Proceedings of the 2023 International Conference on Machine Learning and Automation
DOI: 10.54254/2755-2721/36/20230416

Handwritten digit recognition based on deep learning techniques

Xinshen Zhang

Faculty of Engineering, Hong Kong Polytechnic University, Hong Kong, 999077, China

22099321d@connect.edu.hk

Abstract. The identification of handwritten digits in images recognition and machine learning is a prominent research area. In order to create a handwritten digit recognition model for this investigation, deep learning is introduced. The proposed approach integrates Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), and EfficientNetB0, three separate deep learning models. Specifically, the CNN model utilizes pooling layers and data augmentation techniques to enhance its classification ability, the RNN model takes advantage of its ability to process sequential data, and the EfficientNetB0 model benefits from a deeper and more complex network structure. These models are trained and evaluated using the Modified National Institute of Standards and Technology (MNIST) dataset. The experimental results demonstrate the efficacy of the proposed approach: the CNN model attains a remarkable accuracy of 98.9% on the test set, thereby showcasing its exceptional classification performance. Similarly, the RNN model achieves an accuracy of 96.7%, underscoring its suitability for analyzing sequential data. Furthermore, the EfficientNetB0 model attains an accuracy of 98.1%, thereby elucidating the benefits of the deeper network architecture. The models constructed in this study have significant real-world implications, such as improved object recognition systems, medical diagnostics, and autonomous driving. The EfficientNetB0 model produced an accuracy of 98.1% with its complex network architecture when applied to recognise handwritten digits.

Keywords: deep learning, handwritten digit recognition, Convolutional Neural Network, recurrent neural network, EfficientNetB0.

1. Introduction

Research on handwritten digit recognition represents a widely explored topic within the domains of computer vision and machine learning. Many scholars have proposed various approaches to improve the performance of handwritten digit recognition. LeCun et al. first put forward the Convolutional Neural Network (CNN) for the purpose of recognizing handwritten digit in 1998 and attained the highest level of performance [1]. Later, many researchers proposed improvements to the CNN architecture. Residual connectivity and attention mechanisms were subsequently proposed [2,3]. Other machine learning techniques, which include decision trees and support vector machines (SVM), have been employed to recognise handwritten numbers in addition to CNN [4,5]. Although these algorithms may not be as efficient as CNNs, they have achieved encouraging results in handwritten digit recognition. Deep learning approaches for handwritten digit identification are emerging as the construction of deep

DOI: 10.54254/2755-2721/36/20230416

learning models develops. Gao et al. proposed a composite residual structure deep network for the recognition of handwritten digits in 2020. This work achieved breakthrough results on the Modified National Institute of Standards and Technology (MNIST) dataset [6]. Meanwhile, the application of attention mechanisms in sequence-to-sequence based deep learning models is introduced by T. Lupinski et al. to enhance the performance of offline handwritten digit string recognition [7]. S. Ahlawat et al introduces the application of CNNs and attention mechanisms to improve handwritten digit recognition. Related studies have provided evidence of substantial advancements in the realm of handwritten digit recognition, attributable to the incorporation of deep learning methodologies, specifically CNNs [1]. The exact requirements of the application and the available processing resources determine the methods and architectures to be used.

The study's primary objective is to use deep learning to build a model for handwritten digit recognition. In this paper, three neural network models are introduced including EfficientNetB0, Recurrent Neural Network (RNN) and CNN. Specifically, first, the MNIST dataset is preprocessed. The input data is reshaped into appropriate shapes and the pixel values are normalized. Second, three neural networks are constructed separately as a baseline. The model based on CNN architecture employs multiple convolutional layers that are concluded by a fully connected layer utilizing a softmax activation function. Additionally, the model utilizing RNN architecture incorporates a Simple RNN layer with 128 neurons, which is succeeded by a fully connected layer implementing a softmax activation function. The EfficientNetB0-based model consists of several fully connected layers and an output layer for classification purposes, followed by a softmax activation function. During the training process, the model employs the RMSprop optimizer and utilizes a categorical cross-entropy loss function. By comparing the three models' accuracy and loss on a test dataset, the study evaluates the level to which they perform. The study also develops confusion matrices to visualise the models' performance in each category. The experimental results demonstrate that the model based on EfficientNetB0 achieves the accuracy of 98.1% on the MNIST dataset. Furthermore, the analysis of the confusion matrix reveals that the EfficientNetB0-based model surpasses the performance of the other models, particularly in the challenging category. It is important to note that all three models display excellent performance across all categories. This study analyzes the accuracy of deep learning models for recognizing handwritten digits and highlights the importance of choosing the right model architecture for a given task. These findings can be used to create more accurate and efficient handwriting recognition systems.

