Extraction des Top-k Motifs par Approximer-et-Pousser

Arnaud Soulet et Bruno Crémilleux

GREYC, CNRS - UMR 6072, Université de Caen
Campus Côte de Nacre
14032 Caen Cedex France
{Prenom.Nom}@info.unicaen.fr

Résumé. Cet article porte sur l'extraction de motifs sous contraintes *globales*. Contrairement aux contraintes usuelles comme celle de fréquence minimale, leur vérification est problématique car elle entraine de multiples comparaisons entre les motifs. Typiquement, la localisation des k motifs maximisant une mesure d'intérêt, i.e. satisfaisant la contrainte top-k, est difficile. Pourtant, cette contrainte globale se révèle très utile pour trouver les motifs les plus significatifs au regard d'un critère choisi par l'utilisateur. Dans cet article, nous proposons une méthode générale d'extraction de motifs sous contraintes globales, appelée Approximer-et-Pousser. Cette méthode peut être vue comme une méthode de relaxation d'une contrainte globale en une contrainte locale évolutive. Nous appliquons alors cette approche à l'extraction des top-k motifs selon une mesure d'intérêt. Les expérimentations montrent l'efficacité de l'approche Approximer-et-Pousser.

Mots clés : extraction de motifs, contraintes.

1 Introduction

L'extraction de motifs contraints est un champ significatif de l'Extraction de Connaissances dans les Bases de Données, notamment pour dériver des règles d'association. L'intérêt des motifs extraits est garanti par le point de vue de l'analyste exprimé à travers la sémantique de la contrainte. Par ailleurs, la complétude de l'extraction assure qu'aucun motif jugé pertinent par l'utilisateur ne sera manqué. La contrainte la plus populaire est certainement celle de fréquence minimale (Agrawal et al., 1993) qui permet de rechercher des régularités au sein d'une base de données. Malheureusement, le nombre de motifs fréquents est souvent prohibitif. Les motifs les plus pertinents sont alors noyés au milieu d'informations triviales ou redondantes que même d'autres contraintes d'agrégats (Ng et al., 1998) n'arrivent pas davantage à isoler.

Dans ces conditions, plusieurs approches proposent de comparer les motifs entre eux pour ne sélectionner que les meilleurs (Fu et al., 2000) ou une couverture (Mannila et Toivonen, 1997; Pasquier et al., 1999). De tels motifs révèlent alors une structure globale au sein des données. Le critère d'appartenance ou non à cette structure s'apparente à une contrainte *globale*. L'extraction de motifs satisfaisant une contrainte globale présente donc une finalité importante pour les utilisateurs. Cependant, leur extraction s'avère souvent ardue car leur localisation dans l'espace de recherche est loin d'être triviale. En particulier, trouver les k motifs maximisant

une mesure d'agrégat (e.g., la fréquence (Fu et al., 2000)) devient problématique dès que la mesure ne satisfait aucune propriété particulière comme l'anti-monotonie.

Dans cet article, nous proposons d'extraire des motifs satisfaisant une contrainte globale. Notre première contribution est une méthode générale d'extraction appelée *Approximer-et-Pousser*. L'idée fondamentale est de déduire une contrainte locale qui est affinée au cours de l'extraction. Cette contrainte locale évolutive, exploitée par un algorithme indépendant, réduit alors l'espace de recherche. Ensuite, nous appliquons cette méthode pour rechercher les top-k motifs selon une mesure d'intérêt spécifiée par l'utilisateur. Nous expliquons comment constituer une approximation des top-k motifs à extraire. Puis, nous montrons que cette approximation peut être poussée pour réduire considérablement l'espace de recherche en se fondant sur une méthode de relaxation exposée dans (Soulet et Crémilleux, 2005). L'une des originalités de notre approche est d'autoriser des mesures sans bonne propriété de monotonie et ainsi, de ne pas se cantonner aux seuls top-k motifs fréquents.

Cet article est organisé de la manière suivante. Dans le contexte des contraintes globales, la section 2 définit la notion de top-k motifs selon une mesure et en présente la problématique de l'extraction. La section 3 décrit l'approche Approximer-et-Pousser dédiée à l'extraction des contraintes globales. La section 4 applique cette méthode pour la recherche des top-k motifs selon une mesure en détaillant les deux étapes. Enfin, cette approche est évaluée à la section 5.

2 Contexte et travaux relatifs

2.1 Contexte et définitions

Étant donné un ensemble $\mathcal{I}$ de littéraux distincts appelés *items*, un motif d'items correspond à un sous-ensemble non vide de $\mathcal{I}$. Tous ces motifs sont regroupés dans le langage $\mathcal{L}_\mathcal{I}$: $\mathcal{L}_\mathcal{I} = 2^\mathcal{I} \setminus \{\emptyset\}$. Un contexte transactionnel est alors défini comme un multi-ensemble de motifs de $\mathcal{L}_\mathcal{I}$. Chacun de ces motifs, appelé *transaction*, constitue une entrée de la base de données. Typiquement le tableau 1 présente un contexte transactionnel $\mathcal{D}$ où 6 transactions étiquetées $t_1, \ldots, t_6$ sont décrites par 6 items $A, \ldots, F$.

