International Journal of Engineering Applied Sciences and Technology, 2022
Vol. 6, Issue 10, ISSN No. 2455-2143, Pages 146-149
Published Online February 2022 in IJEAST (http://www.ijeast.com)

HANDWRITTEN MATHEMATICAL EQUATION SOLVER

Rajwardhan Shinde, Onkar Dherange, Rahul Gavhane, Hemant Koul, Nilam Patil
Department of CSE
DYP College of Engineering, Akurdi, Pune.

***Abstract*— With recent developments in Artificial intelligence and deep learning every major field which is using computers for any type of work is trying to ease the work using deep learning methods. Deep learning is used in a wide range of fields due to its diverse range of applications like health, sports, robotics, education, etc. In deep learning, a Convolutional neural network (CNN) is being used in image classification, pattern recognition, Text classification, face recognition, live monitoring systems, handwriting recognition, Digit recognition, etc. In this paper, we propose a system for educational use where the recognition and solving process of mathematical equations will be done by machine. In this system for recognition of equations, we use a Convolutional neural network (CNN) model.**

The proposed system can recognize and solve mathematical equations with basic operations (-,+,/,*) of multiple digits as well as polynomial equations. The model is trained with Modified National Institute of Standards and Technology (MNIST) dataset as well as a manually prepared dataset of operator symbols ("-","+", "/", "*", "(", ")"). Further, the system uses the RNN model to solve the recognized operations.

***Keywords*— Convolutional neural network, Recurrent neural network, Digit recognition, Handwritten equation, MNIST, EMNIST**

I. INTRODUCTION

Mathematical equation recognition is one of the challenging tasks in the field of computer vision. A dense layer in a neural network is connected deeply and receives input from all neurons in the previous layer. It is the most commonly used layer in models. In single digit or symbol recognition, a dense layer can give a test score of 99% when trained on 70,000 numbers between 0-9 which is the MNIST dataset along with 49,000 samples of operators which are obtained by populating 4,900 actual operators dataset to eliminate the imbalance between MNIST and operators dataset which include addition, subtraction, multiplication, and division. Recurrent Neural Network (RNN) is a neural network in which the result from the previous step is given as an input source for the current step. It is generally used when we need to remember the previous information to solve the current problem. RNN is able to achieve this with the help of a hidden layer. It is heavily used in speech recognition, video tagging, generating image descriptions, but it also gives a satisfying performance with a test accuracy of 96% and an error rate of 0.13% when trained on 8,00,000 data points while predicting the solution for simple mathematical operations.

In recent times, more and more researchers have tried to tackle the problem of recognizing complex mathematical equations. Convolution Neural Networks (CNN) are mostly used in such kinds of problems which include face recognition, recommender systems, search engines. A convolution is how the input is modified by a filter. In convolutional networks, multiple filters are taken to slice through the image and map them one by one, and learn different portions of an input image. CNN, when used to recognize equations of any degree (linear, quadratic) can achieve a test accuracy of 85% when trained on EMNIST dataset is a set of handwritten character digits derived from the NIST Special Database 19 and converted to a 28x28 pixel image format and dataset structure that directly matches the MNIST dataset.

II. ARCHITECTURE

A. Simple Equation Solver

This module comprises two models: Dense and RNN.

Dense layers are used to extract features from input images with relu as an activation function for the hidden layers and softmax for the final layer for predicting numbers and operators is a multi-class classification problem. This model is trained with an adam optimizer which is reported to have a faster convergence rate than normal stochastic gradient descent (SGD) with momentum.

First hidden layer has 20 neurons densely connected with the second layer 20 neurons and the final output layer has 17 neurons equal to the number of classes in the dataset.

International Journal of Engineering Applied Sciences and Technology, 2022
Vol. 6, Issue 10, ISSN No. 2455-2143, Pages 146-149
Published Online February 2022 in IJEAST (http://www.ijeast.com)

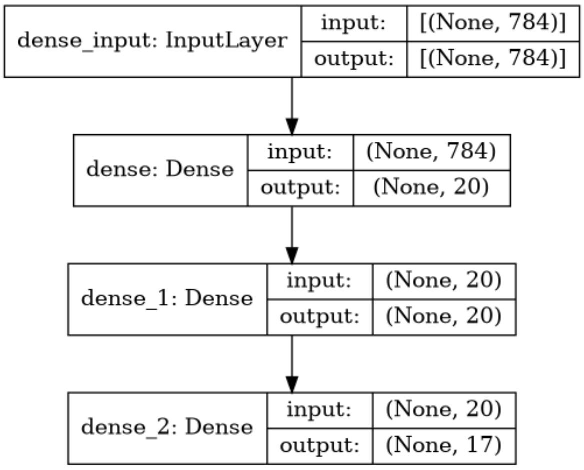

Fig 1. Dense Model Summary

The RNN model is used to predict the result of the current operation based on the previous results. The first layer is a simple RNN layer with 1024 units followed by a repeat vector to repeat the input as that would increase the overall probability of RNN to remember the input. Then it has a second simple RNN layer with 1024 units and for the last layer, it has Time Distributed which applies a Dense layer with 15 units (equal to the number of classes - " 0123456789.+*-/ ") at every timestep.
The model is compiled using adam optimizer and categorical cross-entropy is used as a loss function.
Early stopping along with the model checkpoint is used to save the model with the lowest loss value.

