Constrained NMF-Based Multi-View Clustering on Unmapped Data

Xianchao Zhang, Linlin Zong, Xinyue Liu, Hong Yu
School of Software Dalian University of Technology, Dalian 116620, China
xczhang@dlut.edu.cn, zonglinlin324@gmail.com, xyliu@dlut.edu.cn, hongyu@dlut.edu.cn

Abstract

Existing multi-view clustering algorithms require that the data is completely or partially mapped between each pair of views. However, this requirement could not be satisfied in most practical settings. In this paper, we tackle the problem of multi-view clustering for unmapped data in the framework of NMF based clustering. With the help of inter-view constraints, we define the disagreement between each pair of views by the fact that the indicator vectors of two instances from two different views should be similar if they belong to the same cluster and dissimilar otherwise. The overall objective of our algorithm is to minimize the loss function of NMF in each view as well as the disagreement between each pair of views. Experimental results show that, with a small number of constraints, the proposed algorithm gets good performance on unmapped data, and outperforms existing algorithms on partially mapped data and completely mapped data.

Introduction

Multi-view clustering gains increasing attention in the past decade (Bickel and Scheffer 2004) (Kumar and III 2011) (Kumar, Rai, and III 2011) (Liu et al. 2013) (Blaschko and Lampert 2008) (Chaudhuri et al. 2009) (Tzortzis and Likas 2012). Most existing multi-view clustering algorithms require that the data is completely mapped, i.e., every object has representations in all the views, representations from different views representing a same object are exactly known, and the representations of the same object have the same index in different views. However, this requirement could not be satisfied in most practical settings. Since data from different views are usually collected, processed, and stored independently, it is hard to ensure the complete mapping. For example, a same news maybe reported by many reporters from different views, thus a period's news could be clustered from multiple views. However, a news may not be reported by all the concerned news agencies, and the same news may be named with different titles and reported in different ways (even in different languages), thus it will need much effort to identify which reports from different agencies represent the same news. Also, the news agencies have no consistency in indexing the news, since they may regard different news as important ones.

Recently, (Eaton, desJardins, and Jacob 2012) proposed a constrained multi-view clustering algorithm that deal with incompletely mapped data with the help of intra-view constraints. This algorithm propagates constraints in each view to those instances that can be mapped to the other views. Nevertheless, partial mapping is still necessary in this work. And the more partial mapping provided, the better the algorithm performs.

In this paper, based on the fact that a small number of constraints could be easily obtained in many real applications (Kulis et al. 2009) (Lu and Peng 2013), we propose a constrained multi-view clustering algorithm for unmapped data in the framework of NMF (Nonnegative Matrix Factorization) based clustering. We use inter-view constraints (Lu and Peng 2013) to establish the connections between different views. We define the disagreement between each pair of views by the fact that the indicator vectors of two instances from different views should be similar if they belong to the same cluster and dissimilar otherwise. We use the disagreement between the views to guide the factorization of the matrices. The overall objective of our algorithm is to minimize the loss function of NMF in each view as well as the disagreement between each pair of views. Experimental results show that, with a small number of constraints, the proposed CMVNMF (Constrained Multi-View clustering based on NMF) algorithm gets good performance on unmapped data, and outperforms existing algorithms on partially mapped data and completely mapped data.

NMF and Clustering

NMF (Lee and Seung 2001) aims to factorize a matrix into two or more non-negative matrices whose product approximates to the original matrix. It emphasizes the non-negativity of the factor matrices. Let $X = [x_1, \ldots, x_n] \in R_+^{d\times n}$ be the original matrix, each data point $x_i (1 \le i \le n)$ is d-dimensional. The basic form of NMF tries to optimize the following problem:

$$\min_{U,V} \|X - UV^T\|_F^2 \quad s.t. \quad U \ge 0, V \ge 0$$

where $U \in R_+^{d\times K}$, $V \in R_+^{n\times K}$, $\|.\|_F$ is the Frobenius norm. Obviously, the objective function is not convex to both U

and V, it's hard to find its global minima. However, there are plenty of approaches that can be used to obtain its local minima, one possible solution is to iteratively execute the following rules in eq.(1). Also, some fast algorithms are proposed for near-separable NMF (Recht et al. 2012) (Gillis and Vavasis 2013) (Kumar, Sindhwani, and Kambadur 2013).

$$U_{ij} \leftarrow U_{ij}\frac{(XV)_{ij}}{(UV^TV)_{ij}}, V_{ij} \leftarrow V_{ij}\frac{(X^TU)_{ij}}{(VU^TU)_{ij}} \quad (1)$$

NMF can be used for clustering (Ding and He 2005) (Kuang, Park, and Ding 2012) (Li and Ding 2006), if each cluster can be represented by a single basis vector and different clusters correspond to different basis vectors. As in (Liu et al. 2013), NMF can also be written as

$$x_j \approx U(V_{j,.})^T = \sum_{k=1}^{K} U_{.,k}V_{j,k}$$

where $U_{.,k}$ is the k-th column vector of U and $V_{j,.}$ is the j-th row vector of V. If K represents the number of clusters, $U_{.,k}$ can be seen as the centroid of the k-th cluster and $V_{j,.}$ is the indicator vector, i.e., the instance belongs to the cluster having largest value in the indicator vector.

