A Matrix Factorization Framework for Jointly Analyzing Multiple Nonnegative Data Sources

Sunil Kumar Gupta* † Dinh Phung * ‡ Brett Adams * § Svetha Venkatesh * ¶

Abstract

Nonnegative matrix factorization based methods provide one of the simplest and most effective approaches to text mining. However, their applicability is mainly limited to analyzing a single data source. In this paper, we propose a novel joint matrix factorization framework which can jointly analyze multiple data sources by exploiting their shared and individual structures. The proposed framework is flexible to handle any arbitrary sharing configurations encountered in real world data. We derive an efficient algorithm for learning the factorization and show that its convergence is theoretically guaranteed. We demonstrate the utility and effectiveness of the proposed framework in two real-world applications–improving social media retrieval using auxiliary sources and cross-social media retrieval. Representing each social media source using their *textual tags*, for both applications, we show that retrieval performance exceeds the existing state-of-the-art techniques. The proposed solution provides a generic framework and can be applicable to a wider context in data mining wherever one needs to exploit mutual and individual knowledge present across multiple data sources.

1 Introduction

Feeding our insatiable appetite for content, multiple data sources surround us. Data from a single source is often not rich enough and users look for information across multiple sources and modalities. The research community has focused on data mining and analysis from single data source, but limited work addresses issues arising from the joint analysis of multiple data sources. This has created open opportunities to develop formal frameworks for analyzing multiple data sources, exploiting common properties to strengthen data analysis and mining. Discovering patterns from multiple data sources often provides information such as commonalities and differences, otherwise not possible with isolated analysis. This information is valuable for various data mining and representation tasks.

As an example, consider social media. Entirely new genres of media have been created around the idea of participation, including wikis (e.g. Wikipedia), social networks (e.g. Facebook), media communities (e.g. YouTube), news aggregators (e.g. Digg), blogs and micro-blogs (e.g. Blogspot, Twitter). These applications are significant because they are often ranked highest by traffic volume and attention. Modeling collective data across semantically similar yet disparate sources is critical for social media mining and retrieval tasks. Open questions are: how can we effectively analyze such disparate data sources together exploiting their mutual strengths for improving data mining tasks? Can we establish the correspondence or similarity of items in one data source with items in other data sources?

This paper attempts to address these questions and develops a framework to model multiple data sources jointly by exploiting their mutual strengths while retaining their individual knowledge. To analyze multiple disparate data sources jointly, a unified piece of meta data–*textual tags*–are used. Although we use textual tags in this work, any other feature unifying the disparate data sources can be used. Textual tags are rich in semantics [5, 14] as they are meant to provide higher level description to the data, and are freely available for disparate data types e.g. images, videos, blogs, news etc. However, these tags cannot be used directly to build useful applications due to their noisy characteristics. The lack of constraints during their creation are part of their appeal, but consequently they become ambiguous, incomplete and subjective [14, 5], leading to poor performance in data mining tasks. Work on tagging systems has been mainly aimed at refining tags by determining their relevance and recommending additional tags [14, 17] to reduce the ambiguity. But the performance of these techniques is bounded by the information content and noise characteristics of the tagging source in question, which can vary wildly depending on many factors, including the design of the tagging system and the uses to which it is being put by its users. To reduce tag ambiguity, the use of auxiliary data sources along with the domain of interest is suggested in [9]. The intuition behind the joint analysis of multiple data sources is that the the combined tagging knowledge tend to reduce the subjectivity of tags [7] as multiple related sources often provide complementary knowledge and strengthen one another.

Departing from single data source based methods, we formulate a novel framework to leverage tags as the unifying metadata across multiple disparate data sources. The key idea is to model the data subspaces that allows flexibility in representing the commonalities whilst retaining their

*Department of Computing, Curtin University, Perth, Western Australia
†sunil.gupta@postgrad.curtin.edu.au
‡d.phung@curtin.edu.au
§b.adams@curtin.edu.au
¶s.venkatesh@curtin.edu.au

individual differences. Retaining the individual differences of each data source is crucial when dealing with heterogeneous multiple data sources as ignoring this aspect may lead to negative knowledge transfer [6]. Our proposed framework is based on nonnegative matrix factorization (NMF) [10] and provides shared and individual basis vectors wherein tags of each media object can be represented by linear combination of shared and individual topics. We extend NMF to enable joint modeling of multiple data sources deriving common and individual subspaces.

Pairwise analysis using two data sources has been considered in our previous work [7]. However, it is limited and unusable for many real-world applications where one needs to include several auxiliary sources to achieve more meaningful improvement in performance as shown in this paper. Furthermore, extension to multiple data sources requires efficient learning of arbitrarily shared subspaces which is non-trivial and fundamentally different from the work in [7]. For example, consider three sources D_1, D_2 and D_3; jointly modeling (D_1, D_2, D_3) is different from pairwise modeling (D_1, D_2) or (D_2, D_3). Figure 1 depicts an example of the possible sharing configurations (refer section 3) for three data sources. We note that the work in [7] can not handle the sharing configuration of Figure 1c. To demonstrate the effectiveness of our approach, we apply the proposed model on two real world tasks–improving social media retrieval using auxiliary sources and cross-social media retrieval–using three disparate data sources (Flickr, YouTube and Blogspot). Our main contributions are :

- A joint matrix factorization framework along with an efficient algorithm for extraction of shared and individual subspaces across an arbitrary number of data sources. We provide complexity analysis of the learning algorithm and show that its convergence is guaranteed via a proof of convergence. (in section 3 and and Appendix A)
- We further develop algorithms for social media retrieval in a multi-task learning setting and cross-social media retrieval. (in section 4)
- Two real world demonstrations of the proposed framework using three representative social media sources–blogs (Blogspot.com), photos (Flickr) and videos (YouTube). (in section 5)

By permitting differential amounts of sharing in the subspaces, our framework can transfer knowledge across multiple data sources and thus, can be applied to a much wider context–it is appropriate wherever one needs to exploit the knowledge across multiple related data sources avoiding negative knowledge transfer. Speaking in social media context, it provides efficient means to mine multimedia data, and partly transcend the semantic gap by exploiting the diversity of rich tag metadata from many media domains.

2 Related Background

Previous works on shared subspace learning are mainly focused on supervised or semi-supervised learning. Ando and Zhang [1] propose *structure learning* to discover predictive structures shared by the multiple classification problems to improve performance on the target data source in transfer learning settings. Yan et al [19] propose a multi-label learning algorithm called model-shared subspace boosting to reduce information redundancy in learning by combining a number of base models across multiple labels. Ji et al [8] learn a common subspace shared among multiple labels to extract shared structures for multi-label classification task. In a transfer learning work [6], Gu and Zhou propose a framework for multi-task clustering by learning a common subspace among all tasks and use it for transductive transfer classification. A limitation of their framework is that it learns a single shared subspace for each task which often violates data faithfulness in many real world scenarios. Si et al [16] propose a family of transfer subspace learning algorithms based on a regularization which minimizes Bregman divergence between the distributions of the training and test samples. Though, this approach, fairly generic for domain adaptation setting, is not directly applicable for multi-task learning and does not model multiple data sources. In contrast to the above works, our proposed framework not only provides varying levels of sharing but is flexible to support arbitrary sharing configurations for any combination of multiple data sources (tasks).

