Une approche de clustering conceptuel via des k-motifs relaxés fréquents

Motaz Ben Hassine[*,**], Saïd Jabbour[*], Mourad Kmimech[***], Badran Raddaoui[****], Mohamed Graiet[**]

[*]CRIL, Université d'Artois & CNRS, Lens, France
{benhassine,jabbour}@cril.fr
[**]Université de Monastir, UR-OASIS-ENIT, Monastir, Tunisie
{motaz.benhassine@fsm,mohamed.graiet@isimm}.u-monastir.tn
[***]EFREI, Université Paris-Panthéon-Assas, France
mourad.kmimech@efrei.fr
[****]SAMOVAR, Télécom SudParis, Institut Polytechnique de Paris, France
badran.raddaoui@telecom-sudparis.eu

Résumé. Cet article présente une nouvelle approche basée sur l'exploitation d'un modèle de motifs appelé *k-motif relaxé fréquent* pour le clustering conceptuel, obtenue en généralisant la notion de la couverture d'un motif. Pour l'extraction de ces motifs, nous utilisons une traduction vers le problème de la satisfiabilité propositionnelle. Par la suite, nous adoptons une approche de programmation linéaire en nombres entiers pour déterminer l'ensemble des clusters disjoints. Enfin, nous démontrons l'efficacité de notre approche à travers des expérimentations sur plusieurs ensembles de données transactionnelles réels connus.

1 Introduction

De nombreuses approches de clustering de données ont été proposées dans la littérature, notamment des techniques d'apprentissage automatique non supervisées telles que k-Means MacQueen (1967) et sa variante pour les clusters chevauchants, Neo-k-Means Whang et al. (2015). D'autres méthodes incluent le clustering spectral Shi (2003), BIRCH Zhang et al. (1996) et le clustering agglomératif Hartigan et Algorithms (1975). Cet article s'intéresse particulièrement aux approches de clustering conceptuel. Certaines techniques s'appuient sur des méthodes déclaratives, telles que la satisfiabilité booléenne (SAT) Métivier et al. (2012), la programmation par contraintes (CP) Guns et al. (2013); Laghzaoui et Lebbah (2023) et la programmation linéaire en nombres entiers (PLNE) Ouali et al. (2016, 2017); Dao et al. (2018). Les méthodes de clustering conceptuel reposent sur l'extraction de motifs Cependant, ces motifs classiques peuvent ne pas couvrir quelques transactions, ce qui limite de trouver des clusters de haute qualité. Pour remédier à ce **problème**, nous proposons de relaxer ces motifs pour améliorer la qualité des clusters. Nous présentons une approche en deux phases : d'abord, le modèle des k-motifs relaxés fréquents (k-MRF) avec un encodage basé sur SAT pour énumérer ces motifs, utilisé précédemment dans le contexte des règles d'association Jabbour et al. (2023). Ensuite, en utilisant des modèles PLNE, nous sélectionnons les meilleurs clusters.

L'article est organisé comme suit : la Section 2 offre un aperçu de la logique propositionnelle, de la satisfiabilité propositionnelle et du clustering conceptuel. La Section 3 présente notre motivation et discute des modèles k-MRF et PLNE. La Section 4 expose nos expérimentations sur des ensembles de données réels. Enfin, la Section 5 conclut l'article. Ce travail est une **version traduite** de l'article **accepté** à ECAI 2024 (**Hassine et al. (2024)**).

