Neural Computing and Applications
https://doi.org/10.1007/s00521-021-05813-1

ORIGINAL ARTICLE

Character-based handwritten text transcription with attention networks

Jason Poulos[1,2] · Rafael Valle[3]

Received: 6 March 2020 / Accepted: 5 February 2021

Abstract

The paper approaches the task of handwritten text recognition (HTR) with attentional encoder–decoder networks trained on sequences of characters, rather than words. We experiment on lines of text from popular handwriting datasets and compare different activation functions for the attention mechanism used for aligning image pixels and target characters. We find that softmax attention focuses heavily on individual characters, while sigmoid attention focuses on multiple characters at each step of the decoding. When the sequence alignment is one-to-one, softmax attention is able to learn a more precise alignment at each step of the decoding, whereas the alignment generated by sigmoid attention is much less precise. When a linear function is used to obtain attention weights, the model predicts a character by looking at the entire sequence of characters and performs poorly because it lacks a precise alignment between the source and target. Future research may explore HTR in natural scene images, since the model is capable of transcribing handwritten text without the need for producing segmentations or bounding boxes of text in images.

Keywords Attention · Convolutional neural networks · Handwritten text recognition · Recurrent neural networks

1 Introduction

Handwritten text recognition (HTR) on character sequences is an open research problem because it is harder to segment and recognize individual characters, rather words [1]. Moreover, transcription models must solve the problem of finding and classifying characters at each time-step without knowing the alignment between the input sequence of image pixels and the target sequence of characters [2].

Previous approaches to HTR include using a hidden Markov model (HMM), or HMM-neural network hybrid, to match image features to character labels. The HMM approach is outperformed by models that combine a single recurrent neural network (RNN) with a connectionist temporal classification (CTC) output layer [3–8]. The CTC-based models calculate a probability distribution over all possible target sequences, conditional on the input sequence. The CTC-based models assume strict monotonicity in input-target sequence alignments, and generally assume a target sequence length that is bounded by the input sequence length.

In this work, we employ the encoder–decoder networks proposed by Deng et al. [9], which extends the encoder–decoder RNNs of Vinyals et al. [10] and Bahdanau et al. [11] for the problem of decompiling images into presentational markup. The encoder–decoder model encodes a variable-length sequence of characters into a fixed-length vector and then decodes the vector into a variable-length target label. Encoder–decoder RNNs are suitable for handling long sequences of data and have become standard for neural machine translation, speech recognition [12], and image captioning [13] tasks.

The model of Deng et al. consists of a convolutional neural network (CNN) that extracts visual features from the images and arrange the features on a grid. An RNN encoder re-encodes each row of the grid, learning additional features such as text directionality. Lastly, an RNN decoder outputs a character sequence one step at a time, using an

✉ Jason Poulos
jason.poulos@duke.edu

[1] Department of Statistical Science, Duke University, Durham, North Carolina, USA

[2] The Statistical and Applied Mathematical Sciences Institute, Durham, North Carolina, USA

[3] NVIDIA Corporation, Santa Clara, California, USA

Published online: 27 February 2021

attention mechanism to emphasize the most important columns of re-encoded features at each decoding step. The use of attention mechanism in the decoder relaxes the monotonicity assumption of the CTC-based model, and improves the ability of the encoder–decoder networks to learn the correct alignment between image pixels and target characters, and to extract the most relevant information for each part of the output sequence [14]. Attention-based networks are capable of modeling the language structures within the output sequence, rather than simply mapping the input to the correct output [15].

Encoder–decoder RNNs have been previously employed for recognizing text in natural images [16, 17], and more recently for HTR. Several recent papers propose a hybrid architecture consisting of a CNN to encode the input image and an RNN decoder to predict sequences of characters [18–26]. For example, Sueiras et al. [27] and Kang et al. [28] use attentional encoder–decoder networks very similar to ours, but train their model to transcribe words, rather than sequences of characters, and employ a word-based lexicon (i.e., a list of words found in the training set) for decoding.

The main differentiator in our approach is that we employ a CNN to extract image features and a separate RNN encoder to re-encode the features so that the encoder can learn new features such as text directionality. Another difference is that we use an unidirectional RNN decoder to predict the sequence of characters. Gui et al. [29] train character-aware attention networks, but the architecture differs in that they use an attention-based bidirectional RNN decoder and CTC output layer to convert predictions made by the decoder into a character sequence.

There are recent developments towards architecture based entirely on CNNs or attention mechanisms, bypassing any recurrence. Fully convolutional architectures have performed well against encoder–decoder networks on neural machine translation tasks [30], handwriting generation [31, 32], and HTR tasks [33–38]. The entirely attention-based transformer model initially proposed by Vaswani et al. [39] have outperformed encoder–decoder networks on several HTR tasks [40].

In this work, we focus on developing character-aware models for HTR. Character-aware models view the input and output text lines as a sequence of characters rather than words, and each character prediction is explicitly conditioned on the previous character. These models are capable of making inferences about unseen source words and also generating unseen target words. In addition, character-aware models do not require lexicons because only characters are explicitly modeled [41].

Our primary contributions are applying character-aware attention networks to the task of transcribing lines of unconstrained (i.e., cursive or overlapping) handwritten text and comparing different activation functions for the attention mechanism. Section 2 describes attention networks in the context of character-based HTR. Section 3 describes the benchmark datasets used for the experiments and provides details on the network architecture and training. Section 4 describes the results on benchmark datasets, comparing the performance of different attention mechanisms. Section 5 concludes and suggests directions for future research.

2 Attention networks for character-based HTR

The character-based HTR problem is one of converting images to hand-transcribed sequences of discrete characters. Following the notation of Deng et al., the input $\mathbf{x} \in \mathcal{X}$ is an image with height and width dimensions $\mathbb{R}^{H \times W}$. The target $\mathbf{y} \in \mathcal{Y}$ consists of a sequence of characters, $y_1, y_2, \ldots, y_T$, where T is the sequence length and each character exists within a known vocabulary, Σ. The supervised task is to learn a function that maps $\mathcal{X} \rightarrow \mathcal{Y}$ using training example pairs $(\mathbf{x}, \mathbf{y})$.

