Une nouvelle approche pour la génération efficace des motifs graduels

Durande Kamga Nguifo*,**, Jerry Lonlac Konlac*
Anthony Fleury* Engelbert Mephu Nguifo**

*IMT Nord Europe, Institut Mines-Télécom, Univ. Lille, Centre for Digital Systems,
F-59000 Lille, France
{durande.kamga-nguifo,jerry.lonlac,anthony.fleury}@imt-nord-europe.fr,
**Université Clermont Auvergne, Clermont Auvergne INP, ENSMSE, CNRS, LIMOS,
F-63000 Clermont–Ferrand, France
engelbert.mephu_nguifo@uca.fr,

Résumé. Les motifs graduels mettent en évidence des corrélations entre différents attributs grâce à des règles de la forme "plus/moins X, plus/moins Y" avec X et Y des attributs. Ces motifs représentent des connaissances précieuses pour les experts. Dans la littérature, de nombreuses méthodes permettent de les extraire en s'appuyant sur une représentation binaire des motifs. Bien que certaines de ces méthodes permettent des traitements en parallèle, elles consomment beaucoup de ressources (temps CPU et mémoire). Dans cet article, nous présentons un critère permettant d'élaguer le nombre de motifs candidats entrainant ainsi une réduction de l'espace de recherche. Grâce à des expérimentations menées sur des jeux de données réelles et synthétiques, nous avons comparé l'effet du critère proposé sur les performances de deux algorithmes d'extraction de motifs dans la littérature, GRITE et Paraminer. Les résultats montrent une réduction significative du temps d'exécution.

1 Introduction

De nos jours, la recherche explore dans de nombreux domaines, l'analyse de données à grande échelle. Les capteurs, désormais omniprésents, collectent des données sur divers phénomènes étudiés par des experts, générant ainsi un volume colossal de données numériques. Une grande partie des activités de recherche consiste à corréler différents aspects de ces données afin de comprendre les relations existantes dans les phénomènes observés. L'extraction de motifs fréquents est un sous-domaine du datamining permettant de dégager des connaissances qui permettront aux experts de prendre des décisions adaptées. Ces motifs représentent des abstractions du contenu de vastes ensembles de données, susceptibles de fournir des informations pertinentes. Bien que l'analyse des données quantitatives fait l'objet de nombreux travaux Di-Jorio et al. (2009); Boujike et al. (2023), l'extraction de connaissances utiles, telles que les motifs graduels, demeure une tâche difficile. De nombreux domaines sont concernés par ces enjeux. Par exemple, dans le secteur automobile, les constructeurs peuvent découvrir

que "plus la vitesse du véhicule augmente, plus la consommation de carburant s'accroît, et plus l'efficacité du freinage diminue", ce qui guide la conception de systèmes de sécurité et performants.

La principale difficulté réside dans la taille volumineuse des données, tant en termes de nombre de transactions que d'attributs. Face à cela, l'exploration de toutes les combinaisons d'attributs pour détecter d'éventuelles corrélations s'avère très consommateur de temps et de mémoire. Dans ces travaux, nous proposons une amélioration de deux méthodes existantes(Di-Jorio et al. (2009); Negrevergne et al. (2014)) permettant d'extraire des motifs graduels en réduisant l'espace mémoire utilisé ainsi que le temps d'exécution.

Cet article est organisé comme suit : d'abord, nous commencerons par donner les définitions nécessaires puis nous présenterons quelques techniques d'extraction de motifs graduels. Ensuite, nous introduirons le critère nous permettant d'améliorer ces méthodes. Enfin, nous détaillerons les expérimentations menées, ainsi que les résultats obtenus.

2 Etat de l'art

Soit $\mathcal{D}$ un jeu de données numériques constitué d'un ensemble de transactions $\{t_1, t_2, ..., t_m\}$ décrit par n d'attributs $\{i_1, ..., i_n\}$. Notons par $|\mathcal{D}|$ son nombre de transactions et n son nombre d'attributs. Le tableau 1 présente un jeu de données comportant 3 attributs et 4 transactions.