2 Préliminaires

2.1 Logique propositionnelle & SAT

On considère un langage propositionnel $\mathcal{L}$, construit à partir d'un ensemble dénombrable $\mathcal{PS}$ de lettres propositionnelles, des constantes booléennes $\top$ (vrai) et $\bot$ (faux), et des connecteurs logiques $\{\neg, \wedge, \vee, \rightarrow, \leftrightarrow\}$. Les symboles x, y, z, etc., itèrent $\mathcal{PS}$, et les formules propositionnelles sont notées Φ, Ψ, Γ. Un littéral est soit une variable propositionnelle (x), soit sa négation ($\neg x$). Une clause disjonctive est une disjonction finie de littéraux, et une clause conjonctive est leur conjonction. Une clause formé par un seul littéral est appelée unitaire. Pour une formule Φ, $\mathcal{Z}(\Phi)$ désigne les symboles de $\mathcal{PS}$ présents dans Φ. Une formule en forme normale conjonctive (CNF) est une conjonction de clauses disjonctives, et une forme normale disjonctive (DNF) est une disjonction de clauses conjonctives. Une interprétation booléenne $\mathcal{I}$ d'une formule Φ associe à $\mathcal{Z}(\Phi)$ des valeurs $\{0, 1\}$. Si $\mathcal{I}(\Phi) = 1$, $\mathcal{I}$ est un modèle de Φ. Le problème de satisfiabilité propositionnelle (SAT) consiste à déterminer si une formule CNF admet un modèle. Le problème SAT est connu comme étant $\mathcal{NP}$-complet.

2.2 Problème de clustering conceptuel

Soit Ω un ensemble d'éléments représentant des articles, des pages web, des attributs, etc., qui peuvent être représentés par des lettres a, b, c, etc. Un **motif classique** (ou *itemset*, ou *pattern* en anglais) est un sous-ensemble non vide de Ω, soit $P \subseteq \Omega$. L'ensemble des motifs sur Ω, noté 2^{Ω}, sont représenté par P, Q, R, etc. Une **transaction** notée T, est un ensemble d'items de Ω, c'est-à-dire $T = \{b_1, b_2, \ldots, b_n\}$, avec $T \subseteq \Omega$. Une **donnée transactionnelle** est un ensemble fini de transactions $\mathcal{D} = \{T_1, T_2, \ldots, T_m\}$. La **couverture** de P dans $\mathcal{D}$ associe chaque motif à l'ensemble des transactions contenant P, soit $\mathsf{C}(P, \mathcal{D}) = \{i \in [1..m] \mid T_i \in \mathcal{D} \text{ et } P \subseteq T_i\}$. Le **support** de P, noté $\mathsf{Supp}(P, \mathcal{D}) = |\mathsf{C}(P, \mathcal{D})|$. P est appelé un **motif clos ou fermé** si et seulement si $\nexists\ R$ tel que $P \subset R$ et $\mathsf{Supp}(R, \mathcal{D}) = \mathsf{Supp}(P, \mathcal{D})$. Un **concept** est une paire (O, P) où $O \subseteq \mathcal{D}$ et $P \subseteq T_i$, $\forall\ T_i \in O$. Le **problème de clustering conceptuel** consiste à trouver $\beta > 1$ clusters disjoints $Cl = \{O_1, O_2, \ldots, O_\beta\}$ qui couvrent $\mathcal{D}$ et correspondent à des concepts.

3 Approche de clustering conceptuel via k-MRF

3.1 Motivation

Notre travail est motivé par la rigidité inhérente des motifs classiques dans la résolution du problème de clustering conceptuel. Pour illustrer, considérons le suivant décrit par l'exemple 1.

Exemple 1. *Considérons la donnée transactionnelle $\mathcal{D}$ du Tableau 1, avec un nombre cible de clusters souhaités $\beta = 2$, soit $\{T_1, T_2, T_3, T_4\}$ et $\{T_5, T_6, T_7, T_8\}$. Avec des motifs classiques, prenons $P = \{d\}$ et $R = \{e\}$, chacun avec un support de 3. Cependant, P et R couvrent respectivement $\{T_2, T_3, T_4\}$ et $\{T_6, T_7, T_8\}$, laissant T_1 et T_5 non couverts ce qui pose problème.*

Transactions	Items							
T_1	a	b	c					
T_2	a	b		d				
T_3	a		c	d				
T_4		b	c	d				
T_5						f	g	h
T_6					e		g	h
T_7					e	f		h
T_8					e	f	g	

TAB. 1 – *Une donnée transactionnelle simple $\mathcal{D}$*

3.2 k-motifs relaxés fréquents

Pour résoudre ce problème, nous revisitons d'abord le modèle de motif traditionnel en rappelant les concepts de k-couverture et de k-support auparavant définit par Jabbour et al. (2023).