The general architecture of the attention networks of Deng et al., which we extend for HTR, is illustrated in Fig. 1. The CNN inputs $\mathbf{x}$ and arranges the visual features on a grid, $\mathbf{V}$ with dimensions $H' \times W' \times C$, where C is the number of channels, and H' and W' are reduced dimensions following max pooling operations.

The RNN encoder slides across each row of $\mathbf{V}$, and at each time-step t, recursively updates a hidden state $\mathbf{h}_t$ using $\mathbf{v}_t \in \mathbf{V}$ as input:

$$\mathbf{h}_t = f(\mathbf{v}_t, h_{t-1}; \theta), \tag{1}$$

where $f(\cdot)$ is a nonlinear activation and θ is a learned parameter. The encoder outputs a re-encoded feature grid $\tilde{\mathbf{V}}_{h,w} = \mathrm{RNN}(\tilde{\mathbf{V}}_{h,w-1}, \mathbf{V}_{h,w})$, for rows $h \in \{1, \ldots, H'\}$ and columns $w \in \{1, \ldots, W'\}$. Encoding row-wise is useful for transcription tasks because the encoder can learn sequential order information, such as text directionality. The networks capture column-wise sequential information by learning a positional encoding in the form of an initial hidden state, $\tilde{\mathbf{V}}_{h,0}$, which is added to each row of $\tilde{\mathbf{V}}$.

The decoder RNN learns a conditional language model to give the probability of the next character given the history and re-encoded feature grid:

$$p(y_{t+1} | y_1, \ldots, y_t, \tilde{\mathbf{V}}) = \mathrm{softmax}(\mathbf{W}_1 \mathbf{o}_t), \tag{2}$$

$$\text{where} \quad \mathbf{o}_t = f(\mathbf{W}_2[\mathbf{h}'_t; \mathbf{c}_t]). \tag{3}$$

In the above equations, the matrices $\mathbf{W}_1$ and $\mathbf{W}_2$ are learned parameters of the model, and the softmax

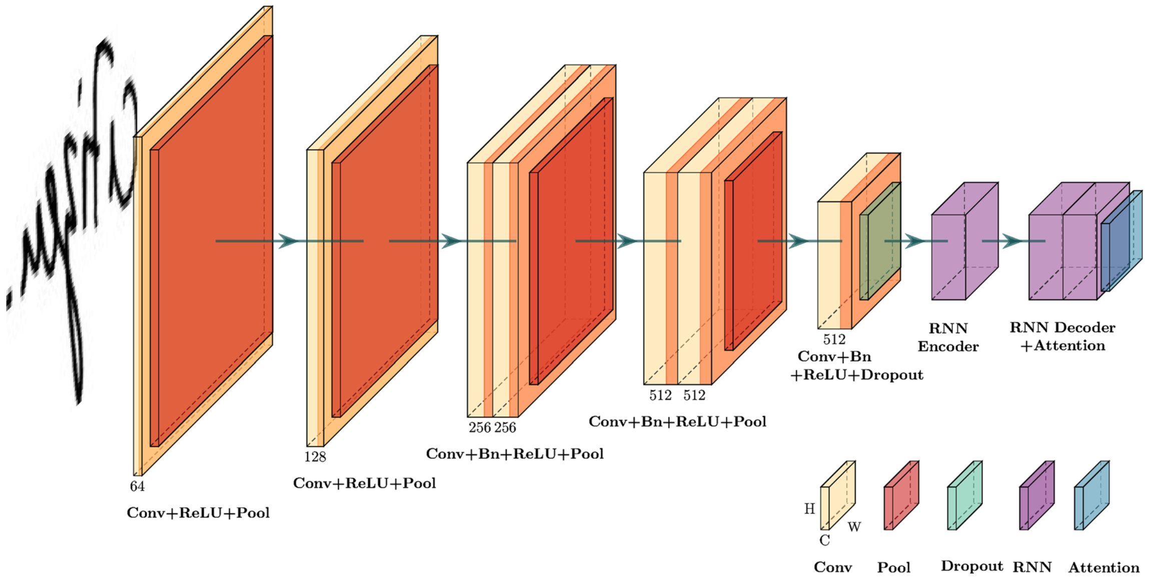


Fig. 1 Attention networks architecture. *Notes:* 'Conv': convolution layer, 'Pool' max-pooling layer, 'Bn': batch normalization

activation function assigns probabilities over Σ. The hidden state of the RNN decoder, $\mathbf{h}'_t$, is updated recursively by

$$\mathbf{h}'_t = f(\mathbf{h}'_{t-1}, \mathbf{y}_{t-1}; \theta'), \tag{4}$$

where θ' is a learned parameter. The context vector, $\mathbf{c}_t$, provides the most important elements of the re-encoded feature grid at each t:

$$\mathbf{c}_t = \sum_{h,w} \boldsymbol{\alpha}_t \tilde{\mathbf{V}}_{h,w}, \tag{5}$$

$$\text{where} \qquad \boldsymbol{\alpha}_t = \mathrm{softmax}(a(\mathbf{h}'_t, \tilde{\mathbf{V}}_{h,w})), \tag{6}$$

$$\text{and} \qquad a_{t,h,w} = \boldsymbol{\beta}^{\top} f(\mathbf{W}_3 \mathbf{h}'_t + \mathbf{W}_4 \tilde{\mathbf{V}}_{h,w}), \tag{7}$$

where the vector $\boldsymbol{\beta}$ and matrices $\mathbf{W}_3$ and $\mathbf{W}_4$ are learned parameters, and the attention mechanism $a(\cdot)$ approximates the vector $\boldsymbol{\alpha}_t$ of unnormalized attention weights.

The attention weights are distributed over columns of $\tilde{\mathbf{V}}_t$ so that each feature in the column is given identical weight, which is standard for typical character recognition tasks. This approach differs from the attention mechanism used by Deng et al., which places attention over rows and columns, so that attention weights vary for each element of $\tilde{\mathbf{V}}_t$, which may be more appropriate for complex images such as math formulas or tables. While the standard attention of Bahdanau et al. uses the softmax activation for Eq. (6), we experiment with two alternative activations to produce attention weights: sigmoid (i.e., Bernoulli) and linear (i.e., $a_t = e_t$).

