Efficient and Effective Optimal Transport-Based Biclustering

Anonymous Author(s)
Affiliation
Address
email

Abstract

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.

1 Introduction

Let $G = (U, V, E)$ be a *bipartite graph* which is a graph whose vertices can be divided into two disjoint sets $U = \{1, 2, \ldots, |U|\}$ with $|U| = n$, $V = \{1, 2, \ldots, |V|\}$ with $|V| = d$ and the set of edges E where each edge connects a vertex of U to a vertex of V. The adjacency matrix for this type of graph has the following structure

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

where $\mathbf{B}$ of size $n \times d$ is called the *biadjacency matrix* of G, its rows and columns correspond to the two sets of vertices; each entry represents an edge between a row and a column. *Biclustering* (or *Co-clustering*) is the extention of clustering to this type of graphs. Following [15], several biclustering models attempted to solve the problem by viewing $\mathbf{B}$ as a two-mode matrix and searching for a simultaneous partition of its rows and columns [6]. In this way, biclustering aims to reveal subsets of U which exhibit a similar behaviour across a subset of V in matrix $\mathbf{B}$.

Biclustering has been used in several contexts, [9] used microarray data to find relations between genes and conditions, finding that genes with similar functions often cluster together. [14] used this paradigm to identify drug groups with adverse effects on the US Food and Drug Administration reporting system. [8] used it to find market segments among tourists that are similar to each other to allow for targeted marketing, as well as many other applications [6, 30, 13].

Multiple solutions for the biclustering problem have been proposed in literature, [7] used an information-theoretic approach to solve the problem by minimizing the difference in mutual information between $\mathbf{B}$ and a summary matrix. [2] adapted classical modularity to bipartite networks and then used it to identify modules within them. [31] proposed a biclustering paradigm based on nonnegative matrix tri-factorization of the biadjacency matrix.

Recently, *Optimal Transport* (OT) took the machine learning community by storm and was used in the resolution of various data mining problems and biclustering was not an exception. First,

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

[19] proposed two models for biclustering, a first one called CCOT which does co-clustering based on the scaling vectors obtained from the application of the Sinkhorn-knopp algorithm on a square subsampled version of matrix $\mathbf{B}$ and another one called CCOT-GW that uses scaling vectors obtained from computing entropic Gromov-Wasserstein barycenters and which does not require subsampling. Then came [26] where authors did biclustering by minimizing a new metric COOT, which generalizes the Gromov-Wasserstein distance, between $\mathbf{B}$ and a summary matrix similar to what was done in [7]. In particular, they proposed two metrics COOT and an entropically regularized one COOT_λ. These approaches, however, have some drawbacks. One of them is that both of them suffer from high computational complexity and in the case of CCOT and CCOT-GW also from a large memory consumption. Another one is that their empirical performance on dyadic data is not satisfactory.

In this paper, we propose a generic framework for biclustering through optimal transport which generalizes several previous biclustering approaches. We propose two efficient methods for solving this problem, one that results in an almost hard biclustering and another that results in a *fuzzy* or *soft* biclustering through entropic regularization. These methods outperform other optimal transport biclustering models both in term of document and term clustering on several regular and large scale datasets while being more computationally and memory efficient. We emphasize once again the fact that the approach we propose is specifically tailored to datasets consisting in dyadic data which can be represented using a bipartite graph, meaning that it should not be applied on other data types such as images directly.

2 Methodology

Notations In what follows, $\Delta^n = \{\mathbf{p} \in \mathbb{R}_+^n | \sum_{i=1}^n p_i = 1\}$ denotes the n-dimensional standard simplex. $\Pi(\mathbf{w}, \mathbf{v}) = \{\mathbf{Z} \in \mathbb{R}_+^{n \times k} | \mathbf{Z}\mathbf{1} = \mathbf{w}, \mathbf{Z}^\top \mathbf{1} = \mathbf{v}\}$ denotes the transportation polytope, where $\mathbf{w} \in \Delta^n$ and $\mathbf{v} \in \Delta^k$ are the marginals of the joint distribution $\mathbf{Z}$ and $\mathbf{1}_n$ is a vector of ones. We denote matrices with uppercase boldface letters and vectors with lower case boldface letters. For a matrix $\mathbf{M}$, its i-th row is $\mathbf{m}$ and its j-th column is $\mathbf{m}'_j$ We have that $\|.\|_0$ is the 0-norm which returns the number of nonzero elements of its argument.

2.1 Preliminaries

We first need to introduce exact discrete OT and its entropically regularized counterpart and show how biclustering can be posed as an integer program.

Discrete OT as a linear program. The goal of discrete optimal transport is to find a minimal cost transport map between a source probability distribution $\mathbf{w}$ and a target distribution $\mathbf{v}$. Here we are interested in the discrete case of the Kantorovich formulation of OT, that is

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

where $\mathbf{M} \in \mathbb{R}^{n \times k}$ is the cost matrix, m_{ij} quantifies the effort needed to transport a probability mass from $\mathbf{w}_i$ to $\mathbf{v}_j$.

Discrete entropy regularized OT. Several works [4, 3] have suggested that the use of a regularization such as the entropic regularization can lead to better computational and statistical efficiency.

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

where H is the entropy defined as $H(\mathbf{Z}) \triangleq \sum_{i,j} -z_{ij} \log z_{ij}$ and λ controls the strength of regularization. The computational efficiency comes from the fact that the unique solution of this problem has the following structure $\mathbf{K} := -\texttt{diag}(\mathbf{a}) \exp(\mathbf{M}/\lambda) \texttt{diag}(\mathbf{b})$, a rescaled elementwise negative exponential of the cost $\mathbf{M}$ where $\mathbf{a}$ and $\mathbf{b}$ are scaling vectors. These vectors can be found efficiently using the Sinkhorn-Knopp algorithm.

