EasyChair Preprint

№ 7037

A Self-Attention Based Model for Offline Handwritten Text Recognition

Nam Tuan Ly, Trung Tan Ngo and Masaki Nakagawa

November 11, 2021

A Self-Attention based Model for Offline Handwritten Text Recognition

Nam Tuan Ly[0000-0002-0856-3196], Trung Tan Ngo[0000-0002-8021-1072], and Masaki Nakagawa[0000-0001-7872-156X]

Tokyo University of Agriculture and Technology, Tokyo, Japan
{namlytuan, trungngotan94}@gmail.com, nakagawa@cc.tuat.ac.jp

Abstract. Offline handwritten text recognition is an important part of document analysis and it has been receiving a lot of attention from numerous researchers for decades. In this paper, we present a self-attention-based model for offline handwritten textline recognition. The proposed model consists of three main components: a feature extractor by CNN; an encoder by a BLSTM network and a self-attention module; and a decoder by CTC. The self-attention module is complementary to RNN in the encoder and helps the encoder to capture long-range and multi-level dependencies across an input sequence. According to the extensive experiments on the two datasets of IAM Handwriting and Kuzushiji, the proposed model achieves better accuracy than the state-of-the-art models. The self-attention map visualization shows that the self-attention mechanism helps the encoder capture long-range and multi-level dependencies across an input sequence.

Keywords: Self-Attention, Multi-Head, Handwritten Text Recognition, CNN, BLSTM, CTC.

1 Introduction

Offline handwritten text recognition is an important part of document analysis and it has been receiving a lot of attention from numerous researchers for decades. Starting with the recognition of isolated handwritten characters and digits, the focus has shifted to the recognition of words and sentences. Recognizing them is significantly more difficult than characters because of a large vacabulary in each language, and multiple touches between characters. Other challengings of offline handwritten recognition are various backgrounds, noises, and diversity of writing styles. Most of early works in handwritten Japanese/Chinese text recognition often took the segmentation-based approach that segmented or over-segmented text into characters and fragments and then merged the fragments in the recognition state [1, 2]. This segmentation-based approach is costly and error-prone because the segmentation of characters directly affects the whole system's performance. On the other hand, segmentation-free methods can avoid segmentation errors and have been employed for western handwritten documents based on the Hidden Markov Model (HMM) [3, 4] so far. However, a

weakness of HMMs is the local modeling, which cannot capture long-term dependencies in an input sequence.

In recent years, many segmentation-free methods have been proposed and proven to be powerful for both western and oriental text recognition [5–13] based on Recurrent Neural Network (RNN) and Connectionist Temporal Classification (CTC). The core recognition engine has been shifted from Hidden Markov Models (HMMs) to RNNs with CTC. The RNNs, such as Gated recurrent unit (GRU) or Long-short term memory (LSTM), are good at sequence modeling and solve the weakness of the local modeling of HMMs. However, the number of hidden nodes in RNNs is usually fixed, which implies all historical information is compressed into a fixed-length vector, so that RNNs are difficult to capture long-range context. Recently, A. Vaswani et al. [14] proposed a self-attention mechanism in the Transformer model, which achieved the state-of-the-art performance in some machine translation tasks. The self-attention mechanism can capture the dependencies between different positions of arbitrary distance in an input sequence and replaces the LSTM in both the encoder and the decoder of the sequence-to-sequence models.

In this paper, we present a self-attention-based model for offline handwritten text-line recognition. The proposed model consists of three main components: a feature extractor by CNN; an encoder by a BLSTM network and a self-attention module; and a decoder by CTC. The self-attention module complements RNN in the encoder and helps the encoder capture long-range and multi-level dependencies across an input sequence. According to our extensive experiments on the two datasets of IAM Handwriting and Kuzushiji, the proposed model achieves better accuracy than the state-of-the-art models. Furthermore, the self-attention map visualization shows that the self-attention mechanism helps the encoder capture the dependencies between different positions of arbitrary distance in an input sequence.

The rest of this paper is organized as follows: Section 2 describes the related work. Section 3 presents an overview of the proposed model. Section 4 reports our experiments, results and analysis. Finally, section 5 draws conclusions.

2 Related Work

In recent years, based on Deep Neural Networks, many segmentation-free methods have been proposed and shown to be effective, especially for recognizing noisy, complex, and handwritten text [5–10]. They can be categorized into two main approaches: CTC and attention-based sequence-to-sequence methods.

