Cyclical Learning Rates (CLR's) for Improving Training Accuracies and Lowering Computational Cost

Rushikesh Chopade
Indian Institute of Technology
Aditya Stanam
University of Iowa
Anand Narayanan
Yale University
Shrikant Pawar (shrikant.pawar@yale.edu)
Yale University

Research Article

Keywords: One-Class Classifier, Optimizer, Cyclical Learning Rates

Posted Date: December 13th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-1129014/v1

Abstract

Prediction of different lung pathologies using chest X-ray images is a challenging task requiring robust training and testing accuracies. In this article, one-class classifier (OCC) and binary classification algorithms have been tested to classify 14 different diseases (atelectasis, cardiomegaly, consolidation, effusion, edema, emphysema, fibrosis, hernia, infiltration, mass, nodule, pneumonia, pneumothorax and pleural-thickening). We have utilized 3 different neural network architectures (MobileNetV1, Alexnet, and DenseNet-121) with four different optimizers (SGD, Adam, and RMSProp) for comparing best possible accuracies. Cyclical learning rate (CLR), a tuning hyperparameters technique was found to have a faster convergence of the cost towards the minima of cost function. Here, we present a unique approach of utilizing previously trained binary classification models with a learning rate decay technique for re-training models using CLR's. Doing so, we found significant improvement in training accuracies for each of the selected conditions. Thus, utilizing CLR's in callback functions seems a promising strategy for image classification problems.

Introduction:

Speech recognition, computer vision and text analysis are major fields in which deep learning is prominently used for image classification [1, 2, 3]. Cyclical learning rates (CLR's) allow the learning rates to vary between a range of boundary values. Selecting learning rate manually is a time consuming and computationally costly task [4]. Optimal learning rate is important as the model can converge slowly if the learning rate is too slow or the model can diverge from the minima of the cost function if the learning rate is too high [5]. Even if an optimal learning rate for the model is achieved, the model can take many epochs to reach the minima of the loss function. The model doesn't have a regular cost function, moreover, the gradient of the cost function is different in different parts of the cost function curve [6]. To overcome this issue, instead of using constant single learning rate, a learning rate decay policy can be used to obtain better results. However, the learning rate decay also has several drawbacks including getting stuck in a local minimum or plateau of cost function due to very small learning rates in later epochs [7]. CLR's can be an effective technique to make the model converge faster in minimal number of epochs and to decrease the efforts of finding optimal learning rates.

Experimental Results and Analysis:

A. Data collection, preprocessing, model architecture, and learning rates:

A1. Data collection:

The publicly open accessed data used for binary and one-class classification has been made available by National Institutes of Health (NIH), USA [8]. This dataset consists of 112,120 chest X-ray images, each with a 1024*1024-pixel resolution. Images belong to 15 classes, 14 classes of diseased individuals and 1 class of healthy individuals ('No Finding'). The disease classes contain 'Atelectasis', 'Cardiomegaly', 'Consolidation', 'Effusion'; 'Emphysema', 'Edema', 'Fibrosis', 'Infiltration', 'Mass', 'Nodule', 'Pneumonia',

'Pneumothorax', 'Pleural Thickening' and 'Hernia'. A metadata associated with the image dataset consists of patient's age, gender, unique patient id, and the view position (anterior-posterior and posterior-anterior) of the X-ray image. All methods were performed in accordance with the relevant guidelines and regulations with human participants.

A2. Exploratory data analysis:

From the total set, 60,361 images have the label 'No Finding' (healthy), while others have multiple labels with combinations of 14 classes. Overall, the unique constitutes to around 836 labels. Unique can be any of the 14 primary classes (‘No Finding’ label excluded) or any combination of these 14 primary classes. Figure 1 depicts the distribution of these 15 unique labels.

A one-hot encoding was applied to convert 836 unique labels to 15 primary class labels [9]. Comparison of the number of images in 15 primary classes before and after performing one-hot encoding is shown in Table 1. A plot for the number of images after performing one-hot encoding is shown in figure 2.

Table 1
Counts per class for primary labels before and after one-hot encoding.

Image Label	No. of Images before One Hot Encoding	No. of Image before One Hot Encoding
No Finding	60361	60361
Atelectasis	4215	11559
Cardiomegaly	1093	2776
Consolidation	1310	4667
Edema	628	2303
Emphysema	892	2516
Effusion	3955	13317
Fibrosis	727	1686
Infiltration	9547	19894
Mass	2139	5782
Nodule	2705	6331
Pneumothorax	2194	5302
Pneumonia	322	1431
Pleural Thickening	1126	3385
Hernia	110	227

Binary classifiers have been developed on each disease and the 'No Finding' class. The 'No Finding' class has approximately 3 times more images than the 'Infiltration' class, this type of unbalanced dataset can raise a state where the algorithm will overfit the class having more images. To avoid this, the number of images in the 'No Finding' class has been taken approximately the same as the number of images in the class for which the binary classifier was developed.

A3. Pre-processing of data:

A3.1. Binary classifier:

A 1:4-fold split of test to training set was performed for 14 binary classifiers (Table 2). To save overhead memory and making model more robust, we passed all the images through a ImageDataGenerator class of Keras [10] (shear range of 0.05, zoom range of 0.1, rotation range of 7 degrees, width, and height shift range of 0.1, brightness range of 0.4 to 1.5 with a horizontal flip), while subsequently applying image augmentation technique. These techniques helped the model to generalize and reduce the overfitting state.

Table 2
List of binary classifiers and the number of images in their training and test sets.

