Regularized bi-directional co-clustering

Séverine Affeldt, Lazhar Labiod, Mohamed Nadif

Statistics and Computing manuscript No.
(will be inserted by the editor)

Regularized Bi-Directional Co-Clustering

Séverine Affeldt · Lazhar Labiod · Mohamed Nadif

Received: date / Accepted: date

Abstract The simultaneous clustering of documents and words, known as *co-clustering*, has proved to be more effective than one-sided clustering in dealing with sparse high-dimensional datasets. By their nature, text data are also generally unbalanced and directional. Recently, the von Mises-Fisher (vMF) mixture model was proposed to handle unbalanced data while harnessing the directional nature of text. In this paper we propose a general co-clustering framework based on a matrix formulation of vMF model-based co-clustering. This formulation leads to a flexible framework for text co-clustering that can easily incorporate both *word-word* semantic relationships and *document-document* similarities. By contrast with existing methods, which generally use an additive incorporation of similarities, we propose a *bi-directional multiplicative* regularization that better encapsulates the underlying text data structure. Extensive evaluations on various real-world text datasets demonstrate the superior performance of our proposed approach over baseline and competitive methods, both in terms of clustering results and co-cluster topic coherence.

Keywords Co-clustering · Regularization · Information Retrieval · text mining

Séverine Affeldt
Université de Paris, CNRS, Centre Borelli
F-75006 Paris, France
E-mail: severine.affeldt@u-paris.fr

Lazhar Labiod
Université de Paris, CNRS, Centre Borelli
F-75006 Paris, France
E-mail: lazhar.labiod@u-paris.fr

Mohamed Nadif
Université de Paris, CNRS, Centre Borelli
F-75006 Paris, France
E-mail: mohamed.nadif@u-paris.fr

1 Introduction

The simultaneous partitioning of features and objects into consistent homogeneous blocks, referred as to *co-clusters*[1], is a successful extension of one-sided clustering that can make large datasets easier to analyze (Hartigan, 1972; Bock, 1979; Govaert, 1983; Vichi, 2001; Van Mechelen et al., 2004; Rocci and Vichi, 2008; Govaert and Nadif, 2008, 2013; Bock, 2020). Starting from a data matrix, a co-cluster can be defined as a submatrix whose elements have a particular pattern in common. The basic idea behind co-clustering is to identify a structure that is shared by objects and features through their permutations. A variety of co-clustering methods have been applied in different areas, such as in bioinformatics (Madeira and Oliveira, 2004; Cho and Dhillon, 2008; Tanay et al., 2005; Hanczar and Nadif, 2012) to group genes and experimental conditions, in collaborative filtering (Hofmann and Puzicha, 1999; Deodhar and Ghosh, 2010) to group users and items, and in text mining (Dhillon et al., 2003; Ailem et al., 2017a; Govaert and Nadif, 2018; Salah and Nadif, 2019; Role et al., 2019) to group words[2] and documents. Through its ability to relate rows and columns, co-clustering generally gives better results than clustering along a single dimension. Besides, co-clustering makes an implicit adaptive dimensionality reduction that allows the use of efficient scalable algorithms for high-dimensional sparse text data. This is crucial in text mining, since

[1] Given a data matrix $\mathbf{X} = (x_{ij})$, $i \in I$, $j \in J$, a co-cluster is a submatrix defined by $I_k \times J_\ell (I_k \subseteq I, J_\ell \subseteq J)$.

[2] The generally understood difference between *words* and *terms* is that *terms* are *words* used in a particular specialized field. The *words* that we are concerned with in co-clustering can in most cases also be qualified as *terms*, and consequently we use *words* and *terms* interchangeably in this paper.

the exponential growth of online documents has created an urgent need for effective methods in handling and interpreting high-dimensional sparse *document-term* matrices, i.e. matrices where documents are represented in the space of terms, and vice versa. Most importantly, text co-clustering can identify the most discriminating words that characterize topics in document classes.

Standard text-focused co-clustering approaches seek to relate documents and words. They do not usually attempt to incorporate side information such as semantic relationships between words, or similarities in document content. The clustering of documents relating to the same topics might nevertheless benefit from additional information about word similarities, since these documents can be expected to contain semantically related terms. Conversely, word clustering might usefully harness side information about (similarities in) document content, given that it is the documents that provide the context for the words. Side information on document latent space *and* on word latent space could together improve the co-clustering of document-text data.

2 Related work

Inspired by the recent success of neural word embedding models, Ailem et al. (2017b) proposed performing NMF (Non-negative Matrix Factorization) jointly on the document-word and word-context matrices, with shared word factors. Recently, an extension of NMTF-based (Non-negative Matrix Tri-Factorization) co-clustering, namely `WC-NMTF` (Word Co-Occurence regularized NMTF) (Salah et al., 2018), a technique that takes account of semantic relationships between terms, has successfully been applied on various text datasets. As well as being high-dimensional and sparse, text data are also heavily unbalanced, and co-clustering methods that focus on document-term matrices need take this into account. The `DCC` algorithm (Directional Co-clustering with a Conscience) (Salah and Nadif, 2019), has been shown to be particularly suited to tackling this issue. `DCC` uses the von Mises-Fisher (vMF) mixture model and introduces a *conscience* mechanism (DeSieno, 1988; Ahalt et al., 1990) to avoid empty or highly unbalanced clusters (Banerjee and Ghosh, 2004). It exploits the fact that text data are naturally *directional*, which means that only the directions of data vectors are relevant, and not their magnitude (Mardia and Jupp, 2009). In contrast with `WC-NMTF`, `DCC` does not use any regularization.

In this work we harness the directional property of text data and describe a *Regularized Bi-Directional Co-clustering* (`RBDCo`) algorithm for document-term data. The bi-directional aspect of our approach resides in the use of side information for the two dimensions of the document-term matrix. The primary contribution of this work is a general framework based on a matrix formulation of vMF-based co-clustering. A significant outcome of this novel formulation is a very rich, flexible framework for text co-clustering that allows an easy multiplicative regularization on both the *word-word* semantic relationships and the *document-document* content similarities. In contrast with existing methods, which generally rely on additive incorporation of similarities, we propose a *bi-directional multiplicative* regularization that better encapsulates the underlying text data structure. Another contribution of this work is an original method for evaluating the coherence of word clusters. Experimental results on various real-life datasets provide clear empirical evidence of the effectiveness of our co-clustering framework.

The rest of the paper is organized as follows. After reviewing the von Mises-Fisher-based clustering method in Section 3, we introduce a matrix view of a derived co-clustering algorithm, namely Directional Co-Clustering with a Conscience (`DCC`), in Section 4. We then show in Section 5 how a generalized regularization framework can be built from the von Mises-Fisher model while taking into account the directional property of text data, and this section also looks at how our generalized framework is linked to a variety of other co-clustering approaches. Section 6 is devoted to comparative numerical experiments that demonstrate the effectiveness of our generalized regularization framework. We conclude and suggest future paths in Section 7.

Notation. Let $\mathbf{X} = (x_{ij})$ be a data matrix of size $n \times d$, $x_{ij} \in \mathbb{R}$. The i^{th} row of this matrix is represented by a vector $\mathbf{x}_i = (x_{i1}, \ldots, x_{id})^\top$, where $\top$ denotes the transpose. The partition of the set of rows into g clusters can be represented by a classification matrix $\mathbf{Z}$ of elements z_{ik} in $\{0, 1\}$ satisfying $\sum_{k=1}^{g} z_{ik} = 1$. We denote by $z_{.k} = \sum_i z_{ik}$ the cardinality of the kth row cluster. The notation $\mathbf{z} = (z_1, \ldots, z_n)^\top$, where $z_i \in \{1, \ldots, g\}$ corresponds to the cluster label of i, will be also used. Similarly, the notations $\mathbf{W} = (w_{jk})$, $w_{jk} \in \{0, 1\}$ satisfying $\sum_{k=1}^{g} w_{ik} = 1$, $\mathbf{w} = (w_1, \ldots, w_d)$, where $w_j \in \{1, \ldots, g\}$ represents the cluster label of j represented by the vector $\mathbf{x}^j$, and $w_{.k} = \sum_j w_{jk}$ the cardinality of the kth column cluster, will be used to represent the partition of the set of columns.

3 Directional Co-clustering

Mixture models have undoubtedly made a very useful contribution to clustering in that they offer considerable flexibility (McLachlan and Peel, 2004). A mixture

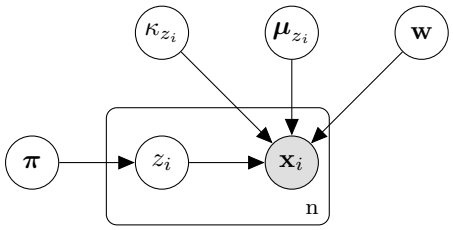


Fig. 1: Graphical model. The parameters μ_{z_i} and κ_{z_i} are the mean direction and concentration parameter of vMF distribution $f(\mathbf{x}_i|\boldsymbol{\mu}^{\mathbf{w}}_{z_i}, \kappa_{z_i}) = c_d(\kappa)\exp[\kappa_{z_i}(\boldsymbol{\mu}^{\mathbf{w}}_{z_i})^\top \mathbf{x}_i]$, respectively. The normalization term takes the following form $c_d(\kappa) = \kappa^{\frac{d}{2}-1}(2\pi)^{\frac{d}{2}} I_{\frac{d}{2}-1}(\kappa)$, where $I_r(\kappa)$ represents the modified Bessel function of the first kind and order r. For more details on the vMF distribution, the reader can refer to Mardia and Jupp (2009).

of von Mises-Fisher (vMF) distributions can be a wise choice (Banerjee et al., 2005; Salah and Nadif, 2017b) when dealing with directional data distributed on a unit hypersphere $\mathbb{S}$. In fact, this model is one of the most appropriate models for clustering high-dimensional sparse data such as the document-term matrices encountered in text mining. In this kind of application it has been empirically demonstrated that vMF-based clustering methods perform better than a number of existing approaches; see, e.g., (Zhong and Ghosh, 2005; Gopal and Yang, 2014).

