Data Min Knowl Disc (2013) 26:57–97
DOI 10.1007/s10618-011-0244-8

Regularized nonnegative shared subspace learning

Sunil Kumar Gupta · Dinh Phung · Brett Adams · Svetha Venkatesh

Received: 10 January 2011 / Accepted: 24 October 2011 / Published online: 16 November 2011

Abstract Joint modeling of related data sources has the potential to improve various data mining tasks such as transfer learning, multitask clustering, information retrieval etc. However, diversity among various data sources might outweigh the advantages of the joint modeling, and thus may result in performance degradations. To this end, we propose a regularized shared subspace learning framework, which can exploit the mutual strengths of related data sources while being immune to the effects of the variabilities of each source. This is achieved by further imposing a mutual orthogonality constraint on the constituent subspaces which segregates the common patterns from the source specific patterns, and thus, avoids performance degradations. Our approach is rooted in nonnegative matrix factorization and extends it further to enable joint analysis of related data sources. Experiments performed using three real world data sets for both retrieval and clustering applications demonstrate the benefits of regularization and validate the effectiveness of the model. Our proposed solution provides a formal framework appropriate for jointly analyzing related data sources and therefore, it is applicable to a wider context in data mining.

Keywords Nonnegative shared subspace learning · Transfer learning · Auxiliary sources · Multi-task clustering

Responsible editor: Kristian Kersting.

S. K. Gupta (✉) · D. Phung · B. Adams · S. Venkatesh
Department of Computing, Curtin University, Perth, WA 6102, Australia
e-mail: sunil.gupta@postgrad.curtin.edu.au

1 Introduction

Subspace learning is an important research area in data mining with diverse applications—information retrieval, face recognition, and data visualization among many others. Typical subspace learning algorithms are guided by a modeling goal to transform the data to a smaller dimension space, e.g. optimum classification in Fisher linear discriminant analysis (LDA) (Duda et al. 2001), optimum global representation in principal component analysis (PCA) (Jolliffe 2002) or local part-based representation in nonnegative matrix factorization (NMF) (Lee and Seung 2001). However, these methods fall short when applied to the burgeoning data from multiple data sources such as news feeds or social media sites. Diverse data sources relate to real world events, imparting their unique flavor. For example, Flickr focuses on images whilst YouTube on videos; different news channels have unique political views. Since standard subspace learning techniques are designed to analyze the data from a single source, they can either model the data from each source individually or model the combined data from all the sources by treating them as if they come from the same source. In practice, both these extremes lead to poor performance as the former approach does not exploit the mutual relationship between these similar data sources, whereas the later approach completely ignores their differences. Thus, cross channel modeling of similarities and differences is crucial.

It is known that modeling related sources together provides rich information, enabling discovery of hidden patterns. Multitask learning exploits the joint modeling of multiple related data sources (Caruana 1997; Pan and Yang 2008). There is a growing interest in learning joint subspaces (Ando and Zhang 2005; Ji et al. 2008; Yan et al. 2007; Si et al. 2009; Gu and Zhou 2009a; Zhuang et al. 2010; Zhang and Zhang 2011; Yang et al. 2010). Most of these approaches (Ando and Zhang 2005; Ji et al. 2008; Yan et al. 2007; Yang et al. 2010; Agarwal et al. 2010) focus on supervised/semi-supervised learning tasks, learning subspaces for classifier parameters. Relatively few attempts (Si et al. 2009; Gu and Zhou 2009a) have been made to model joint data subspaces for unsupervised tasks. Si et al. (2009) propose a transfer subspace learning algorithm to learn a subspace by minimizing the Bregman divergence between the target and auxiliary data distributions. Although their approach captures the common aspects between the two sources, it does not provide a way to preserve the individual variability within each source. Individual aspects are modeled by Gu and Zhou (2009a), however their framework requires the same number of clusters and forces identical cluster centroids for each data source. This assumption becomes too restrictive in application to real world data (Zhang and Zhang 2011; Zhuang et al. 2010).

Overcoming these open issues, we propose a shared subspace learning framework that can explicitly model the relationship between two data sources through a shared subspace, while maintaining their unique variations through individual subspaces. Our approach is flexible in modeling the level of sharing as well as individual variations between the data sources, and is controlled using the dimensionalities of the learnt subspaces. The shared subspace can be visualized as a space where the related data from different sources gets clustered, whereas individual subspaces are where the data points specific to a particular source are clustered.

To learn the constituent subspaces, we formulate our problem under a joint nonnegative matrix factorization framework. We use regularization to segregate the shared and individual subspaces, ensuring that the shared subspace does not capture individual aspects, and the individual subspaces do not capture shared aspects. To achieve the segregation, we model the constituent subspaces to be mutually orthogonal, minimizing the possibility of negative transfer learning (Dai et al. 2009) that may occur if shared subspaces capture domain specific aspects. To learn the joint factorization, we provide an optimization scheme based on multiplicative updates and show that its convergence is theoretically guaranteed. We analyze the computational complexity of the proposed algorithm and show that it remains similar in complexity to standard NMF. To demonstrate the usefulness of our framework, we apply it to diverse applications: tag-based social media retrieval and clustering. Using two real world data sets (Blogspot–Flickr–YouTube data set and BBC–CNN news data set), we show that our approach improves the performance over many state-of-the-art single and multi-task clustering techniques for both applications. In addition, we also provide a comparison of our method with these state-of-the-art techniques using 20 Newsgroup benchmark data set.

The novelty of our approach lies in proposing a flexible framework for modeling joint subspaces based on regularized NMF, with theoretical guarantees on algorithmic convergence. The significance of our work is its wide applicability to many problems in which our framework can leverage the common knowledge present in auxiliary domains and transfer it to a target domain. Crucially, by preserving the domain differences through individual subspaces, our approach is robust against negative transfer learning. This has implications in data mining tasks where joint analysis of two related but diverse data sources is required. Additionally, our framework returns subspaces that are intuitively meaningful compared to PCA or other mixed-sign factorizations as we retain the advantages of NMF in discovering local and parts-based representations (Lee and Seung 2001). For example, when analyzing CNN–BBC news data jointly, the shared subspace provides the common topics reported by both CNN and BBC, whereas individual subspaces represent topics related to specific channel coverage. In application to clustering, our method differs from other multitask clustering methods (Gu and Zhou 2009a; Zhang and Zhang 2011) which combine both subspace learning and clustering tasks. Our method does this in two steps. In the first step, it learns the regularized common and individual subspaces whilst in the second step, it uses these subspaces for clustering. This provides an important flexibility to use any state-of-the-art clustering method in conjunction with our shared subspace framework. In application to retrieval, our framework is capable of transferring knowledge from a cleaner data source to a related noisy data source and therefore, can boost the performance.

The rest of the paper is organized as follows. Section 2 briefly covers the necessary background and related works. Section 3 describes the formulation proposed earlier in Gupta et al. (2010) and extends it to develop a potentially more useful model (referred to as RJSNMF) by imposing a regularization scheme. Section 4 discusses two real world applications of the proposed RJSNMF and presents algorithms for them. Section 5 describes the experimental results conducted for the two applications and compares them with the state-of-the-art methods along with further analysis and discussion. Section 6 provides the details of learning model parameters along with the empirical study of convergence. Concluding remarks are given in Sect. 7. Necessary proofs and derivations are provided in Appendix A.

2 Related background

Increasing availability of data from related sources has given rise to their joint analysis. Previous works (Thrun 1996; Caruana 1997; Baxter 2000; Ben-David and Schuller 2003) provide a foundation for such analysis in multi-task learning. As a result, there has been lot of work (see the survey in Pan and Yang 2008) to improve both the supervised and unsupervised tasks using data from related auxiliary sources. Ando and Zhang (2005) discover predictive structures shared among various classification tasks by learning a subspace of parameters and use it to improve target data classification. Ji et al. (2008) provide a framework for extracting shared structures in multi-label classification by learning a shared subspace which is assumed to be shared among multiple labels. Since multiple labels share the same input space, and the semantics conveyed by different labels are often correlated, the authors exploit the correlation information contained in different labels. Yan et al. (2007) use a shared subspace boosting algorithm for multilabel classification which combines a number of base models across multiple labels to reduce the information overlap. The base models are learnt from random feature subspace and bootstrap data samples. All these methods focus on classification tasks and learn joint subspaces to model either labels or classification parameters. There have been few attempts (Si et al. 2009; Gu and Zhou 2009a) to learn joint subspace which directly model the data from related sources in unsupervised or semi-supervised settings. Si et al. (2009) propose to learn a shared subspace by minimizing the Bregman divergence between the distributions of the related data sources. This approach is generic to learn subspaces with different modeling goals. Although the shared subspace learnt using this approach is appropriate to model the commonalities between two related sources, it has no explicit provision for modeling individual variations of each data source. Modeling both shared and individual variations explicitly ensures that knowledge from auxiliary sources is exploited without sacrificing the knowledge available locally at each data source. Recently, a framework, which uses both shared and individual subspaces for jointly modeling data from related sources is proposed in Gu and Zhou (2009a) for a multi-task clustering application. Authors exploit the relationship of the similar tasks to enhance the clustering performance of each task and also propose a transductive transfer classification method under the propose framework. However, this framework needs to maintain the same number of clusters for each data source in their individual subspaces and forces identical cluster centroids in the shared subspace. This limitation renders the framework too restrictive for modeling real world data sources.

To address the above concerns, we proposed a shared subspace learning in Gupta et al. (2010) which uses both shared as well as individual subspaces. However, in this model, there are no explicit constraints on shared and individual subspaces which can ensure that shared subspace captures only shared aspects and individual subspace subspaces capture individual aspects only. For real world data sets, the shared subspace may capture some individual aspects of a particular domain creating noise for other domains. As a result, when using this noisy shared subspace for knowledge transfer from auxiliary to source domain, negative transfer knowledge may occur. This is illustrated empirically in our experiments section. To overcome this problem, we propose a regularized shared subspace learning framework that imposes a mutual orthogonality constraint among the constituent subspaces and ensures that shared subspace captures only shared aspects and individual subspaces capture only individual aspects.

The regularization based approaches have been very useful in achieving the desired solutions. Zhou et al. (2004) propose a semi-supervised learning technique to predict the labels for unlabeled data through regularizations applied to ensure both local and global consistency. The idea of local consistency is useful because nearby points are likely to have the same labels. Similarly, the global consistency exploits the fact that the points on the same structure (a cluster or a manifold) tend to have the same labels. Bringing the local consistency idea to NMF based algorithms, Gu and Zhou (2009b) propose to optimize a regularized cost function to obtain a nonnegative clustering algorithm with an assumption that the cluster label of each point can be predicted by the points in its neighborhood. Along similar lines, Cai et al. (2011) propose a graph regularized NMF by utilizing a graph Laplacian matrix to preserve the geometric structure of the data. Regularization techniques have also been used in conjunction with NMF to obtain orthogonal nonnegative factors. Choi (2008) proposes various schemes where regularization is used to impose orthogonality on either the basis matrix or the feature matrix. To obtain the desired solution, gradient-descent is computed directly in Stiefel manifold which reduces to multiplicative updates. Further generalizing this concept, Ding et al. (2006) propose bi-orthogonal tri-factor NMF and use it to simultaneously cluster rows and columns of a given data matrix. Although the regularization scheme used by us to obtain the mutually orthogonal subspaces seems to be similar to the above related works, there are some important and fundamental differences. First, our method is a joint subspace learning method, thus different from all above methods. Second, the regularization used in our optimization function does not attempt to obtain orthogonal subspaces. Rather, it seeks the shared and individual subspaces which are mutually orthogonal. In other words, the shared subspace is *not* orthogonal in itself but is mutually orthogonal to the individual subspaces. Similarly, each individual subspace is *not* orthogonal in itself but it is mutually orthogonal[1] to both the other individual subspace and the shared subspace.

Another set of works that are related to the proposed work are based on multi-view learning. Kailing et al. (2004) propose a density-based approach to cluster data points having multiple representations by using all available representations of the data. The authors extend the DBSCAN algorithm to enable its application on data having multiple representations. Bickel and Scheffer (2004) propose a partitioning and agglomerative, multi-view clustering algorithm by utilizing different available views of the data. The authors empirically show that the multi-view versions of clustering algorithms, especially K-means and EM, greatly improve their single-view counter-

[1] Formally, we have $(w_i)^T u_j = 0$, $(w_i)^T v_k = 0$ and $\left(u_j\right)^T v_k = 0$ but $(w_i)^T w_{i'} \neq 0$, $\left(u_j\right)^T u_{j'} \neq 0$, $(v_k)^T v_{k'} \neq 0$ if w_i, u_j and v_k are the basis vectors of shared subspace and the two individual subspaces respectively.

parts. These techniques assume all the representations of the data to have the same clustering structures and attempt to find a consensus among different views. Real world data, on the other hand, may not fully share the common structures across different views and under these scenarios, the above assumption is only partially true. Recently, Wiswedel et al. (2010) propose a technique to cluster data objects which have multiple representations. Instead of assuming the common structure over each view, this work combines local models from individual views into a global model learnt across all views. These approaches are clearly different from ours as they confine themselves to a single data source and focus on modeling data points that have multiple representations while our approach models two related data sources simulataneously and data points of each source have only single representation instead of multiple representations or views.

There has been a parallel approach to model the correlated data using canonical correlation analysis (CCA) which learns two orthogonal transformations such that transformed data pairs are maximally correlated. However, to learn such transformations, CCA needs the data from the two related sources such that every data point in the first source has a correspondence in the second data source. In other words, it requires two different views of the same object. Therefore, CCA is appropriate to model a pair of related data sources that provide two different views of the same phenomenon and comes under the multi-view approaches discussed in previous paragraph. This intuition is exploited through recent multi-view formulations (Chaudhuri et al. 2009; Hardoon et al. 2004). Since, these correspondences are not available in general (e.g. the context of this paper), the use of CCA is restricted within multi-view applications.

Finally, we distinguish between our work and *alternate clustering* paradigm (Bae and Bailey 2006; Cui et al. 2007; Qi and Davidson 2009; Niu et al. 2010). Often, high dimensional data can be visualized in different ways leading to multiple ways of data clustering. In particular, it is useful to consider various *non-redundant* alternate clusters that provide information complementary to one another. Given an existing clustering partition, Bae and Bailey (2006) find an alternate clustering partition by generating a set of ‘cannot-link’ constraints from the data-pairs in the same cluster group of the given clustering partition, and use agglomerative clustering to find the alternate clustering. The idea is to get another clustering partition that can discover patterns distinctively different from those captured by the existing clustering. Qi and Davidson (2009) propose an approach for alternate clustering by transforming data to a space different from the original space and, to avoid significant distortions in the data, they minimize KL-divergence between the distributions of the data in the two spaces. However, to ensure that the alternate clustering partition is different from the clustering in original space, they impose a constraint such that the probability of a data sample belonging to the same cluster in the projected space is smaller than a pre-specified threshold. Both the above works are *different* from ours as first, they attempt to find multiple non-redundant clustering partitions of a *single data source* whereas we are dealing with *multiple data sources,* finding a single clustering partition for each data source under transfer learning setting; second, these approaches do not use orthogonality constraints. Other alternate clustering techniques (Cui et al. 2007; Niu et al. 2010) use some form of orthogonalization, but their idea of using orthogonalization is fundamentally different from ours. Cui et al. (2007) propose a clustering scheme to discover multiple non-redundant clustering partitions by using orthogonality either in the cluster or the feature space. Non-redundant clusters are obtained by first clustering the data in a discriminating subspace and then, alternate clusters are obtained by projecting the data to subspaces orthogonal to the former subspace. Unlike this approach, which attempts to find multiple non-redundant clustering partitions, we focus on finding a single clustering partition for one data source while leveraging data from a related auxiliary source to improve the clustering performance. Since our goal is to transfer only *relevant* knowledge (i.e. the aspects which are shared across the two data sources), we need to learn a shared subspace that does not capture any individual aspects. Orthogonality constraints in our model ensure this by learning shared and individual subspaces that are as distinct as possible. As a result, the shared subspace is restricted to only shared aspects, and each individual subspace is restricted to the unique aspects of the respective source. In another work, Niu et al. (2010) extend spectral clustering for discovering multiple non-redundant clustering partitions of data. To reduce the redundancy across the different views, they utilize Hilbert–Schmidt Independence Criterion (HSIC)—a criterion that measures the dependency (not just linear dependency but also the higher orders) between two random variables without requiring the knowledge of their joint distribution. This approach is based on an intuition that different views most likely exist in different subspaces and thus the authors provide a way to learn the subspaces in addition to the alternate clustering. These subspaces are assumed to be orthogonal *per se*; however, different from our work, there is no assumption or imposition of mutual orthogonality among the subspaces of different views.

