Biclustering Basé sur le Transport Optimal

Chakib Fettal[*,**], Lazhar Labiod[*], Mohamed Nadif[*]

[*] Centre Borelli UMR 9010, Université Paris Cité, 75006 Paris
{prenom.nom}@u-paris.fr
[**] Informatique Caisse des Dépôts et Consignations

Résumé. Les graphes bipartis peuvent être utilisés pour modéliser une grande variété d'informations dyadiques telles que les paires utilisateur-score, document-terme et gène-conditions expérimentales. Le biclustering est une extension du clustering au graphe biparti sous-jacent induit par ce type de données. Dans cet article, nous tirons parti du transport optimal (OT), qui s'est popularisé dans la communauté de l'apprentissage automatique, pour proposer un nouveau modèle de biclustering efficace qui généralise plusieurs approches classiques de biclustering. Nous réalisons des expériences approfondies pour montrer l'intérêt de notre approche par rapport à d'autres algorithmes de biclustering de type OT.

1 Introduction

Le présent article est un résumé de l'article publié dans la conférence NeurIPS (Fettal et al., 2022a). Soit $G = (U, V, E)$ un *graphe biparti*, c'est-à-dire un graphe dont les sommets peuvent être divisés en deux ensembles disjoints $U = \{1, 2, \ldots, |U|\}$ avec $|U| = n$, $V = \{1, 2, \ldots, |V|\}$ avec $|V| = d$ et l'ensemble des arêtes E où chaque arête relie un sommet de U à un sommet de V. La matrice d'adjacence pour ce type de graphe a la structure suivante

$$\mathbf{A} = \begin{pmatrix} \mathbf{0}_{n\times n} & \mathbf{B} \\ \mathbf{B}^\top & \mathbf{0}_{d\times d} \end{pmatrix} \tag{1}$$

où $\mathbf{B}$ de taille $n \times d$ est appelée la *matrice de biadjacence* de G, ses lignes et ses colonnes correspondant aux deux ensembles de sommets ; chaque entrée représente une arête entre une ligne et une colonne. Le *Biclustering* (ou *Co-clustering*) est l'extension du clustering à ce type de graphe. À la suite des travaux de Hartigan (1972), plusieurs modèles de biclustering ont tenté de résoudre le problème en considérant $\mathbf{B}$ comme une matrice à deux modes et en recherchant une partition simultanée de ses lignes et colonnes (Dhillon, 2001; Govaert et Nadif, 2003, 2013). De cette façon, le biclustering cherche à révéler les sous-ensembles de U qui présentent un comportement similaire à travers un sous-ensemble de V dans la matrice $\mathbf{B}$.

Le biclustering ou co-clustering a été utilisé dans différents contextes. Eisen et al. (1998) ont utilisé des données de biopuces pour identifier des relations entre les gènes et les conditions, en constatant que les gènes ayant des fonctions similaires se regroupent souvent. Harpaz et al. (2011) ont appliqué ce paradigme aux données du système de notification de "Administration des aliments et des médicaments" américaine afin d'identifier les groupes de médicaments ayant

des effets indésirables. Dolnicar et al. (2012) l'ont utilisé pour trouver des segments de marché parmi les touristes afin de permettre un marketing ciblé plus efficace. Il y a eu diverses autres applications et approches, voir par exemple (Gu et Liu, 2008; Salah et Nadif, 2019).

Récemment, le *Transport Optimal* (OT) a suscité beaucoup d'intérêt au sein de la communauté *apprentissage machine*. L'OT a aidé à résoudre une variété de problèmes d'exploration de données, et le biclustering ne fait pas exception. Laclau et al. (2017) ont proposé deux modèles de biclustering : un premier modèle, CCOT, qui effectue le co-clustering sur la base des vecteurs d'échelle obtenus en appliquant l'algorithme de Sinkhorn-Knopp sur une version sous-échantillonnée carrée de la matrice $\mathbf{B}$, et un second modèle, CCOT-GW, qui utilise les vecteurs d'échelle obtenus en calculant les barycentres entropiques de Gromov-Wasserstein, et qui ne nécessite pas de sous-échantillonnage. Puis vint (Titouan et al., 2020), où les auteurs ont fait du biclustering en minimisant une nouvelle métrique, COOT, qui généralise la distance de Gromov-Wasserstein entre $\mathbf{B}$ et une matrice de résumé, de façon similaire à ce qui a été fait dans (Dhillon et al., 2003). Plus précisément, ils ont proposé deux nouvelles métriques : COOT, ainsi qu'une métrique régularisée entropiquement COOT$_\lambda$. Cependant, dans ces travaux les propositions dans (Laclau et al., 2017) et (Titouan et al., 2020) présentent tous deux certains inconvénients. Tout d'abord, les deux algorithmes ne s'attaquent pas au biclustering dès le début ; les co-clusters sont déduits à la convergence. Ainsi, le biclustering est une conséquence et non un objectif principal. Deuxièmement, ils souffrent d'une complexité de calcul élevée ; CCOT et CCOT-GW consomment également de grandes quantités de mémoire. Enfin, nous verrons que ces algorithmes ne sont pas adaptés aux données sparses dyadiques.

