COMP 551 - MiniProject 4: Reproducibility in ML

Tyler Watson
260867260

Anthony Porporino
260863300

Aaron Lohner
260863277

December 13th, 2021

Abstract

In the paper "Are wider nets better given the same number of parameters?", the authors claim that when observing increased accuracy in wider networks, width is the key factor as opposed to the increased number of weights [1]. Their results indicate that widening both MLP and ResNet models, while keeping the number of weights constant, increases performance. Our project attempts to reproduce these results and further investigate the general claims made in the paper.

1 Introduction

For this project, we conducted several experiments based on a paper that focuses on answering the following question: "Is the observed improvement [in a wider model] due to the larger number of parameters, or is it due to the larger width itself?" [1]. In particular, we tried to reproduce the paper's results, and investigate whether their claims could be generalized. Our results showed that (1) their experiments appear to be reproducible, (2) the claims about the MLP model can generalize to a new dataset, and (3) the claims do not generalize well on a new model.

2 Paper Analysis

2.1 Summary

The paper under investigation studies the impact of model width on model performance by observing image classification accuracies of two models on various datasets. To change the model width without increasing the parameters, they attempted three strategies of which static sparsity was the most efficient. To do this, a baseline model with dense weight tensors is widened by a predetermined factor and then sparsified until the number of weights equals that of the original baseline model. This method allows for the width to be increased while keeping number of parameters constant without changing the network's architecture by simply reducing the model's connectivity. The widening factor is the ratio between the new width and the baseline width. The authors define connectivity of a sparse model as "the ratio between the number of its parameters and the number of parameters in a dense model of the same width" [1]. They find that for experiments on both MLP (one hidden layer) and ResNet18, the best results are achieved when connectivity is high in the last layer, so fewer weights should be removed from this layer than others when sparsifying the model. They conclude that, while fixing the number of parameters and increasing width, there is a marked improvement in performance up to a certain widening factor, which depends on the model and the classification task. Furthermore, they attribute this improvement more to width than to increased parameters by comparing sparse models to dense ones of equal width.

2.2 Investigated Claims

The primary claim we investigate in this project is that "model performance can be improved by increasing the width, without increasing the number of parameters" [1]. Another claim is that this can be generalized to other models and datasets. These two claims drove the experimentation design.

3 Results

3.1 Exp 1: Results reproduction

Our first experiment attempted to reproduce the results for the MLP model on the MNIST dataset. It used the default batch size of 100, and a learning rate of 0.1. Figure 1 shows the results of our experiments when using ReLU activation trained on 300 epochs. Figure 4 in the Appendix shows the same results for Linear activation. Both show increased performance with increased width (lower connectivity).

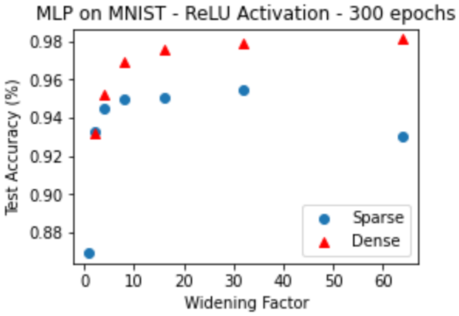

Figure 1: MLP network trained on 300 epochs with ReLU activation function

The baseline model had a width of 5 and various widening factors were used to increase width. The table also shows accuracies for the respective dense models which unlike the sparse data points all contain different number of weights. The connectivity of the last layer remained constant at 1 so only the weights of the middle layer were decreased. The results obtained are similar to that of the paper in that the accuracy of the model increases with width while keeping the number of weights constant. We also see that most of the improvement is due to the increase in width rather than number of weights.

We also attempted to reproduce the results from the ResNet18 experiments of the paper on three datasets: CIFAR10, CIFAR100 and SVHN. Since this model was more complex we were not able to train on the full 300 epochs and only trained on 50 epochs. We ran these experiments with starting base widths of 8 and 12 using the default hyperparameters. The results of this experiment, shown in the Appendix Figure 5, indicate that a wider model increases accuracy because for all experiments in this section, the model with the highest accuracy was always a wider model than the baseline model. However, our experiments also contain many examples where increasing the widening factor resulted in a decreased accuracy as well.