In (Salah and Nadif, 2017a, 2019) the authors proposed a vMF mixture model for co-clustering. The graphical model is depicted in Figure 1, and its probability density function is given by

$$f(\mathbf{x}_i|\Theta) = \sum_{k=1}^{g} \pi_k f_k(\mathbf{x}_i|\boldsymbol{\mu}^{\mathbf{w}}_{z_i}, \kappa_{z_i}),$$

where $\Theta = \{\boldsymbol{\pi} = (\pi_1, \ldots, \pi_g),\ \boldsymbol{\mu}^{\mathbf{w}} = (\boldsymbol{\mu}^{\mathbf{w}}_1, \ldots, \boldsymbol{\mu}^{\mathbf{w}}_g), \kappa_1, \ldots, \kappa_g\}$. Note that $\boldsymbol{\mu}^{\mathbf{w}}$ depends on $\mathbf{w}$, i.e, $w_j = k$ if the j^{th} column belongs to k^{th} cluster, that is "associated" with the k^{th} row cluster.
Note that with this model the d-dimensional centroids $\boldsymbol{\mu}^{\mathbf{w}}_k = (\mu_{k1}, \ldots, \mu_{k1}, \ldots, \mu_{kg}, \ldots, \mu_{kg})^\top$ such that μ_{kh} is repeated $w_{.h}$ times, and $\mu_{kh} = 0$ for all $k \neq h$ are assumed to be orthonormal. The parameter κ_k denotes the concentration of the k^{th} distribution. The proportion of points $\mathbf{x}_i$ generated from the k^{th} component is denoted by the parameter π_k, such that $\sum_k \pi_k = 1$ and $\pi_k > 0$, $\forall k \in \{1, \ldots, g\}$. The complete data log-likelihood is thereby given by

$$L_c(\Theta|\mathbf{X}, \mathbf{Z}) = \sum_k z_{.k} \log \pi_k + \sum_k z_{.k} \log(c_d(\kappa_k)) + \sum_{i,k} z_{ik}\kappa_k (\boldsymbol{\mu}^{\mathbf{w}}_k)^\top \mathbf{x}_i.$$

Assuming that all the mixing proportions are equal, i.e., $\pi_k = \frac{1}{g}$, $\forall k$ (this does not penalize the quality of clustering as a result of the conscience mechanism) and for high dimensionality, i.e., large order $r = d/2 - 1$, a small κ_k (due to the sparsity) gives $4(r+1) + \kappa_k^2 \approx 4(r+1)$ and then $\log c_d(\kappa_k) \approx -\frac{d}{2}\log 2\pi - \log c$ where $c = \frac{4(r+1)}{2^{r+2}(r+1)!}$; for details the reader can refer to Salah and Nadif (2017a). Thus $L_c(\Theta|\mathbf{X}, \mathbf{Z})$ becomes

$$L_c(\Theta|\mathbf{X}, \mathbf{Z}) = \sum_{i,k} z_{ik}\kappa_k (\boldsymbol{\mu}^{\mathbf{w}}_k)^\top \mathbf{x}_i + constant. \tag{1}$$

The concentration parameter κ_k is made inversely proportional to the root square of the number of elements in cluster k, i.e., $\kappa_k = 1/\sqrt{z_{.k}}$ where the row assignments are done by maximizing a weighted `Skmeans`-like criterion where the weights $1/\sqrt{z_{.k}}$ ($k \in \{1, \ldots, g\}$) discourage the absorption of new objects by larger clusters. This is also the case for the column assignments, where $w_{.k}$ is the cardinality of the kth column cluster and $\boldsymbol{\mu}^{\mathbf{z}}_k = (\mu_{k1}, \ldots, \mu_{k1}, \ldots, \mu_{kg}, \ldots, \mu_{kg})^\top$ its n-dimensional centroid. To sum up, following (Salah and Nadif, 2017a) we have,

$$\begin{aligned} L_c(\Theta|\mathbf{X}, \mathbf{Z}) &\equiv \sum_{i,k} z_{ik} \frac{1}{\sqrt{z_{.k}}} (\boldsymbol{\mu}^{\mathbf{w}}_k)^\top \mathbf{x}_i \\ &\equiv \sum_{j,k} w_{jk} \frac{1}{\sqrt{w_{.k}}} (\boldsymbol{\mu}^{\mathbf{z}}_k)^\top \mathbf{x}^j. \end{aligned} \tag{2}$$

$A \equiv B$ means that optimizing A is equivalent to optimizing B. The maximizing of mean directions (2) is defined as follows: $\mu_{kh} = 1/\sqrt{w_{.k}}$ if $k = h$, and $\mu_{kh} = 0$ for all $k \neq h$. Similarly, we deduce $\boldsymbol{\mu}^{\mathbf{w}}_k$ from $1/\sqrt{z_{.k}}$. Note that Θ is now reduced to $\boldsymbol{\mu}^{\mathbf{w}}$ and $\boldsymbol{\mu}^{\mathbf{z}}$ the centers of row and column clusters.

The authors have derived a co-clustering algorithm that we refer to as *Directional Co-clustering with a Conscience* (`DCC`), tailored to high-dimensional sparse data (Alg. 1). The `DCC` algorithm intertwines row and column clusterings at each step so as to optimize $L_c(\Theta|\mathbf{X}, \mathbf{Z})$. Integrating the conscience mechanism makes it possible to avoid highly skewed solutions with empty or very small/large clusters. Applied on unbalanced document-term matrices, `DCC` proves more suitable than most existing co-clustering approaches for handling directional data distributed on the surface of a unit-hypersphere.

4 Matrix view of the DCC Model

In this section we propose a matrix formulation of `DCC`. To this end, we first make use of the matrix formulation of $\boldsymbol{\mu}^{\mathbf{w}} = (\boldsymbol{\mu}^{\mathbf{w}}_1, \ldots, \boldsymbol{\mu}^{\mathbf{w}}_k, \ldots, \boldsymbol{\mu}^{\mathbf{w}}_g) \in \mathbb{R}^{d\times g}$ and $\boldsymbol{\mu}^{\mathbf{z}} = (\boldsymbol{\mu}^{\mathbf{z}}_1, \ldots, \boldsymbol{\mu}^{\mathbf{z}}_k, \ldots, \boldsymbol{\mu}^{\mathbf{z}}_g) \in \mathbb{R}^{n\times g}$. Let us consider the binary classification matrices $\mathbf{Z} \in \{0, 1\}^{n\times g}$ and

Algorithm 1 Directional Co-clustering with a Conscience (DCC).

Input: $\mathbf{X}$ ($\mathbf{x}_i \in \mathbb{S}^{d-1}$ the unit hypersphere), g the number of co-clusters.
Output: $\mathbf{Z}$, $\mathbf{W}$ and Θ
Steps:
Initialization: $\Theta \leftarrow \Theta^{(0)}$;
repeat
1. Assignment of objects:
for $i = 1$ **to** n **do**
$z_i = \arg\max_{k'} \frac{1}{\sqrt{z_{.k'}}} cos(\boldsymbol{\mu}_{k'}^{\mathbf{w}}, \mathbf{x}_i)$
end for
2. Assignment of features:
for $j = 1$ **to** d **do**
$w_j = \arg\max_{k'} \frac{1}{\sqrt{w_{.k'}}} cos(\boldsymbol{\mu}_{k'}^{\mathbf{z}}, \mathbf{x}^j)$
end for
3. Computation of μ_{kk}'s maximizing (2):
for $k = 1$ **to** g **do**
$\mu_{kk} \leftarrow \frac{1}{\sqrt{w_{.k}}}$
end for
until the (2) value change is small or there is no change

$\mathbf{W} \in \{0,1\}^{d \times g}$, where the cluster sizes of $\mathbf{Z}$ and $\mathbf{W}$ are on the diagonal of $\mathbf{D_z} = \mathbf{Z}^\top\mathbf{Z}$ and $\mathbf{D_w} = \mathbf{W}^\top\mathbf{W}$, respectively. We therefore have

$$\boldsymbol{\mu}^{\mathbf{w}} = \mathbf{W}\mathbf{D}_{\mathbf{w}}^{-0.5} = \widetilde{\mathbf{W}}. \quad \text{and} \quad \boldsymbol{\mu}^{\mathbf{z}} = \mathbf{Z}\mathbf{D}_{\mathbf{z}}^{-0.5} = \widetilde{\mathbf{Z}}. \quad (3)$$

Using the above matrix formulations, and given a document-term matrix $\mathbf{X}$, the optimization of the complete data log-likelihood of $\mathbf{X}$ $L_c(\Theta|\mathbf{X}, \mathbf{Z})$ in (2) leads to

$$\begin{aligned}\sum_{i,k} z_{ik} \frac{1}{\sqrt{z_{.k}}} (\boldsymbol{\mu}_k^{\mathbf{w}})^\top \mathbf{x}_i &\equiv Tr(\mathbf{Z}^\top \mathbf{X}\widetilde{\mathbf{W}}\mathbf{D}_{\mathbf{z}}^{-0.5}) \\ &\equiv Tr(\mathbf{W}^\top \mathbf{X}^\top \widetilde{\mathbf{Z}}\mathbf{D}_{\mathbf{w}}^{-0.5}). \quad (4)\end{aligned}$$

In virtue of (3) the formulas for updating the algorithm DCC can be rewritten in matrix form:

$$\mathbf{Z} = \mathbf{Binarize}(\mathbf{X}\widetilde{\mathbf{W}}\mathbf{D}_{\mathbf{z}}^{-0.5})$$
and
$$\mathbf{W} = \mathbf{Binarize}(\mathbf{X}^\top\widetilde{\mathbf{Z}}\mathbf{D}_{\mathbf{w}}^{-0.5}), \quad (5)$$

where $\mathbf{Binarize}(\mathbf{B})$, means $\forall i; \mathbf{b}_{ik} = \text{argmax}_{k'}\, \mathbf{b}_{ik'}$. The update rules show the mutual interaction between the set of documents and the set of words. If a word w is common to many documents associated with a co-cluster $\mathbf{C}_i$, then the word w will be associated with the co-cluster $\mathbf{C}_i$. Conversely, if a document contains many words that are associated with a co-cluster $\mathbf{C}_i$, then the document will be associated with the co-cluster $\mathbf{C}_i$. To find the desired solution for the partitions $\mathbf{Z}$ and $\mathbf{W}$, we can alternate the two rules (5) until a fixed point is reached.

5 Regularized Bi-directional Co-clustering

Text data co-clustering relies on the duality between the document and word spaces, i.e. documents can be grouped based on their distribution with respect to words, while words can be grouped based on their distribution with respect to documents. Existing co-clustering algorithms generally rely on the input *document-term* matrix $\mathbf{X}$. While some of them consider also pure *word-word* semantic correlations, such as proposed by Salah et al. (2018), co-clustering methods fail to consider side information arising from both *word-word* semantic correlations and *document-document* similarities. To fill this gap, we propose a *Regularized Bi-directional Co-clustering* (RBDCo) based on an appropriate matrix formulation. We construct two similarity matrices – the first, $\mathbf{S}_r$, for similarities in document content, and the second, $\mathbf{S}_c$, for semantic correlations between words: see Section 5.5 – in order to exploit hidden structures in documents and words. Our co-clustering method is then formulated as an iterative matrix multiplication process with two similarity matrices as regularizers, which means that the partitions of documents and words need to be smoothed with respect to document similarities and semantic correlations of words.

Formally, let us consider the block matrix $[\mathbf{Z}\ \mathbf{W}]^\top$. Utilizing the diagonal structure of $\begin{bmatrix} 0 & \mathbf{X} \\ \mathbf{X}^\top & 0 \end{bmatrix}$, we can write the update rules of the aforementioned DCC as an iterative matrix multiplication procedure based on the appropriate block matrices:

$$\begin{bmatrix} \mathbf{Z} \\ \mathbf{W} \end{bmatrix} \leftarrow \begin{bmatrix} 0 & \mathbf{X} \\ \mathbf{X}^\top & 0 \end{bmatrix} \begin{bmatrix} \widetilde{\mathbf{Z}}\mathbf{D}_{\mathbf{w}}^{-0.5} \\ \widetilde{\mathbf{W}}\mathbf{D}_{\mathbf{z}}^{-0.5} \end{bmatrix} = \begin{bmatrix} \mathbf{X}\widetilde{\mathbf{W}}\mathbf{D}_{\mathbf{z}}^{-0.5} \\ \mathbf{X}^\top\widetilde{\mathbf{Z}}\mathbf{D}_{\mathbf{z}}^{-0.5} \end{bmatrix}. \quad (6)$$

This formulation[3] clearly shows how DCC utilizes the duality between document and word spaces. The document clustering $\mathbf{Z}$ is derived as a weighted projection of the data matrix $\mathbf{X}$ on the subspace spanned by the word partition $\mathbf{W}$. Similarly, the word partition is derived as a weighted projection of the data matrix $\mathbf{X}$ on the subspace spanned by the document partition $\mathbf{Z}$.

5.1 RBDCo method

We propose a regularized bi-directional data co-clustering method, RBDCo, that draws advantage from our block matrix formulation of DCC (Eq. 6) and harnesses two regularized data matrices, $\mathbf{M_z}$ and $\mathbf{M_w}$ with values in

[3] to simplify notation, in the rest of the paper the symbol $\leftarrow$ in the updating rules for $\mathbf{Z}$ and $\mathbf{W}$ will indicate that the function $\mathbf{Binarize}(.)$ is applied to the formulas of both $\mathbf{Z}$ and $\mathbf{W}$.