Dans cet article, tout en intégrant l'objectif de biclustering dès le début, nous proposons un cadre générique pour le biclustering par transport optimal, qui généralise d'ailleurs certaines approches de biclustering existantes. Ainsi, nous proposons deux méthodes efficaces pour résoudre ce problème : une qui donne un biclustering presque dur, et une seconde qui donne un biclustering *flou* par régularisation entropique. Ces méthodes s'avèrent plus performantes que d'autres modèles de biclustering de transport optimal, en termes de clustering de documents et de termes, sur plusieurs ensembles de données réguliers et à grande échelle, tout en étant plus efficaces en termes de calcul et de mémoire. Nous soulignons une fois de plus que l'approche que nous proposons est particulièrement adaptée aux ensembles de données dyadiques.

2 Méthodologie

Notations. Dans ce qui suit, $\Delta^n = \{\mathbf{p} \in \mathbb{R}_+^n | \sum_{i=1}^n p_i = 1\}$ désigne le simplexe standard à n dimensions. $\Pi(\mathbf{w}, \mathbf{v}) = \{\mathbf{Z} \in \mathbb{R}_+^{n \times k} | \mathbf{Z}\mathbf{1}_k = \mathbf{w}, \mathbf{Z}^\top \mathbf{1}_n = \mathbf{v}\}$ désigne le polytope de transport, où $\mathbf{w} \in \Delta^n$ et $\mathbf{v} \in \Delta^k$ sont les marginales de la distribution conjointe $\mathbf{Z}$ et $\mathbf{1}_n$ est un vecteur de de taille n remplie de 1. Les matrices sont désignées par des lettres majuscules en caractères gras et les vecteurs par des lettres minuscules en caractères gras. Pour une matrice $\mathbf{M}$, sa i-ième ligne est $\mathbf{m}_i$ et sa j-ième colonne est $\mathbf{m}'_j$.

2.1 Préliminaires

Nous allons d'abord introduire l'OT discret et sa version régularisée, et montrer comment le biclustering peut être posé comme un programme en nombres entiers.

OT discret comme programme linéaire. Le but du transport optimal discret est de trouver une carte de transport de coût minimal entre une distribution de probabilités source $\mathbf{w}$ et une distribution de probabilités cible $\mathbf{v}$. Nous nous intéressons ici au cas discret de la formulation de Kantorovich de l'OT, à savoir

$$\mathrm{OT}(\mathbf{M}, \mathbf{w}, \mathbf{v}) \triangleq \min_{\mathbf{Z} \in \Pi(\mathbf{w}, \mathbf{v})} \langle \mathbf{M}, \mathbf{Z} \rangle \tag{2}$$

où $\mathbf{M} \in \mathbb{R}^{n \times k}$ est la matrice de coût, et m_{ij} quantifie l'effort nécessaire pour transporter une masse de probabilité de $\mathbf{w}_i$ à $\mathbf{v}_j$.

OT discret avec régularisation entropique. Il a été suggéré dans la littérature (Cuturi, 2013; Chizat et al., 2020) que l'utilisation d'une régularisation telle que la régularisation entropique peut conduire à une meilleure efficacité computationnelle et statistique.

$$\mathrm{OT}_\lambda(\mathbf{M}, \mathbf{w}, \mathbf{v}) \triangleq \min_{\mathbf{Z} \in \Pi(\mathbf{w}, \mathbf{v})} \langle \mathbf{M}, \mathbf{Z} \rangle - \lambda H(\mathbf{Z}) \tag{3}$$

où H est l'entropie définie par $H(\mathbf{Z}) \triangleq -\sum_{i,j} z_{ij} \log z_{ij}$ et où λ contrôle la régularisation.