2. Methodology

2.1. Dataset description and preprocessing

This study makes use of the MNIST Handwritten Digits dataset, which is a collection of pictures of handwritten digits compiled from the National Institute of Standards and Technology (NIST) [1]. In addition, the MNIST Handwritten Digits contains 10,000 samples for testing and 60,000 training examples. Every sample in the dataset is a grayscale image of dimensions 28 * 28 pixels. The label assigned to each image corresponds to an integer ranging from 0 to 9, representing the respective assigned number. The MNIST dataset, which is frequently used for testing purposes and extensively employed in the area of machine learning, provides a reliable approach for evaluating algorithm performance.

The data is preprocessed in a several step prior to training the model. The labels are first transformed into one-hot encoded arrays. Then, the pixel values of the images are normalized to the range [0,1] and the input data is reshaped to have a channel dimension. The picture is subsequently compressed to 32 × 32 pixels in order to analyse the supplied data. After that, normalise the enlarged image by multiplying the result by the ImageNet dataset's standard deviation and subtracting its average pixel value from each input pixel. The purpose of these preprocessing steps is to improve the machine learning model's accuracy and efficacy. Figure 1 showcases some instances from the dataset.

DOI: 10.54254/2755-2721/36/20230416

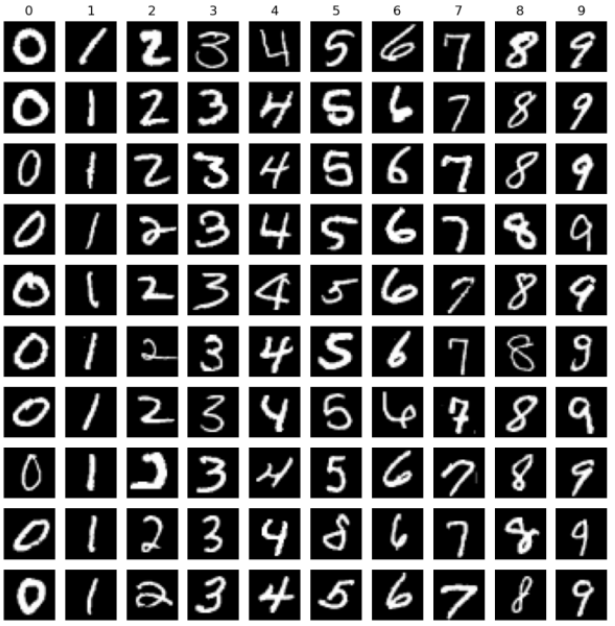

Figure 1. Images from the MNIST dataset.

2.2. Proposed approach

The project aims to create a deep learning model employing several architectures for image classification tasks. The accuracy, loss, and validation measures of three separate models have been constructed and compared. The first model employs a straightforward CNN architecture, the second model make use of a SimpleRNN model, and the third model make use of a pre-trained EfficientNetB0 model. The MNIST dataset serves for model training and evaluation, allowing for a comparison of relative performance. The processing of this paper involves loading and preprocessing the data, creating the model, training and testing the model, and assessing the outcomes.

2.2.1. CNN. With precise and simple architecture, convolutional neural networks are primarily used for solving challenging pattern recognition tasks driven by images [9]. This model is characterized by trainable convolutional layers, pooling layers, and fully connected layers. The CNN model performs effectively when processing picture data because of its automatic feature extraction and learning capabilities. This model's primary purpose in the experiment is to evaluate the effectiveness of other models. Figure 2 below illustrates the structure of this model.

