SCIENCE CHINA
Information Sciences

• **Supplementary File** •

Robust Encoder-Decoder Learning Framework towards Offline Handwritten Mathematical Expression Recognition Based on Multi-Scale Deep Neural Network

Guangcun SHAN[1*], Hongyu WANG[1], Wei LIANG[2,3] & Kai CHEN[4]

[1]*School of Instrumentation Science and Optoelectronics Engineering, Beihang University, Beijing* 100191*, China;*
[2]*Institute of Electronics, Chinese Academy of Sciences, Beijing* 100190*, China;*
[3]*University of Chinese Academy of Sciences, Beijing* 100049*, China;*
[4]*Boheng Technology (Hangzhou) Co. Ltd, Hangzhou* 310016*, China*

Appendix A Recognition of Mathematical Expressions

For the recognition of mathematical expressions, the traditional method usually divides the mathematical expressions first, then synthesizes the information of each part, and finally obtains the recognition results. For example, Fateman [1,2] divides mathematical expressions into 25 parts, and identifies them by processing their information. Lee[3,4] divides the mathematical expression into 16 parts, each part obtains the features from four directions, and then synthesizes these features.However, the effect of mathematical expression recognition by traditional methods is not good. The reason is that dividing mathematical expressions often fails to pay attention to the characteristics of the whole. The method of synthesizing whole features from local features will bring about great differences due to the different writing of mathematical expressions, which can easily lead to misjudgments. In addition, this method can not take advantage of the context information in mathematical expressions. Therefore, we use convolutional neural network(CNN) plus recurrent neural network(RNN) structure to deal with these problems. Because CNN can extract not only the local information of mathematical expressions, but also the whole information. At the same time, RNN can make better use of the context information in mathematical expressions to predict the results. So, CNN plus RNN is more conducive to the recognition of mathematical expressions with complex two-dimensional structure.

Appendix B Dataset

The dataset used in this work is the CROHME 2014 dataset[5], which is the largest dataset in handwritten mathematical expressions. It is a public dataset composed of 8836 math expressions in training data and 986 math expressions in testing data. Meanwhile, there are 110 different math symbols including numbers, almost all common operators and two start and stop symbols <sos>, <eol>. Generally, the size of pictures in CROHME 2014 is highly uneven, which will make the recognition very difficult. In the train and test datasets, the biggest picture size is almost 400,000 pixels and the smallest picture size is only 1,400 pixels.

Appendix C WER Loss

The WER loss is used to evaluate the prediction performance of neural networks [6,7]. WER loss is the ratio of edit distance between the prediction and the truth to the length of the truth. Edit distance is the smallest edit operation from one string to another. Allowed editing operations include: insert a character, delete a character, replace a character. An example of the edit distance is shown in Table C1.

* Corresponding author (email: gcshan@buaa.edu.cn)

Table C1 An Example of Edit Distance

Edit operation	Example
Insert	Prediction: ab=1 Truth: a+b=1
Delete	Prediction: a++b=1 Truth: a+b=1
Replace	Prediction: a+c=1 Truth: a+b=1

If our prediction sequence is A, the real sequence is B. From A to B, a minimum of W_1 insertions, W_2 deletions, and W_3 replacements are required. Also know that the length of sequence B is W_4, then the WER loss of this prediction is:

$$WER = \frac{W_1 + W_2 + W_3}{W_4} \quad . \tag{C1}$$

According to the equation C1, we can evaluate the quality of a model.

Appendix D Experiments Results

In our work, we propose a new multi-scale CNN architecture which enhances the Densenet. This new architecture includes not only the high-level but also the low-level features in CNN and the reason for using this structure is to make full use of the information extracted from the input image by CNN. This network system is publicly available in github[8]. In the experiment, we tested the predictive effect of the model under different learning rates and whether Teacher-forcing was used or not. The Teacher-forcing is a method which basic idea is using the truth values instead of the predictive values as the next input of RNN. If we do not use the Teacher-forcing, each step of the input to RNN will be the predicted values. If the Teacher-forcing is used, the input for each step of RNN will be the truth values.The test results of different learning rate and whether Teacher-forcing was used are shown in Table D1 and Table D2.