Our proposed framework is based on nonnegative matrix factorization (NMF). When dealing with nonnegative data, NMF has been shown to perform very well for variety of tasks such as text mining (Berry and Browne 2005; Lin et al. 2009), document and image clustering (Xu et al. 2003; Cai et al. 2008), video content representation (Bucak and Gunsel 2007) etc. We extend NMF to a transfer learning framework where we can leverage data from related auxiliary sources to improve various unsupervised or semi-supervised tasks (e.g. retrieval, clustering) on target data. Mathematically, NMF (Lee and Seung 2001) aims to factorize a nonnegative data matrix into another two nonnegative matrices such that

$$\boldsymbol{X} \approx \mathbf{FH},\ \mathbf{F} \geq 0,\ \mathbf{H} \geq 0$$

where assuming that a $M \times N$ data matrix $\boldsymbol{X}$ contains N data points in an M-dimensional space, the first $M \times R$ factor matrix $\mathbf{F}$ can be thought of representing a reduced dimensional nonnegative subspace ($R < N$) whose bases vectors are given by the columns of matrix $\mathbf{F}$ and the second $R \times N$ factor matrix $\mathbf{H}$ contains the representation of each data point in the subspace. Since, the above factorization is not unique, the solution depends on the algorithm used and its initialization. To see an instance of non-uniqueness, consider two matrices $\mathbf{F}$ and $\mathbf{H}$ that are a solution to $\boldsymbol{X} = \mathbf{FH}$, then $\mathbf{FS}$ and $\mathbf{S}^{-1}\mathbf{H}$ are also a solution for any positive diagonal matrix $\mathbf{S}$. Normalization of $\mathbf{F}$ to L_2-norm one eliminates the degree of freedom caused by such diagonal matrices. Therefore, a common practice (Xu et al. 2003; Cai et al. 2011) is to normalize each column of matrix $\mathbf{F}$ to L_2-norm one. To keep the product unchanged, matrix $\mathbf{H}$ is accordingly adjusted.

The local (part-based) nature of NMF comes from the fact that nonnegative basis vectors (parts) from matrix $\mathbf{F}$ combine in an nonnegative fashion using the coefficients of matrix $\mathbf{H}$ to construct a data point (create "whole") in matrix $\boldsymbol{X}$. There have been many algorithms proposed for learning such factorization (see the survey in Berry et al. 2007) and probably, the most popular of them is the approach taken by Lee and Seung (2001) who propose an iterative multiplicative update based solution to the following constrained optimization problem

$$\text{minimize } \|\boldsymbol{X} - \mathbf{F} \cdot \mathbf{H}\|_F^2 \text{ , subject to } \mathbf{F}, \mathbf{H} \geq 0$$

Often, the factorization matrices $\mathbf{F}$ and $\mathbf{H}$ may have special interpretation for real world data. For example, consider text mining application where data matrix $\boldsymbol{X}$ is the usual term-document matrix constructed using the term-frequencies. In this case, each column of the matrix $\mathbf{F}$ when normalized to sum to one, represents a probability distribution on the vocabulary and can be interpreted as a topic (Blei et al. 2003). Thus, the subspace spanned by columns of matrix $\mathbf{F}$ can be interpreted as a topic space. Given such interpretation, each column of matrix $\mathbf{H}$ provides the mixing proportions of these topics and represents the corresponding document in the topic space.

3 Preliminaries and problem formulation

3.1 Nonnegative shared subspace learning (JSNMF)

To analyze two data sources jointly, in our previous work (Gupta et al. 2010), we proposed a shared subspace learning framework (referred to as JSNMF) which explicitly learns both the shared and individual subspaces. Formally, JSNMF aims to achieve the following factorization

$$\boldsymbol{X} \approx \underbrace{[\mathbf{W} \mid \boldsymbol{U}]}_{\mathbf{F}} \underbrace{\begin{bmatrix} \mathbf{H}_w \\ \hline \mathbf{H}_u \end{bmatrix}}_{\mathbf{H}} = \mathbf{W}\mathbf{H}_w + \boldsymbol{U}\mathbf{H}_u = \mathbf{F}\mathbf{H} \tag{1}$$

$$\boldsymbol{Y} \approx \underbrace{[\mathbf{W} \mid \boldsymbol{V}]}_{\mathbf{G}} \underbrace{\begin{bmatrix} \mathbf{L}_w \\ \hline \mathbf{L}_v \end{bmatrix}}_{\mathbf{L}} = \mathbf{W}\mathbf{L}_w + \boldsymbol{V}\mathbf{L}_v = \mathbf{G}\mathbf{L} \tag{2}$$

where columns of a $M \times K$ matrix $\mathbf{W}$ span the shared subspace; matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ span the individual subspaces and have dimensions $M \times (R_1 - K)$ and $M \times (R_2 - K)$ respectively. R_1, R_2 denote the total subspace dimensionalities whereas K denotes the dimensionality of the shared subspace. Matrices $\mathbf{F}$ and $\mathbf{G}$ are used to denote the total subspaces and are defined as $\mathbf{F} \triangleq [\mathbf{W} \mid \boldsymbol{U}]$ and $\mathbf{G} \triangleq [\mathbf{W} \mid \boldsymbol{V}]$. Akin to standard NMF, $R_1 \times N_1$ matrix $\mathbf{H}$ and $R_2 \times N_2$ matrix $\mathbf{L}$ contain the representation of data points in the subspaces spanned by $\mathbf{F}$ and $\mathbf{G}$ respectively. $\mathbf{H}_w$ (and $\mathbf{L}_w$) denotes the representation of data matrix $\boldsymbol{X}$ (and $\boldsymbol{Y}$) in the shared subspace $\mathbf{W}$ whereas $\mathbf{H}_u$ and $\mathbf{L}_v$ denote the representation of data matrices $\boldsymbol{X}$ and $\boldsymbol{Y}$ in their corresponding individual subspaces $\boldsymbol{U}$ and $\boldsymbol{V}$. Note that although $\boldsymbol{X}$ and $\boldsymbol{Y}$ usually have different number of rows (e.g. vocabularies in text mining) but they can be merged together to construct a common input space that has dimension M (total number of words in the merged vocabulary becomes M). To learn the required subspaces, JSNMF minimizes the following cost function

$$\min_{\mathbf{W},\boldsymbol{U},\boldsymbol{V},\mathbf{H},\mathbf{L}\geq 0}\left\{\lambda_{\boldsymbol{X}}\left\|\boldsymbol{X}-[\mathbf{W}\mid\boldsymbol{U}]\mathbf{H}\right\|_F^2+\lambda_{\boldsymbol{Y}}\left\|\boldsymbol{Y}-[\mathbf{W}\mid\boldsymbol{V}]\mathbf{L}\right\|_F^2\right\}$$

where $\lambda_{\boldsymbol{X}}\triangleq\|\boldsymbol{X}\|_F^{-2}$, $\lambda_{\boldsymbol{Y}}\triangleq\|\boldsymbol{Y}\|_F^{-2}$ and by $\boldsymbol{A}\geq 0$, we mean that (i,j)-th element of matrix $\boldsymbol{A}$, i.e. $\boldsymbol{A}_{ij}$ is greater or equal to zero for each i, j.

An extension of JSNMF to more than two data sources having an arbitrary sharing configuration has been proposed in Gupta et al. (2011b). Another similar extension to mixed sign factorization using Bayesian framework is proposed in Gupta et al. (2011a).

3.2 Regularized nonnegative shared subspace learning (RJSNMF)

The above model has a crucial drawback in segregating the shared and individual subspaces: there is no constraint on the individual subspaces to prevent them from capturing basis vectors from the shared subspace. Neither is there a constraint that stops leakage of individual basis vectors into the shared subspace. Of these two issues, the first is important when we are interested in the individual aspects of the two data sources. However, the second issue is more important when looking from the point of view of transfer learning or multi-task learning. This is because when the matrix $\mathbf{W}$ captures some topics or basis vectors corresponding to the individual subspace of one data source and used later for modeling the other data source, this leakage degrades performance.

To provide a solution to the above problems, we propose a regularization on the common and individual subspaces which can avoid both of the above problems. It not only ensures that $\boldsymbol{U}$ and $\boldsymbol{V}$ capture only the individual basis vectors but also that $\mathbf{W}$ captures only the common basis vectors. With this regularization, our optimization cost function takes the following form

$$\min_{\mathbf{W},\boldsymbol{U},\boldsymbol{V},\mathbf{H},\mathbf{L}\geq 0}\left\{\lambda_{\boldsymbol{X}}\left\|\boldsymbol{X}-[\mathbf{W}\mid\boldsymbol{U}]\mathbf{H}\right\|_F^2+\lambda_{\boldsymbol{Y}}\left\|\boldsymbol{Y}-[\mathbf{W}\mid\boldsymbol{V}]\mathbf{L}\right\|_F^2+R\left(\mathbf{W},\boldsymbol{U},\boldsymbol{V}\right)\right\} \tag{3}$$

where $R\left(\mathbf{W},\boldsymbol{U},\boldsymbol{V}\right)$ is a regularization term used to penalize the “similarity” between subspaces spanned by matrices $\mathbf{W}$, $\boldsymbol{U}$ and $\boldsymbol{V}$. When $R\left(\mathbf{W},\boldsymbol{U},\boldsymbol{V}\right)=0$, in this special case, the model reduces to the unregularized version proposed in Gupta et al. (2010).

When seeking subspaces which do not capture the similar basis vectors (e.g. similar topics in case of text data) and be complementary to each other, one way to formulate them is by considering them as mutually orthogonal. Note for example that if the subspaces spanned by matrices $\mathbf{W}$, $\boldsymbol{U}$, are mutually orthogonal[2], we have $\mathbf{W}^{\mathsf{T}}\boldsymbol{U} = \mathbf{0}$. To impose this constraint, we choose to minimize the sum-of-squares of entries of the matrix $\mathbf{W}^{\mathsf{T}}\boldsymbol{U}$, i.e. $\left\|\mathbf{W}^{\mathsf{T}}\boldsymbol{U}\right\|_F^2$ which uniformly optimizes each entry of $\mathbf{W}^{\mathsf{T}}\boldsymbol{U}$. With this choice, the regularization term of Eq. 3 is given by

$$R(\mathbf{W}, \boldsymbol{U}, \boldsymbol{V}) = \alpha \left\|\mathbf{W}^{\mathsf{T}}\boldsymbol{U}\right\|_F^2 + \beta \left\|\mathbf{W}^{\mathsf{T}}\boldsymbol{V}\right\|_F^2 + \gamma \left\|\boldsymbol{U}^{\mathsf{T}}\boldsymbol{V}\right\|_F^2 \tag{4}$$

where α, β and γ are the regularization parameters.

We note that an alternative formulation could have been based on minimizing the sum of entries[3] in matrix $\mathbf{W}^{\mathsf{T}}\boldsymbol{U}$, i.e. $\mathbf{1}_{\mathbf{w}}^{\mathsf{T}}\mathbf{W}^{\mathsf{T}}\boldsymbol{U}\mathbf{1}_u$ where $\mathbf{1}_w$ and $\mathbf{1}_u$ are the vector of ones of appropriate lengths. This will tend to give solutions such that entries of $\mathbf{W}^{\mathsf{T}}\boldsymbol{U}$ are sparse, meaning that most of the entries in $\mathbf{W}^{\mathsf{T}}\boldsymbol{U}$ would be either small or zero but there could be few large entries in $\mathbf{W}^{\mathsf{T}}\boldsymbol{U}$. However, since our goal here is to get a solution such that shared subspace spanned by matrix $\mathbf{W}$ does not capture any basis vector similar to the basis vectors of individual subspaces (spanned by $\boldsymbol{U}$ or $\boldsymbol{V}$), none of the entries in $\mathbf{W}^{\mathsf{T}}\boldsymbol{U}$ or $\mathbf{W}^{\mathsf{T}}\boldsymbol{V}$ are desired to be large.[4]

3.2.1 Unified formulation

After choosing the regularization term $R(\mathbf{W}, \boldsymbol{U}, \boldsymbol{V})$ as in Eq. 4 and substituting into Eq. 3, our final optimization formulation is given as

$$\Pi : \begin{cases} \text{minimize} & \left\{\lambda_X \|\boldsymbol{X} - [\mathbf{W} \mid \boldsymbol{U}]\mathbf{H}\|_F^2 + \lambda_Y \|\boldsymbol{Y} - [\mathbf{W} \mid \boldsymbol{V}]\mathbf{L}\|_F^2\right\} \\ & + \left\{\alpha \left\|\mathbf{W}^{\mathsf{T}}\boldsymbol{U}\right\|_F^2 + \beta \left\|\mathbf{W}^{\mathsf{T}}\boldsymbol{V}\right\|_F^2 + \gamma \left\|\boldsymbol{U}^{\mathsf{T}}\boldsymbol{V}\right\|_F^2\right\} \\ \text{subject to} & \mathbf{W}, \boldsymbol{U}, \boldsymbol{V}, \mathbf{H}, \mathbf{L} \geq 0 \end{cases}$$

In the above formulation, there is a trade-off between fitting the data well and having subspaces which are mutually orthogonal. This trade-off is controlled by the regularization parameters α, β and γ. In Sect. 6, we provide graphs depicting the variation of the cost function w.r.t. these regularization parameters and empirically show that an optimum value exists for these parameters.

[2] To see this, consider any two vectors p_i and q_j from the subspaces spanned by $\mathbf{W}$ and $\boldsymbol{U}$ respectively and note that $p_i = \mathbf{W}r_i$ and $q_j = \boldsymbol{U}s_j$. For the two subspaces to be mutually orthogonal, we have $p_i^{\mathsf{T}}q_j = 0, \forall p_i, q_j$ which leads to $r_i^{\mathsf{T}}\mathbf{W}^{\mathsf{T}}\boldsymbol{U}s_j = 0$. Since this relation has to hold for every r_i and s_j, we have $\mathbf{W}^{\mathsf{T}}\boldsymbol{U} = \mathbf{0}$.

[3] Note that taking the modulus of the entries of matrix $\mathbf{W}^{\mathsf{T}}\boldsymbol{U}$ is not needed as matrices $\mathbf{W}$ and $\boldsymbol{U}$ are already constrained to be nonnegative.

[4] Assuming that matrices $\mathbf{W}$, $\boldsymbol{U}$, $\boldsymbol{V}$ have their columns normalized to norm one (which is a usually done to reduce the ill-posed nature or non-uniqueness of the problem in NMF based factorizations), in practice, it is considered enough small if $\mathbf{W}^{\mathsf{T}}\boldsymbol{U} \leq 0.1$.

As an alternative to the optimization problem Π, one can express the regularization conditions explicitly through a set of constraints which is expressed as follows

$$\Pi' \ : \ \begin{cases} \text{minimize} & \left\{\lambda_X \left\|X - [\mathbf{W} \mid U]\mathbf{H}\right\|_F^2 + \lambda_Y \left\|Y - [\mathbf{W} \mid V]\mathbf{L}\right\|_F^2\right\} \\ \text{subject to} & \mathbf{W}, U, V, \mathbf{H}, \mathbf{L} \geq 0\ ,\ \left\|\mathbf{W}^\mathsf{T} U\right\|_F^2 \leq T_\alpha\ , \\ & \left\|\mathbf{W}^\mathsf{T} V\right\|_F^2 \leq T_\beta\ ,\ \left\|U^\mathsf{T} V\right\|_F^2 \leq T_\gamma \end{cases}$$

However, it can be verified that, for every solution $\{\mathbf{W}, U, V, \mathbf{H}, \mathbf{L}\}$ to the optimization problem Π using some particular values of parameters α, β and γ, there are equivalent parameters T_α, T_β and T_γ in the optimization problem Π' which lead to the same solution $\{\mathbf{W}, U, V, \mathbf{H}, \mathbf{L}\}$ for Π'. Thus, Π and Π' are equivalent and it is sufficient to solve the optimization problem Π.

3.2.2 Optimization and algorithm

Note that the optimization problem Π is not convex in variables $\{\mathbf{W}, U, V, \mathbf{H}, \mathbf{L}\}$ together. However, when considering one variable at a time, the cost function turns out to be convex. For example, given $\{U, V, \mathbf{H}, \mathbf{L}\}$, the cost function is a convex function w.r.t. $\mathbf{W}$. Therefore, although we can not expect to get a global minimum of above problem, we shall develop an algorithm which is not only simple and efficient but its convergence can also be guaranteed.