The model architecture is made up of a series of layers including two fully connected layers, a dropout layer, a max pooling layer, and a convolutional layer. In the model's initial layer, a 5x5 convolution kernel with 16 filters is used to convolve the input picture. Subsequently, the feature maps are then reduced in size employing a 2x2 max pooling layer. To address the risk of overfitting, a Dropout layer is incorporated, randomly deactivating 25% of the neurons during training. Finally, a flatten layer is employed to transform the multidimensional feature map into a one-dimensional vector.

The model's next layer applies the ReLU activation function and has 128 neurons in a fully linked layer. The model includes a Dropout layer, where 50% of the neurons will be lost at random, to further reduce overfitting. The probability distribution of the N classes is output by the last layer, a fully connected layer with 10 neurons that uses a softmax activation function. Accuracy is set as the evaluation criterion for this model, which was built utilizing the Adam optimizer and the cross-entropy loss function. Finally, this model trains the model and evaluates the model with the test set.

DOI: 10.54254/2755-2721/36/20230416

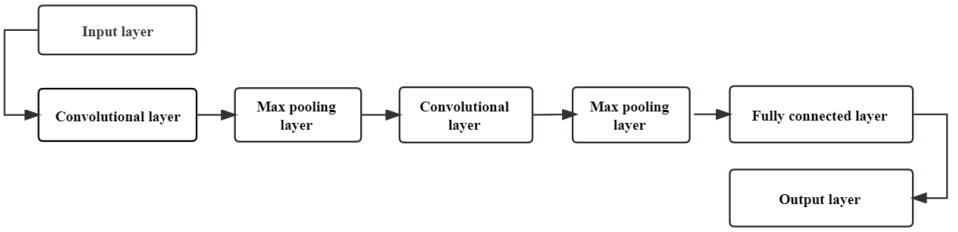

Figure 2. The pipeline of the CNN.

2.2.2. RNN. RNN model uses SimpleRNN to perform classification of photographs tasks. The model is characterized by treating the input data as a sequence and using a SimpleRNN layer to process the sequence. The suitability of RNNs for learning sequences can be demonstrated by expanding within the framework of FIR models approximating IIR systems [10]. Because it can detect temporal relationships and correlations inside the sequence, the SimpleRNN layer performs well on sequential data. This model's primary function in this experiment is to compare the effectiveness of various models. Figure 3 below illustrates the structure of this model.

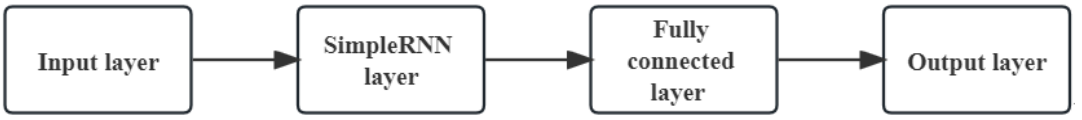

Figure 3. The pipeline of the SimpleRNN.

After adding the SimpleRNN and completely connected layers in accordance with the aforementioned network structure, this model creates a sequence model. The selected parameter used as the evaluation metric in the RNN model remains consistent with the previous one. Following training, the model is assessed using the test set.

2.2.3. EfficientNetsB0. EfficientNets can employ neural architecture search to create new baseline networks and expand them to produce a family of models that are more accurate and efficient than prior ConvNets [11]. The EfficientNetB0 architecture is employed to establish and train a deep learning model specifically designed for the purpose of image classification. The MNIST dataset serves as the training data for the model The main advantage of using a model such as EfficientNetB0 is that it can significantly reduce the total quantity of training data required to attain a discernible level of accuracy. Figure 4 below shows the structure of this model.

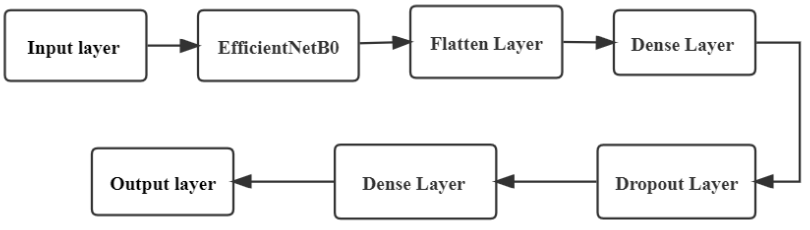

Figure 4. The architecture of the EfficientNetB0.

