Text Recognition in Real Scenarios with a Few Labeled Samples

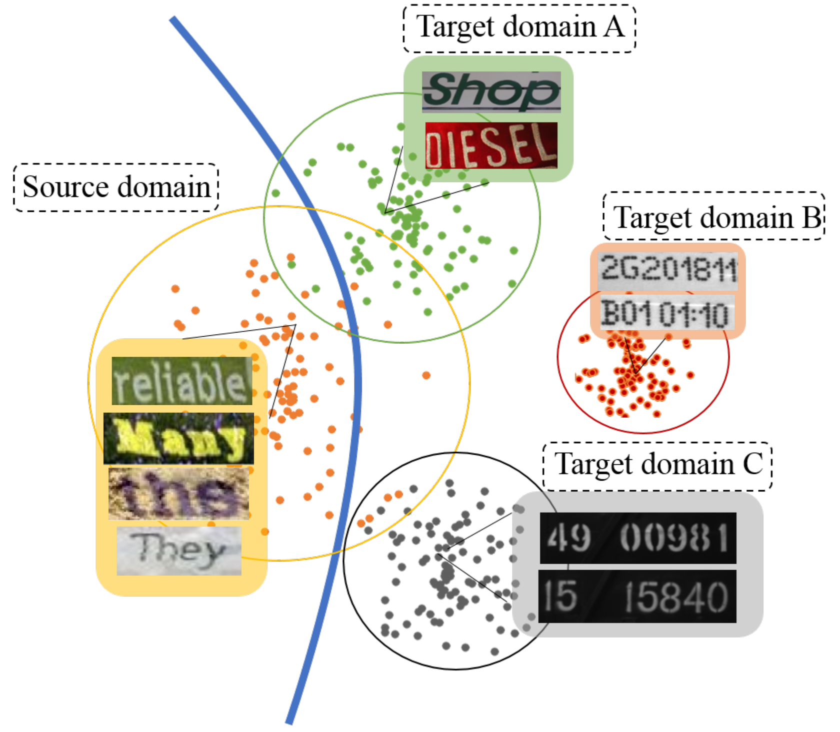
Jinghuang Lin[1] Zhanzhan Cheng[2] Yi Niu[2] Shiliang Pu[2] Shuigeng Zhou[1]

[1] School of Computer Science, Fudan University

[2] Hikvision Research Institute, China

Motivation

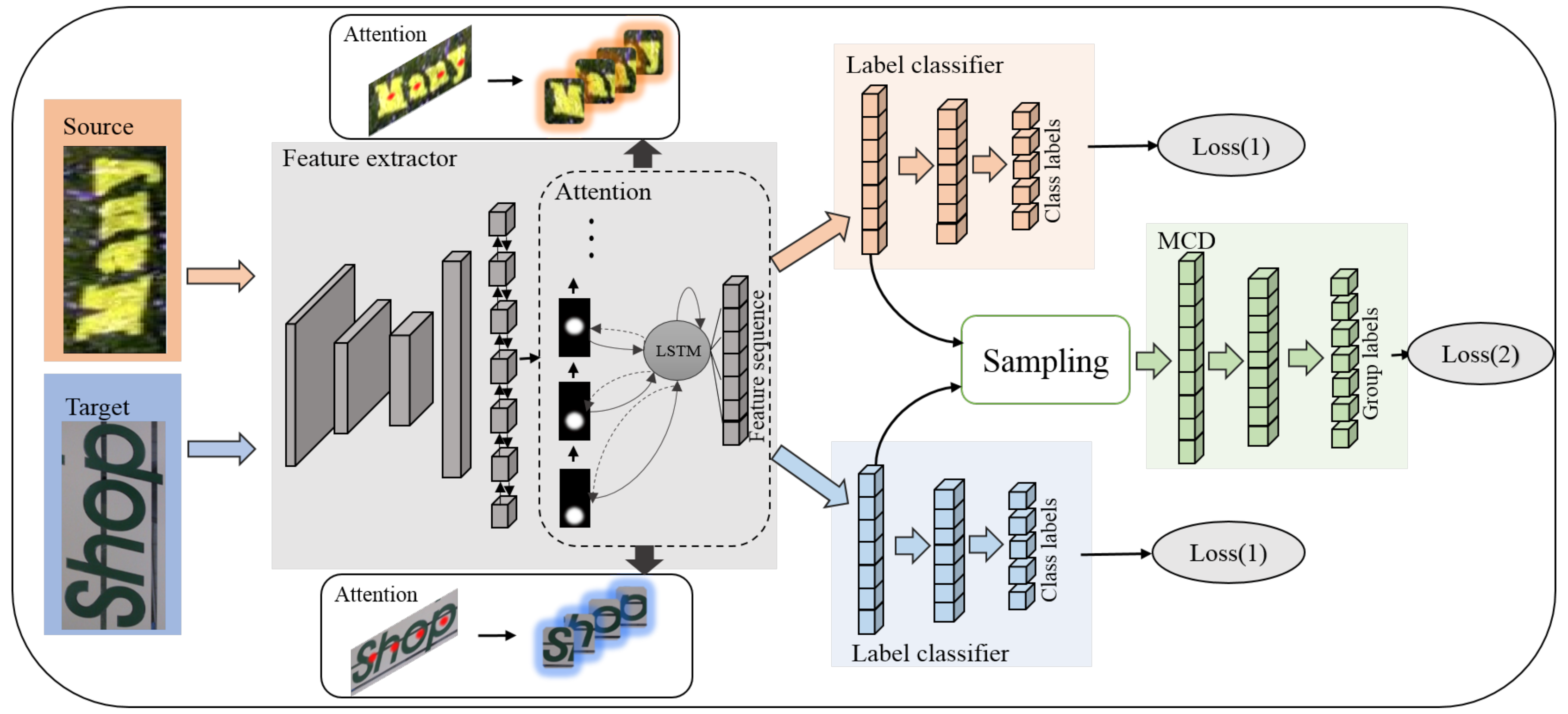
- Few-shot domain adaptation techniques have shown their effectiveness in handling scenarios where labeled samples are lacked.
- Most few-shot domain adaptation techniques are focus on character-level task and can hardly handle STR problem because it is a sequence-level image classification task.