$\mathcal{D}$

Trans.	Items					
t_1	A	B			E	F
t_2	A		C		E	
t_3	A	B	C	D		
t_4	A	B	C			
t_5	A			D		
t_6			C		E	

TAB. 1 – *Exemple d'un contexte transactionnel $\mathcal{D}$.*

L'extraction de motifs cherche la collection de tous les motifs de $\mathcal{L}_\mathcal{I}$ satisfaisant un prédicat q, appelé *contrainte*, et présents dans le contexte transactionnel $\mathcal{D}$. Un motif X est présent dans $\mathcal{D}$ s'il apparaît dans au moins une de ses transactions. Introduite dans (Agrawal et

Srikant, 1994), l'une des contraintes les plus utilisées est celle de fréquence minimale. La fréquence d'un motif X, dénotée par $freq(X)$, donne le nombre de transactions contenant X. La contrainte de fréquence minimale (i.e., $freq(X) \geq \gamma$) sélectionne les motifs dont la fréquence excède un seuil γ fixé par l'utilisateur. De nombreuses contraintes remplacent la fréquence par une autre mesure d'intérêt pour juger au mieux la pertinence d'un motif (Ng et al., 1998). Parmi ces mesures, l'aire d'un motif X, notée $area(X)$, correspond au produit de sa fréquence par sa longueur (i.e., $freq(X) \times count(X)$ où $count(X)$ dénote la cardinalité de X). La vérification de certaines mesures d'intérêts $m(X) \geq \gamma$ où $m : \mathcal{L}_{\mathcal{I}} \rightarrow \Re$, nécessite de complèter $\mathcal{D}$ avec des informations supplémentaires (e.g., une table associant des valeurs à chaque item pour $sum(X.val)$).

Ces mesures d'intérêt ne suffisent pas toujours à focaliser directement sur les motifs les plus significatifs. Il s'avère alors nécessaire de comparer les motifs entre eux pour n'en conserver que les meilleurs ou une couverture. De tels motifs révèlent alors une structure globale au sein des données. L'appartenance ou non à cette structure se formalise avec la notion de contrainte *globale* que nous définissons maintenant :

Définition 1 (Contrainte globale) *Une contrainte est globale si sa vérification nécessite de comparer plusieurs motifs entre eux.*

Typiquement, les contraintes "être-maximal" (Mannila et Toivonen, 1997), "être-fermé" (Pasquier et al., 1999) ou "être-libre" (Boulicaut et al., 2003) qui dégagent une représentation des motifs, sont des contraintes globales. Par exemple, la contrainte "être-fermé" extrait une couverture de la base de données en sélectionnant uniquement les motifs X dont toutes les spécialisations $Y \supset X$ ont une fréquence strictement inférieure à celle de X. La vérification de "être-fermé" nécessite donc la comparaison de X avec d'autres motifs. De même, tester si X est un motif fermé en le comparant avec sa fermeture est une comparaison entre deux motifs. Les contraintes globales mettent intrinsèquement en relation plusieurs motifs contrairement aux contraintes usuelles, dites *locales*, qui peuvent se vérifier isolément sur chacun des motifs.

Dans cet article, nous nous intéressons plus particulièrement à la contrainte globale correspondant aux top-k motifs selon une mesure d'intérêt. Le choix du seuil pour la contrainte de fréquence ou d'aire minimale (et plus généralement de $m(X) \geq \gamma$) se révèle souvent difficile pour l'utilisateur. En effet, si ce seuil est trop élevé, trop peu de motifs sont extraits (au risque de n'obtenir que des informations triviales). A l'inverse, si γ est trop bas, le nombre de motifs explose et les motifs les plus intéressants sont noyés dans la masse. Comme plusieurs tentatives d'extraction sont nécessaires pour estimer γ, l'utilisateur préfère souvent fixer ce dernier relativement bas rendant parfois les extractions infaisables. Puis, parmi tous les motifs obtenus, il focalise son intérêt sur les premiers motifs maximisant sa mesure d'intérêt. La recherche de ces k motifs optimisant une mesure d'intérêt m est ainsi une tâche qui présente un vif intérêt et qui peut aussi se formuler sous forme d'une contrainte :

Définition 2 (Contrainte des top-k motifs) *Soient un entier $k > 0$ et une mesure $m : \mathcal{L}_{\mathcal{I}} \rightarrow \Re$, la contrainte des top-$k$ motifs selon m correspond à :*

$$top_{k,m}(X) \equiv |\{Y \in \mathcal{L}_{\mathcal{I}} | Y \neq X \wedge m(Y) > m(X)\}| < k$$

La contrainte $top_{k,m}$ est clairement globale puisque X et Y sont présents conjointement dans la définition. Cette contrainte compare les motifs entre eux pour conserver ceux dont la

mesure fait partie des k meilleures. Par exemple, les 3 motifs de plus grande aire correspondent exactement aux motifs satisfaisants $top_{3,area}$[1] : AB ($3 \times 2 = 6$), AC ($3 \times 2 = 6$) et ABC ($2 \times 3 = 6$). Les motifs associés à la contrainte $top_{k,m}$ sont nommés les top-k motifs selon la mesure m. En fait, leur nombre est parfois supérieur à k (tous les motifs au-delà du $k^{\text{ème}}$ ont alors la même mesure). Typiquement les top-3 motifs fréquents sont 4 à savoir A (5), C (4), B (3) et E (3), car la fréquence ne permet pas de distinguer les motifs B et E. Notons que les k motifs minimisant une mesure m satisfont la contrainte $top_{k,-m}$.