Biclustering comme programme en nombres entiers. Le problème *Sériation en blocs* (Marcotorchino, 1987) consiste à trouver deux matrices de permutation, une pour les lignes et une pour les colonnes, de sorte que les blocs denses apparaissent le long de la diagonale de la matrice après des permutations appropriées ; il s'agit d'un problème de biclustering comme le montre les auteurs dans (Laclau et Nadif, 2016). Plus précisément, une définition possible du problème de la sériation en blocs serait la suivante : étant donnée une matrice $\mathbf{B} \in \mathbb{R}^{n \times d}$ où b_{ij} décrit en quelque sorte la force de l'association entre la ligne i et la colonne j , le but est d'apprendre $\mathbf{C}$, une matrice diagonale en blocs jusqu'à une permutation de ses lignes et colonnes près qui represente la bipartition. Une formulation possible utilise une contrainte de rang sur $\mathbf{C}$. Nous pouvons ainsi définir un nouveau problème par factorisation de rang inférieur de $\mathbf{C}$, soit $\mathbf{C} = \mathbf{Z}\mathbf{W}^\top$, que nous formulons par

$$\max_{\mathbf{Z} \in \Gamma(n,k), \mathbf{W} \in \Gamma(d,k)} \sum_{i,j,h} b_{ij} z_{ih} w_{jh} \tag{4}$$

où $\Gamma(n, k) = \{\mathbf{Z} \in \{0, 1\}^{n \times k} \, | \mathbf{Z}\mathbf{1} = \mathbf{1}\}$ est l'ensemble des partitions dures de taille $n \times k$.

2.2 Biclustering via Transport Optimal

Nous proposons ici un nouveau problème de biclustering basé sur la sériation en blocs et le transport optimal. À cette fin, nous définissons d'abord ce que nous appelons une *matrice d'anti-adjacence*. Notez qu'un concept similaire a été discuté dans (Wang et al., 2018).

Definition 1 (Matrice d'anti-adjacence) *Étant donné un graphe caractérisé par une matrice d'adjacence $\mathbf{A}$, nous avons une matrice d'anti-adjacence correspondante $\overline{\mathbf{A}}$ s.t. $\overline{a}_{ij}$ quantifie la divergence entre le nœud i et j.*

Nous considérons un graphe biparti caractérisé par sa matrice de biadjacence $\mathbf{B} = (b_{ij}) \in \mathbb{R}^{n \times d}$. Les lignes de $\mathbf{B}$ sont dotées de poids $\mathbf{w} \in \Delta^n$ et ses colonnes de poids $\mathbf{v} \in \Delta^d$. Nous considérons également une distribution de lignes $\mathbf{r} \in \Delta^r$ et une distribution de colonnes $\mathbf{c} \in \Delta^c$. Selon la disponibilité d'informations *a priori* sur les données, ces vecteurs de poids peuvent être fixés à des distributions uniformes.

Maintenant, soit $\overline{\mathbf{B}} = L(\mathbf{B})$, où $L : \mathbb{R}^{n \times d} \to \mathbb{R}^{n \times d}$. Cela signifie que b_{ij}, l'association entre le nœud i et le nœud j, est transformée en une mesure de divergence $L(\mathbf{B})_{ij}$. De cette façon, nous définissons le problème de la sériation en blocs via transport optimal comme le programme bilinéaire suivant :

$$\text{BCOT}(\mathbf{w}, \mathbf{v}, \mathbf{r}, \mathbf{c}) \triangleq \min_{\mathbf{Z} \in \Pi(\mathbf{w}, \mathbf{r}), \mathbf{W} \in \Pi(\mathbf{v}, \mathbf{c})} \left\langle L(\mathbf{B}), \mathbf{Z}\mathbf{W}^\top \right\rangle \tag{5}$$

où $\mathbf{Z}$ est une carte de transport entre la distribution des lignes $\mathbf{w}$ et la distribution *représentante* des lignes $\mathbf{r}$, et de même pour $\mathbf{W}$ par rapport à la distribution des colonnes $\mathbf{v}$ et la distribution *représentante* des colonnes $\mathbf{c}$.

