Approche ensemble pour le co-clustering par blocs sur des données textuelles: Application au biomédical

Séverine Affeldt*, Lazhar Labiod*, Mohamed Nadif*

*Université de Paris, CNRS, Centre Borelli, F-75006 Paris
<prénom.nom>@u-paris.fr

Résumé. Nous proposons un co-clustering par blocs via une approche ensemble qui fusionne plusieurs co-clusterings élémentaires en une matrice d'affinité *consensus* structurée. Les co-clusterings de base sont issus des mêmes données textuelles et générés par la même méthode de co-clustering. Ce processus de fusion renforce la qualité individuelle des co-clusterings par blocs au sein d'une seule matrice consensus. Notre approche permet un co-clustering complètement non supervisé, où le nombre de co-clusters est automatiquement déduit d'un critère de modularité non trivial généralisé. La fonction objective associée permet l'apprentissage conjoint de l'agrégation des co-clusterings élémentaires et du co-clustering consensus. Les résultats expérimentaux sur plusieurs jeux de données réelles démontrent l'intérêt de notre approche comparée à des méthodes compétitives de co-clustering (Affeldt et al., 2020).

1 Introduction

La classification non supervisée ou *clustering* permet le regroupement d'objets similaires en *clusters* homogènes. C'est une approche incontournable dans la science des données et elle est particulièrement utile pour le traitement des données massives. Le choix d'un algorithme de clustering efficace pour l'obtention d'un partitionnement *profitable* n'est pas une tâche simple. En effet, plusieurs algorithmes de clustering ne produisent pas nécessairement la même partition optimale. L'approche *ensemble*, très populaire en apprentissage supervisé, s'avère un très bon moyen pour pallier cet inconvénient. Ainsi, des méthodes proposant une partition *consensus* ont été proposées pour améliorer la pertinence du partitionnement (Strehl et Ghosh, 2003). Toutefois, de telles méthodes ne sont pas conçues pour des données telles que les données textuelles, notamment biomédicales, qui sont généralement clairsemées (*sparse*) et de très grande dimension.

Pour ce type de données, le *co-clustering* ou classification croisée (Govaert et Nadif, 2013) permet, cependant, d'exploiter efficacement la relation naturellement pré-existante entre l'ensemble des *objets* (eg. documents) et leurs *caractéristiques* (eg. termes). Ainsi à partir des matrices document-terme et une classification simultanée des deux ensembles, on obtient généralement une meilleure réorganisation en blocs que par les approches de clustering appliquées séparément sur les deux ensembles (Labiod et Nadif, 2011; Ailem et al., 2016; Salah et Nadif, 2017; Govaert et Nadif, 2018). Pour un corpus de documents, chaque co-cluster fournit un

regroupement des documents *et* une caractérisation des thématiques grâce au regroupement simultané des mots. La réduction de dimension qui s'opère implicitement à chaque itération d'un co-clustering rend ce type d'approche très performante pour des données massives. Peu d'approches *ensemble* pour le co-clustering ont été développées (Hanczar et Nadif, 2012). Récemment, Huang et al. (2015) ont conçu SCCE (*Spectral Co-Clustering Ensemble*) qui se présente comme un problème de partitionnement de graphe bipartite. Plus récemment, Yu et al. (2019) ont proposé CoCE (*Co-Clustering ensemble*), dont la complexité en temps est très inférieure à celle de SCCE. Elle reste toutefois trop élevée pour des données textuelles, généralement clairsemées et de très grande dimension.

Notre contribution La robustesse des approches *ensemble* pour le clustering et les bonnes performances des méthodes de co-clustering pour les données textuelles, nous amène aujourd'hui à proposer une approche ensemble pour le co-clustering par blocs, EBCO (*Ensemble Block CO-clustering*) (Affeldt et al., 2020).