Naïvement l'extraction des top-k motifs peut s'effectuer avec un post-traitement. Après l'extraction de tous les motifs dont la mesure m excède un seuil γ, il suffit de sélectionner les k motifs maximisant m. Outre l'inefficacité algorithmique, la difficulté du choix du seuil minimal persiste. Si celui-ci est fixé trop haut, moins de k motifs peuvent être extraits. En revanche, si ce seuil est trop bas, des motifs inutiles sont extraits et ce processus ne profitant pas du paramètre k devient très lent voire infaisable. Pour résoudre ce problème, il est préférable de pousser la contrainte $top_{k,m}$ au sein de l'extraction de motifs. Cette tâche est peu aisée car se pose une double problématique à travers la localisation des motifs dont la mesure est potentiellement élevée et la comparaison de ces mesures pour garantir la maximalité des mesures correspondant aux motifs finalement retenus. En fait, cette contrainte recouvre des problèmes inhérents à la vérification des contraintes globales.

2.2 Travaux relatifs

À notre connaissance, aucune méthode générale d'extraction des contraintes globales n'a été proposée dans la littérature auparavant. Individuellement certaines contraintes globales comme "être-maximal" (Mannila et Toivonen, 1997), "être-fermé" (Pasquier et al., 1999) ou "être-libre" (Boulicaut et al., 2003) ont des algorithmes spécifiques reposant principalement sur des élagages anti-monotones. Par exemple, la contrainte de liberté est anti-monotone : si un motif n'est pas libre aucune de ses spécialisations ne sera libre et on peut alors élaguer cette partie de l'espace de recherche. L'extraction des top-k motifs *fréquents* a été introduite dans (Fu et al., 2000). Les auteurs adaptent APRIORI (Agrawal et Srikant, 1994) pour ajuster le seuil de fréquence minimale au fur et à mesure de l'extraction en bénéficiant à nouveau de l'anti-monotonie de la fréquence. Dans (Hirate et al., 2004), la structure FP-tree permet d'optimiser l'extraction des top-k motifs fréquents. Plus récemment, la structure COFI-tree a aussi été utilisée (Ngan et al., 2005). Dans (Tzvetkov et al., 2003), les auteurs remplacent le langage $\mathcal{L}_{\mathcal{I}}$ par celui des motifs séquentiels pour extraire les top-k séquences fréquentes. Plusieurs travaux extraient aussi les top-k motifs fréquents satisfaisant un critère additionnel. Par exemple, les k motifs fermés les plus fréquents et de longueur minimale sont recherchés dans (Han et al., 2002) en utilisant la structure FP-tree. D'autres recherchent les motifs les plus fréquents, fermés ou non, et de longueur minimale (Cong, 2001). Tous ces travaux sont restreints à la mesure de fréquence comme mesure d'intérêt car la contrainte de fréquence minimale est anti-monotone (i.e., les motifs satisfaisant $top_{k,freq}$ sont les plus généraux). Remarquons aussi qu'ils se focalisent principalement sur les motifs d'items.

Notre démarche se distingue donc en proposant une méthode adaptée à n'importe quelle mesure d'intérêt basée sur les primitives. Même si cet article est dédié aux motifs d'items, l'approche Approximer-et-Pousser est adaptable à d'autres langages (séquences, arbres, etc).

[1] Les k motifs de plus grande aire ont ici la même aire, mais ce n'est pas toujours le cas.

Par ailleurs, l'approche Approximer-et-Pousser constitue une première proposition générique pour l'extraction des contraintes globales.

3 Approximer-et-Pousser : une approche générique d'extraction pour les contraintes globales

L'approche Approximer-et-Pousser est une méthode générique d'extraction des motifs satisfaisant une contrainte globale. Brièvement, l'idée est de restreindre l'espace de recherche lors du parcours en affinant la localisation des motifs susceptibles de vérifier la contrainte globale. Pour cela, cette approche s'appuie sur la répétition de deux étapes majeures (et qui forment son nom) : (1) approximer la collection finale à extraire, (2) pousser des informations issues de cette approximation pour diminuer l'espace de recherche. Plutôt que d'algorithme Approximer-et-Pousser, nous préférons parler d'approche Approximer-et-Pousser car par la suite, cette approche délègue l'élagage de l'espace de recherche à un algorithme indépendant. La condition d'élagage lui est donnée sous forme d'une contrainte locale d'extraction qui est dynamiquement affinée à chaque itération. Une telle approche Approximer-et-Pousser peut être alors vue comme une relaxation évolutive de la contrainte globale en une contrainte locale. Plusieurs illustrations de cette approche sont proposées dans (Soulet, 2006) pour extraire les contraintes "être-maximal" et "être-libre". La section suivante instancie cette aproche pour l'extraction des top-k motifs selon une mesure d'intérêt. Auparavant, nous développons, de manière générale, les deux étapes majeures de l'approche Approximer-et-Pousser :