Définition 1. *Soit $\mathcal{D}$ une donnée transactionnelle et k un entier positif. Alors, la k-**couverture** d'un motif P est $\mathsf{C}^k(P, \mathcal{D}) = \{i \in [1..m] | T_i \cap P \neq \emptyset$ et $|P \setminus T_i| \leq k\}$. Le k-**support** de P est défini comme suit :* $\mathsf{Supp}^k(P, \mathcal{D}) = |\mathsf{C}^k(P, \mathcal{D})|$.

Contrairement au modèle traditionnel qui exige l'inclusion complète d'un motif dans une transaction, la k-couverture permet jusqu'à k éléments manquants dans une transaction pour qu'elle soit considérée comme couverte par le motif. Le k-support indique combien de transactions de $\mathcal{D}$ appartiennent k-couverture.

Définition 2. *Soit $\mathcal{D}$ une donnée transactionnelle et α un seuil de support minimal t.q. $\alpha > 0$. Alors, le motif P est appelé k-**motif relaxé fréquent** (k-MRF) ssi $\mathsf{Supp}^k(P, \mathcal{D}) \geq \alpha$.*

Nous caractérisons un k-MRF comme **clos** par rapport au k-support ssi pour tout $P \subset R$, $\mathsf{Supp}^k(R, \mathcal{D}) < \mathsf{Supp}^k(P, \mathcal{D})$.

Nous présentons l'encodage SAT pour calculer les k-MRF (clos) à partir des données. D'abord, nous établissons un lien entre les modèles SAT et les k-MRF (clos), en associant chaque item $a \in \Omega$ et chaque transaction $T_i \in \mathcal{D}$ à des variables propositionnelles x_a et o_i. Ensuite, nous introduisons une formule propositionnelle avec des contraintes garantissant une correspondance exacte avec l'ensemble des k-MRF.

Contrainte de couverture. La première contrainte *Contrainte de Couverture* est formellement exprimée comme suit :

$$\bigwedge_{T_i \in D} (o_i \leftrightarrow (\sum_{a \in \Omega \setminus T_i} x_a \leq k)) \tag{1}$$

La contrainte (1) assure que une transaction T_i est couverte par le k-MRF lorsque au maximum on a k items de k-MRF qui ne sont pas présent dans T_i.

Contrainte de fréquence. La deuxième contrainte est la *Contrainte de fréquence* est formellement exprimée comme suit :

$$\sum_{i=1}^{m} o_i \geq m \times \alpha \tag{2}$$

La contrainte (2) assure que le k-MRF couvre au moins $m \times \alpha$ transactions. Où α est un pourcentage.

Contrainte de clôture. La troisième contrainte, appelée *Contrainte de Clôture*, est exprimée comme suit :

$$\bigwedge_{a \in \Omega} (\neg x_a \rightarrow \bigvee_{T_i \in D, a \in T_i} (o_i \wedge \sum_{b \in \Omega \setminus T_i} x_b = k)) \tag{3}$$

La contrainte (3) garantit que l'élément a ne peut pas faire partie du k-MRF candidat si son inclusion viole la contrainte de couverture dans au moins une transaction.

Contrainte de fréquence des items. La quatrième contrainte, appelée *Contrainte de Fréquence des Items*, est exprimée comme suit :

$$\bigwedge_{T_i \in D} (x_a \rightarrow (\sum_{T_i \in D \mid a \in T_i} o_i \geq \gamma)) \tag{4}$$

La contrainte (4) garantit que chaque item apparaît au moins γ pour cent dans un k-MRF. Où γ est un pourcentage aussi comme α.

