Reproducing Analysis on Batch Size and Learning Rate for Model Generalization

Aly Elgharabawy* Michael Li* William Zhang*
Faculty of Engineering
McGill University, Montreal, Quebec
{aly.elgharabawy, er.li, william.zhang2}@mail.mcgill.ca

Abstract

Stochastic gradient descent (SGD) is a method commonly used in training neural networks. While SGD offers great flexibility in fine-tuning the optimization process, it can often lead to a tedious search of optimal hyperparameters, which include batch size and learning rate. These parameters can not only affect the performance of the model but can also greatly impact the amount of time necessary to train and test these models. In their paper titled "Control Batch Size and Learning Rate to Generalize Well: Theoretical and Empirical Evidence," Fengxian He, Tongliang Liu and Dacheng Tao outline a strategy in selecting an optimal batch size and learning rate in order to increase the generalization ability of the model. In order to reproduce their findings, we train VGG-19, ResNet-50, Xception and a custom Convolutional Neural Network with a set of batch sizes and learning rate. Through our result, we arrive at the same conclusion as He et al, demonstrating a positive relationship between the learning rate and test accuracy and a negative correlation between the batch size and the generalizability of neural networks. Finally, the previous conclusions prove that there exist a negative correlation between the ratio of batch size to learning rate and the test accuracy.

1 Introduction

Reproducibility is a growing issue in the modern Machine Learning community. There are many factors which contribute to this, most notably the high complexity of the models being reported and the sheer amount of computation power required to run these models. These factors make it difficult, if not nearly impossible, for researchers from other labs to reproduce and study these models. It is under this premise that the NeurIPS Reproducibility Challenge was made, with the goal of investigating the reproducibility of papers submitted to NeurIPS 2019.

The selected paper is "Control Batch Size and Learning Rate to Generalize Well: Theoretical and Empirical Evidence."[1] by Fengxiang He, Tongliang Liu and Dacheng Tao from University of Sydney. This paper offers both a theoretical proof and empirical results to demonstrate their claim which states that a low batch size to learning rate ratio is important in the generalization ability in a neural network model. In fact, it affirms that when SGD is employed to train deep neural networks, the batch size should not be too large, and the learning rate should not be too small in order to optimize its performance. In other words, the ratio between the batch size to the learning rate must be controlled in order to ensure generalizability of deep learning models. They have done so by proving a PAC-Bayes generalization bound for neural networks trained by SGD, which has a positive correlation with the ratio of batch size to learning rate. Furthermore, they showed that the generalization ability of deep neural networks has a negative correlation with the ratio of batch size to learning rate. This property builds the theoretical foundation of the training strategy. To validate their theory, they have conducted a large-scale experiment by training 1,600 models based on ResNet-110 [2], and VGG-19[3] architectures tested on both CIFAR-10 and CIFAR-100 datasets while strictly

* Equal contribution.

controlling unrelated variables such as momentum. Using the highest accuracy in the test set, the Spearman's rank-order correlation coefficients and the corresponding p values from 164 groups of data demonstrate that the correlation is statistically significant.[1]

2 Background and Related Work

The NeurIPS Reproducibility Challenge aims to encourage researchers in the Machine Learning field to ensure the reproducibility of their results and provide an opportunity for newcomers to contribute to research. The goal of this challenge is to identify which parts of the contribution can be reproduced, and at what cost in terms of resources. In fact, reproducibility is a growing issue in research. Consequently, new guidelines have been established to ensure reproducible results. Dodge et al. demonstrated that "test-set performance scores are insufficient to drawing accurate conclusions" concerning model performance.[4] His research demonstrates that the amount of computation used in the research has profound impact on the conclusions reached. They conclude by providing a set of best practices for presenting results for "robust future comparisons." Furthermore, Joelle Pineau also has a set of guidelines called "The Machine Learning Reproducibility Checklist," which ensure that researchers take a pro-active approach in ensuring that their research is reproducible.[5] In essence, the reproducibility movement has for objective to promote open research, where researchers across labs can quickly understand and integrate recent findings into their own work.

