, SPoID (Sequential Patterns over Incomplete Database) a été implémentée et testée sur des jeux de données synthétiques, ce qui nous a permis de montrer sa robustesse aux valeurs manquantes jusqu'à un taux d'incomplétude d'environ 40%, alors que les méthodes classiques, après prétraitement, donnent de mauvais résultats dès 10% de valeurs manquantes.

Nous envisageons maintenant d'étendre cette méthode afin de pouvoir prendre en compte d'autres types de valeurs manquantes (non distribuées au hasard, par exemple), après avoir détecté les différents types d'informations incomplètes contenues dans la base. Il apparaît également nécessaire de pouvoir distinguer, quand cela ne l'a pas été fait au préalable, les valeurs d'attribut inexistant, qui n'ont donc pas à être considérées comme manquantes. Enfin, le bruit est également une imperfection courante dans les bases de données du monde réel. Il pourra donc être intéressant de le prendre en compte dans une version ultérieure de notre algorithme.

Références

Agrawal, R., T. Imielinski, et A. N. Swami (1993). Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, pp. 207–216. Peter Buneman and Sushil Jajodia.

Agrawal, R. et R. Srikant (1995). Mining sequential patterns. In *Eleventh International Conference on Data Engineering*, Taipei, Taiwan, pp. 3–14. IEEE Computer Society Press.

Chen, R.-S., G.-H. Tzeng, C.-C. Chen, et Y.-C. Hu (2001). Discovery of Fuzzy Sequential Patterns for Fuzzy Partitions in Quantitative Attributes. In *ACS/IEEE Int. Conf. on Computer Systems and Applications*, pp. 144–150.

Fiot, C., A. Laurent, et M. Teisseire (2005). Motifs séquentiels flous : un peu, beaucoup, passionnément. In *5èmes journées d'Extraction et Gestion des Connaissances (EGC '05)*, pp. 507–519.

Fiot, C., A. Laurent, et M. Teisseire (2006). Des motifs séquentiels généralisés aux contraintes de temps étendues. In *6èmes journées d'Extraction et Gestion des Connaissances*, pp. 603–614.

Hong, T., K. Lin, et S. Wang (2001). Mining Fuzzy Sequential Patterns from Multiple-Items Transactions. In *Joint 9th IFSA World Congress and 20th NAFIPS Int. Conf.*, pp. 1317–1321.

Liu, W. Z., A. P. White, S. G. Thompson, et M. A. Bramer (1997). Techniques for dealing with missing values in classification. *Lecture Notes in Computer Science 1280*, 527– ? ?

Masseglia, F., F. Cathala, et P. Poncelet (1998). The PSP Approach for Mining Sequential Patterns. In *Principles of Data Mining and Knowledge Discovery*, pp. 176–184.

Masseglia, F., P. Poncelet, et M. Teisseire (1999). Extraction efficace de motifs séquentiels généralisés : le prétraitement des données. In *15 ème Journée Bases de Données Avancées (BDA '99)*, pp. 341–360.

Masseglia, F., P. Poncelet, et M. Teisseire (2000). Incremental mining of sequential patterns in large databases. Technical report, LIRMM, France.

Nayak, J. et D. Cook (2001). Approximate association rule mining. In *Florida Artificial Intelligence Research Symposium*.

Ng, V. et J. Lee (1998). Quantitative association rules over incomplete data. In *IEEE International Conference*, pp. 2821–2826.

Ragel, A. et B. Cremilleux (1998). Treatment of missing values for association rules. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 258–270.

Srikant, R. et R. Agrawal (1996). Mining sequential patterns : Generalizations and performance improvements. In *5th International Conference on Extending Database Technology (EDBT '96)*, London, UK, pp. 3–17. Springer-Verlag.

Toivonen, H. (1996). Sampling large databases for association rules. In *VLDB '96 : Proceedings of the 22th International Conference on Very Large Data Bases*, pp. 134–145.

Zaki, M. J., S. Parthasarathy, W. Li, et M. Ogihara (1996). Evaluation of sampling for data mining of association rules. Technical report, Rochester, NY, USA.

Summary

Industrial databases often contains a large amount of unfilled information. During the knowledge discovery process one specific processing step is often necessary in order to remove these incomplete data either by deleting them or by assessing them. When the data mining task consists in mining frequent sequences, incomplete data are, most of the time, deleted, which leads to an important loss of information. Extracted knowledge becomes so less representative of the whole database. We thus propose a method that uses the partial information contained into incomplete records, only ignoring missing values when they may imply errors in the extracted sequential patterns.