Biclustering via BCOT. Nous allons maintenant montrer comment obtenir une partition des lignes et des colonnes à partir d'une paire de solutions $(\mathbf{Z}, \mathbf{W})$. Dans ce qui suit, notre objectif est d'identifier un couple *clustering h-almost hard* pour les lignes et les colonnes à partir des solutions $\mathbf{Z}$ et $\mathbf{W}$.

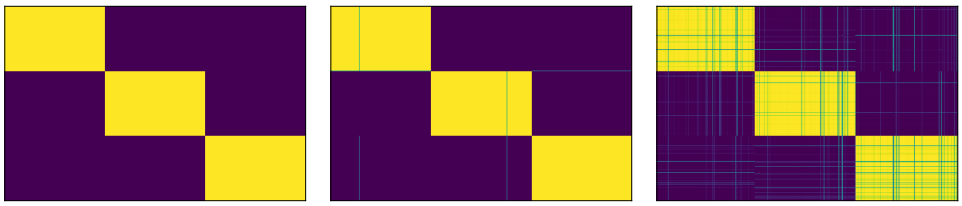
Definition 2 (clustering h-almost hard) *Nous définissons un clustering h-almost hard comme un clustering dont la matrice d'affectation est $\mathbf{C} \in \mathbb{R}^{n \times k}$ avec $\|\mathbf{C}\|_0 = n + h$ et pour chaque ligne $\boldsymbol{c}$ de $\mathbf{C}$ nous avons que $\|\boldsymbol{c}\|_0 > 0$; $\|.\|_0$ retourne le nombre d'éléments non nuls. Lorsque $h = 0$, on obtient un clustering dur standard avec un élément non nul par ligne.*

Proposition 1 *Pour $\mathbf{w}$, $\mathbf{v}$, $\mathbf{r}$ et $\mathbf{c}$ ne contenant pas de zéros, il existe une paire optimale de matrices de transport $\mathbf{Z}$ et $\mathbf{W}$ qui sont des clusterings h-almost hard avec $h \in \{0, \ldots, k-1\}$. De plus, lorsque $n = k$ (resp. $d = k$) et $\mathbf{w} = \mathbf{r}$ (resp. $\mathbf{v} = \mathbf{c}$), ce $\mathbf{Z}$ (resp. $\mathbf{W}$) devient un clustering dur, c'est-à-dire, $\mathbf{Z} \in \Gamma(n, n)$ (resp. $\mathbf{W} \in \Gamma(d, d)$).*

Cela signifie que les solutions sont déjà presque une partition dure des données, puisque $k << n, d$. Pour obtenir un clustering dur final au sens strict, nous assignons chaque ligne (resp. colonne) à celle correspondant à la ligne de $\mathbf{Z}$ (resp. $\mathbf{W}$) avec la plus grande valeur. Cela ne devrait pas modifier de manière significative la structure de la solution. La figure 1b en fournit une illustration : nous voyons ici la structure de la diagonale de bloc générée par le produit des deux matrices de transport $\mathbf{C} = \mathbf{Z}\mathbf{W}^\top$, dont l'apparence est similaire à celle du biclustering produit par la sériation en blocs dure (Figure 1a), à l'exception de quelques entrées non nulles hors de la diagonale de blocs qui sont difficiles à identifier immédiatement.

Intuition de BCOT. Pour expliquer l'intuition derrière l'approche proposée, nous devons examiner la manière dont le problème est résolu. La procédure d'optimisation telle que décrite dans l'algorithme 1 consiste à alterner entre le calcul d'une carte de transport optimal $\mathbf{Z}$ étant donné $\mathbf{W}$ et vice versa. En ce qui concerne la résolution de $\mathbf{Z}$ étant donné $\mathbf{W}$, le problème peut être réécrit comme suit

$$\text{BCOT}(\mathbf{w}, \mathbf{v}, \mathbf{r}, \mathbf{c}) \equiv \min_{\mathbf{Z} \in \Pi(\mathbf{w}, \mathbf{r})} \langle L(\mathbf{B})\mathbf{W}, \mathbf{Z} \rangle \,. \tag{6}$$



(a) Block Seriation. (b) BCOT. (c) BCOT$_\lambda$.

FIG. 1 – Biclusters formés à l'aide de trois méthodes différentes sur l'ensemble de données Pubmed. La sériation en blocs classique donne un biclustering dur. La méthode BCOT donne un biclustering *presque dur* avec peu d'entrées non nulles en dehors de la diagonale du bloc principal. BCOT$_\lambda$ aboutit à un biclustering *flou* avec de nombreux éléments non nuls en dehors de la diagonale en blocs.