The CMVNMF Algorithm

Objective Function

We are given a dataset X with v views. Let $X^{(a)} = \{x_1^{(a)}, x_2^{(a)}, \cdots, x_{n_a}^{(a)}\}$ denote the set of instances, where n_a is the number of its instances in view a. If X is a mapped dataset, $x_i^{(a)}$ and $x_i^{(b)}$ represent the same instance, else, $x_i^{(a)}$ and $x_i^{(b)}$ may represent different instances. In an unmapped dataset, the number of instances may also be different in different views, i.e., $n_a \neq n_b$. We aim to deal with the multi-view clustering problem on unmapped data.

We are also given a set of must-link constraints $ML^{a,b}$ between view a and view b in which $(x_i^{(a)}, x_j^{(b)}) \in ML^{a,b}$ means that $x_i^{(a)}$ and $x_j^{(b)}$ should belong to the same cluster, and a set of cannot-link constraints $CL^{a,b}$ in which $(x_i^{(a)}, x_j^{(b)}) \in CL^{a,b}$ means that $x_i^{(a)}$ and $x_j^{(b)}$ should belong to different clusters. A constraint is intra-view if $a = b$, and inter-view otherwise.

The main challenge of multi-view clustering is to explore the relationship between each pair of the views. Since the data has v views, we have v indicator matrixes. The indicator vectors of two instances from two different views should be similar if they belong to the same cluster and dissimilar otherwise. So we establish the relationship between two different views by minimizing the deviation of indicator vectors if two instances from the two views belong to the same cluster and maximizing the deviation if they belong to two different clusters. We use inter-view constraints to accomplish the task and use the following function to measure the disagreement between the indicator vectors from view a and view b.

$$\begin{aligned} \Delta_{a,b} &= \sum_{(x_i^{(a)}, x_j^{(b)}) \in ML^{a,b}} (V_i^{(a)} - V_j^{(b)})^2 \\ &+ 2 \sum_{(x_i^{(a)}, x_j^{(b)}) \in CL^{a,b}} V_i^{(a)} V_j^{(b)} \\ s.t. \quad & i \in [1, n_a], \quad j \in [1, n_b], \\ & a \in [1, v], \quad b \in [1, v], \quad a > b \end{aligned}$$

where $V_i^{(a)}$ is the indicator vector of the i-th instance in view a and $V_j^{(b)}$ is the indicator vector of the j-th instance in view b. We set the coefficient of cannot-link to 2 to simplify the iteration of V in eq.(4).

We use the disagreement between the views to guide the factorization of the matrices. The overall objective of our algorithm is to minimize the loss function of NMF in each view as well as the disagreement between each pair of views. The objective function is as follows, where β is the regularization parameter controlling importance of different parts.

$$\begin{aligned} \phi &= \sum_a \|X^{(a)} - U^{(a)}V^{(a)T}\|_F^2 + \beta \sum_{a,b \in [1,v], a>b} \Delta_{a,b} \\ & s.t. \quad U^{(a)} \geq 0, V^{(a)} \geq 0 \end{aligned} \quad (2)$$

Solution

Denote matrix $M^{a,b} \in R^{n_a \times n_b}$ and $C^{a,b} \in R^{n_a \times n_b}$,

$$M_{ij}^{a,b} = \begin{cases} 1, & (x_i^{(a)}, x_j^{(b)}) \in ML^{a,b} \\ 0, & \text{otherwise.} \end{cases}$$

$$C_{ij}^{a,b} = \begin{cases} 1, & (x_i^{(a)}, x_j^{(b)}) \in CL^{a,b} \\ 0, & \text{otherwise.} \end{cases}$$

$\Delta_{a,b}$ can be rewritten as the form of matrix.

$$\begin{aligned} \Delta_{a,b} &= tr(V^{(a)T}D^{a,b}V^{(a)}) + tr(V^{(b)T}D^{b,a}V^{(b)}) \\ & -2tr(V^{(a)T}M^{a,b}V^{(b)}) + 2tr(V^{(a)T}C^{a,b}V^{(b)}) \end{aligned}$$

where $tr(.)$ is the trace function. $D^{a,b}$ and $D^{b,a}$ are the diagonal matrix, and their ii-th element are separately the sum of the i-th row of $M^{a,b}$ and the i-th column of $M^{a,b}$. We solve $U^{(a)}$ and $V^{(a)}$ by the following multiplicative updating rules (for brevity, we use X, U, V to represent $X^{(a)}, U^{(a)}, V^{(a)}$) in the following.