Proposition 1. *La formule CNF $\Phi^{\alpha,\gamma,k} = (1) \wedge (2) \wedge (3) \wedge (4)$ représente le résultat de traduction des contraintes pseudo-booléenes pour le problème d'extraction des k-MRF clos.*

Il important de noter que lorsque $k = 0$, il n'y a pas de relaxation. En effet, les motifs 0-MRF sont identiques aux motifs clos classiques.

Exemple 2. *Considérons la même donnée transactionnelle $\mathcal{D}$ présenté dans le Tableau 1. Prenons $k = 1$, $\alpha = 37.5\%$, $\gamma = 37.5\%$ et $P = \{a, b, c, d\}$ et $Q = \{e, f, g, h\}$, tous deux étant des 1-MRFs clos. Alors, les clusters couverts respectivement par P et Q sont $\{T_1, T_2, T_3, T_4\}$ et $\{T_5, T_6, T_7, T_8\}$, représentant les meilleurs clusters souhaités qui couvrent toutes les données.*

3.3 Modèles de programmation linéaire en nombres entiers pour le clustering conceptuel

Pour notre problème, suivant Ouali et al. (2016), le clustering conceptuel est modélisé en PLNE. Nous maximisons le nombre de transactions couvertes par un k-MRF, sous deux contraintes. Premièrement, chaque transaction T_i doit être couverte par un seul k-MRF, mais comme ces derniers offrent une couverture plus large que les motifs clos, nous limitons un peu cette couverture avec un seuil nommé `ILP-cover-threshold` σ, où $\sigma < k$. Deuxièmement, le nombre de clusters est fixé à β_0. Ce modèle est appelé **M1**. Nous étudions ensuite l'impact de la relaxation de la contrainte (1) de M1 sur le nombre de solutions optimales. Un second modèle, **M2**, est proposé, autorisant un chevauchement des transactions. M2 reprend

la fonction, les variables et la contrainte (2) de M1, mais la contrainte (1) est modifiée pour imposer que chaque transaction soit couverte par au plus θ k-MRFs. Les modèles M1 et M2 sont définis comme suit :

$$\begin{array}{lll} \texttt{Maximiser} & \sum_{c \in C} v_c \cdot y_c & \\ \texttt{Sous contraintes} & (1) \quad \sum_{c \in C} a_{T_i,c} \cdot y_c = 1, \quad \forall\, T_i \in \mathcal{D} & \\ & (2) \quad \sum_{c \in C} y_c = \beta_0 & \\ & y_c \in \{0,1\}, c \in C & \\ & i = 1, \ldots, m & \end{array} \qquad \begin{array}{lll} \texttt{Maximiser} & \sum_{c \in C} v_c \cdot y_c & \\ \texttt{Sous contraintes} & (1) \quad \sum_{c \in C} a_{T_i,c} \cdot y_c \leq \theta, \quad \forall\, T_i \in \mathcal{D} & \\ & (2) \quad \sum_{c \in C} y_c = \beta_0 & \\ & y_c \in \{0,1\}, c \in C & \\ & i = 1, \ldots, m & \end{array}$$

où $\mathcal{D}$ est une donnée transactionnelle avec m transactions, chacune est décrite par f items. Soit C l'ensemble des p k-MRF. Soit $a_{T_i,c}$ une matrice binaire $m \times p$ où $a_{T_i,c} = 1$ ssi $|c \backslash T_i| \leq \sigma$ où $c \in C$. De plus, nous utilisons p variables booléennes y_c, où $y_c = 1$ ssi le cluster représenté par le k-MRF clos c appartient au clustering. La fonction objective est définie en associant à chaque cluster c une valeur v_c reflétant le nombre de transactions couvertes par c, qui doit être maximisée.

4 Expérimentations

Pour démontrer l'efficacité de notre approche proposée, nous avons réalisé des expérimentations sur divers données transactionnelles réels bien connues [1], comme présenté dans le Tableau 2.