Biclustering as an integer program. The *Block seriation* problem [21] consists in finding two permutations matrices, one for the rows and one for the columns s.t. dense blocks appear along the

diagonal of the permuted matrix. A possible definition of the block seriation problem is that given a matrix $\mathbf{B} \in \mathbb{R}^{n \times d}$ s.t b_{ij} gives the strength of the association between row i and column j e.g. a biadjacency matrix, it is formulated as follows

$$\begin{aligned} \max_{\mathbf{C}} \quad & \sum_{i,j} b_{ij} c_{ij} \\ \text{s.t} \quad & \forall\, i,j \quad c_{ij} \in \{0,1\} && \text{(binarity)} \\ & \forall\, j \quad \sum_{i} c_{ij} \geq 1; \quad \forall i \quad \sum_{j} c_{ij} \geq 1 && \text{(assignment)} \\ & \forall\, i,j \quad \sum_{j} c_{ij} + c_{i'j} + c_{ij'} - c_{i'j'} \leq 2 && \text{(impossible triads)} \end{aligned} \tag{4}$$

A solution $\mathbf{C}$ matrix is a block diagonal matrix up to a permutation of its rows and columns. The block seriation problem is an integer programming problem and is consequently NP-hard. One approach for solving this problem uses a relaxed version where a rank constraint $\texttt{rank}(\mathbf{C}) \leq k$ is added for k the number of desired biclusters. When integrating this constraint to 4, we can define a new equivalent problem by low-rank factorization of $\mathbf{C}$ i.e. $\mathbf{C} = \mathbf{Z}\mathbf{W}^\top$, which we formulate as

$$\max_{\substack{\mathbf{Z} \in \Gamma(n,k) \\ \mathbf{W} \in \Gamma(d,k)}} \sum_{i,j,h} b_{ij} z_{ih} w_{jh} \tag{5}$$

where $\Gamma(n,k) = \{\mathbf{Z} \in \{0,1\}^{n \times k} | \mathbf{Z}\mathbf{1} = \mathbf{1}\}$ is the set of hard partitions of dimension $n \times k$. A simple heuristic for solving this problem involves alternatingly solving for $\mathbf{Z}$ given $\mathbf{W}$ and vice-versa using classical clustering algorithms before identifying biclusters through the rearranged matrix $\mathbf{C}$ which displays a block diagonal structure as seen in figure 1a. The biclusters are identified by grouping together the rows and columns that form a block along diagonal.

2.2 Biclustering using Optimal Transport

Here, we propose a new biclustering problem based on block seriation and optimal transport. We first start by introducing a necessary concept which we call *anti-adjacency matrix*

Definition 1. **(Anti-adjacency matrix)** Given a graph characterized by an adjacency matrix $\mathbf{A}$, we denote its anti-adjacency matrix $\overline{\mathbf{A}}$ s.t. $\overline{a}_{ij}$ quantifies a discrepancy between node i and j.

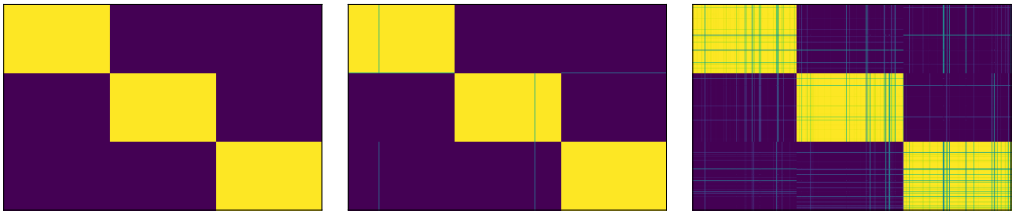
We consider a bipartite graph characterized by its biadjacency matrix $\mathbf{B} = (b_{ij}) \in \mathbb{R}^{n \times d}$. The rows of $\mathbf{B}$ are endowed with weights $\mathbf{w} \in \Delta^n$ and its columns with weights $\mathbf{v} \in \Delta^d$. We also consider a row exemplar distribution $\mathbf{r} \in \Delta^r$ and a column exemplar distribution $\mathbf{c} \in \Delta^c$. Depending upon the availability of a priori information about the data, these weight vectors can be set to uniform distributions.

Now let its anti-adjacency matrix be $\overline{\mathbf{B}} = L(\mathbf{B})$ where $L : \mathbb{R}^{n \times d} \to \mathbb{R}^{n \times d}$ leads to transform b_{ij}, the association between node i and node j, into a discrepancy measure $L(b)_{ij}$. $\mathbf{Z}$. Thus, we define the optimal transport block seriation problem as

$$\text{BCOT}(\mathbf{w},\mathbf{v},\mathbf{r},\mathbf{c}) \triangleq \min_{\substack{\mathbf{Z} \in \Pi(\mathbf{w},\mathbf{r}) \\ \mathbf{W} \in \Pi(\mathbf{v},\mathbf{c})}} \sum_{i,j,k} L(\mathbf{B})_{ij} z_{ik} w_{jk} \;\equiv\; \min_{\substack{\mathbf{Z} \in \Pi(\mathbf{w},\mathbf{r}) \\ \mathbf{W} \in \Pi(\mathbf{v},\mathbf{c})}} \left\langle L(\mathbf{B}), \mathbf{Z}\mathbf{W}^\top \right\rangle \tag{6}$$

where $\mathbf{Z}$ is a transport map (or coupling) between between the row distribution $\mathbf{w}$ and the row exemplar distribution $\mathbf{r}$ and similarly for $\mathbf{W}$ w.r.t. the column distribution $\mathbf{v}$ and the column exemplar distribution $\mathbf{c}$. Similar to [26] is an indefinite Bilinear Program and is related to the quadratic assignment problem (QAP) [18].

Inducing a Biclustering using BCOT We now are going to show how to obtain a partition of the rows and the columns given a solution pair $(\mathbf{Z}, \mathbf{W})$. In what follows, we are interested in inducing a couple of *almost-hard clustering* for rows and columns from couplings $\mathbf{Z}$ and $\mathbf{W}$.



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

Figure 1: Biclusters formed through different approaches on the Pubmed dataset. Classical block seriation results in a biclustering that is hard. BCOT results in a biclustering that is almost hard with few nonzero entries outside the main block diagonal and BCOT$_\lambda$ results in a soft biclustering with many nonzero elements outside the block diagonal.

Definition 2. **(h-almost hard clustering)** We define an h-almost hard clustering as a clustering whose assignment matrix is $\mathbf{C} \in \mathbb{R}^{n \times k}$ s.t. $\|\mathbf{C}\|_0 = n + h$ and for each row $\mathbf{c}$ of $\mathbf{C}$ we have that $\|\mathbf{c}\|_0 > 0$. When $h = 0$, we obtain a standard hard clustering with one non-zero element per row.

