A probabilistic framework for multi-view feature learning with many-to-many associations via neural networks

Akifumi Okuno[1,2], Tetsuya Hada[3], and Hidetoshi Shimodaira[1,2]

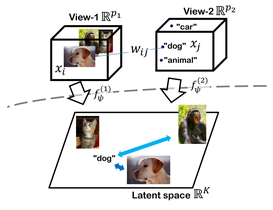
[1]Graduates school of Informatics, Kyoto University, Kyoto, Japan, [2]RIKEN Center for Artificial Intelligence Project (AIP), Tokyo, Japan, [3]Recruit Technologies Co., Ltd., Tokyo, Japan.

Introduction to Multi-view feature learning

Different types of data, such as user information, posted images, and their tags, etc. in Social Networking Service (SNS) is called **multi-view data**.

The number of view is $D \in \mathbb{N}$, and i-th data v_i belongs to view-$d_i (\in \{1, 2, \ldots, D\})$. i-th data v_i has p_{d_i}-dimensional vector representation $\boldsymbol{x}_i \in \mathbb{R}^{p_{d_i}}$, which we call **data vector**. The strength of association between $\boldsymbol{x}_i, \boldsymbol{x}_j$ is dentoed as $w_{ij} = w_{ji} \geq 0$, and is called **matching weight**. $\{\boldsymbol{x}_i\}$ and $\{w_{ij}\}$ are observed variables in our setting.

These multi-view data is hard to analyze, because their dimension may differ depending on the view. To get over the problem, data vector is transformed into **feature vector** $\boldsymbol{y}_i := f_{\boldsymbol{\psi}}^{(d_i)}(\boldsymbol{x}_i) \in \mathbb{R}^K$ by continuous maps $f_{\boldsymbol{\psi}}^{(d)} : \mathbb{R}^{p_d} \to \mathbb{R}^K$, $(d = 1, 2, \ldots, D)$ so that the inner product similarity of $\{\boldsymbol{y}_i\}$ in $\mathbb{R}^K$ represents matching weights $\{w_{ij}\}$. The dimension K is the same among different views.



We call the procedure in general to transform multi-view data vectors $\{\boldsymbol{x}_i\}$ into feature vectors $\{\boldsymbol{y}_i\}$ as **multi-view feature learning (MvFL)**.

However, every existing methods for MvFL has at least one of following **drawbacks**: (i)Limited to **linear** setting, (ii) Limited to **one-to-one** association, (iii)Requires **high computational cost**.

Proposed probabilistic model

To realize MVFL simultaneously solving above mentioned problems (i)–(iii), we propose a novel probabilsitic model of link weights:

$$w_{ij} \mid \boldsymbol{x}_i, \boldsymbol{x}_j \overset{\text{indep.}}{\sim} \mathrm{Po}(\exp(\mu(\boldsymbol{x}_i, \boldsymbol{x}_j; \boldsymbol{\theta}))), \qquad (1)$$

where $\mu(\boldsymbol{x}_i, \boldsymbol{x}_j; \boldsymbol{\theta}) := \alpha^{(d_i, d_j)} \exp(\langle f_{\boldsymbol{\psi}}^{(d_i)}(\boldsymbol{x}_i), f_{\boldsymbol{\psi}}^{(d_j)}(\boldsymbol{x}_j) \rangle)$ and $\boldsymbol{\theta} := (\boldsymbol{\psi}, \{\alpha^{(d,e)}\})$ is to be estimated.

As the continuous map $f_{\boldsymbol{\psi}}^{(d)} : \mathbb{R}^{p_d} \to \mathbb{R}^K$, we use

- **Neural network (NN)**. Any deterministic NN can be used, but we basically consider multi layer perceptron (MLP).
- **Linear transformation (LT)**. By applying identity activation function, MLP reduces to LT; LT is a special case of NN.