$\{\mathbf{X}, \mathbf{S}_r\mathbf{X}, \mathbf{X}\mathbf{S}_c, \mathbf{S}_r\mathbf{X}\mathbf{S}_c\}$ (see Section 5.5). The objective of RBDCo is to optimize the following trace criterion:

$$\begin{aligned} J_{RBDCo} &\equiv \frac{1}{2} Tr\left(\begin{bmatrix} \mathbf{Z} \\ \mathbf{W} \end{bmatrix}^\top \begin{bmatrix} 0 & \mathbf{M_z} \\ \mathbf{M_w}^\top & 0 \end{bmatrix} \begin{bmatrix} \widetilde{\mathbf{Z}}\mathbf{D}_\mathbf{w}^{-0.5} \\ \widetilde{\mathbf{W}}\mathbf{D}_\mathbf{z}^{-0.5} \end{bmatrix} \right) \\ &\equiv \frac{1}{2} Tr\left(\mathbf{Z}^\top \mathbf{M_z} \widetilde{\mathbf{W}} \mathbf{D}_\mathbf{z}^{-0.5}) + Tr(\mathbf{W}^\top \mathbf{M}_\mathbf{w}^\top \widetilde{\mathbf{Z}} \mathbf{D}_\mathbf{w}^{-0.5} \right) \\ &\equiv \frac{1}{2} Tr\left(\widetilde{\mathbf{Z}}^\top (\mathbf{M_z} + \mathbf{M_w}) \widetilde{\mathbf{W}} \right). \end{aligned} \tag{7}$$

The data co-clustering task is carried out by iteratively computing $\mathbf{Z}$ and $\mathbf{W}$ based on the interplay between the two updating rules derived from the maximization of the objective criterion J_{RBDCo},

$$\begin{bmatrix} \mathbf{Z} \\ \mathbf{W} \end{bmatrix} \leftarrow \begin{bmatrix} 0 & \mathbf{M_z} \\ \mathbf{M_w}^\top & 0 \end{bmatrix} \begin{bmatrix} \widetilde{\mathbf{Z}}\mathbf{D}_\mathbf{w}^{-0.5} \\ \widetilde{\mathbf{W}}\mathbf{D}_\mathbf{z}^{-0.5} \end{bmatrix} = \begin{bmatrix} \mathbf{M_z}\widetilde{\mathbf{W}}\mathbf{D}_\mathbf{z}^{-0.5} \\ \mathbf{M}_\mathbf{w}^\top \widetilde{\mathbf{Z}}\mathbf{D}_\mathbf{w}^{-0.5} \end{bmatrix}. \tag{8}$$

If we set $\mathbf{M_z} = \mathbf{M_w} = \mathbf{S}_r\mathbf{X}\mathbf{S}_c$, this leads to,

$$\begin{bmatrix} \mathbf{Z} \\ \mathbf{W} \end{bmatrix} \leftarrow \begin{bmatrix} \mathbf{S}_r\mathbf{X}\mathbf{S}_c\widetilde{\mathbf{W}}\mathbf{D}_\mathbf{z}^{-0.5} \\ \mathbf{S}_c\mathbf{X}^\top\mathbf{S}_r\widetilde{\mathbf{Z}}\mathbf{D}_\mathbf{w}^{-0.5} \end{bmatrix}. \tag{9}$$

The RBDCo updating rules in (8) mutually exploit the duality of the documents and words, and reinforce their joint clustering with bi-directional multiplicative regularizations using $\mathbf{S}_c$ and $\mathbf{S}_r$. By generating explicit assignments of words, RBDCo produces interpretable descriptions of the resulting co-clusters. In addition, by iteratively alternating between the two updating rules, RBDCo performs an implicit adaptive word selection at each iteration and flexibly measures the distances between documents. It therefore works well with high-dimensional sparse data. The conscience mechanism embedded in RBDCo also means that it performs well with unbalanced document-term data (see Section 6). Algorithm 2 details the alternating procedure of RBDCo.

In the case of a *symmetric* regularization, we set $\mathbf{M_z} = \mathbf{M_w} = \mathbf{M}$, i.e. the same regularization is applied to the update rules for both $\mathbf{Z}$ and $\mathbf{W}$. The objective of RBDCo is then reduced to

$$J_{RBDCo} \equiv Tr(\mathbf{Z}^\top \mathbf{M} \widetilde{\mathbf{W}} \mathbf{D}_\mathbf{z}^{-0.5}). \tag{10}$$

If $\mathbf{M_z} = \mathbf{M_w} = \mathbf{X}$, then RBDCo is equivalent to the particular case DCC. In fact, comparing (8) and (6), it is easy to see that RBDCo generalizes DCC – DCC being RBDCo with all similarity matrices equal to $\mathbf{I}$ –.

5.2 A generalized regularization framework

RBDCo offers a highly flexible framework in the context of text data co-clustering for the integration of supplementary information embedded in matrices that encapsulate similarities between documents and semantic correlations between words. We distinguish two types of regularization: *(i)* **symmetric regularization**, which consists in the application of the same regularization for the update of $\mathbf{Z}$ and $\mathbf{W}$ ($\mathbf{M_z} = \mathbf{M_w}$), and *(ii)* **asymmetric regularization**, which considers different regularizations for the update of $\mathbf{Z}$ and $\mathbf{W}$ ($\mathbf{M_z} \neq \mathbf{M_w}$). Table 1 summarizes the different symmetric and asymmetric configurations covered by RBDCo.

Algorithm 2 Regularized Bi-Directional Co-Clustering (RBDCo).

Input: $\mathbf{X}$ ($\mathbf{x}_i \in \mathbb{S}^{d-1}$), g number of co-clusters, $\mathbf{S}_r$, $\mathbf{S}_c$
Output: partitions $\mathbf{Z}$ and $\mathbf{W}$
Initialization: random initialization of $\mathbf{Z}$ and $\mathbf{W}$
repeat
1. Assignment of objects (8)
   - $\mathbf{Z} \leftarrow \mathbf{M_z}\widetilde{\mathbf{W}}\mathbf{D}_\mathbf{z}^{-0.5}$
   - **Binarize** $\mathbf{Z}$: $\forall i \quad z_i = \arg\max_{k'} z_{ik'}$
2. Assignment of features (8)
   - $\mathbf{W} \leftarrow \mathbf{M}_\mathbf{w}^\top \widetilde{\mathbf{Z}}\mathbf{D}_\mathbf{w}^{-0.5}$
   - **Binarize** $\mathbf{W}$: $\forall j \quad w_j = \arg\max_{k'} w_{jk'}$

until convergence of J_{RBDCo} (7)

Table 1: Description of RBDCO regularization schemes

Regularization type		Data Regularization $\mathbf{M_z}$	Data Regularization $\mathbf{M_w}$	Notation
symmetric		$\mathbf{X}$	$\mathbf{X}$	RBDCo$_{[I,I]}$
symmetric		$\mathbf{S}_r\mathbf{X}$	$\mathbf{S}_r\mathbf{X}$	RBDCo$_{[S_r,S_r]}$
symmetric		$\mathbf{X}\mathbf{S}_c$	$\mathbf{X}\mathbf{S}_c$	RBDCo$_{[S_c,S_c]}$
symmetric		$\mathbf{S}_r\mathbf{X}\mathbf{S}_c$	$\mathbf{S}_r\mathbf{X}\mathbf{S}_c$	RBDCo$_{[S_rS_c,S_rS_c]}$
asymmetric		$\mathbf{S}_r\mathbf{X}$	$\mathbf{X}$	RBDCo$_{[S_r,I]}$
asymmetric		$\mathbf{X}\mathbf{S}_c$	$\mathbf{X}$	RBDCo$_{[S_c,I]}$
asymmetric		$\mathbf{X}$	$\mathbf{S}_r\mathbf{X}$	RBDCo$_{[I,S_r]}$
asymmetric		$\mathbf{X}$	$\mathbf{X}\mathbf{S}_c$	RBDCo$_{[I,S_c]}$
asymmetric	*(uncross)*	$\mathbf{S}_r\mathbf{X}$	$\mathbf{X}\mathbf{S}_c$	RBDCo$_{[S_r,S_c]}$
asymmetric	*(cross)*	$\mathbf{X}\mathbf{S}_c$	$\mathbf{S}_r\mathbf{X}$	RBDCo$_{[S_c,S_r]}$

The different regularization schemes described in Table 1 highlight the flexibility of the proposed model and the connections with other approaches that can derive from it (Section 5.3). In our study, we indicated and justified the choice of the model retained for the case of *document-term* data on which we focused (see *Particular cases* and Section 6.3.1). For other types of data, the user may favour one model over another. An automatic model selection could be part of an interesting future study.

Particular cases The high degree of flexibility offered by RBDCo for multiplicative bi-directional regularizations gives rise to a variety of versions. For instance, if the identity matrix is assigned to the right-hand side of the regularization matrices $\mathbf{M_z}$ and $\mathbf{M}_\mathbf{w}^\top$, we obtain the *asymmetric uncross* case:

$$\begin{bmatrix} \mathbf{Z} \\ \mathbf{W} \end{bmatrix} \leftarrow \begin{bmatrix} 0 & \mathbf{S}_r\mathbf{X} \\ \mathbf{S}_c\mathbf{X}^\top & 0 \end{bmatrix} \begin{bmatrix} \widetilde{\mathbf{Z}}\mathbf{D}_{\mathbf{w}}^{-0.5} \\ \widetilde{\mathbf{W}}\mathbf{D}_{\mathbf{z}}^{-0.5} \end{bmatrix} = \begin{bmatrix} \mathbf{S}_r\mathbf{X}\widetilde{\mathbf{W}}\mathbf{D}_{\mathbf{z}}^{-0.5} \\ \mathbf{S}_c\mathbf{X}^\top\widetilde{\mathbf{Z}}\mathbf{D}_{\mathbf{w}}^{-0.5} \end{bmatrix}. \tag{11}$$

Similarly, if the identity matrix is assigned to the left-hand side of the $\mathbf{M}_\mathbf{z}$ and $\mathbf{M}_\mathbf{w}^\top$ regularization matrices, we obtain the *asymmetric cross* case:

$$\begin{bmatrix} \mathbf{Z} \\ \mathbf{W} \end{bmatrix} \leftarrow \begin{bmatrix} 0 & \mathbf{X}\mathbf{S}_c \\ \mathbf{X}^\top\mathbf{S}_r & 0 \end{bmatrix} \begin{bmatrix} \widetilde{\mathbf{Z}}\mathbf{D}_{\mathbf{w}}^{-0.5} \\ \widetilde{\mathbf{W}}\mathbf{D}_{\mathbf{z}}^{-0.5} \end{bmatrix} = \begin{bmatrix} \mathbf{X}\mathbf{S}_c\widetilde{\mathbf{W}}\mathbf{D}_{\mathbf{z}}^{-0.5} \\ \mathbf{X}^\top\mathbf{S}_r\widetilde{\mathbf{Z}}\mathbf{D}_{\mathbf{w}}^{-0.5} \end{bmatrix}. \tag{12}$$

This second particular case, $\texttt{RBDCo}_{[S_c,S_r]}$, usually produces the best performance with document-text data (see Section 6). Here, row/document clustering $\mathbf{Z}$ is regularized with the word co-occurrence information $\mathbf{S}_c$, and column/word clustering $\mathbf{W}$ is regularized with the document content similarities $\mathbf{S}_r$. This *cross* regularization is the most *natural* bi-directional regularization, reflecting the iterative alternating projections of words in the document space, and vice versa.