Approximer La mise à jour de la collection de motifs candidats se décline en trois opérations : l'initialisation, l'ajout et la suppression. L'*initialisation* de la collection des motifs candidats doit être choisie avec attention afin de ne pas manquer de motifs. Lorsque l'espace de recherche est parcouru dans son ensemble, la collection est initialisée à vide ce qui assure la complétude. Ensuite, l'*ajout* et la *suppression* des motifs interviennent à chaque nouvelle étape d'approximation i.e., un nouveau motif postule pour entrer dans la collection. Ce dernier est ajouté à celle-ci si et seulement si au vu des motifs candidats déjà présents dans la collection, il peut éventuellement satisfaire la contrainte globale. Enfin, un motif est supprimé de la collection s'il est exclu par un motif postulant. Un motif peut être supprimé soit *positivement* (i.e., il est conservé car il satisfait la contrainte globale), soit *négativement* (sinon). Lorsqu'un motif est exclu par le motif postulant, cela n'implique pas toujours l'entrée de ce dernier.

Pousser Par l'intermédiaire de la collection de motifs candidats, cette étape doit permettre de pousser la contrainte globale au cœur de l'extraction et ainsi, réduire l'espace de recherche. Dans un premier temps, cette étape déduit certaines informations de l'approximation (e.g., un calcul effectué sur l'ensemble des motifs candidats). Ces informations évoluent au gré de l'ajout et de la suppression des motifs. Ensuite, celles-ci sont converties en une condition d'élagage afin d'éliminer des motifs de l'espace de recherche. Cette condition d'élagage peut par exemple être une contrainte locale adaptée à un algorithme d'extraction.

4 Extraction des top-k motifs selon une mesure

4.1 Aperçu de l'approche

Cette section donne un aperçu général de notre approche d'extraction des top-k motifs selon une mesure m en exploitant la méthode Approximer-et-Pousser.

L'extraction des top-k motifs selon une mesure m est épineuse car en général, on ne sait pas où se situeront dans l'espace de recherche les motifs vérifiant la contrainte. Par ailleurs, la définition 2 ne permet pas directement d'obtenir une contrainte locale qui pourrait être exploitée par un algorithme usuel. Afin de pallier en partie ce dernier point, nous introduisons une définition alternative des top-k motifs avec la propriété 1 :

Propriété 1 *Le seuil minimal d'appartenance aux top-k motifs selon la mesure m, dénoté $\rho_{k,m}$, est $\min\{m(X)|X \in \mathcal{L}_\mathcal{I} \wedge top_{k,m}(X)\}$, et on a $top_{k,m}(X) \equiv m(X) \geq \rho_{k,m}$.*

Preuve. Soient m une mesure et $k > 0$, on fixe $\rho_{k,m} = \min\{m(X)|X \in \mathcal{L}_\mathcal{I} \wedge top_{k,m}(X)\}$. Soit $X \in \mathcal{L}_\mathcal{I}$, si $m(X)$ est supérieure à $\rho_{k,m}$, on a bien $top_{k,m}(X)$ qui est vraie par définition. Sinon, si $m(X)$ est strictement inférieure à $\rho_{k,m}$, X ne peut satisfaire $top_{k,m}$ car par définition, $\rho_{k,m}$ est inférieur ou égal à tous les motifs satisfaisant $top_{k,m}$.

Cette reformulation de la contrainte des top-k motifs pour m est à nouveau une contrainte globale. Le seuil $\rho_{k,m}$ concentre implicitement les comparaisons entre motifs nécessaire pour vérifier la contrainte $top_{k,m}$. Néanmoins, cette reformulation rend possible la définition d'une contrainte locale en fixant le seuil $\rho_{k,m}$ (même arbitrairement). Nous verrons que ce point est essentiel par la suite.

Dans la suite, nous proposons d'exploiter cette propriété avec l'approche Approximer-et-Pousser en considérant :

1. **Approximer :** cette étape d'approximation permettra de déterminer un seuil ρ tendant à égaler $\rho_{k,m}$ à partir d'une collection de motifs candidats.
2. **Pousser :** cette étape poussera la contrainte $m(X) \geq \rho$ pour réduire l'espace de recherche.

Chacune de ces deux étapes est difficile. La première doit permettre de fixer le seuil temporaire ρ de façon à ne pas éliminer de motifs satisfaisant $top_{k,m}$ (section 4.2.1). La contrainte à pousser $m(X) \geq \rho$ n'est pas forcément anti-monotone. Nous utiliserons alors le principe de la relaxation anti-monotone (section 4.2.2). Ainsi, avec un algorithme d'extraction de contrainte anti-monotone (comme l'algorithme par niveaux (Mannila et Toivonen, 1997)), notre approche permet de traiter un large ensemble de mesures.

4.2 Description des deux étapes

4.2.1 Approximer les top-k motifs

L'étape d'approximation conserve les k motifs maximisant la mesure m parmi les motifs déjà extraits. De cette façon, lorsque l'algorithme d'extraction aura parcouru l'intégralité de l'espace de recherche, les k motifs candidats retenus seront exactement les top-k motifs selon la mesure m.