ID	Prix (€)	Ventes (unités)	Bénéfice (€)
t1	100	50	1000
t2	80	62	900
t3	60	120	800
t4	90	65	950

TAB. 1 – *Jeu de données de ventes $\mathcal{D}$*

2.1 Définitions

Definition 1 (Item graduel & Motif/itemset graduel) Un item graduel est une paire (i, v) dans laquelle i est un attribut et v sa variation associée, avec $v \in \{\uparrow, \downarrow\}$. $\uparrow$ représente une variation croissante tandis que $\downarrow$ représente une variation décroissante. Un motif graduel G est un ensemble d'items graduels. $G = \{(i_1, v_1), ..., (i_n, v_n)\}$.

Notons par $*$ l'opérateur associé à la variation v : si $v = \uparrow$ alors $* = \leq$ sinon $* = \geq$. $(prix, \downarrow)$ est un item graduel extrait du tableau 1 qui peut être interprété comme "la baisse du prix". $G_1 = \{(prix, \downarrow), (ventes, \uparrow)\}$ est un motif graduel interprété comme "plus le prix de vente baisse, plus les ventes augmentent".

Definition 2 (Relation d'ordre entre deux transactions) Soit t_1 et t_2 deux transactions de $\mathcal{D}$ et $G = \{(i_1, v_1), ..., (i_n, v_n)\}$ un motif graduel. t_1 précède t_2 et sera noté $t_1 \ll t_2$ si $\forall k \in [1, n], t_1[i_k] *^k t_2[i_k]$ avec $*^k$ l'opérateur associé à la variation v_k de l'item k de G.

Definition 3 (Liste ordonnée de transactions) Etant donné un motif graduel G, une liste de transactions $\mathcal{L} = < t_1, ..., t_m >$ respecte G si $\forall p \in [1, m-1]$ t_p précède t_{p+1}. Pour le motif $\{(prix, \downarrow), (ventes, \uparrow)\}$, on a $\mathcal{L}_1 = < t_1, t_2, t_3 >$, $\mathcal{L}_2 = < t_1, t_4, t_3 >$.

Definition 4 (Support) Dans la littérature, nous avons relévé deux approches sémantiques permettant de calculer le support d'un motif graduel :la première approche, utilisée par (Di-Jorio et al. (2009); Negrevergne et al. (2014); Do et al. (2015)), définit le support d'un motif graduel comme le ratio entre la taille de la plus longue séquence de transactions qui respecte le motif et le nombre total de transactions. Le calcul du support des motifs graduels implique l'ordonnancement d'au moins deux transactions ou plus, puisque les motifs sont construits suivant la nature croissante ou décroissante des attributs. Le support correspond à la liste $\mathcal{L}_i$ du motif de taille maximale.

$$sup(G) = \frac{\max_{1 \leq i \leq m}(|\mathcal{L}_i|)}{|\mathcal{D}|} \quad | \quad sup(prix, \downarrow), (ventes, \uparrow)) = \frac{|\mathcal{L}_1|}{|\mathcal{D}|} = 0.75$$

La seconde approche, proposée par Laurent et al. (2009), considère le nombre de paires de transactions qui sont concordantes en exploitant la corrélation de rang τ de Kendall. Au lieu de la taille de la plus longue liste ordonnée, elle compte le nombre de paires de transactions qui satisfont l'ordre induit par le motif. Ainsi, le support est défini par la formule suivante :

$$\begin{aligned} sup(G) &= \frac{\left|\left\{(t,t') \in \mathcal{D}^2 \mid t \ll t'\right\}\right|}{\frac{|\mathcal{D}|(|\mathcal{D}|-1)}{2}} \\ sup(G_1) &= \frac{|\{(t_1,t_2),(t_1,t_3),(t_1,t_4),(t_2,t_3),(t_4,t_3)\}|}{\frac{|\mathcal{D}|(|\mathcal{D}|-1)}{2}} \\ &= 0.83 \end{aligned}$$

Definition 5 (Matrice binaire d'ordres) L'ensemble des listes ordonnées d'un motif graduel G peut être représenté par une matrice binaire $M = (m_{i,j})$, où $m_{i,j} \in \{0, 1\}$. Si $t_i \ll t_j$, alors le bit correspondant à la ligne de t_i et à la colonne de t_j est fixé à 1, 0 sinon. Notons par M_i^L la somme des éléments de la ligne i et M_j^C la somme des éléments de la colonne j.