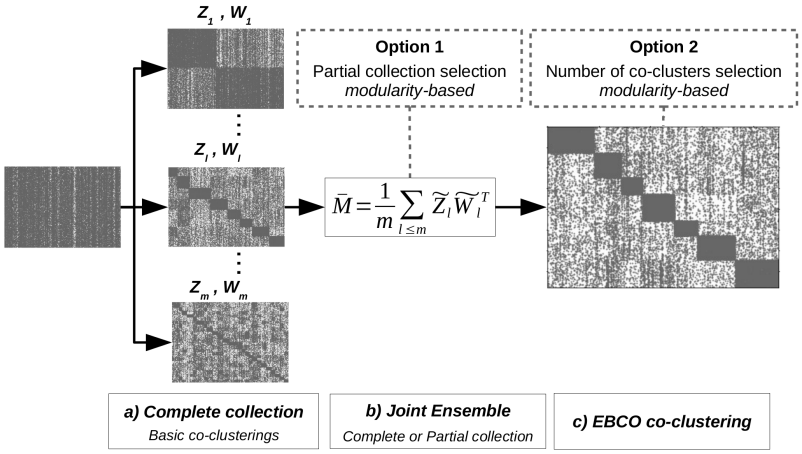


FIG. 1: EBCO : (a) Collection de m co-clusterings $(\mathbf{Z}_\ell, \mathbf{W}_\ell)_{\ell \in [1,m]}$ issus d'une matrice document-terme. (b) Matrice d'affinité intégrant les informations des co-clusterings de base via $\overline{\mathbf{M}}$. (c) Factorisation de la matrice $\overline{\mathbf{M}}$ en un co-clustering consensus par blocs.

La Figure 1 résume notre méthode. EBCO fournit un co-clustering *consensus* par blocs à partir de co-clusterings élémentaires. Notre approche se démarque des méthodes existantes à trois niveaux. Tout d'abord, en combinant les multiples co-clusterings élémentaires dans l'esprit d'un double k-means, EBCO présente une faible complexité en temps. De plus, EBCO prend en compte la nature *directionnelle* des données textuelles. Enfin, notre méthode est complètement non supervisée. Elle permet de choisir automatiquement les co-clusterings de base les mieux adaptés *et* le nombre de co-clusters consensus (Fig. 1 ; Options 1 & 2).

2 Ensemble co-clustering par blocs

Soit une matrice document-terme $\mathbf{X} = (x_{ij})$ de taille $n \times d$, où $x_{ij} \in \mathbb{N}$ est la fréquence du mot j dans le document i. La partition de l'ensemble des documents I en g clusters

peut être représentée par la matrice de classification $\mathbf{Z} = (z_{ik}) \in \{0,1\}^{n \times g}$ pour laquelle $\forall i, \sum_{k=1}^{g} z_{ik} = 1$. De même, pour la partition de l'ensemble des mots J, on considère la matrice de partition $\mathbf{W} = (w_{jk}) \in \{0,1\}^{d \times g}$ pour laquelle $\forall j, \sum_{k=1}^{g} w_{jk} = 1$.

Définition du problème et fonction objective Les méthodes de *block seriation* réorganisent I et J selon des blocs diagonaux dont les partitions décrivent de façon optimale $\mathbf{Z}$ et $\mathbf{W}$. On peut concevoir le co-clustering comme une tâche de *block seriation* $\mathbf{Q} = (\mathbf{q}_{ij})$ définie sur $I \times J$ par $\mathbf{Q} = \mathbf{Z}\mathbf{W}^\top$ où $\mathbf{q}_{ij} = 1$ si le document i a les mêmes attributs de bloc que j, sinon $\mathbf{q}_{ij} = 0$. Ainsi, $\mathbf{q}_{ij} = \sum_{k=1}^{g} \mathbf{z}_{ik}\mathbf{w}_{jk} = (\mathbf{Z}\mathbf{W}^\top)_{ij}$. Toutefois, l'approche de *block seriation* n'est pas pondérée par la taille des clusters lignes et des clusters colonnes, ce qui implique qu'un cluster peut être extrêmement petit s'il est affecté par des *outliers*. Nous proposons avec `EBCO` une nouvelle relation pondérée de *block seriation*,

$$\widetilde{\mathbf{q}}_{ij} = \sum_{k=1}^{g} \frac{\mathbf{z}_{ik}\mathbf{w}_{jk}}{\sqrt{\mathbf{z}_{.k}\mathbf{w}_{.k}}} = \sum_{k=1}^{g} \tilde{\mathbf{z}}_{ik}\tilde{\mathbf{w}}_{jk} = (\widetilde{\mathbf{Z}}\widetilde{\mathbf{W}}^T)_{ij} \tag{1}$$

où $\widetilde{\mathbf{Z}} = \mathbf{Z}\mathbf{D}_{\mathbf{z}}^{-0.5}$, $\widetilde{\mathbf{W}} = \mathbf{W}\mathbf{D}_{\mathbf{w}}^{-0.5}$, et la taille des clusters $\mathbf{Z}$ et $\mathbf{W}$ sont sur la diagonale de $\mathbf{D}_{\mathbf{z}} = \mathbf{Z}^\top\mathbf{Z}$ et $\mathbf{D}_{\mathbf{w}} = \mathbf{W}^\top\mathbf{W}$. Pour une approche de type ensemble, nous pouvons exploiter l'Eq. (1) afin de combiner des co-clusterings. Soit $\mathcal{M} = \{(\mathbf{Z}_\ell, \mathbf{W}_\ell)\}_{\ell \in [1;m]}$ une collection de co-clusterings obtenus par un même algorithme. Chaque co-cluster peut être modélisé par une relation de *block seriation* pondérée, avec $\widetilde{\mathbf{M}}_l = \widetilde{\mathbf{Z}}_l\widetilde{\mathbf{W}}_l^\top$. La matrice d'affinité *consensus* $\widetilde{\mathbf{Q}}$, telle que chaque $\widetilde{\mathbf{M}}_l$ puisse être modélisé par $\widetilde{\mathbf{M}}_l = \widetilde{\mathbf{Q}} + E_l$, peut alors résulter de la minimisation de $\sum_{l=1}^{m} D(\widetilde{\mathbf{M}}_l, \widetilde{\mathbf{Q}})$, où D est une fonction de coût qui quantifie la qualité de l'approximation de $\widetilde{\mathbf{M}}_l$ par $\widetilde{\mathbf{Q}}$. La fonction objective de `EBCO` peut être ainsi définie par,