3.2 Exp 2: New dataset

Our second experiment attempted to show that this main claim generalizes to a new dataset. We trained the MLP model on the fashion MNIST dataset [2] for 50 epochs and results are shown in Figure 2. Although the results still indicate that the denser model performed better, we still see that most of the improvement is shown in simply having a larger width.

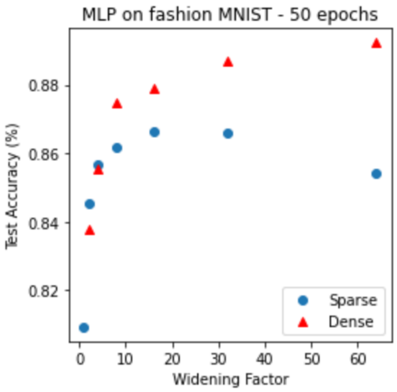

Figure 2: MLP network trained on 50 epochs on the Fashion MNIST dataset

3.3 Exp 3: New model

The third experiment we conducted aimed to verify the claims by using a new model on a base width of 8 for 20 epochs on the CIFAR100 dataset. For simplicity's sake and ease of integration, an SENet model was taken from the same repository as the ResNet18 model. SENet is a convolutional neural network architecture that "employs squeeze-and-excitation blocks to enable dynamic channel-wise feature recalibration" [3]. Besides those pre-activation blocks, it is quite similar to the ResNet18 architecture, therefore we expected to see very similar results for both. However, this was not the case. As seen in Figure 3 below, when width is increased, SENet drops in both the training and testing accuracy.

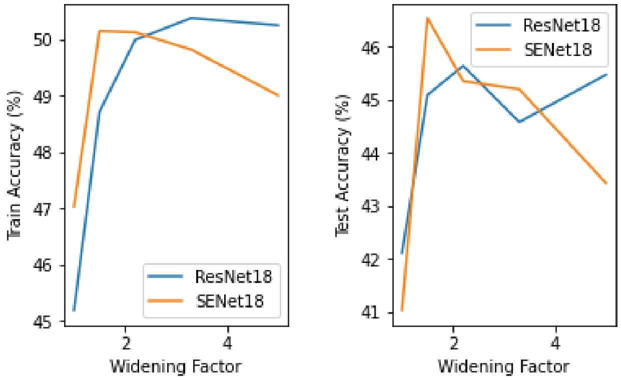

Figure 3: Train and test accuracy of ResNet18 and SENet18 trained on 20 epochs

4 Discussion

The results from our experiments on the MLP model match the results from the paper quite closely and the main claim generalizes to a new dataset. The results for our ResNet18 model experiments do show that accuracy can improve by widening the model, however do not directly match the results from the paper. It is likely that the reduced number of epochs may have skewed these results and this is why they differ from the paper. Despite this fact, we still observed that widening a network, while keeping the number of weights the same, can improve performance, depending on the widening factor.

The results obtained in the third experiment did not quite align with the claim that this would generalize to other models. As seen in Figure 3, widening the SENet width seemed to lead to decreased performance in the long run. This may indicate that their claims are only true for certain model architectures and that other architectures suffer when width is increased. However, this conclusion comes again with the caveat that only 20 epochs were run which may not be indicative of a model that is trained on somewhere in the range of 300 epochs.

5 Reproducibility Details

The only tweeks made were to fix a minor bug in the MLP code involving Cuda and moving a folder within the ResNet folder. Besides that, everything was easily reproduced.

6 Challenges

The majority of the challenges faced were related to access to resources to be able run our experiments. We lacked both time and computational power to obtain the results that we initially intended to produce. Although we had access to GPUs through Google Colab and a McGill SOCS server, we found our Google Colab sessions ending due to excessive GPU usage and the McGill server (@open-gpu-(1-32).cs.mcgill.ca) sometimes killed the bash process that was running our experiments. This was very frustrating since we would need to check the output folder to see at what point the process was killed and re-run the experiments that did not complete training. The time it took to run the ResNet18 model was also extremely large. We found out early on that running this model on 300 epochs was not feasible using our available resources. Our solution was to run the model on only 50 epochs and sometimes 20. It is worth noting that even these experiments sometimes took half a day to run due to the complexity of the model and the number of experiments run.