Il s'agit d'un problème de transport optimal avec $L(\mathbf{B})\mathbf{W}$ comme matrice de coût. La carte de transport résultante $\mathbf{Z}$ peut être vue comme une sorte de matrice d'affectation des lignes vers des clusters : si $z_{ih} > 0$, alors la rangée i est affectée au cluster h. Il en va de même pour $\mathbf{W}$, qui peut être considérée comme une matrice d'affectation des colonnes. Cela signifie également que, puisque $L(\mathbf{B})$ est la dissimilarité entre les lignes et les colonnes, alors la matrice des coûts $L(\mathbf{B})\mathbf{W}$ représente la dissimilarité entre les lignes et les representants des lignes (par exemple, centroides). Ainsi, $L(\mathbf{B})_i\mathbf{w}_h$ est la dissimilarité ou le coût du transport de masse de probabilité entre la ligne i et le représentant de la classe des lignes h. Le raisonnement est le même pour les colonnes et le couplage optimal $\mathbf{W}$.

2.3 Biclustering Flou via le Transport Optimal Régularisé

Comme mentionné précédemment, l'utilisation de la régularisation entropique peut être intéressante en raison de ses diverses caractéristiques utiles, notamment l'efficacité statistique et informatique. Cependant, une autre caractéristique de la régularisation entropique est que les couplages optimaux $\mathbf{Z}$ et $\mathbf{W}$ sont des matrices denses, conséquence de la structure de la solution optimale des problèmes d'OT régularisés entropiquement. Nous formulons le problème comme suit

$$\text{BCOT}_\lambda(\mathbf{w}, \mathbf{v}, \mathbf{r}, \mathbf{c}) \triangleq \min_{\mathbf{Z}\in\Pi(\mathbf{w},\mathbf{r}),\mathbf{W}\in\Pi(\mathbf{v},\mathbf{c})} \left\langle L(\mathbf{B}), \mathbf{Z}\mathbf{W}^\top \right\rangle - \lambda_\mathbf{Z} H(\mathbf{Z}) - \lambda_\mathbf{W} H(\mathbf{W}) \quad (7)$$

où $\lambda_\mathbf{Z}$ et $\lambda_\mathbf{W}$ sont les paramètres de régularisation. La figure 1c montre les biclusters produits par les solutions de BCOT$_\lambda$. Comme pour BCOT, une structure diagonale en blocs est formée. Cependant, il existe également plusieurs entrées non nulles hors de la diagonale des blocs qui représentent les probabilités d'appartenance des paires ligne-colonne aux mêmes biclusters.

3 Optimisation et Complexité

Le problème de la sériation en blocs étant NP-difficile, le calcul d'une solution exacte est prohibitif. Une heuristique efficace et largement utilisée pour résoudre ce type de problèmes

implique l'utilisation de la descente de coordonnées par bloc ; alternativement les affectations des lignes sont calculées sachant des affectations de colonnes fixées et vice versa. Nous exprimons l'algorithme proposé en pseudo-code dans Algorithme 1.

Algorithme 1 : BCOT

Input : $\mathbf{B}$ matrice de biadjacence,
$\mathbf{w}$ et $\mathbf{v}$ les poids de lignes et colonnes,
$\mathbf{r}$ and $\mathbf{c}$ les poids des clusters ligne et colonne.
Output : π^r, π^c row and column partitions
$\mathbf{W} \leftarrow \mathbf{W}_{init}$;
while *not converged* **do**
  $\mathbf{Z} \leftarrow \arg \mathtt{OT}\left(L(\mathbf{B})\mathbf{W}, \mathbf{w}, \mathbf{r}\right)$;
  $\mathbf{W} \leftarrow \arg \mathtt{OT}\left(L(\mathbf{B})^\top \mathbf{Z}, \mathbf{v}, \mathbf{c}\right)$;
end
Generate π^r, π^c from $\mathbf{Z}$ and $\mathbf{W}$;

4 Expériences

Dans cette section nous présentons les données sur lesquelles seront évaluées les méthodes proposées et d'autres méthodes s'appuyant sur OT. D'autres expériences et compraisons sont également disponibles dans le papier original (Fettal et al., 2022a). Nous évaluons BCOT par rapport à six matrices termes-documents ; voir table 1 pour leurs descriptions. Les résultats du

TAB. 1 – Caractéristiques des ensembles de données.