Fixing $V^{(a)}$, computing $U^{(a)}$ In this case, the second part of eq.(2) is a constant, and the solution of $U^{(a)}$ is the same with that of NMF.

$$U_{ij} \leftarrow U_{ij}\frac{(XV)_{ij}}{(UV^TV)_{ij}} \quad (3)$$

Fixing $U^{(a)}$, $V^{(b)}$, computing $V^{(a)}$ At first, we introduce the definition and lemma of auxiliary function (Lee and Seung 2001).

Definition 1 *(Lee and Seung 2001) $G(H, H^t)$ is an auxiliary function for $\Phi(H)$ if the conditions*

$$G(H, H^t) \geq \Phi(H), G(H^t, H^t) = \Phi(H^t)$$

are satisfied.

Lemma 1 *(Lee and Seung 2001) If G is an auxiliary function, then $\Phi(H)$ is non-increasing under the update rule*

$$H^{t+1} = \arg\min_H G(H, H^t)$$

The key process is to find the auxiliary function for eq.(2). Like (Cai et al. 2008), we define the auxiliary function as

$$\begin{aligned} G(V_{ij}, V_{ij}^t) &= \phi(V_{ij}^t) + \phi'(V_{ij}^t)(V_{ij} - V_{ij}^t) \\ &+ \Omega(V_{ij}^t)(V_{ij} - V_{ij}^t)^2 \end{aligned}$$

where $\varphi = D^{a,b}V^t + C^{a,b}V^{(b)}$,

$$\begin{aligned} \phi'(V_{ij}^t) &= 2(V^tU^TU - X^TU)_{ij} \\ &+ 2\beta \sum_{b=1,b\neq a}^{v} (\varphi - M^{a,b}V^{(b)})_{ij} \\ \Omega(V_{ij}^t) &= \frac{(V^tU^TU)_{ij} + \beta \sum_{b=1,b\neq a}^{v} \varphi_{ij}}{V_{ij}^t} \end{aligned}$$

It's easy to verify that if $V_{ij}^t = V_{ij}$, then $G(V_{ij}, V_{ij}^t) = \phi(V_{ij}^t)$ is obviously. We only need to prove $G(V_{ij}, V_{ij}^t) \geq \phi(V_{ij})$ if $V_{ij}^t \neq V_{ij}$. We compare the Taylor expansion of $\phi(V_{ij})$

$$\phi(V_{ij}) = \phi(V_{ij}^t) + \phi'(V_{ij}^t)(V_{ij} - V_{ij}^t) + \frac{\phi''(V_{ij}^t)}{2}(V_{ij} - V_{ij}^t)^2$$

with $G(V_{ij}, V_{ij}^t)$ and

$$\phi''(V_{ij}^t) = 2(U^TU)_{jj} + 2\beta \sum_{b=1,b\neq a}^{v} D_{ii}^{a,b}$$

Since

$$(V^tU^TU)_{ij} = \sum_{s=1}^{k} V_{is}^t (U^TU)_{sj} \geq V_{ij}^t (U^TU)_{jj}$$

and

$$\varphi_{ij} = \sum_{s=1}^{n_a} (D_{is}^{a,b} V_{sj}^t) + (C^{a,b}V^{(b)})_{ij} \geq D_{ii}^{a,b} V_{ij}^t$$

thus $G(V_{ij}, V_{ij}^t) \geq \phi(V_{ij})$ holds. Replacing $G(H, H^t)$ in eq.(1) by eq.(4) results in the update rule:

$$\begin{aligned} V_{ij} &= -\frac{-2\Omega(V_{ij}^t)V_{ij}^t + \phi'(V_{ij}^t)}{2\Omega(V_{ij}^t)} \qquad (4) \\ &= V_{ij}^t \frac{(X^TU)_{ij} + \beta \sum_{b=1,b\neq a}^{v} (M^{a,b}V^{(b)})_{ij}}{(V^tU^TU)_{ij} + \beta \sum_{b=1,b\neq a}^{v} \varphi_{ij}} \end{aligned}$$

Algorithm

Summarizing the former analysis, we give the algorithm framework (names as CMVNMF) in Algorithm 1. We initialize the algorithm by the result of a basic clustering method, e.g. *k*-means, in each view. We normalize the dataset in each view at first. CMVNMF will work in the case that *V* in each view is comparable. Analyzing the updating rule of *U* and *V* , it can be seen that if *X* and the initializations of *U* and *V* are comparable, *V* will be comparable.

