Fast Clustering with Co-clustering via Discrete Non-negative Matrix Factorization for Image Identification – ICASSP 2020

Feiping Nie, Shenfei Pei, Rong Wang, Xuelong Li

Center for OPTical IMagery Analysis and Learning (OPTIMAL)
Northwestern Polytechnical University

shenfeipei@gmail.com

April 17, 2020

Fast Clustering with Co-clustering via Discrete Non-negative Matrix Factorization for Image Identification – ICASSP 2020

Feiping Nie, Shenfei Pei, Rong Wang, Xuelong Li

Center for OPTical IMagery Analysis and Learning (OPTIMAL)
Northwestern Polytechnical University

shenfeipei@gmail.com

April 17, 2020

Outline

Large-scale Datasets

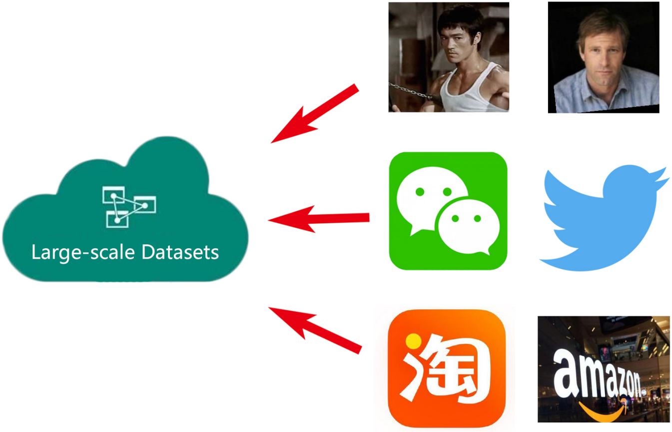


Figure 1: Large-scale dataset can be found everywhere in our lives

Large-scale Datasets

Why Clustering?

- Clustering has no requirement on data.
- Collecting unlabeled data is easy.

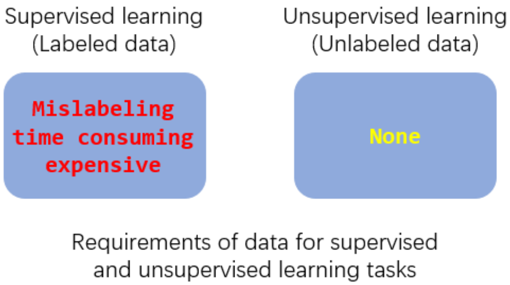


Figure 2: Supervised Learning and Unsupervised learning

Introduction to Clustering

Definition of clustering

Clustering is the task of grouping a set of objects in such a way that objects in the same group are more similar (in some sense) to each other than to those in other groups.

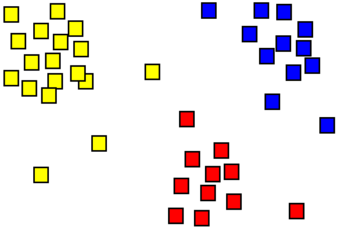


Figure 3: Schematic diagram of clustering

Introduction to Clustering

Related work

- The family of spectral clustering methods gains the most popularity.
- Despite its good performance, the time and space complexity of SC are $O(n^3)$ and $O(n^2)$, respectively.
- Much effort has been devoted for accelerating the spectral clustering algorithm, in recent years.

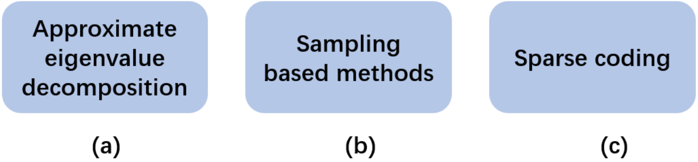


Figure 4: Related work

Motivation

Key observation

- Bipartite spectral graph partition (BSGP) is the most famous co-clustering algorithm because of its remarkable performance.
- The similarity between the sample and the anchor can be treated as another description of sample.
- Recent studies have shown that using anchor graph to construct similar matrix can still yield promising results.

