Sequential Pattern Learning via Kernel Alignment

Miao Cheng

Email: mcheng@mailbox.gxnu.edu.cn
College of Computer Science and Information Engineering
Guangxi Normal University, Guilin, China

June 6, 2019

Motivation

- The original consideration is to design an affinity learning approach to feature (pattern) alignment.
- The incremental extension is available for futher enhancements.

Motivation

- The original consideration is to design an affinity learning approach to feature (pattern) alignment. ($\surd$)
- The incremental extension is available for futher enhancements. ($\surd$)

Motivation

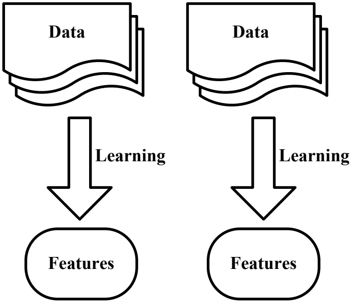
- The original consideration is to design an affinity learning approach to feature (pattern) alignment. $(\surd)$
- The incremental extension is available for futher enhancements. $(\surd)$
- The calculation efficiency can be preserved for sequential learning.

Motivation

- The original consideration is to design an affinity learning approach to feature (pattern) alignment. ($\surd$)
- The incremental extension is available for futher enhancements. ($\surd$)
- The calculation efficiency can be preserved for sequential learning. ($\times$, as a fresh attempt)

Pattern Alignment

- General Learning Approaches



Pattern Alignment

To make pattern alignment applicable

- Cross-domain learning
- Manifold matching

Pattern Alignment

To make pattern alignment applicable

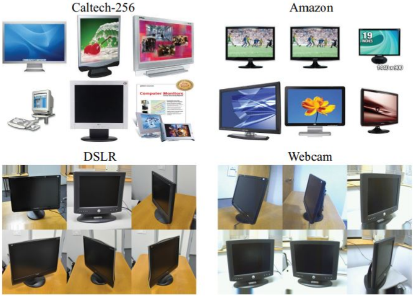
- Cross-domain learning
- Manifold matching

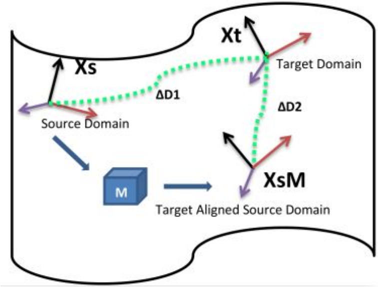
Suspicions, e.g.,

- Sequential Pattern Alignment

Domain Adaptation

The figures from B. Gong's GFK and B. Fernando's SA papers



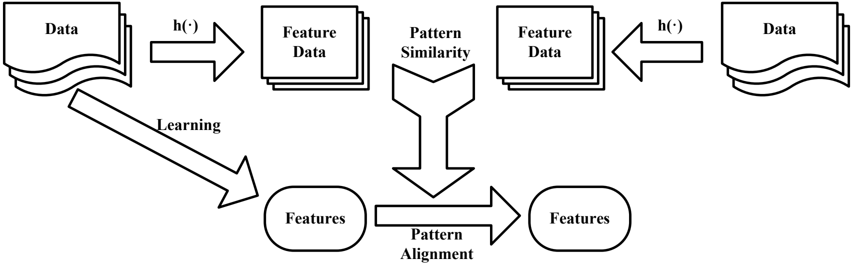


Continuous Manifold Based Adaptation

- CMA: A continuous extension of GFK
- SA (Subspace Alignment): Can be conveniently extended
- The further considerations are necessary for advancements

Pattern Alignment

Sequential alignment of incremental patterns with feature mapping



Pattern Alignment

- Sequential alignment of incremental patterns with feature mapping
- It is, but not restricted to reference patterns derived from feature kernel spaces

Basic Idea

- The source data is affected by target data sequentially.
- The subspace patterns of source and target data are aligned.
- Parallel mapping is involved to indirect alignments.