Algorithm 1: *CMVNMF algorithm*

Input:
$X^{(a)} = \{x_1^{(a)}, x_2^{(a)}, \cdots, x_{n_a}^{(a)}\}$: the multi-view data sets
$M^{a,b}$: matrix represent the must-link constraints between view a and view b
$C^{a,b}$: matrix represent the cannot-link constraints between view a and view b
k: number of clusters
Output:
$label^{(a)}$: the labels of each instance in view a
For every view a, normalize $X^{(a)}$;
For every two view a and b, compute $D^{a,b}$;
Initialize $U^{(a)}$, $V^{(a)}$;
repeat
  for $a = 1$ *to* v **do**
    Fixing $V^{(a)}$, update $U^{(a)}$ by eq.(3);
    Fixing $U^{(a)}$ and $V^{(b)}$, update $V^{(a)}$ by eq.(4);
  end
until $eq.(2)$ *is converged*;
Label each data point $x_i^{(a)}$ using $label_i^{(a)}$,
$label_i^{(a)} = \arg\max_{1\leq j\leq k} V_{ij}^{(a)}$.

Given a dataset $X^{(a)} \in R^{d_a \times n_a}$ with k clusters, the complexity of updating $U^{(a)}$ is $O(d_a n_a k)$, which is the same with NMF. In terms of the updating of $V^{(a)}$, CMVNMF needs more calculation than NMF. The complexity of updating $V^{(a)}$ is $O(d_a n_a k + \sum_{b=1,b\neq a}^{v} n_a n_b k)$. The more views, the more multiplication operation. This is reasonable because CMVNMF tries to learn from all views.

Discussion

CMVNMF uses the same basic NMF framework as (Liu et al. 2013) (Liu and Han 2013). However, the work in (Liu et al. 2013) (Liu and Han 2013) uses a regularization term between different views, which can only work on mapped data. CMVNMF uses inter-view constraints to build connections between different views and can work on mapped data, partially mapped data and unmapped data.

We separately use squared error to penalize the deviation of indication vectors on inter-view must-link constraints and use dot product to penalize the deviation of indication vectors on inter-view cannot-link constraints. This is because that they have opposite impacts, thus the different penalization ensures the non-negativity of each part in the objective function.

Note that $\Delta_{a,b}$ can also utilize intra-view constraints to indicate the deviation of indication vectors of two instances in the same view, this can be realized be setting $a = b$. However, with the same number of constraints, mixed inter-view constraints and intra-view constraints do not provide as much as inter-view connection information as pure inter-view constraints.

Experiments

Experiments on Unmapped Data

Datasets We use three benchmark datasets, UCI Handwritten Digit[1], Reuters[2] and Webkb[3] to investigate the impact of inter-view constraints on CMVNMF. UCI Handwritten consists of handwritten digits (0-9). We experiment on 3 views: 76 Fourier coefficients of the character shapes, 240 pixel averages and 47 Zernike moments. Webkb is composed of web pages collected from computer science department websites of four universities: Cornell, Texas, Washington and Wisconsin. The web pages are classified into 7 categories. Here, we choose four most populous categories (course, faculty, project, student) for clustering. A web page is made of three views: the text on it, the anchor text on the hyperlinks pointing to it and the text in its title. Reuters contains 1200 documents over the 6 labels. Each sample is translated into five languages. We experiment on the English, French and German views. UCI Handwritten, Reuters and Webkb are all benchmarks for traditional multi-view clustering algorithms, thus they are arranged with complete mapping. To get unmapped datasets, we randomly select 95% samples on each view, then permute each view and regardless the new indices of representations of an object in all the views.

Settings As both inter-view must-link constraints and inter-view cannot-link constraints are helpful in our method, in these experiments and the following experiments we use both of the two type of constraints[4]. We get the inter-view constraints by randomly selecting representations of two objects from different views, judge whether they belong to the same ground truth cluster or not, and set them must-link or cannot-link. We set $\beta = 1$ in all experiments. We use accuracy (ACC) and normalized mutual information (NMI) (Xu, Liu, and Gong 2003) to measure the performance. To avoid randomness, we conduct the algorithms 20 runs with different initializations and report the average results.

Results Figure 1 $\sim$ Figure 3 show the clustering performances with increasing the number of inter-view constraints between any two views by using 0.5% to 10% constraints with step 0.5%. From the results it can be seen that the performance of CMVNMF generally increases with the increasing of the number of constraints. From 0.5% constraints to some inflection point between 1% to 3%, the performance increases sharply with the number of constraints. After the inflection point, the increasing slows down, and finally the performance becomes stable. This is because that inter-view constraints can transfer the information across views and guide the clustering process. When the number of constraints reaches the inflection point, they can provide enough mutual information to guide multi-view clustering.