Stochastic gradient descent (SGD), an efficient optimization algorithm, has played a big part in the success of Deep Learning over the past decade. However, the optimal way of tuning hyper-parameters of neural networks to generalize remains an important question that is unresolved. A lot of work addressing strategies to tune hyper-parameters of neural networks has been done, however the influence of hyper-parameters on generalizability is still under debate. "Don't decay learning rate, increase batch size"[6] by Smith et al. shows that the same learning curve can be obtained by increasing batch size instead of decaying learning rate during training with SGD. In fact, an equivalent test accuracy can be reached after the same number of training epochs with fewer parameter updates. This leads to greater parallelism and shorter training times.

The paper by He et al.[1] demonstrates a strategy in setting the batch size and learning rate hyper-parameters in neural networks in order to optimize the accuracy. Masters and Luschi also find the presence of a relationship between batch size and learning rate, and conclude that for mini-batch stochastic gradient optimization, a batch size between 2 and 32 is most optimal.[1] While Masters and Luschi found that increasing the batch-size reduces the range of stable learning rates that will converge appropriately, Balles et al. and Smith et al. Both demonstrate that an approach of batch size increase during training in order to reduce the quantity of parameter updates and training time.[6,7] These methods are in constrast with previous approaches of setting an adaptive learning rate, as proposed by Ravaut and Gorti.[8]

These methods are based on training a model with stochastic gradient descent and all attempt to either decrease training time, increase accuracy, or both. The paper by He et al. is inline with these methods, and propose an approach utilizing a ratio of batch size to learning rate to effectively tune these hyperpameters.[1] In order to do so, the authors utilize both empirical and theoretical methods in order to prove their claim. The former is by training over 1600 neural networks based on the ResNet-110 and VGG-19 architecture on the CIFAR-10 and CIFAR-100 datasets, while the latter is done through a PAC-Bayes generalization bound, effectively demonstrating that there is a positive correlation between the generalization bound of the network and the ratio of batch size and learning rate.

3 Methodology

In order to validate the results obtained by He et al, we decided to conduct similar experiments. Although the code was not explicitly provided, they referenced code that implemented modified versions of VGG-19 and ResNet-110 architectures that are adapted to CIFAR-10 and CIFAR-100 datasets.

Initially, we attempted to reproduce these results with the Keras' VGG-19 implementation. However, due to poor weight initialization, we were unable to train those models with any learning rate greater

than 0.02. Therefore, we had to refer to code referenced by the paper[1] Since there is no additional information on the models architectures used by the authors, we assumed that they utilized the models in the repository without any major modifications. Since this code is from 2017, we had to fix numerous deprecated methods.

The experiments were conducted on ResNet-50, VGG-19, XCeption and a custom CNN on CIFAR-10 and CIFAR-100[2]. The default training and testing set split is used for our experiments.

Due to limited computation resources, it was impossible to attempt to reproduce all 1600 models ran by the paper. Therefore, we employed a strategy where we would run a set of batch sizes with a fixed learning rate, and a set of learning rates with a fixed batch size. While the authors tested their training strategy on over 1600 models from a set of 20 batch sizes, $S_{BS} = \{16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, 240, 256, 272, 288, 304, 320\}$ and 20 learning rates, $S_{LR} = \{0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.10, 0.11, 0.12, 0.13, 0.14, 0.15, 0.16, 0.17, 0.18, 0.19, 0.20\}$. Our testing methodology uses the following sets, which are $S_{BS} = \{16, 32, 64, 96, 128, 160, 192, 224, 256, 320\}$ with $LR = 0.01$ and $S_{LR} = \{0.01, 0.02, 0.03, 0.05, 0.75, 0.1, 0.125, 0.15, 0.175, 0.2\}$ with $BS = 64$. Hence, each model will be trained for exactly 200 epochs with a specific pair of batch size to learning rate described above. All batch sizes and learning rates are constant in our experiments. All unrelated parameters such as momentum, learning rate decay and dropout are disabled as the paper mentioned. This would allows us to investigate the relationships between accuracy, learning rate and batch size.