$$\min_{\widetilde{\mathbf{Q}}} \mathcal{J}_{EBCO}(\overline{\mathbf{M}}, \widetilde{\mathbf{Q}}) \equiv \min_{\widetilde{\mathbf{Q}}} ||\overline{\mathbf{M}} - \widetilde{\mathbf{Q}}||_F^2. \tag{2}$$

où $||.||_F^2$ correspond à la norme de Frobenius. Il est aisé de montrer que la solution optimale $\widetilde{\mathbf{Q}}^*$ de l'Eq. (2) est la matrice d'affinité moyenne $\overline{\mathbf{M}} = \frac{1}{m}\sum_{l=1}^{m} \widetilde{\mathbf{M}}_l$,

Optimisation et algorithme On peut montrer que l'optimisation de la fonction objective de `EBCO` présente l'équivalence suivante (voir Affeldt et al., 2020, Proposition 3.1),

$$\min_{\widetilde{\mathbf{Q}}} ||\overline{\mathbf{M}} - \widetilde{\mathbf{Q}}||_F^2 \equiv \max_{\widetilde{\mathbf{Q}}} Tr(\overline{\mathbf{M}}\widetilde{\mathbf{Q}}^\top) \equiv \max_{\widetilde{\mathbf{Z}}, \widetilde{\mathbf{W}}} Tr(\overline{\mathbf{M}}\widetilde{\mathbf{W}}\widetilde{\mathbf{Z}}^\top) \equiv \max_{\widetilde{\mathbf{Z}}, \widetilde{\mathbf{W}}} Tr(\widetilde{\mathbf{Z}}^\top\overline{\mathbf{M}}\widetilde{\mathbf{W}}). \tag{3}$$

Les propriétés $Tr(AB) = Tr(BA)$ et $Tr(A^\top) = Tr(A)$, A matrice carrée, impliquent,

$$(a) \quad Tr(\widetilde{\mathbf{Z}}^\top\overline{\mathbf{M}}\widetilde{\mathbf{W}}) = Tr(\mathbf{Z}^\top\overline{\mathbf{M}}\widetilde{\mathbf{W}}\mathbf{D}_{\mathbf{z}}^{-0.5}) = \langle\overline{\mathbf{M}}\widetilde{\mathbf{W}}, \mathbf{Z}\rangle_{\mathbf{D}_{\mathbf{z}}^{-0.5}} \tag{4}$$

$$(b) \quad Tr(\widetilde{\mathbf{Z}}^\top\overline{\mathbf{M}}\widetilde{\mathbf{W}}) = Tr(\mathbf{W}^\top\overline{\mathbf{M}}^\top\widetilde{\mathbf{Z}}\mathbf{D}_{\mathbf{w}}^{-0.5}) = \langle\overline{\mathbf{M}}^\top\widetilde{\mathbf{Z}}, \mathbf{W}\rangle_{\mathbf{D}_{\mathbf{w}}^{-0.5}}. \tag{5}$$

Après initialisation de $\mathbf{Z}$ et $\mathbf{W}$, les paramètres peuvent être mis à jour itérativement en maximisant alternativement les termes à droite des Eq. (4 & 5). Cette alternance exploite bien l'interaction document/terme dans le cadre du texte. L'Algorithme 1 détaille la procédure alternée.

Algorithm 1 Ensemble Block Co-Clustering (EBCO).