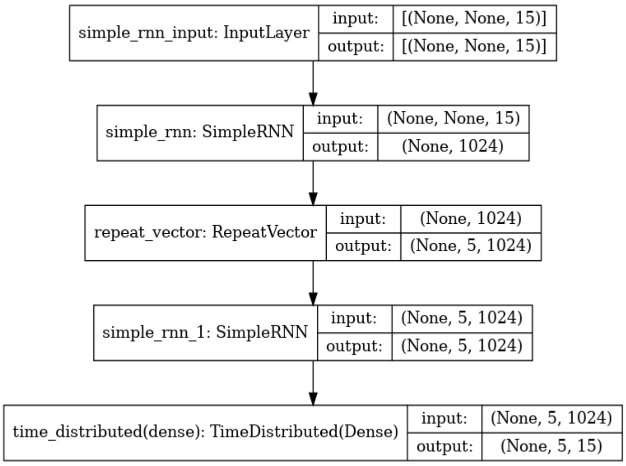

Fig 2. RNN Model Summary

B. Complex Equation Solver

This module consists of a CNN model to recognize equations of different degrees.

OPERATIONAL LAYER	INPUT	OUTPUT
Convolution 2D	28, 28, 1	28, 28, 1
Convolution 2D	28, 28, 1	28, 28, 32
MaxPool2D	28, 28, 32	14, 14, 32
Dropout	14, 14, 32	14, 14, 32
Convolution 2D	14, 14, 32	14, 14, 64
Convolution 2D	14, 14, 64	14, 14, 64
MaxPool2D	14, 14, 64	7, 7, 64
Dropout	7, 7, 64	7, 7, 64
Flatten	7, 7, 64	3136
Dense	3136	256
Dropout	256	256
Dense	256	52

Fig 3. CNN Model Summary

First two layers are of 2D convolution that applies a filter which is passed to the maxpooling2D layer to downsample the input along with its dimension by taking the maximum value. The dropout layer randomly discards some of the inputs to avoid overfitting. Input is flattened before passing it to the dense layer which has units equal to the number of classes with softmax as the activation function.
RMSprop is used as an optimizer that uses the average of partial gradients in the adaptation of the step size of each parameter. Categorical cross-entropy is used as a loss function as there are multiple classes and the model is trained with a learning rate of 0.1.

III. PREPROCESSING

A. Binarization

Binarization is the method of converting any grayscale image (multi-tone image) into a black-white image (two-tone image). To perform the binarization process, first, find the threshold value of the grayscale and check whether a pixel has a particular gray value or not.
If the gray value of the pixels is greater than the threshold, then those pixels are converted into white. Similarly, if the gray value of the pixels is lesser than the threshold, then those pixels are converted into black.

B. Line Segmentation

Lines Segmentation is one the most important process in image preprocessing. Line segmentation divides images into lines. The main goal of line segmentation is to separate out the line of text from the image.

C. Character Segmentation

Separating out the individual characters from the image which is already divided into lines of text in line segmentation. Here CNN distinguishes between character and non-character regions creating fixed size bounding boxes around characters to improve the accuracy.

Fig 4. Preprocessed Image

IV. RESULTS

Sr. no	Model	Accuracy
1.	Dense Model (Simple Equation)	98%
2.	RNN Model (Simple Equation)	96%
3.	CNN Model (Complex Equation)	85%

