Final Project Report

Omri Zeevy - 313327041 , Uriel Zaed - 319435608

July 20, 2023

Abstract

Deep convolutional neural networks have brought significant advancements in various domains, such as image classification and detection. This project explores novel extensions of the GoogleNet architecture with a focus on enhancing the model's learning rate parameter. By investigating different learning rate schedules, such as Cyclical Learning Rates, Learning Rate Warm-up, Learning Rate Annealing, One-cycle Learning Rate Policies, and Uneven Cyclical Learning Rate, we aim to improve the generalization capabilities of the model. Tested on the CIFAR-10 data set, the experiments demonstrate substantial enhancements in the model's performance, leading to improved generalization. These findings have implications for optimizing deep learning models' performance and advancing their effectiveness in practical applications. Link to: Git.

1 Introduction

Deep learning has significantly impacted various fields, leveraging its potential in image recognition, natural language processing, and various other tasks [1]. It involves training complex neural networks on large-scale data sets and defining their architectures to learn hierarchical representations of data. Convolutional neural networks (CNNs), exemplified by the GoogleNet architecture, excel in tasks of image classification and recognition and are suitable for visual data analysis [2].

GoogleNet is the name of a deep convolutional neural network architecture (code-named Inception), which is known for revolutionizing image classification and detection in the ILSVRC14. A distinguishing feature of this architecture is its remarkable ability to maximize computational resources within the network.

The GoogleNet architecture is based on inception, and its development was motivated by the need to improve the performance of deep neural networks while addressing the limitations of uniformly increased network size. Increasing the depth and width of networks seemed like an easy way to achieve higher-quality models, but it also led to increased overfitting risk, especially when labeled training data was limited. Additionally, larger networks demanded more computational resources, making efficiency a crucial concern [7]. Through iterative refinement and tuning, the Inception architecture showed promising results, especially in the context of localization and object detection tasks.

The learning rate is a crucial hyper-parameter that influences the training process of deep learning models. It determines the step size during gradient descent and affects convergence speed and generalization performance. Selecting an appropriate learning rate presents challenges, as a high value may cause instability or overshooting, while a low value can lead to slow convergence or getting stuck in local optima.

Besides tuning the learning rate in traditional ways to improve the performance of a model, one can use different learning rate schedules to do so. Learning rate schedules play a vital role in optimizing the training process of deep learning models. Each learning rate schedule offers unique benefits to address specific challenges. By understanding and utilizing these various learning rate schedules, researchers and practitioners can optimize the performance of deep learning models, ultimately leading to more efficient and effective training processes for a wide range of tasks.

This project aims to explore innovative extensions of the GoogleNet model, focusing on optimizing the learning rate parameter. In order to improve the model's convergence speed and generalization capabilities, we investigate five learning rate schedules: Cyclical Learning Rates, Learning Rate Warm-up, Learning Rate Annealing, One-cycle Learning Rate Policies, and Uneven Cyclical Learning Rate. Through extensive experimentation, we aim to identify how each learning rate tuning strategy influences the performance of the GoogleNet model.

2 Details of Proposal

Five techniques regarding the learning rate were examined throughout our research. First, we focused on the Cyclical Learning Rate which is a technique of implementing a cyclical learning rate schedule instead of using a fixed learning rate throughout training. These schedules periodically change the learning rate between a minimum and maximum value, enabling the model to explore different regions of the loss landscape. Cyclical learning rates have been shown to improve classification accuracy without a need to tune and often converged in fewer iterations [3].

Next, we delved into the concept of Learning Rate Warm-up. This approach entails a gradual increment of the learning rate during the initial stages of training. Its primary objective is to prevent abrupt and potentially destabilizing gradient updates that can occur at the start of the training process [4]. With the gradual increase in the learning rate, the model can attain a better balance between effective learning and stability, ultimately enhancing its overall performance.

Our exploration continued with the technique of Learning Rate Annealing. Learning Rate Annealing is a sophisticated approach that involves a gradual reduction of the learning rate over time, mostly by a fixed factor or a predetermined schedule. This strategy allows the model to make substantial early updates and then finely adjust its parameters throughout the training process. The shrinkage of the learning rate reduces the stochastic noise and helps avoid oscillation near the optimal point [5]. As a result, the model gains the ability to avoid local minima, making it possible to discover superior solutions during optimization.