Proposition 1. [1] For $\mathbf{w}$, $\mathbf{v}$, $\mathbf{r}$ and $\mathbf{c}$ containing no zeros, the resulting optimal coupling matrices $\mathbf{Z}$ and $\mathbf{W}$ are always an h-almost hard clustering with $h \in \{0, \dots, k-1\}$. Furthermore, when $n = k$ (resp. $d = k$) and $\mathbf{w} = \mathbf{r}$ (resp. $\mathbf{v} = \mathbf{c}$), $\mathbf{Z}$ (resp. $\mathbf{W}$) represents a hard clustering $\mathbf{Z} \in \Gamma(n, n)$ (resp. $\mathbf{W} \in \Gamma(d, d)$).

This means that the solutions are already almost a hard partition of the data since $k << n, d$. To obtain a final hard clustering in the strict sense, we assign each row (resp. column) to the one corresponding to the largest value of each row of $\mathbf{Z}$ (resp. $\mathbf{W}$), this should not significantly change the structure of the solution. To illustrate this, we look at figure 1b, it shows the block diagonal structure generated by the product of the two coupling matrices $\mathbf{C} = \mathbf{Z}\mathbf{W}^\top$. We see how it is looks similar to the biclustering produced by the hard block seriation 1a except for a few nonzero entries off the block diagonal that are hard to see immediately.

Intuition for BCOT To explain the intuition behind the proposed approach we have to look at the way the problem is solved. The optimization procedure as described in algorithm 1 consists in alternating between the computation of an optimal transportation map $\mathbf{Z}$ given $\mathbf{W}$ and vice versa. If we look at the solving for $\mathbf{Z}$ given $\mathbf{W}$, the problem can be rewritten as

$$\text{BCOT}(\mathbf{w}, \mathbf{v}, \mathbf{r}, \mathbf{c}) \equiv \min_{\mathbf{Z} \in \Pi(\mathbf{w}, \mathbf{r})} \langle L(\mathbf{B})\mathbf{W}, \mathbf{Z} \rangle \,. \tag{7}$$

This is an optimal transport problem with $L(\mathbf{B})\mathbf{W}$ as the cost matrix. The resulting transportation map $\mathbf{Z}$ can be seen as a sort of a row cluster assignment matrix, if $z_{ih} > 0$ then row i is assigned to cluster h. The same thing holds for $\mathbf{W}$ which can be seen as a column cluster assignment matrix. This also means that since $L(\mathbf{B})$ is the dissimilarity between the rows and the columns, then the cost matrix $L(\mathbf{B})\mathbf{W}$ represents the dissimilarity between rows and row exemplars (or representatives or centroids). In particular, $L(\mathbf{B})_i \mathbf{w}_h$ is the dissimilarity or cost of probability mass transportation between row i and row cluster exemplar h. The reasoning is the same for the columns and optimal coupling $\mathbf{W}$.

Low-Rank Optimal Transport Biclustering is the main purpose of the approach we proposed, but another interesting use case comes up.

Proposition 2. Suppose that the target row and column representatives distributions is the same i.e. $\mathbf{r} = \mathbf{c}$ with no zero entries, then given a solution pair $\mathbf{Z}$ and $\mathbf{W}$ to BCOT, the matrix $\mathbf{Q} = \mathbf{Z}\,\texttt{diag}(1/\mathbf{r})\mathbf{W}^\top$ is an approximation of the optimal transport map that is a solution to problem 2 and whose rank is of at most $\min(\texttt{rank}(\mathbf{Z}), \texttt{rank}(\mathbf{W}))$.

Some recent works [12, 28] suggested that this kind of low-rank regularization is preferable to entropic regularization in some aspects such as for the fact that the rank parameter is easier to select

[1] proofs for the propositions are available under Proofs in the appendix.

since it has simple bounds (an integer between 1 and n) contrary to regularization strength λ in the Sinkhorn algorithm which is continuous.

2.3 Fuzzy biclustering using regularized Optimal Transport

As previously mentionned, using entropic regularization can be interesting due to the many properties it entails like statistical and computational efficiency. However, another property is that the optimal couplings $\mathbf{Z}$ and $\mathbf{W}$ are dense matrices due to the structure of the optimal solution of entropically regularized OT problems. We formulate the problem as follows

$$\text{BCOT}_\lambda(\mathbf{w}, \mathbf{v}, \mathbf{r}, \mathbf{c}) \triangleq \min_{\substack{\mathbf{Z}\in\Pi(\mathbf{w},\mathbf{r}) \\ \mathbf{W}\in\Pi(\mathbf{v},\mathbf{c})}} \left\langle L(\mathbf{B}), \mathbf{Z}\mathbf{W}^\top \right\rangle - \lambda_{\mathbf{Z}} H(\mathbf{Z}) - \lambda_{\mathbf{W}} H(\mathbf{W}) \tag{8}$$

where $\lambda_{\mathbf{Z}}$ and $\lambda_{\mathbf{W}}$ are the regularization parameters.

Fuzzy Block Seriation We propose a fuzzy variant of the block seriation problem which allows us by extension to define a fuzzy variant for BCOT using entropic regularization. Let the fuzzy block seriation problem be defined as

$$\max_{\substack{\mathbf{Z}\in\Gamma_s(n,k) \\ \mathbf{W}\in\Gamma_s(d,k)}} \sum_{i,j,h} b_{ij} z_{ih} w_{jh} + \Omega(\mathbf{Z}, \mathbf{W}) \tag{9}$$

where $\Omega(\mathbf{Z}, \mathbf{W})$ is some regularization term introduced to make the partition matrices $\mathbf{Z}$ and $\mathbf{W}$ dense e.g. entropic regularization or low-rank constraints and $\Gamma_s(n,k) = \{\mathbf{Z} \in \mathbb{R}_+^{n\times k} | \mathbf{Z}\mathbf{1} = \mathbf{1}\}$ is the set of fuzzy partitions. Intuitively, for a solution pair $(\mathbf{Z}, \mathbf{W})$, up to a constant factor, each entry of the block seriation matrix $\mathbf{C} = \mathbf{Z}\mathbf{W}^\top$ can be seen as the probability of its corresponding row and column belonging to the same bicluster i.e. $c_{ij} = \mathbf{z}_i \mathbf{w}_j = \sum_{h=1}^r z_{ih} w_{jh} = p(\mathbf{b}_i, \mathbf{b}'_j) = \sum_{h=1}^r p(\mathbf{b}_i, \mathbf{b}'_j \in h)$. It is easy to see how problem 9 is a related to problem 8 and that the couplings corresponding to solutions to the problem give the probability of membership to the same biclusters for the different rows and columns. Figure 1c shows biclusters produced by the solutions of BCOT_λ. Similarly to BCOT there is a block diagonal structure that is formed. However, there are also several off-block diagonal nonzero entries that represent the probabilities of the row-columns pairs belonging to the same biclusters.