The problem Π is a constrained optimization problem due to the nonnegative constraints on the factorization matrices, and can be solved using the Lagrange multiplier method. Let A_{ij}^w be the Lagrangian multiplier for constraints $\mathbf{W}_{ij} \geq 0$ i.e. for (i, j)-th element of matrix $\mathbf{W}$ and let $A^w = \left[A_{ij}^w\right]$. Similarly if A^u, A^v, A^h, A^l are the Lagrangian multiplier matrices for nonnegative constraints of matrices $U, V, \mathbf{H}, \mathbf{L}$, then the above cost function in an unconstrained form (denoted by C) can be written as below

$$C = \lambda_X \left\|X - [\mathbf{W} \mid U]\mathbf{H}\right\|_F^2 + \lambda_Y \left\|Y - [\mathbf{W} \mid V]\mathbf{L}\right\|_F^2 + D\left(\mathbf{W}, U, V, \mathbf{H}, \mathbf{L}\right) \quad (5)$$

where $D\left(\mathbf{W}, U, V, \mathbf{H}, \mathbf{L}\right)$ is defined as

$$\begin{aligned} D\left(\mathbf{W}, U, V, \mathbf{H}, \mathbf{L}\right) \triangleq{}& \alpha \left\|\mathbf{W}^\mathsf{T} U\right\|_F^2 + \beta \left\|\mathbf{W}^\mathsf{T} V\right\|_F^2 + \gamma \left\|U^\mathsf{T} V\right\|_F^2 + \operatorname{Tr}\left(A^w \mathbf{W}^\mathsf{T}\right) \\ &+ \operatorname{Tr}\left(A^u U^\mathsf{T}\right) + \operatorname{Tr}\left(A^v V^\mathsf{T}\right) + \operatorname{Tr}\left(A^h \mathbf{H}^\mathsf{T}\right) + \operatorname{Tr}\left(A^l \mathbf{L}^\mathsf{T}\right) \end{aligned}$$

Optimize $\mathbf{W}$ *given* $\{U, V, \mathbf{H}, \mathbf{L}\}$

The first derivative of the cost function C with respect to matrix $\mathbf{W}$ is given by

$$\nabla_\mathbf{W} C = 2\left[\lambda_X \left(X^{(t)} - X\right)\mathbf{H}_w^\mathsf{T} + \lambda_Y \left(Y^{(t)} - Y\right)\mathbf{L}_w^\mathsf{T} + \left(\alpha U U^\mathsf{T} + \beta V V^\mathsf{T}\right)\mathbf{W} + A^w\right]$$

where $\boldsymbol{X}^{(t)} \triangleq [\mathbf{W}^{(t)} \mid \boldsymbol{U}^{(t)}]\mathbf{H}^{(t)}$ and $\boldsymbol{Y}^{(t)} \triangleq [\mathbf{W}^{(t)} \mid \boldsymbol{V}^{(t)}]\mathbf{L}^{(t)}$. Using Karush–Kuhn–Tucker (KKT) conditions $\boldsymbol{A}^{w}_{ij}\mathbf{W}_{ij} = 0$ with the expression of the gradient $\nabla_{\mathbf{W}} C$, for any stationary point, we get the following

$$\left[\lambda_{\boldsymbol{X}}\left(\boldsymbol{X}^{(t)} - \boldsymbol{X}\right)\mathbf{H}_w^{\top} + \lambda_{\boldsymbol{Y}}\left(\boldsymbol{Y}^{(t)} - \boldsymbol{Y}\right)\mathbf{L}_w^{\top} + \left(\alpha \boldsymbol{U}\boldsymbol{U}^{\top} + \beta \boldsymbol{V}\boldsymbol{V}^{\top}\right)\mathbf{W}\right]_{ij}\mathbf{W}_{ij} = 0$$

which leads to the following update equation

$$\mathbf{W}_{ij} \leftarrow \mathbf{W}_{ij}\frac{\left[\lambda_{\boldsymbol{X}}\boldsymbol{X}\mathbf{H}_w^{\top} + \lambda_{\boldsymbol{Y}}\boldsymbol{Y}\mathbf{L}_w^{\top}\right]_{ij}}{\left[\lambda_{\boldsymbol{X}}\boldsymbol{X}^{(t)}\mathbf{H}_w^{\top} + \lambda_{\boldsymbol{Y}}\boldsymbol{Y}^{(t)}\mathbf{L}_w^{\top} + \left(\alpha \boldsymbol{U}\boldsymbol{U}^{\top} + \beta \boldsymbol{V}\boldsymbol{V}^{\top}\right)\mathbf{W}\right]_{ij}} \tag{6}$$

Optimize $\boldsymbol{U}$ *given* $\{\mathbf{W}, \boldsymbol{V}, \mathbf{H}, \mathbf{L}\}$

The first derivative of the cost function C with respect to matrix $\boldsymbol{U}$ is given by

$$\nabla_{\boldsymbol{U}} C = 2\left[\lambda_{\boldsymbol{X}}\left(\boldsymbol{X}^{(t)} - \boldsymbol{X}\right)\mathbf{H}_u^{\top} + \left(\alpha \mathbf{W}\mathbf{W}^{\top} + \gamma \boldsymbol{V}\boldsymbol{V}^{\top}\right)\boldsymbol{U} + \boldsymbol{A}^{u}\right]$$

Similar as above, using KKT conditions $\boldsymbol{A}^{u}_{ij}\boldsymbol{U}_{ij} = 0$ with the expression of the gradient $\nabla_{\boldsymbol{U}} C$, for any stationary point, we get the following update equation

$$\boldsymbol{U}_{ij} \leftarrow \boldsymbol{U}_{ij}\frac{\left[\lambda_{\boldsymbol{X}}\boldsymbol{X}\mathbf{H}_u^{\top}\right]_{ij}}{\left[\lambda_{\boldsymbol{X}}\boldsymbol{X}^{(t)}\mathbf{H}_u^{\top} + \left(\alpha \mathbf{W}\mathbf{W}^{\top} + \gamma \boldsymbol{V}\boldsymbol{V}^{\top}\right)\boldsymbol{U}\right]_{ij}} \tag{7}$$

Similarly, optimizing for $\boldsymbol{V}$ given $\{\mathbf{W}, \boldsymbol{U}, \mathbf{H}, \mathbf{L}\}$, we get the following update equation

$$\boldsymbol{V}_{ij} \leftarrow \boldsymbol{V}_{ij}\frac{\left[\lambda_{\boldsymbol{Y}}\boldsymbol{Y}\mathbf{L}_w^{\top}\right]_{ij}}{\left[\lambda_{\boldsymbol{Y}}\boldsymbol{Y}^{(t)}\mathbf{L}_w^{\top} + \left(\beta \mathbf{W}\mathbf{W}^{\top} + \gamma \boldsymbol{U}\boldsymbol{U}^{\top}\right)\boldsymbol{V}\right]_{ij}} \tag{8}$$

Update equations for matrices $\mathbf{H}$ and $\mathbf{L}$, given $\{\mathbf{W}, \boldsymbol{U}, \boldsymbol{V}, \mathbf{L}\}$ and $\{\mathbf{W}, \boldsymbol{U}, \boldsymbol{V}, \mathbf{H}\}$ respectively, are similar to standard NMF and are given by

$$\mathbf{H}_{ij} \leftarrow \mathbf{H}_{ij}\frac{\left[\mathbf{F}^{\top}\boldsymbol{X}\right]_{ij}}{\left[\mathbf{F}^{\top}\mathbf{F}\mathbf{H}\right]_{ij}},\ \mathbf{L}_{ij} \leftarrow \mathbf{L}_{ij}\frac{\left[\mathbf{G}^{\top}\boldsymbol{Y}\right]_{ij}}{\left[\mathbf{G}^{\top}\mathbf{G}\mathbf{L}\right]_{ij}} \tag{9}$$

We note that multiplicative updates given by Eqs. 6–9 are obtained by extending the updates of standard NMF (Lee and Seung 2001). There are alternative ways of optimizing the objective function Π such as alternating least squares and the active set method (Kim and Park 2008) or the projected gradients approach (Lin 2007), which often have better convergence behavior. Nonetheless, the multiplicative updates derived in this work have reasonably fast convergence behavior as shown empirically in Sect. 6.3.

Similarly, a good initialization of matrices $\mathbf{W}$, $\boldsymbol{U}$, $\boldsymbol{V}$, $\mathbf{H}$ and $\mathbf{L}$ may lead to quicker convergence of the proposed algorithm. Several methods have been proposed for NMF in literature (Wild et al. 2004; Langville et al. 2006; Boutsidis and Gallopoulos 2008). However, it is not obvious how to use them for initializing shared subspace matrix $\mathbf{W}$ and the corresponding coefficient matrices $\mathbf{H}_w$ and $\mathbf{L}_w$. Other matrices such as $\boldsymbol{U}$, $\boldsymbol{V}$ etc also depend on $\mathbf{W}$ given the data. Therefore, we confine ourselves to the random initialization of these matrices.

For future references, we denote the parameters required for RJSNMF collectively as a set $\Psi = \{R_1, R_2, K, \alpha, \beta, \gamma\}$. Algorithm 1 provides the details of learning the subspace matrices $\mathbf{W}$, $\boldsymbol{U}$, $\boldsymbol{V}$, $\mathbf{H}$ and $\mathbf{L}$ given the data matrices $\boldsymbol{X}$ and $\boldsymbol{Y}$.

Algorithm 1 Regularized nonnegative shared subspace learning (RJSNMF).

```
Input: Data matrices $X$, $Y$, parameter set $\Psi$, convergence threshold $\epsilon$ or maximum number of iteration ($Maxiter$).
compute $\lambda_X$ and $\lambda_Y$ as $\lambda_X = \|X\|_F^{-2}$ and $\lambda_Y = \|Y\|_F^{-2}$.
initialize matrices $W$, $U$, $V$, $H$ and $L$ with random nonnegative values.
$r = 1$.
while $(r < Maxiter)$ or $(C > \epsilon)$ do
  update matrices $W$, $U$, $V$, $H$ and $L$ using Eqs. 6–9.
  normalize each column of matrices $W$, $U$ and $V$ to norm one.
  compute the cost function $(C)$ of optimization problem $\Pi$.
  $r = r + 1$.
end while
Output: Return subspace matrices $W$, $U$, $V$, $H$ and $L$.
```

3.2.3 Convergence analysis

In this section, we prove the convergence of multiplicative updates given by Eqs. 6–9 and analyze the computational complexity Algorithm 1. We first prove few lemmas required later to prove a theorem which states that following the multiplicative updates given by Eqs. 6–9, the cost function of Eq. 3 converges. The proof follows similar lines as the convergence proof of Expectation-Maximization (EM) algorithm (Dempster et al. 1977) and NMF (Lee and Seung 2001) where the desired cost function is minimized indirectly by minimizing an upper bound function to the cost function.

Definition 1 $J\left(w, w'\right)$ is an auxiliary function for $C\left(w\right)$ if $C\left(w\right) \leq J\left(w, w'\right)$ and equality holds if and only if $w = w'$.

Lemma 1 (Lee and Seung 2001) *If J is an auxiliary function for C, C is non-increasing under the update*

$$w^{t+1} = \underset{w}{argmin}\ J\left(w, w^t\right)$$

Proof By definition, $C\left(w^{t+1}\right) \leq J\left(w^{t+1}, w^t\right) \leq J\left(w^t, w^t\right) = C\left(w^t\right)$. □

Lemma 2 *If* $C_{ij}\left(\mathbf{W}_{ij}^{(t)}\right)$ *consists of all the terms of cost function* $C\left(\mathbf{W}^{(t)}\right)$ *involving* $\mathbf{W}_{ij}^{(t)}$ *and if* $S_{ij}\left(\mathbf{W}^{(t)}\right)=\frac{\left[\lambda_{\boldsymbol{X}}\boldsymbol{X}^{(t)}\mathbf{H}_w^\top+\lambda_{\boldsymbol{Y}}\boldsymbol{Y}^{(t)}\mathbf{L}_w^\top+\left(\alpha\boldsymbol{U}\boldsymbol{U}^\top+\beta\boldsymbol{V}\boldsymbol{V}^\top\right)\mathbf{W}^{(t)}\right]_{ij}}{\mathbf{W}_{ij}^{(t)}}$ *then*

$$J\left(\mathbf{W}_{ij},\mathbf{W}_{ij}^{(t)}\right)=C_{ij}\left(\mathbf{W}_{ij}^{(t)}\right)+\left(\mathbf{W}_{ij}-\mathbf{W}_{ij}^{(t)}\right)\nabla C_{ij}\left(\mathbf{W}_{ij}^{(t)}\right)$$
$$+\frac{1}{2}\left(\mathbf{W}_{ij}-\mathbf{W}_{ij}^{(t)}\right)^2 S_{ij}\left(\mathbf{W}^{(t)}\right)$$

is an auxiliary function for $C_{ij}\left(\mathbf{W}_{ij}\right)$.

Proof See Appendix A. □

Lemma 3 *If* $C_{ij}\left(\boldsymbol{U}_{ij}^{(t)}\right)$ *consists of all the terms of cost function* $C\left(\boldsymbol{U}^{(t)}\right)$ *involving* $\boldsymbol{U}_{ij}^{(t)}$ *and if* $S_{ij}\left(\boldsymbol{U}^{(t)}\right)=\frac{\left[\lambda_{\boldsymbol{X}}\boldsymbol{X}^{(t)}\mathbf{H}_u^\top+\left(\alpha\mathbf{W}\mathbf{W}^\top+\gamma\boldsymbol{V}\boldsymbol{V}^\top\right)\boldsymbol{U}^{(t)}\right]_{ij}}{\boldsymbol{U}_{ij}^{(t)}}$ *then*

$$J\left(\boldsymbol{U}_{ij},\boldsymbol{U}_{ij}^{(t)}\right)=C_{ij}\left(\boldsymbol{U}_{ij}^{(t)}\right)+\left(\boldsymbol{U}_{ij}-\boldsymbol{U}_{ij}^{(t)}\right)\nabla C_{ij}\left(\boldsymbol{U}_{ij}^{(t)}\right)$$
$$+\frac{1}{2}\left(\boldsymbol{U}_{ij}-\boldsymbol{U}_{ij}^{(t)}\right)^2 S_{ij}\left(\boldsymbol{U}^{(t)}\right)$$

is an auxiliary function for $C_{ij}\left(\boldsymbol{U}_{ij}\right)$.

Proof The proof is similar to the proof of the Lemma 2. □

Lemma 4 *If* $C_{ij}\left(\boldsymbol{V}_{ij}^{(t)}\right)$ *consists of all the terms of cost function* $C\left(\boldsymbol{V}^{(t)}\right)$ *involving* $\boldsymbol{V}_{ij}^{(t)}$ *and if* $S_{ij}\left(\boldsymbol{V}^{(t)}\right)=\frac{\left[\lambda_{\boldsymbol{Y}}\boldsymbol{Y}^{(t)}\mathbf{L}_v^\top+\left(\beta\mathbf{W}\mathbf{W}^\top+\gamma\boldsymbol{U}\boldsymbol{U}^\top\right)\boldsymbol{V}^{(t)}\right]_{ij}}{\boldsymbol{V}_{ij}^{(t)}}$ *then*

$$J\left(\boldsymbol{V}_{ij},\boldsymbol{V}_{ij}^{(t)}\right)=C_{ij}\left(\boldsymbol{V}_{ij}^{(t)}\right)+\left(\boldsymbol{V}_{ij}-\boldsymbol{V}_{ij}^{(t)}\right)\nabla C_{ij}\left(\boldsymbol{V}_{ij}^{(t)}\right)$$
$$+\frac{1}{2}\left(\boldsymbol{V}_{ij}-\boldsymbol{V}_{ij}^{(t)}\right)^2 S_{ij}\left(\boldsymbol{V}^{(t)}\right)$$

is an auxiliary function for $C_{ij}\left(\boldsymbol{V}_{ij}\right)$.

Proof Again, the proof is similar to the proof of the Lemma 2. □

We do not provide any Lemma for proving update equation for matrix $\mathbf{H}$ as it can be seen from the optimization problem Π that solution for $\mathbf{H}$ can be obtained using the results of standard NMF (Lee and Seung 2001). By considering the definitions $\mathbf{F}\triangleq[\mathbf{W}\mid\boldsymbol{U}]$ and $\mathbf{G}\triangleq[\mathbf{W}\mid\boldsymbol{V}]$ and realizing that regularization terms have no effect as $\mathbf{W},\boldsymbol{U},\boldsymbol{V},\mathbf{L}$ are fixed, it becomes immediately clear. Similar arguments hold for the update equation of matrix $\mathbf{L}$.

Theorem 1 *The cost function $C\left(\mathbf{W}, \boldsymbol{U}, \boldsymbol{V}, \mathbf{H}, \mathbf{L}\right)$ is non-increasing under the alternating multiplicative update rules of Eqs. 6–9.*

Proof Since we are minimizing $C\left(\mathbf{W}_{ij}\right)$ using the auxiliary function $J\left(\mathbf{W}_{ij}, \mathbf{W}_{ij}^{(t)}\right)$. Therefore, evaluating $\nabla J\left(\mathbf{W}_{ij}, \mathbf{W}_{ij}^{(t)}\right) = 0$ and utilizing the results of Lemmas 1 and 2, we get the following update equation

$$\mathbf{W}_{ij}^{(t+1)} = \mathbf{W}_{ij}^{(t)} - \left[\nabla C_{ij}\left(\mathbf{W}_{ij}^{(t)}\right) / S_{ij}\left(\mathbf{W}_{ij}^{(t)}\right)\right]$$

Noting that

$$\nabla C_{ij}\left(\mathbf{W}_{ij}^{(t)}\right) = \Big[\lambda_{\boldsymbol{X}}\left(\boldsymbol{X}^{(t)} - \boldsymbol{X}\right)\mathbf{H}_w^{\mathsf{T}} + \lambda_{\boldsymbol{Y}}\left(\boldsymbol{Y}^{(t)} - \boldsymbol{Y}\right)\mathbf{L}_w^{\mathsf{T}} + \alpha \boldsymbol{U}\boldsymbol{U}^{\mathsf{T}}\mathbf{W}^{(t)} + \beta \boldsymbol{V}\boldsymbol{V}^{\mathsf{T}}\mathbf{W}^{(t)}\Big]_{ij}$$

and substituting $S_{ij}\left(\mathbf{W}_{ij}^{(t)}\right)$ from Lemma 2, we get the desired update of Eq. 6. □

3.2.4 Shared subspace dimensionality and algorithm complexity

Ideally, the shared subspace dimensionality should be learnt automatically using the two data sources. However, this presents a general model selection problem which is known to be hard. Nonetheless, since RJSNMF framework learns mutually orthogonal subspaces, a rough estimate of subspace dimensionalities can be easily obtained from the data. Since, we have $\mathbf{W}^{\mathsf{T}}\boldsymbol{U} \approx \mathbf{0}$, $\mathbf{W}^{\mathsf{T}}\boldsymbol{V} \approx \mathbf{0}$ and $\boldsymbol{U}^{\mathsf{T}}\boldsymbol{V} \approx \mathbf{0}$, shared subspace dimensionality is given by the rank of matrix $\boldsymbol{X}^{\mathsf{T}}\boldsymbol{Y}$, i.e. $K \approx \text{rank}\left(\boldsymbol{X}^{\mathsf{T}}\boldsymbol{Y}\right)$. Similarly, rough estimates of R_1 and R_2 are given by rank $(\boldsymbol{X})$ and rank $(\boldsymbol{Y})$ respectively. A detailed procedure to estimate these parameters is demonstrated in Sect. 6. Alternatively, a rule of thumb for deciding the dimensionality is proposed in Mardia et al. (1979). According to this, if the number of features common to the two data sets are equal to M_{12}, the shared subspace dimensionality is approximately equal to $\sqrt{M_{12}/2}$. As an example, considering two text corpora, M_{12} is given by number of common words between them.