Our proposed shared subspace learning method is formulated under the framework of NMF. NMF is chosen to model the tags (as tags are basically textual keywords) due to its success in text mining applications [18, 3, 15]. An important characteristic of NMF is that it yields parts based representation of the data.

Previous approaches taken for cross-media retrieval [20, 21] use the concept of a Multimedia Document (MMD), which is a set of *co-occurring* multimedia objects that are of different modalities carrying the same semantics. The two multimedia objects can be regarded as context for each other if they are in the same MMD, and thus the combination of content and context is used to overcome the semantic gap. However, this line of research depends on *co-occurring* multimedia objects, which may not be available.

3 Multiple Shared Subspace Learning

3.1 Problem Formulation In this section, we describe a framework for learning individual as well as arbitrarily shared subspaces of multiple data sources. Let $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$ represent the feature matrices constructed from a set of n data sources where $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$ can be, for example, *user-item rating* matrices (where each row corresponds to a user, each column corresponds to an item and the features are user ratings) in case of collaborative filtering application or

term-document matrices for tag based social media retrieval application (where each row corresponds to a tag, each column corresponds to an item and features are usual tf-idf weights [2]) and so on. Given $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$, we decompose each data matrix $\boldsymbol{X}_i$ as a product of two matrices $\boldsymbol{X}_i = \boldsymbol{W}_i \cdot \mathbf{H}_i$ such that the subspace spanned by the columns of matrix $\boldsymbol{W}_i$ explicitly represents arbitrary sharing among n data sources through *shared subspaces* and individual data by preserving their *individual subspaces*. For example, when $n = 2$, we create three subspaces: a shared subspace spanned by matrix W_{12} and two individual subspaces spanned by matrices W_1, W_2. Formally,

(3.1)
$$\boldsymbol{X}_1 = \underbrace{[W_{12} \mid W_1]}_{\boldsymbol{W}_1} \underbrace{\begin{bmatrix} H_{1,12} \\ \\ H_{1,1} \end{bmatrix}}_{\mathbf{H}_1} = W_{12} \cdot H_{1,12} + W_1 \cdot H_{1,1}$$

(3.2)
$$\boldsymbol{X}_2 = \underbrace{[W_{12} \mid W_2]}_{\boldsymbol{W}_2} \underbrace{\begin{bmatrix} H_{2,12} \\ \\ H_{2,2} \end{bmatrix}}_{\mathbf{H}_2} = W_{12} \cdot H_{2,12} + W_2 \cdot H_{2,2}$$

Notationwise, we use *bold capital letters* $\boldsymbol{W}$,$\mathbf{H}$ to denote the decomposition at the data source level and *normal capital letters* W, H to denote the subspaces partly. In the above expressions, the shared basis vectors are contained in W_{12} while individual basis vectors are captured in W_1 and W_2 respectively, hence giving rise to the full subspace representation $\boldsymbol{W}_1 = [W_{12} \mid W_1]$ and $\boldsymbol{W}_2 = [W_{12} \mid W_2]$ for the two data sources. However, note that the encoding coefficients of each data source in the shared subspace corresponding to W_{12} are different, and thus, an extra subscript is used to make it explicit as $H_{1,12}$ and $H_{2,12}$.

To generalize these expressions for arbitrary n datasets, we continue with this example ($n = 2$) and consider the power set over $\{1, 2\}$ given as

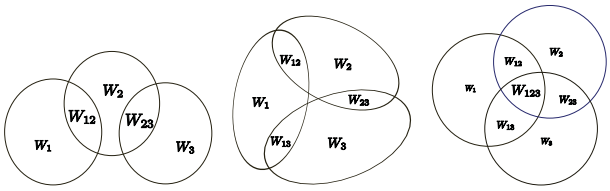
$$S(2) = \{\emptyset, \{1\}, \{2\}, \{1, 2\}\}$$

We can use the power set $S(2)$ to create an index set for the subscripts '1', '2' and '12' used in matrices of Eqs (3.1) and (3.2). This helps in writing the factorization conveniently using a summation. We further use $S(2, i)$ to denote the subset of $S(2)$ in which only elements involving i are retained, i.e.

$$S(2, 1) = \{\{1\}, \{1, 2\}\} \text{ and } S(2, 2) = \{\{2\}, \{1, 2\}\}$$

With a little sacrifice of perfection over the set notation, we rewrite them as $S(2, 1) = \{1, 12\}$ and $S(2, 2) = \{2, 12\}$. Now, using these sets, Eqs (3.1) and (3.2) can be re-written as

$$\boldsymbol{X}_1 = \sum_{v \in \{1,12\}} W_v \cdot H_{1,v} \text{ and } \boldsymbol{X}_2 = \sum_{v \in \{2,12\}} W_v \cdot H_{2,v}$$



(a) chain sharing (b) pairwise sharing (c) general sharing

Figure 1: Some possible sharing configurations for $n = 3$ datasets. In this paper, we consider general sharing as in (c).

For an arbitrary set of n datasets, let $S(n)$ denote the power set of $\{1, 2, \ldots n\}$ and for each $i = 1, \ldots, n$, let the index set associated with the i-th data source be defined as $S(n, i) = \{v \in S(n) \mid i \in v\}$. Our proposed joint matrix factorization for n data sources can then be written as

$$\boldsymbol{X}_i = \boldsymbol{W}_i \cdot \mathbf{H}_i = \sum_{v \in S(n,i)} W_v \cdot H_{i,v} \tag{3.3}$$

Our above expression is in its most generic form considering all possible sharing opportunities that can be formulated. In fact, the total number of subspaces equates to $2^n - 1$ which is the cardinality of the power set $S(n)$ minus the empty set $\emptyset$. We consider this generic form in this paper. However, our framework is directly applicable where we can customize the index set $S(n, i)$ to tailor any combination of sharing one wish to model. Figure 1 illustrates some of the possible scenarios when there are three data sources ($n = 3$).