- We present a few-shot adversarial sequence domain adaptation approach to achieve sequence-level domain confusion by integrating a well-designed attention mechanism with sequence-level adversarial learning strategy into a framework.

Few-shot Adversarial Sequence Domain Adaptation (FASDA)

- The architecture of FASDA consists of two procedures :
  1) Weakly-supervised Character Feature Representation
     - provide "weak" character-level feature representation
  2) Few-shot Adversarial Learning
     - maximize the character-level confusion between the source domain and the target domain



- Attention mechanism with Inclusive attending process

Define $\alpha_{t,j}$ as the attending weight of the t-th character

$$\alpha_{t,j} = \frac{exp(e_{t,j})}{\sum_{i=1}^{M} exp(e_{t,i})}$$

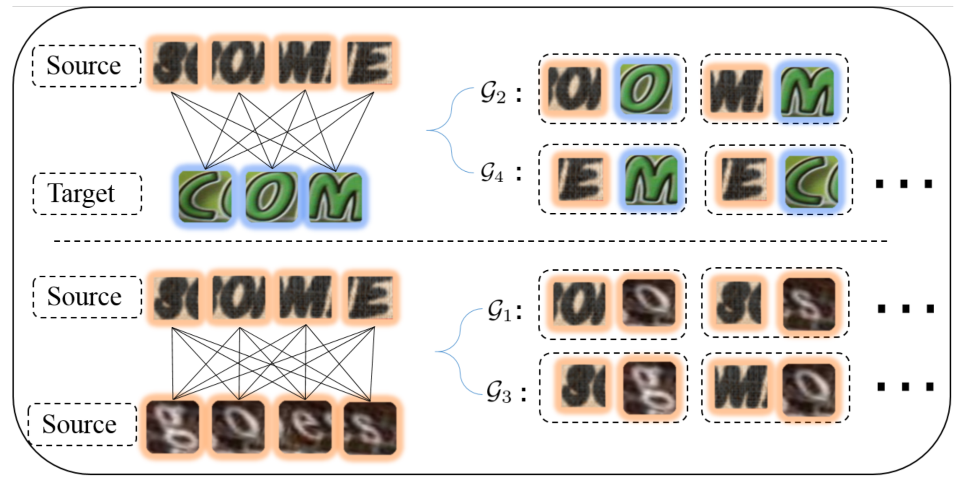
$$e_{t,j} = w^T tanh(W s_{t-1} + V x_j + b)$$