Table D1 The experiment in different Learning rate

Learning rate	WER loss	ExpRate
0.00009	25.715%	28.216%
0.00010	28.089%	26.162%

Table D2 The experiment in Teacher-forcing

Teacher-forcing	WER loss	ExpRate
use	25.715%	28.216%
not use	28.619%	21.730%

From Table D1, it can be seen that the model using 0.00009 as learning rate can get better results than using 0.00010. Meanwhile, from Table D2, it can be seen that the model using Teacher-forcing has a positive effect. So, the best testing results for the experiments performed here were obtained by using Teacher-forcing at a learning rate of 0.00009, with a WER error of 25.715% and an ExpRate of 28.216%.

Appendix E Experiments of Coverage Model

In order to better show the effectiveness of coverage model, we compared the result of the model with and without coverage. The experiments were completed at a learning rate of 0.00009 and using the Teacher-forcing. From Figure E1, it can be found that after using the coverage, the model correctly predicted the results. But, the model repeatedly outputs a "$/theta$" if we do not use the coverage.

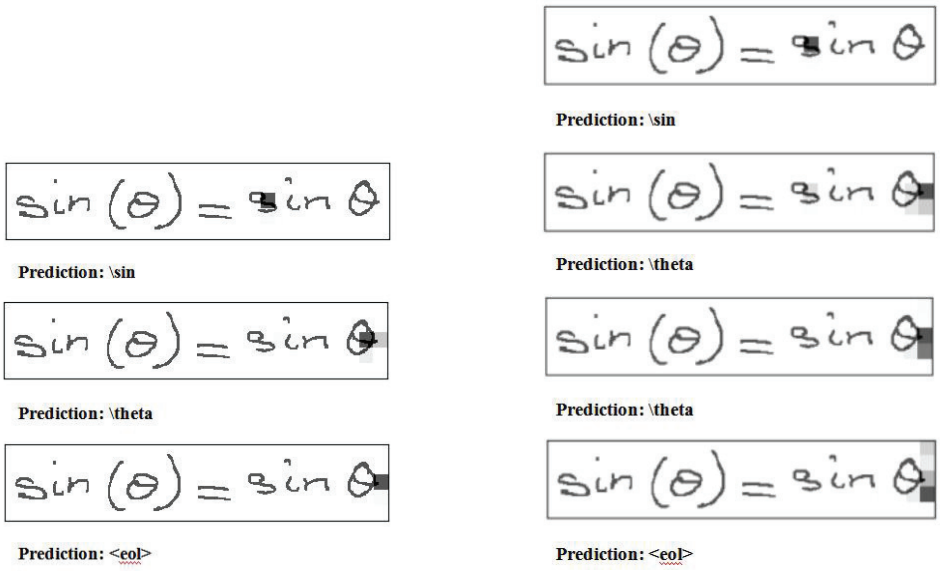

Figure E1 The forecast result: the left is with coverage, the right is without coverage

Appendix F Visualization of Experiment Results

In order to demonstrate the entire mathematical expression recognition process, the output of the model is visualized. The first convolution layers output, and the output of each step in recurrent neural network were visualized.

Appendix F.1 Input Image

In this work, there are 912 pictures in testing dataset and we randomly selected an input picture shown in Figure F1.

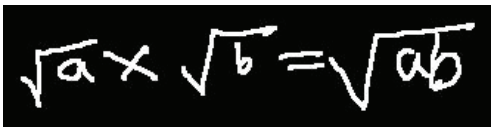

Figure F1 Input image

Appendix F.2 The first convolution layers output

The first layer of the convolutional neural network used in this paper has 48 convolution kernels. Each convolution kernel corresponds to the advanced features of different parts of the original image. Therefore, the output of the first convolutional neural network has 48 different feature maps. Figure F2 randomly show four of the 48 features.