Binary Classifier	No. of images containing respective disease label	No. of Images with 'No Finding' Label	Total Images	No. of training images (80% of total images)	No. of test images (20% of total images)
Atelectasis	11559	12000	23599	18847	4712
Cardiomegaly	2776	2800	5576	4460	1116
Consolidation	4667	4700	9367	7493	1874
Edema	2303	2300	4603	3682	921
Emphysema	2516	2600	5116	4092	1024
Effusion	13317	13500	26817	21453	5364
Fibrosis	1686	1700	3386	2708	678
Infiltration	19894	20000	39894	31915	7979
Mass	5782	6000	11782	9425	2357
Nodule	6331	6500	12831	10264	2567
Pneumothorax	5302	5500	10802	8641	2161
Pneumonia	1431	1500	2931	2344	587
Pleural Thickening	3385	3500	6885	5508	1377
Hernia	227	250	447	381	96

A dynamic batch training was utilized to decrease computational time and memory. Based on optimal performance, an iterative loop of 32 images/batch was used for training till all the images in batch were exhausted (Table 3). Apart from utilizing less memory, this method helps to save fewer errors in the memory for updating hyperparameters through backpropagation which increases the training speed drastically. The high-resolution X-ray images for training have higher fractional improvements in area under curve (AUC) [11], and also can help localize a disease pattern.

Table 3
Number of training and testing batches with respective batch sizes for all the binary classifiers.

Binary Classifier	Total Images	Batch Size	No. of training batches	No. of test batches
Atelectasis	23599	16	1178	295
Cardiomegaly	5576	32	140	35
Consolidation	9367	32	235	59
Edema	4603	32	116	29
Emphysema	5116	32	128	32
Effusion	26817	32	671	168
Fibrosis	3386	32	85	22
Infiltration	39894	16	1995	499
Mass	11782	32	295	74
Nodule	12831	32	321	81
Pneumothorax	10802	32	271	68
Pneumonia	2931	16	147	37
Pleural Thickening	6885	32	173	44
Hernia	447	4	96	24

A3.2. One class classifier:

With the idea of choosing a balanced data, the dataset for one-class classifier contains 2,800 images of "No Finding" class and 200 images from each disease class. We again choose 1:4-fold split of test to training set to be consistent with binary classifiers. Further, the preprocessing through ImageDataGenerator class with same parameters as binary classifiers was performed for this split. Dynamic training with an optimal batch of 16 images/batch was performed.

A4. Model architectures for binary & one-class classifiers:

A4.1. Binary classifier:

A 2D convolutional neural network is applied using an MobileNetV1 network architecture [12]. The model parameters of MobileNet previously trained on ImageNet have been utilized using transfer learning (Figure 3).

For MobileNetV1 previously trained ImageNet weights are passed through a global average pooling layer considering averages of each feature map instead of adding fully connected layers. This technique helps to easily interpret feature maps as categories confidence maps, to reduce overfitting, and is more robust

to spatial translations of the input as it sums out the spatial information [13]. To further reduce overfitting, a dropout regularization layer to drop ~50% of the input units for variance reduction has been applied after the global average pooling layer. The model is then passed through 4 dense layers of output nodes 250, 50, 10, and 2 with linear activation functions in them. In each dense layer, L1 and/or L2 regularization is applied to the layer's kernel, bias, and activity. Kernel regularizer with both L1 and L2 penalties of 0.001 and 0.01 respectively are applied on the kernel's layer. A bias regularizer with an L2 penalty of 0.01 is applied on the layer's bias. Activity regularizer with an L2 penalty of 0.001 is applied on the layer's output. After each dense layer, batch normalization is used to stabilize the learning process and dramatically reduce the number of training epochs required to train a deep neural network. Finally, the model architecture is complete with application of a dense layer comprising of sigmoid activation function and 1 output node. The stochastic gradient descent (SGD) optimizer with learning rate decay has been used to train the model as it gave a superior performance compared to RMSProp and adam optimizer for all the classifiers except "Hernia". Adam optimizer with a learning rate of 0.01 has been found to perform better in case of "Hernia". A momentum parameter has been used to help accelerate gradient vectors in right directions (Table 4).

Table 4
Chart showing optimizer, its momentum, learning rates, and the decay constants used with SGD optimizer for all binary classifiers (Except Hernia which has Adam optimizer).

Binary Classifier	Optimizer Used	Learning Rate	Decay constant	Momentum
Atelectasis	SGD	0.01	0.001	0.9
Cardiomegaly	SGD	0.1	0.0005	0.9
Consolidation	SGD	0.05	0.0005	0.9
Edema	SGD	0.01	0.0005	0.9
Emphysema	SGD	0.01	0.0005	0.9
Effusion	SGD	0.01	0.001	0.9
Fibrosis	SGD	0.001	0.00005	0.9
Infiltration	SGD	0.01	0.001	0.9
Mass	SGD	0.01	0.001	0.9
Nodule	SGD	0.01	0.001	0.9
Pneumothorax	SGD	0.01	0.0005	0.9
Pneumonia	SGD	0.01	0.0001	0.9
Pleural Thickening	SGD	0.05	0.0005	0.9
Hernia	Adam	0.01	0.0001	0.9

A4.2. One class classifier:

A false-positive predictions arise when the algorithm is unable to identify the "No Finding" class, a problem falling under the category of "Anomaly Detection". One-class classifier is an unsupervised learning algorithm focusing on the problem of anomaly detection [14]. The model contains a negative class (inlier or normal class) and a positive class (outlier or anomaly class). In our case, the normal class or inlier class is the "No Finding" class. The anomaly class is formed by combining 200 images of each disease class. The benefit of this approach is that if the prediction/test image fed to the algorithm is not from any of the 14 disease classes, it will still categorize it as an "Anomaly" simply because the algorithm could not classify it as an image with "No Finding" class. If the algorithm classifies the image with a disease other than these 14 diseases as a "No Finding" class, it will give rise to a problem of false negative prediction. One-class classifier serves the purpose of solving the problem of both false positives and false negative predictions. The model architecture for one-class classifier is same as the binary classifier.

