Extraction de motifs fermés dans des relations n-aires bruitées

Loïc Cerf, Jérémy Besson, Jean-François Boulicaut

Université de Lyon, CNRS
INSA-Lyon, LIRIS UMR5205, F-69621 Villeurbanne, France
Prénom.Nom@liris.cnrs.fr

Résumé. L'extraction de motifs fermés dans des relations binaires a été très étudiée. Cependant, de nombreuses relations intéressantes sont n-aires avec $n > 2$ et bruitées (nécessité d'une tolérance aux exceptions). Récemment, ces deux problèmes ont été traités indépendamment. Nous introduisons notre proposition pour combiner de telles fonctionnalités au sein d'un même algorithme.

1 Introduction

La fouille de relations binaires a été très étudiée via notamment les usages multiples des ensembles fermés fréquents. Cependant, il est courant que les données à traiter se représentent dans des relations n-aires avec $n \geq 3$ et il semble donc naturel de vouloir étendre le calcul de motifs fermés dans ce contexte (Ji et al., 2006; Jaschke et al., 2006; Cerf et al., 2008b). Dans le cas des relations binaires (calcul de 2-ensembles fermés ou concepts formels selon (Ganter et al., 2005)), nous savons que le nombre et la qualité des motifs extraits sont déjà problématiques. De nombreuses raisons (e.g., une erreur de mesure) peuvent mener à l'absence d'un couple dans la relation et un « véritable » motif donne lieu à plusieurs motifs fermés distincts et plus petits : quand la quantité de bruit augmente, le nombre de motifs fermés explose et leur pertinence se dégrade. Cette situation empire dramatiquement lorsque l'arité de la relation à fouiller augmente. Nous introduisons ici un algorithme de calcul de tous les motifs fermés ayant un nombre borné d'exceptions par élément (de n'importe quel attribut) sur n'importe quelle relation n-aire. Cet article est une version courte de (Cerf et al., 2008a).

2 Notion de ET-n-ensemble fermé

Soit $D^1, \ldots, D^n$ les domaines de n attributs. Soit $\mathcal{R}$ une relation n-aire sur ces attributs, i.e., $\mathcal{R} \subseteq D^1 \times \cdots \times D^n$. Appelons X un motif $\langle X^1, \ldots, X^n \rangle \in 2^{D^1} \times \cdots \times 2^{D^n}$. $\forall i = 1 \ldots n, \forall e \in D^i$, l'hyper-plan de X sur e, noté $H(X, e)$, est $\langle X^1, \ldots, \{e\}, \ldots, X^n \rangle$. $0(X)$ est le nombre de n-uplets de X qui sont absents de $\mathcal{R}$, i.e., $|(X^1 \times \cdots \times X^n) \setminus \mathcal{R}|$. Un n-ensemble fermé de $\mathcal{R}$ désigne l'extension naturelle de la notion de concept formel aux relations n-aires quand $n > 2$. Un n-ensemble fermé satisfait deux contraintes, la contrainte dite de connexion et celle dite de fermeture. $X = \langle X^1, \ldots, X^n \rangle$ vérifie la contrainte de connexion notée $\mathcal{C}_{\text{cx}}$ ssi $\forall i = 1 \ldots n, \forall e \in X^i, 0(H(X, e)) = 0$. X est dit fermé, i.e. satisfait la contrainte

	1	2	3	1	2	3	1	2	3
A	1	1	1	1	1	1			1
B	1	1		1			1	1	1
C		1		1	1	1	1		1
D		1		1	1		1	1	1
	α			β			γ		

TAB. 1 – *La relation* $\mathcal{R}_E \subseteq \{A, B, C, D\} \times \{1, 2, 3\} \times \{\alpha, \beta, \gamma\}$

notée $\mathcal{C}_{\text{fm}}$, ssi $\forall i = 1 \dots n$, $\forall e \in D^i \setminus X^i$, $0(H(X, e)) > 0$. X est un n-ensemble fermé ssi $\mathcal{C}_{\text{cx}}(X) \wedge \mathcal{C}_{\text{fm}}(X)$.

