Memory-Augmented Attention Model for Scene Text Recognition

Cong Wang[1,2], Fei Yin[1,2], Cheng-Lin Liu[1,2,3]

[1] National Laboratory of Pattern Recognition
Institute of Automation, Chinese Academy of Sciences

[2] University of Chinese Academy of Sciences

[3] CAS Center for Excellence of Brain Science and Intelligence Technology

Outline

- Introduction
- Motivation and the proposed method
- Experiments
- Conclusion

Introduction

- Scene text recognition
  - Bottom-up based methods
    - Detect individual characters and treat isolated character classification and subsequent word recognition separately.
  - Top-down based methods
    - The entire text from the original image is directly recognized.
  - State-of-the-art methods
    - Model scene text recognition as a sequence recognition problem
      - CTC framework
      - Attention mechanism

Motivation

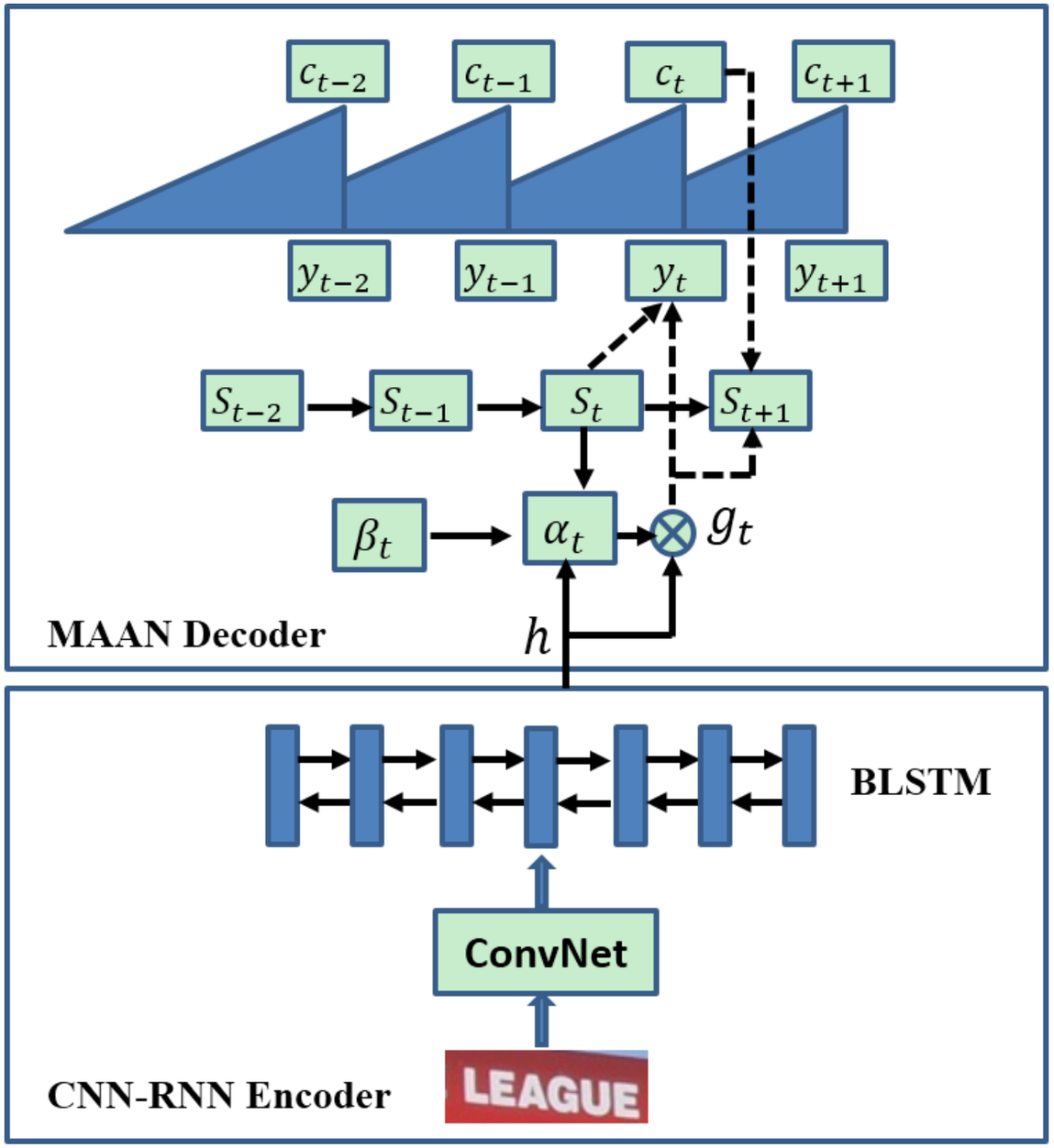
- The disadvantages of state-of-the-art attention-based methods for scene text recognition
  - They explicitly use the character label information only at the (t-1)-th time step when predicting the character at the t-th time step.
  - They do not make full use of the past alignment information.
- Solution
  - Propose a memory-augmented attention network (MAAN)
    - Augment the memory for historical label information.
    - Make full use of the alignment history.