Early works of the CTC-based approach were introduced by A. Graves et al. [5, 6]. They proposed BLSTM followed by CTC for recognizing both online and offline handwritten English text and achieved better accuracy than HMM-based methods [5]. They also presented Multi-Dimensional LSTM (MDLSTM) with CTC for offline handwritten Arabic text recognition [6]. Following the works in [6], V. Pham et al. presented an end-to-end MDLSTM with dropout followed by CTC for handwritten text recognition [7]. B. Shi et al. proposed the combination of CNN and BLSTM, followed by CTC, which is called Convolutional Recurrent Neural Network (CRNN)

for image-based sequence recognition [8]. Based on the CRNN model, T. Bluche et al. proposed the Gated Convolutional Recurrent Neural Networks (GCRNN) for Multilingual Handwriting Recognition [9]. At the same time, N. T. Ly et al. presented the pre-trained CNN with sliding window followed by BLSTM with CTC to recognize offline handwritten Japanese text and achieve better accuracy than the traditional segmentation-based method [10]. J. Puigcerver et al. applied MDLSTM or CNN + LSTM, both followed by CTC for offline handwritten English and French text recognition [13].

The sequence-to-sequence (seq2seq) model with the attention mechanism has been proven to be a powerful model for many tasks, such as machine translation [15] and speech recognition [16]. Based on the attention-based seq2seq model, many segmentation-free models have been studied for image-based sequence recognition tasks [17–22]. J. Sueiras et al. presented an attention-based seq2seq model using a horizontal sliding window for handwritten English and French text recognition [17]. T. Bluche et al. proposed an attention-based end-to-end model with an MDLSTM network in the encoder for handwritten paragraph recognition [18, 19]. N. T. Ly et al. also proposed an attention-based seq2seq model with residual LSTM for recognizing multiple text-lines in Japanese historical documents [20, 21]. Zhang et al. presented an attention-based seq2seq model with a CNN-encoder and a GRU decoder for robust text image recognition [22]. Following the success of the self-attention mechanism in the Transformer model [14], L. Kang et al. presented the CNN-Transformer model for handwritten text line recognition [23]. Meanwhile, N. T. Ly et al. presented an Attention Augmented Convolutional Recurrent Network with a self-attention mechanism for Handwritten Japanese Text Recognition [24].

Recently, Trung et al. proposed the pretrained ResNet32 followed by RNN-Transducer for Japanese and Chinese offline handwritten text line recognition and achieved state-of-the-art accuracies on the SCUT-EPT and Kuzushiji datasets [25].

3 The Proposed Method

3.1 Self-Attention Mechanism

The self-attention mechanism is one of the main ideas of the Transformer model [14]. It uses all position-pairs in an input sequence to extract more expressive representations of the input. Therefore, the self-attention mechanism helps the model capture long-range and multi-level dependencies across the input sequence. To obtain these representations, the input sequence is linearly projected to get the *queries Q*, *keys K*, and *values V*. Then, the self-attention mechanism performs Scaled Dot-Product Attention to the *queries*, *keys*, and *values* to compute the output as shown in Eq. (1).

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{1}$$

where $Q \in R^{T_q \times d_k}$, $K \in R^{T_k \times d_k}$, and $V \in R^{T_k \times d_v}$; T and d are the number of features and the dimension of the feature in *Q*, *K*, *V*, respectively.

The self-attention mechanism can be further extended to the multi-head self-attention mechanism, which jointly attends to information from different representation subspaces at different positions. The multi-head self-attention mechanism firstly obtains h different representations of (Q, K, V) by linear projections and then independently performs the self-attention mechanism to each representation to get h heads. Finally, h heads are concatenated and then projected to produce the output encodings, as following:

$$\text{head}_\text{i} = \text{Attention}\left(QW_i^Q, KW_i^K, VW_i^V\right) \tag{2}$$

$$\text{MultiHead}(Q,K,V) = [\text{head}_1, \ldots, \text{head}_\text{h}]W^o \tag{3}$$

where h is the number of heads, $W_i^Q \in R^{d_\text{model} \times d_k}$, $W_i^K \in R^{d_\text{model} \times d_k}$, $W_i^V \in R^{d_\text{model} \times d_v}$, and $W^o \in R^{d_\text{model} \times d_\text{model}}$ are parameter matrices of the linear projections, d_model is the dimension of the input sequence, $d_\text{K} = d_V = d_\text{model}/h$.