3 Connections to Existing Work

Modularity Maximization in bipartite graphs [2] This model allows to co-cluster binary and contingency matrices by directly maximizing an adapted version of the modularity measure traditionally used for networks. The criterion it optimizes is

$$\max_{\substack{\mathbf{Z}\in\Gamma(n,k) \\ \mathbf{W}\in\Gamma(d,k)}} \sum_{i,j,h} z_{ih} w_{jh} \left(b_{ij} - \frac{b_{.j} b_{i.}}{b_{..}} \right) \tag{10}$$

by setting $L(\mathbf{B}) = -(\mathbf{B} - \frac{1}{b_{..}} \mathbf{B}\mathbf{1}\mathbf{1}^\top \mathbf{B})$, this problem becomes equivalent to ours with the difference lying in the constraints on $\mathbf{Z}$ and $\mathbf{W}$. Thereby BCOT serves as a convex relaxation to this problem.

Modularity-based Sparse Soft Graph Clustering [17] Here the authors proposed to fuzzy variant of the previous problem (although not in a bicustering context but rather traditionanal clustering) whose solution gives for each element of the dataset a probability to belong to a given cluster. Our proposed entropic regularization variant constitutes a sort of extension to bipartite graphs for this problem.

Directional co-clustering with a conscience [27] This model relies on the block von Mises-Fisher mixture model for co-clustering directional data on the unit-sphere. It optimizes the following criterion

$$\max_{\substack{\mathbf{Z}\in\Gamma(n,k) \\ \mathbf{W}\in\Gamma(d,k)}} \sum_{i,j,h} \frac{1}{\sqrt{z_{.h} w_{.h}}} z_{ih} w_{jh} b_{ij} \tag{11}$$

In our formulation, if we define $L(\mathbf{B}) = -\mathbf{B}$ and apply cluster size normalization on the optimal transport plans $\tilde{\mathbf{Z}} = \mathbf{Z}\mathtt{diag}(\mathbf{Z}^\top \mathbf{1})^{-1/2}$ and $\tilde{\mathbf{W}} = \mathbf{W}\mathtt{diag}(\mathbf{W}^\top \mathbf{1})^{-1/2}$ after the computation of $\mathbf{Z}$ and $\mathbf{W}$ respectively in algorithm 1 we obtain a more general version of the algorithm that the authors proposed to solve problem 11.

Bipartite Correlation Clustering [1] In the case where the cost function results in a complete bipartite graph with '+' and '-' edges with a function

$$L(b)_{ij} = \begin{cases} -1 & \text{if } b_{ij} > 0 \\ +1 & \text{otherwise} \end{cases} \tag{12}$$

we get what is known as Bipartite Correlation Clustering. The solution to this problem maximizes the number of agreements i.e. the number of all '+' edges within clusters plus all '-' edges distributed across clusters.

4 Optimization and Complexity

Optimization The block seriation problem being NP-hard means that computing an exact solution is prohibitive. An efficient and widely used heuristic to solve these kind of problems in using block coordinate descent where we alternatingly compute row assignments for fixed column assignments and vice versa. The proposed algorithm available in pseudo-code 1, in each iteration we solve two intermediate optimal transport problems with cost matrices of dimensions $n \times k$ and $d \times k$, since $\mathbf{B}$ is generally sparse and we can define L in way that make $L(\mathbf{B})$ keep a similarly sparse structure, the computation of the intermediate cost matrices $L(\mathbf{B})\mathbf{W}$ and $L(\mathbf{B})^\top \mathbf{Z}$ is quite efficient. Furthermore, we observed that the algorithm does not need many iterations to converge as seen in figure 2, be it for BCOT or BCOT$_\lambda$.

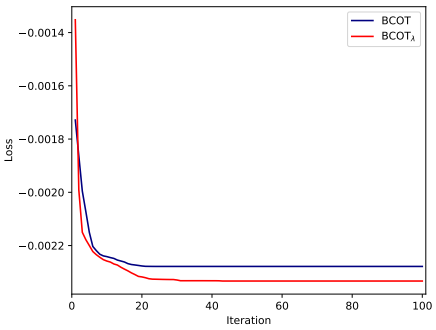


Figure 2: Evolution of the loss for BCOT and BCOT$_\lambda$ on Pubmed.

Algorithm 1: `BCOT`

Input : $\mathbf{B}$ bi-adjacency matrix, $\mathbf{w}$ and $\mathbf{v}$ row and column weights, $\mathbf{r}$ and $\mathbf{c}$ row and column exemplar distributions
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 $\mathbf{W}$;

Proposition 3. The computational complexity of the `BCOT` algorithm 1 when using an exact OT solver is $\mathcal{O}\left(tk\|\mathbf{B}\|_0 + tnk(n+k)\log(n+k) + tdk(d+k)\log(d+k)\right))$ and when using entropic regularization, the complexity is $\mathcal{O}(tk\|\mathbf{B}\|_0 + tkn + tkd)$ where t is the number of iterations.

In table 1, we report the computational and spatial complexities of the different biclustering approaches. Our model has the same spatial complexity as the COOT variants and is a better one than CCOT variants. For the computational complexity, our model should be faster in most cases, our experiments support this observation. For reproducibility, we publicly release our code [2].