$\mathcal{D}$	#Transactions	#Items	Densité (%)
Lymph	148	68	40
Mushroom	8124	119	18
Primary-Tumor	336	31	48
Soybean	630	50	32
Tic-tac-toe	958	27	33
Vote	435	48	33
Zoo-1	101	36	44

TAB. 2 – *Données transactionnelles réelles.*

Nous avons d'abord calculé les motifs clos classiques (MCC) puis les k-MRF ($k \geq 1$), en fixant $k = 1$ pour limiter le nombre de motifs. Pour énumérer les k-MRF, nous avons utilisé le solveur MiniSAT Eén et Sörensson (2003) modifié en C++. Ensuite, en exploitant les motifs énumérés, nous avons identifié les clusters avec des modèles PLNE décrits dans la sous-section 3.3. L'implémentation [2] est réalisée en Python v3.9. Nous avons fixé σ à 0 et limité l'exécution à 7 heures, toutes les expériences ayant été effectuées sur un MacBook Air avec 16 Go de RAM. Nous avons appliqué nos k-MRF au modèle M1, en faisant varier le support minimal α de 10% à 40% et β de 3 à 30 (28 variations). Avec α fixe à 10%, nous avons appelé cette approche `CCA-`k`-MRF-M1`, comparée à `MCC-M1` et à d'autres méthodes de clustering partitionnelles. Nous avons également appliqué nos k-MRF avec $\alpha = 10\%$ au

1. Les données ont été collectés à partir de l'UCI repository et sont disponibles à l'adresse suivante : https ://dtai.cs.kuleuven.be/CP4IM/datasets/.

2. le code source est disponible à l'adresse suivante : https ://zenodo.org/records/13285611

modèle M2, nommé CCA−k−MRF−M2, et comparé à MCC−M2 en faisant varier θ de 2 à 5 pour observer l'impact de la relaxation de la contrainte (1) sur le nombre de solutions optimales. En outre, nous avons comparé CCA−k−MRF−M2 avec Neo−β−means. Toutes les comparaisons ont été faites en termes de temps d'exécution pour trouver la solution optimale dans les modèles PLNE, du nombre de solutions optimales et de la qualité des clusters. Pour évaluer la qualité des clusters, nous avons utilisé la Similarité Intra-Cluster (ICS), basée sur la mesure de similarité de Jaccard, calculée comme suit : Étant donné deux transactions T_i et T_j où $i \neq j$ et $i, j \in [1, \ldots, m]$, nous avons $s : \mathcal{D} \times \mathcal{D} \mapsto [0, 1]$, $s(T_i, T_j) = \frac{|T_i \cap T_j|}{|T_i \cup T_j|}$. Alors :

$$ICS(c_1, \ldots, c_\beta) = \frac{1}{2} \sum_{1 \leq r \leq \beta} \Big(\sum_{T_i, T_j \in c_r} s(T_i, T_j) \Big)$$

$\mathcal{D}$	α	k-MRF			MCC		
		#k-MRF	#Sol	CPU_M	#MCC	#Sol	CPU_M
Lymph	10%	3605378	**27**	3760.40	51862	8	**62.85**
	20%	759630	**27**	415.81	13934	2	**2.37**
	30%	202602	**27**	74.12	4910	1	**0.92**
	40%	60470	0	-	2058	0	-
Mushroom	10%	128962	**27**	930.68	3287	6	**34.21**
	20%	19712	**27**	74.59	817	1	**6.03**
	30%	4055	0	-	293	**1**	**3.06**
	40%	1135	0	-	107	0	-
Primary-Tumor	10%	256991	**27**	42.71	32183	7	**19.70**
	20%	76081	**27**	13.89	9891	2	**6**
	30%	30372	**27**	8.88	3614	1	**1.42**
	40%	14778	0	-	1382	0	-
Soybean	10%	69191	**27**	16.19	2907	6	**2.57**
	20%	11900	0	-	844	**2**	**0.53**
	30%	3383	0	-	380	0	-
	40%	1484	0	-	205	0	-
Tic-tac-toe	10%	4479	**28**	19.94	191	2	**0.17**
	20%	811	**28**	1.45	26	1	**0.10**
	30%	171	0	-	18	0	-
	40%	15	0	-	5	0	-
Vote	10%	280386	**28**	67.38	37399	3	**14.93**
	20%	34098	0	-	7227	0	-
	30%	6606	0	-	658	0	-
	40%	693	0	-	79	0	-
Zoo-1	10%	92711	**28**	4.71	3291	7	**0.32**
	20%	35081	**28**	2.02	1743	2	**0.23**
	30%	12614	**28**	0.86	818	1	**0.14**
	40%	3555	**28**	**0.28**	316	0	-