If we explicitly list the elements of $S(n, i)$ as $S(n, i) = \{v_1, v_2, \ldots, v_Z\}$ then $\boldsymbol{W}_i$ and $\mathbf{H}_i$ are

$$\boldsymbol{W}_i = [W_{v_1} \mid W_{v_2} \mid \ldots \mid W_{v_Z}] \text{ , } \mathbf{H}_i = \begin{bmatrix} H_{i,v_1} \\ \vdots \\ H_{i,v_Z} \end{bmatrix} \tag{3.4}$$

3.2 Learning and Optimization Our goal is to achieve sparse part-based representation of the subspaces and therefore, we impose nonnegative constraints on $\{\boldsymbol{W}_i, \mathbf{H}_i\}_{i=1}^n$. We formulate an optimization problem to minimize the Frobenius norm of joint decomposition error. The objective function accumulating normalized decomposition error across all data matrices is given as

$$J(\boldsymbol{W}, \mathbf{H}) = \frac{1}{2} \left\{ \sum_{i=1}^{n} \lambda_i \|\boldsymbol{X}_i - \boldsymbol{W}_i \cdot \mathbf{H}_i\|_F^2 \right\}$$

$$= \frac{1}{2} \left\{ \sum_{i=1}^{n} \lambda_i \left\| \boldsymbol{X}_i - \sum_{v \in S(n,i)} W_v \cdot H_{i,v} \right\|_F^2 \right\} \tag{3.5}$$

where $\|.\|_F$ is the Frobenius norm and $\lambda_i \triangleq \|\boldsymbol{X}_i\|_F^{-2}$ is the normalizing factor for data $\boldsymbol{X}_i$. Thus, the final optimization is given as

$$\text{minimize } J(\boldsymbol{W}, \mathbf{H}) \tag{3.6}$$
$$\text{subject to } W_v, H_{i,v} \geq 0 \text{ for all } 1 \leq i \leq n \text{ and } v \in S(n,i)$$

where $J(\boldsymbol{W}, \mathbf{H})$ is defined as in Eq (3.5). A few directions are available to solve this nonnegatively constrained optimization problem, such as gradient-descent based multiplicative updates [10] or projected gradient [11]. We found that optimization of $J(\boldsymbol{W}, \mathbf{H})$ using multiplicative updates provides a good trade off between automatically selecting gradient-descent step size and fast convergence for both synthetic and real datasets, and therefore, will be used in this paper. Expressing the objective function element-wise, we shall show that multiplicative update equations for W_v and $H_{i,v}$ can be formulated efficiently as in the standard NMF [10]. Since the cost function of Eq (3.5) is non-convex jointly for all W_v and $H_{i,v}$, the multiplicative updates lead to a local minima solution. However, unlike NMF, this problem is less ill-posed due to the constraints of common matrices in the joint factorization. The gradient of the cost function in Eq (3.5) w.r.t. W_v is given by

$$\nabla_{W_v} J(\boldsymbol{W}, \mathbf{H}) = \sum_{i \in v} \lambda_i \left[-\boldsymbol{X}_i H_{i,v}^\mathsf{T} + \boldsymbol{X}_i^{(t)} H_{i,v}^\mathsf{T} \right]$$

where $\boldsymbol{X}_i^{(t)}$ is defined as

$$\boldsymbol{X}_i^{(t)} = \sum_{v \in S(n,i)} W_v \cdot H_{i,v} \tag{3.7}$$

Using Gradient-Descent optimization, we update matrix W_v as the following

$$(W_v)_{lk}^{t+1} \leftarrow (W_v)_{lk}^t + \eta_{(W_v)_{lk}^t} \left(-\nabla_{(W_v)_{lk}^t} J(\boldsymbol{W}, \mathbf{H}) \right) \tag{3.8}$$

where $\eta_{(W_v)_{lk}^t}$ is the optimization step-size and given by

$$\eta_{(W_v)_{lk}^t} = \frac{(W_v)_{lk}^t}{\sum_{i \in v} \lambda_i \left(\boldsymbol{X}_i^{(t)} H_{i,v}^\mathsf{T} \right)_{lk}^t} \tag{3.9}$$

In Appendix, we prove that the updates in Eq (3.8) when combined with step-size of Eq (3.9), converge to provide a locally optimum solution of the optimization problem 3.6. Plugging the value of $\eta_{(W_v)_{lk}^t}$ from Eq (3.9) in Eq (3.8), we obtain the following multiplicative update equation for W_v

$$(W_v)_{lk} \leftarrow (W_v)_{lk} \frac{\left(\sum_{i \in v} \lambda_i \boldsymbol{X}_i \cdot H_{i,v}^\mathsf{T} \right)_{lk}}{\left(\sum_{i \in v} \lambda_i \boldsymbol{X}_i^{(t)} \cdot H_{i,v}^\mathsf{T} \right)_{lk}} \tag{3.10}$$

Multiplicative updates for $H_{i,v}$ can be obtained similarly and given by

$$(H_{i,v})_{km} \leftarrow (H_{i,v})_{km} \frac{\left(W_v^\mathsf{T} \cdot \boldsymbol{X}_i \right)_{km}}{\left(W_v^\mathsf{T} \cdot \boldsymbol{X}_i^{(t)} \right)_{km}} \tag{3.11}$$

As an example, for the case of $n = 2$ data sources mentioned earlier, the update equations for the shared subspace W_{12} (corresponding to $v = \{1, 2\}$) reduce to

$$(W_{12})_{lk} \leftarrow (W_{12})_{lk} \frac{\left(\lambda_1 \boldsymbol{X}_1 \cdot H_{1,12}^\mathsf{T} + \lambda_2 \boldsymbol{X}_2 \cdot H_{2,12}^\mathsf{T} \right)_{lk}}{\left(\lambda_1 \boldsymbol{X}_1^{(t)} \cdot H_{1,12}^\mathsf{T} + \lambda_2 \boldsymbol{X}_2^{(t)} \cdot H_{2,12}^\mathsf{T} \right)_{lk}} \tag{3.12}$$

and the update equations for the individual subspaces W_1 (when $v = \{1\}$) and W_2 (when $v = \{2\}$) become:

$$(W_1)_{lk} \leftarrow (W_1)_{lk} \frac{\left(\boldsymbol{X}_1 \cdot H_{1,1}^\mathsf{T} \right)_{lk}}{\left(\boldsymbol{X}_1^{(t)} \cdot H_{1,1}^\mathsf{T} \right)_{lk}} \tag{3.13}$$

$$(W_2)_{lk} \leftarrow (W_2)_{lk} \frac{\left(\boldsymbol{X}_2 \cdot H_{1,2}^\mathsf{T} \right)_{lk}}{\left(\boldsymbol{X}_2^{(t)} \cdot H_{1,2}^\mathsf{T} \right)_{lk}} \tag{3.14}$$

We note the intuition carried in these update equations. First, it can be verified by inspection that at the ideal convergence point when $\boldsymbol{X}_i = \boldsymbol{X}_i^{(t)}$, the multiplicative factors (second term on the RHS) in these equations become unity, thus no more updates are necessary. Secondly, updating a particular shared subspace W_v involves only relevant data sources for that share (sum over its index set $i \in v$, cf. Eq 3.10). For example updating W_{12} in Eq (3.12) involves both $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ but updating W_1 in Eq (3.13) involves only $\boldsymbol{X}_1$; the next iteration takes into account the joint decomposition effect and regularize the parameter via Eq (3.7). From this point onwards, we refer to our framework as *Multiple Shared Nonnegative Matrix Factorization* (MS-NMF).