[2] `https://anonymous.4open.science/r/BCOT-06C1`

Table 1: Computational and spatial complexity of the different OT biclustering approaches. For the COOT variants, we report complexities for a restricted class of cost functions, for a generic cost, the time complexity is greater. For simplicity, we suppose that $d \in O(n)$ and that for COOT we want a biclustering with the same number of row and column clusters. t denotes the number of iterations and for CCOT, s denotes the number of necessary samplings.

Method	Spatial complexity	Time complexity
CCOT	$O(n^2)$	$O(sn^3)$
CCOT-GW	$O(n^2)$	$O(n^3)$
COOT*	$O(nk)$	$O((n+k)nk + k^2n + t(n+k)nk\log(n+k))$
COOT$^*_\lambda$	$O(nk)$	$O((n+k)nk + k^2n + tnk)$
BCOT	$O(nk)$	$O(k\|\mathbf{B}\|_0 + t(n+k)nk\log(n+k))$
BCOT$_\lambda$	$O(nk)$	$O(k\|\mathbf{B}\|_0 + tnk)$

5 Experiments

We conduct experimentation on term-document matrices. The benefit of using biclustering on this kind of data is that the resulting biclusters contain both documents and the words that characterize which will help us with interpreting clustering of the documents.

5.1 Datasets

We evaluate BCOT on six benchmark document-term datasets, ACM, DBLP, PubMed, Wiki, Ohscal and 20 Newsgroups. Their characteristics are shown in Table 2. ACM, DBLP, Pubmed and Wiki are attributed networks from which we use only the node-level features that correspond to term-document matrices. We also selected the Ohscal collection and 20 Newsgroups as large scale document-term matrices as the computational efficiency benchmarks.

Table 2: Characteristics of datasets .

Dataset	Documents	Terms	Document Clusters	Sparsity (%)
ACM [10]	3025	1870	3	95.52
DBLP [10]	4057	334	4	96.4
PubMed [29]	19717	500	3	89.98
Wiki [32]	2405	4973	17	86.99
Ohscal [16]	11162	11465	10	99.47
20 Newsgroups (NG20) [20]	18846	14390	20	99.41

5.2 Experimental setup

In our experiments, we define the loss function as $L(\mathbf{B}) = -c\mathbf{B}$ where c is selected from $\{1, k, d, n\}$. For BCOTλ, the regularization parameter lambda is selected from $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10\}$. The best hyper-parameters are the ones that minimize the number of empty clusters. In the case of ties, we select according to the value of the Davies-Bouldin index of the partition. We do not use random restarts for any algorithm including k-means. We use the implementation provided by the authors for CCOT, CCOT$_\lambda$ and CCOT-GW. The code for CCOT was not available so we had to implement it based on the one for CCOT-GW. All the reported figures are the averages of 10 runs. All the experiments were performed on the same machine with a RAM of 12GB and Intel(R) Xeon(R) CPU. For OT solvers, we relied on the POT package [11].

5.3 Document clustering

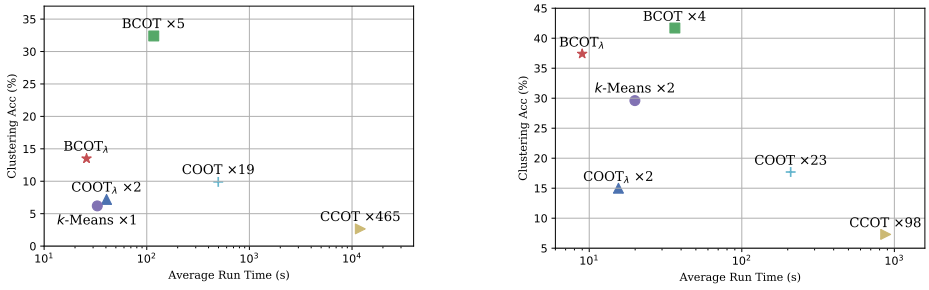
Metrics Here, the evaluation is straightforward, We adopt three popular clustering metrics: clustering accuracy (ACC), normalized mutual information (NMI), adjusted rand index (ARI).

Performance Results for document clustering on ACM, DBLP, PubMed and Wiki are summarized in table 3 in terms of ACC, NMI and ARI. On all these datasets and for all the different metrics either BCOT or BCOT$_\lambda$ offer the best result. Furthermore, on the wiki dataset BCOT$_\lambda$ gives competitive results when compared with state of the art attributed graph clustering methods such as [33] without even having access to the graph structure information of the Wiki citation network.

Table 3: Document clustering performance on the four datasets. OOM denotes out of memory.

Method	ACM			DBLP			PubMed			Wiki		
	Acc	NMI	ARI	Acc	NMI	ARI	Acc	NMI	ARI	Acc	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	**77.6±0.0**	**39.8±0.0**	**45.1±0.0**	**63.2±0.0**	**26.9±0.0**	**28.0±0.0**	53.6±4.5	15.9±1.9	12.9±2.4	51.1±0.0	47.9±0.0	30.9±0.0
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**	**53.1±0.0**	**50.1±0.0**	**32.5±0.0**

Efficiency In Figure 3, we plot the performance of the different methods over their training time relative to that of BCOT$_\lambda$ on the two large scale document-term matrices, 20 Newsgroup and Ohscal. BCOT offers the best accuracy while BCOT$_\lambda$ is fastest method on both datasets. We see that for both BCOT and COOT, the entropic-regularized versions outspeed their exact counterparts and that CCOT suffers from very high computation times mostly due to the fact that a calculation of a pairwise distance matrices on the rows and columns is necessary.



(a) 20 Newsgroups.

(b) Ohscal.

Figure 3: Accuracy over training time on NG20 and Ohscal. BCOT$_\lambda$ is the fastest while achieving competitive performance and BCOT has the best performances whilst being relatively efficient. We use BCOT$_\lambda$ as the reference (e.g. $\times 5$ for BCOT means that it is approximately five times slower than BCOT$_\lambda$). We were not able to benchmark CCOT-GW since it failed to scale to these datasets.