MAAN

■ The architecture of the proposed MAAN model

Decoder

Encoder



MAAN

■ Sequence encoder

Table. The architecture of resnet-based CNN we adopt for feature extraction

Layer name	32 layers	Output size
Convolution	$3 \times 3,\ 1 \times 1,\ 1 \times 1,\ 32$	32×100
Convolution	$3 \times 3,\ 1 \times 1,\ 1 \times 1,\ 64$	32×100
Max-pooling	$2 \times 2,\ 2 \times 2,\ 0 \times 0$	16×50
Residual unit	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 1$	16×50
Convolution	$3 \times 3,\ 1 \times 1,\ 1 \times 1,\ 256$	16×50
Max-pooling	$2 \times 2,\ 2 \times 2,\ 0 \times 0$	8×25
Residual unit	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$	8×25
Convolution	$3 \times 3,\ 1 \times 1,\ 1 \times 1,\ 256$	8×25
Max-pooling	$1 \times 2,\ 1 \times 2,\ 0 \times 0$	4×25
Residual unit	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 5$	4×25
Convolution	$3 \times 3,\ 1 \times 1,\ 1 \times 1,\ 256$	4×25
Residual unit	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$	4×25
Convolution	$3 \times 3,\ 1 \times 2,\ 1 \times 1,\ 512$	2×25
Convolution	$2 \times 2,\ 1 \times 2,\ 0 \times 0,\ 512$	1×24