Define $\alpha'_{t,j}$ as the re-weighted attenting weight

$$\alpha'_{t,j} = \lambda\alpha_{t,j} + \frac{1-\lambda}{\eta(1+\eta)}\sum_{i=1}^{\eta} A(t, j-i)(\eta+1-i) + \frac{1-\lambda}{\eta(1+\eta)}\sum_{i=1}^{\eta} A(t, j+i)(\eta+1-i)$$

$$s.t. \quad A(t, j\pm i) = \begin{cases} \alpha_{t,j\pm i} & 1 \le j\pm i \le M \\ \alpha_{t,j} & otherwise \end{cases}$$

- Categories of character representation pair



$\mathcal{G}_1$ / $\mathcal{G}_2$: same class, same/different domain

$\mathcal{G}_3$ / $\mathcal{G}_4$: different class, same/different domain

- Adversarial learning target:

$$\mathcal{L}_D = -\sum_{i=1}^{4}\sum_{S\in\mathcal{G}_i} y_{\mathcal{G}_i} log(D(\phi(\mathcal{S}))) \qquad \mathcal{L}_G = -\left[\sum_{\mathcal{S}\in\mathcal{G}_2} y_{\mathcal{G}_1} log(D(\phi(\mathcal{S}))) + \sum_{\mathcal{S}\in\mathcal{G}_4} y_{\mathcal{G}_3} log(D(\phi(\mathcal{S})))\right]$$

Experiments

- Comparison with the state-of-the-art

Method	SVT		IC03			IC13	IC15
	50	None	50	Full	None	None	None
Yao *et al.*(2014)[45]	75.9	-	88.5	80.3	-	-	-
Jaderberg *et al.*(2016)[21]	95.4	80.7	98.7	**98.6**	93.1	90.8	-
Shi *et al.*(2017)[46]	96.4	80.8	98.7	97.6	89.4	86.7	-
Lee&Osindero (2016)[3]	96.3	80.7	97.9	97.0	88.7	90.0	-
Cheng *et al.*(2018)[25]	96	82.8	98.5	97.1	91.5	-	68.2
Bai *et al.*(2018)[1]	96.6	87.5	98.7	97.9	94.6	**94.4**	73.9
Liu *et al.*(2018)[24]	96.8	87.1	98.1	97.5	94.7	94.0	-
Shi *et al.*(2018)[5]	**99.2**	**93.6**	**98.8**	98.0	94.5	91.8	76.1
Li *et al.*(2019)[47]	98.5	91.2	-	-	-	94.0	**78.8**
Luo *et al.*(2019)[48]	96.6	88.3	98.7	97.8	**95.0**	92.4	68.8
Zhang *et al.*(2019)[11]	-	84.5	-	-	92.1	91.8	-
Shi *et al.*(baseline)(2016)[26]	96.1	81.5	97.8	96.4	88.7	87.5	-
Cheng *et al.*(baseline)(2017)[2]	95.7	82.2	98.5	96.7	91.5	89.4	63.3
Shi *et al.*(baseline)(2018)[5]	-	**91.6**	-	-	93.6	90.5	-
Luo *et al.*(baseline)(2019)[48]	-	84.1	-	-	92.5	90.0	68.8
Source Only	**96.8**	85.2	99.0	97.5	92.3	91.6	68.2
FT *w/* S+T	96.4	86.5	98.7	**97.6**	93.0	92.4	71.8
FASDA	96.5	88.3	**99.1**	97.5	**94.8**	**94.4**	**73.3**

Table 1. Results on SVT, ICDAR 2003, ICDAR 2013 and ICDAR 2015 datasets.

- Comparison with baselines

Method	SVT	IC03	IC13	IC15
Source Only	19.6	44.1	46.8	14.5
FT *w/* T	23.9	46.9	49.7	15.5
FT *w/* S+T	25.1	52.3	51.1	16.4
FASDA-$\mathcal{CR}$	27.5	55.8	54.9	18.6
FASDA-$\mathcal{CR}^{+}$	28.8	56.8	56.6	19.1
FASDA-IA-$\mathcal{CR}^{+}$	**29.4**	**58.1**	**57.5**	**19.2**

Table 2. Results on SVT, ICDAR 2003, ICDAR 2013 and ICDAR 2015 datasets.

FASDA shows the competitive performance with the state-of-the-art STR methods. And we can also see that FASDA performs clearly better than FT w/ S+T almost on all benchmarks.

Conclusion

We introduced FASDA to implement sequence-level domain adaptation for STR and it can maximize the character-level confusion between the source domain and the target domain to handle the scenarios that only have a few labeled samples. Our contribution can be summarized as below:

- achieve sequence-level domain confusion in STR
- the framework can be trained end-to-end with much fewer sequence-level annotations.