The model, in this particular instance, undergoes initialization without the utilization of pre-trained weights. Input images are resized to 32x32 pixels, normalized and passed through a EfficientNetB0-

DOI: 10.54254/2755-2721/36/20230416

based model. Following the application of the ReLU activation function and a fully connected layer consisting of 512 units, the output of the base model is flattened. Incorporating a dropout layer with a rate of 0.2 into the model architecture serves as a measure to mitigate the risk of overfitting. Subsequently, a dense layer consisting of 10 units with softmax activation is constructed to produce the predicted class probabilities for ten distinct classes. During the training process, an Adam optimizer is employed in conjunction with a categorical cross-entropy loss function. The performance and efficacy of the model are assessed using accuracy metrics. The training procedure is conducted with a batch size of 128 and spans a duration of 10 epochs.

2.3. Implemented details

The CNN model utilizes a batch size of 64 batches. Data augmentation techniques, including rotation, translation, shearing, and scaling, are applied, and pixel values are normalized by dividing them by 255. In contrast, the RNN model employs a batch size of 32. During the image processing stage, the pixel values are divided by 255 for normalization purposes. Moreover, these images undergo grayscale conversion as part of the image processing pipeline. As for the efficient netb0 model, the batch size is set to 128, and data augmentation techniques like rotation, translation, shearing, and scaling are implemented. Similarly, pixel values are normalized by division with 255. The chosen optimizer for all three models is Adam optimizer due to its efficient handling of gradient descent in high-dimensional spaces. Data augmentation techniques are applied to the dataset to augment its size and diversity, thereby reducing overfitting. All three models adopt a learning rate of 0.001 and train for 10 epochs. Additionally, all models normalize their pixel values.

3. Result and discussion

The performance displayed by three different models in the area of handwritten digital image classification is examined in this chapter. The CNN model uses data augmentation methods and a Dropout layer to enhance performance. The RNN model makes predictions about sequential data based on prior knowledge. Finally, despite being initialised with 'None' weights, the EfficientNetB0 model displays higher performance and generalisation abilities due to its inherent architectural design. The EfficientNetB0 model is renowned for its effective and efficient feature extraction skills, which enable it to identify handwritten digits well without the use of pre-trained weights.

3.1. The performance of CNN model

Figure 5 above demonstrates the accuracy and the loss curve of the CNN. The results of model training indicate a consistent increase in accuracy of both the training and the validation set. Simultaneously, the loss value demonstrates a gradual decrease, indicating a progressive improvement in the model's learning and performance. These findings affirm that the model continuously learns and enhances its predictive capabilities throughout the training process. The model also efficiently reduces overfitting and enhances the model's generalizability because it makes use of the Dropout layer. The model has an excellent classification capacity and may be used in real-world application scenarios, as shown by the test set's final accuracy rate of 98.9%. The explanation for this result could be that the data set quality is high, the model structure and parameter settings are appropriate, and the dropout layer and other technologies are employed for model optimization, which enhances the model's performance and generalizability.

DOI: 10.54254/2755-2721/36/20230416

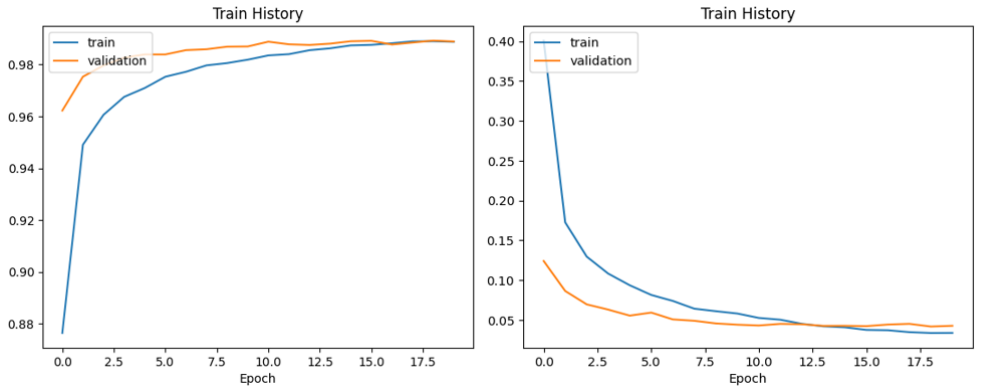

Figure 5. The accuracy (left) and the loss (right) curve of the CNN.