Initially, we ran VGG-19 and ResNet-50 for 50 epochs, after plotting the relationships we found that models with small learning rates did not converge. Hence, the number of iterations has been increased to 200 as He et al. used.

Finally, each architecture (ResNet-50, VGG-19, Xception and custom CNN) will have at least 20 models each representing a different batch size to learning rate combination. The best accuracy on the test set for each one of them is collected for analysis. This value expresses the generalization ability of each model, because the training error is always zero across all models. Three relationships are evaluated for each architecture: 1. Correlation between the generalization ability of neural networks and the batch size, 2. the correlation between the generalization ability of the neural network and the learning rate, 3. the correlation between the generalization ability of neural networks and the ratio of batch size to learning rate.

3.1 Models Tested

While He et al. test their claims on two architectures, ResNet-110 and VGG-19, this paper employs two more architectures in order to further investigate the generalizability of their training strategy. Four architectures were tested, namely ResNet-50[2], Xception[9], VGG-19[3] and a custom convolutional neural network. The ResNet-50 and Xception architectures consist of the exact implementation by Keras using pre-trained weights on the ImageNet dataset. The VGG-19 architecture has been modified to adapt to the CIFAR-10 and CIFAR-100 datasets. The ResNet-50 and VGG-19 implementations were experimented on both CIFAR-10 and CIFAR-100 whereas the custom Convolutional Neural Network (CNN) and Xception were only tested on CIFAR-10. We designed our custom CNN using a simple architecture to investigate whether their correlations would still be valid for a simple CNN architecture. Conversely, we chose Xception due to its depth wise separable convolutions, which assume independence in cross-channel and spatial correlation.[9] This wide selection in neural network architecture is key to determine how robust the relationship between model generalization ability with the ratio between batch size and learning rate. Figure 1 showcases the architecture of our CNN.

3.2 Training

All models use a stochastic gradient descent optimizer, with all hyperparameters kept constant except for batch size and learning rate. ResNet-50 and VGG-19 were trained for 200 epochs per chosen combination of batch size and learning rate, as done by He et al. in their paper. This is to ensure that the results replicate the original's as much as possible, and thus allowing a more rigid and sound

[1] Wei Yang, https://github.com/bearpaw/pytorch-classification, 2017.

[2] Available at https://www.cs.toronto.edu/ kriz/cifar.html

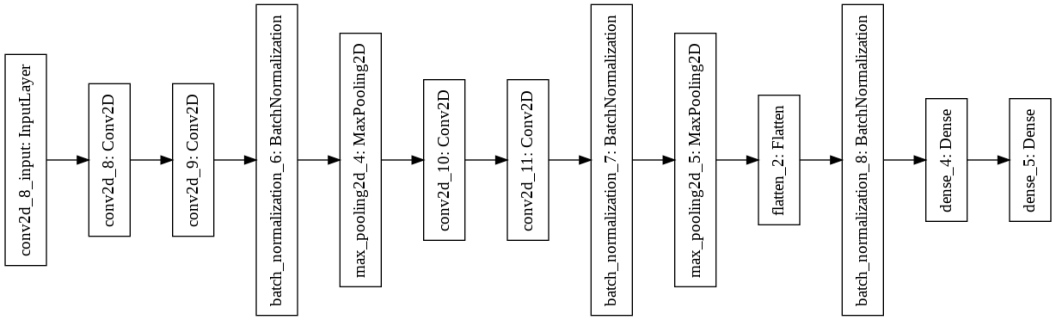

Figure 1: Architecture of our CNN

conclusion to be reached based on the tests. However, the training for both the custom CNN and the Xception model was done with only 50 epochs. The main reason for which was to save computation time to run as many tests as possible, as both models were verified to have successfully converged within 50 epochs. In addition, the custom CNN and Xception will only be trained on CIFAR-10, which is again due to the limited computational resources at our disposal. During the training of all the models, a logger was called at the end of each epoch to record relevant metrics at that point in training. Among the metrics recorded were: batch size, learning rate, training accuracy, training loss, validation accuracy and validation loss. These results were then compiled and analyzed, the results of which will be discussed in the following section.