Finally, the networks are trained end-to-end to minimize the cross-entropy loss:

$$\mathcal{L} = \sum_{t=1}^{T} -\log p\big(y_{t+1} \mid y_1, \ldots, y_t, \tilde{\mathbf{V}}\big). \tag{8}$$

3 Experimental evaluation

We experiment on two widely-used HTR benchmark datasets, IAM (modern English) and RIMES (modern French), and two historical datasets, Saint Gall (9th c. Latin) and Parzival (13th c. German) [42–45]. The datasets, which are described in Table 1, consist of images of handwritten text lines and their corresponding ground-truth transcriptions.

We follow the image preprocessing steps of Puigcerver et al. [19, 46], which includes binarizing the images in a manner that preserves their original grayscale information [47], rescaling the images, and converting the images to JPEG format. Figure 2 provides an example of a preprocessed image from each benchmark dataset.

3.1 Evaluation

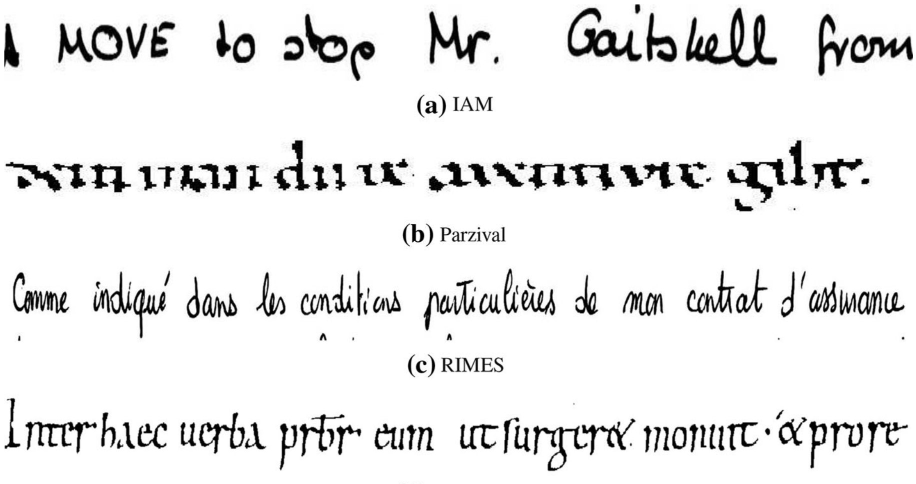
We measure the performance of the attention networks by comparing the estimated transcription $\hat{\mathbf{y}}$ with the ground-truth $\mathbf{y}$. Since the networks are trained on sequences of characters rather than words, we measure the Character Error Rate (CER) instead of the Word Error Rate. The CER is calculated as the edit distance normalized by the number of characters in the ground truth:

Table 1 Training, validation, and test set splits and language characteristics for benchmark datasets

	Lines				Maximum length			Unique Characters		
Dataset	Train	Val.	Test	Total	Train	Val.	Test	Train	Val.	Test
IAM	6161	966	2915	10,042	81	73	95	79	76	75
Parzival	2237	912	1328	4477	70	71	66	57	56	55
RIMES	10,171	1162	778	12,111	100	110	94	97	88	85
Saint Gall	468	235	707	1410	74	69	73	47	46	47

Notes: unique characters include case-sensitive alphanumeric characters, punctuation, and whitespace

Fig. 2 Example preprocessed images from the benchmark datasets



(a) IAM

(b) Parzival

(c) RIMES

(d) Saint Gall

$$\text{CER} = \sum_t \frac{\text{Edit Distance}(y_t, \hat{y_t})}{|y_t|}, \tag{9}$$

where the edit distance (or, Levenshtein distance), is the minimum number of insertions, substitutions, and deletions required to alter the target y_t to the prediction $\hat{y_t}$ at each time-step. We also measure the character perplexity (CPPL) of the character-based conditional language model, which is the exponent of the cross-entropy loss defined in Eq. (8). Language models with smaller perplexity generally perform better in predicting characters given the history, and are thus strongly correlated with the CER [48, 49].

3.2 Implementation details

When training the networks, we fix the image height to 64 pixels while maintaining the aspect ratio, group images with similar widths, and pad with whitespace to facilitate batching. We implement a biased importance sampling scheme to speed up training and decoding [50].

The CNN converts the text line images into a sequence of visual feature vectors. It consists of seven convolutional layers, each followed by a Rectified Linear Unit (ReLU) activation and then a max-pooling layer to reduce the spatial size of the representation. The third, fifth, and seventh layers use batch normalization following the convolution in order to speed up training. Dropout is applied to the output of the seventh convolutional layer in order to prevent overfitting. Table 2 provides further detail on the CNN specifications.

Stacked on the CNN is a single-layer, bidirectional long short-term memory (BLSTM) encoder with 512 hidden units and a two-layer gated recurrent unit (GRU) decoder, each with 256 hidden units. The bidirectional recurrent layers allow the encoder to compute a representation that depends on both past and present characters in the sequence, and row-wise encoding refines the feature representation to include horizontal context. The attentional decoder interprets the feature representation, focusing on the most important columns of re-encoded features.

We train the networks for 200 epochs with a batch size of 8, stochastic gradient descent to learn the parameter weights, and the Adam optimizer to adapt the learning rate. As a regularization strategy, we implement ℓ^2 regularization loss and data augmentation by applying random affine transformations to 20% of the training set images, including scaling, translating, rotating, and shearing. In addition, we employ gradient norm clipping and gradient normalization in order to prevent exploding gradients.

Table 2 CNN specification

Conv				Pool	
# filters	Filter size	Stride size	Bn	Pool size	Stride size
64	(3,3)	(1,1)		(2,2)	(2,2)
128	(3,3)	(1,1)		(2,2)	(2,2)
256	(3,3)	(1,1)	✓	–	–
256	(3,3)	(1,1)		(2,1)	(2,1)
512	(3,3)	(1,1)	✓	–	–
512	(3,3)	(1,1)		(2,1)	(2,1)
512	(2,2)	(1,1)	✓	–	–

Notes: the sizes are ordered (height, width). See notes to Fig. 1

4 Results

We train the attention networks without the assistance of any lexicon or explicit language model and record their performance in terms of CER and CPPL on the validation and test in Table 3. The networks perform comparatively well on the Parzival and Saint Gall datasets, which have fewer training examples, and have shorter lines and vocabularies. The networks perform less well on the IAM and RIMES datasets, which have longer lines, and a larger vocabulary and number of training examples.