TAB. 3 – *k-MRF sur M1 vs. les motifs clos classique sur M1*

$\mathcal{D}$	θ	CCA-k-MRF-M2		MCC-M2	
		#Sol	CPU_M	#Sol	CPU_M
Lymph	2	28	4156.28	28	**2536.88**
	3	**28**	3519.44	26	**351.28**
	4	**28**	6115.981	26	**623.40**
	5	**28**	3998.53	24	**218.67**
Mushroom	2	**28**	**355.77**	2	15244.37
	3	**28**	362.99	1	**195.29**
	4	**28**	355.79	1	**187.84**
	5	**28**	332.93	1	**196.36**
Primary-Tumor	2	28	**47.96**	28	198.12
	3	28	**48.54**	28	183.98
	4	28	**59.25**	28	159.136
	5	28	**52.73**	28	123.17
Soybean	2	28	**11.97**	28	137.062
	3	28	**11.72**	28	100.11
	4	28	**12.72**	28	80.66
	5	28	**10.68**	28	73.489
Tic-tac-toe	2	**28**	**27.01**	27	1693.54
	3	**28**	**22.46**	27	447.67
	4	**28**	**19.60**	25	227.98
	5	**28**	**16.24**	24	221.14
Vote	2	28	**77.77**	28	2716.59
	3	28	**103.17**	28	2660.54
	4	28	**531.33**	28	1436.24
	5	28	**240.68**	28	963.59
Zoo-1	2	28	5.12	28	**0.61**
	3	28	5.27	28	**0.53**
	4	28	5.25	28	**0.54**
	5	28	5.24	28	**0.53**

TAB. 4 – *CCA-k-MRF-M2 vs. MCC-M2*

Pour comparer avec d'autres approches, nous avons utilisé les paramètres suivants : Pour le modèle M1, nous avons fixé $\alpha = 10\%$ et $\beta = 30$. Pour l'approche β-Means, nous avons utilisé l'encodage one-hot, afin qu'elle soit adaptée aux données qualitatives. Pour M2, α et β ont été maintenus les mêmes que dans M1, tandis que le paramètre de relaxation $\theta = 2$. Le tableau 3 montre les résultats pour les motifs classiques et relaxés appliqués au modèle M1, en se concentrant sur les solutions optimales. Comme attendu, l'augmentation de α réduit le nombre de k-MRFs et de motifs clos. Pour les motifs classiques, le nombre de solutions optimales ne dépasse pas 8 pour plusieurs ensembles de données, tandis qu'avec les motifs relaxés, on en trouve jusqu'à 28, notamment pour Zoo−1, Vote, et Tic−tac−toe. Le tableau 3 présente aussi le temps d'exécution moyen pour trouver ces solutions. Le temps de résolution de M1 en utilisant les motifs classiques est plus faible que celui en utilisant les k-MRFs, car les k-MRFs trouvent plus de solutions, ce qui allonge le temps d'exécution moyen (CPU_M). Le tableau 4 présente le résultat de la comparaison de CCA−k−MRF−M2 avec MCC−M2. Notre méthode trouve des solutions optimales pour toutes les variations de θ et β, alors que MCC−M2 échoue pour certaines valeurs de β, notamment sur Lymph, Mushroom et Tic−tac−toe. De plus, CCA−k−MRF−M2 surpasse MCC−M2 en temps de résolution sur plusieurs ensembles de données, comme Primary−Tumor, Soybean, Tic−tac−toe et Vote. Les résultats