Input : $\mathcal{M} = \{(\mathbf{Z}_l, \mathbf{W}_l); l = 1, \ldots, m\}$ une collection de co-clusterings,
Output : co-clustering $(\mathbf{Z}, \mathbf{W})$, g nombre de co-clusters
Initialisation :
a) Sélection d'un bon sous-ensemble de $\mathcal{M}$ et calcul de $\overline{\mathbf{M}}$
b) Initialisation au hasard de $\mathbf{W}$ and $\mathbf{Z}$ [1].
repeat
1. Affectation des documents : • $\mathbf{Z} \leftarrow Binmax(\overline{\mathbf{M}}\widetilde{\mathbf{W}}\mathbf{D}_{\mathbf{z}}^{-0.5})$
2. Affectation des mots : • $\mathbf{W} \leftarrow Binmax(\overline{\mathbf{M}}^\top\widetilde{\mathbf{Z}}\mathbf{D}_{\mathbf{w}}^{-0.5})$

until convergence de $J_{EBCO}(\overline{\mathbf{M}}, \widetilde{\mathbf{Q}}) = ||\overline{\mathbf{M}} - \widetilde{\mathbf{Q}}||_F^2$

Mise à jour de Z : à $\mathbf{W}$ fixée, $\mathbf{Z}$ est obtenue par maximisation de $\langle\overline{\mathbf{M}}\widetilde{\mathbf{W}}, \mathbf{Z}\rangle_{\mathbf{D}_{\mathbf{z}}^{-0.5}} = \sum_{i,k} z_{ik} \frac{1}{\sqrt{z_{.k}}} \widetilde{\mathbf{w}}_k^\top \overline{\mathbf{m}}_i$. La mise à jour de $\mathbf{Z}$ correspond alors à $\mathbf{Z} = Binmax$ [2]$(\overline{\mathbf{M}}\widetilde{\mathbf{W}}\mathbf{D}_{\mathbf{z}}^{-0.5})$ c'est-à-dire $\forall i, z_{ik} = \text{argmax}_{k'} \frac{1}{\sqrt{z_{.k'}}} \widetilde{\mathbf{w}}_{k'}^\top \overline{\mathbf{m}}_i \in \{0, 1\}$.

Mise à jour de W : à $\mathbf{Z}$ fixée, $\mathbf{W}$ est obtenue par maximisation de $\langle\overline{\mathbf{M}}^\top\widetilde{\mathbf{Z}}, \mathbf{W}\rangle_{\mathbf{D}_{\mathbf{w}}^{-0.5}} = \sum_{j,k} w_{jk} \frac{1}{\sqrt{w_{.k}}} \widetilde{\mathbf{z}}_k^\top \overline{\mathbf{m}}^j$. La mise à jour de $\mathbf{Z}$ correspond alors à $\mathbf{W} = Binmax(\overline{\mathbf{M}}^\top\widetilde{\mathbf{Z}}\mathbf{D}_{\mathbf{w}}^{-0.5})$ c'est-à-dire $\forall j, w_{jk} = \text{argmax}_{k'} \frac{1}{\sqrt{w_{.k'}}} \widetilde{\mathbf{z}}_{k'}^\top \overline{\mathbf{m}}^j \in \{0, 1\}$.

On peut noter que $\overline{\mathbf{M}}\widetilde{\mathbf{W}}\mathbf{D}_{\mathbf{z}}^{-0.5}$ et $\overline{\mathbf{M}}^\top\widetilde{\mathbf{Z}}\mathbf{D}_{\mathbf{w}}^{-0.5}$ sont les projections des documents et des mots dans un espace de dimension inférieure. Par ailleurs, on observe le principe de *conscience mechanism* [3] émanant ici des matrices diagonales $\mathbf{D}_{\mathbf{z}}^{-0.5}$ and $\mathbf{D}_{\mathbf{w}}^{-0.5}$. Enfin, EBCO est efficace en terme de calcul et on peut montrer que sa complexité est en $O(n \cdot it \cdot (2g))$ où it est le nombre d'itérations, qui est faible (environ quelques dizaines).

3 Une approche ensemble co-clustering non supervisée

Nous proposons une approche *ensemble* de co-clustering non supervisée en nous basant sur un critère de modularité non trivial (Ailem et al., 2016). Etant donné une matrice d'affinité $\overline{\mathbf{M}}$ définie sur $I \times J$, dans le cadre de la tâche de co-clustering, on considère la mesure de modularité généralisée,

$$\mathcal{J}_{Mod}(\mathbf{Q}) = \frac{1}{2|E|} \sum_{i=1}^{n} \sum_{j=1}^{n} (\overline{m}_{ij} - \frac{\overline{m}_{i.}\overline{m}_{.j}}{2|E|})\mathbf{q}_{ij}. \tag{6}$$

où $2|E| = \sum_{i,j} \overline{m}_{ij} = \overline{m}_{..}$ est le poids total des liens, $\overline{m}_{i.} = \sum_j \overline{m}_{ij}$ - le degré de i et $\overline{m}_{.j} = \sum_i \overline{m}_{ij}$- le degré de j. Cette mesure de modularité prend la forme matricielle suivante,

$$\mathcal{J}_{Mod}(\mathbf{Q}) = \frac{1}{2|E|} Tr[(\overline{\mathbf{M}} - \delta)^\top \mathbf{Q}] \text{ where } \delta_{ij} = \frac{\overline{m}_{i.}\overline{m}_{.j}}{\overline{m}_{..}}. \tag{7}$$