5.3 Connection to Matrix decomposition

5.3.1 Connection to NMF

Basically, $\widetilde{\mathbf{Z}} = \mathbf{Z}\mathbf{D}_\mathbf{z}^{-0.5}$ denotes the likelihood of documents being associated with document clusters, and $\widetilde{\mathbf{W}} = \mathbf{W}\mathbf{D}_\mathbf{w}^{-0.5}$ the likelihood of words being associated with word clusters. The ij^{th} entry of $\widetilde{\mathbf{Z}}\widetilde{\mathbf{W}}^\top$ therefore indicates the possibility that the j^{th} word will be present in the i-document, computed as the dot product of the i^{th} row of $\widetilde{\mathbf{Z}}$ and the j^{th} row of $\widetilde{\mathbf{W}}$. Hence, $\widetilde{\mathbf{Z}}\widetilde{\mathbf{W}}^\top$ can be interpreted as the approximation of the original data $\mathbf{X}$. Our goal is then to find a $\mathbf{Z}$ and a $\mathbf{W}$ that minimize the squared error between $\mathbf{X}$ and its approximation $\widetilde{\mathbf{Z}}\widetilde{\mathbf{W}}^\top$. From a *Nonnegative Matrix Factorization* (NMF) perspective (Lee and Seung, 2001) we have

$$\begin{aligned} \min_{\mathbf{Z},\mathbf{W}} ||\mathbf{X} - \mathbf{Z}\mathbf{D}_\mathbf{z}^{-0.5}\mathbf{D}_\mathbf{w}^{-0.5}\mathbf{W}^\top||_F^2 &\equiv \min_{\widetilde{\mathbf{Z}},\widetilde{\mathbf{W}}} ||\mathbf{X} - \widetilde{\mathbf{Z}}\widetilde{\mathbf{W}}^\top||_F^2 \\ &\equiv \max_{\widetilde{\mathbf{Z}},\widetilde{\mathbf{W}}} Tr(\widetilde{\mathbf{Z}}^\top\mathbf{X}\widetilde{\mathbf{W}}) \\ &\equiv \max_{\Theta,\mathbf{Z}} L_c(\Theta|\mathbf{X},\mathbf{Z}). \end{aligned}$$

It will be remarked that, by construction, both $\widetilde{\mathbf{Z}}$ and $\widetilde{\mathbf{W}}^\top$ are non-negative and orthogonal; we have $\widetilde{\mathbf{Z}}^\top\widetilde{\mathbf{Z}} = \mathbf{D}_\mathbf{z}^{-0.5}\mathbf{Z}^\top\mathbf{Z}\mathbf{D}_\mathbf{z}^{-0.5} = \mathbf{I}$, and similarly we also have $\widetilde{\mathbf{W}}^\top\widetilde{\mathbf{W}} = \mathbf{D}_\mathbf{w}^{-0.5}\mathbf{W}^\top\mathbf{W}\mathbf{D}_\mathbf{w}^{-0.5} = \mathbf{I}$. Our proposed generalized regularization framework RBDCo therefore allows us to see that in its basic configuration, vMF model-based co-clustering with a conscience mechanism is in fact equivalent to a double orthogonal NMF applied to spherical data.

5.3.2 Link to NMTF

In a similar way, we can identify the link to Non-negative Matrix Tri-Factorization. Let us consider the weighting matrix $\mathbf{D} = \mathbf{D}_\mathbf{z}^{-0.5}\mathbf{D}_\mathbf{w}^{-0.5}$, which is diagonal by construction and where each diagonal value $\mathbf{D}_{kk}$ represents the square root of the geometric mean of document and word cluster sizes in block k. It follows that $\min_{\mathbf{Z},\mathbf{W}} ||\mathbf{X} - \mathbf{Z}\mathbf{D}_\mathbf{z}^{-0.5}\mathbf{D}_\mathbf{w}^{-0.5}\mathbf{W}^\top||_F^2$ is equivalent to

$$\begin{aligned} \min_{\mathbf{Z},\mathbf{W},\mathbf{D}=\mathbf{D}_\mathbf{z}^{-0.5}\mathbf{D}_\mathbf{w}^{-0.5}} ||\mathbf{X} - \mathbf{Z}\mathbf{D}\mathbf{W}^\top||_F^2 &\equiv \max_{\mathbf{Z},\mathbf{W},\mathbf{D}} Tr(\mathbf{Z}^\top\mathbf{X}\mathbf{W}\mathbf{D}) \\ &\equiv \max_{\Theta,\mathbf{Z}} L_c(\Theta|\mathbf{X},\mathbf{Z}) \end{aligned}$$

which is also equivalent to Fast NMTF proposed in (Wang et al., 2011), with an additional constraint on the centroid matrix $\mathbf{D}$ in order to meet the requirement of directional data.

5.3.3 Link to spectral co-clustering

If, on the other hand, we relax the non-negativity constraint on both $\widetilde{\mathbf{Z}}$ and $\widetilde{\mathbf{W}}$, we have

$$\max_{\Theta,\mathbf{Z}} L_c(\Theta|\mathbf{X},\mathbf{Z}) \equiv \max_{\widetilde{\mathbf{Z}}^\top\widetilde{\mathbf{Z}}=\mathbf{I},\widetilde{\mathbf{W}}^\top\widetilde{\mathbf{W}}=\mathbf{I}} Tr(\widetilde{\mathbf{Z}}^\top\mathbf{X}\widetilde{\mathbf{W}}),$$

where $\widetilde{\mathbf{Z}} = \mathbf{Z}\mathbf{D}_\mathbf{z}^{-0.5}$ and $\widetilde{\mathbf{W}} = \mathbf{W}\mathbf{D}_\mathbf{w}^{-0.5}$. It is easy to verify that $\widetilde{\mathbf{Z}}$ and $\widetilde{\mathbf{W}}$ satisfy the orthogonality constraint, i.e. $\widetilde{\mathbf{Z}}^\top\widetilde{\mathbf{Z}} = \mathbf{I}$ and $\widetilde{\mathbf{W}}^\top\widetilde{\mathbf{W}} = \mathbf{I}$. This optimization problem can be transformed using Lagrange multipliers into an eigenvalue problem. Then, given $svd(\mathbf{X}) = \widetilde{\mathbf{Z}}\Sigma\widetilde{\mathbf{W}}^\top$, the discrete co-clustering is obtained by performing k-means on the concatenated data $[\mathbf{Z}\ \mathbf{W}]^\top$. This is equivalent to the spectral co-clustering method proposed in (Dhillon and Modha, 2001).

5.4 Link to Block Seriation

The basic idea of block co-clustering consists in modelling the simultaneous row and column partitions using a block seriation relation $\mathbf{Q}$ defined on $I \times J$ (where I is the set of objects and J the set of attributes). Given that $\mathbf{Q} = \mathbf{Z}\mathbf{W}^T$, the general term can be expressed as follows: $\mathbf{q}_{ij} = 1$ if object i is in the same block as attribute j, and $\mathbf{q}_{ij} = 0$ otherwise. Thus we have

$$\mathbf{q}_{ij} = \sum_{k=1}^{g} \mathbf{z}_{ik}\mathbf{w}_{jk} = (\mathbf{Z}\mathbf{W}^\top)_{ij}. \tag{13}$$

The matrix $\mathbf{Q}$ represents a block seriation relation (see (Marcotorchino, 1991) for further details) that must respect the following properties:

- **Binarity.** $\mathbf{q}_{ij} \in \{0,1\}, \forall (i,j) \in I \times J$.
- **Assignment constraints.** These constraints ensure the bijective correspondence between classes in two partitions, meaning that each class in the partition of I has one corresponding class in the partition of J, and vice versa. These constraints are expressed linearly as follows:

$$\begin{cases} \sum_{j \in J} \mathbf{q}_{ij} \geq 1 \ \forall i \in I \\ \sum_{i \in I} \mathbf{q}_{ij} \geq 1 \ \forall j \in J. \end{cases}$$

- **Triad impossible.** The role of these constraints is to ensure the disjoint structure of the blocks, which is expressed by the following system inequality:

$$\begin{cases} \mathbf{q}_{ij} + \mathbf{q}_{ij'} + \mathbf{q}_{i'j'} - \mathbf{q}_{i'j} - 1 \leq 1 \\ \mathbf{q}_{i'j'} + \mathbf{q}_{i'j} + \mathbf{q}_{ij} - \mathbf{q}_{ij'} - 1 \leq 1 \\ \mathbf{q}_{i'j} + \mathbf{q}_{ij} + \mathbf{q}_{ij'} - \mathbf{q}_{i'j'} - 1 \leq 1 \\ \mathbf{q}_{ij'} + \mathbf{q}_{i'j'} + \mathbf{q}_{i'j} - \mathbf{q}_{ij} - 1 \leq 1. \end{cases}$$

These constraints also generalize transitivity for non-symmetric data. In the case where $I = J$, it is easy to show that the block seriation relation $\mathbf{Q}$ becomes an equivalence relation, i.e. $\mathbf{Q} = \mathbf{Z}\mathbf{Z}^\top$ or $\mathbf{Q} = \mathbf{W}\mathbf{W}^\top$.

It will be remarked that (13) is not balanced in terms of the cluster sizes for rows and columns, meaning that a cluster might become small when affected by outliers. For this reason we propose a new scaled block seriation relation that considers both row and column cluster size:

$$\tilde{\mathbf{q}}_{ij} = \sum_{k=1}^{g} \frac{\mathbf{z}_{ik}\mathbf{w}_{jk}}{\sqrt{\mathbf{z}_{.k}\mathbf{w}_{.k}}} = \sum_{k=1}^{g} \tilde{\mathbf{z}}_{ik}\tilde{\mathbf{w}}_{jk} = (\widetilde{\mathbf{Z}}\widetilde{\mathbf{W}}^T)_{ij} \ . \tag{14}$$

A new measure, which we call *scaled block seriation criterion*, is defined as follows:

$$\begin{aligned} \min_{\mathbf{Z},\mathbf{W}} ||\mathbf{X} - \mathbf{Z}\mathbf{D}_{\mathbf{z}}^{-0.5}\mathbf{D}_{\mathbf{w}}^{-0.5}\mathbf{W}^\top||_F^2 &\equiv \min_{\widetilde{\mathbf{Z}},\widetilde{\mathbf{W}}} ||\mathbf{X} - \widetilde{\mathbf{Z}}\widetilde{\mathbf{W}}^\top||_F^2 \\ &\equiv \min_{\widetilde{\mathbf{Q}}} ||\mathbf{X} - \widetilde{\mathbf{Q}}||_F^2 \\ &\equiv \max_{\widetilde{\mathbf{Q}}} Tr(\mathbf{X}\widetilde{\mathbf{Q}}^\top) \\ &\equiv \max_{\Theta,\mathbf{Z}} L_c(\Theta|\mathbf{X},\mathbf{Z}). \end{aligned}$$

This is a scaled variant of the Block seriation method (Marcotorchino, 1991).

