Supplementary Material for LISTER

Changxu Cheng, Peng Wang, Cheng Da, Qi Zheng, Cong Yao
DAMO Academy, Alibaba Group
{ccx0127,wdp0072012,dc.dacheng08,zhengqisjtu,yaocong2010}@gmail.com

1. Cases of Long Text Images

Long text occurs in our daily life frequently. It is a basic requirement to read long text for STR model. Fig. 1 gives some examples, including website, email address, file name, random code, compound word, etc.

2. More Illustration on Attention Sharpening

We would like to explain why we use Eq. (15) in the original paper to sharpen the character attention map.

Usually, the softmax function with temperature is exploited to sharpen probability distributions. Suppose that we apply it to our attention sharpening directly, *i.e.*,

$$\hat{A}^{(i)}_{j-1,s} = \frac{\exp\left(\alpha_j A^{(i)}_{j-1,s}\right)}{\sum_t \exp\left(\alpha_j A^{(i)}_{j-1,t}\right)} \tag{1}$$

Note that the exponential function can be converted as the following mathematically:

$$e^x = 1 + x + o(x) \tag{2}$$

where $o(x)$ is a high-order infinitesimal. If x_1 and x_2 are two infinitesimals tending to 0, and if $x_1 < x_2$, then we have:

$$\frac{e^{x_1}}{e^{x_1} + e^{x_2}} \approx \frac{1 + x_1}{1 + x_1 + 1 + x_2} > \frac{x_1}{x_1 + x_2} \tag{3}$$

$$\frac{e^{x_2}}{e^{x_1} + e^{x_2}} \approx \frac{1 + x_2}{1 + x_1 + 1 + x_2} < \frac{x_2}{x_1 + x_2} \tag{4}$$

which means the discrepancy (normalized ratio) between x_1 and x_2 is reduced after the softmax. In other words, the softmax function not only fails to sharpen the attention distribution, but flattens the distribution even more. It is because $+1$ in Eq. (2) dominates and dilutes x in the normalization when x is small. Since $0 \leq A^{(i)}_{j-1,s} \leq 1$, Eq. (1) has the same problem.

To avoid flattening the attention distribution, we simply *replace the exponential function in softmax (Eqs. (3) and (4)) with* $e^x - 1$. In this way, Eq. (1) evolves into Eq. (15) in the original paper. In experiments, we find that Eq. (15) is more insensitive to α_j and achieves better results.

Figure 1. Examples of long text in TUL.

Table 1. Comparison on MACs. The methods are all tested with input of size 32×128. For LISTER, the number of decoding steps is set to 12.

Method	Params (M)	MACs (G)
ABINet [3]	36.7	5.94
MGP-STR$_{\text{vision}}$ [6]	85.5	23.7
LISTER-B	49.9	2.69

3. Memory Access Cost

The low memory access costs (MACs) is another advantage of the proposed LISTER, which allows us to use a large batch size. As shown in Tab. 1, LISTER-B costs far less GPU memory than the hybrid convolution-Transformer architecture ABINet [3] and the fully-Transformer network MGP [6]. We owe it to the depth-wise convolution in the feature extractor, the proposed simple neighbor decoder, and the sliding-window self-attention layer that only takes aligned character features as input in the proposed FEM. Besides, the height of the final feature map is 1, which also matters.

4. Training using Real Dataset

Recently, some works [2, 1] trained their models using real text dataset. To evaluate LISTER more comprehensively, we further extend experiments by using the same real training dataset as in PARSeq [2]. The 1cycle learning rate scheduler [5] and Stochastic Weight Averaging (SWA) [4] are also used during training. The augmentation ways used in both ABINet [3] and PARSeq [2] are both exploited. The maximum text length is restricted to 32 to be efficient during training, while arbitrary during inference. The number of classes is still 37 to avoid inconsistence with the way of

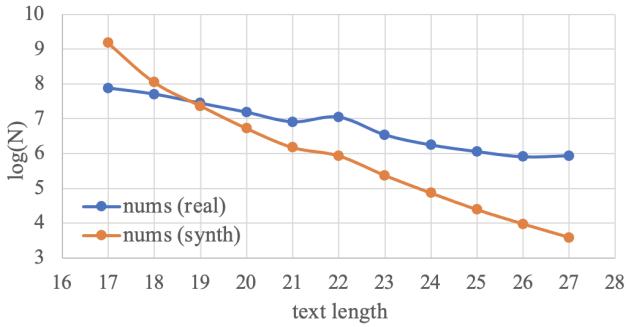

Figure 2. Comparison on length distribution (17-27) between the real and synthetic training set. The number is in logarithmic.

Table 2. Comparison of models trained using real data. LISTER-B^h is illustrated in Sec. 4.3. of the paper.

Method	CoB	ArT	COCO	Uber	AVG
ABINet [3]	95.9	81.2	76.4	71.5	74.6
PARSeq$_N$ [2]	95.7	**83.0**	77.0	82.4	82.1
LISTER-B	**96.4**	81.8	77.0	79.4	79.9
LISTER-B^h	96.3	82.8	**78.0**	**83.1**	**82.6**

length calculation.

4.1. Length Distribution Comparison

The real training set has much fewer samples than the widely-used synthetic dataset (MJ+ST), which is mainly reflected on short text. However, the text length in the real has a wider distribution. There are more long text images in the real set, as shown in Fig. 2.

4.2. Results

The results of models trained using real data are shown in Tab. 2. Among non-autoregressive models, LISTER performs the best except on ArT. The gap between LISTER-B and LISTER-B^h indicates that maintaining a proper height of feature map is necessary for irregular-shape text recognition.

The comparison on TUL is not appropriate here, since there are some overlaps between the real training set and TUL, as pointed in Sec. 4.1 in the paper. Nonetheless, LISTER achieves 88.6% on TUL, which is convincing enough compared with PARSeq$_A$ (80.6%).

References

[1] Jeonghun Baek, Yusuke Matsui, and Kiyoharu Aizawa. What if we only use real datasets for scene text recognition? toward scene text recognition with fewer labels. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3112–3121, 2021. 1

[2] Darwin Bautista and Rowel Atienza. Scene text recognition with permuted autoregressive sequence models. *ArXiv*, abs/2207.06966, 2022. 1, 2

[3] Shancheng Fang, Hongtao Xie, Yuxin Wang, Zhendong Mao, and Yongdong Zhang. Read like humans: Autonomous, bidirectional and iterative language modeling for scene text recognition. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7094–7103, 2021. 1, 2

[4] Pavel Izmailov, Dmitrii Podoprikhin, T. Garipov, Dmitry P. Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. In *Conference on Uncertainty in Artificial Intelligence*, 2018. 1

[5] Leslie N. Smith and Nicholay Topin. Super-convergence: Very fast training of residual networks using large learning rates. *ArXiv*, abs/1708.07120, 2017. 1

[6] Peng Wang, Cheng Da, and Cong Yao. Multi-granularity prediction for scene text recognition. In *European Conference on Computer Vision*, 2022. 1