1. L'initialisation de $\mathbf{W}$ et $\mathbf{Z}$ peut par exemple être faite avec un spherical k-means.
2. Soit $\mathbf{A} = (a_{ik}) \in \{0, 1\}^{n \times g}$ avec $\forall i, \sum_k a_{ik} = 1$ et $\mathbf{B} = (b_{ik}) \in \mathbb{R}^{n \times g}$, alors $\mathbf{A} \leftarrow Binmax(\mathbf{B})$ signifie $\forall i, a_{ik} = \text{argmax}_{k'} b_{ik'}, k' = 1, \ldots, g$.
3. Quand des clusters sont par nature très déséquilibrés, le mécanisme de conscience a un effet de régularisation qui permet d'échapper à un mauvais minimum local où certains clusters sont très grands/petits ou même vides.

La fonction objective de Eq. (7) est linéaire par rapport à $\mathbf{Q}$, et les contraintes que $\mathbf{Z}$ doit respecter correspondent à des équations linéaires. Il est donc en théorie possible de résoudre ce problème de façon exacte. Toutefois, il s'agit d'un problème NP complet qui nécessite des heuristiques. Dans l'Eq. (7), si $\delta = 0$ et si on considère la *block seriation* pondérée $\widetilde{\mathbf{Q}}$ au lieu de la *block seriation* binaire $\mathbf{Q}$, alors $\mathcal{J}_{Mod}$ est équivalente à $\mathcal{J}_{EBCO}$. EBCO peut donc être considérée comme une relaxation du critère de modularité qui prend en compte la taille des clusters en lignes et en colonnes suivant un problème d'optimisation qui peut être traité.

Soit une collection de m co-clusterings de base issus d'un même jeu de données et pour chacun desquels on calcule la modularité. La pertinence des co-clusterings est fortement corrélée avec la valeur maximale de la modularité. Notre approche *ensemble* ne considère donc que les co-clusterings de base ayant une forte modularité. EBCO peut ainsi automatiquement garder les co-clusterings les plus intéressants pour le consensus final (Fig. 1, *Option 1*). EBCO réalise aussi un co-clustering de la matrice consensus pour un nombre de co-clusters qui varie de 2 à K et calcule la modularité. Le nombre optimal de co-clusters étant fortement corrélé à la valeur maximale de la modularité (Ailem et al., 2016), EBCO s'appuie sur cette valeur pour définir automatiquement le nombre final de co-clusters (Fig. 1, *Option 2*).

4 Expérimentations

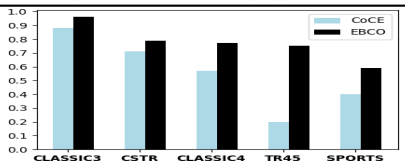
EBCO est comparée à des méthodes compétitives de type *diagonal co-clustering* telles que DCC (Salah et Nadif, 2017), CoClustMod (Ailem et al., 2016), CoClustSpecMod (Labiod et Nadif, 2011) ou non-diagonal CROINFO (Govaert et Nadif, 2018) et la méthode de *co-clustering ensemble* CoCE (Yu et al., 2019). Les évaluations sont faites sur un large éventail de jeux de données textuelles réelles (Fig. 2, Table, *Caractéristiques*), ayant parfois un fort déséquilibre de classes (faible coefficient de *Balance*). Les données forment une matrice document-terme $\mathbf{X}$ où x_{ij} indique le nombre d'occurrences du mot j dans le document i pondéré selon la méthode TF-IDF[4]. Les évaluations s'appuient sur l'ARI et le NMI[5]. Les labels ne sont connus que pour les documents. Mais, la partition des mots étant associée à celle des documents ; la qualité du regroupement des documents nous informe sur celle des mots.

Nous évaluons d'abord EBCO dans sa version *supervisée* (Fig. 2, Table, g fixé). EBCO surpasse les autres méthodes sur tous les jeux de données avec une augmentation moyenne de $0,128$ pour l'ARI et de $0,098$ pour la NMI. Nos expériences montrent également la bonne capacité de EBCO à traiter les clusters fortement déséquilibrés (eg. SPORTS, TR45). On peut aussi noter les très bonnes performances de EBCO par rapport à CoCE (Fig. 2, (a,b)).