5.5 RBDCo regularization matrices

The regularization matrices $\mathbf{S}_c$ and $\mathbf{S}_r$ are built from the original document-term matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$. We first consider $\mathbf{S}_c$, for which we use a non-linear transformation of the word co-occurrences, namely the Pointwise Mutual Information (PMI). It must be emphasized that the PMI has been shown to be strongly correlated with human assessment for word relatedness (Newman et al., 2009; Role and Nadif, 2011). However, in other contexts, the user can easily introduce his own specific information about words/documents meaning. The PMI between words w_i and w_j is defined as $\log\left(p(w_i, w_j)/p(w_i)p(w_j)\right)$. Assuming that the documents are the context in which words co-occur, and using the matrix $\mathbf{C} = \mathbf{X}^\top\mathbf{X}$, we can empirically estimate the PMI as follows:

$$\mathrm{PMI}_{\mathrm{C}}(w_i, w_j) = \log \frac{c_{ij} \times c_{..}}{c_{j.}c_{.j}}, \tag{15}$$

where $c_{..} = \sum_{ij} c_{ij}$, $c_{i.} = \sum_j c_{ij}$ and $c_{.j} = \sum_i c_{ij}$. PMI values can be positive or negative. Positive values indicate that a word pair co-occurs more than by chance. Negative values are harder to interpret, since they would seem to indicate word pairs that co-occur *less* than by chance. A generally accepted approximation consists in replacing all negative values with 0, giving the Positive Pointwise Mutal Information (PPMI). One advantage of the PPMI is that it reduces the density of the PMI matrix. In `RBDCo`, we consider the $\mathrm{PPMI}_{\mathrm{c}}$ matrix as our word regularization matrix $\mathbf{S}_c$. Similarly, we can compute a matrix $\mathbf{R}$ such that $\mathbf{R} = \mathbf{X}\mathbf{X}^\top$. In virtue of (15) we can define a $\mathrm{PMI}_{\mathrm{r}}(d_i, d_j)$ between documents d_i and d_j that gives the co-occurrence frequency of two documents in the latent space of words. Just like in the case of $\mathbf{S}_c$, we consider the $\mathrm{PPMI}_{\mathrm{R}}$ as being $\mathbf{S}_r$.

We have chosen to make use of PPMI for `RBDCo` regularization matrices, since these matrices are very general and suitable for incorporating side similarity information. They can also be computed quite easily from the original data matrix. However, other document or word embeddings obtained via external methods might also be used (e.g., `Word2Vec` (Mikolov et al., 2013), `Doc2Vec` (Le and Mikolov, 2014)).

6 Experimental analysis

We now present our extensive experimental results that show the good performance of our method across a wide range of *real-world* text datasets. We first compare several variants of our `RBDCo` approach (Section 6.3.1). We then evaluate the best `RBDCo` variant in relation to baseline clustering and co-clustering methods (Section 6.3.2). Specifically, the clustering algorithms that we consider are k-`means`, *spectral clustering* (`Spec`), Non-Negative Matrix Factorization (`NMF`), and *spherical k-means* (`Skmeans`). It is generally recognized that `Skmeans` in particular is well-suited to high-dimensional sparse text data. The baseline co-clustering algorithms

are NMTF, DCC, and CoClustMod. The latter, CoClustMod, is a recent graph modularity-based co-clustering algorithm proposed by Ailem et al. (2016), in which CoClustMod was shown, through extensive experiments on text datasets, to outperform several other established co-clustering methods designed for the same task, including the well-known spectral co-clustering (Dhillon and Modha, 2001), and ITCC (Dhillon et al., 2003). Finally, we demonstrate the effectiveness of RBDCo in comparison with a competitive regularized text co-clustering method, WC-NMTF (Salah et al., 2018). Although WC-NMTF is a regularized co-clustering approach, it is based on an *additive unidirectional* regularization, where only word co-occurrence is used. By contrast, RBDCo proposes a *bi-directional multiplicative* regularization and embeds a *conscience* mechanism that makes it able to handle strongly unbalanced textual data. k-means, Spec and NMF come from the scikit-learn[4] Python package, and Skmeans and CoClustMod from the coclust[5] Python package. We implemented RBDCo, DCC and WC-NMTF in Python.

6.1 Benchmark Datasets

We analyzed 8 benchmark datasets widely used for document clustering purposes, namely SPORTS, TR45, LA12, CLASSIC4, CSTR, OHSCALE, PUBMED5, and CLASSIC3. Each dataset can be viewed as a contingency matrix where the coefficients x_{ij} indicate the number of occurrences of word j in document i. Together, these datasets contain a number of different challenging situations, including different degrees of cluster balance, diverse cluster sizes, and various degrees of cluster overlap.

Table 2: Description of Datasets

Datasets	Characteristics				
	#documents	#words	g	Sparsity (%)	Balance
SPORTS	8580	14870	7	99.14	0.036
TR45	690	8261	10	96.60	0.088
LA12	6279	31472	6	99.52	0.281
CLASSIC4	7094	5896	4	99.41	0.323
CSTR	475	1000	4	96.60	0.399
OHSCALE	11162	11465	10	99.47	0.437
PUBMED5	12648	19518	5	99.68	0.580
CLASSIC3	3891	4303	3	98.95	0.710

Table 2 provides an overview of the important characteristics of the datasets sorted in increasing order of their *Balance* coefficient, which is the ratio of the smallest cluster size to the largest cluster size. As is frequently the case in document-term co-clustering, within these benchmark datasets labels are known only for the documents, and not for the words. However, given that the word partition is inherently linked to the document partition, we would expect the quality of the document clustering to be informative about the quality of the word clustering.

6.2 Experimental settings and evaluation

In all our experiments the document-term count matrix is normalized using the TF-IDF weighting scheme (*term-frequency* times *inverse document frequency*), as implemented in the scikit-learn Python package. The results are averaged over 20 different runs. For RBDCo, each run is done with 10 different initializations and a number of iterations below 100. Specifically, the final RBDCo co-clustering is automatically obtained based on the best criterion (Eq. 7) among the different initializations. To avoid poor local solutions that could be produced by early hard word assignments in the RBDCo iteration, we perform stochastic column assignments during the first 70 iterations, as described in (Salah and Nadif, 2019). Whenever applicable, the approaches that RBDCo is being compared with were also performed with 10 initializations and not more than 100 iterations.

We evaluate the document clustering quality of RBDCo using two measures that are widely used for assessing the similarity between the estimated clustering and the true clustering. The measures are *Normalized Mutual Information* (NMI) (Strehl and Ghosh, 2003) and *Adjusted Rand Index* (ARI) (Hubert and Arabie, 1985; Steinley, 2004). Specifically, NMI evaluates to what extent the estimated clustering is informative about the known clustering, and ARI quantifies the agreement between the estimated clustering and the true labels. NMI is less sensitive than ARI to cluster splitting or merging.

6.3 Empirical results on document clustering

6.3.1 Comparing RBDCo *variants*

We first compare the four RBDCo versions that incorporate information on both the document and the word dimensions, namely $\texttt{RBDCo}_{[S_c,S_r]}$, $\texttt{RBDCo}_{[S_r,S_c]}$, $\texttt{RBDCo}_{[S_c,S_c]}$ and $\texttt{RBDCo}_{[S_r,S_r]}$ (see Table 1 for details on RBDCo schemes). Table 3 summarizes the NMI and ARI evaluations for these versions on all the benchmark datasets.

These four RBDCo schemes give good NMI and ARI results, with a null standard deviation for almost all

[4] https://scikit-learn.org/stable/
[5] https://pypi.python.org/pypi/coclust

Table 3: Mean±sd clustering NMI and ARI on documents × terms matrices for RBDCo variants. Bold values indicate the best result over all methods.

Datasets		$\texttt{RBDCo}_{[S_c,S_r]}$	$\texttt{RBDCo}_{[S_r,S_c]}$	$\texttt{RBDCo}_{[S_c,S_c]}$	$\texttt{RBDCo}_{[S_r,S_r]}$
SPORTS	NMI	0.67 ± 0.01	**0.71 ± 0.00**	0.64 ± 0.00	0.70 ± 0.00
	ARI	0.57 ± 0.03	**0.68 ± 0.00**	0.57 ± 0.00	0.65 ± 0.00
TR45	NMI	**0.76 ± 0.00**	0.74 ± 0.00	**0.76 ± 0.00**	0.74 ± 0.00
	ARI	**0.68 ± 0.00**	0.67 ± 0.00	**0.68 ± 0.00**	0.68 ± 0.00
LA12	NMI	**0.58 ± 0.00**	0.55 ± 0.00	0.56 ± 0.03	0.57 ± 0.01
	ARI	**0.56 ± 0.00**	0.53 ± 0.00	0.54 ± 0.03	0.53 ± 0.01
CLASSIC4	NMI	**0.77 ± 0.00**	0.75 ± 0.00	0.76 ± 0.00	0.76 ± 0.00
	ARI	**0.78 ± 0.00**	0.76 ± 0.00	0.77 ± 0.00	0.78 ± 0.00
CSTR	NMI	**0.78 ± 0.00**	0.73 ± 0.00	0.77 ± 0.00	0.73 ± 0.00
	ARI	**0.82 ± 0.00**	0.77 ± 0.00	0.81 ± 0.00	0.78 ± 0.00
OHSCALE	NMI	0.44 ± 0.00	0.44 ± 0.01	0.44 ± 0.00	**0.45 ± 0.00**
	ARI	0.35 ± 0.00	0.34 ± 0.01	0.34 ± 0.00	**0.36 ± 0.00**
PUBMED5	NMI	**0.91 ± 0.00**	0.90 ± 0.00	**0.91 ± 0.00**	**0.91 ± 0.00**
	ARI	**0.94 ± 0.00**	0.94 ± 0.00	**0.94 ± 0.00**	**0.94 ± 0.00**
CLASSIC3	NMI	**0.95 ± 0.00**	0.93 ± 0.00	**0.95 ± 0.00**	0.93 ± 0.00
	ARI	**0.97 ± 0.00**	0.96 ± 0.00	**0.97 ± 0.00**	0.96 ± 0.00

datasets. Overall, $\texttt{RBDCo}_{[S_c,S_r]}$ is the most effective variant. As already mentioned (Section 5.2), this version has the most *natural* bi-directional regularization for co-clustering. SPORTS is an exception: for this dataset it is the *uncross* bi-directional regularization $\texttt{RBDCo}_{[\mathbf{S}_r,\mathbf{S}_c]}$ that provides the best results (Table 3, first row). This may be explained by its high degree of document cluster imbalance. Among all the datasets, SPORTS has the lowest balance coefficient (0.036) and the lowest *ratio of minimum to expected* (RME = 0.099) – this corresponds to the smallest cluster size with respect to the expected cluster size, that is to say n/g, where g is the number of clusters. SPORTS also has the greatest *standard deviation in cluster sizes* (SDCS = 1253.01), which is defined as $\{1/(g-1)\sum_{k=1}^{g}(n_k - n/g)^2\}^{0.5}$, where n is the total number of documents and n_k is the cardinality of the k^{th} cluster. This dataset therefore requires that the document similarity $\mathbf{S}_r$ be applied on the document dimension rather than on the word dimension in order to achieve a significantly higher NMI (0.71) and ARI (0.68).

Below, in the light of these results, we will consider only the *cross* regularized RBDCo scheme $\texttt{RBDCo}_{[S_c,S_r]}$, where the document clustering $\mathbf{Z}$ is regularized with word co-occurrence information $\mathbf{S}_c$, and the word clustering $\mathbf{W}$ is regularized with document content similarity $\mathbf{S}_r$.

6.3.2 Evaluating RBDCo *against baselines*

Table 4 is a synopsis of our results for RBDCo and the other clustering/co-clustering methods, comparing their NMI and ARI values across the various benchmark datasets. The first thing to notice is that RBDCo clearly outperforms the standard or competitive co-clustering methods shown in the rightmost three columns of Table 4 (see also *Average ranks*). In contrast with DCC, CoClustMod and NMF, RBDCo uses two regularization terms, specifically a word correlation matrix $\mathbf{S}_c$ and a document similarity matrix $\mathbf{S}_r$. The superior performance of RBDCo can therefore be attributed to these regularization terms. The ARI metric is generally more sensitive than NMI to cluster merging or splitting. However, RBDCo has good performances for both NMI and ARI, even for highly unbalanced datasets (such as TR45 and SPORTS).