Les n-ensembles fermés sont mal adaptés à la découverte de connaissance dans des relations bruitées. En effet, la contrainte de connexion est, en pratique, trop forte et nous proposons de l'affaiblir. Soit $(\epsilon^i)_{i=1\dots n} \in \mathbb{N}^n$ des seuils de tolérance aux exceptions. X vérifie la nouvelle contrainte de connexion $\mathcal{C}_{\text{ET-cx}}$ ssi $\forall i = 1 \dots n$, $\forall e \in X^i$, $0(H(X, e)) \leq \epsilon^i$. Etant donné un motif, le seuil de tolérance aux exceptions ϵ^i est, sur n'importe quel élément (à l'intérieur du motif) du $i^{\text{ème}}$ attribut, le nombre maximal de n-uplets « autorisés » à être absents de la relation. Lorsque $\forall i = 1 \dots n, \epsilon^i = 0$, on a $\mathcal{C}_{\text{ET-cx}} \equiv \mathcal{C}_{\text{cx}}$. Pour définir un ET-$n$-ensemble fermé, l'utilisation de $\mathcal{C}_{\text{ET-cx}}$ nous conduit à renforcer la contrainte de fermeture. X satisfait la nouvelle contrainte $\mathcal{C}_{\text{ET-fm}}$ et est dit fermé ssi $\forall i = 1 \dots n$, $\forall e \in D^i \setminus X^i$, $0(H(X, e)) > \epsilon^i$. Lorsque $\forall i = 1 \dots n, \epsilon^i = 0$, on a $\mathcal{C}_{\text{ET-fm}} \equiv \mathcal{C}_{\text{fm}}$. X est un ET-n-ensemble fermé ssi $\mathcal{C}_{\text{ET-cx}}(X) \wedge \mathcal{C}_{\text{ET-fm}}(X)$. Il s'agit donc de la généralisation d'un n-ensemble fermé tolérant plus ou moins d'exceptions (en fonction des seuils $(\epsilon^i)_{i=1\dots n}$).

Exemple 1 *La Table 1 montre une relation ternaire notée $\mathcal{R}_E$. Chaque '1' indique la présence dans $\mathcal{R}_E$ du triplet correspondant. Des exemples de 3-ensembles fermés dans $\mathcal{R}_E$ sont $\langle (A), (1, 2, 3), (\alpha, \beta) \rangle$, $\langle (B), (1, 2), (\alpha, \gamma) \rangle$ et $\langle (C), (1, 3), (\beta, \gamma) \rangle$. Avec $\epsilon^1 = \epsilon^2 = \epsilon^3 = 1$, $X_E = \langle (A, B), (1), (\alpha, \beta, \gamma) \rangle$ est un exemple de ET-3-ensemble fermé dans $\mathcal{R}_E$. En effet, aucun de ses hyper-plans ne contient plus de 1 triplet absent ($0(H(X_E, A)) = 1$, $0(H(X_E, B)) = 0$, $0(H(X_E, 1)) = 1$, $0(H(X_E, \alpha)) = 0$, $0(H(X_E, \beta)) = 0$ et $0(H(X_E, \gamma)) = 1$) et il est fermé, i.e., quelque soit l'élément que l'on ajouterait, la contrainte $\mathcal{C}_{\text{ET-cx}}$ serait violée (e.g., si on lui ajoutait D on aurait $0(H(X_E, 1)) = 2$).*

3 Exploiter les contraintes $\mathcal{C}_{\text{ET-cx}}$ et $\mathcal{C}_{\text{ET-fm}}$

Nous présentons les mécanismes de calcul efficace des ET-n-ensembles fermés. Nous adoptons le principe d'un arbre d'énumération tel qu'il a été utilisé, par exemple, dans (Cerf et al., 2008b). Il s'agit d'un arbre binaire où, à chaque nœud, un élément e est énuméré : tous les motifs qui seront issus du fils gauche *contiendront* e alors que tous les motifs qui seront issus du fils droit *ne* le contiendront *pas*. Si l'on peut être certain qu'une contrainte ne sera jamais vérifiée par les motifs issus du nœud courant, on pourra procéder à un *élagage*. Nous exploitons la contrainte de fermeture $\mathcal{C}_{\text{ET-fm}}$ de cette manière puisqu'elle est *anti-monotone*. Cela signifie que si le plus grand motif qui pourrait être extrait à partir du nœud courant (en ne suivant plus que des branches gauches) ne la vérifie pas, aucun motif issu de ce nœud ne la vérifie. Une contrainte sera dite *monotone* (cas, e.g., de la contrainte de connexion $\mathcal{C}_{\text{ET-cx}}$) si