A5. Cyclical learning rates:

The first step in applying CLR's is to define a maximum learning rate and a base learning rate [4]. The learning rate can then be allowed to vary between maximum learning rate and base learning rate. We have utilized learning rate finder technique (described in section A6) to decide maximum learning and the base learning rates. For one condition, "Pneumothorax" binary classifier, maximum and the base learning rates of 0.03 and 0.0075 respectively were obtained using learning rate finder. A step size is an important parameter which simply is the number of batches in which the learning rate will become equal to the maximum learning rate starting from the base learning rate or vice-versa. It is the number of training batches to reach half cycle. Typically, the step size of 2-8 times the number of training batches in 1 epoch is ideal [4]. For "Pneumothorax", the total number of training batches in 1 epoch is equal to 541. Therefore, a step size of 1082 was used for learning rate finder. Finally, a mode policy needs to be defined for calculating learning rates. Mode is the pattern in which the learning rate will vary within the bounds of maximum and minimum learning rates. The "triangular" policy for "Pneumothorax" binary classifier is shown in figure 4. The learning rate monotonically increases to maximum learning rate from base learning rate in two epochs and decreases back to base learning rate in the next two epochs. Since the "Pneumothorax" model with CLR technique and "triangular" policy is trained for 36 epochs, a total of 9 full cycles can be observed in figure 4.

We also have parallelly utilized a more complex policy called a "modified triangular2" policy. In this policy, the maximum learning rate is not taken to be the average of previous maximum learning rate unlike "triangular2" policy. After 3 complete cycles of the "triangular2" policy, the training is continued with "triangular2 policy" with original maximum learning rate obtained from the learning rate finder technique. This process is carried out until whole training is exhausted. In the "Pneumothorax" binary classifier, the maximum learning rate in the first cycle is 0.03 from the first learning rate finder cycle, followed by

second cycle with maximum learning rate of 0.01875, followed by third cycle with maximum learning rate of 0.013125 (figure 5), etc.

A6. Learning rate finder:

The upper and lower bounds of the CLR have been determined by learning rate finder technique where the cost function is minimum. Training the model with a learning rate finder as a callback for 1-5 epochs was enough to get the learning rate with minimum cost function. In case of the “Pneumonia” binary classifier, the minimum and maximum values for the learning rates were 1e-7 as minimum and 1 as maximum (figure 6). The training increases exponentially after each batch on minimum learning rate. The “Pneumothorax” model loss vs. learning rate curve trained for 10 epochs is found to have a learning rate of 3e-2 with minimum loss (figure 6). This loss increased as the learning rate approached to 1. The base learning rate for CLR can be accounted to one-fourth of the maximum learning rate [4].

A7. With binary classifiers CLR’s out-perform normal training with a learning rate decay policy:

We have run 3 model architectures (MobileNetV1, AlexNet, and DenseNet121) for comparing the performance (computational cost & accuracy) of classifiers [15, 16]. MobileNetV1 with an SGD optimizer was found to be most efficient, while DenseNet121 had good accuracy but significantly more computational cost, AlexNet had significantly lower accuracies when trained for the same number of epochs (Table 5).

Table 5
Accuracies of all the binary classifiers after training for given number of epochs.

Binary Classifier	No. of Epochs	Accuracy (in %)
Atelectasis	10	75.10
Cardiomegaly	12	75.78
Consolidation	10	73.32
Edema	12	93.37
Emphysema	10	85.60
Effusion	10	86.53
Fibrosis	10	66.58
Infiltration	10	64.60
Mass	10	70.11
Nodule	10	68.23
Pneumothorax	10	70.12
Pneumonia (with CLR)	30	88.43
Pleural Thickening	10	71.67
Hernia	30	90.81

The problem of false-positive predictions was addressed using one-class classifiers. For the models of “Infiltration”, “Atelectasis”, “Fibrosis” & “Pneumothorax” the accuracies have been consistently low after training for the selected number of epochs. So, we chose these conditions to test CLR’s on (Table 6).

Table 6
Comparison of the network architectures for “Atelectasis” binary classifier.

Name of Model Architecture	Approx. training time per epoch in hours	Training Epochs	Accuracy (in %)
MobileNetV1	2.5	10	75.10
DenseNet121	5	10	78.07
AlexNet	1.5	30	75.50

The problem of false-positive and false-negative predictions was resolved with one class classifiers. After which, a selected model trained for 32 epochs using CLR’s with a maximum learning rate of 0.1, a base learning rate of 0.025, a step size of 2, and with a "triangular" policy provided a final training accuracy of

83.01%. CLR's showed improved accuracy and a lower computational cost compared to training a network with constant learning rates (Table 7 and 8).

Table 7
Classifier accuracies after application of CLR's.

Binary Classifier	Accuracy before CLR application (in %)	Epochs taken to achieve the accuracy before CLR application	Accuracy after CLR application (in %)	Epochs taken to achieve the accuracy after CLR application	Policy Used
Atelectasis	75.10	10	79.59	32	Triangular
Infiltration	64.6	10	76.15	10	Modified Triangular2
Fibrosis	66.58	10	88.96	32	Modified Triangular2
Pneumothorax	70.12	10	79.83	36	Triangular
Pneumonia	-	-	88.43	30	Triangular

Table 8
Parameters and specifications of the CLR's.