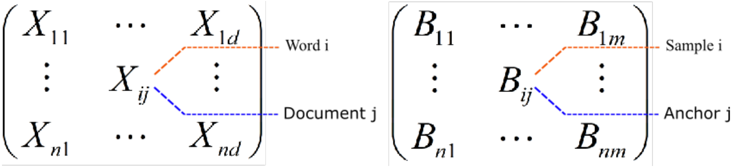


Figure 5: From BSGP to our model

Motivation

Our model

According to BSGP, it is not difficult to get our model (replace the data matrix X with the similarity matrix between samples and anchors B).

$$\min_{Y\in\Phi^{(n+m)\times c}} \sum_{k=1}^{c} \frac{y_k^T L y_k}{y_k^T D y_k}$$

where $L = D - W$, D is a diagonal matrix, $D_{ii} = \sum_{j=1}^{n+m} W_{ij}$,
$W = \begin{pmatrix} 0 & B \\ B^T & 0 \end{pmatrix}$

Our Model

Derivation

$$\min_{Y \in \Phi^{(n+m)\times c}} \sum_{k=1}^{c} \frac{y_k^T L y_k}{y_k^T D y_k} \tag{1}$$

$$\min_{Y \in \Phi^{(n+m)\times c}} Tr(Y^T L Y (Y^T D Y)^{-1}) \tag{2}$$

$$\max_{Y \in \Phi^{(n+m)\times c}} Tr(Y^T W Y (Y^T D Y)^{-1}) \tag{3}$$

Taking $Y^T = \left[\begin{array}{cc} P^T & Q^T \end{array} \right]$ into Eq. (5.3), we have

$$Y^T W Y = P^T B Q + Q^T B^T P \tag{4}$$

$$Y^T D Y = P^T D^{(1)} P + Q^T D^{(2)} Q \tag{5}$$

where $D^{(1)}$ and $D^{(2)}$ are both diagonal matrices, $D^{(1)}_{ii} = \sum_{j=1}^{m} B_{ij}$, $D^{(2)}_{jj} = \sum_{i=1}^{n} B_{ij}$

Our Model

With those notations in Eq. (4) and Eq. (5), the problem in Eq. (3) can be rewritten as follows:

$$\max_{P\in\Phi^{n\times c},Q\in\Phi^{m\times c}} Tr(P^TBQ(P^TD^{(1)}P+Q^TD^{(2)}Q)^{-1}) \tag{6}$$

Relaxation

We relax the problem in eq. (6) into the following form by adding two terms $Tr(T^{-1}P^TPT^{-1}Q^TQ)$ and $Tr(B^TB)$, where T represents $P^TD^{(1)}P+Q^TD^{(2)}Q$.

$$\min_{P,Q} \|B-P(P^TD^{(1)}P+Q^TD^{(2)}Q)^{-1}Q^T\|_F^2 \tag{7}$$

Our Model

Relaxation

$$\min_{P,Q} \|B - P(P^T D^{(1)} P + Q^T D^{(2)} Q)^{-1} Q^T\|_F^2$$

$$\mathbf{P^T D^{(1)} P} + \mathbf{Q^T D^{(2)} Q} \Rightarrow \text{A new variable } S$$

$$\Downarrow$$

$$\min_{P,Q,S} \|B - PSQ^T\|_F^2, \tag{8}$$

$$s.t. \quad P \in \Phi^{n\times c}, Q \in \Phi^{m\times c}, S \in R^{c\times c}$$

The optimization problem in Eq. (8) can be solved using standard techniques (alternating minimization).

Optimization

Update S

$$\min_{S \in \mathbb{R}^{c \times c}} \|B - PSQ^T\|_F^2. \tag{9}$$

Let $J(S)$ denote the objective function in Eq. (9). The derivative of $J(S)$ with respect to S is as follows:

$$\frac{\partial J(S)}{\partial S} = -2P^T BQ + 2P^T PSQ^T Q. \tag{10}$$

By setting the derivative of the objective function with respect to S to zero, we have

$$S_{ij} = \frac{(P^T BQ)_{ij}}{(P^T P)_{ii}(Q^T Q)_{jj}}. \tag{11}$$

Update P

$$\min_{P\in\Phi^{n\times c}} \|B - P(SQ^T)\|_F^2. \tag{12}$$

The solution can be determined by

$$P_{ij} = \begin{cases} 1 & j = \arg min_k \|B_i - (SQ^T)_k\|_2^2, \\ 0 & otherwise. \end{cases} \tag{13}$$

Update Q

$$\min_{Q\in\Phi^{m\times c}} \|B - (PS)Q^T\|_F^2, \tag{14}$$

The solution is determined by

$$Q_{ij} = \begin{cases} 1 & j = \arg min_k \|B^i - (PS)^k\|_2^2, \\ 0 & otherwise, \end{cases} \tag{15}$$

where $B^i((PS)^k)$ denote i-th (k-th) column of $B(PS)$.