When analyzing the computational complexity of RJSNMF algorithm, we refer to Eqs. 6–9. Computational complexity of learning matrix $\mathbf{W}$ is $\mathcal{O}\left(M \times N \times K\right)$ per iteration where $N = \max\left(N_1, N_2\right)$. Similarly, for each iteration, learning matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ takes $\mathcal{O}\left(M \times N_1 \times K_u\right)$ and $\mathcal{O}\left(M \times N_2 \times K_v\right)$ respectively where $K_u = R_1 - K$ and $K_v = R_2 - K$. Learning matrices $\mathbf{H}$ and $\mathbf{L}$ takes $\mathcal{O}\left(M \times N_1 \times R_1\right)$ and $\mathcal{O}\left(M \times N_2 \times R_2\right)$ operations per iteration. Therefore, overall complexity of the algorithm, dominated by computation of matrices $\mathbf{H}$ and $\mathbf{L}$, is $\mathcal{O}\left(M \times N \times R\right)$ where $R = \max\left(R_1, R_2\right)$. This is also the order of complexity of JSNMF algorithm (Gupta et al. 2010) and NMF algorithm (Lee and Seung 2001) for each iteration. Note that, for implementation efficiency, when computing matrices such as $\mathbf{F}^{\mathsf{T}}\mathbf{F}\mathbf{H}$, we should compute $\mathbf{F}^{\mathsf{T}}\mathbf{F}$ first and then multiply it to matrix $\mathbf{H}$. Similarly, when computing $\boldsymbol{U}\boldsymbol{U}^{\mathsf{T}}\mathbf{W}$, we should compute $\boldsymbol{U}^{\mathsf{T}}\mathbf{W}$ first and pre-multiply by $\boldsymbol{U}$. The main idea is to group the matrix multiplications by their inner-products first rather than their outer products.

4 Social media applications

Given related data sources, RJSNMF learns both the shared and the individual structures present in the data. This is desired in many applications such as multimedia indexing, text mining, computer vision and social media. For example, it might be of interest to determine the commonality among documents from computer science and biological science and at the same time find the differences. The commonality among the multiple data sources often implies the basic features across all the data sources while discriminating features can be utilized for classifying documents from one source against others. Another application is in social media, where there exist many popular social networking sites. Users of these websites upload and share content with one another. As an example, YouTube users upload videos related to some topics and Flickr users upload photos which may be similar to and/or different from YouTube videos. The data from different social media sources are semantically related due to the common cause of their creation (for example, in response to the real world events such as travel, oscars, olympics, wedding receptions, earthquakes etc). When modeled by RJSNMF, the shared features can be used to extract the basic tagging structures present across the two media (YouTube and Flickr) accurately by making it less subjective to either medium, and help improving the retrieval and clustering performance for each medium. In addition, the shared features can also be used to relate items from the two media using the shared subspace representation. While shared subspace across the related sources can boost the performance, the provision to maintain individual variations of each source ensures that domain-specific information is not lost.

In the rest of this section, we focus on the social media domain and show the usefulness of RJSNMF for two tasks (1) Improved social media retrieval by leveraging tags from auxiliary data sources. (2) Improved simultaneous clustering of related data sources and discovery of shared and individual clusters.

4.1 Improving social media retrieval using auxiliary sources

For social media retrieval, our key intuition is that tag ambiguities in a target domain are partially resolved by jointly modeling the tags of both the target and the auxiliary data. This is because any two data sources that share underlying structures when analyzed together, often provide richer information and have the potential to disambiguate the context for each other. RJSNMF exploits this aspect by joint modeling and is able to learn the shared structures of the two data sources. Continuing on from the RJSNMF framework described in the previous section, we use matrix $\boldsymbol{X}$ to denote *tf-idf* weighted (Salton and Buckley 1988) tag-item matrix (akin to the term-document matrices where the tag list of each media item is considered as a document with the tags as words) from the target data source and matrix $\boldsymbol{Y}$ to denote the counterpart from the auxiliary source. Using RJSNMF framework, we learn common subspace (spanned by matrix $\mathbf{W}$), individual subspaces (spanned by matrices $\boldsymbol{U}$ and $\boldsymbol{V}$) and the matrices **H** and **L** which contain the representations of the data $\boldsymbol{X}$ and $\boldsymbol{Y}$ in the learnt subspaces as given in Eqs. 1 and 2.

Given a set of query keywords S_q, we construct a query vector q by setting its elements to the *tf-idf* values at each index if the vocabulary contains a word from the keywords set S_q, otherwise setting them to zero. Next, we project the vector q onto the subspace (refer steps 4–6 in Algorithm 2) spanned by matrix $[\mathbf{W} \mid \boldsymbol{U}]$ to find its subspace representation, denoted by h. Now, for retrieval, cosine similarities are computed between the vector h and each column of the matrix **H** to find similarly tagged items from the target medium. Note that cosine similarity is often used when the magnitude of vectors is not an important factor while considering similarity. This is especially true in text based retrieval because two documents may be very similar due to the use of common words, irrespective of their document lengths. Generally, the set of query keywords has few words while the documents to be retrieved are much longer in length. Algorithm 2 provides details of the retrieval procedure.

Algorithm 2 Social media retrieval using RJSNMF.

```
1: Input: Target $\boldsymbol{X}$, auxiliary $\boldsymbol{Y}$, query vector $q$, number of items to be retrieved $N$.
2: learn matrices $\mathbf{W}, \boldsymbol{U}, \boldsymbol{V}, \mathbf{H}$ and $\mathbf{L}$ by following Algorithm 1.
3: set $\epsilon = 10^{-2}$, $\mathbf{F} \triangleq [\mathbf{W} \mid \boldsymbol{U}]$ and project $q$ onto $[\mathbf{W} \mid \boldsymbol{U}]$ (following Eq. 9) to get $h$ by an initialization
   and then looping as below
4: while $(t < \text{Maxiter})$ or $\left(\|\mathbf{F}h - q\|_2 \geq \epsilon\right)$ do
5:     $\left(h^{(t+1)}\right)_a \leftarrow \left(h^{(t)}\right)_a \left(\mathbf{F}^{\mathsf{T}} q\right)_a / \left(\mathbf{F}^{\mathsf{T}}\mathbf{F}h^{(t)}\right)_a$
6: end while
7: for each media item (indexed by $r$) in $\boldsymbol{X}$, with representation $h_r$ = $r$-th column of $\mathbf{H}$, compute its
   similarity with query projection $h$ as following
       $\text{CosSim}(h, h_r) = \frac{h^{\mathsf{T}} h_r}{\|h\|_2 \|h_r\|_2}$
8: Output: Return the top $N$ items in decreasing order of similarities.
```

4.2 Joint clustering of related data sources

Clustering is one of the most important problems in data mining and is considered to be an unsupervised task. Often, in real world applications, related data may arise from different sources and have significant variations in their distributions. Although these data sources have a great degree of commonality, direct clustering of the data combined together from these sources results in poor performance due to their individual differences. In such scenarios, the clustering method needs to deal with both shared and individual characteristics. RJSNMF is suitable for this joint clustering task.

Given two data sources, let us denote their data corpora by $\boldsymbol{X}$ and $\boldsymbol{Y}$. Using RJSNMF, we learn the common subspace **W** and individual subspaces $\boldsymbol{U}$ and $\boldsymbol{V}$. Representation of the two data matrices $\boldsymbol{X}$ and $\boldsymbol{Y}$ in the subspaces spanned by the columns of $[\mathbf{W} \mid \boldsymbol{U}]$ and $[\mathbf{W} \mid \boldsymbol{V}]$ is given by **H** and **L** respectively. Note that, originally, data points of $\boldsymbol{X}$

and $\boldsymbol{Y}$ are in M-dimensional space but RJSNMF projects them to a R-dimensional (note that dimensionality is reduced as $R < M$) space and thus partially avoids the problems such as the curse of high dimensionality and widely known problems in text mining such as polysemy and synonymy by discovering latent patterns in the data. After projecting the data points in the combined shared and individual subspaces, we use standard single task clustering methods such as K-means (or any other clustering method) to cluster the data in the regularized subspaces. We choose basic K-means with the purpose of emphasizing the clustering strength of subspaces and not of the clustering method per se. Algorithm 3 provides details of the RJSNMF based clustering.

Algorithm 3 RJSNMF based clustering.

```
1: Input: Target $\boldsymbol{X}$, auxiliary $\boldsymbol{Y}$, parameter set $\Psi$, convergence threshold $\epsilon$ and number of clusters $P$.
2: learn subspace matrices $\mathbf{W}$, $\boldsymbol{U}$, $\boldsymbol{V}$, $\mathbf{H}$ and $\mathbf{L}$ using Algorithm 1.
3: perform K-means clustering on matrix $\mathbf{H}$ considering its columns as subspace features for data $\boldsymbol{X}$ and denote the cluster membership vector as $m_{\boldsymbol{X}}$.
4: perform K-means clustering on matrix $\mathbf{L}$ considering its columns as subspace features for data $\boldsymbol{Y}$ and denote the cluster membership vector as $m_{\boldsymbol{Y}}$.
5: Output: Return cluster membership vectors $m_{\boldsymbol{X}}$, $m_{\boldsymbol{Y}}$ for data items in $\boldsymbol{X}$ and $\boldsymbol{Y}$ respectively.
```

5 Experiments

In this section, we demonstrate the effectiveness of RJSNMF for the social media retrieval and clustering applications as described in Sect. 4. We conduct two sets of experiments. In the first experiment set, we show the performance improvement for social media retrieval using auxiliary sources and clearly bring out the superiority of RJSNMF compared to other appropriate baselines. In the second set of experiments, we perform simultaneous clustering of related data sources. Through these experiments, we demonstrate the improvement in clustering performance achieved by simultaneous clustering and show that RJSNMF significantly outperforms other recently proposed single and multi-task clustering methods.

5.1 Experiment-I: social media retrieval

5.1.1 Data set

We conduct our experiments on a social media data set consisting of the textual tags of three social media genres: *blog*, *image* and *video*. To create the data set, three popular social media websites namely, Blogspot[5], Flickr[6] and YouTube[7] were used. To obtain the data, we first chose common concepts—‘Academy Awards’, ‘Australian

[5] http://www.blogger.com/.

[6] http://www.flickr.com/services/api/.

[7] http://code.google.com/apis/youtube/overview.html.

Table 1 Description of the Blogspot–Flickr–YouTube data set

Media genres	Concepts used for creating data set	Data set size (download/clean)	Avg. tags per item	Download information
Blogs	'Academy Awards', 'Australian Open', 'Olympic Games', 'US Election', 'Christmas', 'Cricket World Cup', 'Earthquake'	10000/7493	6	Performed a search within Blogspot.com website and crawled tags listed under the field "Labels".
Images	'Academy Awards', 'Australian Open', 'Olympic Games', 'US Election', 'Christmas', 'Holi', 'Terror Attacks'	20000/14113	8	Used Flickr Service APIs for downloading the tag metadata
Videos	'Academy Awards', 'Australian Open', 'Olympic Games', 'US Election', 'Earthquake', 'Global Warming', 'Terror Attacks'	7000/5007	7	Used Google code YouTube APIs for downloading the tag metadata

Second column provides the concepts used to download data items. Third column lists the number of data items downloaded from Web along with the number of data items remaining after removing those having less than 5 tags. Fourth column shows average number (rounded) of tags per item

Open', 'Olympic Games', 'US Election' and queried all three websites. To have some pairwise sharing, we additionally used the concept 'Christmas' to query Blogspot and Flickr, 'Terror Attacks' to query YouTube and Flickr, and 'Earthquake' to query Blogspot and YouTube. Lastly, to retain some differences between each medium, we used the concepts 'Cricket World Cup', 'Holi' and 'Global Warming' to query Blogspot.com, Flickr and YouTube respectively. The total number of unique tags combined from the three data sets were 3740. Further details of the data set are provided in Table 1. We refer to this data set as Blogspot–Flickr–YouTube data set. For each media genre, using their tag data, we generate a *tf-idf* weighted tag-item matrix (similar to the term-document matrix used in text analysis).

5.1.2 The baseline methods

To compare the performance with other methods, we choose three baselines:

- The first baseline performs retrieval by matching the query keyword with the tag-lists of each YouTube video without learning any subspace. To perform this matching, we use Jaccard[8] coefficient and rank the results based on these scores. This baseline is referred to as "Tag-based matching".
- In the second baseline, we take the tags of YouTube only (no auxiliary source is used) and apply standard NMF for retrieval. The number of basis vectors for NMF is set to 56. This baseline is referred to as "Standard NMF".

[8] Jaccard $(A, B) = |A \cap B| / |A \cup B|$.

- As a third baseline, we take the JSNMF framework proposed in Gupta et al. (2010) which is a special case of RJSNMF without any mutual orthogonality constraints on the learnt subspaces. This baseline is referred to as “JSNMF”.

To learn the subspace dimensionalities required for JSNMF and RJSNMF, we follow the procedure described in Sect. 6 and do cross-validation based on retrieval performance. The best performance was achieved by setting $R_Y = 56$, $R_F = 65$, $R_B = 62$ and $K_{YB} = 37$, $K_{YF} = 40$ where R_Y, R_F, R_B are total subspace dimensionalities of YouTube, Flickr and Blogspot data respectively and K_{YB}, K_{YF} are the shared subspace dimensionalities. To perform retrieval using the subspace based baseline methods (NMF and JSNMF), we use Algorithm 2 with a difference that subspace learning of RJSNMF is replaced by that of the respective method.

5.1.3 Evaluation metrics

For the purpose of evaluation, we define a query set {‘beach’, ‘america’, ‘bomb’, ‘animal’, ‘bank’, ‘movie’, ‘river’, ‘cable’, ‘climate’, ‘federer’, ‘disaster’, ‘elephant’, ‘europe’, ‘fire’, ‘festival’, ‘ice’, ‘obama’, ‘phone’, ‘santa’, ‘tsunami’} and denote it as Q. To evaluate retrieval methods, we use the popular *11-point precision recall curve* and *mean average precision* (*MAP*) criteria (Baeza-Yates and Ribeiro-Neto 1999). For social media retrieval, items relevant to user queries should be ranked high, as users would typically like every result on the first few pages to be as relevant as possible. *Precision-scope (P@N)* curve has been used as a ranking measure by many researchers (Rui and Huang 2000; Cai et al. 2007). Therefore, to evaluate the ranking performance of different methods clearly, we use the *P@N* curve. *P@N* curve calculates the average precision using only the top N retrieved items, and thus measures the ranking capability of a retrieval algorithm more effectively.

5.1.4 Experimental results

YouTube/Flickr retrieval results: Figure 1a depicts the YouTube video retrieval results and compares the performance of RJSNMF with the three baselines in terms of *11-point precision recall* curve, *P@N* curve and *MAP* evaluation criteria. It can be seen from Fig. 1 that RJSNMF clearly outperforms all the baselines in terms of all three evaluation criteria. Performance of the method based on tag matching using Jaccard similarity is poor due to the high dimensionality of input tag space and the ambiguities caused by polysemy and synonymy. Subspace learning methods (NMF, JSNMF and RJSNMF) overcome these problems to some extent by learning a reduced dimensional representation in latent space. NMF, being a subspace learning method, performs clearly better than tag-based matching but falls short when compared with JSNMF and RJSNMF. This is mainly due to the additional knowledge acquired from the auxiliary source, which helps in disambiguating tag co-occurrences. When we compare JSNMF with RJSNMF, we see that RJSNMF clearly performs better than JSNMF in terms of all evaluation criteria. This gain in performance can be attributed to the segregation of shared and individual subspaces which ensures the transference of useful knowledge only. Looking at the top 10 results, we see that RJSNMF

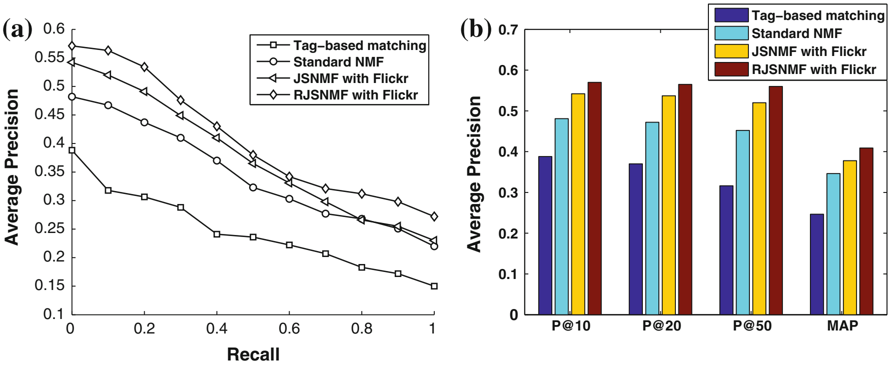


Fig. 1 YouTube video retrieval results using tags of Flickr as auxiliary source: **a** 11-point interpolated precision-recall, **b** precision-scope (P@N) and MAP; for tag-based matching (baseline 1), standard NMF (baseline 2), JSNMF (Gupta et al. 2010) (baseline 3) and the proposed RJSNMF

achieves around 57% precision as opposed to 54%, 48% and 39% precisions achieved by JSNMF, NMF and tag-based matching methods.