3.3 Subspace Dimensionality and Complexity Analysis

Let M be the number of rows for each $\boldsymbol{X}_i$ (although $\boldsymbol{X}_i$'s usually have different vocabularies but they can be merged together to construct a common vocabulary that has M words) and N_i be the number of columns. Then, the dimensions for $\boldsymbol{W}_i$ and $\mathbf{H}_i$ are $M \times R_i$ and $R_i \times N_i$ respectively using R_i as reduced dimension. Since each $\boldsymbol{W}_i$ is an augmentation of individual and shared subspace matrices W_v, we further use K_v to denote the number of columns in W_v. Next, from Eq (3.4), it implies that $\sum_{v \in S(n,i)} K_v = R_i$. The value of K_v depends upon the sharing level among the involved data sources. A rule of thumb is to use $K_v \approx \sqrt{M_v/2}$ according to [13] where M_v is equal to the number of features common in data

configuration specified by v. For example, if $v = \{1, 2\}$, M_v is equal to the number of common tags between source-1 and source-2.

Given above notation, the computational complexity for MS-NMF algorithm is $\mathcal{O}(M \times N_{\max} \times R_{\max})$ per iteration where $N_{\max} = \max_{i \in [1,n]} \{N_i\}$ and $R_{\max} = \max_{i \in [1,n]} \{R_i\}$. The standard NMF algorithm [10] when applied on each matrix $\boldsymbol{X}_i$ with parameter R_i will have a complexity of $\mathcal{O}(M \times N_i \times R_i)$ and total complexity of $\mathcal{O}(M \times N_{\max} \times R_{\max})$ per iteration. Therefore, computational complexity of MS-NMF remains equal to that of standard NMF.

4 Applications

Focusing on the social media domain, we show the usefulness of MS-NMF framework through two applications

1. Improving social media retrieval in one medium (target) with the help of other auxiliary social media sources.
2. Retrieving items across multiple social media sources.

Our key intuition in the first application is to use MS-NMF to improve retrieval by leveraging statistical strengths of tag co-occurrences through shared subspace learning while retaining the knowledge of the target medium. Intuitively, improvement is expected when auxiliary sources share underlying structures with the target medium. These auxiliary sources can be readily found from the Web. For cross-media retrieval, the shared subspace among multiple media provides a common representation for each medium and enables us to compute cross-media similarity between items of different media.

4.1 Improving Social Media Retrieval with Auxiliary Sources

Let the target medium for which retrieval is to be performed be $\boldsymbol{X}_k$. Further, let us assume that we have other auxiliary media sources $\boldsymbol{X}_j$, $j \neq k$, which share some underlying structures with the target medium. We use these auxiliary sources to improve the retrieval precision from the target medium. Given a set of query keywords S_Q, a vector q of length M (vocabulary size) is constructed by putting *tf-idf* values at each index where vocabulary contains a word from the keywords set or else putting zero. Next, we follow Algorithm 1 for retrieval using MS-NMF.

Algorithm 1 Social Media Retrieval using MS-NMF.

1: **Input**: target $\boldsymbol{X}_k$, auxiliary $\boldsymbol{X}_j$ $(\forall j \neq k)$, query q, number of items to be retrieved N.
2: learn $\boldsymbol{X}_k = \boldsymbol{W}_k \mathbf{H}_k$ using Eqs.(3.10–3.11).
3: set $\epsilon = 10^{-2}$, project q onto $\boldsymbol{W}_k$ to get h by an initialization then looping as below
4: **while** $(\|\boldsymbol{W}_k h - q\|_2 \geq \epsilon)$ **do**
5: $\quad (h)_a \leftarrow (h)_a \left(\boldsymbol{W}_k^\mathsf{T} q\right)_a / \left(\boldsymbol{W}_k^\mathsf{T} \boldsymbol{W}_k h\right)_a$
6: **end while**
7: for each media item (indexed by r) in $\boldsymbol{X}_k$, with representation $h_r = r$-th column of $\mathbf{H}_k$, compute its similarity with query projection h as following

$$\text{sim}(h, h_r) = \frac{h^\mathsf{T} h_r}{\|h\|_2 \|h_r\|_2}$$

8: **Output**: return the top N items in decreasing order of similarities.

4.2 Cross-Social Media Retrieval and Correspondence

Social media users assign tags to their content (blog, images and videos) to retrieve them later and share them with other users. Often these user generated content are associated with real world events, e.g., travel, sports, wedding receptions etc. In such a scenario, when users search for items from one medium, they are also interested in semantically similar items from other media to obtain more information. For example, one might be interested in retrieving 'olympics' related blogs, images and videos at the same time (cross-media retrieval) as together they service the user information need better.

A naïve method of cross-media retrieval is to match the query keywords with the tag lists of items of different media. Performance of this method is usually poor due to poor semantic indexing caused by noisy tags, polysemy and synonymy. Subspace methods such as LSI or NMF, although robust against these problems, don't support cross-media retrieval in their standard form. Interestingly, MS-NMF provides solutions to both the problems. First, being a subspace based method, it is less affected by the problems caused by noisy tags, 'polysemy' and 'synonymy' and second, it is appropriate for cross-media retrieval as it represents items from each medium in a common subspace enabling to define a similarity for cross-media retrieval.

To relate items from medium i and j, we use the common subspace spanned by $\boldsymbol{W}_{ij}$. As an example, $\boldsymbol{W}_{12} = [W_{12} \mid W_{123}]$, $\boldsymbol{W}_{23} = [W_{23} \mid W_{123}]$ and $\boldsymbol{W}_{13} = [W_{13} \mid W_{123}]$ for three data source case, illustrated in Figure 1c. More generally, if $S(n, i, j)$ is the set of all subsets in $S(n)$ involving both i and j, i.e. $S(n, i, j) \triangleq \{v \in S(n) \mid i, j \in v\}$, the common subspace between i-th and j-th medium $\boldsymbol{W}_{ij}$ is then given by horizontally augmenting all W_v such that $v \in S(n, i, j)$. Similarly, representation of $\boldsymbol{X}_i$ (or $\boldsymbol{X}_j$) in this common subspace, i.e. $\mathbf{H}_{i,ij}$ (or $\mathbf{H}_{j,ij}$), is given by vertically augmenting all $H_{i,v}$ (or $H_{j,v}$) such that $v \in S(n, i, j)$. For $n = 3$, $\mathbf{H}_{1,12}^\mathsf{T} = \left[H_{1,12}^\mathsf{T} | H_{1,123}^\mathsf{T}\right]$, $\mathbf{H}_{2,12}^\mathsf{T} = \left[H_{2,12}^\mathsf{T} | H_{2,123}^\mathsf{T}\right]$ and so on.