Dataset	**#Documents**	**#Termes**	**#Clusters**	**Sparsité** (%)
ACM (Fan et al., 2020)	3025	1870	3	95.52
DBLP (Fan et al., 2020)	4057	334	4	96.4
PubMed (Sen et al., 2008)	19717	500	3	89.98
Wiki (Yang et al., 2015)	2405	4973	17	86,99
Ohscal (Hersh et al., 1994)	11162	11465	10	99.47
20 Newsgroups (Lang, 1995)	18846	14390	20	99.41

TAB. 2 – Performance du clustering de documents sur les jeux de données.

Method	**ACM**			**DBLP**			**PubMed**			**Wiki**		
	CA	NMI	ARI	CA	NMI	ARI	CA	NMI	ARI	CA	NMI	ARI
k-Means	51.1±11.3	13.7±11.2	14.0±10.6	36.9±2.4	10.4±2.0	4.3±2.0	52.3±4.7	18.2±10.5	15.3±10.1	26.0±6.1	18.6±9.3	3.3±2.9
CCOT	12.4±2.0	1.0±0.2	0.4±0.2	28.6±0.5	0.6±0.0	0.4±0.0	32.7±0.2	3.0±0.0	3.1±0.1	10.6±0.5	4.9±0.1	0.6±0.15
CCOT-GW	8.1±0.0	1.5±0.0	0.3±0.0	9.4±0.0	1.7±0.0	0.3±0.0		OOM		10.9±0.0	4.3±0.0	0.48±0.0
COOT*	39.0±0.0	1.9±0.0	2.0±0.0	30.5±1.4	1.4±0.3	1.2±0.3	43.2±1.5	1.7±0.6	1.3±1.5	25.9±1.8	28.7±2.2	12.3±1.7
COOT$_\lambda$	41.5±0.2	1.9±0.1	2.2±0.0	30.6±0.0	0.7±0.0	0.6±0.0	42.4±1.5	1.7±0.5	1.0±1.3	17.2±0.0	1.7±0.0	0.31±0.0
BCOT	**76.6±1.5**	**38.3±2.2**	**43.3±2.6**	**61.5±6.2**	**27.4±4.3**	**28.3±5.5**	53.6±4.5	15.9±1.9	12.9±2.4	49.8±1.5	47.9±1.0	30.6±1.0
BCOT$_\lambda$	76.2±0.6	37.6±0.8	42.4±1.0	59.4±9.9	26.6±7.6	27.2±9.5	**56.5±3.1**	**18.4±1.3**	**15.4±1.8**	**50.8±1.5**	**49.4±0.9**	**31.9±0.8**

clustering de documents sur ACM, DBLP, PubMed et Wiki sont donnés dans le tableau 2 pour les trois métriques. Dans tous les cas, le meilleur résultat est obtenu soit par BCOT, soit par BCOT$_\lambda$. De plus, sur Wiki, BCOT$_\lambda$ donne des résultats compétitifs par rapport aux méthodes

de pointe de clustering de graphes attribués présentées dans (Fettal et al., 2022b), bien qu'il n'ait pas accès aux informations sur la structure des graphes dans le réseau de citations Wiki.

5 Conclusion

Le clustering et le biclustering par transport optimal n'en sont encore qu'à leurs débuts, et de nombreux défis restent à relever. Cet article présente un nouveau problème de biclustering par transport optimal qui tient compte de la nature sparse de certains types de données dyadiques. Le problème est posé comme un programme bilinéaire que nous résolvons en utilisant un algorithme efficace de descente de coordonnées par bloc. Les expériences menées sur un certain nombre d'ensembles de données sur des documents suggèrent que l'approche proposée permet de trouver des groupes qui correspondent aux classes réelles. Dans ce contexte, notre modèle surpasse les méthodes récentes de biclustering OT par une marge significative. D'autres details sur des connexions avec d'autres approches, des preuves, ainsi que des expériences et évaluations de notre approche sont disponibles dans (Fettal et al., 2022a).

Remerciements. Ce travail a été financé par la Caisse des Dépôts et Consignations (CDC), l'ANRT et l'Idex-Spectrans d'Université Paris Cité.