Binary Classifier	Policy Used	Step Size	Epochs	Maximum Learning Rate	Base Learning Rate
Atelectasis	Triangular	2	32	0.1	0.025
Infiltration	Modified Triangular2	2	10	0.02	0.005
Fibrosis	Modified Triangular2	2	32	0.002	0.0005
Pneumothorax	Triangular	2	36	0.03	0.0075
Pneumonia	Triangular	2	30	0.01	0.0001

The "Pneumothorax" model is found to perform best when the CLR's is used with a "triangular" policy. As shown in figure 7, it took 47 epochs for the model with a constant learning rate to reach an accuracy of 79.26%. With CLR using "modified triangular2" policy crossed the accuracy level of 79.26% at 38 epoch and reached the accuracy of 80.92% in 41 epochs. While, the "Pneumothorax" model with CLR using a "triangular" policy crossed the accuracy level of 79.26% in just 36 epochs to achieve final accuracy of 79.83%.

The loss compared to the number of epochs was seen to be decreased with CLR's in both "triangular" and "modified triangular2" policies (Figure 8). The loss of the "Pneumothorax" model with CLR reduced quicker than the "Pneumothorax" model with a constant learning rate.

The “Fibrosis” model was found to give better results in the case of the CLR technique with a “modified triangular2” policy. A comparison of “fibrosis” model trained for 32 epochs is shown in figure 9. The model reached an accuracy of 85.04% in 32 epochs when trained with a constant learning rate policy. The model reached an accuracy of 86.96% in 32 epochs when trained with CLR using a “triangular” policy. It crossed the 85% accuracy level in 30 epochs. The model reached an accuracy of 88.15% in 32 epochs when trained with CLR using a “modified triangular2” policy. It crossed the accuracy level of 85% in just 25 epochs. The loss was observed to be always less in CLR’s with a “modified triangular2” (figure 10).

Discussion & Future Scope:

Depthwise separable convolutions like MobileNets have been gradually pruned for improving the speed of dense network [17]. MobileNetV1 Imagenet weights with SGD optimizer is found to outperform other optimizers and architectures in terms of training time taken and accuracy attained. Achieving a high test accuracy is directly depended on learning rate hyper-parameter for training neural networks [18, 19, 20, 21]. Three forms of triangle CLR’s have been stated to accelerate neural network training [18, 19]. Further, tuning the batch size hyper-parameter for adjusting learning rates have also been shown to improve learning accuracy [22]. Some hyperparameter tools like Hyperopt, SMAC, and Optuna, using grid search, random search and bayesian optimization have been seen efficient in tuning batch sizes [23, 24]. To the best of our knowledge, our work is the first to present a comprehensive characterization of CLR function on training and testing accuracy of dense network models. In general, training any model with a CLR technique is found to perform better than training with a constant learning rate. For the “Pneumothorax” binary classifier, the CLR technique with the "triangular" policy is found to outperform both CLR with the "modified triangular2" policy and constant learning rate training. For the “Fibrosis” binary classifier, the CLR with the "modified triangular2" policy was found to give better results than the rest two policies. Primarily, we found that there are two main advantages of training with CLR’s over constant learning rates, with decay learning rates the model can get stuck into the saddle points or local minima due to low learning rates, and secondly CLR’s reduces the effort of choosing an optimal learning rate by hit and trial method. Poor choice of initial learning rate can make the model circle infinitely. In setting a learning rate, there is a trade-off between the rate of convergence and overshooting, a high learning rate will make the learning jump over minima but a too low learning rate will either take too long to converge or get stuck in an undesirable local minimum [25]. The CLR’s cyclically provided higher learning rates too, which helped the model to jump out of the local minima of the cost function. With these findings, implementing CLR’s for improving prediction accuracies seems a promising strategy for object detection and machine translation.

Declarations

Supplementary data:

None

Author contributions:

SP, AS, AN, and RC conceived the concepts, planned, and designed the article. SP, AS, AN, and RC primarily wrote and edited the manuscript.

Competing interests:

The authors declare that they have no competing interests. No experiments on humans and/or the use of human tissue samples was performed in this study.

References

1. R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference, 580–587. 2014.

2. A. Graves and N. Jaitly. Towards end-to-end speech recognition with recurrent neural networks. In Proceedings of the 31st International Conference on Machine Learning (ICML14), 1764–1772. 2014.

3. Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, 1701–1708. 2014. IEEE.

4. Leslie N. Smith. Cyclical Learning Rates for Training Neural Networks. 2017. arXiv:1506.01186v6.

5. Wilson, R.C., Shenhav, A., Straccia, M. et al. The Eighty Five Percent Rule for optimal learning. Nat Commun 10, 4646. 2019. https://doi.org/10.1038/s41467-019-12552-4.

6. Santanu Pattanayak. A Mathematical Approach to Advanced Artificial Intelligence in Python. Pro Deep Learning with TensorFlow. 2017. DOI: 10.1007/978-1-4842-3096-1.

7. Bukhari, S. T., & Mohy-Ud-Din, H. A systematic evaluation of learning rate policies in training CNNs for brain tumor segmentation. Physics in medicine and biology, **66**(10), 10.1088/1361-6560/abe3d3. 2021. https://doi.org/10.1088/1361-6560/abe3d3

8. Wang X, Peng Y, Lu L, Lu Z, Bagheri M, Summers RM. ChestX-ray8: Hospital-scale Chest X-ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseases. IEEE CVPR. 2017.

9. Zhang, S. W., Zhang, X. X., Fan, X. N., & Li, W. N. LPI-CNNCP: Prediction of lncRNA-protein interactions by using convolutional neural network with the copy-padding trick. Analytical biochemistry, **601**, 113767. 2020. https://doi.org/10.1016/j.ab.2020.113767.