As expected, our co-clustering approach generally outperforms baseline clustering methods for the document clustering, with a higher mean margin for NMF, k-means and Spec than for the co-clustering approaches. It will be remarked that Skmeans performs well on two benchmark datasets, PubMed5 and ClASSIC3, with a small mean margin of 0.02 for NMI and 0.01 for ARI. However, our approach significantly outperforms Skmeans on unbalanced text datasets (from SPORTS to OHSCALE), with a mean margin of 0.06 for NMI and 0.11 for ARI (Table 4, first two columns). These good results on text datasets against a strong competitor like Skmeans are an indication of how well RBDCo is able to deal with a wide range of textual datasets, and in particular in relation to text data with very small or very large (co-)clusters (see Table 4, *Average ranks*). It should also be remembered that RBDCo provides a simultaneous clustering of documents *and* words, and hence ensures the identification of the main document cluster topics, while Skmeans gives only a one-sided clustering without automatic association between documents and words.

6.3.3 Bi-directional word-based regularization

We will now compare RBDCo with a competitive regularized method, namely WC-NMTF. WC-NMTF uses an additive *unidirectional* regularization, where only word co-occurrence is used on the partition of columns to obtain block diagonal co-clusters. A comparison of $\texttt{RBDCo}_{[\mathbf{S}_c,\mathbf{S}_c]}$ with WC-NMTF allows us to evaluate the advantage to be derived from mutiplicative *bi-directional* word similarity regularization (i.e., $\mathbf{S}_c$ applied on the columns *and* the rows of $\mathbf{X}$). To further enrich our comparison, we also consider the DCC evaluations. As detailed in Section 3, DCC, just like RBDCo, incorporates a conscience mechanism, but it does not use any regularization. Figure 2 gives the NMI and ARI measures for $\texttt{RBDCo}_{[S_c,S_c]}$, WC-NMTF, and DCC, on all datasets. These results clearly show the better performance of RBDCo. In particular, RBDCo improves the PUBMED5 document partitioning almost by a factor of two in relation to WC-NMTF, although

Table 4: RBDCo baseline Comparisons with clustering and co-clustering approaches. Mean±sd clustering NMI and ARI on documents × terms matrices. Bold values indicate the best result over all methods.

Datasets		RBDCo$_{[S_c,S_r]}$	Skmeans	NMF	k-means	Spec	DCC	CoClustMod	NMTF
SPORTS	NMI	**0.67 ± 0.01**	0.64 ± 0.02	0.53 ± 0.02	0.44 ± 0.03	0.42 ± 0.00	0.57 ± 0.01	0.55 ± 0.05	0.52 ± 0.03
	ARI	**0.57 ± 0.03**	0.47 ± 0.03	0.28 ± 0.03	0.14 ± 0.04	0.11 ± 0.00	0.39 ± 0.01	0.49 ± 0.06	0.28 ± 0.03
TR45	NMI	**0.76 ± 0.00**	0.65 ± 0.04	0.58 ± 0.05	0.63 ± 0.04	0.52 ± 0.00	0.69 ± 0.02	0.51 ± 0.04	0.63 ± 0.03
	ARI	**0.68 ± 0.00**	0.58 ± 0.06	0.44 ± 0.08	0.49 ± 0.09	0.32 ± 0.01	0.56 ± 0.04	0.45 ± 0.04	0.53 ± 0.04
LA12	NMI	**0.58 ± 0.00**	0.55 ± 0.02	0.43 ± 0.02	0.42 ± 0.04	0.35 ± 0.01	0.52 ± 0.02	0.43 ± 0.02	0.41 ± 0.02
	ARI	**0.56 ± 0.00**	0.50 ± 0.04	0.39 ± 0.04	0.26 ± 0.07	0.17 ± 0.02	0.45 ± 0.03	0.41 ± 0.03	0.35 ± 0.04
CLASSIC4	NMI	**0.77 ± 0.00**	0.69 ± 0.01	0.53 ± 0.01	0.54 ± 0.00	0.48 ± 0.00	0.72 ± 0.00	0.67 ± 0.01	0.55 ± 0.03
	ARI	**0.78 ± 0.00**	0.49 ± 0.00	0.44 ± 0.01	0.37 ± 0.01	0.29 ± 0.00	0.71 ± 0.01	0.61 ± 0.03	0.44 ± 0.01
CSTR	NMI	**0.78 ± 0.00**	0.68 ± 0.03	0.64 ± 0.01	0.62 ± 0.03	0.52 ± 0.00	0.68 ± 0.01	0.67 ± 0.04	0.67 ± 0.03
	ARI	**0.82 ± 0.00**	0.70 ± 0.05	0.57 ± 0.04	0.53 ± 0.06	0.35 ± 0.00	0.61 ± 0.05	0.71 ± 0.05	0.63 ± 0.09
OHSCALE	NMI	**0.44 ± 0.00**	0.42 ± 0.02	0.39 ± 0.02	0.37 ± 0.01	0.33 ± 0.00	0.37 ± 0.02	0.32 ± 0.02	0.40 ± 0.03
	ARI	**0.35 ± 0.00**	0.34 ± 0.02	0.30 ± 0.03	0.23 ± 0.03	0.22 ± 0.00	0.28 ± 0.01	0.23 ± 0.02	0.32 ± 0.03
PUBMED5	NMI	0.91 ± 0.00	**0.94 ± 0.02**	0.86 ± 0.05	0.74 ± 0.09	0.68 ± 0.00	0.58 ± 0.05	0.48 ± 0.07	0.81 ± 0.09
	ARI	0.94 ± 0.00	**0.96 ± 0.02**	0.88 ± 0.08	0.63 ± 0.16	0.53 ± 0.00	0.51 ± 0.05	0.45 ± 0.09	0.79 ± 0.13
CLASSIC3	NMI	0.95 ± 0.00	**0.96 ± 0.00**	0.75 ± 0.20	0.90 ± 0.01	0.62 ± 0.00	0.94 ± 0.00	0.94 ± 0.00	0.91 ± 0.00
	ARI	0.97 ± 0.00	**0.98 ± 0.00**	0.75 ± 0.24	0.94 ± 0.00	0.54 ± 0.00	0.97 ± 0.00	0.97 ± 0.00	0.95 ± 0.00
Mean difference	NMI	*reference*	0.04	0.14	0.15	0.24	0.10	0.16	0.12
	ARI	*reference*	0.08	0.20	0.26	0.39	0.15	0.17	0.17
Average ranks	NMI	**1.25**	2	5.25	5.75	7.5	3.37	5.37	4.75
	ARI	**1.25**	2.25	5.25	6.25	7.75	3.87	4.12	4.5

the bi-directional regularization uses only word similarity information.

6.4 Empirical results on word clustering

6.4.1 Block diagonal co-clustering

RBDCo partitions the data in diagonal document-term co-clusters, resulting in a document *and* a word clustering. Figure 3b & 3d show the structures revealed by RBDCo$_{[S_c,S_r]}$ for PUBMED5 and CLASSIC4 (dots indicate strong TF-IDF weights). While the original PUBMED5 matrix does not have any explicit structure (Figures 3a), RBDCo proposes a very clear co-clustering of the documents and terms (Figure 3b; NMI = 0.91, ARI = 0.94). Figure 3d shows the structure uncovered by RBDCo for CLASSIC4 (NMI = 0.77, ARI = 0.78).

The words that occur most frequently within a co-cluster C_i are usually considered to be the most representative terms for that co-cluster. The ranking of these top terms obtained with RBDCo is given in Table 5 for PUBMED5 and CLASSIC4. We can see that these terms provide a good interpretability of the partitioning. Most importantly, they can easily be linked to the topics of the true document classes. PUBMED5 is composed of five document classes, namely Age-related Macular Degeneration (AMD), Otitis, Kidney Stones, Hay Fever and Migraine. RBDCo clearly uncovers associated topic words. Similarly, for CLASSIC4 RBDCo gives top terms that are highly indicative of the true document classes, namely CISI (information retrieval), CACM (computing machinery), MEDLINE (medical) and CRANFIELD (aeronautical systems). We recall that PUBMED5 and CLASSIC4 contain stemmed terms, so for example we have 'ey' rather than 'eye', and 'studi' rather than 'study'.

6.4.2 Quantifying the quality of word clusters

Assessing the quality of word clustering is challenging, since the benchmark datasets commonly used in text document co-clustering provide the true document labels only. To assess the quality of the incorporation of word similarity information on the word partitioning, we first focus on the PPMI score. We then propose an enhanced version of the NPMI (Normalized Pointwise Mutual Information) score, to quantify the quality of the word clustering.

PPMI-score assessment. We first consider the ten most frequent terms in each word cluster as the top terms. Table 5 gives the average pairwise PPMI values for the ten most frequent terms *within* and *between* word clusters. A *random* average is also given as a reference. This is an average pairwise PPMI over 100 random groups of ten words from the 1000 most frequent words within the whole corpus. Interestingly, for both the PUBMED5 and the CLASSIC4 datasets, the *within* PPMI average is much higher than the *between* PPMI average, indicating that RBDCo makes effective use of the PPMI

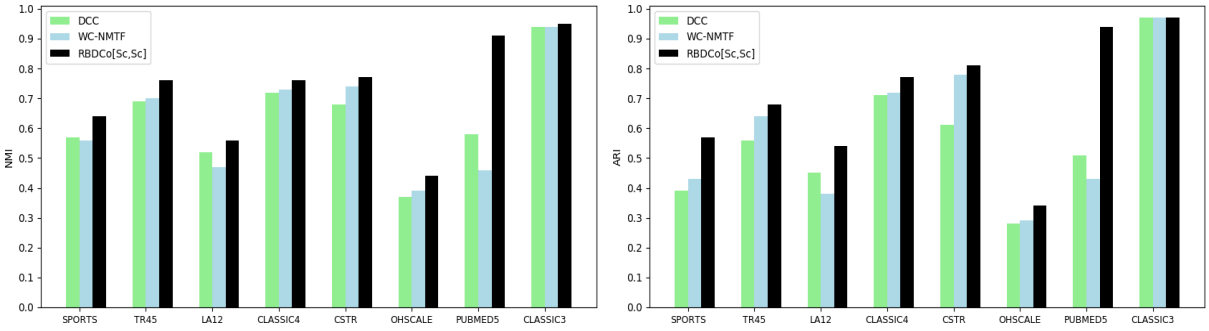
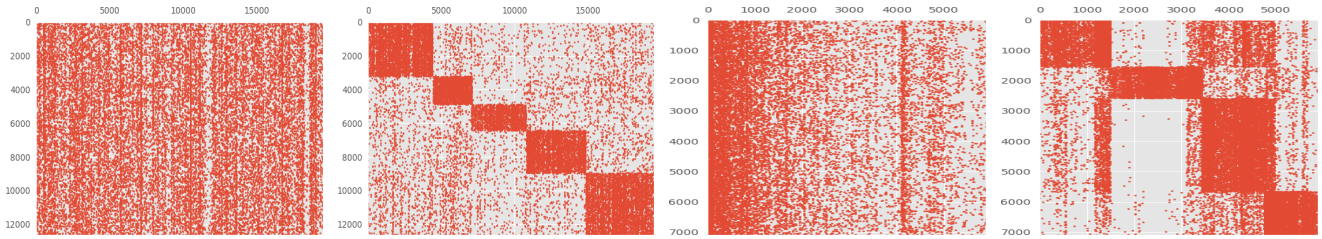


Fig. 2: NMI and ARI comparison of $\texttt{RBDCo}_{[S_c,S_c]}$, `WC-NMTF` and `DCC` on documents × terms matrices.



(a) PUBMED5 original (b) PUBMED5 $\texttt{RBDCo}_{[S_c,S_r]}$ (c) CLASSIC4 original (d) CLASSIC4 $\texttt{RBDCo}_{[S_c,S_r]}$

Fig. 3: Original datasets (a & c) and reorganized version (b & d) using $\texttt{RBDCo}_{[Sc,Sr]}$.