Algorithm

Algorithm 1: Algorithm to solve the problem in Eq. (8)

Data: Date matrix $X \in \mathbb{R}^{n \times d}$, the number of anchors and nearest neighbors;

Result: Indicator matrices P and Q

Construct B according to [21] and initialize P and Q in a random way;

while *not converge* **do**

Compute S by Eq. (11) ;

Compute P by Eq. (13) ;

Compute Q by Eq. (15) ;

end

Contributions

Pros

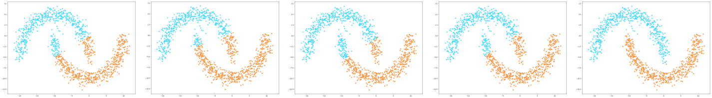
- The limitation of collaborative clustering can only be applied to specific scene can be broken by introducing anchor-based strategy.
- The final clustering result can be obtained directly without any post-processing
- The time and space complexity of FCDMF are both linear with respect to the number of samples.

Experiments

Comparison methods

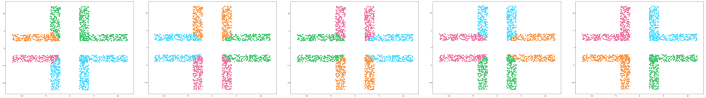
- Traditional Spectral Clustering (SC) (NIPS-2002)
- Scalable Spectral Clustering with cosine similarity (SSC) (ICPR-2018)
- Improved Anchor-based Graph Clustering based on multiplicative update optimization (AGC-I) (RS-2019)
- Fast Spectral Clustering with anchor graph for large hyperspectral images (FSC) (GRSL-2017)
- Large scale Spectral Clustering via landmark-based sparse representation (LSC) (TC-2015)
- Fast Clustering with co-clustering via Discrete non-negative Matrix Factorization (FCDMF) (Our method)

Experiments



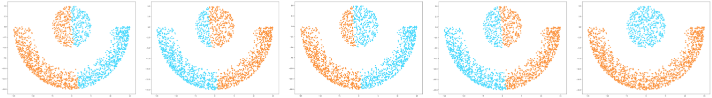
(a) SSC (b) AGC-I (c) FSC (d) LSC (e) FCDMF

Figure 1: Results on the Two-moon synthetic data.



(a) SSC (b) AGC-I (c) FSC (d) LSC (e) FCDMF

Figure 2: Results on the Four-corner synthetic data.



(a) SSC (b) AGC-I (c) FSC (d) LSC (e) FCDMF

Figure 3: Results on the Crescent-fullmoon synthetic data.

Experiments

Table 2: The average accuracy (± standard deviation) of several fast spectral clustering methods (The best result on each data set is highlighted in bold).

	SC	SSC	AGCI	FSC	LSC	FCDMF
BinAlpha	0.449(±0.015)	0.288(±0.013)	0.416(±0.012)	0.405(±0.013)	0.407(±0.015)	**0.421**(±0.012)
FACE-94	0.904(±0.012)	0.719(±0.014)	0.746(±0.017)	0.749(±0.016)	0.778(±0.017)	**0.914**(±0.011)
FACE-95	0.532(±0.012)	0.235(±0.010)	0.396(±0.013)	0.387(±0.012)	0.305(±0.013)	**0.487**(±0.010)
FEI	0.509(±0.020)	0.050(±0.007)	0.379(±0.015)	0.406(±0.017)	0.388(±0.017)	**0.499**(±0.014)
FERET	0.303(±0.007)	0.163(±0.008)	0.214(±0.004)	0.196(±0.009)	0.187(±0.008)	**0.252**(±0.006)
FingerPrint	0.569(±0.028)	0.172(±0.009)	0.334(±0.022)	0.306(±0.012)	0.383(±0.022)	**0.525**(±0.023)
Grimace	0.929(±0.012)	0.130(±0.010)	0.905(±0.026)	0.923(±0.022)	0.820(±0.029)	**0.967**(±0.002)
GTDB	0.548(±0.015)	0.199(±0.011)	0.345(±0.015)	0.336(±0.015)	0.327(±0.015)	**0.472**(±0.015)
IMM	0.593(±0.020)	0.109(±0.009)	0.319(±0.015)	0.451(±0.027)	0.407(±0.023)	**0.547**(±0.018)
JAFFE	0.825(±0.023)	0.640(±0.034)	**0.888**(±0.023)	0.832(±0.012)	0.705(±0.043)	0.845(±0.023)
JAFFE2	0.178(±0.000)	0.174(±0.002)	0.174(±0.004)	0.175(±0.008)	0.187(±0.003)	**0.197**(±0.000)
MPEG-7	0.565(±0.011)	**0.477**(±0.017)	0.194(±0.010)	0.181(±0.007)	0.181(±0.008)	0.429(±0.011)
ORL	0.588(±0.023)	0.514(±0.022)	0.427(±0.017)	0.497(±0.021)	0.463(±0.023)	**0.538**(±0.017)
PALM	0.786(±0.016)	**0.861**(±0.023)	0.600(±0.014)	0.704(±0.017)	0.635(±0.020)	0.747(±0.012)
Pixraw10P	0.910(±0.045)	0.141(±0.003)	0.729(±0.058)	0.632(±0.019)	0.659(±0.058)	**0.930**(±0.000)
UMIST	0.409(±0.014)	0.409(±0.021)	0.390(±0.011)	0.387(±0.011)	0.370(±0.011)	**0.449**(±0.011)
YALE	0.511(±0.023)	0.332(±0.017)	0.393(±0.023)	0.389(±0.022)	0.397(±0.024)	**0.489**(±0.021)