Références

Chizat, L., P. Roussillon, F. Léger, F.-X. Vialard, et G. Peyré (2020). Faster wasserstein distance estimation with the sinkhorn divergence. *Advances in Neural Information Processing Systems 33*, 2257–2269.

Cuturi, M. (2013). Sinkhorn distances : Lightspeed computation of optimal transport. *Advances in neural information processing systems 26*.

Dhillon, I. S. (2001). Co-clustering documents and words using bipartite spectral graph partitioning. In *SIGKDD*, pp. 269–274.

Dhillon, I. S., S. Mallela, et D. S. Modha (2003). Information-theoretic co-clustering. In *SIGKDD*, pp. 89–98.

Dolnicar, S., S. Kaiser, K. Lazarevski, et F. Leisch (2012). Biclustering : Overcoming data dimensionality problems in market segmentation. *Journal of Travel Research 51*(1), 41–49.

Eisen, M. B., P. T. Spellman, P. O. Brown, et D. Botstein (1998). Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences 95*(25), 14863–14868.

Fan, S., X. Wang, C. Shi, E. Lu, K. Lin, et B. Wang (2020). One2multi graph autoencoder for multi-view graph clustering. In *Proceedings of The Web Conference 2020*, pp. 3070–3076.

Fettal, C., L. Labiod, et M. Nadif (2022a). Efficient and effective optimal transport-based biclustering. *Advances in Neural Information Processing Systems 35*.

Fettal, C., L. Labiod, et M. Nadif (2022b). Efficient graph convolution for joint node representation learning and clustering. In *WSDM*, pp. 289–297.

Govaert, G. et M. Nadif (2003). Clustering with block mixture models. *Pattern Recognition 36*(2), 463–473.

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

Gu, J. et J. S. Liu (2008). Bayesian biclustering of gene expression data. *BMC genomics 9*(1), 1–10.

Harpaz, R., H. Perez, H. S. Chase, R. Rabadan, G. Hripcsak, et C. Friedman (2011). Biclustering of adverse drug events in the fda's spontaneous reporting system. *Clinical Pharmacology & Therapeutics 89*(2), 243–250.

Hartigan, J. A. (1972). Direct clustering of a data matrix. *Journal of the american statistical association 67*(337), 123–129.

Hersh, W., C. Buckley, T. Leone, et D. Hickam (1994). Ohsumed : An interactive retrieval evaluation and new large test collection for research. In *SIGIR'94*, pp. 192–201. Springer.

Laclau, C. et M. Nadif (2016). Hard and fuzzy diagonal co-clustering for document-term partitioning. *Neurocomputing 193*, 133–147.

Laclau, C., I. Redko, B. Matei, Y. Bennani, et V. Brault (2017). Co-clustering through optimal transport. In *International Conference on Machine Learning*, pp. 1955–1964. PMLR.

Lang, K. (1995). Newsweeder : Learning to filter netnews. In *Proceedings of the Twelfth International Conference on Machine Learning*, pp. 331–339.

Marcotorchino, J. F. (1987). Block seriation problems : A unified approach. *Applied Stochastic Models and Data Analysis 3*(2), 73–91.

Salah, A. et M. Nadif (2019). Directional co-clustering. *Advances in Data Analysis and Classification 13*(3), 591–620.

Sen, P., G. Namata, M. Bilgic, L. Getoor, B. Galligher, et T. Eliassi-Rad (2008). Collective classification in network data. *AI magazine 29*(3), 93–93.

Titouan, V., I. Redko, R. Flamary, et N. Courty (2020). Co-optimal transport. *Advances in Neural Information Processing Systems 33*, 17559–17570.

Wang, J., M. Lu, F. Belardo, et M. Randić (2018). The anti-adjacency matrix of a graph : Eccentricity matrix. *Discrete Applied Mathematics 251*, 299–309.

Yang, C., Z. Liu, D. Zhao, M. Sun, et E. Y. Chang (2015). Network representation learning with rich text information. In *IJCAI*.

Summary

Bipartite graphs can be used to model a wide variety of dyadic information such as user-rating, document-term, and gene-disorder pairs. Biclustering is an extension of clustering to the underlying bipartite graph induced from this kind of data. In this paper, we leverage optimal transport (OT) which has gained momentum in the machine learning community to propose a novel and scalable biclustering model that generalizes several classical biclustering approaches. We perform extensive experimentation to show the validity of our approach compared to other OT biclustering algorithms along both dimensions of the dyadic datasets.