Boosted Cross-Domain Categorization

Fan Zhu[1]
fan.zhu@sheffield.ac.uk
Ling Shao[1]
ling.shao@ieee.org
Jun Tang[2]
tangjunahu@gmail.com

[1] Departmetn of Electronic and Electrical Engineering
The University of Sheffield
Sheffield, S1 3JD, UK

[2] School of Electronics and Information Engineering
Anhui University
Hefei, 230601, China

We introduce a boosted cross-domain categorization (BCDC) framework that utilizes labeled data from other domains as the source data to span the intra-class diversity of the original learning system. In addition to the manually annotated information in the target domain, partially labeled data from another visual domain are provided as the source domain. In comparison, the proposed learning framework shares the same basic principle of sequentially updating the impacts of training instances; yet our learning framework attempts to sequentially update the data representations of those "dis-similar" samples instead of simply weighting less on them. The learning function is formulated as:

$$\begin{aligned} &\langle D_t, D_s, X_t, \Phi, \mathcal{P} \rangle \\ &= arg \min_{D_t, D_s, X_t, \Phi, \mathcal{P}} \|Y_t - D_t X_t\|_2^2 \\ &+ \alpha \|Q - \Phi X_t\|_2^2 + \beta \|\mathcal{H} - \mathcal{P} X_t\|_2^2 \\ &+ \|Y_s \mathbb{A}^T - D_s X_t\|_2^2 \quad s.t. \forall i, \ \|x_t^i\|_0 \leq T. \end{aligned} \tag{1}$$

In order to distinguish the "dissimilar" data from the smooth data, we include the weighted discriminative sparse codes into the learning function. Specifically, $q_i = [q_i^1, q_i^2, \cdots, q_i^K]^T = [0, \cdots, w_i, w_i, \cdots, 0]^T \in \Re^K$, where the non-zeros occur at those indices where $y_t^i \in Y_t$ and $X_t^k \in X_t$ share the same class label. Given $X_t = [x_1, x_2, \cdots, x_6]$ and $Y_t = [y_1, y_2, \cdots, y_6]$, and assuming x_1, x_2, y_1 and y_2 are from class 1, x_3, x_4, y_3 and y_4 are from class 2, x_5, x_6, y_5 and y_6 are from class 3, Q is then defined with the following form:

$$\begin{pmatrix} w_1 & w_2 & 0 & 0 & 0 & 0 \\ w_1 & w_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & w_3 & w_4 & 0 & 0 \\ 0 & 0 & w_3 & w_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & w_5 & w_6 \\ 0 & 0 & 0 & 0 & w_5 & w_6 \end{pmatrix}, \tag{2}$$

Since predictions are made with respect to the data distribution of X_t, w_i is included in each item of $\mathcal{H}$. Thus $\mathcal{H}$ can be defined as follows according to the same example in Equation (2)

$$\begin{pmatrix} w_1 & w_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & w_3 & w_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & w_5 & w_6 \end{pmatrix}. \tag{3}$$

By defining $Y = (Y_t^T, (Y_s \mathbb{A}^T)^T, \sqrt{\alpha} Q^T, \sqrt{\beta} \mathcal{H}^T)^T$ and $D = D_t^T, D_s^T, \sqrt{(\alpha)}$ $\Phi^T, \sqrt{(\beta)} \mathcal{P}^T)^T$, where column-wise L_2 normalization is applied to D, the objective function in equation 1 can be solved through sequentially updating dictionary atoms and sparse codes as in [8].

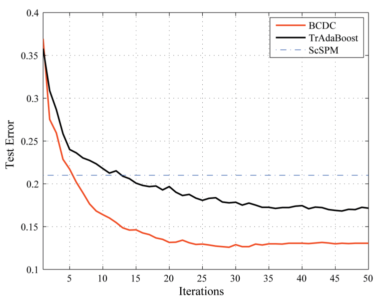

Figure 1: Error rate comparison of the proposed BCDC method with TrAdBoost and ScSPM on the Caltech-101 dataset.

Algorithm 1 Boosted cross-domain dictionary learning

Input the labeled target domain data $\mathcal{D}_l^t$ and the source domain data $\hat{\mathcal{D}}^s$, the maximum number of iterations $Max.iter$ and the Weak Learner.

Output a "strong" classifier $\mathcal{F}(\cdot)$ and updated representations of the source domain instances.

Initialize the data distribution as uniform, i.e., the initial weights $w^1 = (w_1^1, w_2^1, \cdots, w_{N+M}^1)$ have an identical value. Cross-domain discriminative dictionary learning is applied to both target domain and source domain data under the initialized uniform distribution, so that $\mathcal{D}_l^t$ and $\hat{\mathcal{D}}^s$ can be represented by X_t and X_s^1 respectively.

for $j = 1$ to $Max.iter$ **do**

1. Set data distribution $p^j = \frac{w^j}{\sum_{i=1}^{N+M} w_i^j}$
2. Update X_s^j as the new representation of $\hat{\mathcal{D}}^s$ under data distribution p^j with cross-domain discriminative dictionary learning.
3. Compute the hypothesis $h_t^j : X_t \to l(X_t)$ and $h_s^j : X_s^t \to l(X_s)$, providing that p^j is over both $\mathcal{D}_l^t$ and $\hat{\mathcal{D}}^s$.
4. Calculate the error ε^j of h_t^j:
$$\varepsilon^j = \sum_{i=1}^{N} \frac{w_i^j \times |h_t^j(x_i) - l(x_i)|}{\sum_i^N w_i^j},$$
where ε^j is required to be less than 0.5.
5. Set $\beta_t^j = \frac{\varepsilon^j}{1-\varepsilon^j}$ and $\beta_s = \frac{1}{1+\sqrt{2\ln M / Max.iter}}$
6. Update the new weight vector:
$$w_i^{j+1} = \begin{cases} w_i^j \beta_t^{j - |h_t^j(x_i) - l(x_i)|}, & 1 \leqslant i \leqslant N \\ w_i^j \beta_s^{|h_s^j(x_i) - l(x_i)|}, & otherwise. \end{cases}$$

end for

[1] Michal Aharon, Michael Elad, and Alfred Bruckstein. K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, 54(1):4311–4322, 2006.

[2] Y-Lan Boureau, Francis Bach, Yann LeCun, and Jean Ponce. Learning mid-level features for recognition. In *CVPR*, 2010.

[3] Julien Mairal, Francis Bach, Jean Ponce, Guillermo Sapiro, and Andrew Zisserman. Discriminative learned dictionaries for local image analysis. In *CVPR*, 2008.

[4] Julien Mairal, Marius Leordeanu, Francis Bach, Martial Hebert, and Jean Ponce. Discriminative sparse image models for class-specific edge detection and image interpretation. In *ECCV*, 2008.

[5] Julien Mairal, Francis Bach, Jean Ponce, Guillermo Sapiro, and Andrew Zisserman. Supervised dictionary learning. In *NIPS*, 2009.

[6] Jianchao Yang, Kai Yu, and Thomas Huang. Supervised translation-invariant sparse coding. In *CVPR*, 2010.

[7] Qiang Zhang and Baoxin Li. Discriminative k-svd for dictionary learning in face recognition. In *CVPR*, 2010.

[8] Fan Zhu and Ling Shao. Weakly-supervised cross-domain dictionary learning for visual recognition. *International Journal of Computer Vision*, 109(1-2):42–59, 2014.