YouTube/Blogspot retrieval results: When we use Blogspot data as the auxiliary source, the video retrieval results from YouTube follow similar trends. RJSNMF performs the best in terms of all three evaluation criteria followed by JSNMF, NMF and tag-based matching respectively. When looking at the top 10 results, we see that RJSNMF achieves around 56% precision as opposed to 52% precision achieved by JSNMF. The performance of NMF and tag-based matching methods remains the same as they don't use any auxiliary data.

It is interesting to note from Figs. 1 and 2 that YouTube retrieval performance benefits slightly more from Flickr data than Blogspot data. This gain in performance could be due to the fact that the Flickr data set has more data points than the Blogspot data set. The Second reason which may explain this performance gain is that the tags

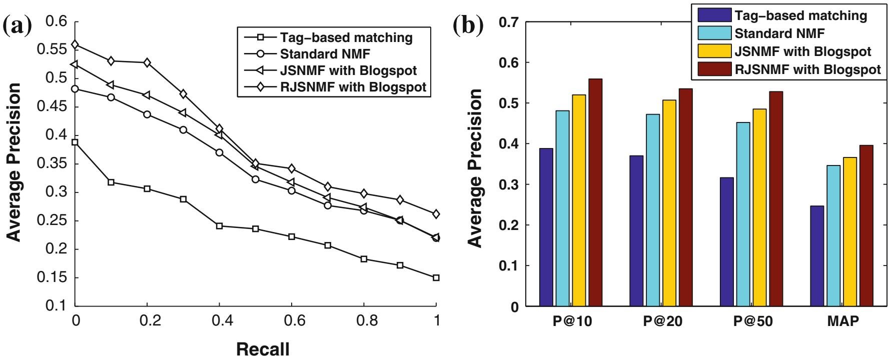


Fig. 2 YouTube video retrieval results using tags of Blogspot as auxiliary source: **a** 11-point interpolated precision-recall, **b** precision-scope (P@N) and MAP; for tag-based matching (baseline 1), standard NMF (baseline 2), JSNMF (Gupta et al. 2010) (baseline 3) and the proposed RJSNMF

Table 2 Description of the CNN–BBC data set

News channel	Feeds used for creating data set	No. of articles
CNN	'Politics', 'Crime', 'Health', 'Living', 'Showbiz', 'Tech', 'Top Stories', 'Travel', 'US', 'World', 'Money Latest', 'Sports Illustrated (SI)'	4593
BBC	'Business', 'Entertainment/Arts', 'Health', 'Miscellaneous', 'Science/Environment', 'Technology', 'Sport', 'World–US–Canada'	7612

The second column shows the news feeds crawled from the two news channels' websites. The third column shows the total number of articles downloaded from each channel

attached to Flickr images may be closer in semantics to the YouTube video tags than the Blogspot tags. However, further experiments on this aspect are beyond the scope of this paper.

5.2 Experiment-II: clustering social media and news articles

In this subsection, we further demonstrate the usefulness of our framework for clustering applications. Beyond the traditional setting of a standard clustering algorithm, where often only one cluster partition is returned, our proposed framework performs simultaneous clustering of the related data sources and provides common and individual clusters. This directly implies the automatic discovery of common and individual topics[9] (or stories) across related data sources.

5.2.1 Data sets

For the clustering experiments, we use three data sets. The first data set is the same social media data set that we used for the retrieval experiments described in Sect. 5.1. The second data set is created by crawling news articles from CNN and BBC feeds.[10] Both news sources cover many of the same high profile real world events, but they also cover events relevant to their particular geographical focus. To have reasonable commonality as well as differences between the two data sets, we chose to crawl mostly similar news feeds from CNN and BBC along with some different feeds. Table 2 provides details about the selected feeds and the number of news articles for each news channel. We refer to this data set as CNN–BBC data set. For each channel, using their news articles, we generate a *tf-idf* weighted term-document matrix.

Our third data set is the 20 Newsgroup benchmark data set widely used for evaluating clustering methods. We mainly use this data set to compare the performance of RJSNMF with other state-of-the-art techniques in single as well as multi-task

[9] Our shared and individual subspaces (represented by matrices $\mathbf{W}$, $\boldsymbol{U}$ and $\boldsymbol{V}$) directly provide common and individual topics. Alternatively, one can use standard topic models e.g. Latent Dirichlet Allocation (LDA) on the documents of common and individual clusters.

[10] All CNN and BBC feeds were accessed in August 2010.

clustering. The details of this data set and the experiments based on it are provided in Sect. 5.3.

5.2.2 *The baseline methods*

To demonstrate the advantages of simultaneous clustering, we compare our proposed RJSNMF with the following baseline methods

- The first baseline is a subspace based clustering which uses NMF in conjunction with K-means (similar to the work in Xu et al. 2003). We refer to it as "NMF+KM". Since RJSNMF is based on NMF, improvement on this baseline directly shows the benefits of simultaneous clustering using related auxiliary sources.
- The second baseline is the recently proposed JSNMF (Gupta et al. 2010) algorithm which can be combined with K-means similar to the proposed RJSNMF. This baseline is referred to as "JSNMF+KM". This baseline is chosen to show how additional mutual orthogonality constraints of RJSNMF clearly segregate the common and individual subspaces and help in avoiding negative transfer learning.
- The third baseline is another subspace based clustering which uses PCA in conjunction with K-means. We refer to it as "PCA+KM". This baseline is chosen to show the benefits of nonnegative shared subspace learning over mixed-sign subspace techniques for document clustering. We also adapt PCA to perform joint clustering by augmenting the data from both sources together and use Kmeans for clustering in the learnt subspace. We refer to this baseline as "Aug-PCA+KM".
- The fourth baseline is the state-of-the-art multi-task clustering technique proposed in Gu and Zhou (2009a) which performs joint clustering of multiple data sources as a linear combination of both original input space and a shared subspace. The linear combination is controlled using a parameter λ. We refer to this method as "LSSMTC". When the parameter $\lambda = 0$, this method reduces to Adaptive Subspace Iteration (ASI) proposed in Li et al. (2004). ASI can be used in two versions—separately clustering each source or jointly clustering the two sources by augmentine the data from the two sources. The two versions are referred to as "ASI" and "Aug-ASI" respectively.

5.2.3 *Evaluation metrics*

We evaluate the clustering performance using four well-known metrics : accuracy (*AC*), normalized mutual information (*NMI*), average cluster entropy (*ACE*), cluster purity (*CP*).

Accuracy Given a data point y_k, let c_k and z_k denote the induced cluster label and the ground-truth category labels respectively. The accuracy (AC) is defined as

$$AC = \frac{\sum_{k=1}^{n} \delta\left(\operatorname{map}\left(c_k\right), z_k\right)}{n} \tag{10}$$

where n denotes the total number of data points, $\delta\left(a, b\right)$ is the delta function that equals one if $a = b$ and zero otherwise and $\operatorname{map}\left(c_k\right)$ is the mapping function that maps each cluster label c_k to the most likely category from the data set. In our experiments, we used the popular Hungarian algorithm (Lovász and Plummer 1986) to implement the mapping function similar to the previous works (Xu et al. 2003; Gu and Zhou 2009a). The higher the AC, the better is the clustering result.

Normalized mutual information Given induced partition $\mathcal{C}$ with P labels $c_1, \ldots, c_P$ and true partition $\mathcal{T}$ with Q ground-truth category labels $z_1, \ldots, z_Q$, normalized mutual information (NMI) is defined as

$$NMI = \frac{\sum_{p,q} n_{p,q} \log \frac{n_{p,q}}{n_p^{\mathcal{C}} n_q^{\mathcal{T}}}}{\sqrt{\left(\sum_p n_p^{\mathcal{C}} \log \frac{n_p^{\mathcal{C}}}{n^{\mathcal{C}}}\right)\left(\sum_q n_q^{\mathcal{T}} \log \frac{n_q^{\mathcal{T}}}{n^{\mathcal{T}}}\right)}} \tag{11}$$

where $n_p^{\mathcal{C}}, n_q^{\mathcal{T}}$ denote the number of data points in p-th cluster from partition $\mathcal{C}$ and q-th cluster from partition $\mathcal{T}$ respectively and $n_{p,q}$ denotes the number of common data points between p-th cluster from partition $\mathcal{C}$ and q-th cluster from partition $\mathcal{T}$. The higher the NMI, the better is the clustering result.

Average cluster entropy Given induced partition $\mathcal{C}$ and true partition $\mathcal{T}$, entropy of j-th induced cluster is defined as

$$E_j = -\sum_i p_{ij} \log p_{ij}$$

where p_{ij} is the probability that a member in j-th induced cluster belongs to the ground-truth category i. Given entropy of clusters (indexed by $j = 1, \ldots, P$) in induced partition $\mathcal{C}$, average cluster entropy (ACE) is defined as follows

$$ACE = \frac{\sum_{j \in \mathcal{C}} n_j E_j}{n}$$

where n_j is the number of data points in j-th induced cluster and n is the total number of data points. To define the average cluster entropy for ground-truth category i, let $\mathcal{C}_i$ denote the subset of clusters in partition $\mathcal{C}$ such that the majority of data points belong to category i; formally, define

$$\mathcal{C}_i \triangleq \{c_j \mid p_{ij} \geq p_{kj}, k \neq i\}$$

and $N_i \triangleq \sum_{j \in \mathcal{C}_i} n_j$ then ACE_i is given as

$$ACE_i = \frac{\sum_{j \in \mathcal{C}_i} n_j E_j}{N_i}$$

Lower the ACE and ACE_i, better is the clustering result.

Cluster purity Similar to entropy definition, we define purity of clusters as a whole and also for each ground-truth category. As a definition, for j-th induced cluster, the major category is i if $p_{ij} \geq p_{kj}, \forall k \neq i$. Now, if, for j-th induced cluster, m_j denotes the number of points from the major category, the purity (Manning et al. 2008) of cluster partition $\mathcal{C}$ is defined as

$$CP = \frac{\sum_{j \in \mathcal{C}} m_j}{n}$$

Again, n is the total number of data points. Cluster purity for ground-truth category i is defined as

$$CP_i = \frac{\sum_{j \in \mathcal{C}_i} m_j}{N_i}$$

where definition of $\mathcal{C}_i$ and N_i remains same as for average cluster entropy. Higher the CP and CP_i, better is the clustering result.

We follow an arrow notation such that symbol ↑ denotes that the performance in terms of an evaluation measure is *better* if its value is *higher*. Similarly, the symbol ↓ denotes that the performance in terms of an evaluation measure is *better* if its value is *lower*.

5.2.4 Experimental results

Clustering based on the proposed RJSNMF first uses Algorithm 1 for learning shared subspaces and then Algorithm 3. Clustering based on NMF (or JSNMF) can be carried out in a similar manner where we replace Algorithm 1 with the subspace learning algorithm based on NMF (or JSNMF).

Blogspot–Flickr clustering results: Using the clustering results on the Blogspot/Flickr data set, we provide a detailed analysis of how shared subspace representing the common knowledge of the two data sources helps to improve the clustering task (JSNMF vs NMF and RJSNMF vs NMF). We also present the comparison between JSNMF and RJSNMF by performing clustering using only shared, only individual and combined subspaces. This provides some intuition on how the regularization scheme applied in RJSNMF helps in segregating common and individual subspaces, and thus provides significant improvements in clustering performance. Following the strategy as explained in Sect. 6, subspace dimensionalities were set to the following values: $R_B = 62$, $R_F = 65$ and $K_{BF} = 43$ where R_B and R_F are the total subspace dimensionalities of Blogspot and Flickr data respectively and K_{BF} is the dimensionality of their shared subspace. The total number of clusters were set to 7 (equal to the true number of categories).

Since the subspace dimensionalities are chosen based on rank estimation, the same dimensionalities were also used for PCA+KM. The best performance for Aug-PCA+KM method was found by setting the number of basis vectors to 93. Since LSSMTC, Aug-PCA+KM and Aug-ASI require an equal number of clusters for each source, we set this value to 9, as the total number of different categories across the two data sets is equal to 9. The best performance of LSSMTC was achieved at $\lambda = 0.2$ and $l = 12$ where l denotes the dimensionality of shared subspace used in this method.

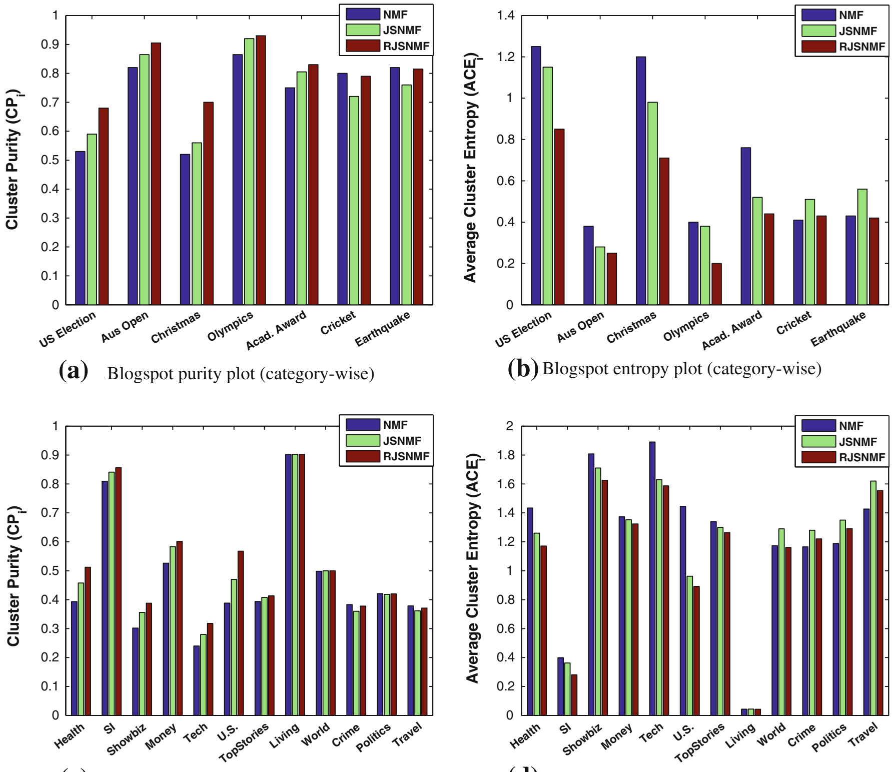
Table 3 provides clustering results for Blogspot data using Flickr as the auxiliary source and presents a comparison of RJSNMF+KM with the baselines in terms of average clustering entropy (ACE), cluster purity (CP), cluster accuracy (AC) and normalized mutual information (NMI). When looking at the results obtained using single source clustering methods (NMF+KM, PCA+KM and ASI+KM), we see that NMF performs better than both PCA and ASI. The similar superiority of NMF has also been reported in Xu et al. (2003). When comparing the joint clustering methods, Aug-ASI performs slightly better than Aug-PCA+KM. Nonetheless, the performance of both the augmented methods fall short compared to the shared subspace learning methods. Out of the shared subspace methods, we see that although LSSMTC performs better than JSNMF+KM, it is outperformed by RJSNMF+KM in terms of all evaluation metrics.

To investigate this performance improvement further, we compute the ACE and CP values at the cluster category level. Figure 3 (the first row) depicts ACE and CP values for each category. It is interesting to note from Fig. 3a and b that RJSNMF+KM performs significantly better than NMF+KM for all but ‘Cricket World Cup’ and ‘Earthquake’ concepts (or categories) which are *individual* to the Blogspot data set (refer Table 1). The concepts ‘Cricket World Cup’ and ‘Earthquake’ are not available in the Flickr data set and therefore, combined learning with Flickr data does not help in improving clustering performance for these concepts. Interestingly, there is not much negative transfer in the RJSNMF+KM case which happens to occur in the case of JSNMF+KM whose performance clearly degrades for these two concepts. At the same time, performance improvement achieved by JSNMF+KM over NMF+KM is minimal as it is unable to clearly segregate the common and individual subspaces. As a result, JSNMF+KM is neither able to exploit the mutual knowledge for related concepts nor is it able to avoid negative transfer for domain specific concepts.