Probabilistic Multi-view Graph Embedding

To find the optimal $\boldsymbol{\theta} = (\boldsymbol{\psi}, \{\alpha^{(d,e)}\})$, we consider to maximize the log-likelihood based on (1),

$$\ell(\boldsymbol{\theta}) := \sum_{1 \leq i < j \leq n} (w_{ij} \log \mu(\boldsymbol{x}_i, \boldsymbol{x}_j; \boldsymbol{\theta}) - \mu(\boldsymbol{x}_i, \boldsymbol{x}_j; \boldsymbol{\theta})),$$

by gradient-descent (GD). However, using a plain GD requires summing up $O(n^2)$ terms for computing the gradient. To reduce the high computational complexity, we alternatively utilize

$$\frac{\partial}{\partial \boldsymbol{\theta}} \left\{ \sum_{(i,j) \in \mathcal{W}_n'} w_{ij} \log \mu(\boldsymbol{x}_i, \boldsymbol{x}_j; \boldsymbol{\theta}) - \lambda \sum_{(i,j) \in \mathcal{I}_n'} \mu(\boldsymbol{x}_i, \boldsymbol{x}_j; \boldsymbol{\theta}) \right\}$$

as an approximation of the gradient, where $\lambda \geq 0$ is a tuning parameter and $\mathcal{I}_n', \mathcal{W}_n'$ contain elements re-sampled from $\mathcal{I}_n := \{(i,j) \mid 1 \leq i < j \leq n\}$, $\mathcal{W}_n := \{(i,j) \in \mathcal{I}_n \mid w_{ij} > 0\}$ uniformly so that they contain m_1, m_2 elements, respectively.

Using $\hat{\boldsymbol{\theta}} = (\hat{\boldsymbol{\psi}}, \{\hat{\alpha}^{(d,e)}\})$ obtained by minibatch SGD, we have optimal feature vector

$$\boldsymbol{y}_i := f_{\hat{\boldsymbol{\psi}}}^{(d_i)}(\boldsymbol{x}_i) \in \mathbb{R}^K. \qquad (2)$$

We call the above mentioned procedure to obtain feature vectors as **Probabilistic Multi-view Graph Embedding (PMvGE)**.

Comparison

(Nv): Number of views, (MM): Many-to-many, (NL): Non-linear, (Ind): Inductive, (Lik): Likelihood-based. PMvGE has all the properties. Nv = D represents that the method can deal with arbitrary number of views.

	(Nv)	(MM)	(NL)	(Ind)	(Lik)
CCA (Hotelling 1936)	2			✓	
Deep CCA (Andrew et al. 2013)	2		✓	✓	
MCCA (Kettenring 1971)	D			✓	
SGE (Belkin et al. 2001)	0	✓			
LINE (Tang et al. 2015)	0	✓			✓
LPP (He et al. 2004)	1	✓		✓	
CvGE (Huang et al. 2013)	2	✓		✓	
CDMCA (Shimodaira 2016)	D	✓		✓	
DeepWalk (Perozzi et al. 2014)	0	✓			✓
SBM (Holland et al. 1983)	1	✓		✓	✓
GCN (Kipf et al. 2017)	1	✓	✓		✓
GraphSAGE (Hamilton et al. 2017)	1	✓	✓	✓	✓
IDW (Dai et al. 2018)	1	✓	✓	✓	✓
PMvGE (Proposed)	D	✓	✓	✓	✓

Representation power of PMvGE with NN

Theorem 1

$f_*^{(d)} : [-M, M]^{p_d} \to [-M', M']^{K_*}$, $(d = 1, 2, \ldots, D)$ are continuous maps and $g_* : [-M', M']^{2K_*} \to \mathbb{R}$ is a positive-definite kernel for some $M, M' > 0$ and $K_* \in \mathbb{N}$. For arbitrary $\varepsilon > 0$, by specifying large T, K, there exist MLPs $f_{\boldsymbol{\psi}}^{(d)} : \mathbb{R}^{p_d} \to \mathbb{R}^K$ with T hidden untis and ReLU or sigmoid activation s.t.