Despite reducing the number of batch size and learning rate permutations, the computational load was high and had to be spread out. The main tools used are: Google Colaboratory[3], which utilize Tesla K80 GPus, Google Cloud[4], which utilized Tesla V100 GPUs, and CodeOcean[5], which also utilized Tesla K80 GPUs. All experiments were conducted over the length of 2 weeks.

4 Results and Analysis

After training all four models, we follow the procedure outline by He et al. and use the results to calculate the Spearman's rank-order correlation coefficients (SCCs) and the corresponding p value to determine the statistical significance of the correlation. Each table will outline either the relationship between batch size and test accuracy, with the learning rate fixed at a certain value or the relationship between learning rate and test accuracy with the batch size fixed at a certain value. For our analysis, an upper bound of 0.05 will be used for the p value (i.e. if p < 0.05 then the relationship is statistically significant). Below are the SCC ranges used to determine the type of correlation:

-1 < SCC < -0.5 Strong negative correlation

-0.5 < SCC < 0 Weak negative correlation

0 < SCC < 0.5 Weak Positive correlation

0.5 < SCC < 1 Strong Positive correlation

4.1 VGG-19

For the VGG-19 architecture, our results demonstrate that there is a negative correlation between batch size and test accuracy (SCC = -0.988 when LR = 0.01 in Table 1). This relationship is statistically significant, as the corresponding p value (9.31×10^{-8}) is indeed less than 0.05. A non-statistically significant strong negative correlation is demonstrated when the CIFAR-100 dataset is used (SCC = -0.829, p=4.15×10^{-2},LR = 0.10 in Table 1). The results obtained support the conclusion made in the original paper that there is a negative correlation between batch size and peak test accuracy, and therefore generalization ability. Based on these results,we have been able to replicate the results obtained in the original paper.

[3]https://colab.research.google.com/

[4]https://cloud.google.com/

[5]https://codeocean.com/

Furthermore, there is a weak positive correlation between the learning rate and test accuracy, as demonstrated by the SCC value (0.115) in Table 2 when the batch size is fixed at 64. This relationship is also non-statistically significant, as the corresponding p value ($0.751 > 0.05$, BS = 64, Table 1). As for CIFAR-100, a very weak positive non-statistically significant correlation (SCC = 0.059, p = 0.863 > 0.05, BS = 64 Table 1) is observed between learning rate and peak test accuracy. The weakness of the correlation prevents us from drawing a conclusion on the data. Therefore, we have not been able to replicate the results from the original paper.

However, upon comparing our graph plotted with test accuracy against batch size to learning rate ratio (Figure 2) to the original paper's, we observe a very significant similarity in the plots. In that regard, we have been able to replicate the paper's results in showing a negative correlation between test accuracy (which is analogous to generalization ability) and the batch size to learning rate ratio.