Experiments

Table 4: The average normalized mutual information (± standard deviation) of several spectral clustering methods (The best result on each data set is highlighted in bold).

	SC	SSC	AGCI	FSC	LSC	FCDMF
BinAlpha	0.592(±0.007)	0.443(±0.011)	0.578(±0.006)	**0.58**(±0.006)	0.562(±0.007)	0.570(±0.006)
FACE-94	0.969(±0.004)	0.881(±0.005)	0.938(±0.004)	0.939(±0.005)	0.937(±0.005)	**0.973**(±0.003)
FACE-95	0.743(±0.005)	0.492(±0.010)	0.676(±0.007)	0.679(±0.009)	0.618(±0.010)	**0.706**(±0.005)
FEI	0.716(±0.009)	0.103(±0.020)	0.656(±0.007)	0.677(±0.010)	0.663(±0.010)	**0.696**(±0.007)
FERET	0.693(±0.003)	0.501(±0.021)	0.641(±0.003)	0.585(±0.012)	0.559(±0.013)	**0.669**(±0.003)
FingerPrint	0.697(±0.017)	0.347(±0.016)	0.577(±0.022)	0.569(±0.013)	0.595(±0.021)	**0.669**(±0.014)
Grimace	0.954(±0.004)	0.214(±0.014)	0.964(±0.008)	**0.974**(±0.007)	0.921(±0.009)	0.972(±0.002)
GTDB	0.712(±0.008)	0.416(±0.014)	0.626(±0.008)	0.605(±0.011)	0.59(±0.011)	**0.664**(±0.007)
IMM	0.767(±0.010)	0.253(±0.028)	0.608(±0.01)	0.728(±0.018)	0.691(±0.016)	**0.744**(±0.009)
JAFFE	0.861(±0.010)	0.650(±0.027)	**0.877**(±0.005)	0.868(±0.009)	0.799(±0.016)	0.822(±0.018)
JAFFE2	0.013(±0.000)	0.052(±0.003)	0.015(±0.002)	0.013(±0.005)	0.035(±0.003)	**0.087**(±0.001)
MPEG-7	0.738(±0.004)	**0.672**(±0.009)	0.485(±0.018)	0.439(±0.019)	0.431(±0.019)	0.652(±0.004)
ORL	0.770(±0.011)	0.708(±0.012)	0.680(±0.008)	0.735(±0.012)	0.693(±0.014)	**0.747**(±0.008)
PALM	0.924(±0.006)	**0.958**(±0.006)	0.844(±0.004)	0.903(±0.005)	0.871(±0.007)	0.891(±0.004)
Pixraw10P	0.928(±0.017)	0.159(±0.005)	0.862(±0.026)	0.816(±0.008)	0.808(±0.026)	**0.935**(±0.000)
UMIST	0.649(±0.008)	0.591(±0.013)	0.638(±0.011)	**0.645**(±0.012)	0.623(±0.011)	0.642(±0.011)
YALE	0.572(±0.016)	0.406(±0.015)	0.494(±0.020)	0.486(±0.020)	0.481(±0.019)	**0.533**(±0.013)

Conclusions

Conclusions

- Our model relaxed the original objective function to a non-negative matrix factorization problem.
- In our model, the final clustering result can be obtained directly without any post-processing.
- An efficient optimization algorithm whose time and space complexity are both linear with respect to the number of samples
- Substantial performance

Thanks for Listening!

Questions?