À l'initialisation de l'extraction, la collection des motifs candidats $\mathcal{C}and$ ne contient aucun motif. La maintenance de cette collection commence alors par une phase de remplissage. Tous les motifs extraits sont ajoutés sans condition jusqu'à obtenir une collection de k motifs candidats. Durant cette phase, aucun motif de $\mathcal{C}and$ n'est supprimé. Ensuite, l'évolution de $\mathcal{C}and$ entre dans une phase sélective guidée par la propriété suivante :

Propriété 2 *Soit un ensemble de motifs C tel que $|C| \geq k$, si la mesure m d'un motif donné est strictement inférieure à celle de chacun des motifs de C, alors ce motif ne satisfait pas la contrainte $top_{k,m}$.*

Preuve. Soit un motif X et $C \subseteq \mathcal{L}_{\mathcal{I}}$ tel que $|C| \geq k$. Fixons ρ' à $\min_{Y \in C} m(Y)$. Comme les motifs satisfaisant la contrainte $top_{k,m}$ maximisent m, on a $\rho_{k,m} \geq \rho'$. Or $m(X) < \rho'$, on obtient que $m(X) < \rho_{k,m}$ et la propriété 1 permet de conclure que X ne satisfait pas la contrainte $top_{k,m}$.

Dans notre approche, la collection C de cette propriété correspond aux motifs candidats $\mathcal{C}and$ (ou à un de ses sous-ensembles). Dès que $\mathcal{C}and$ a atteint k éléments, la propriété peut être appliquée sur un motif postulant pour savoir s'il est bien nécessaire de l'ajouter à la collection des motifs candidats. Plus précisément, un motif postulant X est ajouté à la collection si la mesure de X est supérieure à celle d'au moins un des motifs candidats. Dans le cas contraire, la propriété 2 nous garantit que le motif postulant ne pourra pas faire partie des top-k motifs selon m. En outre, un motif est supprimé de la collection dès que k autres motifs de $\mathcal{C}and$ ont une mesure supérieure à la sienne. En effet, la propriété 2 assure à nouveau que ce motif ne sera jamais parmi les k motifs de plus forte mesure m et donc, ne satisfera pas la contrainte $top_{k,m}$.

L'introduction du seuil d'ajout permet d'unifier ces deux phases distinctes de l'étape approximer :

Définition 3 (Seuil d'ajout) *Le seuil d'ajout, noté ρ, est défini de la manière suivante :*

$$\rho = \begin{cases} -\infty, & \textit{si } |\mathcal{C}and| < k \\ \min_{X \in \mathcal{C}and} m(X), & \textit{sinon} \end{cases}$$

L'intérêt de cette approche est que ce seuil évolue au fur et à mesure des modifications de la collection des motifs candidats $\mathcal{C}and$. Basiquement, un motif postulant est ajouté à la collection si et seulement si sa mesure m est supérieure à celle du seuil d'ajout ρ. Ainsi, durant la phase de remplissage, la collection accepte tous les motifs car leur mesure est toujours supérieure au seuil d'ajout alors égal à $-\infty$. Ensuite, les valeurs de la mesure de chacun des motifs de $\mathcal{C}and$, synthétisées par le seuil d'ajout, conditionne l'introduction ou non du motif postulant au sein de la collection.

Le tableau 2 décrit l'évolution des motifs candidats $\mathcal{C}and$ au cours du processus d'extraction de la contrainte $top_{3,area}$ (cf. section 2.1) avec l'algorithme APRIORI pour le contexte donné au tableau 1. L'algorithme par niveaux génère trois vagues successives de motifs postulants. La section 4.2.2 explique quels sont les motifs extraits par APRIORI. Pour chaque niveau, les motifs dont l'aire est supérieure à ρ (motifs en gras) entrent dans la collection des motifs candidats. La valeur de l'aire est donnée par le chiffre entre parenthèses dans la colonne de

Niveau 1

Motif	$\mathcal{C}and$	ρ
A (5)	A	$-\infty$
B (3)	A, B	$-\infty$
C (4)	A, C, B	3
E (3)	A, C, B, E	3
F (1)	A, C, B, E	3

Niveau 2

Motif	$\mathcal{C}and$	ρ
AB (6)	AB, A, C	4
AC (6)	AB, AC, A	5
AD (4)	AB, AC, A	5
AE (4)	AB, AC, A	5
AF (2)	AB, AC, A	5
BC (4)	AB, AC, A	5
BD (2)	AB, AC, A	5
BE (2)	AB, AC, A	5
BF (2)	AB, AC, A	5
CD (2)	AB, AC, A	5
CE (4)	AB, AC, A	5
EF (2)	AB, AC, A	5

Niveau 3

Motif	$\mathcal{C}and$	ρ
ABC (6)	AB, AC, ABC	6

TAB. 2 – *Les top-*3 *motifs selon l'aire avec* APRIORI.

gauche et les motifs candidats sont rassemblés dans la colonne centrale. Le seuil ρ (colonne de droite) est ajusté au fur et à mesure. Tant que le nombre de motifs de $\mathcal{C}and$ est inférieur à k, le seuil ρ a pour valeur $-\infty$. Ensuite, ρ correspond à l'aire minimale satisfaite par un des motifs de $\mathcal{C}and$. Le motif E n'est pas exclu par l'entrée de B car son aire est égale ρ. En revanche, B et E sont supprimés à l'arrivée du motif AB. À la fin du dernier niveau, $\mathcal{C}and$ correspond aux 3 motifs de plus forte mesure d'aire.