Subsequent to the examination of the aforementioned methodologies, we proceeded to investigate the novel approach of One Cycle Learning Rate. Integrating essential aspects of cyclical learning rates, this technique involves a gradual rise of the learning rate to its maximum, followed by a decrease to its minimum value, resulting in a single cycle. This strategy has been proven to be efficient in achieving rapid convergence towards optimal solutions, while concurrently improving the model's capacity for generalization. For example, it has been shown that a gated recurrent unit (GRU) trained with the one-cycle policy can further improve the performance of a two-hidden-layer GRU [6].

Finally, we delved into the technique known as Uneven Cyclical Learning Rate. This approach involves periodical cycles in which the learning rate reaches its maximum in fewer iterations than a standard cycle approach. By doing so, the model gains the ability to escape local minima, potentially leading to the discovery of better solutions. With each periodic cycle of the learning rate, the model can effectively explore different regions of the loss landscape, thereby resulting in improved overall performance.

3 Results

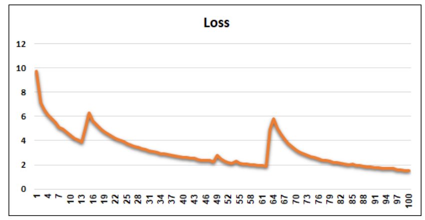
In this research, we explored the efficacy of five learning rate scheduling techniques in order to enhance the performance of the standard GoogleNet model. To assess the impact of these techniques, we compared the results obtained from each method to the performance of the original GoogleNet model. In order to do so, we used GridSearch to examine the best combination of parameters for each of the techniques. After finding the best parameters, we used different types of metrics to evaluate each technique's performance compared to the standard GoogleNet model.

In our GoogleNet standard model, we used a learning rate of 0.1, a batch size of 128, and 100 epochs. The optimization method was stochastic gradient descent with a momentum parameter of 0.9. In all of the examined techniques, we also used batch sizes of 128 and 100 epochs so that we'll be able to compare the effect of the learning rate in each technique. It is important to note that all the models were tested on the CIFAR-10 data set. In this standard model, we get 78% accuracy on the test set.

The results of Cyclical Learning Rates yielded a notable improvement in accuracy compared to the standard GoogleNet model. The parameters that were tuned in this method were the base learning rate (e.g. the minimum value of the learning rate) and the maximum learning rate. Furthermore, we trained our model using the "triangular2" mode, which means that for every cycle the maximum learning rate was divided by 2. Specifically, the CLR model with the best parameters according to the GridSearch (minimum learning rate of 0.01,

category	precision	recall	f1-score	support
airplane	0.78	0.83	0.81	1000
automobile	0.85	0.91	0.88	1000
bird	0.77	0.66	0.71	1000
cat	0.6	0.59	0.6	1000
deer	0.71	0.81	0.76	1000
dog	0.69	0.67	0.68	1000
frog	0.81	0.84	0.83	1000
horse	0.84	0.79	0.82	1000
ship	0.88	0.85	0.87	1000
truck	0.85	0.83	0.84	1000
accuracy			0.78	10000
macro avg	0.78	0.78	0.78	10000
weighted avg	0.78	0.78	0.78	10000

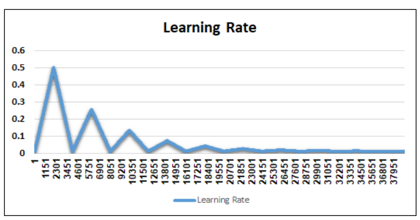
(a) Classification performance



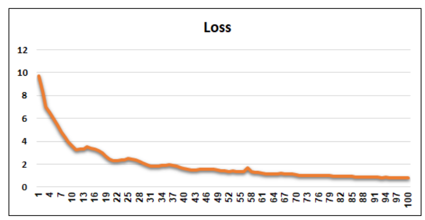
(b) Model's loss

Figure 1: Standard GoogleNet performance and loss

maximum learning rate of 0.5) achieved an accuracy that was 3% higher than the standard model, indicating the effectiveness of this learning rate schedule in guiding the model to explore different regions of the loss landscape.