du tableau 5 montrent que notre approche dépasse MCC−M1 en qualité de clustering sur toutes les données. De plus, CCA−k−MRF−M1 est généralement plus rapide que MCC−M1 sur plusieurs ensembles. Le tableau 7 confirme que notre méthode surpasse les autres méthodes de clustering en qualité sur toutes les datasets, mais en termes de temps, l'approche agglomérative AC est la plus rapide, sauf pour Mushroom, où β−Means l'emporte. Le tableau 6 montre que notre approche surpasse largement Neo−β−Means en qualité, bien que Neo−β−Means soit plus rapide en temps d'exécution.

$\mathcal{D}$	CCA-k-MRF-M1			MCC-M1		
	$k=1, \beta=30, \alpha=10\%$			$\beta=30$		
	#k-MRF	ICS	CPU	#MCC	ICS	CPU
Lymph	3605378	**3723.17**	3305.40	154220	277.93	**270.67**
Mushroom	128962	**5652568.44**	**932.69**	221524	-	-
Primary-	256991	**15984.25**	**44.11**	87230	1895.89	241.95
Soybean	69191	**40133.02**	**14.23**	31759	10795.60	87.50
Tic-tac-toe	4479	**47165.55**	**19.52**	42711	29278.23	1466.06
Vote	280386	**16992.90**	**68.75**	227031	8563.77	1268.51
Zoo-1	92711	**768.37**	4.74	4567	267.79	**0.422**

TAB. 5 – *CCA-k-MRF-M1 vs. MCC-M1*

$\mathcal{D}$	CCA-k-MRF-M2		Neo-β-Means	
	ICS	CPU	ICS	CPU
Lymph	**7470.67**	3288.04	461	**0.21**
Mushroom	**12259500.13**	332.13	1967290.99	**7.36**
Primary-Tumor	**36752.61**	50.74	2965.39	**0.16**
Soybean	**71741.53**	13.12	6344.39	**0.35**
Tic-tac-toe	**92792.95**	34.43	10917.74	**0.58**
Vote	**31761.66**	77.37	6083.92	**0.27**
Zoo-1	**1657.55**	5.01	285.84	**0.06**

TAB. 6 – *CCA-k-MRF-M2 vs. Neo-β-Means pour $\beta = 30$ et $\theta = 2$.*

$\mathcal{D}$	CCA-k-MRF-M1		β-Means		BIRCH		SPECTRAL		AC	
	ICS	CPU	ICS	CPU	ICS	CPU	ICS	CPU	ICS	CPU
Lymph	**3723.17**	3305.40	252.80	0.45	257.89	0.04	452.48	0.429	250.86	**0.008**
Mushroom	**5652568.44**	932.69	883931.50	**3.17**	926729.75	15.90	1442391.09	100.98	1072855.32	11.32
Primary-Tumor	**15984.25**	44.11	1745.32	0.47	2030.70	0.06	5632.47	0.76	1792.95	**0.014**
Soybean	**40133.02**	14.23	5067.89	0.64	5433.03	0.182	17066	1.14	5040.35	**0.04**
Tic-tac-toe	**47165.55**	19.52	7751.78	0.89	7049.87	0.32	7353.16	2.07	7059.29	**0.08**
Vote	**16992.90**	68.75	2794.59	0.51	3189.58	0.12	14807.83	0.86	2533.51	**0.02**
Zoo-1	**768.37**	4.74	218.60	0.39	248.17	0.02	467.03	0.23	164.77	**0.005**

TAB. 7 – *CCA-k-MRF-M1 vs. autres approches de clustering pour $\beta = 30$.*

5 Conclusion

Dans cet article, nous avons présenté une nouvelle approche de clustering conceptuel basée sur les k-MRFs. Nous avons d'abord utilisé l'encodage SAT, déjà employé pour l'extraction de règles d'association, afin d'énumérer les k-MRFs. Ensuite, des modèles PLNE ont servi à identifier les meilleurs clusters. Enfin, une évaluation expérimentale a démontré l'efficacité de l'utilisation des k-MRFs par rapport à l'utilisation des motifs clos classiques.