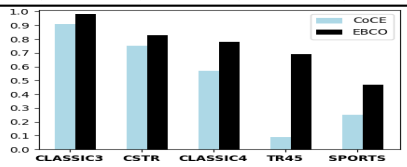
L'évaluation de EBCO avec sélection automatique des co-clusterings de base (Fig. 1, *Option 1*) a montré que les co-clusterings de base avec une modularité supérieure ou égale à 80% de la modularité maximale dans la collection garantissent de bons résultats (voir Affeldt et al., 2020, Section 6.5). Ce seuil est donc préconisé pour tout nouveau jeu de données. On peut alors évaluer EBCO sous sa forme complètement non supervisée, c'est-à-dire en laissant l'approche identifier également le nombre de co-clusters final (Fig. 1, *Option 2*). Les résultats sont très bons à la fois concernant le nombre de co-clusters (Table 1, $g^\star$) et le partitionnement des matrices document-terme (Table 1, $\text{EBCO}_{80\%}$). Il existe toutefois une exception pour PUBMED10, avec un nombre de co-clusters sous-estimé.

4. TF-IDF : Term frequency-inverse document frequency
5. ARI : Adjusted Rand Index ; NMI : Normalized Mutual Information

Caractéristiques #doc.	#mots	*Sparsité* (%)	Balance	g	Données		DCC	CoClust -Mod	CoClust -SpecMod	CROINFO	**EBCO**
8580	14870	99.14	0.036	7	SPORTS	NMI	0.57 ± 0.01	0.53 ± 0.04	0.45 ± 0.00	0.57 ± 0.03	$\mathbf{0.59 \pm 0.01}$
						ARI	0.39 ± 0.01	0.45 ± 0.06	0.30 ± 0.00	0.45 ± 0.04	$\mathbf{0.47 \pm 0.06}$
690	8261	96.60	0.088	10	TR45	NMI	0.69 ± 0.02	0.49 ± 0.03	0.71 ± 0.02	0.54 ± 0.03	$\mathbf{0.75 \pm 0.01}$
						ARI	0.56 ± 0.04	0.42 ± 0.02	0.66 ± 0.05	0.44 ± 0.06	$\mathbf{0.69 \pm 0.02}$
15565	22437	99.72	0.093	10	PUBMED10	NMI	0.56 ± 0.01	0.49 ± 0.02	0.36 ± 0.00	0.56 ± 0.03	$\mathbf{0.63 \pm 0.01}$
						ARI	0.44 ± 0.01	0.44 ± 0.04	0.18 ± 0.00	0.46 ± 0.04	$\mathbf{0.52 \pm 0.02}$
6279	31472	99.52	0.281	6	LA12	NMI	0.52 ± 0.02	0.42 ± 0.02	0.35 ± 0.00	0.45 ± 0.04	$\mathbf{0.59 \pm 0.01}$
						ARI	0.45 ± 0.03	0.41 ± 0.03	0.17 ± 0.00	0.38 ± 0.05	$\mathbf{0.57 \pm 0.02}$
7094	5896	99.41	0.323	4	CLASSIC4	NMI	0.72 ± 0.00	0.67 ± 0.03	0.41 ± 0.00	0.66 ± 0.06	$\mathbf{0.77 \pm 0.01}$
						ARI	0.71 ± 0.01	0.61 ± 0.05	0.22 ± 0.00	0.58 ± 0.11	$\mathbf{0.78 \pm 0.01}$
475	1000	96.60	0.399	4	CSTR	NMI	0.68 ± 0.01	0.66 ± 0.05	0.76 ± 0.00	0.69 ± 0.02	$\mathbf{0.79 \pm 0.00}$
						ARI	0.61 ± 0.05	0.70 ± 0.05	0.80 ± 0.00	0.70 ± 0.07	$\mathbf{0.83 \pm 0.01}$
3891	4303	98.95	0.710	3	CLASSIC3	NMI	0.94 ± 0.00	0.94 ± 0.00	0.91 ± 0.00	0.94 ± 0.00	$\mathbf{0.96 \pm 0.00}$
						ARI	0.97 ± 0.00	0.97 ± 0.00	0.94 ± 0.00	0.96 ± 0.00	$\mathbf{0.98 \pm 0.00}$



(a) CoCE *vs.* EBCO, **NMI**



(b) CoCE *vs.* EBCO, **ARI**

FIG. 2: NMI et ARI moyens pour le clustering de documents±sd (EBCO avec g fixé).

Données		$\text{EBCO}_{80\%}$	$g^\star$ estimé	g attendu
CLASSIC3	NMI	0.95 ± 0.00	3.0	3
	ARI	0.97 ± 0.00		
CSTR	NMI	0.79 ± 0.00	4.0	4
	ARI	0.83 ± 0.01		
CLASSIC4	NMI	0.76 ± 0.02	4.1	4
	ARI	0.76 ± 0.06		
LA12	NMI	0.61 ± 0.02	5.7	6
	ARI	0.59 ± 0.03		
SPORTS	NMI	0.59 ± 0.02	6.0	7
	ARI	0.51 ± 0.06		
TR45	NMI	0.76 ± 0.02	8.5	10
	ARI	0.70 ± 0.04		
PUBMED10	NMI	0.64 ± 0.02	7.1	10
	ARI	0.58 ± 0.03		

TAB. 1: NMI et ARI moyens pour le clustering de documents±sd (EBCO *non supervisé*).