lorsque le plus petit motif qui pourrait être extrait à partir du nœud courant (en ne suivant plus que des branches droites) ne la vérifie pas. Dans ce cas, implique qu'aucun motif issu du nœud courant ne la vérifie. Pour de meilleures performances, on effectue le test détaillé ci-dessus avec un nœud d'avance, i.e., plutôt que d'élaguer un nœud duquel ne découlerait aucun motif satisfaisant $\mathcal{C}_{\text{ET-cx}}$, on va éviter de le générer en retirant, au niveau de son père, l'élément correspondant de l'espace de recherche. On parle alors de *propagation*. Étant donné un nœud N, appelons $U = \langle U^1, \ldots, U^n \rangle$ le plus petit motif qui pourrait être extrait à partir de N (borne inférieure) et $U \cup V$ le plus grand motif qui pourrait être extrait à partir de ce même nœud (borne supérieure). $V = \langle V^1, \ldots, V^n \rangle$ représente donc l'espace de recherche, i.e., les éléments qu'il reste à énumérer. Pour nous, U et V sont matérialisés et chaque nœud N de l'arbre peut être identifié à l'aide de son couple (U, V). Nous dirons que N *représente* l'ensemble des motifs compris entre U et $U \cup V$, i.e., $\{\langle W^1, \ldots, W^n \rangle \mid \forall i = 1 \ldots n, U^i \subseteq W^i \subseteq U^i \cup V^i\}$. Soit un n-ensemble $X \in 2^{D^1} \times \cdots \times 2^{D^n}$ et un élément $e \in D^i$ ($i = 1 \ldots n$), pour simplifier, $X \cup \{e\}$ désignera le motif obtenu en « ajoutant » e à X, i.e., $\langle X^1, \ldots, X^i \cup \{e\}, \ldots, X^n \rangle$. De même, nous écrirons $e \in X$ ou nous parlerons du motif $X \setminus \{e\}$.

Règle 1 (Élagage avec $\mathcal{C}_{\text{ET-fm}}$)
Si $\exists i = 1 \ldots n$ **et** $\exists e \in D^i \setminus (U^i \cup V^i)$ **tel que**
$0(H(U \cup V, e)) \leq \epsilon^i$ **et** $\forall f \in U^{j \neq i}, 0(H(U \cup V \cup \{e\}, f)) \leq \epsilon^j$
Alors le nœud (U, V) est élagué.

En effet, aucun n-ensemble qui serait issu de (U, V) ne vérifie $\mathcal{C}_{\text{ET-fm}}$. En effet, tous peuvent être étendus avec l'élément e satisfaisant la prémisse de la règle (c'est le cas du plus grand d'entre eux, $U \cup V$, et $\mathcal{C}_{\text{ET-fm}}$ est anti-monotone).

Règle 2 (Propagation avec $\mathcal{C}_{\text{ET-cx}}$)
Si $\exists i = 1 \ldots n$ **et** $\exists e \in V^i$ **tel que**
$0(H(U, e)) > \epsilon^i$ **ou** $\exists f \in U^{j \neq i}, 0(H(U \cup \{e\}, f)) > \epsilon^j$
Alors e est retiré de V^i.