10. Chollet, F., & others. Keras. GitHub. 2015. Retrieved from https://github.com/fchollet/keras

11. Carl F. Sabottke & Bradley M. Spieler. The Effect of Image Resolution on Deep Learning in Radiography. Radiology: Artificial Intelligence. 2(1):e190015. 2020. https://doi.org/10.1148/ryai.2019190015

12. Pang, S., Wang, S., Rodríguez-Patón, A., Li, P., & Wang, X. An artificial intelligent diagnostic system on mobile Android terminals for cholelithiasis by lightweight convolutional neural network. PloS one, **14(9)**, e0221720. 2019. https://doi.org/10.1371/journal.pone.0221720

13. Min Lin, Qiang Chen, Shuicheng Yan. Network in Network. 2014. https://arxiv.org/pdf/1312.4400v3.pdf.

14. Dai, H., Cao, J., Wang, T., Deng, M., & Yang, Z. Multilayer one-class extreme learning machine. Neural networks: the official journal of the International Neural Network Society, **115**, 11–22. 2019. https://doi.org/10.1016/j.neunet.2019.03.004

15. Chen, J., Wan, Z., Zhang, J., Li, W., Chen, Y., Li, Y., & Duan, Y. Medical image segmentation and reconstruction of prostate tumor based on 3D AlexNet. Computer methods and programs in biomedicine, **200**, 105878. 2021. https://doi.org/10.1016/j.cmpb.2020.105878

16. Urinbayev, K., Orazbek, Y., Nurambek, Y., Mirzakhmetov, A., & Varol, H. A. End-to-End Deep Diagnosis of X-ray Images. Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual International Conference, 2182–2185. 2020. https://doi.org/10.1109/EMBC44109.2020.9175208

17. Cheng-Hao Tu, Yi-Ming Chan, Jia-Hong Lee, Chu-Song Chen. Pruning Depthwise Separable Convolutions for MobileNet Compression. IEEE WCCI. 2020. **DOI:** 10.1109/IJCNN48605.2020.9207259

18. L. N. Smith and N. Topin, “Super-Convergence: Very Fast Training of Neural Networks Using Large Learning Rates,” arXiv e-prints, p. arXiv:1708.07120, Aug 2017.

19. L. N. Smith, “Cyclical Learning Rates for Training Neural Networks,” arXiv e-prints, p. arXiv:1506.01186, Jun. 2015.

20. P. Goyal, P. Dollar, R. B. Girshick, P. Noordhuis, L. Wesolowski, ´ A. Kyrola, A. Tulloch, Y. Jia, and K. He, “Accurate, large minibatch SGD: training imagenet in 1 hour,” CoRR, vol. abs/1706.02677, 2017. [Online]. Available: http://arxiv.org/abs/1706.02677

21. H. Zulkifli, “”understanding learning rates and how it improves performance in deep learning”,” https://towardsdatascience.com/ understanding-learning-rates-and-how-it-improves-performance-indeep-learning-d0d4059c1c10, 2018, [Online; accessed 23-Sep-2018].

22. F. Hutter, H. H. Hoos, and K. Leyton-Brown, “Sequential modelbased optimization for general algorithm configuration,” in Learning and Intelligent Optimization, C. A. C. Coello, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 507–523. 2011.

23. Hyperopt Developers, ""hyperopt — distributed asynchronous hyperparameter optimization in python"," http://hyperopt.github.io/hyperopt/, 2019, [Online; accessed 13-Aug-2019].

24. T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A nextgeneration hyperparameter optimization framework," in Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, ser. KDD '19. New York, NY, USA: ACM, pp. 2623–2631. 2019.

25. Buduma, Nikhil; Locascio, Nicholas. Fundamentals of Deep Learning : Designing Next-Generation Machine Intelligence Algorithms. O'Reilly. p. 21. ISBN 978-1-4919-2558-4. 2017.