(a) Learning rate per iteration



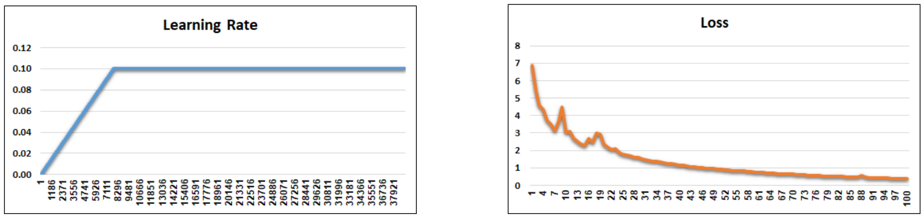
(b) Loss per epoch

Figure 2: Cyclical Learning Rates

Second, the implementation of the Learning Rate Warm-up method resulted in significant performance enhancements for the model. We tuned the warm-up epochs parameter according to the GridSearch. We allowed the learning rate to increase during 20% of the epochs until it reached its maximum value and then remained constant for the remaining epochs during training. As a result, the model trained with LRW achieved outstanding accuracy, surpassing the standard GoogleNet model by an impressive 7%. The gradual learning rate increase during the initial training phase effectively prevented large gradient updates, ensuring a more stable and efficient learning process from the beginning.

Additionally, we investigated the impact of the Learning Rate Annealing method on the performance of the GoogleNet model. The results revealed that this approach achieved slightly better accuracy compared to the standard GoogleNet model.

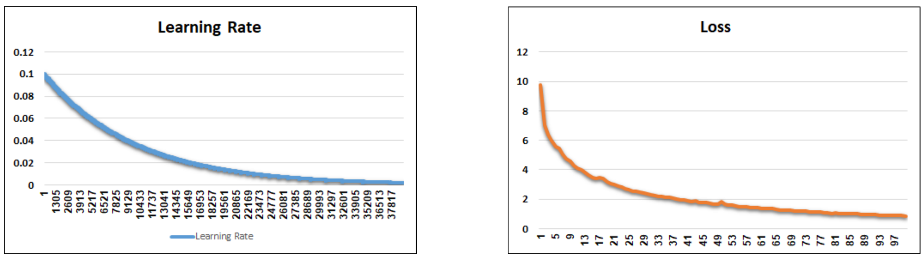
Furthermore, the results of the One-cycle Learning Rate method revealed a



(a) Learning rate per iteration

(b) Loss per epoch

Figure 3: Learning Rate Warm-up

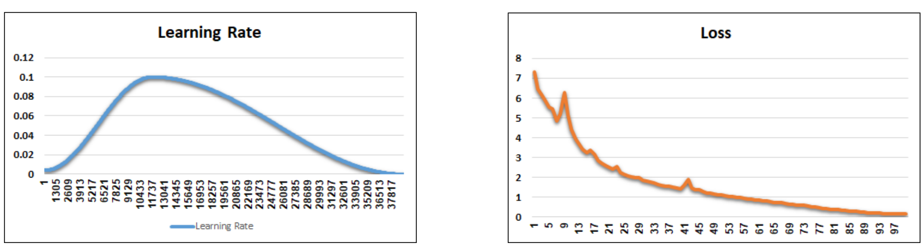


(a) Learning rate per iteration

(b) Loss per epoch

Figure 4: Learning Rate Annealing

significant improvement in accuracy when employing this approach, surpassing the standard GoogleNet model by a remarkable 8%. By utilizing a one-cycle policy that involves increasing the learning rate to a maximum value and subsequently decreasing it back to a minimum value, the model exhibited improved generalization capabilities.