7 Conclusion

7.1 Key Takeaways

Overall, this paper was well written and relatively clear and easy to understand. The code that was provided was organized quite well and made it simple to reproduce. This showed us that reproducibility is extremely important when credibility is called into question and the simpler a result is to reproduce, the more credibility it will hold. Furthermore, this paper demonstrated that is in unfeasible to cover all use cases of a certain claim and that blanket statements should not be made if only a couple experiments were completed. Finally, this project further emphasized that ReadMe files are crucial when exposing a code repository to the public and the more detailed one can make this file, the better.

7.2 Future Investigation

There remain many avenues one could follow in order to further investigate the claims from the paper. The first would be to repeat the experiments we performed with more computational power and time to see if the results hold up when the ResNet18 model is trained on hundreds of epochs as opposed to only 20 or 50. Furthermore, more new datasets and models could be investigated to test the claims as well. This may require additional work to ensure the model follows the same structure as is required by the rest of the codebase. Finally, this problem could be extended to another input data medium such as sound in order to verify if this phenomenon is present in other domains as well.

References

[1] A. Golubeva, B. Neyshabur, and G. Gur-Ari, *Are wider nets better given the same number of parameters?*, 2021. arXiv: 2010.14495 [cs.LG].

[2] H. Xiao, K. Rasul, and R. Vollgraf, *Fashion-mnist: A novel image dataset for benchmarking machine learning algorithms*, cite arxiv:1708.07747Comment: Dataset is freely available at https://github.com/zalandoresearch/fashion-mnist Benchmark is available at http://fashion-mnist.s3-website.eu-central-1.amazonaws.com/, 2017. [Online]. Available: http://arxiv.org/abs/1708.07747.

[3] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, *Squeeze-and-excitation networks*, 2019. arXiv: 1709.01507 [cs.CV].

A Appendix

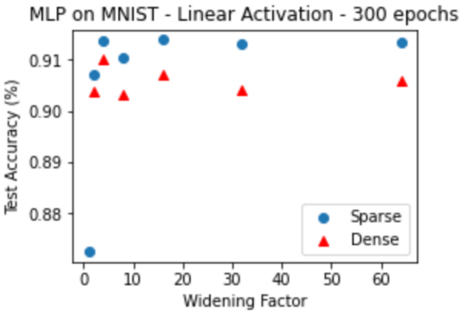

Figure 4: MLP network trained on 300 epochs with Linear Activation

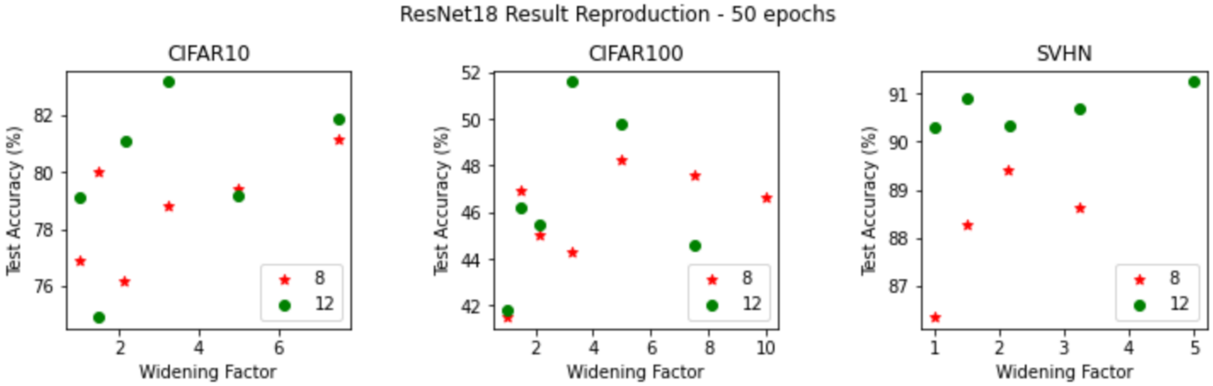

Figure 5: ResNet18 on 50 epochs