Table 4 compares the performance of the (softmax) attention networks on the IAM and RIMES test set with models in the existing literature. The attention networks achieve a CER of 16.6% on the IAM dataset, which outperforms CTC models that encode image features using LSTMs or multidimensional LSTMs (MDLSTMs) [51], but does not approach the current state-of-the-art model of Bluche and Messina [18], which combines convolutional and recurrent layers for encoding with a CTC decoder.

A direct comparison against most of the models in the existing literature is not possible because most of the existing models rely on domain-specific lexicons, and explicit language models for decoding. Bluche [49], for example, uses a word-based lexicon and a word-based language model. The model of Bluche [1], which combines a MDLSTM encoder and a softmax attention-enhanced bidirectional LSTM decoder, inputs and outputs at the character-level, although the decoder output is not conditioned on the previous character. The aforementioned model is also trained with curriculum learning and with a slightly larger training set. The state-of-the-art model of Bluche and Messina [18], in comparison, uses a hybrid word and character-based language model. Gui et al. [29] also train character-aware attention networks, but with a CTC output layer to perform the transcription. Michael et al. [52] is the most comparable to our work because the authors train character-aware attention networks without the use of a language model.

4.1 Comparing attention distributions

In order to gain insight into how the attention mechanism learns alignment between the source and target character, we plot in Fig. 3 a visualization of the source attention distribution for attention networks trained on the IAM dataset. Each row traces the attention weights over the source line at each step of decoding. White values reflect intensity of attention while absence of attention is black.

Softmax attention predicts a character by focusing heavily on single characters, whereas the attention distribution for sigmoid focus on multiple characters at each time-step. Softmax attention is able to learn a linear alignment whereas the alignment generated by sigmoid attention is linear and less precise.[1] When a linear function is used to obtain the attention weights, the model predicts a character by looking at the entire sequence of characters, and there is no clear structure in the alignment.

In order to determine how the model makes mistakes, we visualize attention on the input image drawn from the IAM dataset. For example, the model tends to produce errors when characters are skewed (Fig. 4b), have long tails (Fig. 4a, c), or written in uppercase cursive (Fig. 4d). Figure 5, which provides examples of correct IAM transcriptions and visualized softmax attention, shows that the model can correctly predict illegible handwriting (Fig. 5b) because it leverages information from the entire input sequence.

5 Conclusion and future directions

The paper approaches the task of handwritten text transcription with attention-based encoder–decoder networks trained to handle sequences of characters rather than words. The attention networks are domain and language-agnostic because they are trained without the aid of a lexicon or explicit language model.

We train the model on lines of text from a popular handwriting dataset and experiment with different activation functions for the attention mechanism. Our results show that softmax attention focuses heavily on individual characters, while sigmoid attention focuses on multiple characters at each step of the decoding. When the sequence alignment is one-to-one, softmax attention is able to learn a more precise alignment at each step of the decoding, whereas the alignment generated by sigmoid attention is

[1] Similarly, Kim et al. [71] find that softmax attention performs better than sigmoid attention on word-to-word machine translation tasks.

Table 3 Attention networks: evaluation metrics on benchmark datasets

Dataset	Val.		Test	
	CER (%)	CPPL	CER (%)	CPPL
IAM	14.3	71,075.2	16.6	exp(36.5)
Parzival	4.6	12.0	4.7	52.6
RIMES	11.1	811.9	12.1	92.4
Saint Gall	14.3	24.5	12.7	17,164.4

Notes: networks trained with softmax attention

much less precise. When the model has linear attention, the model predicts a character by looking at the entire sequence of characters and performs poorly because it lacks a precise alignment between the source and text output.

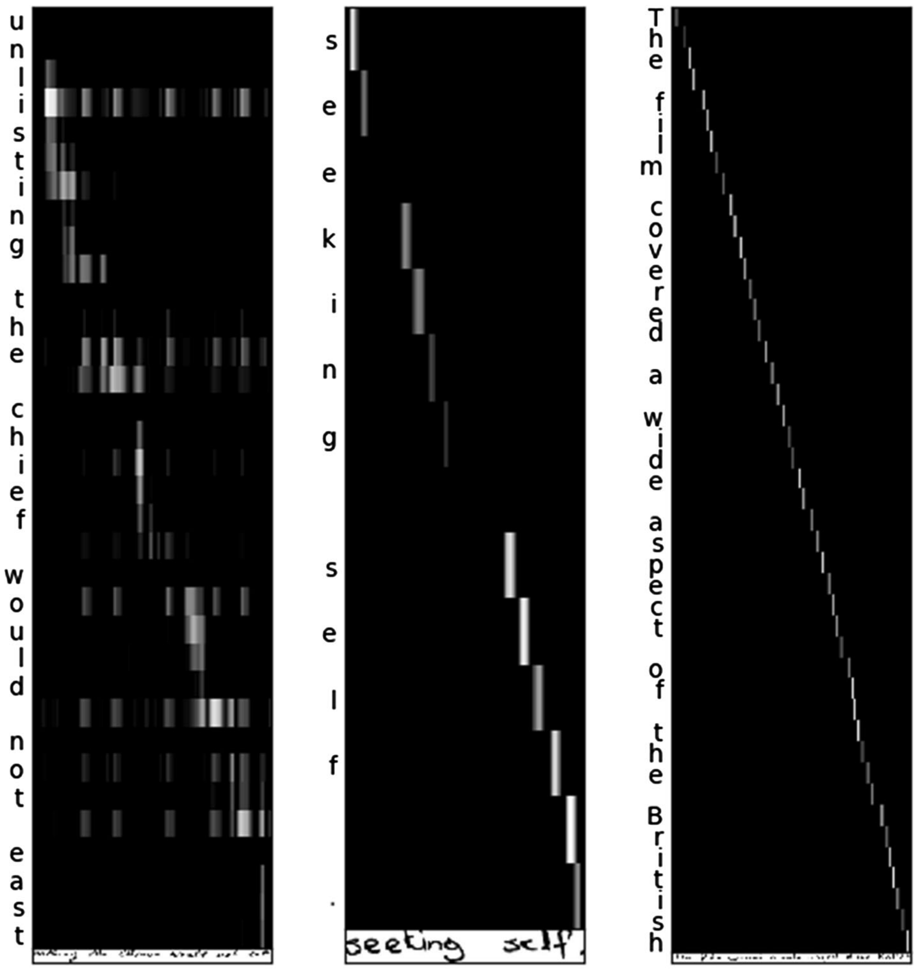
Our primary contributions are applying character-aware attention networks to the task of handwritten text line transcription and also comparing attention configurations for the decoder. Future work might apply attention networks to the problem of HTR in natural scene images [72]. Previous literature has focused on recognizing printed text in natural scene images using standard methods in