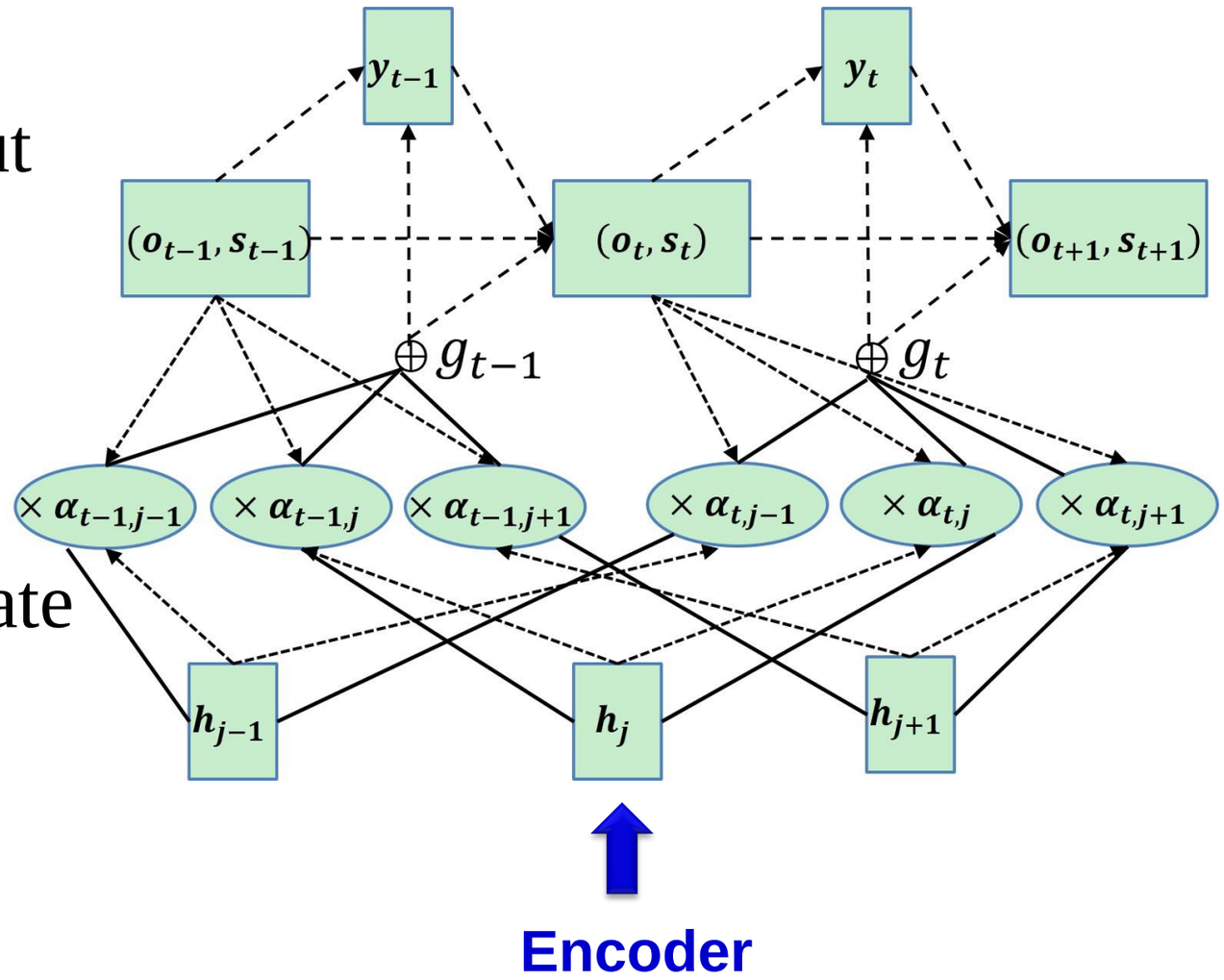
Input image size

MAAN

■ Content-based attention network

➢ At the t-th step, the total input x_t to the RNN is defined as

$$x_t = W_y y_{t-1} + W_{g_1} g_{t-1}$$



➢ The output o_t and the next state s_t of the RNN as follows

$$(o_t, s_t) = LSTM(x_t, s_{t-1})$$

➢ The probability of the output character y_t

$$P(y_t | y_1, \cdots, y_{t-1}, h) = softmax\left(W_o o_t + W_{g_2} g_t\right) \in \mathbb{R}^N$$

- Where N is the number of the character classes.

MAAN

■ Content-based attention network

➢ Glimpse vector g_t is defined as

$$g_t = \sum_{j=1}^{L} \alpha_{t,j} h_j$$

- Where $h = [h_1, \cdots, h_L]$ is the sequential input representation.

➢ The **vector of attention weights α_t** is defined as

$$e_{t,j} = v^T tanh\left(W s_t + V h_j + b\right)$$

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

MAAN

■ Memory-augmented attention network

➢ Augment the memory for historical label information

- Have access to the *k* previous characters by a one dimensional convolution

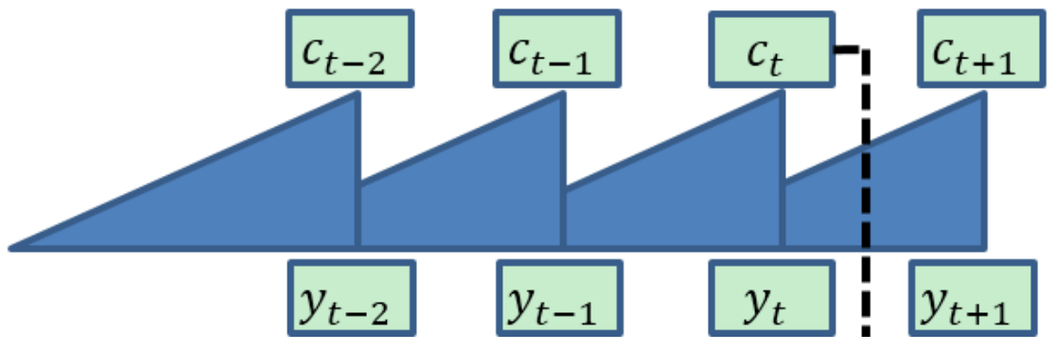
$$\boldsymbol{c_{t-1} = v\left(W_{glu}[y_{t-k}; \cdots; y_{t-1}] + b_{glu}\right) + y_{t-1}}$$

- Gated linear units (GLU)

$$v([A \;\; B]) = A \otimes \sigma(B)$$

- The total input to the RNN at the *t*-th time step is defined as

$$x_t = W_c c_{t-1} + W_{g_1} g_{t-1}$$



MAAN

■ Memory-augmented attention network

➢ Make full use of the alignment history

- Append a **coverage vector f_i** to keep the track of the past alignment information and augment the memory for the alignment history.

$$\beta_t = \sum_{l}^{t-1} \alpha_l$$

$$F = Q \star \beta_t$$

- The attention weights vector at the *t*-th time step:

$$e_{t,j} = v^T tanh\left(W s_{t-1} + V h_j + \boldsymbol{U f_j} + b\right)$$

- Guide the attention model to assign higher attention weights to the unfocused elements of the sequential input representation *h*.