Given the set of query keywords S_Q, we prepare the query vector q as described in subsection 4.1. Given query vector q, we wish not only to retrieve relevant items from i-th domain, but also from j-th domain. In the language

of MS-NMF, this is performed by projecting q onto the common subspace matrix $\boldsymbol{W}_{ij}$ to get its representation h in the common subspace. Next, we compute similarity between h and the columns of matrix $\mathbf{H}_{i,ij}$ and $\mathbf{H}_{j,ij}$ (the representation of media items in the common subspace) to find out similar items from medium i and j respectively and the results are ranked based on these similarity scores either individually or jointly (see Algorithm 2).

Algorithm 2 Cross-Social Media Retrieval using MS-NMF.

1: **Input**: data $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$, query q, number of items to be retrieved from medium i, j as N^i and N^j.
2: learn $\boldsymbol{X}_i = \boldsymbol{W}_i \mathbf{H}_i$ for every i using Eqs.(3.10–3.11).
3: set $\epsilon = 10^{-2}$, project q onto $\boldsymbol{W}_{ij}$ to get h by an initialization then looping as below
4: **while** $\left(\|\boldsymbol{W}_{ij} h - q\|_2 \geq \epsilon\right)$ **do**
5: $\quad (h)_a \leftarrow (h)_a \left(\boldsymbol{W}_{ij}^{\mathsf{T}} q\right)_a / \left(\boldsymbol{W}_{ij}^{\mathsf{T}} \boldsymbol{W}_{ij} h\right)_a$
6: **end while**
7: for each item (indexed by r) in medium i with the representation in shared subspace as $\mathbf{H}_{i,ij}(:, r)$, compute its similarity with query projection h as

$$\text{sim}\left(h, \mathbf{H}_{i,ij}(:, r)\right) = \frac{h^{\mathsf{T}} \mathbf{H}_{i,ij}(:, r)}{\|h\|_2 \left\|\mathbf{H}_{i,ij}(:, r)\right\|_2}$$

8: for each item (indexed by r) in medium j, compute $\text{sim}\left(h, \mathbf{H}_{j,ij}(:, r)\right)$ similar to step 7.
9: **Output**: return the top N^i and N^j items in decreasing order of similarities from medium i and j respectively.

5 Experiments

5.1 Datasets We conduct our experiments on a cross-social media dataset consisting of the textual tags of three disparate media genres : *text*, *image* and *video*. To create the dataset, three popular social media websites namely, Blogspot[1], Flickr[2] and YouTube[3], were used. To obtain the data, we first queried all three websites using common concepts - 'Academy Awards', 'Australian Open', 'Olympic Games', 'US Election'. To have pairwise sharing in the data, we additionally queried Blogspot and Flickr with concept 'Christmas', YouTube and Flickr with concept 'Terror Attacks' and Blogspot and YouTube with concept 'Earthquake'. Lastly, to have some individual data of each medium, we queried Blogspot, Flickr and YouTube with concepts 'Cricket World Cup', 'Holi' and 'Global Warming' respectively. Total number of unique tags (M) combined from the three datasets were 3740. Further details of the three datasets are provided in Table 1.

5.2 Parameter Setting We denote YouTube, Flickr and Blogspot *tf-idf* weighted [2] tag-item matrices (similar to widely known term-document matrices generated from the tag-lists) by $\boldsymbol{X}_1$, $\boldsymbol{X}_2$ and $\boldsymbol{X}_3$ respectively. For learning MS-NMF factorization, recall the notation K_v which is dimensionality of the subspace spanned by W_v; following this notation, we use the individual subspace dimensions as $K_1 = 6, K_2 = 8, K_3 = 8$, pair-wise shared subspace dimension as $K_{12} = 15, K_{23} = 18, K_{13} = 12$ and all sharing subspace dimension as $K_{123} = 25$. To learn these parameters, we first initialize them using the heuristic described in subsection 3.3 based on the number of common and individual tags and then do cross-validation based on retrieval precision performance.

5.3 Experiment 1 : Improving Social Media Retrieval using Auxiliary Sources To demonstrate the usefulness of MS-NMF for social media retrieval application, we carry out our experiments in a multi-task learning setting. Focusing on YouTube video retrieval task, we choose YouTube as target dataset while Blogspot and Flickr as auxiliary datasets. To perform retrieval using MS-NMF, we follow Algorithm 1.

Baseline Methods and Evaluation Measures

- The first baseline performs retrieval by matching the query with the tag-lists of videos (using vector-space model) without learning any subspace.
- The second baseline is the retrieval based on standard NMF. The retrieval algorithm using NMF remains similar to the retrieval using MS-NMF as it becomes a special case of MS-NMF when there is no sharing, i.e. $\boldsymbol{W}_1 = W_1, \mathbf{H}_1 = H_{1,1}$ and $R_1 = 56$.
- The third baseline is the recently proposed JS-NMF [7] which learns shared and individual subspaces but allows only one auxiliary source at a time. Therefore, we use two instances of JS-NMF (1) with Blogspot as auxiliary source (2) with Flickr as auxiliary source. Following [7], we obtained the best performance with parameters setting : $R_Y = 56, R_F = 65, R_B = 62$ and $K_{YB} = 37, K_{YF} = 40, K_{BF} = 43$ where R_Y, R_F, R_B are total subspace dimensionalities of YouTube, Flickr and Blogspot respectively and K_{YB}, K_{YF}, K_{BF} are the shared subspace dimensionalities.

To compare above baselines with the proposed MS-NMF, we use *precision-scope* (P@N), *mean average precision* (MAP) and *11-point interpolated precision-recall* [2]. The performance of MS-NMF is compared with the baselines by averaging the retrieval results over a query set of 20 concepts given by $\mathbb{Q} = \{$'beach', 'america', 'bomb', 'animal', 'bank',

[1] http://www.blogger.com/
[2] http://www.flickr.com/services/api/
[3] http://code.google.com/apis/youtube/overview.html

Table 1: Description of Blogspot, Flickr and YouTube data sets.

	Dataset Size	Concepts Used for Creating Dataset	Avg-Tags/Item (rounded)
Blogspot	10000	'Academy Awards', 'Australian Open', 'Olympic Games', 'US Election', 'Cricket World Cup', 'Christmas', 'Earthquake'	6
Flickr	20000	'Academy Awards', 'Australian Open', 'Olympic Games', 'US Election', 'Holi', 'Terror Attacks', 'Christmas'	8
YouTube	7000	'Academy Awards', 'Australian Open', 'Olympic Games', 'US Election', 'Global Warming', 'Terror Attacks', 'Earthquake'	7

'movie', 'river', 'cable', 'climate', 'federer', 'disaster', 'elephant', 'europe', 'fire', 'festival', 'ice', 'obama', 'phone', 'santa', 'tsunami'}.