The self-attention layer in Transformer [14] consists of two main components: a multi-head self-attention component and a position-wise fully connected feed-forward layer, as shown in Figure 1. A residual connection followed by layer normalization is applied after each of the two sub-components.

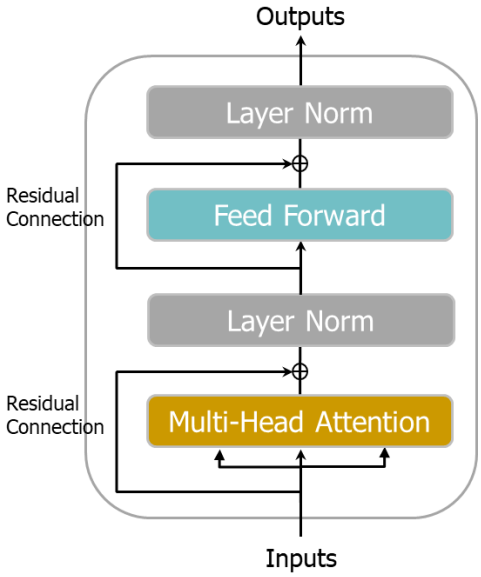

Figure 1. The self-attention layer.

3.2 Self-Attention based Convolutional Recurrent Neural Network

In this work, we propose a model of Self-Attention based Convolutional Recurrent Neural Network for recognizing each handwritten textline. The proposed model is composed of three main components: a feature extractor, an encoder, and a CTC-decoder, as shown in Figure 2. They are described in the following sections.

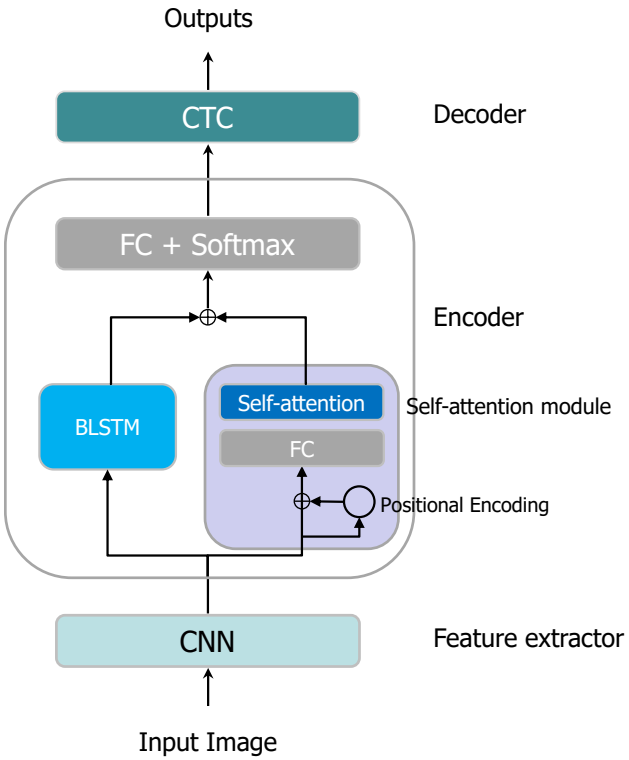

Figure 2. The network architecture of the proposed model.

Feature Extractor: At the bottom of the model, the feature extractor extracts a feature sequence from an input textline image. In this work, we use a standard CNN network without softmax and fully connected layers to build the feature extractor. Given an input image of size $w \times h \times c$ (where w, h, and c are the width, height and color channels of the image, respectively), the CNN network extracts a feature gird F of size $w' \times h' \times k$, where k is the number of feature maps in the last convolutional layer, and $w'$and h' depend on the w and h of input images and the number of pooling layers in the CNN network. Then, the feature grid F is unfolded into a feature sequence, as shown in Figure 3. Finally, the feature sequence will be fed into the encoder component.