[1]http://archive.ics.uci.edu/ml/datasets/Multiple+Features

[2]http://membres-liglab.imag.fr/grimal/data.html

[3]http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-20/www/data/

[4]We also try using only must-links and cannot-links, but these settings perform worth than using both, due to space limitation, we do not report these results.

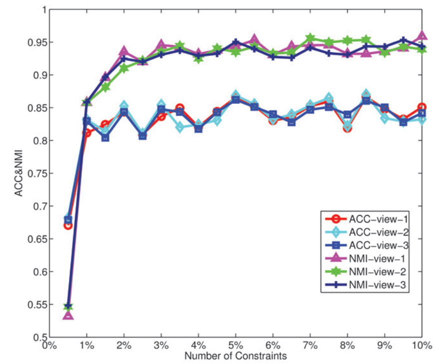


Figure 1: ACC and NMI on Handwritten w.r.t Number of Constraints

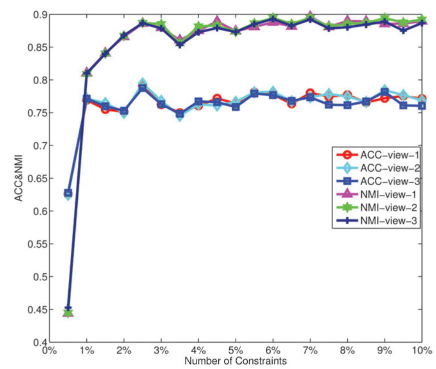


Figure 2: ACC and NMI on Reuters w.r.t Number of Constraints

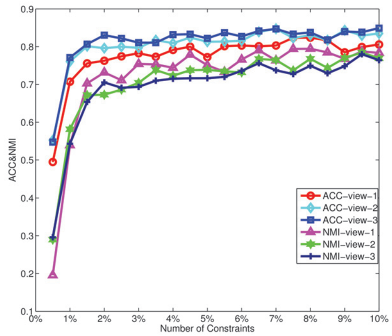
Experiments on Partially Mapped Data

Datasets We use two benchmark datasets, 3-Sources[5] and Wikipedia (Rasiwasia et al. 2010) to demonstrate the effectiveness of CMVNMF on partially mapped data. 3-Sources is collected from three online news sources. There are totally 948 news stories covering 416 distinct ones. Some stories are not reported by all three sources, thus 3-Sources is originally partially mapped. The Wikipedia benchmark dataset contains 2800 documents derived from Wikipedia. Each document is actually a text-image pair, annotated with a label from the vocabulary of 10 semantic classes. Wikipedia is completely mapped, and we randomly select 2000 documents in each view and make 70% of them mapped.

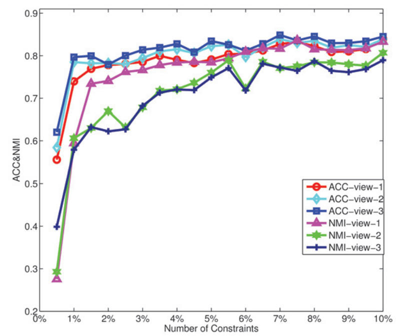
Settings We compare CMVNMF with Constraint Prop (Eaton, desJardins, and Jacob 2012), which is designed for partially mapped data. For CMVNMF, we use 5% inter-view constraints. For Constraint Prop, we use 5% intra-view constraints, and set the parameter t using cross validation.

Results The results are shown in Table1, which show that CMVNMF performs much better than Constraint Prop using

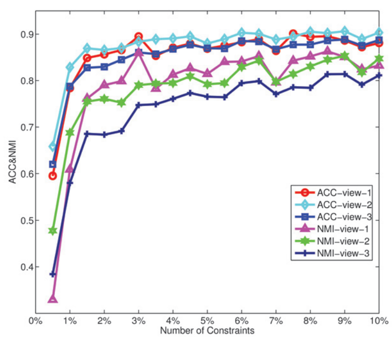
[5]http://mlg.ucd.ie/datasets/3sources.html