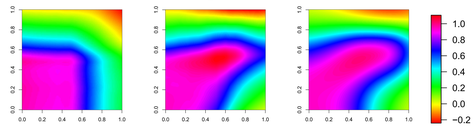
$$\left| g_* \left(f_*^{(d)}(\boldsymbol{x}), f_*^{(e)}(\boldsymbol{x}') \right) - \left\langle f_{\boldsymbol{\psi}}^{(d)}(\boldsymbol{x}), f_{\boldsymbol{\psi}}^{(e)}(\boldsymbol{x}') \right\rangle \right| < \varepsilon,$$

$\forall (\boldsymbol{x}, \boldsymbol{x}') \in [-M, M]^{p_d + p_e}$, $\forall d, e$.

We visualize Theorem 1. With cosine similarity g_*, we define

$$G_*(s,t) := g_*(f_*(s\boldsymbol{e}_1), f_*(t\boldsymbol{e}_2)), \hat{G}_K(s,t) := \langle f_{\boldsymbol{\psi}}(s\boldsymbol{e}_1), f_{\boldsymbol{\psi}}(t\boldsymbol{e}_2) \rangle,$$

where $f_*(\boldsymbol{x}) = (x_1, \cos x_2, \exp(-x_3), \sin(x_4 - x_5))$, $\boldsymbol{e}_1, \boldsymbol{e}_2 \in \mathbb{R}^5$ are directions, $f_{\boldsymbol{\psi}} : \mathbb{R}^5 \to \mathbb{R}^K$ is ReLU-based MLP with $T = 10^3$ hidden units and K output units.



(a) $\hat{G}_2(s,t)$ (b) $\hat{G}_{50}(s,t)$ (c) $G_*(s,t)$

Figures (a)–(c) show that $\hat{G}_K(s,t)$ with sufficiently large T, K approximates $G_*(s,t)$ well.

- Our new paper [Okuno and Shimodaira (2018). "On representation power of neural network-based graph embedding and beyond." arXiv:1805.12332] will be presented at TADGM workshop in ICML2018!!

PMvGE with $\mathrm{L}_{\text{inear}}$ $\mathrm{T}_{\text{ransformation}}$ approximately generalizes various existing methods

As the closest approach, Cross-Domain Matching Correlation Analysis (CDMCA; Shimodaira, 2016) obtains linear transformation (LT) $\boldsymbol{y}_i := \boldsymbol{\psi}^{(d_i)\top} \boldsymbol{x}_i$ by

$$\arg\max_{\{\boldsymbol{\psi}^{(d)}\}} \sum_{i=1}^n \sum_{j=1}^n w_{ij} \langle \boldsymbol{\psi}^{(d_i)\top} \boldsymbol{x}_i, \boldsymbol{\psi}^{(d_j)\top} \boldsymbol{x}_j \rangle,$$

with a quadratic constraint on $\boldsymbol{\psi}$, where $\boldsymbol{\psi}$ is a concatenation of $\{\boldsymbol{\psi}^{(d)}\}_{d=1}^D$. The obtained matrix is denoted as $\hat{\boldsymbol{\psi}}_{\text{CDMCA}}$.

Here, we approximate PMvGE with LT. $\tilde{\ell}_Q(\boldsymbol{\psi})$ denotes a quadratic approximation of the log-likelihood $\ell(\boldsymbol{\theta})$ around $\boldsymbol{\psi}$ with $\alpha^{(d,e)} = 1$, $\forall (d,e)$. The maximizer of $\tilde{\ell}_Q(\boldsymbol{\psi})$ is denoted as $\hat{\boldsymbol{\psi}}_{\text{Apr.PMvGE}}$.