4.2.2 Pousser l'approximation

Cette étape bénéficie de la collection obtenue des motifs candidats afin de réduire l'espace de recherche. Nous montrons maintenant comment il est possible de déduire de cette collection une contrainte anti-monotone afin de réutiliser des algorithmes efficaces bénéficiant de l'anti-monotonie.

Les seuls motifs pouvant satisfaire la contrainte $top_{k,m}$ sont ceux qui peuvent être ajoutés à la collection des motifs candidats (car les autres sont immédiatement rejetés, cf. la propriété 2). Ces motifs doivent donc avoir une mesure supérieure au seuil d'ajout i.e., ils satisfont la contrainte locale $m(X) \geq \rho$. De façon générale, cette contrainte n'est pas anti-monotone. Typiquement, la contrainte $area(X) \geq \rho$ n'est pas anti-monotone. Par exemple, dans le contexte $\mathcal{D}$, le motif ABC satisfait la contrainte $area(X) \geq 6$, mais pas sa généralisation BC dont l'aire est seulement de 4. Afin d'obtenir dans le cas général une contrainte anti-monotone, nous proposons d'approximer la contrainte $m(X) \geq \rho$ par une *relaxation* $m'(X) \geq \rho$ vérifiant les deux conditions suivantes : $m'(X) \geq \rho$ est anti-monotone (condition $C1$) et $\forall X$, $m(X) \geq \rho \Rightarrow m'(X) \geq \rho$ (condition $C2$), cette dernière assurant la complétude du processus.

L'obtention de la contrainte relaxée $m'(X) \geq \rho$ n'est pas une tâche triviale. Une méthode automatique et générale pour toute *contrainte fondée sur des primitives* est donnée dans (Soulet et Crémilleux, 2005). À partir de la contrainte d'aire, nous allons montrer comment procéder.

Tout d'abord, on peut remarquer que la mesure $freq(X) \times l$ est décroissante lorsque X est croissant et celle-ci satisfait la condition $C1$. D'autre part, il est possible de fixer l pour être certain que $freq(X) \times l$ sera plus grande que $freq(X) \times count(X)$ pour tous les motifs X du jeu de données $\mathcal{D}$ (satisfaction de $C2$). Pour cela, il suffit que le seuil l soit supérieur à la longueur de chacun des motifs présents dans $\mathcal{D}$. Or, la taille du plus grand motif correspond exactement à la taille de la plus grande transaction. De cette manière, l est fixé à 4 avec le jeu de données $\mathcal{D}$. Ainsi, la relaxation anti-monotone $freq(X) \times 4 \geq \rho$ pourra être exploitée comme contrainte locale pour extraire les top-k motifs selon la mesure d'aire dans le jeu de données $\mathcal{D}$.

Reprenons le déroulement de l'extraction des top-3 motifs selon l'aire présenté au tableau 2. La relaxation anti-monotone utilisée est $freq(X) \geq \rho/4$. À la fin du premier niveau, $\rho = 3$ et la relaxation $freq(X) \geq 3/4$ n'élimine aucun motif. Tous les motifs de longueur 2 et présents dans le contexte $\mathcal{D}$ sont donc générés. En revanche, à la fin du niveau 2, $\rho = 5$ et la relaxation devient $freq(X) \geq 5/4$. Pour le niveau 3, seuls les motifs de fréquence supérieure ou égale à 2 sont donc générés (il y a uniquement ABC dans cet exemple). Le processus d'extraction s'arrête alors car plus aucun motif ne peut être généré. Au final, l'approche Approximer-et-Pousser a économisé la génération de 8 motifs de longueur 3 et de 2 motifs de longueur 4.

L'efficacité de cette approche Approximer-et-Pousser réside dans l'ajustement dynamique de la contrainte au cours de l'extraction. Plus précisément, la relaxation anti-monotone $m'(X) \geq \rho$ devient de plus en plus sélective car le seuil d'ajout ρ croît pour tendre vers $\rho_{k,m}$. Cette approche Approximer-et-Pousser diminue donc significativement l'espace de recherche pour donner un processus d'extraction rapide comme le montre la section expérimentale suivante.

5 Expérimentations

L'objectif de ces expérimentations est de montrer l'efficacité de l'approche Approximer-et-Pousser pour différentes mesures et différents jeux de données. Au-delà de la rapidité, nous souhaitons montrer la faisabilité de notre approche générique. Aussi, nous ne nous comparons pas aux algorithmes de la littérature limités à la seule mesure de fréquence, mais nous confrontons trois stratégies différentes d'extraction des top-k motifs basées sur l'algorithme APRIORI (Agrawal et Srikant, 1994) :

- **Approximer-et-Pousser** : cette stratégie extrait les top-k motifs en s'appuyant sur l'approche Approximer-et-Pousser.
- **Optimale à 50%** : cette stratégie exploite la relaxation anti-monotone de $m(X) \geq \rho$ en fixant le seuil ρ à 50% du seuil idéal $\rho_{k,m}$. Ce seuil idéal est le seuil permettant d'obtenir exactement et directement les top-k motifs. Bien sûr, dans la réalité, ce seuil n'est pas connu et l'utilisateur procède plutôt par tâtonnement à partir de son intuition.
- **Post-traitement** : les motifs sont extraits avec un seuil de fréquence minimale de 10%. Puis, les k motifs maximisant la mesure sont conservés. Le seuil de 10% est un compromis entre faisabilité et exhaustivité (i.e., ne manquer aucun des top-k motifs).