PUBMED10 comporte environ 15000 résumés biomédicaux, issus de la base de données Medline, qui concernent 10 maladies. EBCO infère pour ces données $g^\star = 7$ co-clusters au lieu de 10. Pour 4 de ces co-clusters, les mots les plus représentatifs [6] indiquent qu'une seule maladie concerne le co-cluster (**AMD**, **Otitis**, **Migraine** et **Hay Fever** ; Table 2). Les mots représentatifs des 3 autres co-clusters de EBCO reflètent des relations biomédicales réelles entre plusieurs maladies qui expliquent ces regroupements (Table 3). On constate l'association de **Kidney Calculi**, **Gout** et **Jaundice**. Les calculs rénaux sont fréquents chez les patients ayant des désordres métaboliques comme la goutte. Plusieurs études ont aussi montré des associa-

6. On calcule pour chaque terme un score de cohérence avec les autres mots. Ce score, NPMI_i (voir Affeldt et al., 2020, Section 6.7.1), combine la NPMI (Normalized Pointwise Mutual Information) et l'idée du k-nearest neighbors.

tions cliniques entre l'insuffisance rénale et la jaunisse obstructive. Nous voyons également une association entre **Migraine**, **Raynaud's disease** et **AMD** ($g' \in [10..7]$, Fig. 4). Le corps des patients ayant la maladie de Raynaud hyper-réagit par une contrition des vaisseaux sanguins, fréquemment en lien avec la migraine. Ces contritions induisent un manque d'oxygénation qui est un facteur de risque pour l'AMD. Enfin, des études ont montré que l'hépatite (**Hepatitis A**) est une complication grave de la varicelle (**Chickenpox**) chez l'adulte.

Disease	#doc.
Gout	543
Chickenpox	732
Raynaud's disease	343
Jaundice	503
Hepatitis A	796
Hay Fever	1517
Kidney Calculi	1549
AMD	3283
Migraine	3703
Otitis	2596

AMD	$\lvert\text{NPMI}_i\rvert$	**Otitis**	$\lvert\text{NPMI}_i\rvert$	**Migraine**	$\lvert\text{NPMI}_i\rvert$	**Hay Fever**	$\lvert\text{NPMI}_i\rvert$
macular	0.61	**otitis**	0.48	placebo	0.34	**allergic**	0.55
degeneration	0.47	pneumonia	0.44	efficacy	0.33	**rhinitis**	0.55
retinal	0.46	antibiotics	0.39	adverse	0.33	allergy	0.54
edema	0.37	**bacterial**	0.38	treatment	0.33	asthma	0.51
acuity	0.37	acute	0.38	dose	0.32	allergen	0.51
diabetic	0.37	chronic	0.37	drug	0.31	**immunotherapy**	0.40
optic	0.33	influenza	0.32	**headache**	0.30	nasal	0.38
visual	0.30	recurrent	0.32	effect	0.30	pollen	0.38
vision	0.27	**effusion**	0.32	pain	0.29	symptom	0.34
eye	0.26	complication	0.31	**triptan**	0.25	skin	0.28
injection	0.26	ear	0.29	treat	0.24	exposure	0.25
laser	0.23	pathogenic	0.28	severe	0.23	cell	0.25
amd	0.22	resistant	0.25	medical	0.23	eosinophil	0.25
therapy	0.21	isolate	0.20	prevention	0.23	airway	0.23
risk	0.21	membrane	0.20	trial	0.22	seasonal	0.23

TAB. 2: PUBMED10 (à gauche) et mots représentatifs de 4 co-clusters de EBCO (à droite).

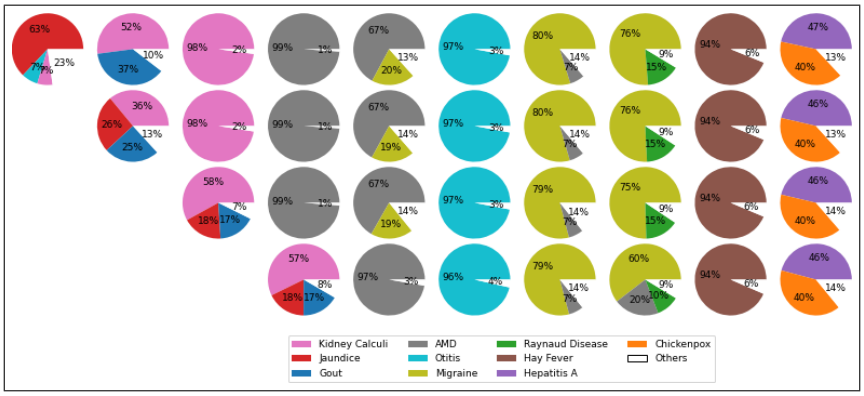


FIG. 3: Regroupements de thématiques avec EBCO pour $g' \in [10..7]$ (haut vers bas).