5.4 Term Clustering

Metrics Unlike document clustering, there is no ground truth partition for terms which means we have to look for another way to evaluate term clustering results. A reasonable way to perform evaluation is to analyse the semantic coherence of the clusters found by the different models. With this in mind, we introduce a metric based on the *point mutual information* (PMI). the PMI is a frequently used information theoretic metric for quantifying the relationship between pairs of discrete random variable outcomes. The PMI measure was chosen because prior research [23] has shown that it is closely associated with human judgements in determining word relatedness.

$$\text{PMI}(w_i, w_j) = \log \frac{p(w_i, w_j)}{p(w_i)p(w_j)} \qquad (13) \qquad\qquad \text{PMI}(w_i, w_j) = \log \frac{k_{..} k_{ij}}{k_{i.} k_{.j}} \qquad (14)$$

The PMI measure between the terms w_i and w_j is calculated as in (13). In the context of term clustering, given the word co-occurrence matrix $\mathbf{K} = \mathbf{B}^\top \mathbf{B}$, PMI is estimated as in (14). To evaluate a partition of terms $\mathcal{P}$, we propose a metric based on *intra* and *inter* PMI metrics given by

$$\text{PMI}_{intra}(P) = \sum_{i \in P} \sum_{j \in P} k_{ij} \qquad (15) \qquad\qquad \text{PMI}_{inter}(P) = \sum_{i \in P} \sum_{j \notin P} k_{ij} \qquad (16)$$

Thereby, a good clustering should reveal a large intra-cluster semantic relatedness corresponding to larger PMI values. From *intra* and *inter* PMIs, we propose the following *coherence* index

$$\text{coherence}(\mathcal{P}) = \frac{1}{\sum_{P \in \mathcal{P}} |P|} \sum_{P \in \mathcal{P}} |P| \left(\text{PMI}_{intra}(P) - \text{PMI}_{inter}(P) \right). \qquad (17)$$

Table 4: Term clustering performance on the four datasets. OOM denotes out of memory.

Method	ACM	DBLP	PubMed	Wiki	Ng20	Ohscal
k-Means	0.19±0.01	0.05±0.03	0.31±0.18	0.28±0.02	0.28±0.04	0.01±0.02
CCOT	0.03±0.00	-0.07±0.06	0.02±0.01	0.02±0.00	0.05±0.00	0.06±0.00
CCOT-GW	0.08±0.00	0.03±0.00	OOM	0.01±0.00	OOM	OOM
COOT	0.12±0.01	0.07±0.00	0.14±0.01	0.40±0.00	0.43±0.02	0.23±0.01
COOT$_\lambda$	0.21±0.00	0.04±0.00	-0.00±0.00	-0.08±0.00	-0.02±0.00	-0.13±0.00
BCOT	**0.24±0.00**	**0.20±0.00**	0.51±0.00	**0.65±0.00**	**0.79±0.01**	**0.44±0.00**
BCOT$_\lambda$	**0.24±0.00**	0.16±0.02	**0.57±0.01**	0.59±0.00	0.27±0.00	0.35±0.00

This index relies on the fact that a partition maximizing this criterion has semantically close terms inside the same clusters and contrasting ones across the different groups.

Results Since there is no ground truth number of term clusters, we make use of the cluster number estimations produced by CCOT-GW for all other models so that it is easy to compare coherence values between them since comparing it for different number of clusters would favor models that used the larger cluster number. Table 4 shows the coherence obtained by our approach along with those of the baselines over the different datasets. It is clear that BCOT succeeds in capturing more semantics than the other approaches as, on all datasets, one of two variants of BCOT offer the best coherence value.

5.5 Statistical significance

We perform a nemenyi post hoc test [22, 5] with a confidence level of 90% on the results we obtained in terms of document and term clustering to see if our model outperforms other OT biclustering approaches in a statistically significant manner. To conduct this test we generate 20 performance rankings of the OT biclustering models based on their performance for each dataset and quality metric pair for both document and term clustering. Figure 4 shows the results of the test, we see that it has found two differently performing groups, one comprising BCOT and BCOT$_\lambda$, which gives better results than the other one which comprises the remaining COOT and CCOT variants, meaning the test could not tell apart COOT and CCOT in a statistically significant manner with this specific number of datasets and metrics.

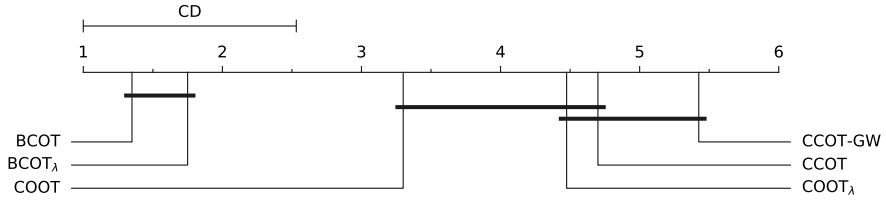


Figure 4: Result of the Nemenyi post hoc test.

6 Conclusion

Clustering and biclustering through optimal transport is till at a nascent stage with many challenges remaining unsolved. Here, we have introduced a novel problem for biclustering using optimal transport that takes into account the sparse nature of certain types dyadic data such document-term matrices to make for more computationally efficient resolution. The problem is posed as a bilinear program that we solve using an efficient coordinate descent algorithm that takes into account the sparse nature of certain types of dyadic data such as document-term matrices. Experiments on a number of document-term datasets suggest that the proposed approach does a good job of finding clusters that correspond to ground truth document classes while generating semantically coherent partitions for the terms. In this setting, our model outperforms recent OT biclustering methods by significant margin, while being more computationally efficient.

References

[1] Nir Ailon, Noa Avigdor-Elgrabli, Edo Liberty, and Anke Van Zuylen. Improved approximation algorithms for bipartite correlation clustering. *SIAM Journal on Computing*, 41(5):1110–1121, 2012.

[2] Michael J Barber. Modularity and community detection in bipartite networks. *Physical Review E*, 76(6):066102, 2007.

[3] Lenaic Chizat, Pierre Roussillon, Flavien Léger, François-Xavier Vialard, and Gabriel Peyré. Faster wasserstein distance estimation with the sinkhorn divergence. *Advances in Neural Information Processing Systems*, 33:2257–2269, 2020.

[4] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.

[5] Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, 7:1–30, 2006.

[6] Inderjit S Dhillon. Co-clustering documents and words using bipartite spectral graph partitioning. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 269–274, 2001.