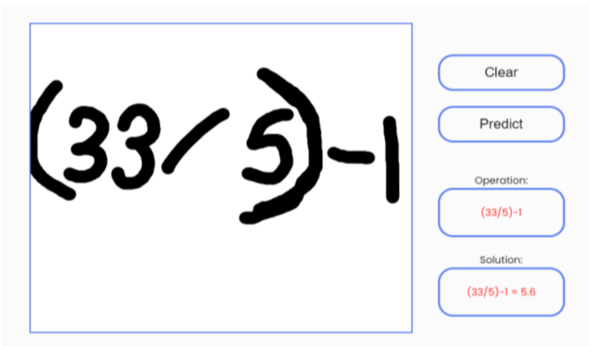

Fig 5. Recognition example with brackets

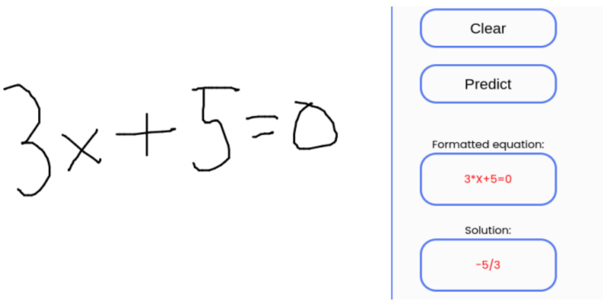

Fig 6. Recognition example with complex equation

A. System Requirements for using the software:

Hardware Requirement: Any modern CPU with 2.0 GHz of clock speed, Harddisk: 20GB minimum, RAM: 1GB
Software Requirements: Windows / Linux / macOS operating system and a web browser. Python Flask, Keras, Tensorflow.

V. CONCLUSION

In the proposed system handwritten equations, be it simple operations or complex polynomial equations, are scanned and recognized. The recognition of simple operations shows better accuracy than polynomial equations. Model for simple operations is also faster compared to the polynomial input. The RNN algorithm is used to solve the operations and give out the result. To improve the accuracy of the CNN model in complex polynomial equation recognition we need to increase the dataset of complex equations furthermore.

VI. ACKNOWLEDGEMENT

We would like to thank the open-source data set website Kaggle for making the MNIST, EMNIST datasets of digits available to all for use. We also extend our wishes to the developers of Tensorflow, Keras for creating and maintaining such libraries which are useful for various purposes.

VII. REFERENCES

[1] Mujadded Al RabbaniAlif, Sabbir Ahmed, Muhammad AbulHasan, “Isolated Bangla Handwritten Character Recognition with Convolutional Neural Network”, 978-1-5386-1150-0/17/$31.00 c 2017 IEEE.

[2] Chan, Kam-Fai, and Dit-Yan Yeung. ”Mathematical expression recognition: a survey.” International Journal on Document Analysis and Recognition 3, no. 1 (2000): 3-15.

[3] P. Sermanet, S. Chintala, and Y. Lecun, “Convolutional neural networks applied to house numbers digit classification,” International Conference on Pattern Recognition, pp. 3288 – 3291, 2013.

[4] A. M. Rush, S. Chopra, and J. Weston, “A neural attention model for abstract sentence summarization,” arXiv preprint arXiv:1509.00685, 2015.

[5] R.Parthiban, R.Ezhilarasi, D.Saravanan, “Optical Character Recognition for English Handwritten Text Using Recurrent Neural Network” 10.1109/ICSCAN49426.2020.9262379 2020 IEEE

[6] Guillaume Lample, Franc ̧oisCharton, “Deep Learning for Symbolic Mathematics”, 2020 ICLR.

[7] Shifat Nayme Shuvo, Fuad Hasan, Syed Akhter Hossain, Sheikh Abujar, “Handwritten Polynomial Equation Recognition and Simplification Using

International Journal of Engineering Applied Sciences and Technology, 2022
Vol. 6, Issue 10, ISSN No. 2455-2143, Pages 146-149
Published Online February 2022 in IJEAST (http://www.ijeast.com)

Convolutional Neural Network", 10.1109/ICCCNT49239.2020. 9225587 2020 IEEE.

[8] Fathma Siddique, Shadman Sakib, Md. Abu Bakr Siddique, " Recognition of Handwritten Digit using Convolutional Neural Network in Python with Tensorflow and Comparison of Performance for Various Hidden Layers", 10.1109/ ICAEE48663. 2019.8975496 2019 IEEE.

[9] Manjot Kaur, Aakash Mohta, "A Review of Deep Learning with Recurrent Neural Network" 10.1109/ICSSIT46314.2019. 8987837 2019 IEEE.

[10] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980, 2014

[11] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," CoRR, vol. abs/1311.2901, 2013.

[12] LeCun, Yann, and Yoshua Bengio. "Convolutional networks for images, speech, and time series." The handbook of brain theory and neural networks 3361, no. 10 (1995): 1995.

[13] Mouchre, Harold, Christian Viard-Gaudin, Richard Zanibbi, and UtpalGarain. "ICFHR2016 CROHME: Competition on Recognition of Online Handwritten Mathematical Expressions." In Frontiers in Handwriting Recognition (ICFHR), 2016 15th International Conference on, pp. 607-612. IEEE, 2016.

[14] Bala Mallikarjunarao Garlapati, Srinivas Rao Chalamala, "A System for Handwritten and Printed Text Classification" 978-1-5386-2735-8/17 $31.00 © 2017 IEEE..

[15] Kha Cong Nguyen, Nakagawa Masaki, "Enhanced Character Segmentation for Format-Free Japanese Text Recognition", 2167-6445/16 $31.00 © 2016 IEEE