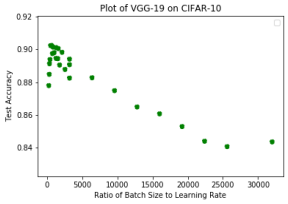

Figure 2: Test accuracy to ratio of batch size to learning rate for VGG-19 and CIFAR-10

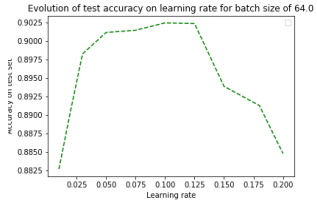

Figure 3: Test accuracy for various learning rate and BS=64 for VGG-19 and CIFAR-10

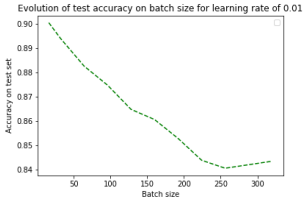

Figure 4: Test accuracy fir various batch size and LR=0.01 for VGG-19 and CIFAR-10

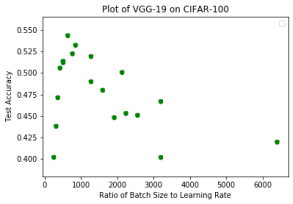

Figure 5: Test accuracy to ratio of batch size to learning rate for VGG-19 and CIFAR-100

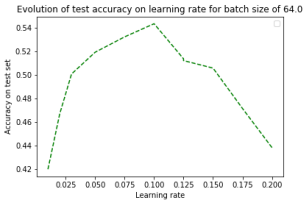

Figure 6: Test accuracy for various learning rate and BS=64 for VGG-19 and CIFAR-100

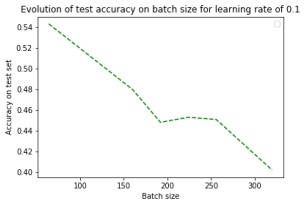

Figure 7: Test accuracy fir various batch size and LR=0.01 for VGG-19 and CIFAR-100

Table 1: Test accuracy of VGG-19 with different batch sizes

BS	VGG-19 on CIFAR-10		VGG-19 on CIFAR-100	
	SCC	p	SCC	p
64	0.115	0.751	0.059	0.863
LR				
0.01	-0.988	9.31×10^{-8}	–	–
0.10	–	–	-0.829	4.15×10^{-2}

4.2 ResNet-50

For the ResNet-50 architecture on CIFAR-10, our results demonstrate that there is a positve correlation between batch size and test accuracy for a learning rate equal to 0.1. The SCC coefficient is 0.915 with a p value of 2.04×10^{-4} which means that the relationship is statistically significant: see Table 2. Furthermore, there is a negative correlation between the learning rate and the generalizability of the models when the batch size is 64. The SCC coefficient is 0.115 and the p value is 0.751 which means that the relationship is not statistically significant (see Table 2). However, these two results

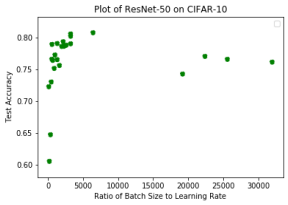

Figure 8: Test accuracy to ratio of batch size to learning rate for ResNet50 and CIFAR-10

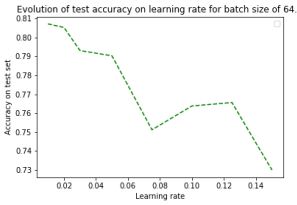

Figure 9: Test accuracy for various learning rates and BS=64 for ResNet-50 and CIFAR-10

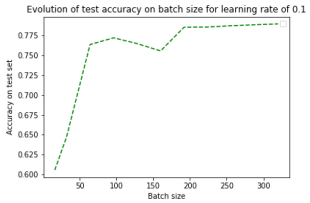

Figure 10: Test accuracy for various batch sizes and LR=0.01 for ResNet-50 and CIFAR-10

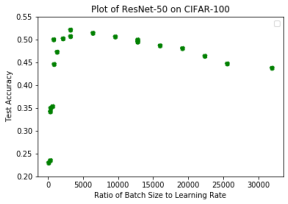

Figure 11: Test accuracy to ratio of batch size to learning rate for ResNet50 and CIFAR-100

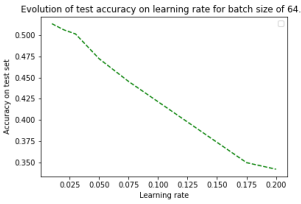

Figure 12: Test accuracy for various learning rates and BS=64 for ResNet-50 and CIFAR-100