Experimental Results Figure 2 compares the retrieval performance of MS-NMF with the three baselines in terms of evaluation criteria mentioned above. It can be seen from Figure 2 that MS-NMF clearly outperforms the baselines in terms of all three evaluation criteria. Since tag based matching method does not learn any subspaces, its performance suffers from the 'polysemy' and 'synonymy' problems prevalent in tag space. NMF, being a subspace learning method, performs better than tag based method but does not perform better than shared subspace methods (JS-NMF and MS-NMF) as it is unable to exploit the knowledge from auxiliary sources. When comparing JS-NMF with MS-NMF, we see that MS-NMF clearly outperforms both the settings of JS-NMF. This is due to the the fact that JS-NMF is limited to work with only one auxiliary source and can not exploit the knowledge available in multiple data sources. Although, JS-NMF, using one auxiliary source at a time, improves the performance over NMF but real strength of the three media sources is exploited by MS-NMF which performs the best among all methods. Better performance achieved by MS-NMF can be attributed to the shared subspace model finding better term co-occurrences and reducing the tag subjectivity by exploiting knowledge across three data sources. Further insight into the improvement is provided through entropy and impurity results given in subsection 5.5.

5.4 Experiment 2 : Cross-Social Media Retrieval For cross-media retrieval experiments, we use the same dataset as used in our first experiment but choose more appropriate baselines and evaluation measures. Subspace learning using MS-NMF remains same, as the factorization is carried out on the same dataset using the same parameter setting. We follow Algorithm 2 which utilizes MS-NMF framework to return the ranked list of cross-media items.

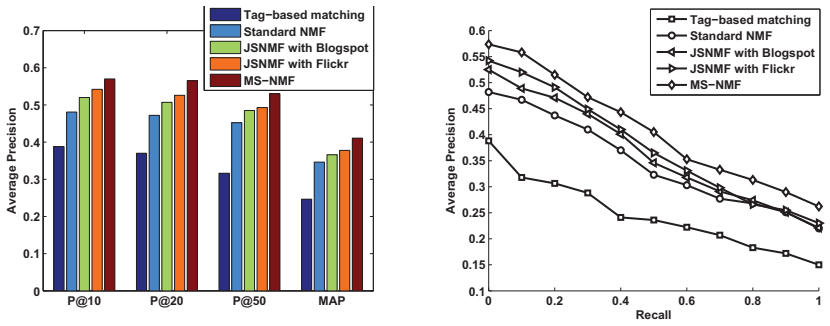
Baseline Methods and Evaluation Measures To see the effectiveness of MS-NMF for cross-media retrieval, the *first* baseline is tag-based matching performed in a typical vector-space model setting. The *second* baseline is the framework in [12] where a subspace is fully shared among three media without retaining any individual subspace. We shall denote this baseline as LIN_ETAL09. We present cross-media results for both pair-wise and across all three media. When presenting pair-wise results, we choose JS-NMF [7] (subspace learning remains same as in the first experiment) as a *third* baseline by applying it on the media pairs.

To evaluate our cross-media algorithm, we again use P@N, MAP and 11-point interpolated precision-recall measures. To explicitly state these measures for cross-media retrieval, we define precision and recall in cross-media scenario. Consider a query term $q \in \mathbb{Q}$, let its ground truth set be G_i for i-th medium. If a retrieval method used with query q results in an answer set A_i from i-th medium, the precision and recall measures across n media are defined as

$$\text{(5.15)}\quad \text{Precision} = \frac{\sum_{i=1}^{n} |A_i \cap G_i|}{\sum_{i=1}^{n} |A_i|}, \text{Recall} = \frac{\sum_{i=1}^{n} |A_i \cap G_i|}{\sum_{i=1}^{n} |G_i|}$$

Experimental Results Cross-media retrieval results across media pairs are shown in Figure 3 whereas those from across all three media (Blogspot, Flickr and YouTube) are shown in Figure 4. To generate the graphs, we average the retrieval results over the *same* query set $\mathbb{Q}$ as defined for YouTube retrieval task in subsection 4.1. It can be seen from Figure 3 that MS-NMF significantly outperforms all baselines including JS-NMF on cross-media retrieval task for each media-pair. This performance improvement is consistent in terms of all three evaluation measures. Note that, to learn the subspaces, MS-NMF uses all three media data whereas JS-NMF uses the data only from the media pair being considered. The ability to exploit knowledge from multiple media helps MS-NMF achieving better performance. When retrieval precision and recall are calculated across all three media domains, MS-NMF still performs better than the tag-based matching as well as LIN_ETAL09. Note that JS-NMF can not be applied on three media simultaneously.

5.5 Topical Analysis To provide further insights into the benefits achieved by MS-NMF, we examine the results at the topical level. Every basis vector of the subspace (when



(a) (b)

Figure 2: YouTube retrieval results with Flickr and Blogspot as auxiliary sources (a) Precision-Scope and MAP (b) 11-point interpolated Precision-Recall; for tag-based matching (baseline 1), standard NMF (baseline 2), JS-NMF [7] with Blogspot (baseline 3a); with Flickr (baseline 3b) and proposed MS-NMF.

normalized to sum one) can be interpreted as a topic. We define a metric for measuring the *impurity* of a topic as

$$P(T) = \frac{1}{L(L-1)} \sum_{\substack{x,y \\ x \neq y}} \text{NGD}(t_x, t_y) \tag{5.16}$$

where L denotes the number of tags in a topic T for which corresponding basis vector element greater than a threshold[4] and $\text{NGD}(t_x, t_y)$ is Normalized Google Distance [4] between tags t_x and t_y.

We compute the entropy and impurity for each subspace basis and plot their distributions in Figure 5 using the boxplots. It can be seen from the figure that topics learnt by MS-NMF have on average lesser entropy and impurity than their NMF and LIN_ETAL09 counterparts for all three datasets. Although, LIN_ETAL09 can model multiple data sources but it uses a single subspace to model each source without retaining their differences. As a consequence of this, the variabilities of the three sources get averaged out and thereby increase the entropy and impurity of the resulting topics. In contrast, MS-NMF having the flexibility of partial sharing, averages the commonalities of three data sources only up to their true sharing extent and thus results in purer and compact (less entropy) topics.

6 Conclusion and Future Works

We have presented a matrix factorization framework to learn individual and shared subspaces from multiple data sources (MS-NMF) and demonstrated its application to two social media problems: improving social media retrieval by leveraging related data from auxiliary sources and cross-media retrieval. We provided an efficient algorithm to learn the joint factorization and proved its convergence. Our first application has demonstrated that MS-NMF can help improving retrieval in YouTube by transferring knowledge from the tags of Flickr and Blogspot. Outperforming JS-NMF [7], it justifies the need for a framework which can simultaneously model multiple data sources with any arbitrary sharing. The second application shows the utility of MS-NMF for cross-media retrieval by demonstrating its superiority over existing methods using Blogspot, Flickr and YouTube dataset. The proposed framework is quite generic and has potentially wider applicability in cross-domain data mining e.g. cross-domain collaborative filtering, cross-domain sentiment analysis etc. In current form, MS-NMF requires the shared and individual subspace dimensionalities to be obtained using cross-validation. As a future work, we shall formulate the joint factorization probabilistically by appealing to Bayesian nonparametric theory and infer these parameters automatically from the data.