Table 4 presents the clustering results for the Flickr data set (task-2) using Blogspot as the auxiliary source and provides a comparison of RJSNMF+KM with the

Table 3 Summary of entropy, purity, accuracy and NMI values for Blogspot clusters with knowledge transfer from Flickr tags

Blogspot (task-1)				
Method	Avg. cluster entropy (↓)	Cluster purity (↑)	Accuracy (↑)	NMI (↑)
NMF+KM	0.69 ± 0.01	0.73 ± 0.08	0.68 ± 0.01	0.40 ± 0.02
PCA+KM	0.78 ± 0.07	0.62 ± 0.11	0.63 ± 0.08	0.38 ± 0.06
Aug-PCA+KM	0.81 ± 0.13	0.58 ± 0.05	0.61 ± 0.12	0.37 ± 0.08
ASI	0.67 ± 0.06	0.74 ± 0.03	0.69 ± 0.05	0.41 ± 0.03
Aug-ASI	0.79 ± 0.06	0.63 ± 0.03	0.62 ± 0.01	0.37 ± 0.05
LSSMTC	0.66 ± 0.07	0.78 ± 0.03	0.71 ± 0.08	0.43 ± 0.02
JSNMF+KM	0.63 ± 0.08	0.75 ± 0.07	0.70 ± 0.00	0.41 ± 0.01
RJSNMF+KM	$\mathbf{0.47 \pm 0.03}$	$\mathbf{0.81 \pm 0.05}$	$\mathbf{0.78 \pm 0.02}$	$\mathbf{0.44 \pm 0.05}$



(a) Blogspot purity plot (category-wise)

(b) Blogspot entropy plot (category-wise)

(c) CNN purity plot (category-wise)

(d) CNN entropy plot (category-wise)

Fig. 3 Category-wise purity/entropy plots depicting cluster quality for Blogspot–Flickr and CNN–BBC data sets. The first row (**a**, **b**) depicts the clustering results of Blogspot articles using Flickr as the auxiliary source. The second row (**c**, **d**) depicts the clustering results of CNN news articles using BBC as the auxiliary source. The proposed RJSNMF is compared with NMF (baseline 1) and JSNMF (Gupta et al. 2010) (baseline 2) to show the benefits of transfer learning using the cluster purity (CP) and the average cluster entropy (ACE) measures

Table 4 Summary of entropy, purity, accuracy and NMI values for Flickr clusters with knowledge transfer from Blogspot tags

Flickr (task-2)				
Method	Avg. cluster entropy (↓)	Cluster purity (↑)	Accuracy (↑)	NMI (↑)
NMF+KM	0.74 ± 0.01	0.66 ± 0.03	0.64 ± 0.01	0.39 ± 0.03
PCA+KM	0.78 ± 0.08	0.65 ± 0.07	0.63 ± 0.05	0.37 ± 0.06
Aug-PCA+KM	0.79 ± 0.10	0.64 ± 0.09	0.64 ± 0.08	0.37 ± 0.09
ASI	0.83 ± 0.06	0.58 ± 0.04	0.53 ± 0.05	0.33 ± 0.02
Aug-ASI	0.87 ± 0.05	0.59 ± 0.03	0.54 ± 0.07	0.34 ± 0.01
LSSMTC	0.89 ± 0.04	0.54 ± 0.03	0.52 ± 0.06	0.32 ± 0.05
JSNMF+KM	0.72 ± 0.02	0.69 ± 0.06	0.67 ± 0.05	0.41 ± 0.04
RJSNMF+KM	**0.68 ± 0.03**	**0.72 ± 0.05**	**0.71 ± 0.01**	**0.45 ± 0.07**

two baselines. As it can be seen from the table, RJSNMF+KM remains to be the best method compared to all the baselines. Moreover, in this case, both ASI and LSSMTC are outperformed by all other methods. This degradation in performance can be attributed to the fact that ASI does not use any individual subspace. Similarly, the performance degradations of LSSMTC can be attributed to its limitation of requiring common cluster centroids for both the data sets, and thus not able to model the individual variations.

CNN–BBC news clustering results: Tables 5 and 6 present the joint clustering results on CNN (task-1; using BBC data as auxiliary source) and BBC (task-2; using CNN data as auxiliary source) news articles. To perform clustering, the total number of clusters were set to 12 and 8 (equal to the number of categories in the data set) for CNN and BBC respectively. To learn the subspaces, we used the following set of parameters: $R_{CNN} = 50$, $R_{BBC} = 64$ and $K_{CB} = 34$ where R_{CNN} and R_{BBC} are

Table 5 Summary of entropy, purity, accuracy and NMI values for CNN clusters with knowledge transfer from BBC tags

CNN (task-1)				
Method	Avg. cluster entropy (↓)	Cluster purity (↑)	Accuracy (↑)	NMI (↑)
NMF+KM	1.22 ± 0.08	0.47 ± 0.02	0.42 ± 0.03	0.27 ± 0.01
PCA+KM	1.26 ± 0.07	0.46 ± 0.06	0.44 ± 0.07	0.28 ± 0.03
Aug-PCA+KM	1.29 ± 0.08	0.45 ± 0.08	0.45 ± 0.03	0.29 ± 0.09
ASI	1.20 ± 0.03	0.49 ± 0.07	0.49 ± 0.05	0.31 ± 0.03
Aug-ASI	1.25 ± 0.06	0.47 ± 0.05	0.48 ± 0.03	0.30 ± 0.02
LSSMTC	1.20 ± 0.09	0.48 ± 0.12	0.47 ± 0.08	0.30 ± 0.06
JSNMF+KM	1.18 ± 0.04	0.49 ± 0.07	0.46 ± 0.02	0.29 ± 0.00
RJSNMF+KM	**1.12 ± 0.06**	**0.52 ± 0.04**	**0.51 ± 0.03**	**0.32 ± 0.07**

Table 6 Summary of entropy, purity, accuracy and NMI values for BBC clusters with knowledge transfer from CNN tags

BBC (task-2)				
Method	Avg. cluster entropy (↓)	Cluster purity (↑)	Accuracy (↑)	NMI (↑)
NMF+KM	1.23 ± 0.03	0.41 ± 0.00	0.39 ± 0.04	0.18 ± 0.02
PCA+KM	1.28 ± 0.07	0.38 ± 0.05	0.39 ± 0.05	0.17 ± 0.01
Aug-PCA+KM	1.32 ± 0.06	0.36 ± 0.02	0.39 ± 0.03	0.17 ± 0.08
ASI	1.43 ± 0.07	0.34 ± 0.06	0.33 ± 0.04	0.14 ± 0.03
Aug-ASI	1.45 ± 0.09	0.33 ± 0.08	0.34 ± 0.03	0.15 ± 0.05
LSSMTC	1.33 ± 0.05	0.36 ± 0.03	0.38 ± 0.04	0.17 ± 0.03
JSNMF+KM	1.20 ± 0.05	0.40 ± 0.02	0.37 ± 0.01	0.16 ± 0.04
RJSNMF+KM	**1.17 ± 0.04**	**0.44 ± 0.03**	**0.40 ± 0.04**	**0.20 ± 0.02**

the total subspace dimensionalities of CNN and BBC data respectively, and K_{CB} is the dimensionality of their shared subspace.

Parameter selection for other baselines remains similar to the Blogspot–Flickr experiments. The best performance for the Aug-PCA+KM method, in this case, was found by setting the number of basis vectors to 78. Again, since LSSMTC, Aug-PCA+KM and Aug-ASI require an equal number of clusters for each sources, we set this value to 13 as the total number of categories in CNN and BBC are 12 and 8 respectively and most of the BBC categories (except the 'miscellaneous' category) are similar to those in CNN. The best performance for LSSMTC was obtained at $\lambda = 0.3$ and $l = 16$.

It can be seen from Table 5 that the clustering results of CNN show similar behavior with respect to the different methods as seen for Blogspot–Flickr data set. RJSNMF+KM clearly outperforms all other methods including JSNMF+KM, ASI, Aug-ASI, PCA+KM, Aug-PCA+KM and LSSMTC. Figure 3 (the second row) depicts the category-wise purity and entropy values for clustering using RJSNMF and JSNMF and compares them with NMF based clustering. Looking at the ACE and CP values in Fig. 3c and d, it is interesting to note that RJSNMF clearly exploits the auxiliary data of BBC channel and improves the performance for all categories *except* 'World', 'Crime', 'Politics' and 'Travel'. This is along the lines of our intuition, as out of these four categories, we do not expect the articles of 'Crime' and 'Travel' to be similar for CNN and BBC as 'Crime' related news articles are usually geographically local and 'Travel' related articles are random in nature. Similarly, there seems to be diversity among the categories 'World' and 'Politics' and the two channels need not cover similar stories.

The clustering results on BBC data set (task-2) is presented in Table 6 in the same fashion as the results of CNN (task-1). However, these results are *more interesting* due to the clear demonstration of JSNMF+KM being affected by *negative transfer learning* while RJSNMF+KM still retains the benefits of auxiliary information. Note from Table 6 that the performance of JSNMF+KM degrades compared to NMF+KM in terms of each evaluation criteria except ACE (still slightly better than NMF) which shows that JSNMF+KM is susceptible to negative transfer learning. This happens because the overall clustering performance for BBC data is poor due to wide variations in the data and JSNMF find it difficult to segregate the commonalities and differences of BBC and CNN articles. Regularization used by RJSNMF stops any CNN specific patterns from entering into shared subspace and therefore transfers only useful knowledge. In addition, RJSNMF+KM clearly outperforms LSSMTC, PCA+KM, ASI and the augmented methods. Performance of ASI and Aug-ASI has degraded heavily followed by Aug-PCA+KM and PCA+KM. Although LSSMTC performs slightly better than JSNMF+KM, its performance remains lower than RJSNMF+KM.

5.2.5 *Common and individual clusters*

Explicit learning of shared and individual subspaces enables us to distinguish between the clusters which are related and different in the two data sources. For clarity, we explain this procedure for Blogspot–Flickr data set only, similar explanation holds for the CNN–BBC data set. However, we provide results for both data sets. After learning the shared and individual subspaces for the Blogspot–Flickr data set, we use them to cluster the Blogspot data. Instead of using the full subspace, we first use only shared subspace representation ($\mathbf{H}_w$) of the Blogspot data and cluster them to get a partition of the data using K-means. This is referred to as "RJSNMF (shared)" or "JSNMF (shared)". Then, we use only individual subspace representation ($\mathbf{H}_u$) of Blogspot data and cluster them again to get another partition of the data using K-means. This is referred to as "RJSNMF (individual)" or "JSNMF (individual)". We compute the purity and entropy values for each induced cluster in both data partitions. Comparing the performance of RJSNMF with JSNMF, these results are shown in Fig. 4. It can be clearly seen from Fig. 4 that categories which are similar in the Blogspot and Flickr data sets are clustered well using shared subspace representation whereas the data from Blogspot specific categories are clustered well using the individual subspace representation. Building upon this idea, we further compute the ratio of the purity values (and similarly, the ratio of entropy values) for the respective clusters in the two partitions. Figure 5a depicts the purity ratio plot for Blogspot clusters. It can be seen from Fig. 5b that purity ratio clearly remains *higher than one* for the clusters which are of *common* category between Blogspot and Flickr data set. Similarly, purity ratio is always *lower than one* for clusters which are *specific* to Blogspot data set.

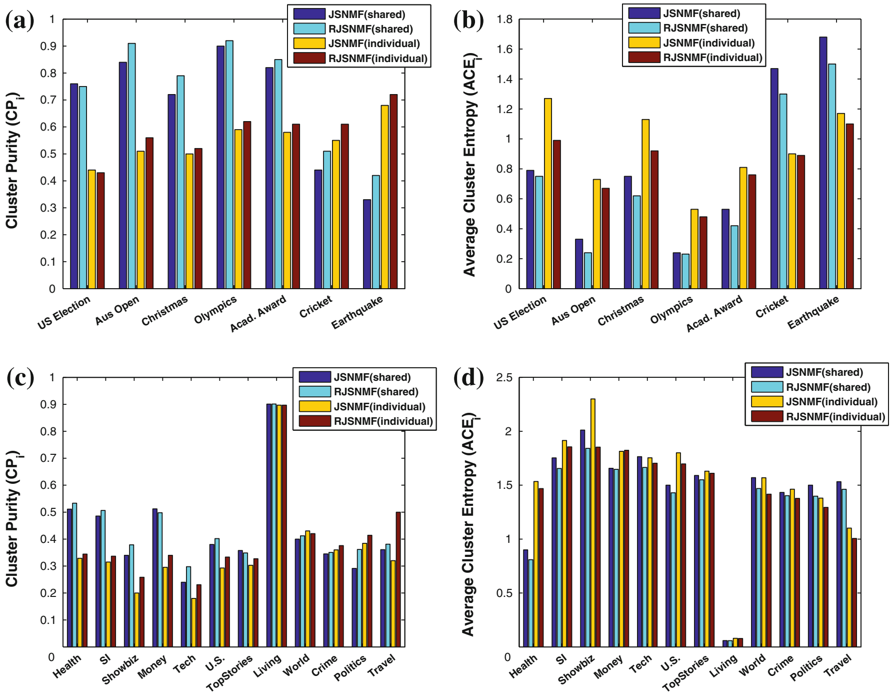
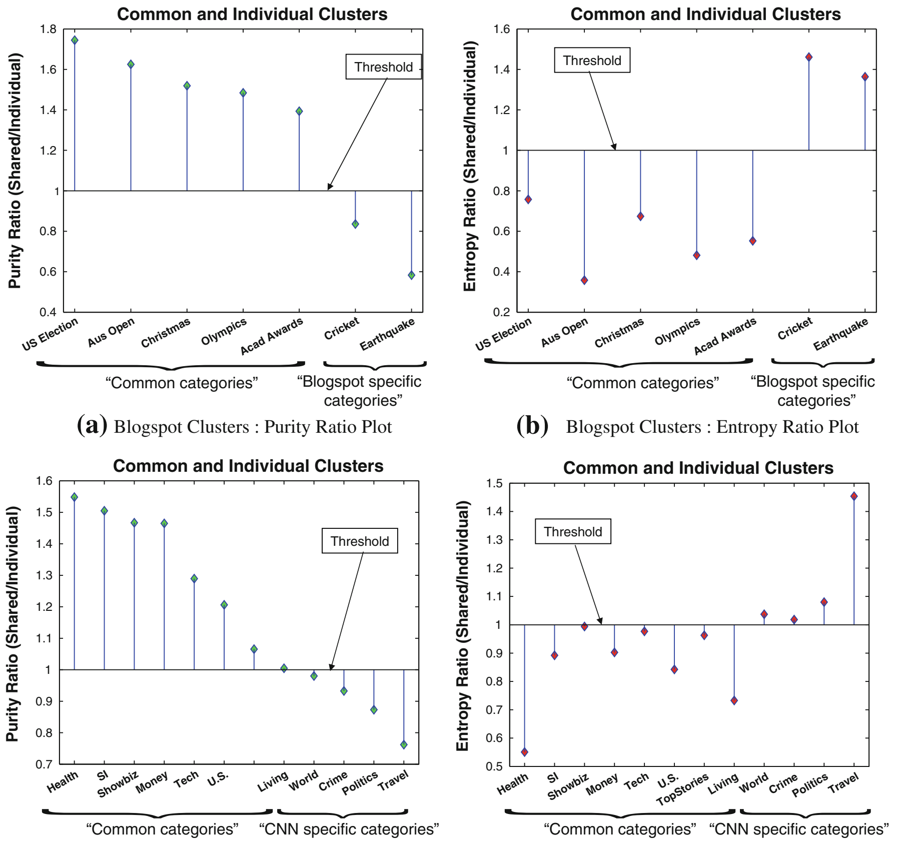


Fig. 4 Comparison of RJSNMF and JSNMF category-wise clustering results using "shared subspace only representation" ($\mathbf{H}_w$) and "individual subspace only representation" ($\mathbf{H}_u$). Using the two representations, clustering results for Blogspot data (with Flickr as auxiliary) are shown in terms of **a** cluster purity and **b** average cluster entropy. Similar results for CNN data (with BBC as auxiliary) are shown in (**c**, **d**)



(a) Blogspot Clusters : Purity Ratio Plot

(b) Blogspot Clusters : Entropy Ratio Plot

(c) CNN Clusters : Purity Ratio Plot

(d) CNN Clusters : Entropy Ratio Plot

Fig. 5 Plots depicting the ratio of category level purities (and entropies) using shared subspace only representation ($\mathbf{H}_w$) and individual subspace only representation ($\mathbf{H}_u$). **a** For a category, purity ratio greater than 1 indicates that the corresponding news feed is common to both Blogspot and Flickr whereas purity ratio less than 1 indicates that the corresponding news feed is specific to Blogspot only. **b** Similar inference can be made using entropy ratio showing opposite behavior. The corresponding results for CNN–BBC data set are shown in (**c**, **d**)

An inverse behavior can be seen from Fig. 5b depicting entropy ratio for each category. For common categories, a value of cluster purity ratio higher than one *implies* that clusters obtained using shared subspace representation are purer than those obtained using individual subspace representation. We note that this phenomenon is due to the clear segregation of shared and individual subspaces. As a result, the clusters which are from the common categories of the two data sets are clustered well (with high purity and low entropy) in shared subspace and not so well (with low purity and high entropy) in individual subspace. Exactly opposite happens to the clusters of specific categories to Blogspot data which are clustered not so well in shared subspace and well in individual subspace. Similar results are obtained for CNN–BBC data set and shown in Fig. 5c and d.

5.3 Comparison using 20 newsgroup benchmark data set

5.3.1 20 Newsgroup data set

20-Newsgroup[11] is one of the most widely used data set for both single domain clustering and cross-domain learning (Xu et al. 2003; Cai et al. 2008; Gu and Zhou 2009a). It is a collection of 20000 newsgroup documents, almost evenly divided into 20 categories. We use this data set to compare with a variety of single-domain clustering techniques and a recent multi-task clustering method proposed in Gu and Zhou (2009a). To have a fair comparison, we create a multi-task clustering (having two tasks) data set from 20 Newsgroup documents in the same fashion as in Gu and Zhou (2009a), and use this data set (we'll refer to it as Rec-Talk as in Gu and Zhou 2009a) for evaluating multi-task clustering capabilities of RJSNMF model. For the first task, we include the documents from "rec.autos" and "talk.politics.guns" categories while, for the second task, we use the documents from "rec.sport.baseball" and "talk.politics.mideast" categories. This type of splitting ensures that the two tasks are related, yet different, due to being constructed from the same categories but different sub-categories.