Encoder: At the top of the feature extractor, the encoder converts the feature sequence extracted from the previous component into a sequence of label-probabilities. Mathematically, the encoder predicts label-probabilities from each feature in the feature sequence. In this model, the encoder consists of two main parts: a self-attention module and a BLSTM network. The self-attention module helps the encoder to capture long-range and multi-level dependencies across an input sequence. Meanwhile, the BLSTM network helps the encoder focus on the dependencies of nearby positions. The self-attention module consists of three sub-layers: a positional encoding layer

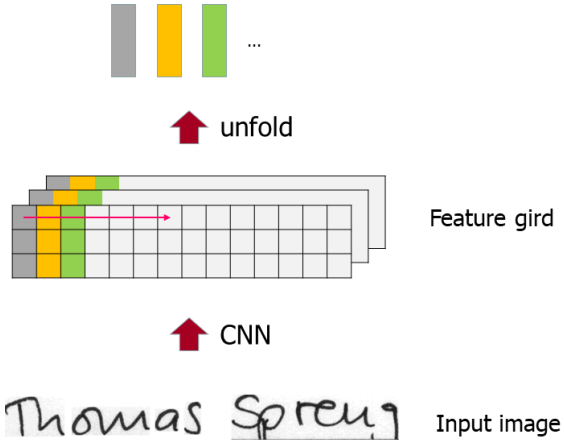

Figure 3. Feature extraction for an input image.

which helps the model to exploit the order of the feature sequence; a fully connected layer which reduces the dimensions of the features; and several self-attention layers, as shown in Figure 2.

Firstly, the feature sequence from the previous component is fed to the self-attention module and the BLSTM network. Then, the output of the self-attention module and the BLSTM network are concatenated and fed to the fully connected layer, which converts the output dimension to the size of the character set. Finally, one softmax layer is placed in the end to predict the label-probabilities at each time step. Let $F = (f_1, f_2 \cdots f_n)$, and $E = (e_1, e_2 \cdots e_n)$ denote the feature sequence, and the sequence of label probabilities, respectively, where n is the number of feature vectors. Then, we have:

$$E = \text{Softmax}\left(\text{Concat}\left(\text{BLSTM}\left(F\right), \text{SelfAttn}\left(F\right)\right)\right) \tag{4}$$

Decoder: At the top of the encoder, the decoder decodes the sequence of label probabilities made by the encoder into a final target sequence. Mathematically, the decoding process finds the final label sequence with the highest probability conditioned on the sequence of label probabilities. In this work, we use the CTC [26] algorithm to build the decoder to obtain conditional probability.

The whole model is trained end-to-end using stochastic gradient descent algorithms to minimize the CTC loss. For the decoding process in the testing phase, we apply the CTC beam search with the *beamwidth* of *2* to obtain the final label sequence with the highest probability conditioned.

4 Experiments

To evaluate the performance of the proposed model, we conducted experiments on the two datasets: IAM handwriting and Kuzushiji datasets. The information of the da-

tasets is given in Sec 4.1; the implementation details are described in Sec 4.2; the results of the experiments are presented in Sec. 4.3; and the analysis is shown in Sec. 4.4.

4.1 Datasets

In this paper, we use two datasets for experiments. IAM Handwriting is offline handwritten English text dataset compiled by the FKI-IAM Research Group. The dataset is composed of 13,353 textlines extracted from 1,539 pages of scanned handwritten English text which were written by 657 different writers. We employ the shared Aachen data splits by T. Bluche from RWTH Aachen University to split the dataset into three subsets: 6,482 lines for training, 2,915 lines for testing, and 976 lines for validation. There are 79 different characters in the dataset, including the space character. The summary of the IAM handwriting is given in Table 1.

Table 1. IAM Handwriting dataset.

	IAM Handwriting		
	Train set	**Valid set**	**Test set**
Text lines	6,482	976	2,915
Pages	747	116	336

Kuzushiji is a dataset of the pre-modern Japanese documents prepared by the National Institute of Japanese Literature (NIJL). The first version of the Kuzushiji (Kuzushiji_v1) dataset consists of 15 pre-modern Japanese books composing of 2,222 pages and was released in 2016 [27]. For every character in each page, its bounding box, location and Shift_JIS code are annotated. The Kuzushiji_v1 textline dataset, consisting of 25,875 textline images, was compiled from the Kuzushiji_v1 dataset. We use all textline images collected from the 15th book as the testing set. The remaining images are divided randomly from the training and validation sets with a ratio of 9:1. Table 2 shows the profile of the Kuzushiji_v1 textline dataset.

Table 2. Kuzushiji_v1 textline dataset.