[4] fixed at 0.05 for selecting the tags with more than 5% weight in a topic

References

[1] R.K. Ando and T. Zhang. A framework for learning predictive structures from multiple tasks and unlabeled data. *The Journal of Machine Learning Research*, 6:1817–1853, 2005.

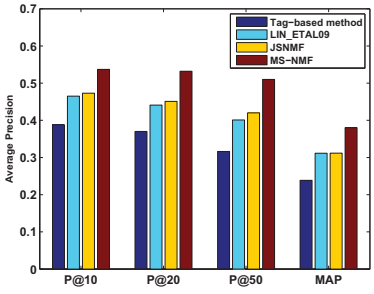
[2] R. Baeza-Yates, B. Ribeiro-Neto, et al. *Modern information retrieval*. Addison-Wesley Reading, MA, 1999.

[3] M.W. Berry and M. Browne. Email surveillance using non-negative matrix factorization. *Computational & Mathematical Organization Theory*, 11(3):249–264, 2005.

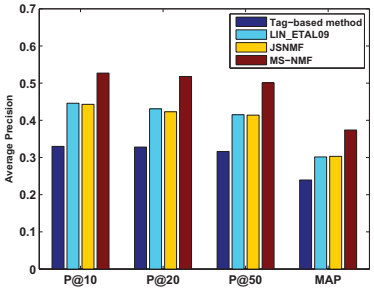
[4] R.L. Cilibrasi, P.M.B. Vitanyi, and A. CWI. The google similarity distance. *IEEE Transactions on Knowledge and Data Engineering*, 19(3):370–383, 2007.

[5] S.A. Golder and B.A. Huberman. Usage patterns of collaborative tagging systems. *Journal of Information Science*, 32(2):198, 2006.

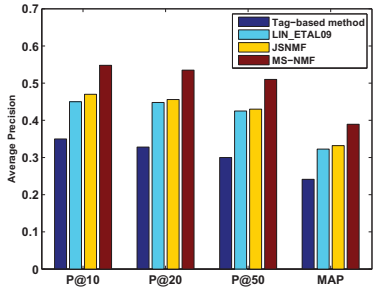
[6] Q. Gu and J. Zhou. Learning the shared subspace for multi-task clustering and transductive transfer classification. *ICDM*, pages 159–168, 2009.



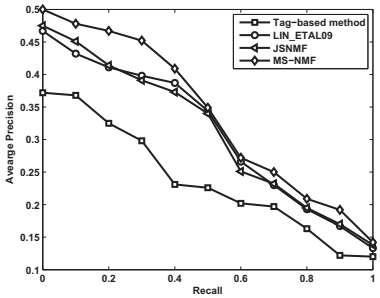
(a) precision-scope/MAP (Blogspot-Flickr)



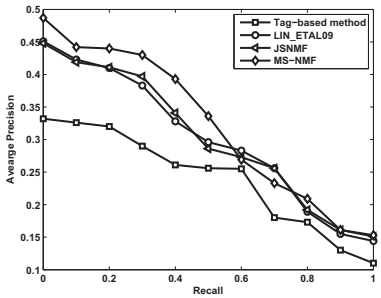
(b) precision-scope/MAP (Blogspot-YouTube)



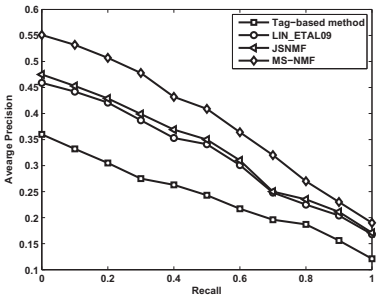
(c) precision-scope/MAP (Flickr-YouTube)



(d) 11-point precision-recall (Blogspot-Flickr)



(e) 11-point precision-recall (Blogspot-YouTube)



(f) 11-point precision-recall (Flickr-YouTube)

Figure 3: *Pairwise* Cross-media retrieval results : Blogspot-Flickr (first column), Blogspot-YouTube (second column) and Flickr-YouTube (third column); for tag-based matching (baseline 1), LIN_ETAL09 [12] (baseline 2), JS-NMF [7] (baseline 3) and MS-NMF.

[7] S.K. Gupta, D. Phung, B. Adams, T. Tran, and S. Venkatesh. Nonnegative shared subspace learning and its application to social media retrieval. *SIGKDD*, pages 1169–1178, 2010.

[8] S. Ji, L. Tang, S. Yu, and J. Ye. A shared-subspace learning framework for multi-label classification. *ACM Transactions on Knowledge Discovery from Data*, 4(2):1–29, 2010.

[9] M.S. Kankanhalli and Y. Rui. Application potential of multimedia information retrieval. *Proceedings of the IEEE*, 96(4):712–720, 2008.

[10] D.D. Lee and H.S. Seung. Algorithms for non-negative matrix factorization. *Advances in Neural Information Processing Systems*, 13, 2001.

[11] C.J. Lin. Projected gradient methods for nonnegative matrix factorization. *Neural Computation*, 19(10):2756–2779, 2007.

[12] Y.R. Lin, H. Sundaram, M. De Choudhury, and A. Kelliher. Temporal patterns in social media streams: Theme discovery and evolution using joint analysis of content and context. In *ICME*, pages 1456–1459, 2009.

[13] K. V. Mardia, J. M. Bibby, and J. T. Kent. *Multivariate analysis*. Academic Press, New York, 1979.

[14] C. Marlow, M. Naaman, D. Boyd, and M. Davis. Ht06, tagging paper, taxonomy, flickr, academic article, toread. *Proceedings Hypertext*, pages 31–40, 2006.

[15] F. Shahnaz, M.W. Berry, V.P. Pauca, and R.J. Plemmons. Document clustering using nonnegative matrix factorization. *Information Processing and Management*, 42(2):373–386, 2006.

[16] S. Si, D. Tao, and B. Geng. Bregman divergence based regularization for transfer subspace learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(7):929–942, 2009.

[17] B. Sigurbjörnsson and R. Van Zwol. Flickr tag recommendation based on collective knowledge. *WWW*, pages 327–336, 2008.

[18] Wei Xu, Xin Liu, and Yihong Gong. Document clustering based on non-negative matrix factorization. *SIGIR*, pages 267–273, 2003.