5.3.2 Baseline methods

We show the superiority of RJSNMF to several single-task clustering methods such as K-means (KM) in input data space, K-means followed by Principal Component Analysis (PCA+KM), Normalized Cut (Ncut) (Shi and Malik 2000), K-means followed by Nonnegative Matrix Factorization (NMF+KM) (Xu et al. 2003), adaptive subspace iteration (ASI) (Li et al. 2004) and a multi-task clustering method LSSMTC (Gu and Zhou 2009a). Single-domain clustering algorithms are applied on the 20 Newsgroup Rec-Talk data set in two ways (1) apply a single-domain clustering algorithm on the two tasks independently (2) apply a single-domain clustering algorithm on the two tasks by merging the data from both the tasks. Therefore, every single-domain clustering task has two versions. For example, the two versions for K-means algorithm are referred to as "KM" and "All KM". Similarly, K-means followed by PCA are referred to as "PCA+KM" and "All PCA+KM" and so on. LSSMTC is a recent multi-task clustering algorithm closely related with our work with the difference that learning the shared subspace is carried out in an entirely different manner. Using Rec-Talk data set, we compare our results with LSSMTC only as LSSMTC has already been shown to outperform other baselines (comparison with NMF is not carried out) in Gu and Zhou (2009a). We use an *identical* setting as used in Gu and Zhou (2009a) so that a comparison can be made with NMF+KM and the proposed RJSNMF. In addition, we also compare RJSNMF with JSNMF (a special case of RJSNMF model without any regularization) proposed recently in Gupta et al. (2010) to clearly show the benefits of regularization.

[11] http://people.csail.mit.edu/jrennie/20Newsgroups/.

5.3.3 20 Newsgroup clustering results

To generate clustering results based on RJSNMF, we use Algorithms 1 (with parameters $R_1 = R_2 = 30$ and $K = 18$; learnt using the procedure described in Sect. 6) followed by Algorithm 3. For NMF and JSNMF, instead of Algorithm 1, we use equivalent subspace learning algorithm and then use K-means as in Algorithm 3. Table 7 presents the comparison of RJSNMF with the above-mentioned baselines (JSNMF, LSSMTC and various single-task clustering methods) using Rec-Talk data set. For the experiment, the number of clusters for each task were set to 2 as in Gu and Zhou (2009a). Since the algorithms are iterative, each algorithm was run 50 times and we report the mean value along with standard deviations. The clustering results for LSSMTC were reported in Gu and Zhou (2009a) under identical settings. We use those results here for a comparison with other NMF based methods.

It can be seen from Table 7 that all three shared subspace learning methods (LSSMTC, JSNMF+KM and RJSNMF+KM) clearly outperform NMF+KM for Rec data set (task-1) whereas only RJSNMF outperforms NMF+KM for Talk data set (task-2). This improvement in performance stems from the ability of RJSNMF to exploit the knowledge available in auxiliary domains and transfer it to the target task appropriately without any negative transfer learning. Note that when single-task clustering algorithms are used on the combined data of task-1 and task-2, they do not necessarily perform better since the data distribution differs between the two tasks and simply combining them may lead to even negative knowledge transfer. This can be clearly seen in the case of Aug-NMF+KM results for Talk data set. When comparing shared subspace learning methods, we can see that RJSNMF+KM (and JSNMF+KM except on task-1) outperforms LSSMTC. This is because, although LSSMTC learns a shared subspace to exploit the common knowledge between the two tasks, it forces common centroids for each task. This is too restrictive to model real world data because, in spite of sharing a subspace, the two data sources would usually have different coordinates in the shared subspace and their clusters need not share the cluster centroids. When comparing the results of RJSNMF and JSNMF, we see that RJSNMF clearly outperforms JSNMF. This is due to the regularization scheme imposed through RJSNMF formulation. Through regularization, RJSNMF tries to get subspaces corresponding to $\mathbf{W}$,$\boldsymbol{U}$ and $\boldsymbol{V}$ matrices such that they are mutually disjoint. This separates the common

Table 7 Comparison with state-of-the-art single and multi-task clustering methods using Rec-Talk (20 Newsgroup) benchmark data set

	Rec (task-1)		Talk (task-2)	
Method	Accuracy (↑)	NMI (↑)	Accuracy (↑)	NMI (↑)
NMF+KM	0.6047 ± 0.13	0.4111 ± 0.05	0.8791 ± 0.14	0.6030 ± 0.08
Aug-NMF+KM	0.6421 ± 0.10	0.4232 ± 0.06	0.7636 ± 0.10	0.5812 ± 0.05
LSSMTC	0.8433 ± 0.08	0.4306 ± 0.06	0.7895 ± 0.08	0.3473 ± 0.08
JSNMF+KM	0.8253 ± 0.00	0.4362 ± 0.00	0.8196 ± 0.13	0.4202 ± 0.17
RJSNMF+KM	$\mathbf{0.9674 \pm 0.11}$	$\mathbf{0.7933 \pm 0.14}$	$\mathbf{0.9029 \pm 0.09}$	$\mathbf{0.6763 \pm 0.08}$

The clustering results for LSSMTC are taken from Gu and Zhou (2009a) for reference purpose knowledge between the tasks from their individual knowledge and transfer only the common knowledge through matrix $\mathbf{W}$ whereas JSNMF does not force any such regularization. Due to the lack of regularization in JSNMF, the matrix $\mathbf{W}$ may contain basis vectors to represent not only common structures but also some individual structures (usually happens for strong topics). This causes negative knowledge transfer (compare NMF+KM and JSNMF+KM results for Talk data set) and results in suboptimal performance.

6 Learning parameters and convergence behavior

In this section, we provide a way to learn the various parameters required for our regularized shared subspace learning framework proposed in Sect. 3.

6.1 Regularization parameters

Consider the regularization term used in Eq. 4

$$R\left(\mathbf{W}, \boldsymbol{U}, \boldsymbol{V}\right) = \alpha \left\| \mathbf{W}^{\mathsf{T}} \boldsymbol{U} \right\|_F^2 + \beta \left\| \mathbf{W}^{\mathsf{T}} \boldsymbol{V} \right\|_F^2 + \gamma \left\| \boldsymbol{U}^{\mathsf{T}} \boldsymbol{V} \right\|_F^2 \qquad (12)$$

Each element in matrices $\mathbf{W}^{\mathsf{T}}\boldsymbol{U}$, $\mathbf{W}^{\mathsf{T}}\boldsymbol{V}$ and $\boldsymbol{U}^{\mathsf{T}}\boldsymbol{V}$ has a value between 0 and 1 as each column of $\mathbf{W}, \boldsymbol{U}$ and $\boldsymbol{V}$ is normalized to L_2-norm 1. Therefore, it is appropriate to normalize the term having $\|\mathbf{W}^{\mathsf{T}}\boldsymbol{U}\|_F^2$ by $K_w K_u$ since there are $K_w \times K_u$ elements in $\mathbf{W}^{\mathsf{T}}\boldsymbol{U}$. Similarly, $\|\mathbf{W}^{\mathsf{T}}\boldsymbol{V}\|_F^2$ and $\left\|\boldsymbol{U}^{\mathsf{T}}\boldsymbol{V}\right\|_F^2$ are normalized by $K_w K_v$ and $K_u K_v$ respectively. Except these differences in normalization, we treat the regularization for all three terms in Eq. 12 equally, i.e. weighting them by a common factor a. Explicitly speaking, we use $\alpha = \frac{a}{K_w K_u}$, $\beta = \frac{a}{K_w K_v}$ and $\gamma = \frac{a}{K_u K_v}$ where $K_w \triangleq K, K_u = R_1 - K, K_v = R_2 - K$ and a is a common regularization factor for each orthogonalization. By increasing the value of a, we obtain solutions which become increasingly mutually orthogonal (i.e. $R\left(\mathbf{W}, \boldsymbol{U}, \boldsymbol{V}\right)$ moves closer to zero) but at the same time, it also causes increase in the joint factorization error, i.e. $\lambda_{\boldsymbol{X}} \left\|\boldsymbol{X} - [\mathbf{W} \mid \boldsymbol{U}]\mathbf{H}\right\|_F^2 + \lambda_{\boldsymbol{Y}} \left\|\boldsymbol{Y} - [\mathbf{W} \mid \boldsymbol{V}]\mathbf{L}\right\|_F^2$. This necessitates a trade-off between the data-fitting and the mutual orthogonality. Motivated by this, when we look at the combined objective function of optimization problem Π, we find that there exists an optimum value for a. Figure 6 shows the variations of combined objective function w.r.t the parameter a. We can see that, for all the data sets except Rec-Talk, the optimum value of a is 100 whereas the optimum value of a for Rec-Talk data set is 10. Even for Rec-Talk, $a = 100$ achieves a value of combined objective function almost equal to the optimum value. Therefore, the parameter a can be fixed at 100 for most of the real-world data sets.

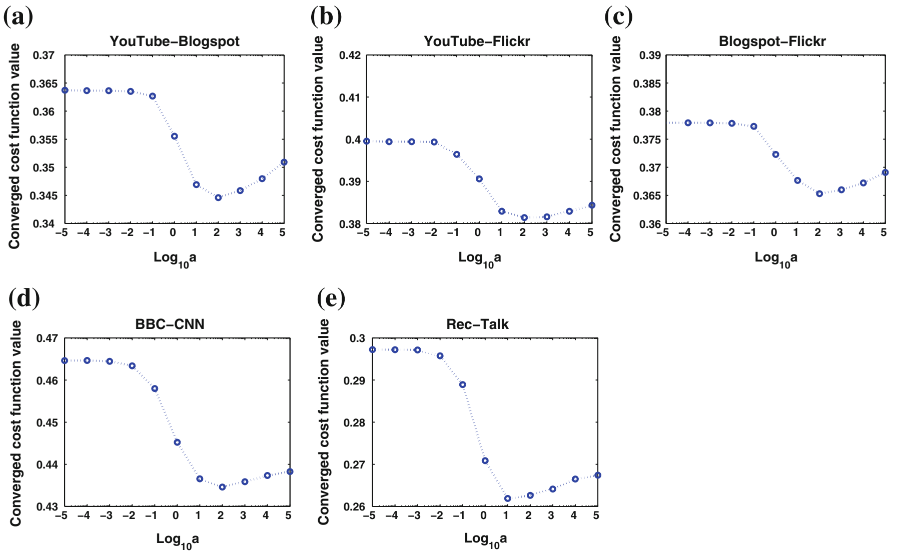


Fig. 6 Variations in cost function w.r.t. the regularization parameter (a) for various target/auxiliary pairs

6.2 Subspace dimensionality

The dimensionality of shared and individual subspaces can be determined by estimating the number of significant nonnegative basis vectors used for fitting the data. In particular, the subspace dimensionalities R_1 and R_2 can be estimated from nonnegative matrix factorization of matrices $\boldsymbol{X}$ and $\boldsymbol{Y}$ respectively with *increasing* value of basis vectors. We plot the rate of change of factorization error (Δ_k) with respect to the number of basis vectors where $\Delta_k \triangleq 1 - \frac{\boldsymbol{E}_{k+1}}{\boldsymbol{E}_k}$ and $\boldsymbol{E}_k$ is the converged NMF factorization error at the number of basis vectors used at k-th index. As can be seen from Fig. 7 that Δ_k decreases sharply in the beginning and then, becomes almost constant. In other words, after a certain value of the number of basis vectors, the value of Δ_k saturates in spite of further increases in the number of basis vectors—indicating that the number of basis vectors have already reached the inherent true dimensionality. As a conservative estimate, the number of basis vectors are set to a value after which Δ_k reduces to a value less than 1%. The first seven plots in Fig. 7a–g show the variation of Δ_k w. r. t. the number of basis vectors (R) for each data set. Since we have generated these plots at an interval of 5, we choose the best value within the selected interval based on the task performance.

For selecting the shared subspace dimensionality, i.e. K, we use a similar strategy as above. Assuming that $\mathbf{H}_w$ and $\mathbf{L}_w$ are full-rank (which is always the case if $N_1 > K$ and $N_2 > K$) and $\mathbf{W}$, $\boldsymbol{U}$ and $\boldsymbol{V}$ are mutually orthogonal, K can be estimated from the nonnegative matrix factorization of matrix $\boldsymbol{X}^{\mathsf{T}}\boldsymbol{Y}$. Consider,

$$\boldsymbol{X}^{\mathsf{T}}\boldsymbol{Y} \approx \mathbf{H}^{\mathsf{T}} \begin{bmatrix} \mathbf{W}^{\mathsf{T}}\mathbf{W} & \mathbf{W}^{\mathsf{T}}\boldsymbol{V} \\ \boldsymbol{U}^{\mathsf{T}}\mathbf{W} & \boldsymbol{U}^{\mathsf{T}}\boldsymbol{V} \end{bmatrix} \mathbf{L} = \mathbf{H}^{\mathsf{T}} \begin{bmatrix} \mathbf{W}^{\mathsf{T}}\mathbf{W} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \mathbf{L} = \left(\mathbf{H}_w^{\mathsf{T}}\mathbf{W}^{\mathsf{T}}\right)(\mathbf{W}\mathbf{L}_w)$$

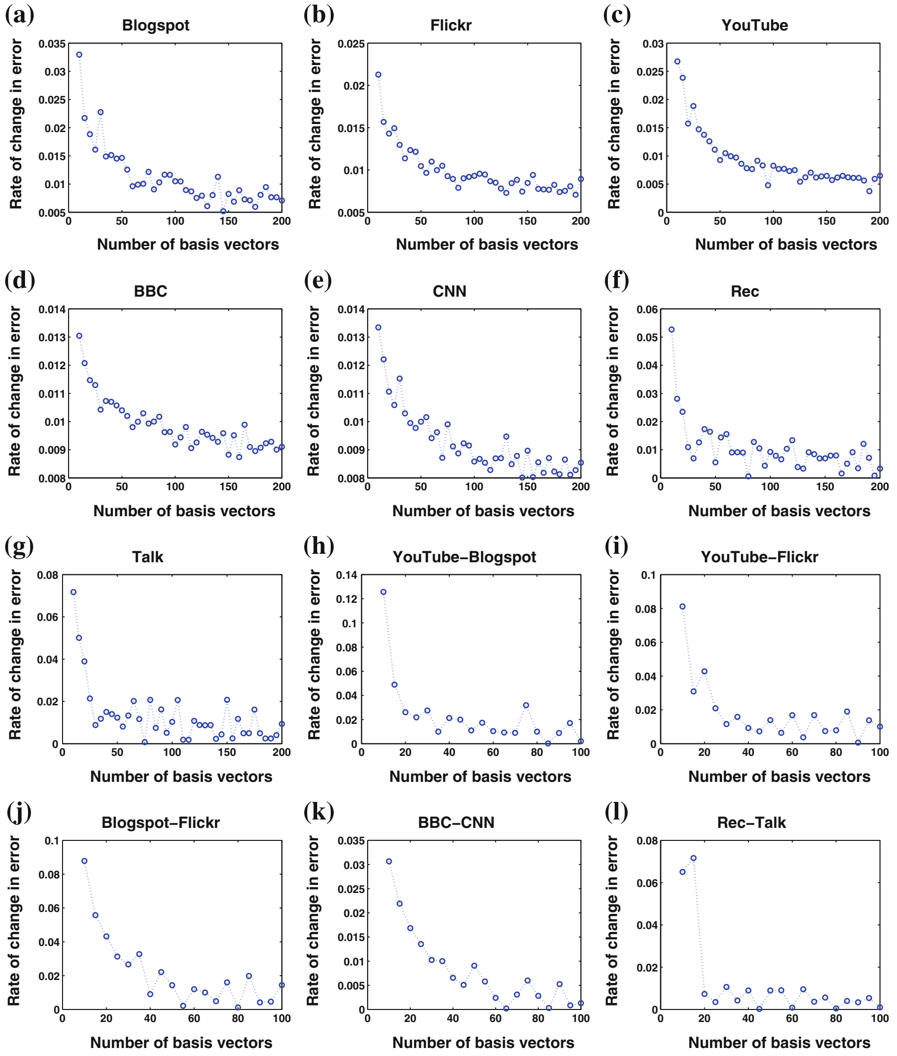


Fig. 7 Subspace dimensionality plots; the variations w.r.t. the total subspace dimensionalities are shown in (**a–g**) whereas those w.r.t. the shared subspace dimensionalities are shown in (**h–l**)

which is a product of two nonnegative matrices. Using above observation, we compute the nonnegative matrix factorization of $\boldsymbol{X}^{\top}\boldsymbol{Y}$ with increasing numbers of basis vectors. Once the number of basis vectors reach the true matrix rank, the rate of change of the cost function (Δ_k) becomes minimal in spite of further increases in the number of basis vectors. Again, we select a value for the number of basis vectors after which Δ_k reduces to a value less than 1% and set this value as K. The last five plots in Fig. 7h–l show the variation of Δ_k w. r. t. the number of shared basis vectors (K) for data set pairs (target/auxiliary) used in our experiments section. Again, after selecting

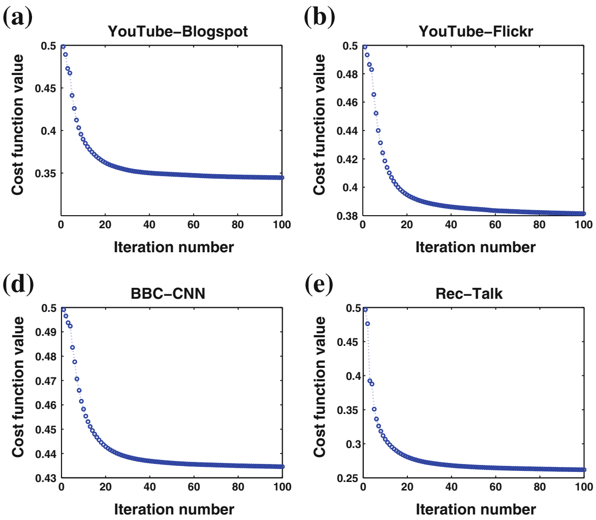


Fig. 8 Convergence behavior of RJSNMF for various target and auxiliary data set pairs

a particular interval with 1% criteria, the best value within the selected interval is chosen based on the task performance.