	Kuzushiji_v1 textline		
	Train set	**Valid set**	**Test set**
Samples	19,797	2,200	3,878
Books	1st~14th		15th

4.2 Implementation Details

In the experiments, the architecture of the CNN network in the feature extractor is deployed as shown in Table 3. It consists of five (six for the Kuzushiji_v1 textline dataset) convolutional (Conv) blocks. Each Conv block consists of one Conv layer with a kernel size of 3×3 pixels and a stride of 1×1 pixel followed by the Batch nor-

malization [28] and the ReLU activation. To reduce overfitting, we apply dropout at the input of the last three Conv blocks (with dropout probability equal to 0.2).

Table 3. Network configurations of the CNN in the feature extractor.

Config.	Values	
	IAM	**Kuzushiji_v1 textline**
Input	128×w	128×w
Conv Block	16 - 32 - 48 - 64 - 80	16 - 32 - 48 - 64 - 80 - 128
Max-Pooling	(2,2) - (2,2) - (1,2) - (2,1) - No	(2,2) - (2,2) - (2,2) - (1,2) - (2,1) - No
Dropout	0 - 0 - 0.2 - 0.2 - 0.2	0 - 0 - 0 - 0.2 - 0.2 - 0.2

At the encoder, we use a Deep BLSTM network with 256 hidden nodes of three layers (two layers for the Kuzushiji_v1 textline dataset). To prevent overfitting when training the model, the dropout (dropout rate=0.5) is also applied in each layer of the Deep BLSTM network. The self-attention module consists of six self-attention layers where each self-attention layer is composed of eight heads and 2,048 nodes of a single full connected layer. the BLSTM network and the self-attention module are followed by a fully connected layer and a softmax layer with the node size equal to the character set size (n=80 for IAM and 4,818 for Kuzushiji).

4.3 Experiments

In order to evaluate the performance of the proposed model, we use the terms of Character Error Rate (CER), Word Error Rate (WER), and Sequence Error Rate (SER) that are defined as follows:

$$\mathrm{CER}(\mathrm{h},\mathrm{S}') = \frac{1}{\mathrm{Z}} \sum_{(\mathrm{x},\mathrm{z})\in\mathrm{S}'} \mathrm{ED}(\mathrm{h}(\mathrm{x}),\mathrm{z}) \tag{5}$$

$$\mathrm{WER}(\mathrm{h},\mathrm{S}') = \frac{1}{\mathrm{Z}_{word}} \sum_{(\mathrm{x},\mathrm{z})\in\mathrm{S}'} \mathrm{ED}_{word}(\mathrm{h}(\mathrm{x}),\mathrm{z}) \tag{6}$$

$$\mathrm{SER}(\mathrm{h},\mathrm{S}') = \frac{100}{|\mathrm{S}'|} \sum_{(\mathrm{x},\mathrm{z})\in\mathrm{S}'} \begin{cases} 0 & \text{if} \quad \mathrm{h}(\mathrm{x})=\mathrm{z} \\ 1 & \text{otherwise} \end{cases} \tag{7}$$

where Z is the total number of target labels in S' and $ED(p, q)$ is the edit distance between two sequences p and q, while Z_{word} is the total number of words in S', and $ED_{word}(p, q)$ is the word-level edit distance between two sequences p and q.

4.3.1. Effects of the Self-Attention Mechanism

In the first experiment, we evaluate the effectiveness of the self-attention module and the fully connected layer in the self-attention module. We prepared two variations. The first one is the same as the proposed model (called SA-CRNN) except using the self-attention module, which is named SA-CRNN_w/o_SelfAttn. The second one is

the same as the proposed model except the fully connected layer in the self-attention module, which is named SA-CRNN_w/o_FC. Table 4 compares their recognition error rates with the proposed model on the test set of the IAM Handwriting dataset. The proposed model slightly outperforms SA-CRNB_w/o_SelfAttn. The results imply that the self-attention module in the encoder improves the performance of the CRNN model for handwritten text recognition. This seems to be due to the self-attention module that helps the encoder capture long-range dependencies in an input sequence. The proposed model again slightly outperforms SA-CRNN_w/o_FC. The results show that the fully connected layer in the self-attention module improves the performance of the proposed model.

Table 4. Recognition error rates (%) with different encoders.

Model	IAM	
	CER	**WER**
SA-CRNN_w/o_SelfAttn	7.54	24.06
SA-CRNN_w/o_FC	7.59	23.85
SA-CRNN	**7.22**	**22.87**

The second experiment explores the effect of head number in the self-attention mechanism. We performed experiments with a different head numbers of 1, 2, 4, and 8 on the IAM dataset. The results are shown in Table 5. As can be seen, the proposed model obtained its best CER when the head number was 8, while the best WER was obtained with the head number of 4.