Références

Dao, T.-B.-H., C.-T. Kuo, S. Ravi, C. Vrain, et I. Davidson (2018). Descriptive clustering : Ilp and cp formulations with applications. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pp. 1263–1269.

Eén, N. et N. Sörensson (2003). An extensible sat-solver. In *International conference on theory and applications of satisfiability testing*, pp. 502–518. Springer.

Guns, T., S. Nijssen, et L. De Raedt (2013). k-pattern set mining under constraints. *IEEE Transactions on Knowledge and Data Engineering 25*(2), 402–418, doi: 10.1109/TKDE.2011.204.

Hartigan, J. et H. C. Algorithms (1975). John wiley & sons : Hoboken. *NJ, USA*.

Hassine, M. B., S. Jabbour, M. Kmimech, B. Raddaoui, et M. Graiet (2024). On the discovery of conceptual clustering models through pattern mining. In *ECAI 2024*, Volume 392 of *Frontiers in Artificial Intelligence and Applications*, pp. 1648–1655. IOS Press.

Jabbour, S., B. Raddaoui, et L. Sais (2023). A symbolic approach to computing disjunctive association rules from data. In *Thirty-Second International Joint Conference on Artificial Intelligence {IJCAI-23}*, pp. 2133–2141. International Joint Conferences on Artificial Intelligence Organization.

Laghzaoui, M. E. A. et Y. Lebbah (2023). A constraint programming approach for quantitative frequent pattern mining. *International Journal of Data Mining, Modelling and Management 15*(3), 297–311.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability/University of California Press*.

Métivier, J.-P., P. Boizumault, B. Crémilleux, M. Khiari, et S. Loudni (2012). Constrained clustering using sat. In *Advances in Intelligent Data Analysis XI : 11th International Symposium, IDA 2012, Helsinki, Finland, October 25-27, 2012. Proceedings 11*, pp. 207–218. Springer.

Ouali, A., S. Loudni, Y. Lebbah, P. Boizumault, A. Zimmermann, et L. Loukil (2016). Efficiently finding conceptual clustering models with integer linear programming. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pp. 647–654. IJCAI/AAAI Press.

Ouali, A., A. Zimmermann, S. Loudni, Y. Lebbah, B. Crémilleux, P. Boizumault, et L. Loukil (2017). Integer linear programming for pattern set mining ; with an application to tiling. In *Advances in Knowledge Discovery and Data Mining : 21st Pacific-Asia Conference, PAKDD 2017*, pp. 286–299. Springer.

Shi (2003). Multiclass spectral clustering. In *Proceedings ninth IEEE international conference on computer vision*, pp. 313–319. IEEE.

Whang, J. J., I. S. Dhillon, et D. F. Gleich (2015). Non-exhaustive, overlapping k-means. In *Proceedings of the 2015 SIAM international conference on data mining*, pp. 936–944. SIAM.

Zhang, T., R. Ramakrishnan, et M. Livny (1996). Birch : an efficient data clustering method for very large databases. *ACM sigmod record 25*(2), 103–114.

Summary

This article presents a novel approach based on the use of novel patterns called k-relaxed frequent patterns for conceptual clustering. To enumerate these patterns, we use a translation to the SAT problem. Subsequently, we adopt an ILP approach to identify the set of disjoint clusters. Finally, we demonstrate the effectiveness of our approach through several experiments on well-known real transactional datasets.