Figures

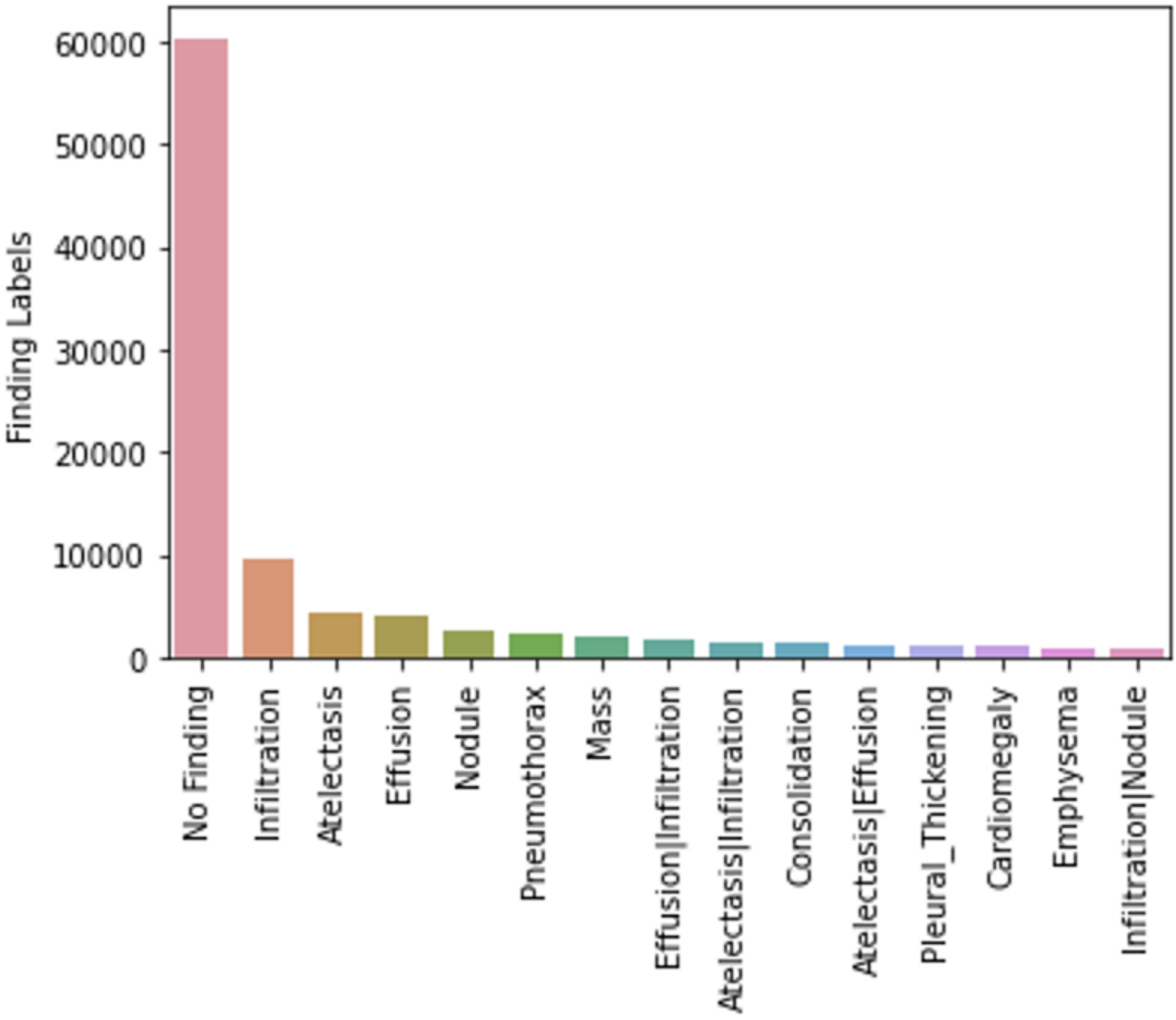

Figure 1

Number of top 15 unique labels.

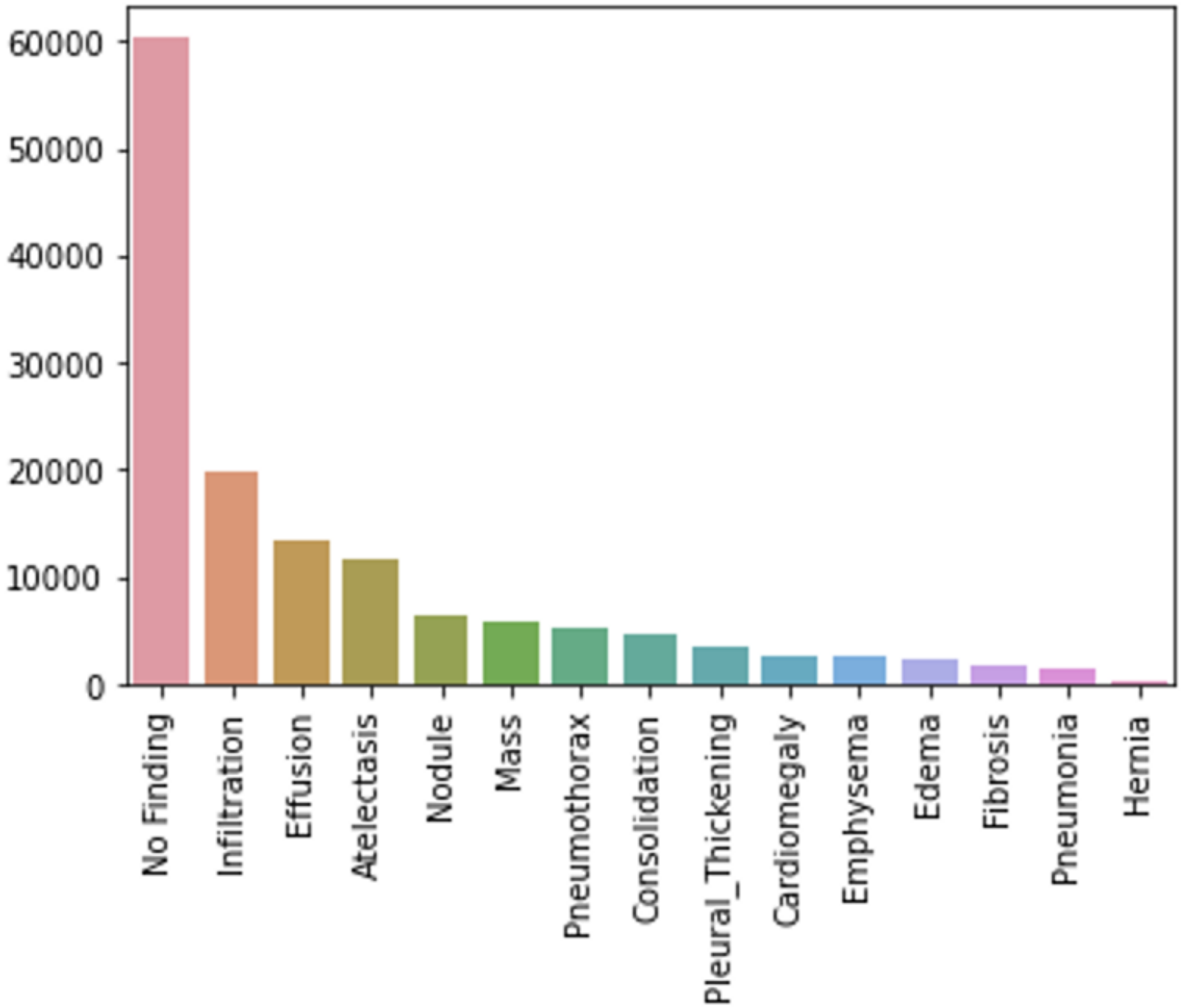

Figure 2

Counts per class for primary labels after one-hot encoding.