[7] Inderjit S Dhillon, Subramanyam Mallela, and Dharmendra S Modha. Information-theoretic co-clustering. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 89–98, 2003.

[8] Sara Dolnicar, Sebastian Kaiser, Katie Lazarevski, and Friedrich Leisch. Biclustering: Overcoming data dimensionality problems in market segmentation. *Journal of Travel Research*, 51 (1):41–49, 2012.

[9] Michael B Eisen, Paul T Spellman, Patrick O Brown, and David Botstein. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences*, 95(25):14863–14868, 1998.

[10] Shaohua Fan, Xiao Wang, Chuan Shi, Emiao Lu, Ken Lin, and Bai Wang. One2multi graph autoencoder for multi-view graph clustering. In *Proceedings of The Web Conference 2020*, pages 3070–3076, 2020.

[11] Rémi Flamary, Nicolas Courty, Alexandre Gramfort, Mokhtar Z Alaya, Aurélie Boisbunon, Stanislas Chambon, Laetitia Chapel, Adrien Corenflos, Kilian Fatras, Nemo Fournier, et al. Pot: Python optimal transport. *Journal of Machine Learning Research*, 22(78):1–8, 2021.

[12] Aden Forrow, Jan-Christian Hütter, Mor Nitzan, Philippe Rigollet, Geoffrey Schiebinger, and Jonathan Weed. Statistical optimal transport via factored couplings. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2454–2465. PMLR, 2019.

[13] Jiajun Gu and Jun S Liu. Bayesian biclustering of gene expression data. *BMC genomics*, 9(1): 1–10, 2008.

[14] Rave Harpaz, Hector Perez, Herbert S Chase, Raul Rabadan, George Hripcsak, and Carol Friedman. Biclustering of adverse drug events in the fda's spontaneous reporting system. *Clinical Pharmacology & Therapeutics*, 89(2):243–250, 2011.

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

[16] William Hersh, Chris Buckley, TJ Leone, and David Hickam. Ohsumed: An interactive retrieval evaluation and new large test collection for research. In *SIGIR'94*, pages 192–201. Springer, 1994.

[17] Alexandre Hollocou, Thomas Bonald, and Marc Lelarge. Modularity-based sparse soft graph clustering. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 323–332. PMLR, 2019.

[18] Tjalling C Koopmans and Martin Beckmann. Assignment problems and the location of economic activities. *Econometrica: journal of the Econometric Society*, pages 53–76, 1957.

[19] Charlotte Laclau, Ievgen Redko, Basarab Matei, Younes Bennani, and Vincent Brault. Co-clustering through optimal transport. In *International Conference on Machine Learning*, pages 1955–1964. PMLR, 2017.

[20] Ken Lang. Newsweeder: Learning to filter netnews. In *Proceedings of the Twelfth International Conference on Machine Learning*, pages 331–339, 1995.

[21] F Marcotorchino. Block seriation problems: A unified approach. reply to the problem of h. garcia and jm proth (applied stochastic models and data analysis, 1,(1), 25–34 (1985)). *Applied Stochastic Models and Data Analysis*, 3(2):73–91, 1987.

[22] Peter Bjorn Nemenyi. *Distribution-free multiple comparisons.* Princeton University, 1963.

[23] David Newman, Sarvnaz Karimi, and Lawrence Cavedon. External evaluation of topic models. In *in Australasian Doc. Comp. Symp., 2009*. Citeseer, 2009.

[24] James B Orlin. A polynomial time primal network simplex algorithm for minimum cost flows. *Mathematical Programming*, 78(2):109–129, 1997.

[25] Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport. *Center for Research in Economics and Statistics Working Papers*, (2017-86), 2017.

[26] Ievgen Redko, Titouan Vayer, Rémi Flamary, and Nicolas Courty. Co-optimal transport. *arXiv preprint arXiv:2002.03731*, 2020.

[27] Aghiles Salah and Mohamed Nadif. Model-based von mises-fisher co-clustering with a conscience. In *Proceedings of the 2017 SIAM International Conference on Data Mining*, pages 246–254. SIAM, 2017.

[28] Meyer Scetbon, Marco Cuturi, and Gabriel Peyré. Low-rank sinkhorn factorization. In *International Conference on Machine Learning*, pages 9344–9354. PMLR, 2021.

[29] Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. Collective classification in network data. *AI magazine*, 29(3):93–93, 2008.

[30] Jonathan Templin, Robert A Henson, et al. *Diagnostic measurement: Theory, methods, and applications*. Guilford Press, 2010.

[31] Hua Wang, Feiping Nie, Heng Huang, and Fillia Makedon. Fast nonnegative matrix tri-factorization for large-scale data co-clustering. In *Twenty-Second International Joint Conference on Artificial Intelligence*, 2011.

[32] Cheng Yang, Zhiyuan Liu, Deli Zhao, Maosong Sun, and Edward Y. Chang. Network representation learning with rich text information. In *IJCAI*, 2015.

[33] Hao Zhu and Piotr Koniusz. Simple spectral graph convolution. In *International Conference on Learning Representations*, 2020.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] Last paragraph of the introduction
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...

(a) Did you state the full set of assumptions of all theoretical results? [Yes]
(b) Did you include complete proofs of all theoretical results? [Yes]

3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you citep the creators? [Yes]
   (b) Did you mention the license of the assets? [No]
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A] All assets used are publicly available
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

Appendix A Proofs

Proposition 1. [3] For $\mathbf{w}$, $\mathbf{v}$, $\mathbf{r}$ and $\mathbf{c}$ containing no zeros, the resulting optimal coupling matrices $\mathbf{Z}$ and $\mathbf{W}$ are always an h-almost hard clustering with $h \in \{0, \dots, k-1\}$. Furthermore, when $n = k$ (resp. $d = k$) and $\mathbf{w} = \mathbf{r}$ (resp. $\mathbf{v} = \mathbf{c}$), $\mathbf{Z}$ (resp. $\mathbf{W}$) represents a hard clustering $\mathbf{Z} \in \Gamma(n, n)$ (resp. $\mathbf{W} \in \Gamma(d, d)$).