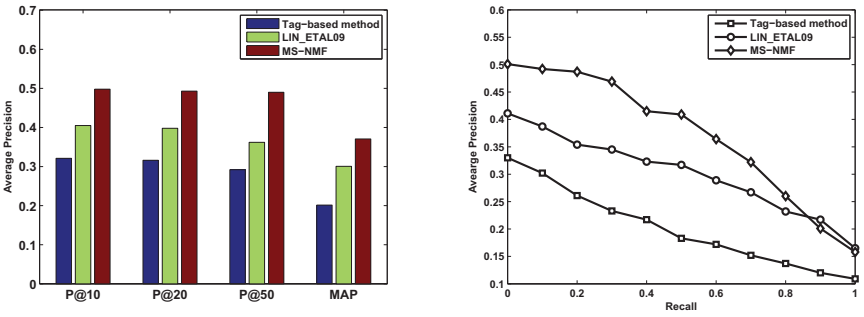
[19] R. Yan, J. Tesic, and J.R. Smith. Model-shared subspace boosting for multi-label classification. *SIGKDD*, pages 834–843, 2007.

[20] Y. Yang, D. Xu, F. Nie, J. Luo, and Y. Zhuang. Ranking with local regression and global alignment for cross media retrieval. *MM*, pages 175–184, 2009.

[21] Y. Yi, Y.T. Zhuang, F. Wu, and Y.H. Pan. Harmonizing hierarchical manifolds for multimedia document semantics understanding and cross-media retrieval. *Language*, 1520:9210, 2008.

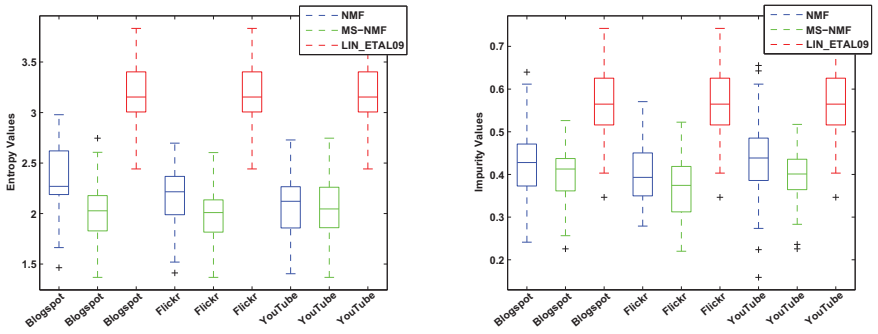
A Appendix

Proof of Convergence We prove the convergence of multiplicative updates given by Eqs (3.10-3.11). Due to space restriction, we only provide a sketch of the proof. Following [10], the auxiliary function $G(w, w^t)$ is defined as an upper bound function for $J(w^t)$. For our MS-NMF case, we prove the following lemma extended from [10]:



(a) precision-scope/MAP (b) 11-point precision-recall curve

Figure 4: Cross-media retrieval results plotted across *all three data sources* (Blogspot, Flickr and YouTube) for tag-based matching (baseline 1), LIN_ETAL09 [12] (baseline 2) and proposed MS-NMF.



(a) entropy distribution (b) impurity distribution

Figure 5: A comparison of MS-NMF with NMF and LIN_ETAL09 [12] in terms of entropy and impurity distributions.

Lemma. *If* $(W_v)_p$ *is p-th row of matrix* W_v, $v \in S(n,i)$ *and* $C\left((W_v)_p\right)$ *is the diagonal matrix with its* $(l,k)^{th}$ *element*

$$C_{lk}\left((W_v)_p\right) = \mathbf{1}_{l,k}\frac{\left(\sum_{i\in v}\lambda_i H_{i,v}\left(\sum_{u\in S(n,i)} H^{\mathsf{T}}_{i,u}\,(W_u)_p\right)\right)_l}{(W_v)_{pl}}$$

then

$$\begin{aligned} G\left((W_v)_p, (W_v)^t_p\right) &= J\left((W_v)^t_p\right) \\ &+ \left((W_v)_p - (W_v)^t_p\right)^{\mathsf{T}} \nabla_{(W_v)^t_p} J\left((W_v)^t_p\right) \\ &+ \frac{1}{2}\left((W_v)_p - (W_v)^t_p\right)^{\mathsf{T}} C\left((W_v)^t_p\right)\left((W_v)^t_p - (W_v)^t_p\right) \end{aligned}$$

is an auxiliary function for $J\left((W_v)^t_p\right)$*, cost function defined for p-th row of the data.*

Proof. The second derivative of $J\left((W_v)^t_p\right)$ i.e. $\nabla^2_{(W_v)^t_p} J\left((W_v)^t_p\right) = \sum_{i\in v}\lambda_i H_{i,v} H^{\mathsf{T}}_{i,v}$. Comparing the expression of $G\left((W_v)_p, (W_v)^t_p\right)$ in the lemma with the Taylor series expansion of $G\left((W_v)_p, (W_v)^t_p\right)$ at $(W_v)^t_p$, it can be seen that all we need to prove is the following

$$\left((W_v)_p - (W_v)^t_p\right)^{\mathsf{T}} T_{W_v}\left((W_v)^t_p - (W_v)^t_p\right) \geq 0$$

where $T_{W_v} \triangleq C\left((W_v)^t_p\right) - \sum_{i\in v}\lambda_i H_{i,v} H^{\mathsf{T}}_{i,v}$. Similar to [10], instead of showing it directly, we show the positive definiteness of matrix E with elements

$$E_{lk}\left((W_v)^t_p\right) = \left((W_v)_p - (W_v)^t_p\right)^{\mathsf{T}}_l (T_{W_v})_{lk}\left((W_v)^t_p - (W_v)^t_p\right)_k$$

For positive definiteness of matrix E, we have to show that for every nonzero z, the value of $z^{\mathsf{T}} M z$ is positive. To avoid lengthy derivation, we only show main step here :

$$\begin{aligned} z^{\mathsf{T}} M z &= \sum_{l,k} z_l\,(W_v)^t_{pl}\,(T_{W_v})_{lk}\,(W_v)^t_{pk}\,z_k \\ &= \sum_{l,k} z_l^2\,(W_v)^t_{pl}\left(\sum_{u\in S(n,i),u\neq v} H^{\mathsf{T}}_{i,u}\,(W_u)_p\right)_l \\ &+ \lambda\sum_{l,k}(W_v)^t_{pl}\left(\sum_{i\in v}\lambda_i\left(H_{i,v}H^{\mathsf{T}}_{i,v}\right)_{lk}\right)(W_v)^t_{pk}\,\frac{(z_l - z_k)^2}{2} \geq 0 \end{aligned}$$

□

At the local minimum of $G\left((W_v)_p, (W_v)^t_p\right)$ for iteration (t), by comparing $\nabla_{(W_v)^t_p} G\left((W_v)_p, (W_v)^t_p\right)$ with gradient-descent update of Eq (3.8), we get the step size $\eta_{(W_v)^t_{lk}}$ as in equation (3.9).