Pour toutes ces expériences, nous utilisons la même implémentation d'APRIORI. Les temps d'extractions sont donc comparables. Toutes les expériences sont effectuées sur un ordinateur doté d'un processeur Xeon 2.2 GHz et de 3GB de mémoire RAM avec le système d'exploitation Linux.

La figure 1 reporte les temps des extractions en fonction du nombre de motifs désirés k pour les jeux de données `mushroom` et `letter` (D.J. Newman et Merz, 1998) (`www.ics.uci.edu/~mlearn/MLRepository.html`). Sur chaque base, deux mesures ont alors été utilisées, à savoir la fréquence et l'aire. En plus, des trois stratégies exposées ci-dessus, nous ajoutons le temps d'extraction optimal comme courbe de référence. Cette valeur de référence consiste à donner directement la relaxation anti-monotone de $m(X) \geq \rho_{k,m}$ pour obtenir exactement les k meilleurs motifs.

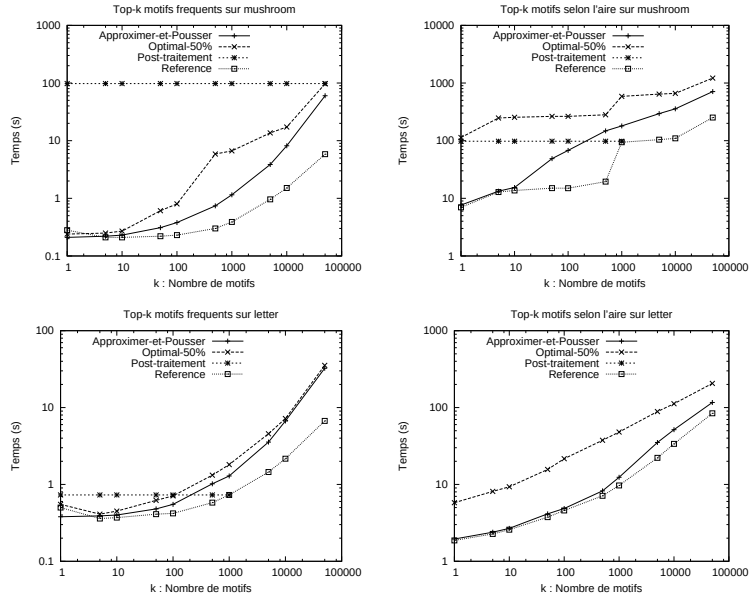

FIG. 1 – *Temps d'extraction des top-k motifs.*

La stratégie Post-traitement se distingue des deux autres car, quelque soit la valeur de k, le temps d'extraction est le même. Le plus souvent cette stratégie est la moins bonne (surtout lorsque k est peu élevé). Dans de rares situations où k est de valeur moyenne, cette stratégie dépasse les deux autres. En revanche, pour des valeurs de k trop grandes, il arrive que cette approche manque des top-k motifs. Cela se traduit par un arrêt des courbes sur les graphiques de la figure 1 car le processus n'effectue plus la tâche demandée. Par exemple, avec le jeu de données `letter`, quelque soit la valeur de k, cette stratégie manque des motifs. Elle ne fournit donc pas toujours le résultat souhaité et échoue parfois en temps.

Il est intéressant de remarquer que, globalement, les deux stratégies Approximer-et-Pousser et Optimale-50% ont le même comportement. Plus le nombre de k motifs à extraire est grand, plus le temps d'extraction augmente. Par ailleurs, lorsqu'une mesure est plus difficile à traiter qu'une autre, elle l'est pour les deux stratégies. Comme attendu, dans tous les cas, la courbe de référence est en deçà des deux stratégies. Un résultat important est que pour toutes les expériences, la stratégie Optimale-50% (pourtant optimiste) a de plus mauvais résultats que

l'approche Approximer-et-Pousser. Notre approche s'intercale donc entre cette stratégie et la courbe de référence. Si le gain par rapport à l'approche Optimale-50% peut parfois être modeste, il peut devenir conséquent dans certaines situations. De plus, l'approche Approximer-et-Pousser évite la fixation hasardeuse du seuil avec les deux autres stratégies.

Dans la pratique, l'utilisateur fixe k relativement bas pour obtenir assez peu de motifs. Or, plus k est petit, plus notre approche est efficace. Cela s'explique par une phase de remplissage rapide des candidats et une approximation immédiate de la contrainte globale. Cette approche novatrice, en laissant à l'utilisateur le choix de la mesure, est donc suffisamment efficace. Par ailleurs, notre approche peut également rechercher les top-k motifs contraints selon une mesure en choisissant un algorithme d'extraction de motifs contraints. Par exemple, une longueur minimale peut être exigée sur les motifs comme c'est le cas pour certaines approches d'extraction des top-k motifs fréquents (Han et al., 2002; Cong, 2001).