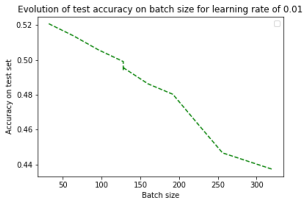

Figure 13: Test accuracy for various batch sizes and LR=0.01 for ResNet-50 and CIFAR-100

go against the conclusion of He et al. which affirms that the correlation between batch size to test accuracy and learning rate to test accuracy should be negative and positive respectively. Finally, the scatter plot showing the correlation between the test accuracy and the ratio of batch size to learning rate shows that it is not negative. Instead, the relationship forms a logarithmic curve. In addition, there exist a ratio around 5000 where the generalizability of the neural network is best according to our results (see Figure 8). Hence, we are unable to reproduce any of He et al.'s three conclusions with ResNet-50 on the CIFAR-10 dataset.

For the ResNet-50 architecture on CIFAR-100, our results are similar to the ones on the CIFAR-10 dataset. There exist a negative correlation between learning rate and test accuracy (see Figure 12), and for the batch size to test accuracy. Hence, the second statement aligns with He et al.'s conclusion saying that there exists a negative correlation between the batch size and the generalizability of neural networks (see Figure 13). This relationship is statistically significant with a p value of 3.84×10^{-9} (see Table 2) and a SCC of -0.991. Finally, the correlation between the ratio of batch size to learning rate and the test accuracy forms a logarithmic curve (see Figure 11) which does not support He et al.'s conclusion. Hence, when the ratio is too small, the neural network does not generalize well. Moreover, ResNet-50 performance peaks when the ratio is around 5000. Therefore, this leads to the hypothesis that there exists a lower bound for the batch size to learning rate ratio from which He et al.'s conclusion holds.

Table 2: Test accuracy for ResNet-50 on various batch size(BR) and learning rate(LR)

BS	ResNet-50 on CIFAR-10		ResNet-50 on CIFAR-100	
	SCC	p	SCC	p
64	-0.905	2.01×10^{-3}	-1	0
LR				
0.01	-0.6	2.08×10^{-1}	-0.991	3.84×10^{-9}
0.1	0.915	2.04×10^{-4}	–	–

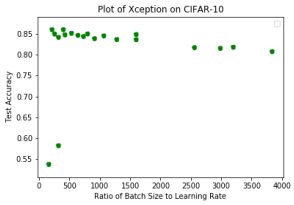

Figure 14: Test accuracy to ratio of batch size to learning rate for Xception

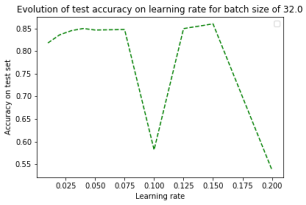

Figure 15: Test accuracy for various learning rate and BS=32 for Xception

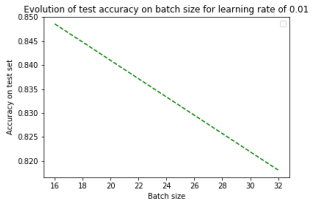

Figure 16: Test accuracy for various batch size and LR=0.01 for Xception

4.3 Xception

When analyzing our results with the Xception model, we observe a non-statistically significant weak positive correlation between learning rate and peak test accuracy (SCC = 0.115, p = 0.751 > 0.05, BS = 32 Table 3). This weak correlation cannot be concluded upon, and we therefore cannot make a correlation between learning rate and generalization ability in this case. In addition, the opposite type of correlation was shown. Therefore, we could not replicate the original paper's results for Xception in this regard.

As for the batch size - test accuracy correlation, we observe a statistically significant very strong negative correlation between batch size and test accuracy. This is in line with the conclusion made in the paper that the test accuracy has a negative correlation with batch size (as observed in Figure 16). Due to the strength of the correlation, we can make the same conclusion. Therefore, we have been able to replicate the paper's results.

