BILINGUAL PUBLISHING GROUP

Journal of Computer Science Research

https://journals.bilpubgroup.com/index.php/jcsr

ARTICLE

Assessing Four Neural Networks on Handwritten Digit Recognition Dataset (MNIST)

***Feiyang Chen*[1], *Ziqian Luo*[2*], *Nan Chen*[3], *Hanyang Mao*[3], *Hanlin Hu*[3], *Ying Jiang*[4], *Xueting Pan*[2], *Huitao Zhang*[5]**

[1] *Coupang, Mountain View, 94043 CA, USA*

[2] *Oracle, Seattle, WA 98101, USA*

[3] *Beijing Forestry University, Beijing 100083, China*

[4] *Carnegie Mellon University, Pittsburgh 15213, PA, USA*

[5] *Northern Arizona University, Flagstaff, 86011 AZ, United States*

ABSTRACT

Although the image recognition has been a research topic for many years, many researchers still have a keen interest in it. In some papers, however, there is a tendency to compare models only on one or two datasets, either because of time restraints or because the model is tailored to a specific task. Accordingly, it is hard to understand how well a certain model generalizes across image recognition field. In this paper, we compare four neural networks on MNIST dataset with different division. Among of them, three are Convolutional Neural Networks (CNN), Deep Residual Network (ResNet) and Dense Convolutional Network (DenseNet) respectively, and the other is our improvement on CNN baseline through introducing Capsule Network (CapsNet) to image recognition area. We show that the previous models despite do a quite good job in this area, our retrofitting can be applied to get a better performance. The result obtained by CapsNet is an accuracy rate of 99.75%, and it is the best result published so far. Another inspiring result is that CapsNet only needs a small amount of data to get the excellent performance. Finally, we will apply CapsNet's ability to generalize in other image recognition field in the future.

Keywords: Neural network; CNN; CapsNet; DenseNet; ResNet; MNIST

*CORRESPONDING AUTHOR:

Ziqian Luo, Oracle, Seattle, WA 98101 USA; Email: luoziqian98@gmail.com

ARTICLE INFO

Received: 28 June 2024 | Revised: 5 July 2024 | Accepted: 5 July 2024 | Published Online: 28 July 2024

DOI: https://doi.org/10.30564/jcsr.v6i3.6804

CITATION

Chen, F., Luo, Z., Chen, N., et al., 2024. Assessing Four Neural Networks on Handwritten Digit Recognition Dataset (MNIST). Journal of Computer Science Research. 6(3): 17–22. DOI: https://doi.org/10.30564/jcsr.v6i3.6804

1. Introduction

Motivated by the rapid development of artificial intelligence and its numerous applications, there has been significant progress in the field of image recognition over the past decade [1]. This progress includes the introduction of many innovative models that have pushed the boundaries of what is possible in recognizing and interpreting visual data. Additionally, the creation of benchmark datasets has provided a standard against which these new models can be measured, facilitating the comparison and evaluation of different approaches [2]. The improvements in computational power and the advent of deep learning have further accelerated advancements in this area, leading to remarkable achievements in various image recognition tasks [3].

In some papers, however, there is a noticeable tendency to evaluate models using only one or two datasets. This is often due to constraints such as limited time or the specific focus of the model on a particular task [4]. As a result, these studies may not fully capture the model's ability to generalize across the broader field of image recognition. Evaluating a model on a limited number of datasets can provide insights into its performance on those specific tasks, but it leaves questions about how well the model would perform in different contexts or on more diverse datasets [5]. This limitation underscores the need for more comprehensive evaluations to better understand the generalizability of image recognition models.

In this paper, our main contributions are, therefore, centered around a comprehensive comparison of four mainstream image recognition models using the MNIST dataset, with varying data divisions [6]. Among these models, three are well-known: Convolutional Neural Network (CNN), Deep Residual Network (ResNet), and Dense Convolutional Network (DenseNet). These models have already been proven to deliver high performance in image recognition tasks, and their characteristics and strengths are summarized in detail in Section 2. However, despite their success, we identify certain drawbacks in the standard CNN model, as highlighted in previous research [7,8]. To address these issues, we use the CNN model as a baseline and introduce improvements by applying Capsule Network (CapsNet) optimizations. This enhanced model, which is our fourth, is described in detail in Section 3.

We use the MNIST dataset for our experiments because the recognition of handwritten digits remains a practical and significant topic in the field of image recognition. The MNIST dataset has been a cornerstone in this research area, serving as a standard benchmark for evaluating new models and techniques [9]. It offers several advantages: firstly, the existence of well-established benchmark datasets like MNIST allows for easy acquisition and comparison of results. Secondly, there is a wealth of publications and established techniques related to MNIST, providing a solid foundation for building upon existing research. To assess the generalizability of the models in the field of image recognition, we divided the MNIST dataset into subsets of 25%, 50%, 75%, and 100% for testing purposes.

Ultimately, our study contributes to a deeper understanding of the performance of different model architectures on the MNIST dataset. Through our experiments, we have determined that CapsNet outperforms the other models across all tasks, consistently providing better results than the baseline. The experimental results are detailed in Section 4, showcasing the superior performance of CapsNet. The conclusion and implications of our findings are discussed in Section 5, offering insights into the future potential of CapsNet and other advanced models in the broader field of image recognition [10].

2. Related works

This section describes MNIST dataset which will be used in the experiments and then discusses the characteristics of the three neural network models.

2.1 Dataset

The MNIST dataset is from the National Institute of Standards and Technology (NIST). The training set consists of handwritten numbers from 250 different people, of which 50% are high school students

and 50% are from the Census Bureau. The test set is also the same proportion of hand- written digital data. MNIST dataset totally contains 60,000 images in the training set and 10,000 patterns in the testing set, each of size 2828 pixels with 256 gray levels [11]. The dataset can be downloaded online and some examples from the MNIST corpus are shown in **Figure 1**.

Figure 1. Example images from the MNIST dataset, including 60,000 images in the training set and 10,000 patterns in the testing set.

2.2 CNN

In machine learning, CNN is a feed-forward artificial neural networks, most commonly applied to analyzing visual imagery. For CNN, the earliest date can be traced back to the 1986 BP algorithm [12]. Then in 1989 LeCun used it in multi-layer neural networks [13]. Until 1998, LeCun proposed the LeNet-5 model, and the neural network prototype was completed. CNN consists of one or more convolutional layers and the top fully connected layer, and it also includes associated weights and a pooling layer. This structure allows the convolutional neural network to take advantage of the two dimensional structure of the input data, so it can give very good results in image recognition [14]. So we try to apply it to the MNIST dataset for testing.

2.3 ResNet

Deep convolutional neural networks have led to a series of breakthroughs for image classification. However, when deeper networks are able to start converging, a degradation problem [15] has been exposed: with the network depth increasing, accuracy gets saturated and then degrades rapidly. Therefore, ResNet is presented in 2017. It can reduce the train_ error while deepening the depth of the network, and solve the problem of gradient dispersion [16], improving network performance, which is shown in the equation (1). Most importantly, ResNet can not only be very deep, but also has a very simple structure. It is a very small single module piled up, its unit module block as shown in **Figure 2**.

$$x_l = H_l(x_l - 1) + x_l - 1 \tag{1}$$

In the equation (1), l represents layer, x_l represents the output of the l layer, H_l represents a nonlinear transformation. For ResNet, the output of the l layer is the output of the $l - 1$ layer plus the nonlinear transformation of the output of the $l - 1$ layer.

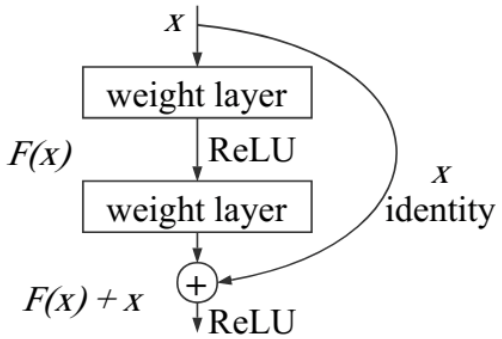

Figure 2. Unit module block, where x means the input and F(x) means the output of the weight layer, the final output is the sum of F(x) and x.

2.4 DenseNet

In the field of image recognition, CNN has become the most popular method. A milestone in the history of CNN is the emergence of the ResNet model [17]. ResNet can train deeper CNN models to achieve higher accuracy. The basic idea of the DenseNet model is the same as that of ResNet, but it establishes a dense connection between all previous and subsequent layers [18]. Its other major feature is feature reuse through the connection of features on the channel. Therefore, we also tested its performance on the MNIST dataset.

3. Experimental setup

We compare four models, three of which are mentioned in Section 2. The other is our retrofitting and improvement based on CNN model. It is described in detail in Section 4.2 and 4.3. We use the MNIST datasets mentioned in Section 2.1 to test these models.

We tested all the models using a workstation built from commodity hardware: dual GeForce GTX 1080 graphics cards, an i7-6800K CPU, and 64 GB of RAM. Our implementation is in TensorFlow and we use the Adam optimizer with TensorFlow default parameters, including the exponentially decaying learning rate, to minimize the sum of the margin losses.

3.1 Baseline

Our model is based on a standard CNN with three convolutional layers, which is demonstrated to achieve a low test error rate on MNIST. The channels of three layers are 256, 256, 128 respectively. Each layer has 5 × 5 kernels and stride of one. Followed by the last convolutional layers are two fully connected layers of size 328, 192. After that is a 10 class softmax with cross entropy loss. However, the baseline model has two shortcomings. First, training a powerful CNN model requires a large number of training data. Second, in the pooling layer, CNN loses some of the information, which leads to the ignorance of interrelation- ships between different component [19].

Thus, for small changes in input, the output of CNN will be almost constant, which may result in a higher error rate.

3.2 Retrofitting

In order to overcome these shortcomings of CNN, we try to introduce CapsNet to optimize the baseline. **Figure 3** shows the structure of the CapsNet. CapsNet uses capsules instead of neurons. The input and output of the capsule are high dimensional vectors, where the module length represents the probability of occurrence of an object, and the direction represents the position, color, size and other information [20]. The output of the low-level capsules is used to generate a prediction through transformation matrices, which are then linearly integrated and passed into high-level capsules according to certain weights. The method of updating the weights is a dynamic routing algorithm, which compares the output of high-level capsules with the prediction of lowlevel capsules, and increases the input weights of low-level capsules with higher similarity until convergence. Through the capsule, we retain the information on the details of the picture. In this way, on the basis of accurately identifying the image, small changes in the image input will cause small changes in the output. It has a human-like understanding of the three-dimensional space. In addition, with less information loss, it only needs a small amount of data to achieve amazing results compared to CNN.

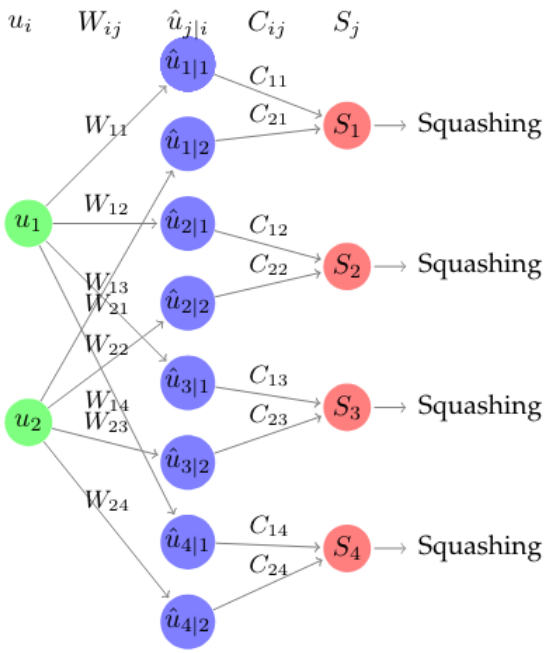

Figure 3. Structure of CapsNet, where u_i is the input layer.

3.3 CapsNet architecture

The architecture is showed in **Figure 4**, it consists of one convolutional layer and two capsule layers [21,22]. The convolutional layer 1 has 256, 9 × 9 convolution kernels with a stride of 1 and ReLU activation. This layer extracts the basic features of the image, and then uses them as the inputs of the

primary capsules layer (PrimaryCaps). The PrimaryCaps contains 32 capsules, which receives the basic features detected by the convolution layer, creating a combination of features. The 32 primary capsules in this layer are essentially similar to the convolutional layer [23]. Each has 8, 9 × 9 × 256 convolution kernels with a stride of 2. The output of PrimaryCaps is 6632 eight-dimensional vector. The last layer is digital capsules layer (DigitCaps), it has 10 digital capsules and each of which represents the prediction of number. Every capsule receives input from all capsules in the PrimaryCaps, and finally outputs the result.

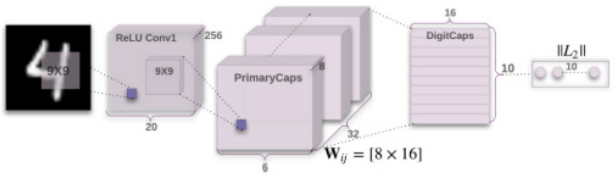

Figure 4. A simple CapsNet with three layers. The convolutional layer extracts image features, PrimaryCaps integration the features, and DigitCaps output the prediction.

4. Result

We randomly divided the MNIST dataset into 25%, 50%, 75%, and 100%. **Table 1** shows the results for the four models across all divided datasets, and we visualize them in **Figure 5**. Obviously, CNN continues to be a strong baseline: Though it never provides the best result on a dataset, it gives better results than ResNet on 25% MNIST. Because the ResNet's network structure requires a larger number of data to train. DenseNet performs better than CNN on all divided datasets. It also improves the results of ResNet across all datasets but 50% dataset. That is related to DenseNet's parameter settings. Inspiringly, CapsNet is the best overall model, which outperforms the other models on all tasks and consistently beats the baseline. In addition, we can observe from the **Table 1** that CapsNet trained with half datasets reach approximately equal accuracy with complete CNN. We attribute this to CapsNet's ability to generalize in image recognition. This is in line with other research [24], which suggests that this model is very robust across tasks as well as datasets.

Table 1. Results of experiment on divided datasets.

MNIST Accuracy(%) Models	25%	50%	75%	100%
CNN	80.73	86.73	91.23	98.32
ResNet	79.46	90.55	93.78	99.16
DenseNet	82.57	89.24	94.20	99.37
CapsNet	87.68	97.12	98.79	99.75

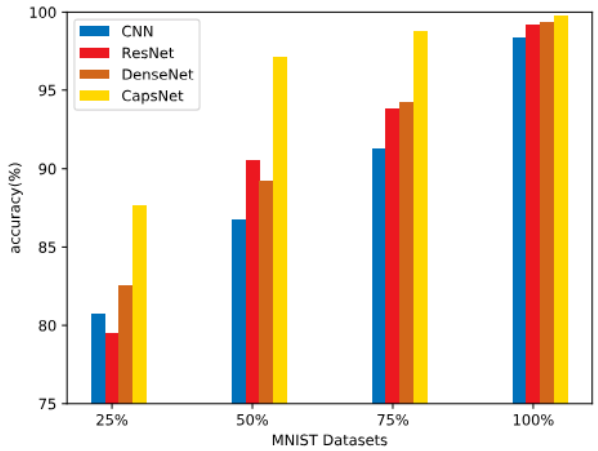

Figure 5. The results for the four models across all divided datasets.

5. Conclusions

The goal of this paper has been to discover which models perform better across divided MNIST datasets. We com- pared four models on MNIST dataset with different division, and showed that CapsNet perform best across datasets. Additionally, we also observe surprisingly that CapsNet requires only a small amount of data to achieve excellent performance. Finally, we will apply CapsNet's ability to generalize in other image recognition field in the future.

References

[1] Chen, F., Luo, Z., Xu, Y., et al., 2019. Complementary fusion of multi-features and multi-modalities in sentiment analysis. arXiv preprint arXiv: 1904.08138.

[2] Luo, Z., Xu, H., Chen, F., 2019. Audio Sentiment Analysis by Heterogeneous Signal Features Learned from Utterance-Based Parallel Neural Network. In AffCon@AAAI. pp. 80–87.

[3] Luo, Z., Zeng, X., Bao, Z., et al., 2019. Deep learning-based strategy for macromolecules classification with imbalanced data from cellular electron cryotomography. In 2019 International Joint Conference on Neural Networks (IJCNN). IEEE. pp. 1–8.

[4] Luo, Z., 2023. Knowledge-guided Aspect-based Summarization. In 2023 International Conference on Communications, Computing and Artificial Intelligence (CCCAI). IEEE. pp. 17–22.

[5] Chen, F., Luo, Z., 2019. Sentiment Analysis using Deep Robust Complementary Fusion of Multi-Features and Multi-Modalities. CoRR.

[6] Chen, F., Luo, Z., 2018. Learning robust heterogeneous signal features from parallel neural network for audio sentiment analysis. arXiv preprint arXiv: 1811.08065.

[7] Luo, Z., Xu, H., Chen, F., 2018. Utterance-based audio sentiment analysis learned by a parallel combination of cnn and lstm. arXiv preprint arXiv: 1811.08065.

[8] Chen, F., Luo, Z., Zhou, L., et al., 2024. Comprehensive survey of model compression and speed up for vision transformers. arXiv preprint arXiv: 2404.10407.

[9] Pan, X., Luo, Z., Zhou, L., 2022. Comprehensive Survey of State-of-the-Art Convolutional Neural Network Architectures and Their Applications in Image Classification. Innovations in Applied Engineering and Technology. pp. 1–16.

[10] Zhou, L., Luo, Z., Pan, X., 2024. Machine learning-based system reliability analysis with Gaussian Process Regression. arXiv preprint arXiv: 2403.11125.

[11] MLA Chatfield, Ken, et al., 2011. The devil is in the details: an evaluation of recent feature encoding methods. British machine vision conference. p. 1–12.

[12] Ba, J., Mnih, V., Kavukcuoglu, K., 2015. Multiple object recognition with visual attention. International conference on learning representations.

[13] Goodfellow, I.J., Bulatov, Y., Ibarz, J., et al., 2014. Multi-digit Number Recognition from Street View Imagery using Deep Convolutional Neural Networks. International conference on learning representations.

[14] Hinton, G.E., Ghahramani, Z., Teh, Y.W., 2000. Learning to Parse Images. Neural Information Processing Systems. 463–469.

[15] Deng, J., Dong, W., Socher, R., et al., 2009. ImageNet: A large-scale hierarchical image database. Computer Vision And Pattern Recognition. 248–255.

[16] Goodfellow, I.J., et al., 2013. Maxout Networks. International Conference on Machine Learning.

[17] Abadi, M., Agarwal, A., Barham, P., et al., 2015. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. arXiv: Distributed, Parallel, and Cluster Computing.

[18] Ren, S., He, K., Girshick, R.B., et al., 2015. Object Detection Networks on Convolutional Feature Maps. IEEE Transactions on Pattern Analysis and Machine Intelligence. 39, 1476–1481.

[19] Chang, J., Chen, Y., 2015. Batch-normalized Maxout Network in Network. arXiv: Computer Vision and Pattern Recognition.

[20] Pan, X., Luo, Z., Zhou, L., 2024. Navigating the landscape of distributed file systems: Architectures, implementations, and considerations. arXiv preprint arXiv: 2403.15701.

[21] He, K., Sun, J., 2015. Convolutional neural networks at constrained time cost. computer vision and pattern recognition. 5353–5360.

[22] Ren, S., He, K., Girshick, R., et al., 2015. Faster R-CNN: towards real-time object detection with region proposal networks. Neural Information Processing Systems. 91–99.

[23] Hinton, G.F., 1981. Shape representation in parallel systems. International Joint Conference on Artificial Intelligence. 1088–1096.

[24] Glorot, X., Bordes, A., Bengio, Y., 2011. Deep Sparse Rectifier Neural Networks. International Conference on Artificial Intelligence And Statistics. 315–323.