Table 4 Benchmark comparison: test set CER on IAM and RIMES datasets

Model	Source	LM	CB	IAM CER (%)	RIMES CER (%)
CNN + BLSTM + CTC	[18]	✓		3.2	1.9
MDLSTM + CTC	[53]	✓		3.5	2.8
MDLSTM + MLP/HMM	[54]	✓		3.6	–
MDLSTM + CTC	[49]	✓		4.4	3.5
CNN + LSTM + CTC	[19]	✓		4.4	2.3
MDLSTM + Attention	[55]	✓		4.4	3.5
Transformer	[40]			4.6	–
LSTM + HMM	[56]	✓		4.7	4.3
LSTM + HMM	[57]	✓		4.8	4.3
CNN + LSTM + Attention	[52]	✓	✓	4.8	–
CNN + CTC	[37]		✓	4.9	–
CNN + LSTM + Attention	[58]		✓	4.9	–
LSTM + HMM	[59]	✓		5.1	4.6
MDLSTM + CTC	[60]	✓		5.1	3.3
CNN + BLSTM + Attention + CTC	[29]			5.1	–
CNN + BLSTM	[61]			5.7	5.0
CNN + BGRU + GRU + Attention	[22]	✓		5.7	2.6
CNN + CTC	[62]			6.1	3.4
MDLSTM + CTC	[1]	✓		6.6	–
CNN + BGRU + GRU	[28]			6.8	–
CNN + BLSTM + LSTM	[20]			8.1	3.5
GMM/HMM	[63]	✓		8.2	–
CNN + LSTM + Attention	[27]			8.8	–
CNN + LSTM + CTC	[64]			9.7	–
MLP/HMM	[65]	✓		9.8	–
MDLSTM + CTC	[66]	✓		11.1	8.29
MLP/HMM	[67]	✓		12.4	–
CNN + BLSTM + GRU + Attention	Ours		✓	16.6	12.1
MDLSTM + CTC	[2]			17.0	–
BLSTM + CTC	[6]	✓		18.2	–
CNN + LSTM + Attention	[58]		✓	–	3.1
CNN + BLSTM + Attention	[68]	✓		–	5.8
HMM/MLP	[69]	✓		–	7.2
BLSTM + CTC	[70]			–	7.6

Notes: ‘BGRU’: bi-directional GRU; ‘CB’: model is character-based; ‘GMM’: Gaussian mixture model; ‘LM’: explicit language model used for decoding; ‘MLP’: multilayer perceptron

Fig. 3 Visualization of the source attention distribution over the input image (horizontal axis). The vertical axis is the transcription. Each row traces the attention weights over the source line at each step of decoding, in grayscale (0: black, 1: white)

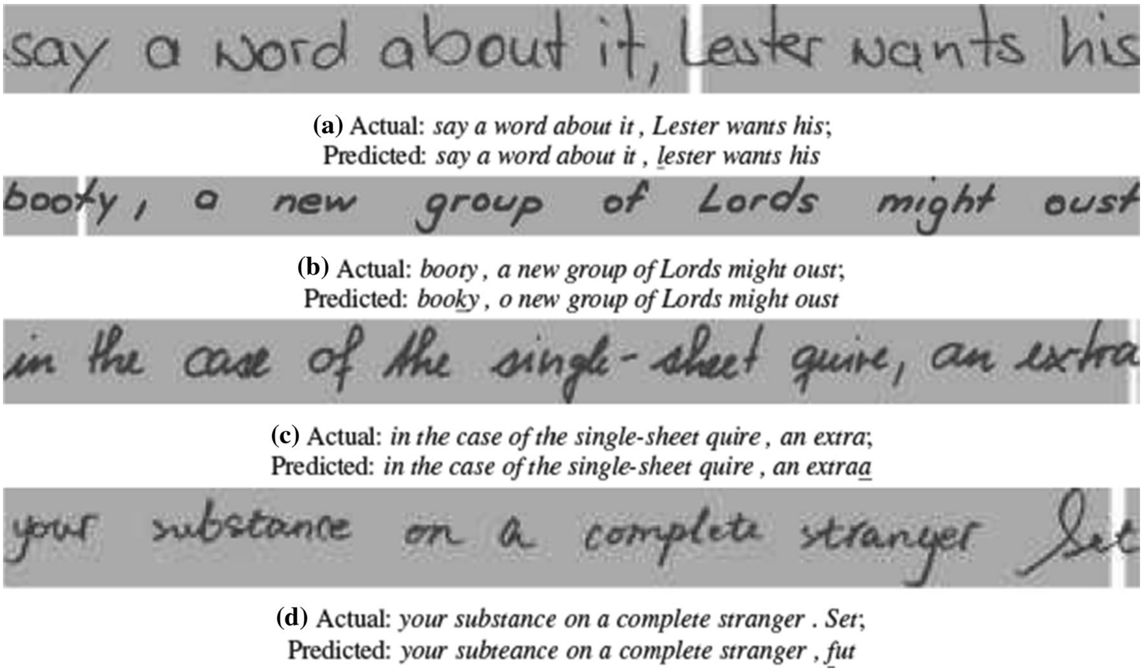


(a) Linear

(b) Sigmoid

(c) Softmax

computer vision for segmentation [73]. The attention networks used in this paper are capable of transcribing handwritten text without the need for producing segmentations or bounding boxes of text in images, so the model can potentially transcribe handwritten text in natural scene images without preprocessing.