3.2. The performance of RNN model

Figure 6 above demonstrates the accuracy and the loss curve of the RNN. From the results, it can be seen that the accuracy rate for identifying handwritten digital images using a RNN is 96.7%, which is a little lower than the prior findings using a simple feedforward neural network (FNN). This may be because feedforward neural networks are adequate for handling static data, such as handwritten digital photographs, whereas RNNs can leverage the prior knowledge to better forecast the results when processing sequence data. Additionally, the accuracy of the model on training and validation sets is gradually increasing, and the loss value is gradually decreasing, indicating that the model is learning and optimizing. However, due to the model's overfitting on the training set, the accuracy on the test set declines marginally. Regularization techniques or extra training data may be applied to prevent overfitting. A possible explanation for this outcome is that the model was optimized using the Adam optimizer and other technologies, which increased the model's performance and generalizability due to the excellent data set quality, suitable model structure, and parameter values. This result is significant since it both demonstrates the recurrent neural network's efficiency in processing sequence data and offers suggestions and guidelines for further optimizing and improving the model.

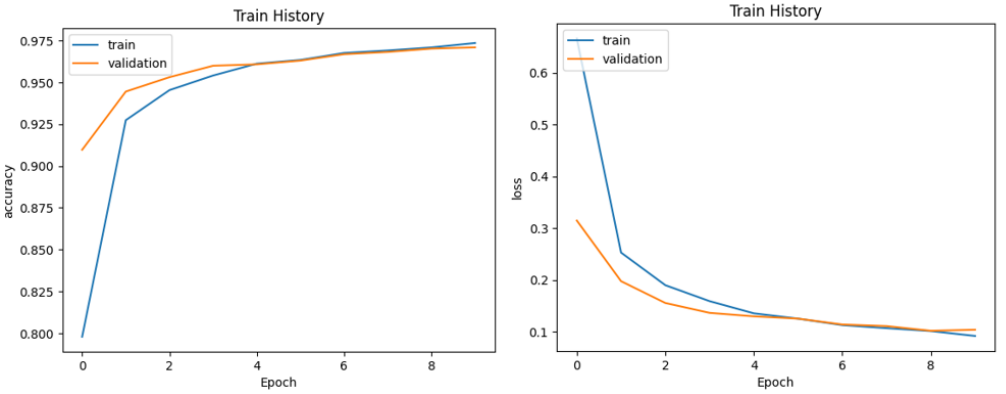

Figure 6. The accuracy (left) and the loss (right) curve of the RNN.

3.3. The performance of EfficientNetB0

Figure 7 above demonstrates the accuracy and the loss curve of the EfficientNetB0. The performance metrics indicate that the EfficientNetB0 model achieves a classification accuracy of 98.1% when tasked with classifying images, which is much higher than the accuracy rate of the prior RNN model. This is because the EfficientNetB0 model has better picture feature extraction and generalization abilities. Moreover, the model's accuracy on both the training and validation sets exhibits a consistent upward trend, while the loss value consistently decreases over time. These observations indicate a continuous

DOI: 10.54254/2755-2721/36/20230416

improvement and learning process of the model. The accuracy rate on the test set is also high, demonstrating the model's good generalization capacity for untried data. The EfficientNetB0 model's deeper network structure and more parameters, which can better extract picture features, are the cause of this finding. According to experimental findings, many models each have their own advantages and situations in which they can be used for picture categorization tasks. RNN model may not be the ideal option because it performs slightly worse than other models in image classification jobs while CNN model performs well on image data. Large image datasets yield better results for the EfficientNetB0 model.