Proof for *proposition 1.* Problem 2 is a bounded linear program since $\Pi(\mathbf{w}, \mathbf{v})$ is a polytope i.e. a bounded polyhedron. The fundamental theorem of linear programming states that if the feasible set is non-empty then the solution lies in extremity the feasible region. This means that a solution $\mathbf{Z}$ to problem 2 is an extreme point of $\Pi(\mathbf{w}, \mathbf{v})$. We have that the extreme points of $\Pi(\mathbf{w}, \mathbf{v})$ can have at most $n + d - 1$ nonzero elements. To prove this we have to show that the bipartite graph induced by biadjacency matrix $\mathbf{Z}$, the solution to the optimal transport problem has no cycles. The maximum number of edges in an acyclic graph is $|V| - 1$ where $|V|$ is the number of nodes in the graph. Since the number of edges in the bipartite graph induced by biadjacency matrix $\mathbf{Z}$ is $n + d - 1$, the matrix $\mathbf{Z}$ can not have more than $n + d - 1$ nonzero entries. For a detailed proof see proposition 3.3 in [25].

We also have to show that for probability measures $\mathbf{w}$ and $\mathbf{v}$ that have no zero probability events, there is at minimum $\max(n, d)$ number of nonzero elements in $\mathbf{Z}$. This is straightforward since $\mathbf{w}$ and $\mathbf{v}$ contain no zeros, there will always be at least one nonzero element in every row and column of $\mathbf{Z}$ that represents some transfer of mass between elements of $\mathbf{w}$ and $\mathbf{v}$.

In the case of BCOT, we then have that $\mathbf{Z}$ has at most $n + k - 1$ and at least $\max(n, k) = n$ nonzero entries and that $\mathbf{W}$ has at most $d + k - 1$ and at least $\max(d, k) = d$ elements which are both h-almost hard clusterings with $h \in \{0, \dots, k-1\}$.

When $n = k$ and $\mathbf{w} = \mathbf{r}$, the solution $\mathbf{Z}$ is a permutation matrix (up to a constant factor) and the number of nonzero elements in it is exactly n which means that it represents a hard partition $\mathbf{Z} \in \Gamma(n, n)$. The proof for $\mathbf{W}$ is the same.

[3] proofs for the propositions are available under Proofs in the appendix.

Proposition 2. Suppose that the target row and column representatives distributions is the same i.e. $\mathbf{r} = \mathbf{c}$ with no zero entries, then given a solution pair $\mathbf{Z}$ and $\mathbf{W}$ to BCOT, the matrix $\mathbf{Q} = \mathbf{Z}\,\mathtt{diag}(1/\mathbf{r})\mathbf{W}^\top$ is an approximation of the optimal transport map that is a solution to problem 2 and whose rank is of at most $\min(\mathtt{rank}(\mathbf{Z}), \mathtt{rank}(\mathbf{W}))$.

Proof for *proposition 2.* From linear algebra, we have that $rank(\mathbf{Q}) \leq \min(rank(\mathbf{Z}), rank(\mathtt{diag}(1/\mathbf{r})), rank(\mathbf{W}))$. Since $\mathbf{Z}$ and $\mathbf{W}$ cannot have a rank greater than k due to their dimension and that $\mathtt{diag}(1/\mathbf{r})$ is a full rank matrix due to the assumption that $\mathbf{r}$ has no zero entries, we then have that $rank(\mathbf{Q}) \leq \min(rank(\mathbf{Z}), rank(\mathbf{W}))$.

For a proof that $\mathbf{Q}$ is indeed a valid transport map i.e. $\mathbf{Q} \in \Pi(\mathbf{w}, \mathbf{v})$, we redirect the reader to proposition 2.2 in [25].

Proposition 3. The computational complexity of the `BCOT` algorithm 1 when using an exact OT solver is $\mathcal{O}\left(tk\|\mathbf{B}\|_0 + tnk(n+k)\log(n+k) + tdk(d+k)\log(d+k))\right)$ and when using entropic regularization, the complexity is $\mathcal{O}(tk\|\mathbf{B}\|_0 + tkn + tkd)$ where t is the number of iterations.

Proof for *proposition 3.* We suppose that $L(\mathbf{B})$ is a sparse matrix with the same number of nonzero entries as $\mathbf{B}$. The complexity of computing $L(\mathbf{B})\mathbf{W}$ and $L(\mathbf{B})\mathbf{W}$ in the BCOT algorithm is $\mathcal{O}(k\|\mathbf{B}\|_0)$.

The optimal transport problem can be formulated and solved as the Earth Mover's Distance (EMD) problem using any algorithm for minimum-cost flow problem such as one of the many variants of network simplex algorithm. Authors in [24] proposed an algorithm for the network simplex in $\mathcal{O}(|V||E|\log|V|)$ where $|V|$ is the number of nodes and $|E|$ is the number of edges in the network. In our case when solving the EMD for $\mathbf{Z}$ and cost matrix $L(\mathbf{B})\mathbf{W}$, the number of nodes is $|V| = n + k$ and the number of edges is $|E| = nk$ which means that the complexity of the operation is $\mathcal{O}(nk(n+k)\log(n+k))$. When computing the optimal transport map for $\mathbf{W}$, for cost matrix $L(\mathbf{B})^\top\mathbf{Z}$, the complexity is $\mathcal{O}(dk(d+k)\log(d+k))$. The overall complexity of the BCOT algorithm is then $\mathcal{O}(k\|\mathbf{B}\|_0) + tnk(n+k)\log(n+k) + tdk(d+k)\log(d+k))$

When using entropic regularization the complexity is smaller since the computation of the optimal map requires only a transformation of the inputs matrix which takes $\mathcal{O}(nk)$ in the $\mathbf{Z}$ computation step and $\mathcal{O}(dk)$ for $\mathbf{W}$. The ensuing application of the Sinkhorn-Knopp algorithm on the transformed matrices has a complexity $\mathcal{O}(tnk)$ and $\mathcal{O}(tdk)$ for $\mathbf{Z}$ and $\mathbf{W}$ respectively, where t is the number of iterations necessary. The overall complexity of BCOT$_\lambda$ is then $\mathcal{O}(k\|\mathbf{B}\|_0) + tnk + tdk)$, here t includes the number of iterations of our algorithm as well as that of Sinkhorn-Knopp.