↱	t1	t2	t3	t4
t1	0	1	1	1
t2	0	0	1	0
t3	0	0	0	0
t4	0	1	1	0

(a) $(prix, \downarrow)$

↱	t1	t2	t3	t4
t1	0	1	1	1
t2	0	0	1	1
t3	0	0	0	0
t4	0	0	1	0

(b) $(ventes, \uparrow)$

↱	t1	t2	t3	t4
t1	0	1	1	1
t2	0	0	1	0
t3	0	0	0	0
t4	0	0	1	0

(c) $\{(prix, \downarrow), (ventes, \uparrow)\}$

TAB. 2 – *Matrice binaire de quelques motifs*

Definition 6 (Problème d'extraction de motifs graduels fréquents) Le problème d'extraction de motifs graduels consiste à trouver l'ensemble des motifs graduels fréquents de $\mathcal{D}$ par rapport à un support minimal $minSupp$ défini par l'utilisateur, c'est-à-dire trouver l'ensemble : $\{G \mid sup(G) \geq minSupp\}$

2.2 Techniques d'extraction des motifs graduels

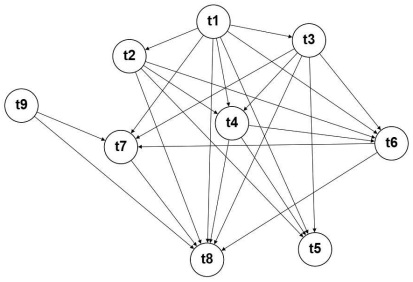
Les méthodes d'extraction de motifs graduels varient en fonction du type de données et de la méthode utilisée pour calculer le support du motif graduel. Comme méthodes nous avons :

GRITE (Extraction d'Itemsets Graduels), proposée par Di-Jorio et al. (2009), qui utilise des graphes de précédence pour extraire des itemsets graduels à partir de bases de données volumineuses. Les objets de la base de données sont représentés comme des nœuds, et les relations de précédence entre eux sont représentées par des liens. Le support d'un motif graduel est calculé en fonction du plus long chemin dans le graphe. Cette méthode permet une extraction efficace des itemsets graduels de taille (k + 1) en combinant ceux de taille k. GRAANK, proposé par Laurent et al. (2009), calcule le support des itemsets graduels en utilisant le coefficient de Kendall, qui mesure la cohérence des paires de transactions. Owuor et al. (2021) a proposé une amélioration heuristique de cette approche, explorant des techniques d'optimisation métaheuristiques comme les colonies de fourmis, les algorithmes génétiques et l'optimisation par essaims de particules afin de réduire le temps de calcul. Ces méthodes génèrent trop de motifs redondants(des des motifs apparaissant dans d'autres motifs avec le même support), rendant difficile leur analyse. Une solution efficace consiste à générer des motifs graduels fermés pour réduire et cibler les motifs les plus pertinents, tout en conservant la possibilité de régénérer l'ensemble des motifs. Paraminer, proposé par Negrevergne et al. (2014), est un algorithme générique et parallèle pour l'extraction de motifs fermés. Il repose sur l'énumération de motifs dans des systèmes d'ensembles fortement accessibles et utilise des techniques de réduction et de projection pour accélérer le calcul dans des architectures multi-cœurs. Ces méthodes représentent les motifs en utilisant la matrice binaire d'ordre ; cette dernière comporte le plus souvent des transactions inutiles occupant de la mémoire et ralentissant le temps d'extraction de ces derniers. Dans la section suivante, nous proposons un critère d'élagage permettant de ne conserver que les transactions utiles pour la suite du traitement.

3 Extraction des motifs graduels avec stratégie d'élagage

Propriété 1 *Soit G'' un itemset graduel généré par la jointure de deux itemsets graduels G et G'. La matrice binaire de G'' est obtenue de la manière suivante : $M'' = M$ **ET** M' avec M et M' les matrices binaires de G et G'. Cette propriété repose sur l'opérateur binaire **ET** qui offre de bonnes performances de calcul Di-Jorio et al. (2009).*