En effet, aucun motif qui serait issu du nœud (U, V) et qui contiendrait un tel élément e ne vérifie $\mathcal{C}_{\text{ET-cx}}$ (cas du plus petit d'entre eux, U, et $\mathcal{C}_{\text{ET-cx}}$ est monotone). Appliquer les Règles 1 et 2 à chaque nœud de l'arbre d'énumération permet d'extraire tous les ET-n-ensembles fermés (et uniquement eux) en évitant d'explorer de grandes parties de l'espace de recherche. Cependant, le coût de calcul lié à l'application de la Règle 1 est élevé. En effet tous les éléments de $\{e \in D^i \setminus (U^i \cup V^i) \mid i = 1 \ldots n\}$ doivent être testés. Or cet ensemble est très grand pour la grande majorité des nœuds qui sont près des feuilles. Cet ensemble peut être réduit en remarquant que si l'un de ses éléments $e \in D^i \setminus (U^i \cup V^i)$ satisfait $0(H(U, e)) > \epsilon^i$, alors il ne peut jamais être utilisé pour déclencher le mécanisme d'élagage d'un nœud issue de (U, V). La même observation est vraie si e satisfait $\exists f \in U^{j \neq i}$ tel que $0(H(U \cup \{e\}, f)) > \epsilon^j$. On peut donc se contenter de tester les éléments de $\mathcal{S} = \langle \mathcal{S}^1, \ldots, \mathcal{S}^n \rangle$ où $\forall i = 1 \ldots n, \mathcal{S}^i = \{e \in D^i \setminus (U^i \cup V^i) \mid 0(H(U, e)) \leq \epsilon^i \wedge \forall f \in U^{j \neq i}, 0(H(U \cup \{e\}, f)) \leq \epsilon^j\}$.

Règle 3 (Nouvelle règle d'élagage avec $\mathcal{C}_{\text{ET-fm}}$)
Si $\exists i = 1 \ldots n$ **et** $\exists e \in \mathcal{S}^i$ **tel que**
$0(H(U \cup V, e)) \leq \epsilon^i$ **et** $\forall f \in U^{j \neq i}, 0(H(U \cup V \cup \{e\}, f)) \leq \epsilon^j$
Alors le nœud (U, V) est élagué.

Un élément propagé par la Règle 2 n'est pas ajouté à $\mathcal{S}$ puisque sa propagation l'empêche d'appartenir à $\mathcal{S}$. En revanche, à chaque fois qu'une branche droite de l'arbre d'énumération

est suivie, l'élément e, qui est énuméré, est ajouté à $\mathcal{S}$. Comme U grandit en allant plus en profondeur dans l'arbre, e risque, par la suite, de ne plus satisfaire la définition d'un élément de $\mathcal{S}$. On ajoute donc une règle pour purger $\mathcal{S}$ de ces éléments.

Règle 4 (Purge de $\mathcal{S}$)
Si $\exists i = 1 \dots n$ **et** $\exists e \in \mathcal{S}^i$ **tel que**
$0(H(U, e)) > \epsilon^i$ **ou** $\exists f \in U^{j \neq i}, 0(H(U \cup \{e\}, f)) > \epsilon^j$
Alors e est retiré de $\mathcal{S}^i$.

Notre algorithme, baptisé ETM, parcourt en profondeur l'espace de recherche. Le nœud à la racine est $(\langle \emptyset, \dots, \emptyset \rangle, \langle D^1, \dots, D^n \rangle)$. $\mathcal{S}$ est initialisé à $\langle \emptyset, \dots, \emptyset \rangle$. A chaque nœud (U, V) de l'arbre d'énumération, l'élément e à énumérer est choisi et l'espace de recherche est partitionné en deux. Les ET-n-ensembles fermés issus du nœud N_G contiendront tous l'élément e, tandis que ceux issus de N_D ne le contiendront jamais. Par appel récursif sur chacun de ces nœuds, tous les ET-n-ensembles fermés représentés par (U, V) sont extraits.

Entrée : Un nœud (U, V) de l'arbre d'énumération, $\mathcal{S}$ tel que défini dans la Section 3.
Sortie : Les ET-n-ensembles fermés représentés par (U, V)
si Règle 3 ne s'applique pas **alors**
  si $V = \langle \emptyset, \dots, \emptyset \rangle$ **alors**
    Sortir $\langle U^1, \dots, U^n \rangle$
  sinon
    Choisir (i, e) tel que $i \in 1 \dots n$ et $e \in V^i$
    $V_G \leftarrow$ Règle 2($U \cup \{e\}, V \setminus \{e\}$)
    $\mathcal{S}_G \leftarrow$ Règle 4($U \cup \{e\}, \mathcal{S}$)
    ETM($(U \cup \{e\}, V_G), \mathcal{S}_G$)
    ETM($(U, V \setminus \{e\}), \mathcal{S} \cup \{e\}$)
  fin si
fin si