6.3 Convergence behavior

In Sect. 3.2.3, we have shown that the iterative solutions of (6–9) are convergent. Here, we empirically show the convergence behavior of RJSNMF for all the target/auxiliary pairs (e.g. Blogspot–Flickr, CNN–BBC etc) used for the experiments. Figure 8 shows that RJSNMF converges within almost 50 iterations for each target/auxiliary data set pair. Moreover, we note that the time taken per iteration for RJSNMF remains similar to the standard NMF as the order of complexity for both NMF and the proposed RJSNMF is the same.

7 Concluding remarks

In this paper, we presented a regularized shared subspace learning framework which captures the commonalities (through a shared subspace) and differences (through individual subspaces) of the related data sources and uses the shared subspace as a bridge for transferring the common knowledge between the two domains. Based on the proposed framework, we also developed efficient social media retrieval and clustering algorithms and demonstrated them using three real-world social media and news data sets. For learning the shared and individual subspaces, our framework imposes a set of mutual orthogonality constraints. This set of constraints ensures that the shared subspace is completely segregated from the individual subspaces. This feature is important in dealing with imbalanced real-world data sets, where without such a segregation, negative transfer learning may occur (Dai et al. 2009). Our experiments consistently validate this point by achieving better performance compared to the unregularized counterpart (Gupta et al. 2010) for both retrieval and clustering applications using three real world data sets. In addition, comparisons made with various state-of-the-art single and multitask clustering techniques using both social media and news data sets demonstrate the effectiveness of RJSNMF for multi-task clustering. Our regularized shared subspace learning solution provides a generic framework and we foresee its wider adoption in several multi-task learning and data mining tasks e.g. cross-domain collaborative filtering, cross-domain sentiment analysis etc.

While in this work, we have presented the joint modeling of two data sources, the proposed approach can readily be extended to model multiple data sources with arbitrary sharing configurations among them. Furthermore, the data set comprising of social media tags is quite sparse and thus data structures from sparse linear algebra can further be utilized to speed up the retrieval algorithm. Lastly, inferring the subspace dimensionalities automatically is an outstanding problem. One possible direction to address this issue is by extending our work along the lines of nonparametric matrix factorization (Yu et al. 2009) under the context of multiple data sources.

A Appendix

Proof of Lemma 2

Proof When $\mathbf{W}_{ij}^{(t)} = \mathbf{W}_{ij}$, we clearly have $J\left(\mathbf{W}_{ij}, \mathbf{W}_{ij}\right) = C_{ij}\left(\mathbf{W}_{ij}\right)$, therefore, for all other values of $\mathbf{W}_{ij}^{(t)}$, we need to show that $J\left(\mathbf{W}_{ij}, \mathbf{W}_{ij}\right) \geq C_{ij}\left(\mathbf{W}_{ij}\right)$. Consider the Taylor expansion of $C_{ij}\left(\mathbf{W}_{ij}\right)$ around $\mathbf{W}_{ij}^{(t)}$, which is given as

$$\begin{aligned} C_{ij}\left(\mathbf{W}_{ij}\right) = C_{ij}\left(\mathbf{W}_{ij}^{(t)}\right) + \left(\mathbf{W}_{ij} - \mathbf{W}_{ij}^{(t)}\right) \nabla C_{ij}\left(\mathbf{W}_{ij}^{(t)}\right) \\ + \frac{1}{2}\left(\mathbf{W}_{ij} - \mathbf{W}_{ij}^{(t)}\right)^2 \nabla^2 C_{ij}\left(\mathbf{W}_{ij}^{(t)}\right) \end{aligned}$$

The second derivative of $C_{ij}\left(\mathbf{W}_{ij}^{(t)}\right)$ is given as

$$\nabla^2 C_{ij}\left(\mathbf{W}_{ij}^{(t)}\right) = \left[\lambda_X \mathbf{H}_w \mathbf{H}_w^{\mathsf{T}} + \lambda_Y \mathbf{L}_w \mathbf{L}_w^{\mathsf{T}}\right]_{jj} + \left[\alpha \boldsymbol{U} \boldsymbol{U}^{\mathsf{T}} + \beta \boldsymbol{V} \boldsymbol{V}^{\mathsf{T}}\right]_{ii}$$

Now, consider the difference between the auxiliary and cost function as

$$J\left(\mathbf{W}_{ij}, \mathbf{W}_{ij}^{(t)}\right) - C_{ij}\left(\mathbf{W}_{ij}\right) = \frac{1}{2}\left(\mathbf{W}_{ij} - \mathbf{W}_{ij}^{(t)}\right)^2 \left[S_{ij}\left(\mathbf{W}^{(t)}\right) - \nabla^2 C_{ij}\left(\mathbf{W}_{ij}^{(t)}\right)\right] \tag{13}$$

and note that

$$\begin{aligned}\left[\lambda_{\boldsymbol{X}}\boldsymbol{X}^{(t)}\mathbf{H}_w^{\top}+\lambda_{\boldsymbol{Y}}\boldsymbol{Y}^{(t)}\mathbf{L}_w^{\top}\right]_{ij} &= \left[\lambda_{\boldsymbol{X}}\left(\mathbf{W}^{(t)}\mathbf{H}_w+\boldsymbol{U}\mathbf{H}_u\right)\mathbf{H}_w^{\top}+\lambda_{\boldsymbol{Y}}\left(\mathbf{W}^{(t)}\mathbf{L}_w+\boldsymbol{V}\mathbf{L}_v\right)\mathbf{L}_w^{\top}\right]_{ij}\\ &\geq \lambda_{\boldsymbol{X}}\sum_k \mathbf{W}_{ik}^{(t)}\left[\mathbf{H}_w\mathbf{H}_w^{\top}\right]_{kj}+\lambda_{\boldsymbol{Y}}\sum_l \mathbf{W}_{il}^{(t)}\left[\mathbf{L}_w\mathbf{L}_w^{\top}\right]_{lj}\\ &\geq \mathbf{W}_{ij}^{(t)}\left[\lambda_{\boldsymbol{X}}\mathbf{H}_w\mathbf{H}_w^{\top}+\lambda_{\boldsymbol{Y}}\mathbf{L}_w\mathbf{L}_w^{\top}\right]_{jj}\end{aligned}$$

and similarly

$$\begin{aligned}\left[\left(\alpha\boldsymbol{U}\boldsymbol{U}^{\top}+\beta\boldsymbol{V}\boldsymbol{V}^{\top}\right)\mathbf{W}^{(t)}\right]_{ij} &= \alpha\sum_k\left[\boldsymbol{U}\boldsymbol{U}^{\top}\right]_{ik}\mathbf{W}_{kj}^{(t)}+\beta\sum_l\left[\boldsymbol{V}\boldsymbol{V}^{\top}\right]_{il}\mathbf{W}_{lj}^{(t)}\\ &\geq \mathbf{W}_{ij}^{(t)}\left[\alpha\boldsymbol{U}\boldsymbol{U}^{\top}+\beta\boldsymbol{V}\boldsymbol{V}^{\top}\right]_{ii}\end{aligned}$$

Therefore,

$$\begin{aligned}&\left[\lambda_{\boldsymbol{X}}\boldsymbol{X}^{(t)}\mathbf{H}_w^{\top}+\lambda_{\boldsymbol{Y}}\boldsymbol{Y}^{(t)}\mathbf{L}_w^{\top}+\left(\alpha\boldsymbol{U}\boldsymbol{U}^{\top}+\beta\boldsymbol{V}\boldsymbol{V}^{\top}\right)\mathbf{W}^{(t)}\right]_{ij}\\ &\quad\geq \mathbf{W}_{ij}^{(t)}\left(\left[\lambda_{\boldsymbol{X}}\mathbf{H}_w\mathbf{H}_w^{\top}+\lambda_{\boldsymbol{Y}}\mathbf{L}_w\mathbf{L}_w^{\top}\right]_{jj}+\left[\alpha\boldsymbol{U}\boldsymbol{U}^{\top}+\beta\boldsymbol{V}\boldsymbol{V}^{\top}\right]_{ii}\right)\end{aligned} \tag{14}$$

Using Eqs. 13 and 14, we note that $J\left(\mathbf{W}_{ij},\mathbf{W}_{ij}^{(t)}\right)\geq C_{ij}\left(\mathbf{W}_{ij}\right)$. □

References

Agarwal A, Daumé H III, Gerber S (2010) Learning multiple tasks using manifold regularization. In: Advances in neural information processing systems, vol 23, pp 46–54

Ando R, Zhang T (2005) A framework for learning predictive structures from multiple tasks and unlabeled data. J Mach Learn Res 6:1817–1853

Bae E, Bailey J (2006) Coala: a novel approach for the extraction of an alternate clustering of high quality and high dissimilarity. In: IEEE international conference on data mining, pp 53–62

Baeza-Yates R, Ribeiro-Neto B (1999) Modern information retrieval. Addison-Wesley, Reading, MA

Baxter J (2000) A model of inductive bias learning. J Artif Intell Res 12:149–198

Ben-David S, Schuller R (2003) Exploiting task relatedness for multiple task learning. In: 16th annual conference on computational learning theory, vol 2777, pp 567–580

Berry M, Browne M (2005) Email surveillance using non-negative matrix factorization. Comput Math Organ Theory 11(3):249–264

Berry M, Browne M, Langville A, Pauca V, Plemmons R (2007) Algorithms and applications for approximate nonnegative matrix factorization. Comput Stat Data Anal 52(1):155–173

Bickel S, Scheffer T (2004) Multi-view clustering. In: Proceedings of the IEEE international conference on data mining, pp 19–26

Blei D, Ng A, Jordan M (2003) Latent Dirichlet allocation. J Mach Learn Res 3:993–1022

Boutsidis C, Gallopoulos E (2008) SVD based initialization: a head start for nonnegative matrix factorization. Pattern Recogn 41(4):1350–1362

Bucak S, Gunsel B (2007) Video content representation by incremental non-negative matrix factorization. ICIP 2:113–116

Cai D, He X, Han J (2007) Semi-supervised discriminant analysis. In: International conference on computer vision, pp 1–7

Cai D, He X, Wu X, Han J (2008) Non-negative matrix factorization on manifold. IEEE international conference on data mining, pp 63–72

Cai D, He X, Han J, Huang T (2011) Graph regularized non-negative matrix factorization for data representation. IEEE Trans Pattern Anal Mach Intell 33(8):1548–1560

Caruana R (1997) Multitask learning. Mach Learn 28(1):41–75

Chaudhuri K, Kakade S, Livescu K, Sridharan K (2009) Multi-view clustering via canonical correlation analysis. In: Proceedings of the 26th international conference on machine learning, pp 129–136

Choi S (2008) Algorithms for orthogonal nonnegative matrix factorization. In: Proceedings of the international joint conference on neural networks, pp 1828–1832

Cui Y, Fern X, Dy J (2007) Non-redundant multi-view clustering via orthogonalization. In: IEEE international conference on data mining. IEEE, pp 133–142

Dai W, Jin O, Xue G, Yang Q, Yu Y (2009) Eigentransfer: a unified framework for transfer learning. ICML, pp 193–200

Dempster A, Laird N, Rubin D et al (1977) Maximum likelihood from incomplete data via the EM algorithm. J R Stat Soc Ser B (Methodol) 39(1):1–38

Ding C, Li T, Peng W, Park H (2006) Orthogonal nonnegative matrix tri-factorizations for clustering. In: Proceedings of the 12th ACM SIGKDD international conference on knowledge discovery and data mining. ACM, pp 126–135

Duda R, Hart P, Stork D (2001) Pattern classification, vol 2. Wiley-Interscience, New York

Gu Q, Zhou J (2009a) Learning the shared subspace for multi-task clustering and transductive transfer classification. In: IEEE international conference on data mining, pp 159–168

Gu Q, Zhou J (2009b) Local learning regularized nonnegative matrix factorization. In: Proceedings of the 21st international joint conference on artificial intelligence, pp 1046–1051

Gupta S, Phung D, Adams B, Tran T, Venkatesh S (2010) Nonnegative shared subspace learning and its application to social media retrieval. In: Proceedings of the 16th ACM SIGKDD international conference on knowledge discovery and data mining, pp 1169–1178

Gupta S, Phung D, Adams B, Venkatesh S (2011a) A Bayesian framework for learning shared and individual subspaces from multiple data sources. In: Advances in knowledge discovery and data mining, 15th Pacific-Asia conference (PAKDD), pp 136–147

Gupta SK, Phung D, Adams B, Venkatesh S (2011b) A matrix factorization framework for jointly analyzing multiple nonnegative data sources. In: Proceedings of text mining workshop, in conjuction with SIAM international conference on data mining

Hardoon D, Szedmak S, Shawe-Taylor J (2004) Canonical correlation analysis: an overview with application to learning methods. Neural Comput 16(12):2639–2664

Ji S, Tang L, Yu S, Ye J (2008) Extracting shared subspace for multi-label classification. In: Proceeding of the 14th ACM SIGKDD international conference on knowledge discovery and data mining, pp 381–389

Jolliffe I (2002) Principle component analysis. Springer, Heidelberg

Kailing K, Kriegel H, Pryakhin A, Schubert M (2004) Clustering multi-represented objects with noise. In: Advances in knowledge discovery and data mining, 8th Pacific-Asia conference (PAKDD), pp 394–403

Kim H, Park H (2008) Non-negative matrix factorization based on alternating non-negativity constrained least squares and active set method. SIAM J Matrix Anal Appl 30(2):713–730

Langville A, Meyer C, Albright R, Cox J, Duling D (2006) Initializations for the nonnegative matrix factorization. In: Proceedings of the 12th ACM SIGKDD international conference on knowledge discovery and data mining

Lee D, Seung H (2001) Algorithms for non-negative matrix factorization. Adv Neural Inform Process Syst 13:556–562

Li T, Ma S, Ogihara M (2004) Document clustering via adaptive subspace iteration. In: Proceedings of the 27th international ACM SIGIR conference on research and development in information retrieval, pp 218–225

Lin C (2007) Projected gradient methods for nonnegative matrix factorization. Neural Comput 19(10):2756–2779

Lin Y, Sundaram H, De Choudhury M, Kelliher A (2009) Temporal patterns in social media streams: theme discovery and evolution using joint analysis of content and context. In: IEEE international conference on multimedia and expo, pp 1456–1459

Lovász L, Plummer M (1986) Matching theory. Elsevier, Amsterdam

Manning C, Raghavan P, Schütze H (2008) Introduction to information retrieval. Cambridge University Press, Cambridge

Mardia KV, Bibby JM, Kent JT (1979) Multivariate analysis. Academic, New York

Niu D, Dy J, Jordan M (2010) Multiple non-redundant spectral clustering views. In: Proceedings of the 27th international conference on machine learning, pp 831–838

Pan S, Yang Q (2008) A survey on transfer learning. Technical Report HKUST-CS08-08. Department of Computer Science and Engineering, HKUST, Hong Kong

Qi Z, Davidson I (2009) A principled and flexible framework for finding alternative clusterings. In: Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining, pp 717–726

Rui Y, Huang T (2000) Optimizing learning in image retrieval. In: Proceedings of the IEEE conference on computer vision and pattern recognition. Published by the IEEE Computer Society, pp 236–243

Salton G, Buckley C (1988) Term-weighting approaches in automatic text retrieval. Inform Process Manag 24(5):513–523

Shi J, Malik J (2000) Normalized cuts and image segmentation. IEEE Trans Pattern Anal Mach Intell 22(8):888–905

Si S, Tao D, Geng B (2009) Bregman divergence based regularization for transfer subspace learning. IEEE Trans Knowl Data Eng 22(7):929–942

Thrun S (1996) Is learning the n-th thing any easier than learning the first? In: Advances in neural information processing systems, pp 640–646

Wild S, Curry J, Dougherty A (2004) Improving non-negative matrix factorizations through structured initialization. Pattern Recogn 37(11):2217–2232

Wiswedel B, Höppner F, Berthold M (2010) Learning in parallel universes. Data Min Knowl Disc 21(1):130–152

Xu W, Liu X, Gong Y (2003) Document clustering based on non-negative matrix factorization. In: Proceedings of the 26th international ACM SIGIR conference on research and development in information retrieval. ACM, New York, NY, pp 267–273

Yan R, Tesic J, Smith J (2007) Model-shared subspace boosting for multi-label classification. In: Proceedings of the 13th ACM SIGKDD international conference on knowledge discovery and data mining. ACM, pp 834–843

Yang T, Jin R, Jain A, Zhou Y, Tong W (2010) Unsupervised transfer classification: application to text categorization. In: Proceedings of the 16th ACM SIGKDD international conference on knowledge discovery and data mining. ACM, pp 1159–1168

Yu K, Zhu S, Lafferty J, Gong Y (2009) Fast nonparametric matrix factorization for large-scale collaborative filtering. In: Proceedings of the 32nd international ACM SIGIR conference on research and development in information retrieval, pp 211–218

Zhang J, Zhang C (2011) Multitask Bregman clustering. Neurocomputing 74(10):1720–1734

Zhou D, Bousquet O, Lal T, Weston J, Schölkopf B (2004) Learning with local and global consistency. Adv Neural Inform Process Syst 16:595–602

Zhuang F, Luo P, Shen Z, He Q, Xiong Y, Shi Z, Xiong H (2010) Collaborative dual-plsa: mining distinction and commonality across multiple domains for text classification. CIKM, pp 359–368