Considérons le jeu de données du tableau 1, la matrice binaire du motif graduel $\{(prix, \downarrow)$, $(ventes, \uparrow)\}$ est obtenue en combinant celle des motifs $(prix, \downarrow)$ et $(ventes, \uparrow)$. Ces matrices deviennent de plus en plus creuse au fur et à mesure que la taille des motifs augmente. Afin de rendre la matrice moins creuse, Di-Jorio et al. (2009) propose de supprimer certaines informations inutiles, à savoir les transactions isolées représentant des tuples qui n'ont aucune relation. Ces dernieres ne peuvent pas contribuer à rendre le motif fréquent et sont donc supprimées. Ce critère n'est pas suffisant car une transaction ne contenant qu'une seule valeur non nulle est conservée bien qu'elle ne contienne aucune information. Etant donné un support minimal $minsup = 5$ et le motif G provenant de la combinaison de deux motifs graduels ayant pour matrice binaire celle présente sur figure 1, nous remarquons la présence de deux chemins de taille maximale (<t1,t2,t4,t6,t7,t8>, <t1,t3,t4,t6,t7,t8>) supérieur au support minimal. Les transactions t9 et t5 n'apparaissant pas dans l'un de ces deux chemins, ne seront pas utile pour la suite mais seront cependant conservées dans (Di-Jorio et al. (2009); Negrevergne et al. (2014)) car elles possèdent des valeurs non nulles dans la matrice binaire.



(a) Graphe associé

↱	t1	t2	t3	t4	t5	t6	t7	t8	t9
t1	0	1	1	1	1	1	1	1	0
t2	0	0	0	1	1	1	0	1	0
t3	0	0	0	1	1	1	1	1	0
t4	0	0	0	0	1	1	0	1	0
t5	0	0	0	0	0	0	0	0	0
t6	0	0	0	0	0	0	1	1	0
t7	0	0	0	0	0	0	0	1	0
t8	0	0	0	0	0	0	0	0	0
t9	0	0	0	0	0	0	1	1	0

(b) Matrice binaire d'ordre

FIG. 1 – *Représentation du motif graduel G*

Proposition 1 *Une transaction t_p est conservée dans la matrice binaire d'un motif G si $M_p^L + M_p^C \geq minsup - 1$*

Preuve 1 *Soit le motif G et L = $< t_1, ..., t_p, t_{minsup}, ..., t_k >$ la plus longue séquence obtenue à partir de la représentation binaire de G. Comme $< t_1, ..., t_p, t_{minsup}, ..., t_k >$ respecte le motif G, il existe une relation d'ordre ($\geq$ ou $\leq$) entre chacun de ces éléments se traduisant par un arc dans le graphe. Une transaction apparaît en position p ssi* ***(i)*** *elle est précédée d'au moins p-1 transactions (nombre de "1" sur la colonne) c'est-à-dire. $M_p^C \geq p - 1$ et* ***(ii)*** *il est suivi d'au moins $minsup - p$ éléments c'est-à-dire : $M_p^L \geq minsup - p$. (i) + (ii) $M_p^C + M_p^L \geq minsup - 1$*

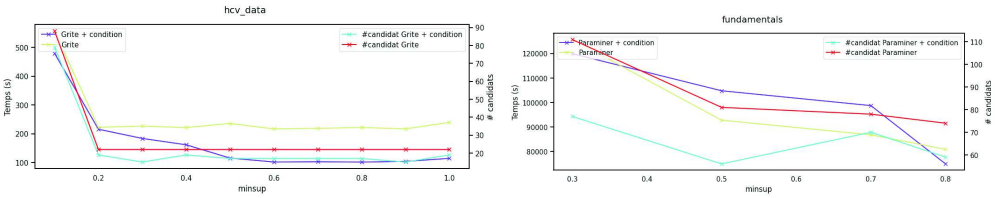
Ainsi, lors de la création de la matrice binaire du motif G de la figure 1, avant le calcul du support pour valider G, nous allons calculer la somme de chaque ligne et colonne de sa matrice binaire et les renseigner dans le tableau 3. Nous remarquons que les transactions t5 et t9 se retrouvent avec un total de 4 et 2 ce qui est inférieur au support minimal, par conséquent ces transactions seront supprimées. La proposition 1 permet de faire apparaître des transactions inutiles. Ce critère réduit la taille mémoire de la représentation binaire du motif sans impacter la complexité de l'algorithme de fouille de motif car ces derniers effectuent des sommes en ligne et en colonne pour rechercher puis supprimer les transactions n'ayant que des valeurs nulles.

	t1	t2	t3	t4	t5	t6	t7	t8	t9
M_p^C	0	1	1	3	4	4	4	7	0
M_p^L	7	4	5	3	0	2	1	0	2
$M_p^C + M_p^L$	7	5	6	6	**4**	6	5	7	**2**