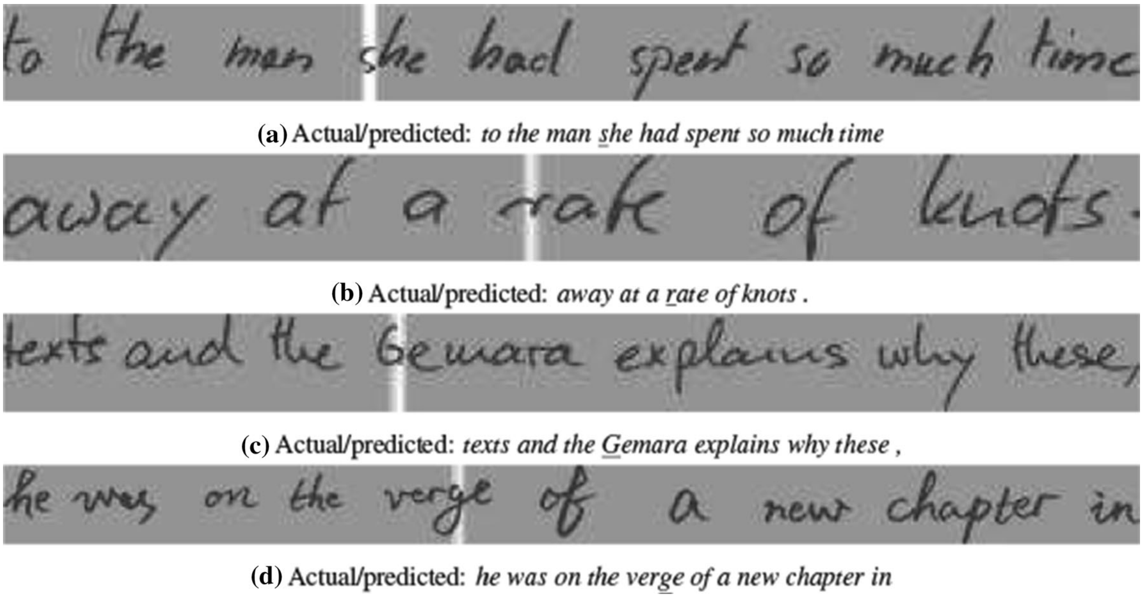
(a) Actual: *say a word about it , Lester wants his*;
Predicted: *say a word about it , lester wants his*

(b) Actual: *booty , a new group of Lords might oust*;
Predicted: *booky , o new group of Lords might oust*

(c) Actual: *in the case of the single-sheet quire , an extra*;
Predicted: *in the case of the single-sheet quire , an extraa*

(d) Actual: *your substance on a complete stranger . Set*;
Predicted: *your subteance on a complete stranger , fut*

Fig. 4 Incorrect IAM transcriptions and visualized softmax attention. White lines indicates the attended regions and underlines in the transcription indicate the corresponding character



(a) Actual/predicted: *to the man she had spent so much time*

(b) Actual/predicted: *away at a rate of knots .*

(c) Actual/predicted: *texts and the Gemara explains why these ,*

(d) Actual/predicted: *he was on the verge of a new chapter in*

Fig. 5 Correct IAM transcriptions and visualized softmax attention. See footnotes to Fig. 4

Acknowledgements A shorter version of this work appeared as an arXiv preprint at https://arxiv.org/abs/1712.04046.

Data Availability Statement The IAM, Saint Gall, and Parzival datasets can be downloaded from: https://fki.tic.heia-fr.ch/databases. The RIMES dataset can be downloaded from: http://www.a2ialab.com/doku.php?id=rimes_database:start.

Declarations

Funding Poulos acknowledges support of the National Science Foundation Graduate Research Fellowship under Grant DGE-1106400, and the National Science Foundation under Grant DMS-1638521 to the Statistical and Applied Mathematical Sciences Institute. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

Code availability Implementation code is available at the repository: https://github.com/jvpoulos/Attention-OCR/.

Conflict of interest The authors declare no conflicts of interest.