Figure 14 shows our plot for test accuracy against batch size to learning rate ratio. It qualitatively shows a very weak negative correlation between test accuracy and batch size to learning rate ratio. Though the same type of relationship is observed in our graph, it is not in the same strength. This might be due to Xception's unique architecture giving it the ability to generalize well partially regardless of the batch size and learning rate values used. This can be seen in Figure 14, as the peak test accuracy only slightly fluctuates when the batch size and learning rate values significantly change.

Table 3: Test accuracy for Xception with different batch sizes

BS	Xception on CIFAR-10	
	SCC	p
32	0.115	7.51×10^{-1}
LR		
0.01	-0.99	0

4.4 Custom CNN

The results of the tests performed on our custom CNN without momentum demonstrate a statistically significant strong positive correlation (SCC = 0.83 ,p = 2.94×10^{-3}<0.05, BS = 96 from table 4) between learning rate and peak test accuracy. Conversely, there is a non-statistically significant strong negative correlation (SCC = -0.548 ,p = 1.6×10^{-1}>0.05, LR = 0.01 from table 4) between batch size and peak test accuracy. The results observed are in line with the the findings in the original paper, as Figure 18 does demonstrate a significant positive correlation between test accuracy and learning rate. Therefore, we were able to replicate this conclusion from this model.

As for the batch size and test accuracy, a non-statistically significant moderately strong negative correlation is observed (SCC = -0.548, p = 0.160>0.05). Despite the similarity of the correlation, we were not able to make a statistically significant conclusion, and have only been able to partially replicate the conclusion.

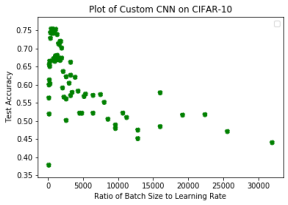

Figure 17: Test accuracy to ratio of batch size to learning rate for custom CNN (no momentum) and CIFAR-10

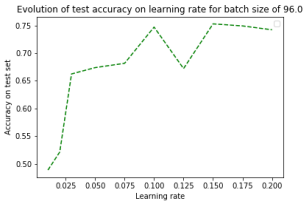

Figure 18: Test accuracy for various learning rates and BS=96 for custom CNN (no momentum) and CIFAR-10

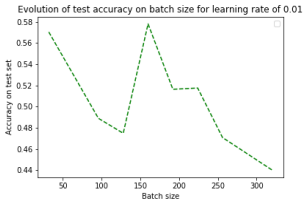

Figure 19: Test accuracy for various batch sizes and LR=0.01 for custom CNN (no momentum) and CIFAR-10

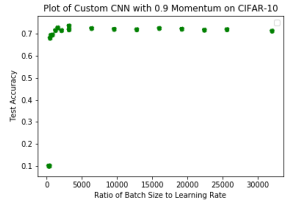

Figure 20: Test accuracy to ratio of batch size to learning rate for custom CNN (with momentum) and CIFAR-10

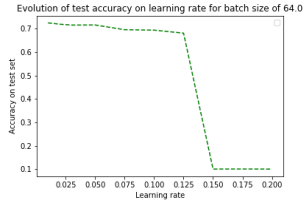

Figure 21: Test accuracy for various learning rates and BS=64 for custom CNN (with momentum) and CIFAR-10

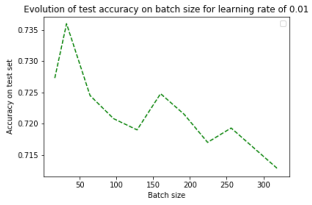

Figure 22: Test accuracy for various batch sizes and LR=0.01 for custom CNN (with momentum) and CIFAR-10

As for the custom CNN with momentum, a statistically significant strong negative correlation is observed between test accuracy and learning rate (SCC = -0.976 ,p = 1.46×10^{-6}<0.05, BS = 64 from table 4). This result does not match the one from the paper, as a positive correlation is concluded instead. Therefore, we could not replicate this conclusion from this model.