Figure F2 The First Convolution Layers Output

Appendix F.3 Output of each step in RNN

After the convolutional neural network extracts the advanced feature maps of pictures, this information will be passed to the RNN. The output of RNN will continue until the network predicts the end symbol <eol>. Figure F3 shows the step-by-step output of recurrent neural network.

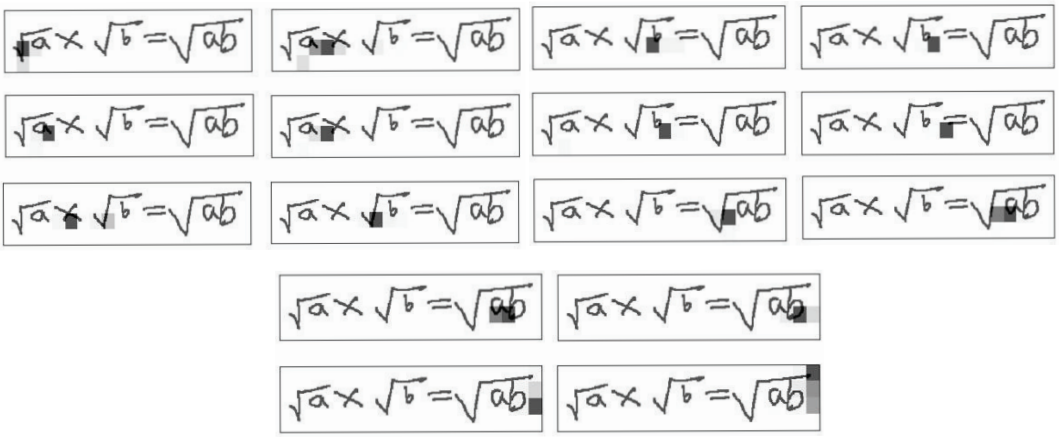

Figure F3 step-by-step output of RNN

In Figure F3, the black area is the result of attention model and represents the area of focus in this step. The prediction LaTeX result in Figure F3 is " \sqrt, {, a, }, \times, \sqrt, {, b, }, =, \sqrt, {, a, b, }, <eol> " which means $\sqrt{a} \times \sqrt{b} = \sqrt{ab}$ and the truth of this picture is also $\sqrt{a} \times \sqrt{b} = \sqrt{ab}$. As a result, the model correctly predicts the result pretty well.

References

1 Fateman R J , Tokuyasu T , Berman B P , et al. Optical Character Recognition and Parsing of Typeset Mathematics1. Journal of Visual Communication & Image Representation, 1996, 7(1):2-15.

2 Berman B P , Fateman R J . Optical Character Recognition for Typeset Mathematics. International Symposium on Symbolic & Algebraic Computation. ACM, 1994.

3 Lee H J , Lee M C . Understanding mathematical expressions using procedure-oriented transformation. Pattern Recognition, 1994, 27(3):447-457.

4 Lee H J , Wang J S . Design of a mathematical expression recognition system. Pattern Recognition Letters, 1997, 18(3):1084-1087 vol.2.

5 CROHME: Competition on Recognition of Online Handwritten Mathematical Expressions. https://www.isical.ac.in/ crohme/

6 Hu, Y, Peng L.R, Tang Y.J. Online handwritten mathematical expression recognition method based on statistical and semantic analysis. 11th IAPR International Workshop on Document Analysis Systems (DAS), 2014, 1:171-175

7 Zhang J, Du J, Zhang S, et al. Watch, attend and parse: An end-to-end neural network-based approach to handwritten mathematical expression recognition. Pattern Recognition, 2017. 71:196-206

8 https://github.com/whywhs/Pytorch-Handwritten-Mathematical-Expression-Recognition for further details and codes.