References

1. Bluche T, Louradour J, Messina R (2016) Scan, attend and read: end-to-end handwritten paragraph recognition with MDLSTM attention. ArXiv e-prints 1604:03286
2. Louradour J, Kermorvant C (2013) Curriculum learning for handwritten text line recognition. ArXiv e-prints 1312:1737
3. Graves A, Fernández S, Gomez F, Schmidhuber J (2006) Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In: Proceedings of the 23rd international conference on Machine learning, pp 369–376
4. Graves A, Liwicki M, Fernández S, Bertolami R, Bunke H, Schmidhuber J (2009) A novel connectionist dystem for unconstrained handwriting recognition. IEEE 31:855–868
5. Liwicki M, Graves A, Bunke H, Schmidhuber J (2007) A novel approach to on-line handwriting recognition based on bidirectional long short-term memory networks. In: Proceedings of the 9th International conference on document analysis and recognition, vol 1, pp 367–371
6. Liwicki M, Graves A, Bunke H (2012) Neural networks for handwriting recognition. Computational intelligence paradigms in advanced pattern classification. Springer, Berlin, pp 5–24
7. Wigington C, Stewart S, Davis B, Barrett B, Price B, Cohen S (2017) Data augmentation for recognition of handwritten words and lines using a CNN-LSTM network. In: 2017 14th IAPR International conference on document analysis and recognition (ICDAR), IEEE, vol 1, pp 639–645
8. Stuner B, Chatelain C, Paquet T (2020) Handwriting recognition using cohort of lstm and lexicon verification with extremely large lexicon. Multim Tools Appl 79(45):34407–34427
9. Deng Y, Kanervisto A, Ling J, Rush AM (2016) Image-to-Markup Generation with Coarse-to-Fine Attention. ArXiv e-prints 1609:04938
10. Vinyals O, Kaiser L, Koo T, Petrov S, Sutskever I, Hinton G (2014) Grammar as a Foreign Language. ArXiv e-prints 1412:7449
11. Bahdanau D, Cho K, Bengio Y (2014) Neural machine translation by jointly learning to align and translate. ArXiv e-prints 1409:0473
12. Chorowski JK, Bahdanau D, Serdyuk D, Cho K, Bengio Y (2015) Attention-based models for speech recognition. In: Advances in neural information processing systems, pp 577–585
13. Xu K, Ba J, Kiros R, Cho K, Courville AC, Salakhutdinov R, Zemel RS, Bengio Y (2015) Show, attend and tell: neural image caption generation with visual attention. ICML 14:77–81
14. Cho K, van Merrienboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y (2014) Learning phrase representations using RNN encoder-decoder for statistical machine translation. ArXiv e-prints 1406:1078
15. Cho K, Courville A, Bengio Y (2015) Describing multimedia content using attention-based encoder-decoder networks. ArXiv e-prints 1507:01053
16. Lee CY, Osindero S (2016) Recursive recurrent nets with attention modeling for OCR in the wild. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2231–2239
17. Shi B, Wang X, Lyu P, Yao C, Bai X (2016) Robust scene text recognition with automatic rectification. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4168–4176
18. Bluche T, Messina R (2017) Gated convolutional recurrent neural networks for multilingual handwriting recognition. In: Proceedings of the 13th International conference on document analysis and recognition (ICDAR), Kyoto, Japan, pp 13–15
19. Puigcerver J (2017) Are multidimensional recurrent layers really necessary for handwritten text recognition? In: Document analysis and recognition (ICDAR), 2017 14th IAPR international conference on, IEEE, vol 1, pp 67–72
20. Chowdhury A, Vig L (2018) An efficient end-to-end neural model for handwritten text recognition. ArXiv e-prints 1807.07965
21. Zhang Y, Nie S, Liu W, Xu X, Zhang D, Shen HT (2019) Sequence-to-sequence domain adaptation network for robust text image recognition. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)
22. Kang L, Riba P, Villegas M, Fornés A, Rusiñol M (2019) Candidate fusion: Integrating language modelling into a sequence-to-sequence handwritten word recognition architecture. ArXiv e-prints 1912.10308
23. Kang L, Rusiñol M, Fornés A, Riba P, Villegas M (2020) Unsupervised writer adaptation for synthetic-to-real handwritten word recognition. In: The IEEE winter conference on applications of computer vision, pp 3502–3511
24. Xiao S, Peng L, Yan R, Wang S (2020) Deep network with pixel-level rectification and robust training for handwriting recognition. SN Comput Sci 1(3):1–13
25. Retsinas G, Sfikas G, Maragos P (2020) Wsrnet: Joint spotting and recognition of handwritten words. ArXiv e-prints 1604:032860
26. Belay B, Habtegebrial T, Belay G, Mesheshsa M, Liwicki M, Stricker D (2020) Learning by injection: Attention embedded recurrent neural network for amharic text-image recognition. 1604:032861
27. Sueiras J, Ruiz V, Sanchez A, Velez JF (2018) Offline continuous handwriting recognition using sequence to sequence neural networks. Neurocomputing
28. Kang L, Toledo JI, Riba P, Villegas M, Fornés A, Rusinol M (2018) Convolve, attend and spell: An attention-based sequence-to-sequence model for handwritten word recognition. In: German Conference on Pattern Recognition. pp 459–472. Springer, Berlin
29. Gui L, Liang X, Chang X, Hauptmann AG (2018) Adaptive context-aware reinforced agent for handwritten text recognition. In: Proceedings of the British machine vision conference (BMVC)