Kidney Calculi, Jaundice, Gout	$\lvert\text{NPMI}_i\rvert$	**Migraine, AMD Raynaud Disease**	$\lvert\text{NPMI}_i\rvert$	**Hepatitis A, Chickenpox**	$\lvert\text{NPMI}_i\rvert$
uric	0.52	mutation	0.42	**varicella**	0.57
kidney	0.46	gene	0.39	**zoster**	0.56
urinary	0.46	allele	0.39	virus	0.48
urine	0.45	genetic	0.39	vzv	0.48
oxalate	0.44	**polymorphism**	0.37	**hepatitis**	0.46
renal	0.44	disease	0.27	infection	0.43
calcium	0.39	**migraine**	0.24	viral	0.42
serum	0.38	affect	0.23	antibodies	0.36
acid	0.37	factor	0.23	immune	0.35
gout	0.36	identify	0.19	prevalence	0.32
obstruction	0.32	associate	0.19	incidence	0.28
jaundice	0.30	analysis	0.18	estimate	0.27
lithotripsy	0.30	evidence	0.18	detect	0.22
patient	0.29	suggest	0.18	outbreak	0.21
calculi	0.28	**blood**	0.15	sample	0.17

TAB. 3: Regroupement de thématiques avec EBCO pour $g^\star = 7$ (EBCO *non supervisé*).

5 Conclusion

Nous proposons EBCO, une méthode ensemble pour le co-clustering particulièrement adaptée aux données textuelles. L'approche est efficace et permet d'obtenir des co-clusters de type document-mot facilement interprétables. Le problème du choix du nombre de co-clusters est crucial. Sur la base d'un critère de modularité non trivial généralisé, EBCO permet l'identification automatique du nombre de co-clusters consensus. EBCO se rapproche, dans une version ensemble, d'un double sphérical k-means pondéré. Cela rend possible le traitement de données clairsemées de grandes dimensions, avec une faible complexité en temps (Affeldt et al., 2020).

Références

Affeldt, S., L. Labiod, et M. Nadif (2020). Ensemble block co-clustering : A unified framework for text data. CIKM '20, pp. 5–14. ACM.

Ailem, M., F. Role, et M. Nadif (2016). Graph modularity maximization as an effective method for co-clustering text data. *Knowledge-Based Systems 109*, 160–173.

Govaert, G. et M. Nadif (2013). *Co-clustering : models, algorithms and applications*. John Wiley & Sons.

Govaert, G. et M. Nadif (2018). Mutual information, phi-squared and model-based coclustering for contingency tables. *Advances in Data Analysis and Classification 12*(3), 455–488.

Hanczar, B. et M. Nadif (2012). Ensemble methods for biclustering tasks. *Pattern Recognition 45*(11), 3938–3949.

Huang, S., H. Wang, D. Li, Y. Yang, et T. Li (2015). Spectral co-clustering ensemble. *Knowledge-Based Systems 84*, 46–55.

Labiod, L. et M. Nadif (2011). Co-clustering for binary and categorical data with maximum modularity. In *2011 IEEE 11th International Conference on Data Mining*, pp. 1140–1145.

Salah, A. et M. Nadif (2017). Model-based von mises-fisher co-clustering with a conscience. In *Proceedings of the 2017 SIAM International Conference on Data Mining*, pp. 246–254.

Strehl, A. et J. Ghosh (2003). Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *The Journal of Machine Learning Research 3*, 583–617.

Yu, X., G. Yu, J. Wang, et C. Domeniconi (2019). Co-clustering ensembles based on multiple relevance measures. *IEEE Transactions on Knowledge and Data Engineering*, 1–1.

Summary

We propose a unified framework for Ensemble Block Co-clustering (EBCO) which aims to fuse multiple basic co-clusterings into a consensus structured affinity matrix. Each basic co-clustering is obtained with a co-clustering method on the same document-term dataset. This fusion process reinforces the individual quality of the multiple basic data co-clusterings within a single consensus matrix. The proposed framework enables an unsupervised co-clustering where the number of co-clusters is inferred based on the non trivial generalized modularity. We define an explicit objective function which allows the joint learning of the basic co-clusterings aggregation and the consensus block co-clustering (Affeldt et al., 2020).