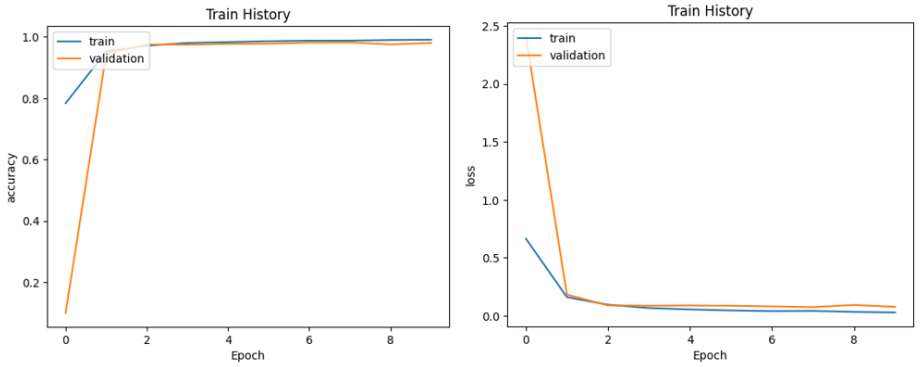

Figure 7. The accuracy (left) and the loss (right) curve of the EfficientNetB0.

4. Conclusion

This study focuses on the research topic of handwritten digit recognition using deep learning models. Three models are introduced including (CNN, RNN and EfficientNetB0) to analyze the images. These models are trained and evaluated using MNIST dataset and their performance is evaluated based on accuracy metrics. The experimental results serve as a reflection of how effective the suggested approach is. The CNN model achieves an impressive 98.9% accuracy on the test set while the RNN model achieves an accuracy of 96.7%, highlighting its ability to process sequential data and utilize prior knowledge. The EfficientNetB0 model, a more complex network, achieved 98.1% accuracy. These findings highlight how effectively deep learning models work to recognise handwritten digits. In addition, the study demonstrates data enhancement techniques and regularization methods to improve model performance. For future work, the research will consider other model as the research objective for the next stage. The focus will be on further analyzing and understanding the real-time number detection, aiming to enhance the performance of the image recognition models.

References

[1] LeCun Y Bottou L Bengio Y Haffner P 1998 Gradient-based learning applied to document recognition Proceedings of the IEEE 86(11): pp 2278-2324

[2] He K Zhang X Ren S Sun J 2016 Deep Residual Learning for Image Recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) IEEE pp 770-778

[3] Wang F Jiang M Qian C et al. 2017 Residual Attention Network for Image Classification. arXiv:1704.06904

[4] Boser B Guyon I Vapnik V 1992 A training algorithm for optimal margin classifiers. Annual Workshop on Computational Learning Theory: Proceedings of the fifth annual workshop on Computational learning theory ACM pp 144-152

[5] Bernard S Adam S Heutte L 2007 Using Random Forests for Handwritten Digit Recognition ICDAR 2007: NINTH INTERNATIONAL CONFERENCE ON DOCUMENT ANALYSIS AND RECOGNITION, VOLS I AND II, PROCEEDINGS IEEE pp 1043-1047

DOI: 10.54254/2755-2721/36/20230416

[6] Barolli L Zhang M Wang X A 2018 A Deep Network with Composite Residual Structure for Handwritten Character Recognition ADVANCES IN INTERNETWORKING DATA & WEB TECHNOLOGIES EIDWT-2017 Springer International Publishing AG pp 160-166
[7] Lupinski T Belaid A Kacem Echi A 2019 On the Use of Attention Mechanism in a Seq2Seq Based Approach for Off-Line Handwritten Digit String Recognition. 2019 International Conference on Document Analysis and Recognition (ICDAR) IEEE pp 502-507
[8] Ahlawat S Choudhary A Nayyar A Singh S Yoon B 2020 Improved handwritten digit recognition using convolutional neural networks (Cnn) Sensors (Basel) 20(12): p 3344
[9] Fukushima K 1980 Neocognitron: a self organizing neural network model for a mechanism of pattern recognition unaffected by shift in position Biological cybernetics 36(4): pp 193-202
[10] Sherstinsky A 2020 Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network. Physica. D 404: p 132306
[11] Tan M Le Q V 2019 EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks arXiv:1905.11946