Table 5: Top frequent terms with $\texttt{RBDCo}_{[S_c,S_r]}$ and average PPMI *within* (w/i) and *between* (b/w) co-clusters.

		PUBMED5				
		AMD (C1)	**Otitis** (C2)	**Kidney Stones** (C3)	**Hay Fever** (C4)	**Migraine** (C5)
		wa	otiti	stone	allerg	migrain
		macular	ear	renal	nasal	patient
		ey	children	calcium	rhiniti	headach
		visual	media	urinari	symptom	studi
		amd	middl	kidnei	pollen	thi
		retin	acut	oxal	season	treatment
		acuiti	antibiot	rate	effect	ar
		associ	aom	urin	allergen	pain
		degen	tube	percutan	asthma	clinic
		group	om	calculi	increa	compar
PPMI	w/i	0.68	1.37	1.67	1.29	0.25
	b/w	0.04	0.02	0.03	0.07	0.08
	random			0.19		

		CLASSIC4			
		CISI (C1)	**CACM** (C2)	**MEDLINE** (C3)	**CRANFIELD** (C4)
		librari	algorithm	cell	flow
		inform	system	patient	pressure
		scienc	program	increas	number
		research	comput	normal	boundari
		studi	method	growth	layer
		index	languag	rat	effect
		develop	gener	group	result
		book	data	blood	heat
		servic	problem	treatment	wing
		retriev	time	tissu	bodi
PPMI	w/i	0.69	0.42	1.10	0.84
	b/w	0.08	0.05	0.06	0.04
	random			0.20	

Table 6: NPMI_i scores *within* (w/i) and *between* (b/w) top frequent word clusters.

AMD	w/i	b/w	**Otitis**	w/i	b/w	**Kidney Stones**	w/i	b/w	**Hay Fever**	w/i	b/w	**Migraine**	w/i	b/w
macular	0.67	0.07	otiti	0.48	0.06	urinari	0.55	0.24	allerg	0.55	0.22	migrain	0.54	0.16
degen	0.59	0.15	**infect**	0.44	0.29	**excret**	0.52	0.11	rhiniti	0.55	0.06	headach	0.40	0.24
retin	0.50	0.19	**pneumonia**	0.44	0.17	**uret**	0.50	-0.01	**allergi**	0.54	0.20	**triptan**	0.39	0.00
edema	0.46	0.27	**bacteri**	0.41	0.19	urin	0.50	0.20	asthma	0.51	0.22	treatment	0.33	0.27
diabet	0.42	0.23	acut	0.39	0.28	kidnei	0.50	0.24	allergen	0.51	0.09	pain	0.32	0.24
acuiti	0.41	0.08	**chronic**	0.39	0.32	renal	0.49	0.26	**immunotherapi**	0.40	0.08	patient	0.32	0.29
visual	0.36	0.09	antibiot	0.38	0.22	**uric**	0.48	0.12	pollen	0.38	0.06	**efficaci**	0.32	0.21
amd	0.32	0.02	**recurr**	0.34	0.18	oxal	0.44	0.02	nasal	0.37	0.16	clinic	0.31	0.23
inject	0.32	0.19	**effu**	0.34	0.08	**acid**	0.41	0.14	symptom	0.34	0.30	**drug**	0.29	0.22
ey	0.29	0.12	**complic**	0.33	0.24	**metabol**	0.40	0.16	**eosinophil**	0.32	0.03	**sever**	0.25	0.22

regularization information given as input throughout its alternating iterations in a way that ultimately favors the grouping of semantically related words. The average pairwise PPMI between RBDCo word clusters even exhibits a stronger antagonism than the average pairwise PPMI for random groups of words ($\text{PPMI}_{random} \sim 0.2$). Word clusters with less specific vocabulary have a lower PPMI average (e.g., **Migraine** from PUBMED5: Table 5). Although informative, these average PPMI evaluations on most frequent words cannot properly assess the quality of the word clusters. One drawback of the PPMI value is that it is unbounded. Furthermore, the frequency of the words might not be the most appropriate ranking score for identifying representative words. As an example, the terms *wa* and *group* can be found among the most frequent terms in the **AMD** word cluster for PUBMED5. In addition, word clusters might also contain different sub-topics that are indirectly related to each other. Therefore, considering the average PPMI over *all* pairs of words has the effect of lowering the global score, and leads to the spurious conclusion that the word cluster is not coherent.

NPMI-score assessment. We propose evaluating the word cluster coherence using the Normalized PMI (NPMI) via a k-nn-like (k nearest neighbors) approach. The NPMI ranges between -1 and $+1$, and is formally defined as $\text{NPMI}(w_i, w_j) = \text{PMI}(\text{w}_\text{i}, \text{w}_\text{j}) / \log(p(w_i, w_j))$. For each word we propose computing an NPMI_i score defined as

$$\text{NPMI}_i = \frac{1}{|\Omega_i|} \sum_{w_j \in \Omega_i} \text{NPMI}(w_i, w_j) \qquad (16)$$

where Ω_i is the set of k words w_j having the highest NPMI score with w_i. NPMI_i quantifies the degree to which a word belongs to a cluster, based on its relationships with its k closest NPMI neighbors. NPMI_i scores can be computed within or between word clusters. Table 6 gives the top NPMI_i words for PUBMED5 word clusters ($k = 5$ neighbors among the 30 most frequent terms). Probabilities are derived from the whole English Wikipedia, using a NPMI implementation proposed by Röder et al. (2015). The NPMI_i scores are therefore independent of the input document-term data and regularization matrices. The top NPMI_i terms contain new meaningful words rather than the most frequent terms. It can be seen that *triptan*, which is a drug specific to migraine, is in third position in the **Migraine** cluster. The top **Hay Fever** terms now include *immunotherapy*, a seasonal allergy treatment, and *oesinophil*, a marker in seasonal allergic rhinitis. The word *pneumonia* can be found in the top **Otitis** terms, reflecting the fact that *Streptococcus pneumoniae* is the most common microbial agent found in otitis. Finally, the top **AMD** terms include *diabet*, an AMD risk factor, *edema*, a symptom of macular degeneration, and *inject*, which that corresponds to *intravitreal injection*, a treatment for AMD (*intravitr* is found in 14^{th} position with $\text{NPMI}_i = 0.21$).

Figure 4 gives more insight into the relationships between the top NPMI_i words in the case of **AMD**. The color of the vertices reflects the NPMI_i word score, with warmer colors corresponding to higher scores, and the thickness of the edges represents the strength of the pairwise NPMI coefficient. The most important word in terms of NPMI_i is *macular*. Directly related to this word are words that generally define the disease, namely *amd* and *degen* for *Age-Related Macular Degeneration*, and *neovascular* for *advanced neovascular AMD*, which is a serious type of AMD. In the upper part of the graph we have eye-related vocabulary (e.g., *acuit(i)y*, *visual*). On the right we see the words *risk*, *factor*, and *associ*, all of which are linked to *diabet*, an AMD risk factor. The *intravitr inject* bigram is directly related to *edema* and *diabet*. This makes sense, given that intravitreal injection is a treatment for diabetic macular edema. All in all, our results based on the NPMI_i score and NPMI coefficient indicate that the word clusters obtained with RBDCo are highly coherent (see Section 6.4.3 for the remaining word clusters on PUBMED5).

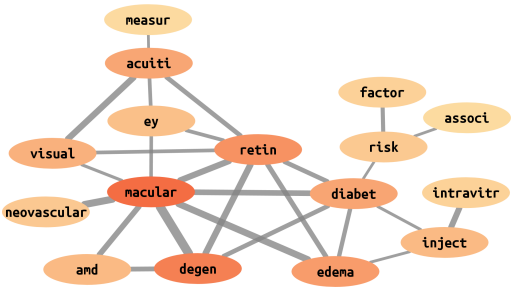


Fig. 4: NPMI graph of RBDCo **AMD** word cluster on PUBMED5.

Figure 5 shows the average of the top ten NPMI_i terms *within* and *between* PUBMED5 word clusters obtained using RBDCo. The *within* average NPMI_i score is seen to be higher than the *between* average NPMI_i score. The cluster with the strongest NPMI_i with other clusters is the **Migraine** word cluster, for which the top terms are related to the topic but not specific to it.

6.4.3 RBDCo *word cluster graphs on* PUBMED5

We now discuss the coherence of the word cluster graphs obtained with $\text{RBDCo}_{[S_c, S_r]}$ on PUBMED5. The graphs are constructed based on the pairwise NPMI coefficient

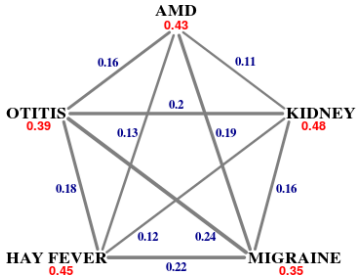


Fig. 5: Mean top NPMI_i for RBDCo clusters on PUBMED5.

between the terms that have the highest individual NPMI_i score (Eq. 16, main text). The color of the vertices reflects the NPMI_i word score, with warmer colors corresponding to higher scores, and the thickness of the edges represents the strength of the pairwise NPMI value.

Otitis (Figure 6). We observe that *otiti*s is logically related to *ear* and *media*, with *otitis media* (OM) being a group of inflammatory diseases of the middle ear. The two main types are *acute otitis media* and *otitis media with effusion*. Our graph relates *acut*e and *effu*sion to *otiti*s. Other words generally used to characterize otitis can also be found in the graph, such as *recurr*ent and *chronic*. Furthermore, *S.pneumoniae* and *H.influenzae* are the most common causes of OM. *S.pneumoniae* is also the main cause of recurrent infections and postinfectious *complic*ations. This is fully coherent with the proposed graphical representation.

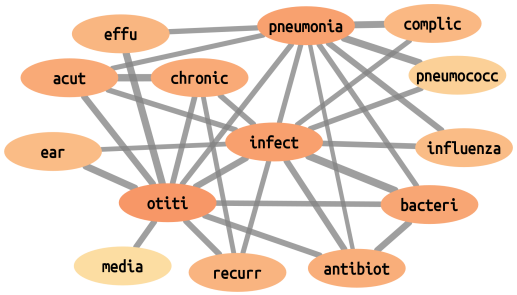


Fig. 6: NPMI graph of RBDCo **Otitis** word cluster on PUBMED5.

Migraine (Figure 7). The most important word in this graph with respect to its NPMI_i score is *migrain*e, which is strongly related to *headach*e and *aura*. Aura is a neurological phenomenon that can accompany migraine, manifesting itself in the form of visual, sensory, and motor disturbances. As for *triptan*, this refers to a family of drugs that have been *clinic*ally *assess*ed as being effective in the treatment of pain. We can find all these terms related in our graph, in addition to *sumatriptan*, a common migraine medication. General headache qualifiers (e.g. *sever*e, *pain*) are also present.

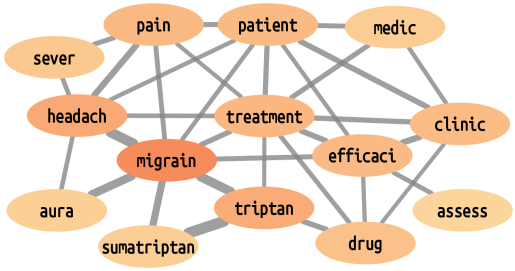


Fig. 7: NPMI graph of RBDCo **Migraine** word cluster on PUBMED5.

Kidney Stones (Figure 8). The *kidnei*(y) stones or *renal calculi* graph is enriched with urinary-tract-related vocabulary, such as *urete*r, *urine*, *uric* and *urinari*(y). The graph also contains common bigrams including *urinari tract* and *uric acid*. The kidney has a clearance function that requires the excretion (*excret*) of certain metabolites (*metabol*) by our organism. In addition, the definition of kidney stones is to be seen (on the right): they are solid masses made of *crystal*s that form *calcium oxal*ate stones. Finally, we note the presence of *shock wave lithotripsi*(y), the most common treatment for kidney stones.