As for the batch size and test accuracy, a statistically significant strong negative correlation is observed (SCC = -0.79 ,p = 6.1×10^{-3}<0.05, LR = 0.01 from table 4). This conclusion supports the one arrived at in the paper, and it is statistically significant. Therefore, for this model, we have been able to replicate the conclusion in the paper.

However, an interesting result is observed upon the addition of momentum, as the model achieves almost the same peak test accuracy for all batch size and learning rate combinations tested. This conclusion may be an explanation as to why optimizers that use momentum have garnered in popularity recently, as they can allow a model to converge to a satisfactory test accuracy with a lower dependency on the batch size and learning rate values used. This result is displayed in the scatter plot in Figure 20. Though the almost non-existent correlation observed in Figure 20 does not match the one arrived at in the paper for this model, and the conclusion could not be replicated in this case.

Table 4: Test accuracy for custom CNN on different batch size(BS)

BS	Custom CNN without momentum		Custom CNN with momentum	
	SCC	p	SCC	p
64	–	–	-0.976	1.46×10^{-6}
96	0.83	2.94×10^{-3}	–	–
LR				
0.01	-0.548	1.60×10^{-1}	-0.79	6.10×10^{-3}

5 Conclusion

We were able to partially reproduce the correlation found by He et al. with both the VGG-19 and our custom CNN architectures. However, we found that there was a lower-bound for which He et al.'s third conclusion (negative correlation between generalizability and ratio of batch size to learning rate) holds for different architectures. If the ratio is too small, the model becomes unable to train effectively given the amount of epochs (200) and fails to converge. In fact, all models failed to demonstrate that there exists a positive correlation between the learning rate and the test accuracy. Furthermore, we were unable to entirely reproduce the results with ResNet-50, Xception and our CNN with momentum. Our custom model with momentum seems to increase the amount of "acceptable" batch size and learning rate combinations and train effectively with a wide range of ratios. Overall, the results could be partially replicated for VGG-19 and the custom CNN without momentum on CIFAR-10. However, the empirical evidences are not strong enough to conclude taht it can be generalized to other models trained on SGD.

These results demonstrate the importance of hyperparameter optimization and the magnitude of their impact on training an effective model. It would be important to investigate the impact of batch size for other optimizers such as Adam or Adagrad. Furthermore, our results were only tested for convolutional neural networks. Therefore, testing these training strategies on other neural networks such as Long Short Term Memory could yield different results.

6 Contribution

- Aly Elgharabawy: Model implementation, training and documentation
- Michael Li: Model implementation, training, result analysis and documentation
- William Zhang: Model implementation, training and documentation

References

[1] F. He, T. Liu, and D. Tao, "Control Batch Size and Learning Rate to Generalize Well: Theoretical and Empirical Evidence," pp. 1141–1150, 2019.

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

[3] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," arXiv 1409.1556, 09/04 2014.

[4] J. Dodge, S. Gururangan, D. Card, R. Schwartz, and N. A. Smith, "Show Your Work: Improved Reporting of Experimental Results," 2019. Available: https://doi.org/10.18653/v1/D19-1224

[5] J. Pineau. (2019). The Machine Learning Reproducibility Checklist. Available: https://www.cs.mcgill.ca/ jpineau/ReproducibilityChecklist.pdf

[6] S. Smith, P.-J. Kindermans, and Q. Le, "Don't Decay the Learning Rate, Increase the Batch Size," 11/01 2017.

[7] L. Balles, J. Romero, and P. Hennig, "Coupling Adaptive Batch Sizes with Learning Rates," 12/15 2016.

[8] M. Ravaut and S. Gorti, "Gradient descent revisited via an adaptive online learning rate," 01/27 2018.

[9] F. Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions," in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 1800-1807.