MAAN

Training

- Loss function

$$\mathcal{L} = -\sum_{t} lnP(\hat{y}_t|I,\theta)$$

Transcription

- Lexicon-free transcription
  - Greedy decoding
    - Straightforwardly select the most probable character at each time step.
- Lexicon-based transcription
  - Choose the sequence in the lexicon that has smallest edit distance with the predicted label sequence via lexicon-free transcription.

Experiments

■ Datasets

➢ Training set

- Use the synthetic dataset (Synth) [1] for training
- Synth dataset contains 8-million training images with corresponding ground truth words.

➢ Test set

- IIIT5K-Words (IIIT5K)
- Street View Text (SVT)
- ICDAR 2003 (IC03)
- ICDAR 2013 (IC13)

[1] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, “Synthetic data and artificial neural networks for natural scene text recognition,” *arXiv preprint arXiv:1406.2227*, 2014.

Experiments

■ Implementation details

- Apply Momentum optimization algorithm with momentum = 0.9 to train the proposed model.
- The mini-batch size is set to 64.
- The learning rate is initially set to 0.02 and is decreased to 0.01 and 0.005 respectively after 150K mini-batch and 250K mini-batch.
- All the images in both training set and test set are resized to 32×100. Our model converges in 2 two days after about 3 epochs over the training set.
- The proposed model is implemented with Tensorflow framework and all our experiments are carried out on a workstation with one Titan X GPU.

■ Performance on Benchmarks

Method	IIIT5K			SVT		IC03			IC13
	50	1k	None	50	None	50	Full	None	None
ABBYY [2]	24.3	-	-	35.0	-	56.0	55.0	-	-
Wang et al. [2]	-	-	-	57.0	-	76.0	62.0	-	-
Mishra et al. [27]	64.1	57.5	-	73.2	-	81.8	67.8	-	-
Wang et al. [30]	-	-	-	70.0	-	90.0	84.0	-	-
Goel et al. [31]	-	-	-	77.3	-	89.7	-	-	-
Bissacco et al. [4]	-	-	-	90.4	78.0	-	-	-	87.6
Alsharif and Pineau [32]	-	-	-	74.3	-	93.1	88.6	-	-
Almázan et al. [6]	91.2	82.1	-	89.2	-	-	-	-	-
Yao et al. [5]	80.2	69.3	-	75.9	-	88.5	80.3	-	-
Rodrguez-Serrano et al. [33]	76.1	57.4	-	70.0	-	-	-	-	-
Jaderberg et al. [34]	-	-	-	86.1	-	96.2	91.5	-	-
Su and Lu [9]	-	-	-	83.0	-	92.0	82.0	-	-
Jaderberg et al. [7]	97.1	92.7	-	95.4	80.7	**98.7**	**98.6**	**93.1***	90.8
Jaderberg et al. [8]	95.5	89.6	-	93.2	71.7	97.8	97.0	89.6	81.8
Shi et al. [11]	97.8	95.0	81.2	**97.5**	82.7	**98.7**	**98.0**	91.9	89.6
Shi et al. [16]	96.2	93.8	81.9	95.5	81.9	98.3	96.2	90.1	88.6
Lee et al. [15]	96.8	94.4	78.4	96.3	80.7	97.9	97.0	88.7	90.0
Yin et al. [13]	98.9	96.7	81.6	95.1	76.5	97.7	96.4	84.5	85.2
Cheng et al. [17] (baseline)	98.9	96.8	83.7	95.7	82.2	98.5	96.7	91.5	89.4
Gao et al. [14]	**99.1**	**97.9**	81.8	97.4	82.7	98.3	96.2	90.1	88.6
Baseline	98.0	95.5	83.1	97.2	82.8	97.1	95.9	91.4	90.3
MAAN	98.3	96.4	**84.1**	96.4	**83.5**	97.4	96.4	**92.2**	**91.1**

*** Note that "Baseline" refers to the system with standard attention model (content-based attention network).**

Experiments

- Some examples of lexicon-free scene text recognition

Conclusion

■ Main contributions

- ➢ We propose a memory-augmented attention model for scene text recognition which augments the memory for historical label information and make full use of the alignment history.
- ➢ The whole network can be trained end-to-end.
- ➢ Experimental results on several benchmark datasets demonstrate that the proposed method achieves a comparable or even better performance compared with state-of-the-art methods.

■ Future work

- ➢ Make a further research on more efficient sequence learning model for scene text recognition.

Thanks for your attention!