TAB. 3 – *Vérification des objets à conserver*

Jeux de données	Domaine	Dimension (ligne x colonne)
AUS_hmd_period_summary [1]	Économie	200x20
hcv [2]	Synthétique	251x12
fundamentals [3]	Économie	300x35
EFWData2023 [4]	Économie	1782x35
downld02 [5]	Météo	601x27

TAB. 4 – *Jeux de données*



(a) HCV (b) fundamentals

FIG. 2 – *Temps d'exécution et candidats générés pour ces jeux de données*

4 Expérimentations et résultats

Nous avons comparé les performances des algorithmes Paraminer et GRITE en évaluant l'impact de la condition d'élagage, en utilisant une implémentation C++ pour Paraminer et Python pour GRITE, sur une machine équipée d'un processeur Intel Xeon Silver, doté de 64 cœurs et de 512 Go de mémoire vive, fonctionnant sous Ubuntu 22.04.4 LTS. Nous avons utilisé des données synthétiques et réelles provenant des domaines tels que la météo et l'économie. Tous ces jeux de données proviennent de Negrevergne et al. (2014); Clémentin et al. (2021). Les caractéristiques de ceux-ci sont données dans la table 4. Le code source de nos expérimentations intégrant le critère proposé aux algorithmes GRITE et Paraminer, peut être obtenu à partir de https://github.com/kndbvortex/graduel-optimise.git.

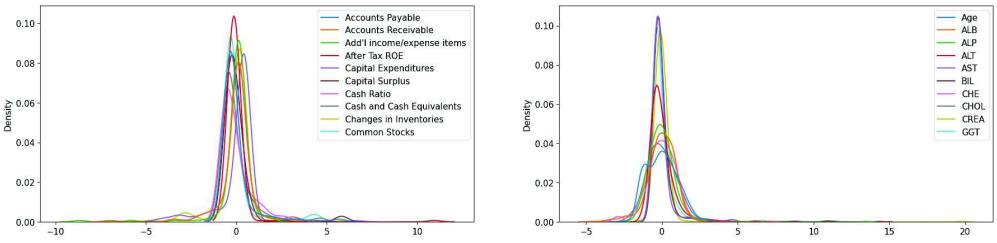
4.1 Résultats et discussions

Pour chaque jeu de données, nous avons évalué les performances de GRITE et Paraminer en mesurant le temps d'exécution et le nombre de candidats générés, avec et sans l'ajout du critère proposé. Nous avons fait varié le support minimal entre les valeurs suivantes $[0.3, 0.5, 0.7, 0.8]$. Les algorithmes d'extraction de motifs ont été exécutés pour chaque seuil, permettant de collecter les données relatives au nombre de candidats et aux temps de traitement. L'ensemble de ces résultats est accessible sur GitHub.

La figure 2a présente les résultats de GRITE sur le jeu de données hcv, on note une réduction importante du temps d'exécution et du nombre de candidat généré lorsque le seuil est inférieur à 0.5. La figure 2b présente les résultats de Paraminer sur le jeu de données fundamentals, Ici nous avons des améliorations lorsque le support minimal vaut 0.3 et 0.8; Sans le critère proposé, Paraminer présente généralement un temps d'exécution plus court sur ce jeu de données.

Les expérimentations nous montrent une réduction du nombre de candidats entrainant une réduction effective de l'espace de recherche. GRITE et Paraminer évaluent systématiquement plus de candidats sans le critère proposé. Le nombre de candidats tend à diminuer avec l'augmentation du seuil. Le nombre de candidats évalués, et le temps d'exécution des algorithmes avec et sans le critère est pratiquement identique en dessous d'un support minimal de 0.3.

Nous avons noté une réelle amélioration des temps d'exécution sur les jeux de données HCV, downld02, AUS_hmd_period et test. Bien que ces jeux de données proviennent de domaines variés, nous notons ici que plusieurs attributs de chacun de ces jeux de données sont moins dispersé par rapport à la valeur centrale (moyenne) et présente très peu de points allant au-delà des $quartiles \pm 1.5 * IQR$[6] dans les attributs. La figure 3a présente la distribution des attribus du jeu de données hcv où l'on améliore GRITE et Paraminer. L'on remarque que 8 sur 11 attributs ont des valeurs regroupées autour de la moyenne. A contrario, sur les jeux données fundamentals et EFWData2023, nous n'avons pas noté une amélioration du temps d'exécution malgré une diminution du nombre de candidats générés. Nous pensons que cela peut-être dû à la distribution des données, à une forte présence de données s'éloignant fortement du premier et troisième quartile. La visualisation de ces données a été faite à l'aide de boxplot où elles sont soit inférieures à $q_1 - 1.5 * IQR$ soit supérieures à $q_3 + 1.5 * IQR$ avec q_1 le premier quartile, q_3 le troisième quartile et IQR la distance interquartile ($IQR = q_3 - q_1$).