Algorithme 1 : ETM

Notez que si le choix de l'élément énuméré ne change pas la sortie d'ETM, il influe énormément sur la taille de l'arbre d'énumération parcouru et donc sur les temps d'extraction. Faute de place, nous ne détaillons ici ni la stratégie d'énumération choisie, ni les mécanismes nécessaires à l'implémentation de ETM. Cerf et al. (2008a) fournit plus de détails au lecteur intéressé. Il en est de même pour la discussion de l'état de l'art (Yang et al., 2001; Pei et al., 2001; Besson et al., 2006; Cheng et al., 2006; Poernomo et Gopalkrishnan, 2007).

4 Etude expérimentale

Nous ne considérons ici qu'un protocole expérimental très simple. Quatre n-ensembles (avec quatre éléments par attribut), pouvant se recouvrir, sont introduits de façon aléatoire dans un espace à trois ou quatre dimensions contenant 16 éléments par dimension. La relation ainsi générée $\mathcal{R}_{\text{cachée}}$ est ensuite bruitée de façon uniforme, i.e., chaque n-uplet a la même probabilité p d'être « mal » encodé (un n-uplet absent de $\mathcal{R}_{\text{cachée}}$ est en fait présent ou vice-versa). La relation obtenue est ensuite fouillée avec ETM. Toutes les extractions sont faites

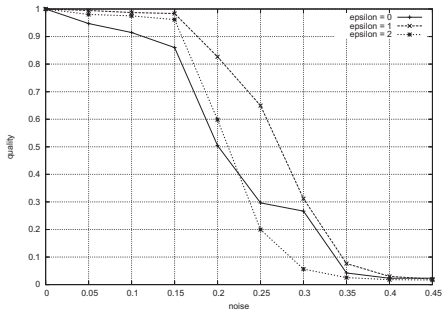
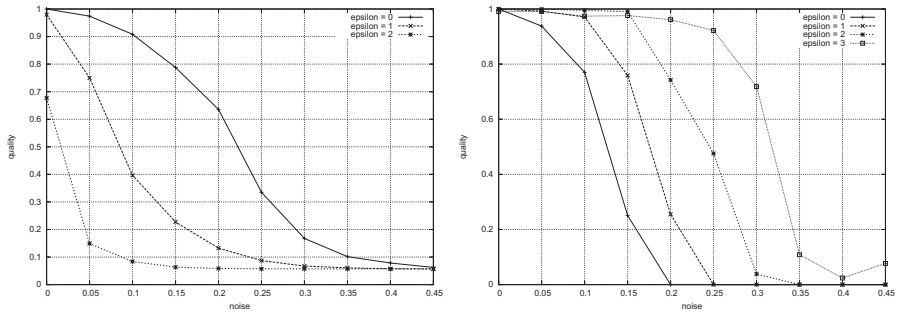


FIG. 1 – *Qualité des ET-*4*-ensembles fermés extraits avec* $\epsilon \in \{0, 1, 2\}$



(a) Espace de taille $16 \times 16 \times 16$ cachant des 3-ensembles de taille $4 \times 4 \times 4$

(b) Espace de taille $32 \times 32 \times 32$ cachant des 3-ensembles de taille $8 \times 8 \times 8$

FIG. 2 – *Qualité des ET-*3*-ensembles fermés*

avec un même seuil ϵ de tolérance aux exceptions et une même contrainte de taille minimale (au moins deux éléments) pour chaque attribut. On note $\mathcal{E}$ l'ensemble des n-uplets couverts par au moins un ET-n-ensemble fermé extrait. On mesure alors la qualité de la collection de ET-n-ensembles fermés à l'aide de l'expression $|\mathcal{R}_{\text{cachée}} \cap \mathcal{E}|/|\mathcal{R}_{\text{cachée}} \cup \mathcal{E}|$.