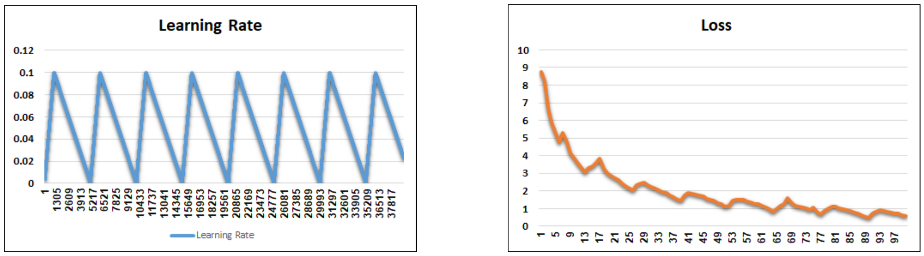


(a) Learning rate per iteration

(b) Loss per epoch

Figure 5: One Cyclical Learning Rate

Eventually, we delved into the impact of the Uneven Cyclical Learning Rate on the performance of the GoogleNet model. The findings of this analysis revealed noteworthy enhancements in accuracy, with this approach surpassing the standard GoogleNet model by 5%. One of the key factors contributing to this improvement is that for each cycle, for the first 20% iterations, the learning rate increases until it reaches its maximum value, and then gradually decreases to its minimum value and starts over again. This enables the model to explore diverse regions within the loss landscape.



(a) Learning rate per iteration

(b) Loss per epoch

Figure 6: Uneven Cyclical Learning Rate

4 Conclusions

model	accuracy	precision	recall	f1-score
Standard model	0.78	0.78	0.78	0.78
Cyclic	0.81	0.82	0.81	0.82
Warm-up	0.84	0.84	0.84	0.84
Annealing	0.78	0.78	0.78	0.78
One cycle	**0.85**	**0.85**	**0.85**	**0.85**
Uneven Cyclic	0.83	0.83	0.83	0.83

Figure 7: Results Summary

Among the learning rate schedule techniques explored, One-cycle Learning Rate exhibited the highest accuracy of 0.85, outperforming other methods. Learning Rate Warm-up achieved an accuracy of 0.84, showcasing stable learning and effective parameter initialization. However, Cyclic Learning Rates, with an accuracy of 0.81, and Uneven Cyclic Learning Rate with an accuracy of 0.83, demonstrated slightly lower accuracy scores. This can be attributed to their unstable learning rates throughout training, potentially affecting convergence. In contrast, the warm-up and one-cycle methods began with low learning rates that increased and stayed stable throughout training, which likely contributed to their slightly higher accuracy scores. The Learning Rate Annealing method almost didn't improve the model's accuracy, which maybe is a result of insufficient parameters range search. These findings emphasize the importance of learning rate schedules in optimizing deep neural network performance and striking the delicate exploration-exploitation balance, where the learning rate influences how much the model explores new areas of the parameter space (exploration) versus exploiting the current knowledge to improve performance (exploitation).

References

[1] Sharifani, K., Amini, M. (2023). Machine Learning and Deep Learning: A Review of Methods and Applications. World Information Technology and Engineering Journal, 10(07), 3897-3904.

[2] Chauhan, R., Ghanshala, K. K., Joshi, R. C. (2018, December). Convolutional neural network (CNN) for image detection and recognition. In 2018 first international conference on secure cyber computing and communication (ICSCCC) (pp. 278-282). IEEE.

[3] Smith, L. N. (2017, March). Cyclical learning rates for training neural networks. In 2017 IEEE winter conference on applications of computer vision (WACV) (pp. 464-472). IEEE.

[4] Liu, L., Jiang, H., He, P., Chen, W., Liu, X., Gao, J., Han, J. (2019). On the variance of the adaptive learning rate and beyond. arXiv preprint arXiv:1908.03265.

[5] Nakamura, K., Derbel, B., Won, K. J., Hong, B. W. (2021). Learning-rate annealing methods for deep neural networks. Electronics, 10(16), 2029.

[6] Hannan, M. A., How, D. N., Mansor, M. B., Lipu, M. S. H., Ker, P. J., Muttaqi, K. M. (2021). State-of-charge estimation of li-ion battery using gated recurrent unit with one-cycle learning rate policy. IEEE Transactions on Industry Applications, 57(3), 2964-2971.

[7] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Rabinovich, A. (2015). Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1-9).