(a) hcv (b) fundamentals

FIG. 3 – *Distribution des attributs de deux jeux données*

5 Conclusion

Dans ce travail, nous proposons un critère permettant d'évaluer la pertinence de conserver une transaction dans la représentation binaire d'un motif graduel. Ce critère vise à réduire efficacement le nombre de motifs graduels candidats pour lesquels le support doit être calculé. Il permet de réduire l'espace de recherche en éliminant au plus tôt les motifs graduels candidats peu prometteurs. Des expérimentations menées sur plusieurs bases de données réelles et synthétiques ont montré que l'intégration de la stratégie proposée aux algorithmes d'extraction de motifs graduels efficaces améliore leurs performances en termes de temps d'exécution et de mémoire. Les expérimentations montrent également que ce critère ne garantit pas forcément une réduction du temps d'exécution en fonction de la distribution des valeurs des attributs dans la base de données. Cette distribution pourrait nous guider sur l'utilisation du critère pro-

6. distance inter-quartile

posé. Nous souhaiterons combiner ce travail à un autre critère proposé par Lonlac et al. (2024) permettant de faire un lien entre les approches GRITE et GRAANK.

6 Remerciements

Ce travail de recherche est effectué dans le cadre d'un contrat doctoral. Nous souhaitons vivement remercier l'Université Clermont Auvergne et l'IMT Nord Europe pour leur soutien financier, les auteurs des jeux de données utilisés dans ce travail, ainsi que les relecteurs pour leurs retours constructifs.

Références

Boujike, M. C., J. Lonlac, N. Tsopzé, E. Mephu Nguifo, et L. P. Fotso (2023). GRAPGT : gradual patterns with gradualness threshold. *Int. J. Gen. Syst. 52*(5), 525–545.

Clémentin, T. D., T. F. L. Cabrel, et K. E. Belise (2021). A novel algorithm for extracting frequent gradual patterns. *Machine Learning with Applications 5*, 100068.

Di-Jorio, L., A. Laurent, et M. Teisseire (2009). Mining Frequent Gradual Itemsets from Large Databases. In *Advances in Intelligent Data Analysis VIII*, pp. 297–308.

Do, T. D. T., A. Termier, A. Laurent, B. Négrevergne, B. Omidvar-Tehrani, et S. Amer-Yahia (2015). PGLCM : efficient parallel mining of closed frequent gradual itemsets. *Knowl. Inf. Syst. 43*(3), 497–527.

Laurent, A., M.-J. Lesot, et M. Rifqi (2009). GRAANK : Exploiting rank correlations for extracting gradual itemsets. In *Flexible Query Answering Systems*, pp. 382–393.

Lonlac, J., B. F. Tchide, A. Bomgni, A. Doniec, et E. Mephu Nguifo (2024). Revisiting frequent (closed) gradual itemsets mining. In *IEEE 36th ICTAI*, pp. 913–920.

Negrevergne, B., A. Termier, M.-C. Rousset, et J.-F. Méhaut (2014). ParaMiner : a generic pattern mining algorithm for multi-core architectures. *Data Min Knowl Disc 28*, 593–633.

Owuor, D. O., T. Runkler, A. Laurent, J. O. Orero, et E. O. Menya (2021). Ant colony optimization for mining gradual patterns. *Int. J. Mach. Learn. & Cyber. 12*(10), 2989–3009.

Summary

Gradual patterns highlight correlations between attributes through rules such as "the more/less X, the more/less Y" where X and Y are features. These patterns represent valuable knowledge for experts. In the literature, numerous methods allow their extraction based on a binary representation introduced in the GRITE algorithm. Although some of these methods allow for parallel processing, they consume significant memory and require substantial execution time. In this paper, we present a criterion that reduces the number of candidate patterns and, consequently, their execution time. Through experiments conducted on real and synthetic data, we compared the effect of the proposed criterion on the performance of the GRITE and Paraminer algorithms. The results show a significant improvement.