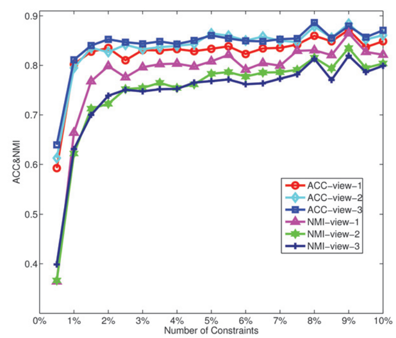
(a) Cornell



(b) Texas



(c) Washington



(d) Wisconsin

Figure 3: ACC and NMI on Webkb w.r.t Number of Constraints

Table 1: ACC and NMI on Partially Mapped Data

DATASET	ALGORITHM	ACC(%)			NMI(%)		
		VIEW 1	VIEW 2	VIEW 3	VIEW 1	VIEW 2	VIEW 3
THREESOURCES	CONSTRAINT PROP(5%)	$55.1 \pm .51$	52.9 ± 1.72	$34.5 \pm .15$	$46.9 \pm .61$	41.0 ± 1.19	34.0 ± 2.84
	CMVNMF(5%)	66.7 ± 7.06	71.4 ± 6.53	74.9 ± 5.68	74.6 ± 6.58	77.3 ± 6.89	79.2 ± 5.50
WIKIPEDIA	CONSTRAINT PROP(5%)	$17.6 \pm .13$	$77.0 \pm .71$	–	$4.46 \pm .02$	$85.8 \pm .21$	–
	CMVNMF(5%)	86.7 ± 2.49	86.7 ± 2.66	–	94.6 ± 1.31	94.5 ± 1.28	–

the same number of (inter-view v.s. intra-view) constraints. This is because that the inter-view constraints of CMVNMF provide enough mutual information for multi-view clustering, and CMVNMF uses NMF as the building block, which has been proved to be very effective for clustering.

Experiments on Completely Mapped Data

Datasets We use two datasets, Cora[2] and UCI Handwritten Digits[1], to demonstrate the effectiveness of CMVNMF on completely mapped data. Cora is comprised of 2708 documents over 7 labels. Every document is represented by 4 views (content, inbound, outbound, cites). Because cites is the sum of inbound and outbound, we only use the content, inbound and outbound views.

Settings We compare CMVNMF with NMF, one fast NMF algorithm XRAY (Kumar, Sindhwani, and Kambadur 2013), two NMF based multi-view clustering algorithms, multiNMF (Liu et al. 2013) and ConcatNMF (Kumar, Rai, and III 2011), and one spectral multi-view clustering algorithm, CoReguar (Kumar, Rai, and III 2011). The baseline algorithms are set with their original settings. For CMVNMF, we use 5% inter-view constraints. Since the data is completely mapped, we also test CMVNMF with 0 constraints.

Results Table 2 shows the clustering results on Cora and Handwritten. From the results it can be seen that CMVNMF (0%) show no superiority over the baseline algorithms, but CMVNMF (5%) performs much better than the other algorithms. This is because that the inter-view constraints not only boost the mutual information between views upon mapping, but also guide the clustering processing to get better results, i.e, inter-view must-links try to minimize the distance between objects from different views in the same cluster, and

Table 2: ACC and NMI on Completely Mapped Data

DataSet	Algorithm	Acc(%)			NMI(%)		
		view 1	view 2	view 3	view 1	view 2	view 3
Cora	NMF	38.3 ± 3.16	29.1 ± 1.83	36.9 ± 2.56	16.3 ± 2.95	$11.0 \pm .90$	23.3 ± 1.78
	XRAY	$32.6 \pm .39$	$30.1 \pm .15$	$33.4 \pm .00$	16.0 ± 1.40	$4.30 \pm .52$	$19.6 \pm .00$
	ConcatNMF	$34.8 \pm .00$	$34.8 \pm .00$	$34.8 \pm .00$	$20.0 \pm .00$	$20.0 \pm .00$	$20.0 \pm .00$
	multiNMF	33.9 ± 2.16	34.8 ± 1.54	35.7 ± 3.21	13.2 ± 1.30	12.0 ± 1.03	21.3 ± 1.46
	CoReguar	40.6 ± 2.53	42.6 ± 2.60	42.0 ± 2.56	$22.8 \pm .60$	$23.3 \pm .45$	$23.1 \pm .57$
	CMVNMF(0%)	37.9 ± 1.16	33.6 ± 1.35	40.7 ± 3.87	$16.0 \pm .00$	$13.8 \pm .44$	23.9 ± 2.75
	CMVNMF(5%)	58.6 ± 12.2	65.8 ± 9.24	64.3 ± 8.94	64.1 ± 9.00	68.5 ± 7.66	65.4 ± 8.45
Handwritten	NMF	71.6 ± 3.18	71.8 ± 4.95	52.2 ± 1.41	66.1 ± 1.24	69.7 ± 1.95	$49.7 \pm .74$
	XRAY	$58.6 \pm .69$	$67.8 \pm .10$	$50.5 \pm .11$	$56.2 \pm .65$	$60.5 \pm .15$	$46.5 \pm .11$
	ConcatNMF	$75.8 \pm .87$	$75.8 \pm .87$	$75.8 \pm .87$	$68.9 \pm .43$	$68.9 \pm .43$	$68.9 \pm .43$
	multiNMF	82.4 ± 3.22	83.5 ± 3.51	62.5 ± 3.22	73.6 ± 2.49	74.6 ± 3.29	57.4 ± 2.19
	CoReguar	75.0 ± 5.19	76.0 ± 3.15	76.5 ± 3.92	68.9 ± 2.09	68.8 ± 1.41	69.7 ± 1.51
	CMVNMF(0%)	72.9 ± 3.37	76.4 ± 6.27	54.6 ± 3.64	69.7 ± 1.93	70.1 ± 3.10	52.0 ± 1.18
	CMVNMF(5%)	82.6 ± 8.71	82.1 ± 7.44	82.1 ± 8.73	93.7 ± 3.78	93.5 ± 3.44	93.5 ± 3.85