6 Conclusion

Nous avons proposé une approche novatrice et efficace pour extraire les top-k motifs selon une mesure d'intérêt formulée par l'utilisateur. Cette méthode est généralisable à d'autres langages ou à des motifs contraints, suivant l'algorithme d'extraction employé. Plus généralement, l'approche Approximer-et-Pousser extrait des motifs satisfaisant une contrainte globale en la relaxant en contrainte locale évolutive exploitable par un algorithme indépendant. Son efficacité réside dans la qualité de l'approximation et sur la manière de l'exploiter pour réduire au mieux l'espace de recherche. Par ailleurs, la collection des motifs candidats tend progressivement vers la solution finale (i.e., les motifs satisfaisant la contrainte globale). À tout moment, le processus peut fournir une solution approchée par le biais de ces motifs. Dans le cas des top-k motifs, l'utilisateur obtiendrait des motifs avec de fortes mesures, mais pas forcément les meilleurs.

Dans de futurs travaux, nous souhaitons porter nos efforts sur la définition de nouvelles contraintes globales pour mieux traduire les attentes des utilisateurs en terme de réduction du nombre de motifs extraits et d'optimisation de leur qualité. Par exemple, pour une mesure d'intérêt, la comparaison d'un motif avec ses spécialisations et ses généralisations immédiates pourrait aboutir à la découverte de motifs plus inattendus au sens de cette mesure. De telles contraintes nécessiteront la ré-utilisation de l'approche Approximer-et-Pousser.

Références

Agrawal, R., T. Imielinski, et A. N. Swami (1993). Mining association rules between sets of items in large databases. In P. Buneman et S. Jajodia (Eds.), *SIGMOD Conference*, pp. 207–216. ACM Press.

Agrawal, R. et R. Srikant (1994). Fast algorithms for mining association rules in large databases. In J. B. Bocca, M. Jarke, et C. Zaniolo (Eds.), *VLDB*, pp. 487–499. Morgan Kaufmann.

Boulicaut, J.-F., A. Bykowski, et C. Rigotti (2003). Free-sets : A condensed representation of boolean data for the approximation of frequency queries. *Data Min. Knowl. Discov. 7*(1), 5–22. Kluwer Academics Publishers.

Cong, S. (2001). Mining the top-k frequent itemset with minimum length m.

D.J. Newman, S. Hettich, C. B. et C. Merz (1998). UCI repository of machine learning databases.

Fu, A. W.-C., R. W. w. Kwong, et J. Tang (2000). Mining *n*-most interesting itemsets. In Z. W. Ras et S. Ohsuga (Eds.), *ISMIS*, Volume 1932 of *Lecture Notes in Computer Science*, pp. 59–67. Springer.

Han, J., J. Wang, Y. Lu, et P. Tzvetkov (2002). Mining top-k frequent closed patterns without minimum support. In *ICDM*, pp. 211–218. IEEE Computer Society.

Hirate, Y., E. Iwahashi, et H. Yamana (2004). TF2P-growth : Frequent itemset mining algorithm without any thresholds. In *Proc. of Workshop on Alternative Techniques for Data Mining and Knowledge Discovery (ICDM'04)*.

Mannila, H. et H. Toivonen (1997). Levelwise search and borders of theories in knowledge discovery. *Data Min. Knowl. Discov. 1*(3), 241–258.

Ng, R. T., L. V. S. Lakshmanan, J. Han, et A. Pang (1998). Exploratory mining and pruning optimizations of constrained association rules. In L. M. Haas et A. Tiwary (Eds.), *SIGMOD Conference*, pp. 13–24. ACM Press.

Ngan, S.-C., T. Lam, R. C.-W. Wong, et A. W.-C. Fu (2005). Mining n-most interesting itemsets without support threshold by the COFI-tree. *Int. J. Business Intelligence and Data Mining 1*(1), 88–106.

Pasquier, N., Y. Bastide, R. Taouil, et L. Lakhal (1999). Discovering frequent closed itemsets for association rules. *Lecture Notes in Computer Science*.

Soulet, A. (2006). *Un cadre générique de découverte de motifs sous contraintes fondées sur des primitives*. Ph. D. thesis, Université de Caen Basse-Normandie, France.

Soulet, A. et B. Crémilleux (2005). Exploiting virtual patterns for automatically pruning the search space. In F. Bonchi et J.-F. Boulicaut (Eds.), *KDID*, Volume 3933 of *Lecture Notes in Computer Science*, pp. 202–221. Springer.

Tzvetkov, P., X. Yan, et J. Han (2003). TSP : Mining top-k closed sequential patterns. In *ICDM*, pp. 347–354. IEEE Computer Society.

Summary

This paper focuses on the mining of patterns satisfying *global* constraints. Comparisons between several patterns are necessary to check them in opposition to the usual constraints and then, global constraints are more complex to mine. Typically, finding the k patterns maximizing a given interestingness measure, i.e. satisfying the top-k constraint, is a hard task. However, this global constraint is very useful in order to mine significant patterns according to a user-specified criterion. In this paper, we propose a generic method for mining patterns under global constraints, named Approximate-and-Push. This method relies on a dynamic relaxation of a global constraint into a local one. Then we applied this approach for mining the top-k patterns with respect to an interestingness measure. Experiments show the efficiency of this Approximate-and-Push approach.