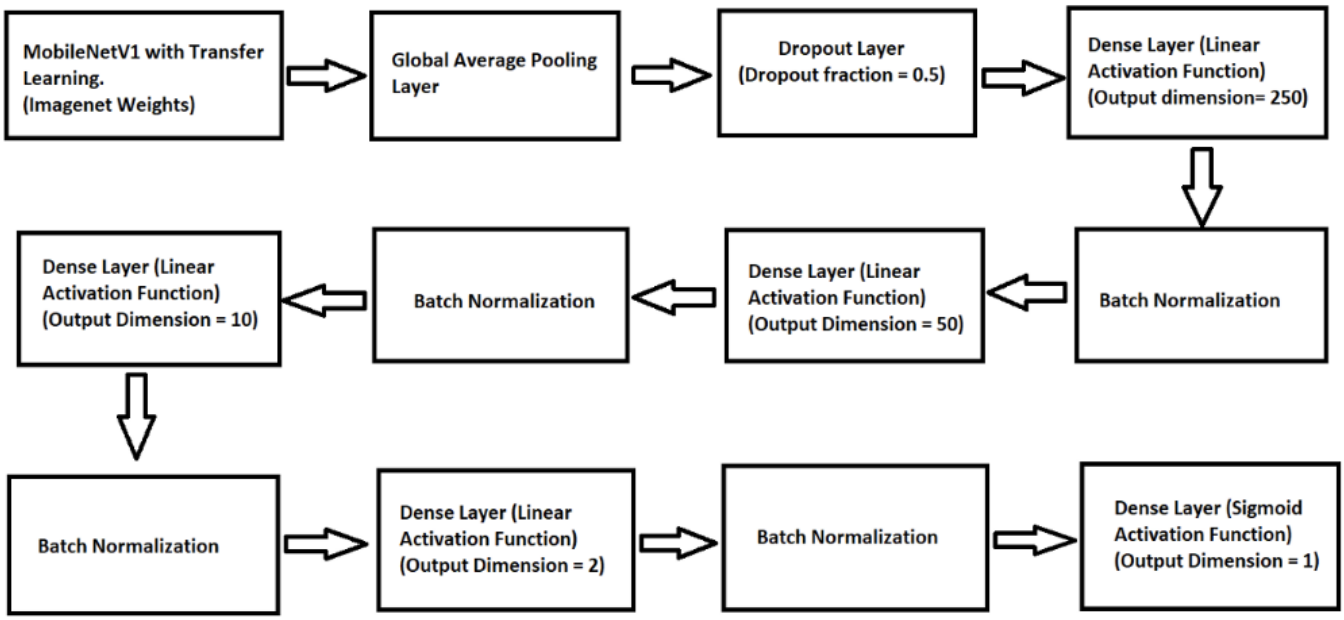

Figure 3

Model architecture used for all the binary classifiers.

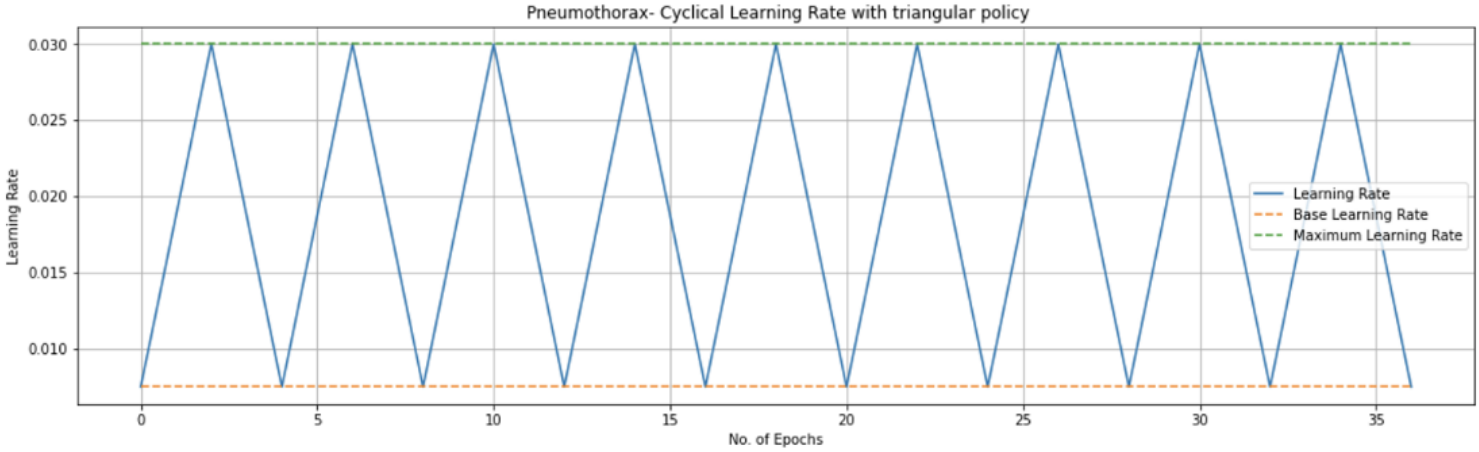

Figure 4

Plot showing the "Triangular" policy for “Pneumothorax” binary classifier trained for 36 epochs.