inter-view cannot-links try to maximize the distance between objects from different views in different clusters.

Related Work

Multi-view Clustering

Spectral algorithms (de Sa 2005) creates a bipartite graph based on the nodes' co-occurring in both views and find a cut that crosses fewest lines. (Zhou and Burges 2007) generalizes the normalized cut from a single view to multiple views via a random walk. (Kumar and III 2011) reconstructs the similarity matrix of one view by the eigenvectors of the Laplacian in other views. (Kumar, Rai, and III 2011) integrates multiple information by co-regularizing the clustering hypotheses.

NMF based algorithms (Akata, Thurau, and Bauckhage 2011) enforces a shared coefficient matrix among different views. (Greene and Cunningham 2009) assumes that clustering results have been obtained from each view, and employs NMF over the clustering results. (Liu et al. 2013) (Liu and Han 2013) minimizes the difference between coefficient matrix of each view and the consensus matrix.

Other algorithms (Bickel and Scheffer 2004) applies Co-EM for multi-view clustering. (Blaschko and Lampert 2008) (Chaudhuri et al. 2009) extracts a shared subspace among the views and conduct clustering in the shared subspace. (Tzortzis and Likas 2012) learns a unified kernel through a weighted combination of kernels of all the views. (Bruno and Marchand-maillet 2009) utilizes consensus analysis to integrate single-view clustering results.

Constrained Clustering

Constrained single-view algorithms Due to the fact that a small number of constraints are easy to obtain in many real applications, constraints have been widely used for clustering. Many constrained (sing-view) clustering methods have been proposed (Wang, Li, and Zhang 2008) (Chen et al. 2008) (Li, Ding, and Jordan 2007) (Kulis et al. 2009) (Wang and Davidson 2010). They use constraints to adjust clustering objective functions or learn new distances, showing that constraints are much helpful to improve clustering result.

Constrained multi-view algorithms (Eaton, desJardins, and Jacob 2012) proposes a constrained multi-view clustering algorithm that deal with incompletely mapped data with the help of intra-view constraints. This algorithm propagates constraints in each view to those instances that can be mapped to the other views. Iteratively, constraints in each view are transferred across views via the partial mapping. (Lu and Peng 2013) propagates both intra-view pairwise constraints and inter-view pairwise constraints to accomplish the task of cross-view retrieval. Our proposed algorithm uses inter-view constraints (intra-view constraints could also be used, but they are not as effective as inter-view constraints). Different from all existing multi-view clustering algorithms, our algorithm can work on unmapped data.

Conclusion

In this paper, in the context of constrained clustering, we have addressed the problem of multi-view clustering for unmapped data, on which existing algorithms do not work. The proposed algorithm CMVNMF uses NMF as the basic clustering framework, and uses a small number of inter-view constraints instead of mapping to get mutual information between views. Experimental results on several benchmark datasets for multi-view clustering show that, with a small number of constraints, CMVNMF obtains high clustering accuracy on unmapped data, and outperforms existing algorithms on partially mapped data and completely mapped data. For future work we will investigate how to improve the effectiveness of both inter-view and intra-view constraints with constraint propagation, and study robustness of the algorithm to some mis-specification of constraints.

Acknowledgments

This work was supported by National Science Foundation of China (No. 61272374, 61300190).

References

Akata, Z.; Thurau, C.; and Bauckhage, C. 2011. Non-negative matrix factorization in multimodality data for segmentation and label prediction. In *Computer Vision Winter Workshop*, 1–8.

Bickel, S., and Scheffer, T. 2004. Multi-View Clustering. In *IEEE International Conference on Data Mining*, 19–26.

Blaschko, M. B., and Lampert, C. H. 2008. Correlational spectral clustering. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, 1–8. IEEE.