Table 5. Recognition error rates (%) with a different head numbers.

Head Number	IAM	
	CER	**WER**
1	7.51	23.97
2	7.45	23.43
4	7.28	**22.80**
8	**7.22**	22.87

4.3.2. Comparison with the state-of-the-art

The third experiment evaluates the performance of the proposed model and compares it with the previous works on the IAM Handwriting dataset in terms of CER and WER. To fairly compare with the previous models [7, 9, 13, 17, 19, 22, 23, 29], we do not use any data augmentation techniques as well as linguistic context information. The results are shown in Table 6. The proposed model achieved CER of 7.22% and WER of 22.87%. These results show that the proposed model achieves state-of-the-art accuracy and outperforms the best model in [23] by about 6% of CER and 9% of WER on the IAM Handwriting dataset without data augmentation techniques as well as linguistic context information.

Table 6. Recognition error rates (%) on the IAM dataset.

Model	IAM	
	CER	**WER**
CNN-1DLSTM (Moysset et al. [29])	11.52	35.64
MDLSTM (Pham et al. [7])	10.80	35.10
GNN-1DLSTM (Bluche et al. [9])*	10.17	32.88
2DLSTM (Moysset et al. [29])	8.88	29.15
2DLSTM-X2 (Moysset et al. [29])	8.86	29.31
CNN-Seq2Seq (Sueiras et al. [17])	8.80	23.80
CNN-Seq2Seq (Zhang et al. [22])	8.50	22.20
CNN-1DLSTM (Puigcerver et al. [13])	8.20	25.40
2DLSTM (Bluche et al. [19])	7.90	24.60
CNN-1DLSTM (Puigcerver et al. [13])*	7.73	25.22
CNN-Transformers (Kang et al. [23])	7.62	24.54
The Proposed Model (Ours)	**7.22**	**22.87**

* Experiments run by Moysset et al. [29]

In the second experiment, we evaluate the performance of the proposed model and compare it with the previous works [11, 20, 24, 25] on the Kuzushiji_v1 textline dataset in terms of CER and SER. We also do not use any data augmentation techniques as well as linguistic context information. As shown in Table 7, the proposed model achieved CER of 20.25% and SER of 94.53% on the test set of the Kuzushiji_v1 textline dataset, which is best among the previous methods without data augmentation and linguistic context. Furthermore, the proposed model also has lower CER than the RNN-Transducer model in the works of [25] which applied data augmentation techniques during the training process.

Table 7. Recognition error rates (%) on the Kuzushiji_v1 textline dataset.

Model	Kuzushiji_v1 textline	
	CER	**SER**
End-to-End DCRN [11]	28.34	97.27
Attention-based seq2seq model [20]	31.38	98.07
AACRN [24]	21.48	94.97
The proposed model (Ours)	**20.25**	**94.53**
RNN-Transducer [25] *	20.33	-

* Data augmentations + Pretrained ResNet32 on ImageNet.

4.4 Analysis

Figure 4 shows the visualization of the multi-head self-attention maps for one image. The top image is the original input image, while each of the two groups of four images shows one query column with the color-coded location (blue and red) and four self-attention maps of four attention heads (eight attention heads in total) for that que-

ry column. These visualizations show that the self-attention mechanism helps the encoder capture long-range and multi-level dependencies across the input sequence.