Feature alignment

$$
\begin{aligned}
x_s \sim h\left(x_s\right) \sim p\left(h\left(x_x\right)\right) \\
x_t \sim h\left(x_t\right) \sim q\left(h\left(x_t\right)\right)
\end{aligned}
\tag{1}
$$

$$
\Psi = \int_{x_s} \int_{x_t} \psi\left(h\left(x_s\right), h\left(x_t\right)\right) dh\left(x_s\right) dh\left(x_t\right), \tag{2}
$$

Pattern alignment

$$\varphi\left(x_s, x_t\right) = \int_{x_s}\int_{x_t} f\left(x_s\right)\cdot g\left(x_t\right) dx_s dx_t \qquad (3)$$

$$\Psi = \int_{x_s}\int_{x_t} \psi\left(h\left(x_s\right), h\left(x_t\right)\right) dh\left(x_s\right) dh\left(x_t\right) \qquad (4)$$

Pattern Alignment

Suppose that only source data are involved,

$$J = \arg\min_{\Gamma} \sum_{x_s} \left\| f\left(x_s\right) f\left(x_s\right) - h\left(x_s\right) \Gamma h\left(x_s\right) \right\|^2 \\ + \lambda\left(\Gamma h\left(x_s\right)\right) \tag{5}$$

$$J = \arg\min_{\Gamma} \sum_{x_s} \left\| f\left(x_s\right) \cdot f\left(x_s\right) - k_s M k_s \right\|^2 + \lambda tr\left(M k_s\right), \tag{6}$$

Pattern Alignment

Suppose that only source data are involved,

$$J = \arg\min_{\Gamma} \sum_{x_s} \left\| f(x_s) f(x_s) - h(x_s) \Gamma h(x_s) \right\|^2 + \lambda (\Gamma h(x_s)) \tag{7}$$

$$J = \arg\min_{\Gamma} \sum_{x_s} \left\| f(x_s) \cdot f(x_s) - k_s M k_s \right\|^2 + \lambda tr(M k_s), \tag{8}$$

$$J = \arg\min_{\Pi} \sum_{x_s} \left\| f(x_s) - \Pi k_s \right\|^2 + \lambda tr(\Pi k_s) \tag{9}$$

Pattern Alignment

$$J = \arg\min_{f} \sum_{x_s} \sum_{x_t} \left\| f(x_s)\, g(x_t) - k_{st} M k_s \right\|^2 + \lambda tr\left(M k_s\right) \tag{10}$$

Pattern Alignment

$$
\begin{aligned}
J &= \arg\min_{f} \sum_{x_s} \sum_{x_t} \left\| f\left(x_s\right) g\left(x_t\right) - k_{st} M k_s \right\|^2 \\
&+ \lambda tr\left(M k_s\right)
\end{aligned}
\tag{11}
$$

Tips: iterative learning is possible by fixing one item while solving another

Sequential Pattern Alignment

The last target data is denoted by ${X_t}' \in R^{d \times a}$, and the source data set X_s is modified with appended X_t'.

$$
\begin{aligned}
J \; &= \arg\min_{f} \sum_{x_s} \sum_{x_t} \left\| f\left(x_s\right) g\left(x_t\right) - k_{st} M k_s \right\|^2 \\
&+\lambda tr\left(M k_s\right)
\end{aligned}
\tag{12}
$$

Sequential Pattern Alignment

The last target data is denoted by ${X_t}' \in R^{d \times a}$, and the source data set X_s is modified with appended X_t'.

- $f(\cdot)$ is necessary to be modified corresponding to incremental data

Sequential Pattern Alignment

The last target data is denoted by $X_t{}' \in R^{d \times a}$, and the source data set X_s is modified with appended X_t'.

- $f(\cdot)$ is necessary to be modified corresponding to incremental data
- Feature alignment M also needs to be updated as well as kernels

Sequential Pattern Alignment

- To update $f(\cdot)$ with subspace alignment, it is straightforward to adopt SVD update algorithm to orthogonal decomposition of X_s

Sequential Pattern Alignment

- To update $f(\cdot)$ with subspace alignment, it is straightforward to adopt SVD update algorithm to orthogonal decomposition of X_s
- Feature alignment M is updated with appended kernel matrices and the main calculation is on inverse of kernels