30. Gehring J, Auli M, Grangier D, Yarats D, Dauphin YN (2017) Convolutional sequence to sequence learning. ArXiv e-prints 1604:032862
31. Fogel S, Averbuch-Elor H, Cohen S, Mazor S, Litman R (2020) Scrabblegan: Semi-supervised varying length handwritten text generation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)
32. Davis B, Tensmeyer C, Price B, Wigington C, Morse B, Jain R (2020) Text and style conditioned GAN for generation of offline handwriting lines. ArXiv e-prints 1604:032863
33. Poznanski A, Wolf L (2016) CNN-n-gram for handwriting word recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2305–2314
34. Such FP, Peri D, Brockler F, Paul H, Ptucha R (2018) Fully convolutional networks for handwriting recognition. In: 2018 16th International conference on frontiers in handwriting recognition (ICFHR), IEEE, pp 86–91
35. Coquenet D, Soullard Y, Chatelain C, Paquet T (2019) Have convolutions already made recurrence obsolete for unconstrained handwritten text recognition? In: 2019 International conference on document analysis and recognition workshops (ICDARW), IEEE, vol 5, pp 65–70
36. Ptucha R, Such FP, Pillai S, Brockler F, Singh V, Hutkowski P (2019) Intelligent character recognition using fully convolutional neural networks. Pattern Recogn 88:604–613
37. Yousef M, Hussain KF, Mohammed US (2020) Accurate, data-efficient, yunconstrained text recognition with convolutional neural networks. Pattern Recogn 108:107482
38. Yousef M, Bishop TE (2020) Origaminet: Weakly-supervised, segmentation-free, one-step, full page text recognition by learning to unfold. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)
39. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. Adv Neural Inf Process Syst 30:5998–6008
40. Kang L, Riba P, Rusiñol M, Fornés A, Villegas M (2020) Pay attention to what you read: Non-recurrent handwritten text-line recognition. ArXiv e-prints 1604:032864
41. Ling W, Trancoso I, Dyer C, Black AW (2015) Character-based neural machine translation. ArXiv e-prints 1604:032865
42. Marti UV, Bunke H (2002) The IAM-database: an english sentence database for offline handwriting recognition. Int J Doc Anal Recogn 5(1):39–46
43. Grosicki E, El-Abed H (2011) ICDAR 2011: French handwriting recognition competition. In: Proceedings of the international conference on document analysis and recognition, pp 1459–1463
44. Fischer A, Frinken V, Fornés A, Bunke H (2011) Transcription alignment of Latin manuscripts using Hidden Markov Models. In: Proceedings of the 2011 workshop on historical document imaging and processing, ACM, pp 29–36
45. Fischer A, Wuthrich M, Liwicki M, Frinken V, Bunke H, Viehhauser G, Stolz M (2009) Automatic transcription of handwritten medieval documents. In: 2009 15th international conference on virtual systems and multimedia, IEEE, pp 137–142
46. Puigcerver J, Martin-Albo D, Villegas M (2016) Laia: A deep learning toolkit for HTR. 1604:032866, gitHub repository
47. Villegas M, Romero V, Sánchez JA (2015) On the modification of binarization algorithms to retain grayscale information for handwritten text recognition. In: Iberian conference on pattern recognition and image analysis. pp 208–215, Springer, Berlin
48. Wang P, Sun R, Zhao H, Yu K (2013) A new word language model evaluation metric for character based languages. In: Chinese computational linguistics and natural language processing based on naturally annotated big data. pp 315–324. Springer, Berlin
49. Bluche T (2015) Deep neural networks for large vocabulary handwritten text recognition. PhD thesis, Université Paris Sud-Paris XI
50. Jean S, Cho K, Memisevic R, Bengio Y (2014) On Using Very Large Target Vocabulary for Neural Machine Translation. ArXiv e-prints 1604:032867
51. Graves A, Schmidhuber J (2009) Offline handwriting recognition with multidimensional recurrent neural networks. In: Advances in neural information processing systems, pp 545–552
52. Michael J, Labahn R, Grüning T, Zöllner J (2019) Evaluating sequence-to-sequence models for handwritten text recognition. In: 2019 International conference on document analysis and recognition (ICDAR), IEEE, pp 1286–1293
53. Voigtlaender P, Doetsch P, Ney H (2016) Handwriting recognition with large multidimensional long short-term memory recurrent neural networks. In: Frontiers in handwriting recognition (ICFHR), 2016 15th international conference on, IEEE, pp 228–233
54. Castro D, Bezerra BL, Valença M (2018) Boosting the deep multidimensional long-short-term memory network for handwritten recognition systems. In: 2018 16th International conference on frontiers in handwriting recognition (ICFHR), IEEE, pp 127–132
55. Bluche T (2016) Joint line segmentation and transcription for end-to-end handwritten paragraph recognition. In: Advances in neural information processing systems, pp 838–846
56. Doetsch P, Kozielski M, Ney H (2014) Fast and robust training of recurrent neural networks for offline handwriting recognition. In: Frontiers in handwriting recognition (ICFHR), 2014 14th international conference on, IEEE, pp 279–284
57. Voigtlaender P, Doetsch P, Wiesler S, Schlüter R, Ney H (2015) Sequence-discriminative training of recurrent neural networks. In: Acoustics, speech and signal processing (ICASSP), 2015 IEEE International Conference on, IEEE, pp 2100–2104
58. Coquenet D, Chatelain C, Paquet T (2020) End-to-end handwritten paragraph text recognition using a vertical attention network. ArXiv e-prints 1604:032868
59. Kozielski M, Doetsch P, Ney H (2013) Improvements in RWTH's system for off-line handwriting recognition. In: 2013 12th International conference on document analysis and recognition, IEEE, pp 935–939
60. Pham V, Bluche T, Kermorvant C, Louradour J (2014) Dropout improves recurrent neural networks for handwriting recognition. In: Frontiers in handwriting recognition (ICFHR), 2014 14th international conference on, IEEE, pp 285–290
61. Dutta K, Krishnan P, Mathew M, Jawahar C (2018) Improving CNN-RNN hybrid networks for handwriting recognition. In: 2018 16th International conference on frontiers in handwriting recognition (ICFHR), IEEE, pp 80–85
62. Huang X, Qiao L, Yu W, Li J, Ma Y (2020) End-to-end sequence labeling via convolutional recurrent neural network with a connectionist temporal classification layer. Int J Comput Intell Syst 13:341–351. 1604:032869
63. Kozielski M, Rybach D, Hahn S, Schlüter R, Ney H (2013) Open vocabulary handwriting recognition using combined word-level and character-level language models. In: Acoustics, speech and signal processing (ICASSP), 2013 IEEE international conference on, IEEE, pp 8257–8261
64. Krishnan P, Dutta K, Jawahar C (2018) Word spotting and recognition using deep embedding. In: 2018 13th IAPR international workshop on document analysis systems (DAS), IEEE, pp 1–6
65. España-Boquera S, Castro-Bleda MJ, Gorbe-Moya J, Zamora-Martinez F (2011) Improving offline handwritten text recognition with hybrid hmm/ann models. IEEE Trans Pattern Anal Mach Intell 33(4):767–779. 1312:17370

66. Chen Z, Wu Y, Yin F, Liu CL (2017) Simultaneous script identification and handwriting recognition via multi-task learning of recurrent neural networks. In: 2017 14th IAPR international conference on document analysis and recognition (ICDAR), IEEE, vol 1, pp 525–530
67. Dreuw P, Doetsch P, Plahl C, Ney H (2011) Hierarchical hybrid MLP/HMM or rather MLP features for a discriminatively trained Gaussian HMM: a comparison for offline handwriting recognition. In: 2011 18th IEEE international conference on image processing, IEEE, pp 3541–3544
68. Doetsch P, Zeyer A, Ney H (2016) Bidirectional decoder networks for attention-based end-to-end offline handwriting recognition. In: 2016 15th international conference on frontiers in handwriting recognition (ICFHR), IEEE, pp 361–366
69. Menasri F, Louradour J, Bianne-Bernard AL, Kermorvant C (2012) The a2ia french handwriting recognition system at the rimes-icdar2011 competition. In: Document recognition and retrieval XIX, international society for optics and photonics, vol 8297, p 82970Y
70. Soullard Y, Ruffino C, Paquet T (2019) CTCModel: a Keras model for connectionist temporal classification. ArXiv e-prints 1312:17371
71. Kim Y, Denton C, Hoang L, Rush AM (2017) Structured attention networks. ArXiv e-prints 1312:17372
72. Veit A, Matera T, Neumann L, Matas J, Belongie S (2016) COCO-Text: dataset and benchmark for text detection and recognition in natural images. ArXiv e-prints 1312:17373
73. Jaderberg M, Simonyan K, Vedaldi A, Zisserman A (2016) Reading text in the wild with convolutional neural networks. Int J Comput Vision 116(1):1–20

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.