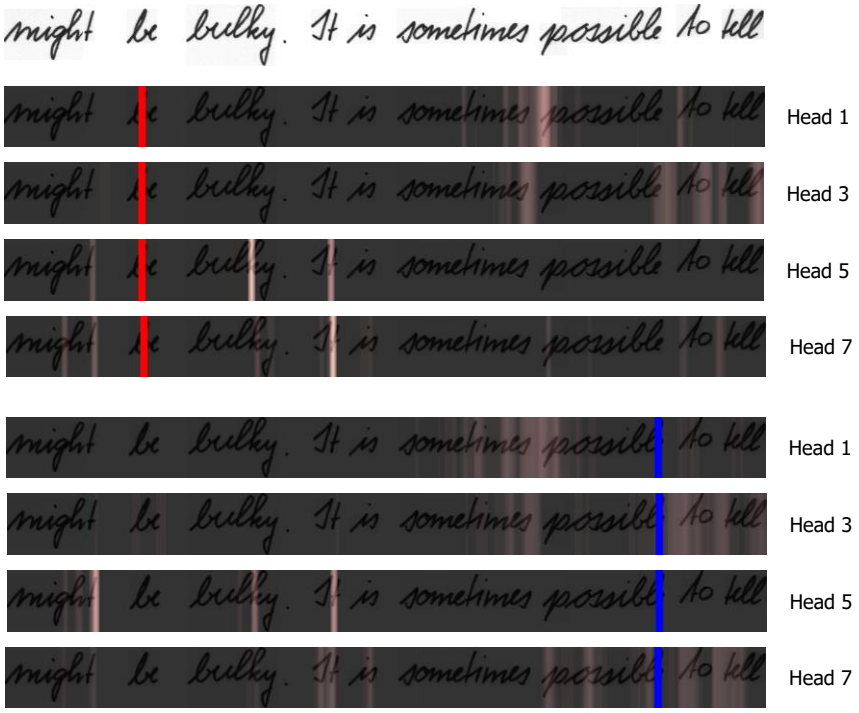

Figure 4. The visualization of multi-head self-attention maps.

Figure 5 shows some correctly recognized and misrecognized samples in the Kuzushiji_v1 textline dataset by the proposed model. For each misrecognized sample, the image on the left is an input image, the text bounded by the blue rectangular shows the ground-truth, and the text bounded by the red rectangular shows the recognition resulted.

5 Conclusion

In this paper, we proposed an self-attention-based model for recognizing offline handwritten textlines. We introduced the self-attention mechanism into the encoder component to help the encoder to capture long-range and multi-level dependencies across an input sequence. The proposed model achieves better accuracy than the state-of-the-art models on the two datasets of IAM Handwriting and Kuzushiji_v1 textline. We also visualized the self-attention map and observed that the self-attention mechanism helps the encoder capture long-range and multi-level dependencies across an input sequence.

a). Correctly recognized samples. b). Misrecognized samples.

Figure 5. Correctly recognized and misrecognized samples by the proposed model.

In future works, we will conduct experiments of the proposed model with other text recognition tasks such as scene text recognition and historical text recognition. We also plan to apply data augmentations and incorporate language models into the proposed model to improve its performance.

Acknowledgments

This research is being partially supported by the grant-in-aid for scientific research (S) 18H05221 and (A) 18H03597.

References

1. Nguyen, K.C., Nguyen, C.T., Nakagawa, M.: A Segmentation Method of Single- and Multiple-Touching Characters in Offline Handwritten Japanese Text Recognition. IEICE Trans. Inf. Syst. E100.D, 2962–2972 (2017).
2. Qiu-Feng Wang, Fei Yin, Cheng-Lin Liu: Handwritten Chinese Text Recognition by Integrating Multiple Contexts. IEEE Trans. Pattern Anal. Mach. Intell. 34, 1469–1481 (2012).