Sequential Pattern Alignment

- To update $f(\cdot)$ with subspace alignment, it is straightforward to adopt SVD update algorithm to orthogonal decomposition of X_s
- Feature alignment M is updated with appended kernel matrices and the main calculation is on inverse of kernels
- The inverse of kernels can be approximated with SVD of kernel matrices $K = V\Lambda^2 V^T$

Sequential Pattern Alignment

- The decomposition of updated kernel matrix is learned by following the incremental KPCA (H. Zhao et. al., 2006)

$$K = \begin{bmatrix} K_{ss} & K_{st} \\ K_{ts} & K_{tt} \end{bmatrix} = \begin{bmatrix} P_s & P_1 \\ P_2 & P_3 \end{bmatrix}^T \begin{bmatrix} P_s & P_1 \\ P_2 & P_3 \end{bmatrix}. \quad (13)$$

Experiments

- Implemented in R language platform
- Ten common classes in three domains, namely Caltech, Amazon, DSLR, are refered, which contains 1123 (Caltech), 958 (Amazon), 157 (DSLR) images, respectively.
- SURF features of each data are extracted, and several state-of-the-art methods are involved to evaluate the performance.

Experiments

Table: Classification accuracy (%) in object recognition with SURF features

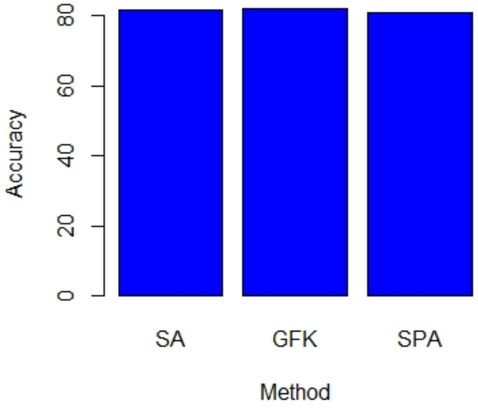
Method	A -> C	C -> A	A -> D	D -> A	C -> D	D -> C	Mean
Baseline	15.94	15.66	3.18	6.68	15.92	12.38	11.63
LSCCA	26.63	28.5	30.57	18.89	28.03	15.05	24.61
GFK	35.55	36.29	**31.97**	**32.5**	**36.97**	**29.58**	**33.81**
unsupervised KEMA	22.8	26.3	20.38	22.96	31.85	26.54	25.14
SA	**35.62**	**36.74**	25.48	24.22	25.48	27.25	29.13
TCA	20.57	25.68	12.74	24.43	18.28	24.58	21.05
SPA	32.41	33.3	26.11	25.68	26.11	25.56	28.2

Experiments

Sequential learning on benchmark traffic scene classification.

- 512 dimensional GIST features are learned from images.
- 500 labeled images are given, and then 2000 images come in a sequence with 100 images per duration.

Experiments



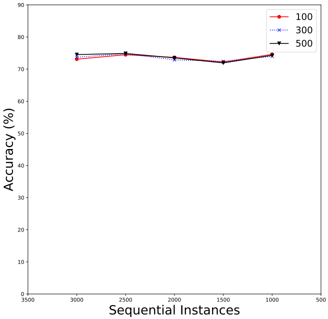
Experiments

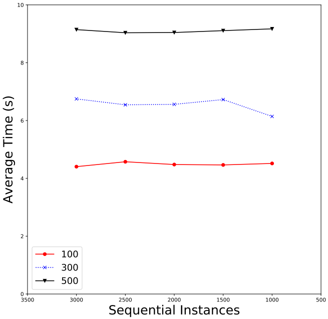
Sequential learning on flower data set

- 512-dimensional VGG features are learned representatively
- 1000 instances are randomly selected to be source data, while target instances ranging from 1000 to 3000

Experiments

Experimental results v.s. different sizes of sequential data





Conclusion and Future Works

- A novel approach to sequential pattern alignment is proposed in this work, which considers the featuer alignment as a reference for subspace alignment, while further more data is absorbed for sequential learning.
- The performance of proposed method is comparable, as a fresh attempt for further advancements.

Conclusion and Future Works

This work can be continued with several open problems:

- Improvements of learning efficiency, and iterative learning could be adopted.
- Other attempts to replace the feature alignment can be done.
- Further extensions to multi-view pattern alignment is possible.

Thank You