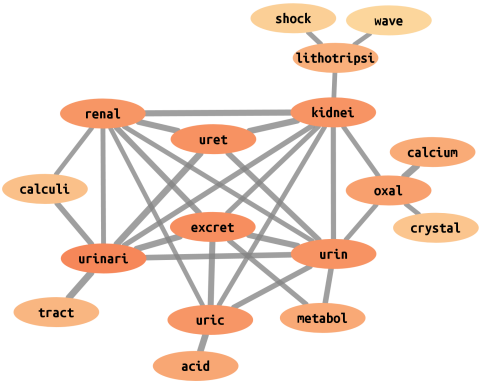


Fig. 8: NPMI graph of RBDCo **Kidney Stones** word cluster on PUBMED5.

Hay fever (Figure 9). Hay fever, also known as *allergic rhinitis*, is a *season*al *allergi*(y) caused in large part by *pollen*. These terms are related in the graph, with *pollen* linked to *allergen*. The words *intranas*al and *nasal* refer to common hay fever medications (e.g., corticosteroid nasal spray, intranasal antihistamine). The term *immunotherapi*(y), also to be seen in the graph,

is a treatment for hay fever involving a desensitization through doses of certain allergens (e.g., grass and pollen). Our graphical representation also contains *oesinophil* and *cell*, which are linked. In fact, *oesinophil* is a specialized *cell* within the immune system that helps promote inflammation and plays a key role in the symptoms of asthma and allergies. Interestingly, the terms *allergi*, *rhiniti* and *asthma* are linked to *symptom*. In fact, the asthma and allergic rhinitis symptoms are so close that people with asthma may not recognize that they also have allergic rhinitis.

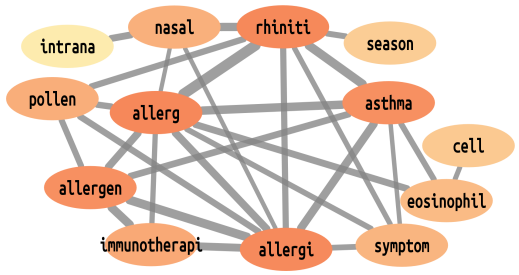


Fig. 9: NPMI graph of RBDCo **Hay Fever** word cluster on PUBMED5.

7 Conclusion

We proposed a flexible general framework, RBDCo, for text data matrix co-clustering. RBDCo derives from a von Mises-Fisher model-based co-clustering suitable for data that is high-dimensional, sparse, and unbalanced. Specifically, we defined our model with a matrix formulation suitable for the incorporation of complementary information to improve the co-clustering. Under some constraints, this formulation bears a close relationship to the well-known Spherical k-means, NMF and NMTF. Our approach utilizes the directional nature of text data and outperforms existing methods by using bi-directional multiplicative regularizations to incorporate side information on the document and word dimensions. Our experiments demonstrate the good performance of RBDCo, its robustness despite its random initialization, and its capabilities in terms of co-clustering quality and interpretability. Although all versions of RBDCo give good results, our results suggest that RBDCo$_{[\mathbf{S}_c,\mathbf{S}_r]}$ should be preferred. The proposed bi-directional regularization may also be seen as a means of performing a semi-supervised co-clustering. The regularization matrices might contain side expert information on the data to be partitioned, thus allowing a co-clustering that does not only depend on the input data.

In our study we assumed that the number of co-clusters is known. Often, in practice, the number of clusters is not known and needs to be determined by the user. Assessing the number of clusters is, however, not straightforward, and remains one of the biggest challenges in co-clustering. Unfortunately, in our approach we cannot rely on the well-established statistical theory of model selection, since our algorithm is not based on the maximization of the likelihood or, more precisely, on the complete-data likelihood. However, based on the vMF-Fisher mixture model designed for co-clustering, Salah and Nadif (2019) showed that (AIC) (Akaike, 1998) and AIC3 (Bozdogan, 2000) are effective. They also showed that these criteria give better results than the versions of the Bayesian information criterion (BIC) (Schwarz, 1978) and the integrated classification likelihood (ICL) (Keribin et al., 2015) derived from *latent block models* (Govaert and Nadif, 2003, 2005). As Salah and Nadif pointed out in their paper, this is due to the effective number of free parameters in the vMF-Fisher mixture model. Inspired by this result, our objective is now to address this issue in future work.

To go further, in the future we are planning to improve upon our proposed method in measuring the impact of each matrix $\mathbf{S}_r$ and $\mathbf{S}_c$ in the construction of the objective function by considering two different weights for both matrices.

Conflict of interest

The authors declare that they have no conflict of interest.

Acknowledgments

This work was supported by a grant overseen by the French National Research Agency (ANR) (ANR-19-CE23-0002). It also received the labelling of *Cap Digital* and *EuroBiomed/Cancer-Bio-Santé* competitiveness clusters.

References

Ahalt, S. C., Krishnamurthy, A. K., Chen, P., and Melton, D. E. (1990). Competitive learning algorithms for vector quantization. *Neural networks*, 3(3):277–290.

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

Ailem, M., Role, F., and Nadif, M. (2017a). Model-based co-clustering for the effective handling of sparse data. *Pattern Recognition*, pages 108–122.

Ailem, M., Salah, A., and Nadif, M. (2017b). Non-negative matrix factorization meets word embedding. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1081–1084.

Akaike, H. (1998). Information theory and an extension of the maximum likelihood principle. In *Selected Papers of Hirotugu Akaike*, pages 199–213. Springer New York.

Banerjee, A., Dhillon, I. S., Ghosh, J., and Sra, S. (2005). Clustering on the unit hypersphere using von mises-fisher distributions. *J. Mach. Learn. Res.*, 6:1345–1382.

Banerjee, A. and Ghosh, J. (2004). Frequency-sensitive competitive learning for scalable balanced clustering on high-dimensional hyperspheres. *IEEE Transactions on Neural Networks*, 15(3):702–719.

Bock, H.-H. (1979). Simultaneous clustering of objects and variables. In Tomassone, R., editor, *Analyse des Données et Informatique*, pages 187–203, Le Chesnay, France. INRIA.

Bock, H.-H. (2020). Co-clustering for object by variable data matrices. In *Advanced Studies in Behaviormetrics and Data Science*, pages 3–17. Springer.

Bozdogan, H. (2000). Akaike's information criterion and recent developments in information complexity. *Journal of Mathematical Psychology*, 44(1):62–91.

Cho, H. and Dhillon, I. S. (2008). Coclustering of human cancer microarrays using minimum sum-squared residue coclustering. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 5(3):385–400.

Deodhar, M. and Ghosh, J. (2010). Scoal: A framework for simultaneous co-clustering and learning from complex data. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 4(3):1–31.

DeSieno, D. (1988). Adding a conscience to competitive learning. In *IEEE international conference on neural networks*, volume 1, pages 117–124, San Diego, CA, USA. Institute of Electrical and Electronics Engineers New York, IEEE.

Dhillon, I. S., Mallela, S., and Modha, D. S. (2003). Information-theoretic co-clustering. In *the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 89–98. ACM.

Dhillon, I. S. and Modha, D. S. (2001). Concept decompositions for large sparse text data using clustering. *Mach. Learn.*, 42(1-2):143–175.

Gopal, S. and Yang, Y. (2014). Von Mises-Fisher clustering models. In *ICML*, pages 154–162, Beijing, China.

Govaert, G. (1983). *Classification croisée*. Thèse d'état, Université Paris 6, France.

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

Govaert, G. and Nadif, M. (2005). An EM algorithm for the block mixture model. *IEEE Transactions on Pattern Analysis and machine intelligence*, 27(4):643–647.

Govaert, G. and Nadif, M. (2008). Block clustering with bernoulli mixture models: Comparison of different approaches. *Computational Statistics & Data Analysis*, 52(6):3233–3245.

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

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

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

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

Hofmann, T. and Puzicha, J. (1999). Latent class models for collaborative filtering. In *IJCAI*, volume 99, pages 688–693, Stockholm, Sweden. Morgan Kaufmann.

Hubert, L. and Arabie, P. (1985). Comparing partitions. *Journal of classification*, 2(1):193–218.

Keribin, C., Brault, V., Celeux, G., and Govaert, G. (2015). Estimation and selection for the latent block model on categorical data. *Statistics and Computing*, 25(6):1201–1216.

Le, Q. and Mikolov, T. (2014). Distributed representations of sentences and documents. In *International conference on machine learning*, pages 1188–1196.

Lee, D. D. and Seung, H. S. (2001). Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems*, pages 556–562.

Madeira, S. C. and Oliveira, A. L. (2004). Biclustering algorithms for biological data analysis: a survey. *IEEE/ACM transactions on computational biology and bioinformatics*, 1(1):24–45.

Marcotorchino, F. (1991). Seriation problems: an overview. *Applied stochastic models and Data Analysis*, 7(2):139–151.

Mardia, K. V. and Jupp, P. E. (2009). *Directional statistics*, volume 494. John Wiley & Sons, New York, NY, USA.

McLachlan, G. J. and Peel, D. (2004). *Finite mixture models*. John Wiley & Sons.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations*, Arizona, USA. ICLR.

Newman, D., Karimi, S., and Cavedon, L. (2009). External evaluation of topic models. In *in Australasian Doc. Comp. Symp.* IEEE.

Rocci, R. and Vichi, M. (2008). Two-mode multipartitioning. *Computational Statistics & Data Analysis*, 52(4):1984–2003.

Röder, M., Both, A., and Hinneburg, A. (2015). Exploring the space of topic coherence measures. In *Proceedings of the eighth ACM international conference on Web search and data mining*, pages 399–408, Shanghai, China.

Role, F., Morbieu, S., and Nadif, M. (2019). Coclust: A python package for co-clustering. *Journal of Statistical Software, Articles*, 88(7):1–29.

Role, F. and Nadif, M. (2011). Handling the impact of low frequency events on co-occurrence based measures of word similarity. In *Proceedings of the International Conference on Knowledge Discovery and Information Retrieval (KDIR-2011). Scitepress*, pages 218–223.

Salah, A., Ailem, M., and Nadif, M. (2018). Word co-occurrence regularized non-negative matrix tri-factorization for text data co-clustering. In *Thirty-Second AAAI Conference on Artificial Intelligence*, pages 3992–3999.

Salah, A. and Nadif, M. (2017a). 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.

Salah, A. and Nadif, M. (2017b). Social regularized von mises–fisher mixture model for item recommendation. *Data Mining and Knowledge Discovery*, 31(5):1218–1241.

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

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464.

Steinley, D. (2004). Properties of the hubert-arable adjusted rand index. *Psychological methods*, 9(3):386.

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

Tanay, A., Sharan, R., and Shamir, R. (2005). Biclustering algorithms: A survey. *Handbook of computational molecular biology*, 9(1-20):122–124.

Van Mechelen, I., Bock, H.-H., and De Boeck, P. (2004). Two-mode clustering methods: astructuredoverview. *Statistical methods in medical research*, 13(5):363–394.

Vichi, M. (2001). Double k-means clustering for simultaneous classification of objects and variables. *Advances in Classification and Data Analysis*, pages 43–52.

Wang, H., Nie, F., Huang, H., and Makedon, F. (2011). Fast nonnegative matrix tri-factorization for large-scale data co-clustering. In *Twenty-Second International Joint Conference on Artificial Intelligence*.

Zhong, S. and Ghosh, J. (2005). Generative model-based document clustering: a comparative study. *Knowledge and Information Systems*, 8(3):374–384.