3. El-Yacoubi, A., Gilloux, M., Sabourin, R., Suen, C.Y.: An HMM-based approach for off-line unconstrained handwritten word modeling and recognition. IEEE Trans. Pattern Anal. Mach. Intell. 21, 752–760 (1999).
4. España-Boquera, S., Castro-Bleda, M.J., Gorbe-Moya, J., Zamora-Martinez, F.: Improving offline handwritten text recognition with hybrid HMM/ANN models. IEEE Trans. Pattern Anal. Mach. Intell. 33, 767–779 (2011).
5. Graves, A., Liwicki, M., Fernandez, S., Bertolami, R., Bunke, H., Schmidhuber, J.: A Novel Connectionist System for Unconstrained Handwriting Recognition. IEEE Trans. Pattern Anal. Mach. Intell. 31, 855–868 (2009).
6. Graves, A., Schmidhuber, J.: Offline Handwriting Recognition with Multidimensional Recurrent Neural Networks. Adv. Neural Inf. Process. Syst. 21, NIPS'21. 545–552 (2008).
7. Pham, V., Bluche, T., Kermorvant, C., Louradour, J.: Dropout Improves Recurrent Neural Networks for Handwriting Recognition. In: Proceedings of International Conference on Frontiers in Handwriting Recognition, ICFHR. pp. 285–290 (2014).
8. Shi, B., Bai, X., Yao, C.: An End-to-End Trainable Neural Network for Image-Based Sequence Recognition and Its Application to Scene Text Recognition. IEEE Trans. Pattern Anal. Mach. Intell. 39, 2298–2304 (2017).
9. Bluche, T., Messina, R.: Gated Convolutional Recurrent Neural Networks for Multilingual Handwriting Recognition. In: Proceedings of the International Conference on Document Analysis and Recognition, ICDAR. pp. 646–651 (2017).
10. Ly, N.-T., Nguyen, C.-T., Nguyen, K.-C., Nakagawa, M.: Deep Convolutional Recurrent Network for Segmentation-Free Offline Handwritten Japanese Text Recognition. In: Proceedings of the International Conference on Document Analysis and Recognition (ICDAR). pp. 5–9 (2017).
11. Ly, N.T., Nguyen, C.T., Nakagawa, M.: Training an End-to-End Model for Offline Handwritten Japanese Text Recognition by Generated Synthetic Patterns. In: Proceedings of the International Conference on Frontiers in Handwriting Recognition (ICFHR). pp. 74–79 (2018).
12. Ly, N.T., Nguyen, K.C., Nguyen, C.T., Nakagawa, M.: Recognition of anomalously deformed kana sequences in Japanese historical documents. IEICE Trans. Inf. Syst. E102D, (2019).
13. Puigcerver, J.: Are Multidimensional Recurrent Layers Really Necessary for Handwritten Text Recognition? In: Proceedings of the International Conference on Document Analysis and Recognition, ICDAR. pp. 67–72 (2017).
14. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems. pp. 5999–6009 (2017).
15. Luong, T., Pham, H., Manning, C.D.: Effective Approaches to Attention-based Neural Machine Translation. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. pp. 1412–1421 (2015).
16. Bahdanau, D., Chorowski, J., Serdyuk, D., Brakel, P., Bengio, Y.: End-to-end attention-based large vocabulary speech recognition. In: Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 4945–4949 (2016).

17. Sueiras, J., Ruiz, V., Sanchez, A., Velez, J.F.: Offline continuous handwriting recognition using sequence to sequence neural networks. Neurocomputing. 289, 119–128 (2018).
18. Bluche, T., Louradour, J., Messina, R.: Scan, Attend and Read: End-to-End Handwritten Paragraph Recognition with MDLSTM Attention. In: Proceedings of the International Conference on Document Analysis and Recognition (ICDAR). pp. 1050–1055 (2017).
19. Bluche, T.: Joint Line Segmentation and Transcription for End-to-End Handwritten Paragraph Recognition. Neural Inf. Process. Syst. (2016).
20. Ly, N.T., Nguyen, C.T., Nakagawa, M.: An attention-based end-to-end model for multiple text lines recognition in japanese historical documents. In: Proceedings of the International Conference on Document Analysis and Recognition, ICDAR. pp. 629–634 (2019).
21. Ly, N.T., Nguyen, C.T., Nakagawa, M.: An attention-based row-column encoder-decoder model for text recognition in Japanese historical documents. Pattern Recognit. Lett. 136, 134–141 (2020).
22. Zhang, Y., Nie, S., Liu, W., Xu, X., Zhang, D., Shen, H.T.: Sequence-to-sequence domain adaptation network for robust text image recognition. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. pp. 2735–2744 (2019).
23. Kang, L., Riba, P., Rusiñol, M., Fornés, A., Villegas, M.: Pay Attention to What You Read: Non-recurrent Handwritten Text-Line Recognition. arXiv. (2020).
24. Ly, N.T., Nguyen, C.T., Nakagawa, M.: Attention Augmented Convolutional Recurrent Network for Handwritten Japanese Text Recognition. In: Proceedings of International Conference on Frontiers in Handwriting Recognition, ICFHR. pp. 163–168 (2020).
25. Ngo, T.T., Nguyen, H.T., Ly, N.T., Nakagawa, M.: Recurrent neural network transducer for Japanese and Chinese offline handwritten text recognition. arXiv. (2021).
26. Graves, A., Fernández, S., Gomez, F., Schmidhuber, J.: Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In: Proceedings of the ACM International Conference Proceeding Series. pp. 369–376 (2006).
27. Kuzushiji dataset, http://codh.rois.ac.jp/char-shape/book/, last accessed 2020/03/07.
28. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: Proceedings of the International Conference on Machine Learning, ICML 2015. pp. 448–456 (2015).
29. Moysset, B., Messina, R.: Are 2D-LSTM really dead for offline text recognition? In: International Journal on Document Analysis and Recognition. pp. 193–208 (2019).