Beyond the Deep Metric Learning: Enhance the Cross-Modal Matching with Adversarial Discriminative Domain Regularization

Li Ren, Kai Li, LiQiang Wang,Kien Hua

UNIVERSITY OF CENTRAL FLORIDA

facebook

ICPR 2020
25th INTERNATIONAL CONFERENCE ON PATTERN RECOGNITION
Milan, Italy 10 | 15 January 2021

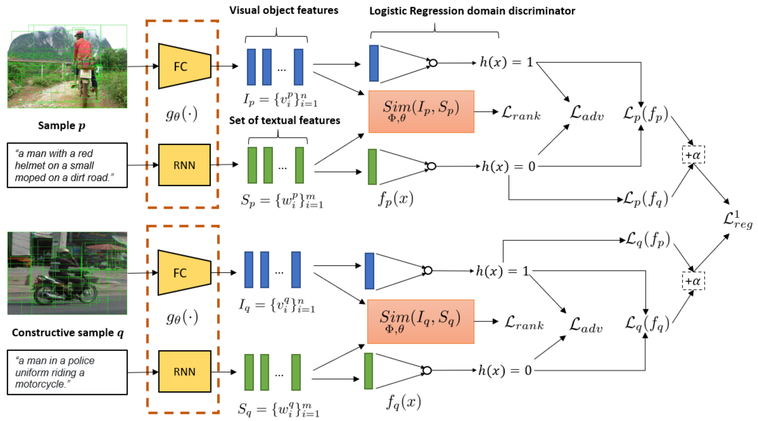
Introduction:

- Cross modal metric learning is a challenging task where we learn the similarity metric to the data from the different modalities. In this paper, we intend to compare the images and the textual sentence that describe the image.
- Our approach first adapts these pairs into two domains with two individual domain classifiers, where we compare the learning errors of these two classifiers in both matched sample pairs.
- We further introduce a regularization term to force each independent discriminators to become distinct to others in order to separate the feature spaces.

Contributions:

- We propose a novel framework Adversarial Discriminative Domain Regularization (ADDR) that generally enhances the cross-modal metric learning networks. It is achieved by learning a group of discriminative domains regularized with a constructive learning term that explicitly aligned to each image-text pair.
- Our ADDR is compatible with existing metric learning networks. It is used as an add-on regularizer to their primary tasks to help match between a group of visual objects and the corresponding sentence.
- Our quantitative experiments show the effectiveness of our approach base on the recent popular metric learning frameworks: the SCAN, VSRN, and BFAN

Discriminative Domain Regularization



Adversarial Training:

$$\min_{W_p, b_p} \mathcal{L}_{adv}(I_p, S_p) = \sum_{i=1}^{n} log(\sigma(W_p^T w_i^p + b_p)) + \sum_{i=1}^{m} log(1 - \sigma(W_p^T v_i^p + b_p)), \quad (2)$$

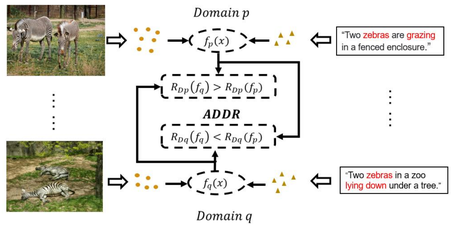
$$\min_{\Phi,\theta} \frac{1}{N} \sum_{p=1}^{N} [\mathcal{L}_{rank}(I_p, S_p) - \beta \mathcal{L}_{adv}(I_p, S_p)], \quad (3)$$

Discriminative Domain Regularization:

$$\mathcal{L}^1_{reg}(I_p, S_p, I_q, S_q) = \max[0, \alpha + \mathcal{L}_p(f_p) - \mathcal{L}_p(f_q),] + \max[0, \alpha + \mathcal{L}_q(f_q) - \mathcal{L}_q(f_p)] \quad (8)$$

$$\mathcal{L}^2_{reg}(I_p, S_p, I_r, S_r) = \max[0, \alpha + \mathcal{L}_q(f_q) - \mathcal{L}_q(f_p)] + \max[0, \alpha + \mathcal{L}_p(f_p) - \mathcal{L}_p(f_r)], \quad (9)$$

$$\mathcal{L}_{reg}(p, q, r) = \mathcal{L}^1_{reg}(I_p, S_p, I_q, S_q) + \mathcal{L}^2_{reg}(I_p, S_p, I_r, S_r)$$



Contact Author: renli@knights.ucf.edu

Experimental Results on MS-COCO and Flickr30k:

Method	Sentence Retrieval R@1	R@5	R@10	Image Retrieval R@1	R@5	R@10	Sum (ALL)
1k Test Set (5-fold)							
SCAN [13] (2018)	72.7	94.8	98.4	58.8	88.4	94.8	507.9
MTFN [25] (2019)	74.3	94.9	97.9	60.1	89.1	95.0	511.3
BFAN [15] (2019)	74.9	95.2	98.3	59.4	88.4	94.5	510.7
VSRN [14] (2019)	76.2	94.8	98.2	62.8	89.7	95.1	516.8
DPRNN [3] (2020)	75.3	95.8	98.6	62.5	89.7	95.1	517.0
ADAPT [27] (2020)	76.5	95.6	98.9	62.2	90.5	96.0	519.7
ADDR-SCAN (Ours)	76.1	95.5	98.4	61.2	88.9	94.8	514.9
ADDR-BFAN (Ours)	76.4	95.8	98.3	62.3	89.4	96.2	518.4
ADDR-VSRN (Ours)	**77.4**	**96.1**	**98.9**	**63.5**	**90.7**	**96.7**	**523.3**
5K Test Set							
SCAN [13] (2018)	50.4	82.2	90.0	38.6	69.3	80.4	410.9
MTFN [25] (2019)	48.3	77.6	87.3	35.9	66.1	76.1	391.3
BFAN [15] (2019)	52.9	82.8	90.6	38.3	67.8	79.3	411.7
VSRN [14] (2019)	53.0	81.1	89.4	40.5	70.6	81.1	415.7
ADDR-SCAN (Ours)	**57.3**	**86.0**	**92.7**	41.8	**72.0**	81.3	**431.1**
ADDR-BFAN (Ours)	54.3	84.0	91.5	40.1	69.2	80.6	419.7
ADDR-VSRN (Ours)	56.6	85.3	90.4	**42.5**	71.9	**82.0**	428.7

Method	Sentence Retrieval R@1	R@5	R@10	Image Retrieval R@1	R@5	R@10	Sum (ALL)
SCAN [13] (2018)	67.4	90.3	95.8	48.6	77.7	85.2	465.0
MTFN [25] (2019)	65.3	88.3	93.3	52.0	80.1	86.1	465.1
BFAN [15] (2019)	68.1	91.4	95.9	50.8	78.4	85.8	470.4
VSRN [14] (2019)	71.3	90.6	96.0	54.7	81.8	88.2	482.6
RDAN [7](2019)	68.1	91.0	95.9	54.1	80.9	87.2	477.2
ADDR-SCAN (Ours)	72.1	**93.1**	96.1	53.5	80.4	87.4	482.6
ADDR-BFAN (Ours)	71.3	91.5	96.4	54.0	80.0	87.6	480.8
ADDR-VSRN (Ours)	**73.0**	92.5	**96.6**	**55.6**	**82.0**	**88.9**	**488.6**

Examples: