Mini Project 3: Modified MNIST

COMP 551

Marcos Cardenas-Zelaya, Raphael Hotter, Viet Nguyen

March 18, 2019

Abstract

We apply and evaluate several convolutional neural network (CNN) architectures — residual neural networks (ResNet 34, ResNet 50, ResNet 101, ResNet 152), Squeeze-and-Excitation Networks (SE-ResNet 56), and a custom 22-layer CNN — to image classification on a modified MNIST dataset. The task is to find the digit in the image which occupies the most space. We found that the Squeeze and Excitation Network (SE-ResNet 56) performed best with an accuracy of 97.48% on the validation set and 97.56% on the test set.

1 Introduction

In this paper we exploit ResNet's increased accuracy to predict images on the modified MNIST data set. However, due to computational restrictions we are forced to look for novel ways to cut on computational costs while maintaining a desired level of accuracy. This accuracy-cost trade-off is addressed by the SE-block architectures that improve accuracy on shallower models at only a nominal increase in computational cost (Hu et al, 2018). By including SE-blocks in our shallow networks we were able to achieve high levels of accuracy comparable to deeper networks while using less computational power. In particular, we explore the use of residual neural networks (ResNet 34, ResNet 50, ResNet 101, ResNet 152, Wide ResNet), Squeeze-and-Excitation Networks (SE-ResNet 56), and a custom 22-layer CNN for the objective. We found that the SE-ResNet 56 has the greatest accuracy on the validation set, with an accuracy of 97.48%. The SE-ResNet 56 also achieved an accuracy of 97.56% on the test set. We also observed that the SE-ResNet converged much more rapidly than the other models, with a validation accuracy of 91.64% after only 2 epochs.

Hu et al. (2018) showed that by adding SE blocks to ResNet's, as SE-ResNet-50, it had comparable validation errors as the deeper ResNet-101 but at half the computational cost. They show that SE-blocks are flexible enought to be added to any model at only a nominal cost. Indeed, by adding SE-blocks to ResNets, Hu at al. (2018) received first place using this method in the ILSVRC 2017 classification competition.

2 Related Work

The ImageNet Large Scale Visual Recognition Competition (ILSVRC) is a competition for large scale object detection and image classification algorithms. The 1st place winners in 2015 were He et al, with the use of deep residual networks (He et al, 2015). Hu et al, built on the design of He et al and invented a Squeeze-and-Excitation Network, which won 1st place in the 2017 ILSVRC competition (Hu et al, 2018).

Object detection involves the drawing of bounding boxes around objects in images and its labeling. This task is related to the selection of the largest digit, by area, in an image. You Only Look Once (Redmon et al, 2015) predicts bounding boxes in one evaluation of full images efficiently.

Wan et al (Wan et al, 2013) achieves an error of 0.21% on MNIST with its DropConnect algorithm, by randomly setting a subset of weights to zero. Maxout Network in Network (Chang & Chen, 2015) has state-of-the-art performance on the MNIST (0.24% error), CIFAR-10 (6.75% error), and CIFAR-100 (28.86 % error) datasets.

3 Dataset and Setup

The classic MNIST dataset contains 60,000 images of handwritten digits. The dataset has been used as a benchmark for evaluating algorithms in computer vision. State-of-the-art performances on the classic MNIST dataset reach 99%+ accuracy and error rates around .2% (Wan et al. (2013), Chang & Chen (2015)).

The modified MNIST dataset consists of a training set of 40,000 images and a test set of 10,000 images. Each image is a 64×64 grayscale image which contains several digits from 0 to 9. Each image is labelled with the digit which covers the largest area, i.e., with the largest bounding box. We split the 40,000 images into 36,000 training images and 4,000 validation images.

For pre-processing, we normalize the data by dividing it by 255 and add two additional channels to each image which contains an exact copy of the image, as to make the images compatible with certain CNN architectures (which expect 3-channel RGB input). We also experimented with thresholding for pixel values above and below 230, in hopes of better denoising.

4 Proposed Approach

We train residual neural networks (ResNet 34, ResNet 50, ResNet 101, ResNet 152), Squeeze-and-Excitation Networks (SE-ResNet 56), and a custom 22-layer CNN(Net, SE-Net) on the modified MNIST dataset. The CNN acts as the base that all other models are compared against. Each network is trained with 20 epochs. We used a decreasing learning rate, α: $\alpha = 0.002 \cdot (0.6)^{\lfloor \frac{3}{3} \rfloor}$. The Se-ResNets used a reduction ratio of 16 as Hu et al. (2018) found this was the optimal value to prevent overfitting. The size of a mini-batch is set to 128, and the optimizer of choice was the Adam optimizer (Kingma et al, 2014).

4.1 Convolutional Neural Networks

CNNs are composed of alternating convolution and pooling layers. Each convolution layer has a set of filters that are learned to express local spatial connectivity patterns along input channels. In particular, convolution filters are combinations of spatial and channel-wise information within local receptive fields. For the implementation of our custom CNN, we used batch normalization, Adam optimization, and dropout regularization with probability 0.25. Max pooling was used on the first, second, and last convolutional blocks. The output of the convolutional layers are flattened and fed into a two-layer neural network for a softmax output.

4.2 Residual Neural Networks

The mantra of neural networks is that network depth is of crucial importance and the current state of the art networks exploit this by adding depth to their models (Simonyan and Zisserman, 2015). Despite the importance of network depth, deep networks suffer from the *degradation* problem where accuracy degrades after reaching a certain depth (He and Sun, 2015). ResNets help solve the problem of deep networks by introducing skip connections shown in the left side of Figure 2 (He, Zhang et al, 2015). He, Zhang et al, (2015) argue that increasing the depth of a ResNet could only increase its performance since the network could artificially reduce its depth by learning identity mappings between layers. We test several standard ResNet models: ResNet-34, ResNet-50, ResNet-101, and ResNet-152.

4.3 Squeeze-and-Excitation Networks

Squeeze-and-Excitation Networks allow for much greater accuracy on shallower models at only a nominal increase in computational cost (Hu et al, 2018). They showed that by adding SE blocks to ResNet's, for example as SE-ResNet-50, it had comparable validation errors as the deeper ResNet-101 but at half the computational cost.

As opposed to traditional CNNs which treat each channel separately, SE Networks model the inter-dependencies between channels. This architectural design acts as a form of feature "re-calibration" by allowing a network to learn to use global information to selecting more relevant features over less informative ones.

For a given transformation $F_{tr} : X \to U$, $X \in \mathbb{R}^{H \times W \times C}, U \in \mathbb{R}^{H' \times W' \times C'}$ where $H, W, andC$ are height, width and number of channels respectively, an SE block is is constructed that passes features U through F_{sq}, aggregating the feature maps across $H \times W$ to produce a channel descriptor which embeds the global distribution of channel-wise feature responses.

Squeeze: Global Information Embedding

SE blocks get a global understanding of each channel by "squeezing" global spatial information into a channel descriptor via global average pooling to generate channel-wise statistics $\boldsymbol{z} \in \mathbb{R}^C$ which is accomplished by shrinking U through spatial dimensions $H \times W$ as

$$z_c = F_{sq}(u_c) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_c(i, j) \tag{1}$$

where z_c is the c^{th} element of $\boldsymbol{z}$. They describe the transformation output U as a collection of the local descriptors whose statistics are expressive for the whole image.

Excitation: Adaptive Recalibration

It is then fed through a two-layer neural network, which represents the "excitation" part. Specifcally, the information obtained in F_{sq} is passed through a simple gating mechanism with a sigmoid activation:

$$s = F_{ex}(z, W) = \sigma(g(z, W)) = \sigma(W_2 \delta(W_1 z)) \tag{2}$$

These represent two fully connected layers that act as a bottleneck, reducing dimensionality. It contains parameters $W_1 \in \mathbb{R}^{\frac{C}{r} \times C}$ with reduction ratio r, a ReLU δ. It also contains a demensionality increasing layer with parameters $W_2 \in \mathbb{R}^{C \times \frac{C}{r}}$. The transformation output U is then rescaled to obtain the final output of the block with the activations: $\tilde{x}_c = F_{scale}(u_c, s_c) = s_c \times u_c$, where $\tilde{x}_c \in \tilde{X}, c = 1$, and $F_{scale}(u_c, s_c)$ refers to the channel-wise multiplication of scalars $s_c \in \boldsymbol{s}$ and $u_c \in U$.

Hu et al note that the activations act as channel weights on z which makes SE blocks introduce dynamics conditioned on the input which helps with feature discrimination.

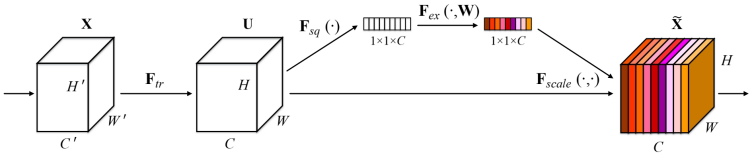

Figure 1: A Squeeze-and-Excitation block (from Hu et al, 2018).

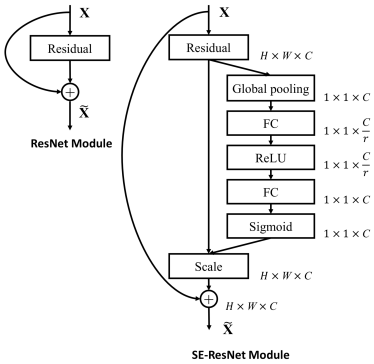

Figure 2: The schema of the original Residual module (left) and the SE-ResNet module (right).

Figure 1 and 2 depict the schema of an SE-ResNet module.

5 Results

As per Hu et al. (2018) predictions, by including SE-blocks, we were able to achieve high levels of accuracy comparable to deeper networks while using less computational power. The SE-ResNet outperformed the custom CNN and ResNets. Indeed, the SE-ResNet-34 performed the best with an accuracy of 97.56% on the validation set. Furthermore, the SE-ResNet had a comparable runtime to the other models without the SE-blocks supporting the Hu et al. (2018) assertion that SE blocks improve performance at a nominal increase in cost. In fact, the SE-ResNet achieves much quicker convergence than other methods, so the runtime until convergence was in fact much less than we report for 20 epochs in Table 1. The experimental results are summarized in Table 1. Figure 3 compares the convergence speed of two deep models, and highlights SE-ResNet's much smaller computation cost compared to other similarly deep networks.

Model	Validation Accuracy	Training Error	Runtime
Custom CNN	94.00 %	3.47 %	735 s
ResNet-34	96.30 %	1.10 %	1025 s
ResNet-34 pre-trained	95.32 %	0.01 %	925 s
ResNet-34 + threshold	91.91 %	2.11 %	1010 s
ResNet-50	94.40 %	0.88 %	1245 s
ResNet-101	95.07 %	0.20 %	1400 s
ResNet-152	93.71 %	0.96 %	1450 s
SE-ResNet-34	**97.56 %**	0.11 %	1150 s
SE-ResNet-50	97.39 %	0.06 %	1750 s

Table 1: Image recognition accuracy and training error rates, and runtimes on the modified MNIST validation set.

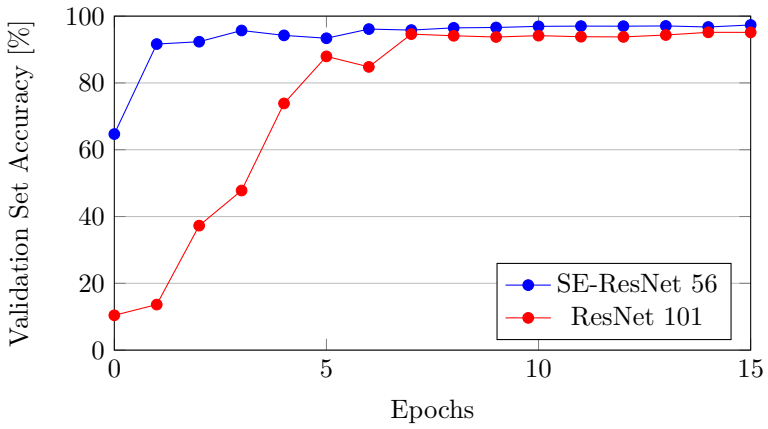

Figure 3: Comparison of Convergence Speeds of Se-ResNet 56 and ResNet 101

All models were run on NVIDIA RTX 2070 GPU.

6 Discussion

The SE-ResNet-56 performed the best among all the competing models with only nominal increases in computational costs (runtimes) relative to the ResNets. The results confirm the use of SE blocks as a very useful tool that is robust enough to be added to any model; which is very important given the sometimes overwhelming computational power required to train neural networks. In our analysis we use runtime as the metric of cost, since pragmatically we care most about which model converges fastest.

Despite the positive performance of the SE-ResNet, recent work shows that ResNets are diverse and redundant, leading to low-efficient modeling (Hu, Wen et al, 2018). They propose modifications to the SE-blocks where rescaling the value for each channel in this new structure will be determined by the residual and identity mappings jointly which expands the meaning of channel relationship modeling in residual blocks. Future work on the modified MNIST dataset could benefit from implementing this.

Throughout our experimentation, we notice that residual networks are resilient to overfitting, as supported by slight increases in validation error after every epoch, (or, rarely, insignificant decreases towards the end of the training process, or loss jumps towards the beginning). This could be understood in terms of residual connections in the network permitting layer skipping as it learns identity mappings between certain layers that through the backpropagation process, were understood to be unnecessary or of minimal contribution to the performance of the network. In a sense, our experiments confirms the hypothesis outlined in Hu et al. (2018) with some tolerance.

On the impact of thresholding the background, we hypothesize that the drop in validation accuracy is due to the image having lesser features to pick up and therefore overfitting to the training set, despite our previous remarks. Being a strong regularization technique, adding noise into the dataset in theory enables models to ignore lower level features such as pixel locations and focus on higher level features such as curvature and density. Therefore, as we thresholded the background, we removed that inherent regularization built into the dataset, which explains the drop in performance of our model.

We observe a slight drop in validation accuracy when comparing the ResNet-34 model pre-trained on ImageNet with the first 4 convolutional blocks frozen, and the original ResNet-34. We infer that the images are inherently different and therefore the picked up features from the convolutional blocks do not transfer well onto the modified MNIST dataset, due to size difference and different visual features.

7 Conclusion

In this paper, we employ several competing neural networks and find that SE-ResNets-34 produced the highest accuracy. We show empirically that ResNet's improve accuracy, and in conjunction with SE-blocks only increase computational costs nominally elative to models without SE-blocks. Evaluation on the modified MNIST dataset show that SE-ResNets can provide comparable accuracy at a lower computational costs compared to deeper models.

As for further investigation, there are several methods one could try in order to potentially score a higher accuracy. Generating more training samples using classical data augmentation techniques as well as using generative adversarial networks (GANs) could provide a slight increase, allowing deep models for better generalization and learn higher level features and their invariance under image transformations. One could also potentially look at other well known models in computer vision such as DenseNets (Gao Huang, et al., 2017), that would potentially perform just as competitively with roughly a third of the number of parameters as a regular ResNet.

8 Statement of Contributions

Viet wrote the code, Marcos and Raffi helped with the code. All three wrote the report.

References

Jia-Ren Chang and Yong-Sheng Chen. Batch-normalized maxout network in network. arXiv preprint arXiv:1511.02583, 2015

Diederik P. Kingma: "Adam: A Method for Stochastic Optimization", arXiv:1412.6980].

K. He and J. Sun. Convolutional neural networks at constrained time cost. In CVPR, 2015.

Gao Huang, et al. "Densely Connected Convolutional Networks", arXiv preprint arXiv:1608.06993, 2017

H.Jie, H.Li, S.Sun; Squeeze-and-Excitation Networks. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 7132-7141

H.Kaiming, Z.Xiangyu, R.Shaoqing, S.Jian. Deep Residual Learning for Image Recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015

K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In ICLR, 2015.

Li Wan, Matthew Zeiler, Sixin Zhang, Yann LeCun, and Rob Fergus. Regularization of neural networks using dropconnect. In Proceedings of the 30th international conference on machine learning (ICML), pp. 1058–1066, 2013.

Hu, Yang, et al. "Competitive Inner-Imaging Squeeze and Excitation for Residual Network." arXiv preprint arXiv:1807.08920 (2018)