La Figure 1 représente les résultats obtenus sur la relation 4-aire plus ou moins bruitée. Quelque soit le niveau de bruit p, les ET-4-ensembles fermés extraits avec $\epsilon = 1$ sont de meilleur qualité que les 4-ensembles fermés exacts (i.e., $\epsilon = 0$). Par exemple, avec $p = 0,25$, la qualité des 4-ensembles fermés exacts est en deçà de $0,3$ alors qu'elle atteint $0,65$ lorsque $\epsilon = 1$. De plus on voit que $\epsilon = 2$ est excessif. Cela s'explique par des contraintes de tailles minimales faibles. La Figure 2(a) représente les résultats obtenus sur la relation ternaire plus ou moins bruitée. Cette fois, le meilleur seuil ϵ de tolérance aux exceptions est 0. Nous observons ici que prendre en compte l'aspect bruité des données revêt d'autant plus d'importance que l'arité est grande. L'intérêt de la tolérance aux exceptions dans les relations ternaires est clairement visible lorsque les 3-ensembles de $\mathcal{R}_{\text{cachée}}$ sont plus grands. Ainsi, avec des 3-ensembles de taille $8 \times 8 \times 8$ dans un espace de taille $32 \times 32 \times 32$, la collection de ET-3-ensembles fermés, extraits avec une contrainte d'au moins quatre éléments par attribut, bénéficie de cette

tolérance. Jusqu'à un niveau de bruit p de $0,15$, $\epsilon = 2$ permet d'obtenir une qualité presque parfaite, alors que, à $p = 0,15$, la qualité de la collection de 3-ensembles fermés exacts est de $0,25$. Avec $p > 0,15$, il est plus avantageux de fixer ϵ à 3. Nous constatons donc que plus une relation est bruitée, plus il est important de fixer un seuil élevé de tolérance aux exceptions.

En conclusion, l'extraction complète de motifs fermés dans des relations n-aires a récemment été étudiée. La présence de bruit dans ces données est d'autant plus désastreuse (explosion du nombre de motifs extraits, baisse de leur pertinence) que l'arité n est grande. Nous proposons un algorithme qui, étant donné une relation n-aire, extrait tous les motifs fermés tolérant un nombre borné d'exceptions par élément. Ayant validé ETM sur des données synthétiques et réelles, nous travaillons maintenant à son adaptation pour le calcul de quasi-cliques dans des graphes dynamiques.

Références

Besson, J., C. Robardet, et J.-F. Boulicaut (2006). Mining a new fault-tolerant pattern type as an alternative to formal concept discovery. In *ICCS*, pp. 144–157.

Cerf, L., J. Besson, et J.-F. Boulicaut (2008a). Calcul de motifs fermés tolérants aux exceptions dans des relations n-aires bruitées. In *Research Report LIRIS, Septembre 2008, 12 pages*.

Cerf, L., J. Besson, C. Robardet, et J.-F. Boulicaut (2008b). DATA-PEELER : Contraint-based closed pattern mining in n-ary relations. In *SDM*.

Cheng, H., P. S. Yu, et J. Han (2006). AC-Close : Efficiently mining approximate closed itemsets by core pattern recovery. In *ICDM*, pp. 839–844.

Ganter, B., G. Stumme, et R. Wille (2005). *Formal Concept Analysis, Foundations and Applications*, Volume 3626. Springer.

Jaschke, R., A. Hotho, C. Schmitz, B. Ganter, et G. Stumme (2006). TRIAS : An algorithm for mining iceberg tri-lattices. In *ICDM*, pp. 907–911.

Ji, L., K.-L. Tan, et A. K. H. Tung (2006). Mining frequent closed cubes in 3D datasets. In *VLDB*, pp. 811–822.

Pei, J., A. K. H. Tung, et J. Han (2001). Fault-tolerant frequent pattern mining : Problems and challenges. In *DMKD*.

Poernomo, A. K. et V. Gopalkrishnan (2007). Mining statistical information of frequent fault-tolerant patterns in transactional databases. In *ICDM*, pp. 272–281.

Yang, C., U. Fayyad, et P. S. Bradley (2001). Efficient discovery of error-tolerant frequent itemsets in high dimensions. In *SIGKDD*, pp. 194–203.

Summary

The extraction of closed patterns in binary relations has been studied extensively. Nevertheless, many relations appear to be n-ary with $n > 2$ and noisy (need for fault-tolerance). Recently, these two issues have been independently studied. We introduce our proposal for the integration of both functionalities within a unique algorithm.