Bruno, E., and Marchand-maillet, S. 2009. Multiview clustering: a late fusion approach using latent models. In *Research and Development in Information Retrieval*, 736–737.

Cai, D.; He, X.; Wu, X.; and Han, J. 2008. Nonnegative Matrix Factorization on Manifold. In *IEEE International Conference on Data Mining*, 63–72.

Chaudhuri, K.; Kakade, S. M.; Livescu, K.; and Sridharan, K. 2009. Multi-view clustering via canonical correlation analysis. In *International Conference on Machine Learning*, 129–136.

Chen, Y.; Rege, M.; Dong, M.; and Hua, J. 2008. Non-negative matrix factorization for semi-supervised data clustering. *Knowledge and Information Systems* 17(3):355–379.

de Sa, V. R. 2005. Spectral clustering with two views. In *ICML workshop on learning with multiple views*.

Ding, C. H. Q., and He, X. 2005. On the Equivalence of Nonnegative Matrix Factorization and Spectral Clustering. In *SIAM International Conference on Data Mining*.

Eaton, E.; desJardins, M.; and Jacob, S. 2012. Multi-view constrained clustering with an incomplete mapping between views. *Knowledge and information systems* 38(1):231–257.

Gillis, N., and Vavasis, S. A. 2013. Fast and robust recursive algorithms for separable nonnegative matrix factorization. *IEEE Transactions on Pattern Analysis & Machine Intelligence*.

Greene, D., and Cunningham, P. 2009. A matrix factorization approach for integrating multiple data views. In *Machine Learning and Knowledge Discovery in Databases*. Springer. 423–438.

Kuang, D.; Park, H.; and Ding, C. H. 2012. Symmetric nonnegative matrix factorization for graph clustering. In *SDM*, volume 12, 106–117. SIAM.

Kulis, B.; Basu, S.; Dhillon, I.; and Mooney, R. 2009. Semi-supervised graph clustering: a kernel approach. *Machine learning* 74(1):1–22.

Kumar, A., and III, H. D. 2011. A Co-training Approach for Multi-view Spectral Clustering. In *International Conference on Machine Learning*.

Kumar, A.; Rai, P.; and III, H. D. 2011. Co-regularized multi-view spectral clustering. In *Advances in Neural Information Processing Systems*.

Kumar, A.; Sindhwani, V.; and Kambadur, P. 2013. Fast conical hull algorithms for near-separable non-negative matrix factorization. In *Proceedings of The 30th International Conference on Machine Learning*, 231–239.

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

Li, T., and Ding, C. H. Q. 2006. The Relationships Among Various Nonnegative Matrix Factorization Methods for Clustering. In *IEEE International Conference on Data Mining*, 362–371.

Li, T.; Ding, C.; and Jordan, M. I. 2007. Solving consensus and semi-supervised clustering problems using nonnegative matrix factorization. In *Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on*, 577–582. IEEE.

Liu, J., and Han, J. 2013. HINMF: A Matrix Factorization Method for Clustering in Heterogeneous Information Networks. In *Proc. of 2013 Int. Joint Conf. on Artificial Intelligence Workshop*.

Liu, J.; Wang, C.; Gao, J.; and Han, J. 2013. Multi-view clustering via joint nonnegative matrix factorization. In *Proc. of SDM*, volume 13, 252–260. SIAM.

Lu, Z., and Peng, Y. 2013. Unified Constraint Propagation on Multi-View Data. In *AAAI*.

Rasiwasia, N.; Pereira, J. C.; Coviello, E.; Doyle, G.; Lanckriet, G. R. G.; Levy, R.; and Vasconcelos, N. 2010. A new approach to cross-modal multimedia retrieval. In *ACM Multimedia Conference*, 251–260.

Recht, B.; Re, C.; Tropp, J.; and Bittorf, V. 2012. Factoring nonnegative matrices with linear programs. In *Advances in Neural Information Processing Systems*, 1214–1222.

Tzortzis, G., and Likas, A. 2012. Kernel-based Weighted Multi-view Clustering. In *International Conference on Data Mining*.

Wang, X., and Davidson, I. 2010. Flexible constrained spectral clustering. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, 563–572. ACM.

Wang, F.; Li, T.; and Zhang, C. 2008. Semi-Supervised Clustering via Matrix Factorization. In *SIAM International Conference on Data Mining*, 1–12.

Xu, W.; Liu, X.; and Gong, Y. 2003. Document clustering based on non-negative matrix factorization. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, 267–273. ACM.

Zhou, D., and Burges, C. J. C. 2007. Spectral clustering and transductive learning with multiple views. In *International Conference on Machine Learning*, 1159–1166.