Then, there exist $\gamma_1, \ldots, \gamma_K \geq 0$ such that

$$\hat{\boldsymbol{\psi}}_{\text{Apr.PMvGE}} = \hat{\boldsymbol{\psi}}_{\text{CDMCA}} \, \mathrm{diag}(\gamma_1^{1/2}, \ldots, \gamma_K^{1/2}).$$

Since CDMCA generalizes various existing methods such as HIMFAC (Nori et al., 2012), CvGE (Huang et al., 2013), CCA, PCA, thus PMvGE with LT approximately generalizes these methods as well.

Experiments

Dataset: We use Cora citation dataset (Sen et al., 2008, C-cite) for task 1 and 2, and Animal with Attribute dataset (Lampert et al., 2009, AwA) for task 3.

- **C-cite** is $(D =)$1-view dataset consisting of 2,708 nodes with 1,433-dim bag-of-words vectors. Each node has a class label of 7 classes.
- **AwA** is $(D =)$2-view dataset, which consists of resampled 2,500 images with 4,096-dim DeCAF features (Donahue et al., 2014), and 85 attributes with 300-dim GloVe (Pennington et al., 2014) features.

For each dataset, we preliminary compute feature vectors. Then several existing methods are applied for the following tasks:

- **Task-1:** Nodes in C-cite is classified into 7-classes using multi-class logistic regression. Results are evaluated by classification accuracy.
- **Task-2:** We made 7 clusters of nodes in C-cite by k-means clustering. Results are evaluated by Normalized Mutual Information (NMI).
- **Task-3:** For each query image, we rank attributes according to the cosine similarity of feature vectors across views. Results are evaluated by Average Precision (AP).

Results are listed in the form [Ave. $\pm$ std.dev] over 10 times experiments.

		(A)	(B)
Task 1 ($D = 1$)	ISOMAP	54.5 ± 1.78	54.8 ± 2.43
	LLE	30.2 ± 1.91	31.9 ± 2.62
	SGE	47.6 ± 1.64	-
	MDS	29.8 ± 2.25	-
	DeepWalk	54.2 ± 2.04	-
	GraphSAGE	60.8 ± 1.73	57.1 ± 1.61
	PMvGE	$\mathbf{74.8 \pm 2.55}$	$\mathbf{71.1 \pm 2.10}$
Task 2 ($D = 1$)	SBM	4.37 ± 1.44	2.81 ± 0.10
	ISOMAP	13.0 ± 0.36	14.3 ± 1.98
	LLE	7.40 ± 3.40	9.47 ± 3.00
	SGE	1.41 ± 0.34	-
	MDS	2.81 ± 0.10	-
	DeepWalk	16.7 ± 1.05	-
	GraphSAGE	19.6 ± 0.93	12.4 ± 3.00
	PMvGE	$\mathbf{35.9 \pm 0.88}$	$\mathbf{30.5 \pm 3.90}$
Task 3 ($D = 2$)	CCA	45.5 ± 0.20	42.4 ± 0.30
	Deep CCA	41.4 ± 0.30	41.2 ± 0.35
	SGE	43.5 ± 0.39	-
	DeepWalk	71.3 ± 0.57	-
	PMvGE	$\mathbf{71.5 \pm 0.48}$	$\mathbf{70.5 \pm 0.53}$

Conclusion

We proposed Probabilistic Multi-view Graph Embedding (PMvGE) that simultaneously realizes (i)Non-linear, (ii)many-to-many, (iii)computationally efficient multi-view feature learning. Experiments demonstrated that PMvGE outperformed existing methods.

References

Okuno, A. and Shimodaira, H. (2018). On representation power of neural network-based graph embedding and beyond. In *Proceedings of the ICML2018 workshop on Theoretical Foundations and Applications of Deep Generative Models (TADGM)*.

Shimodaira, H. (2016). Cross-validation of matching correlation analysis by resampling matching weights. *Neural Networks*, 75:126–140.

Andrew, G., Arora, R., Bilmes, J., and Livescu, K. (2013). Deep Canonical Correlation Analysis. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 1247–1255.