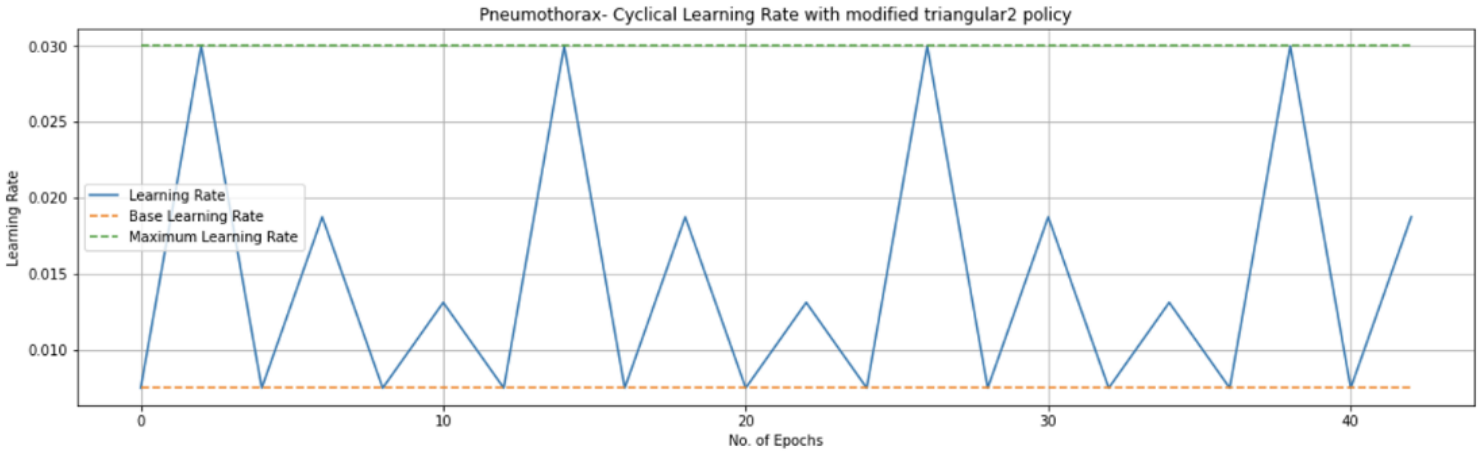

Figure 5

Plot for the "modified triangular2" policy of “Pneumothorax” binary classifier trained for 42 epochs.

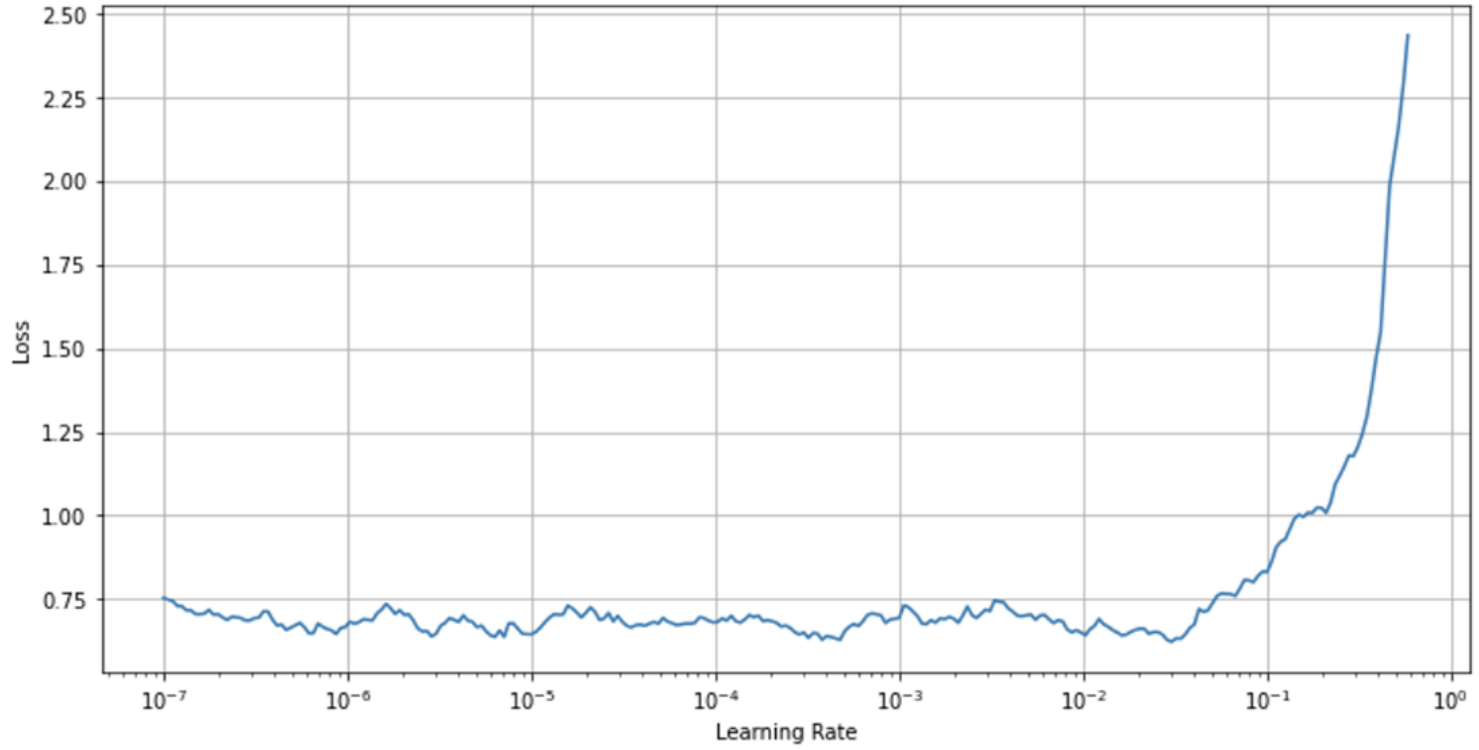

Figure 6

Loss vs. learning rate plot for “Pneumothorax” binary classifier trained for 10 epochs.

Figure 7

Accuracy plot for “Pneumothorax” binary classifier with constant learning rate, CLR with “triangular” and CLR with “modified triangular2” policies.

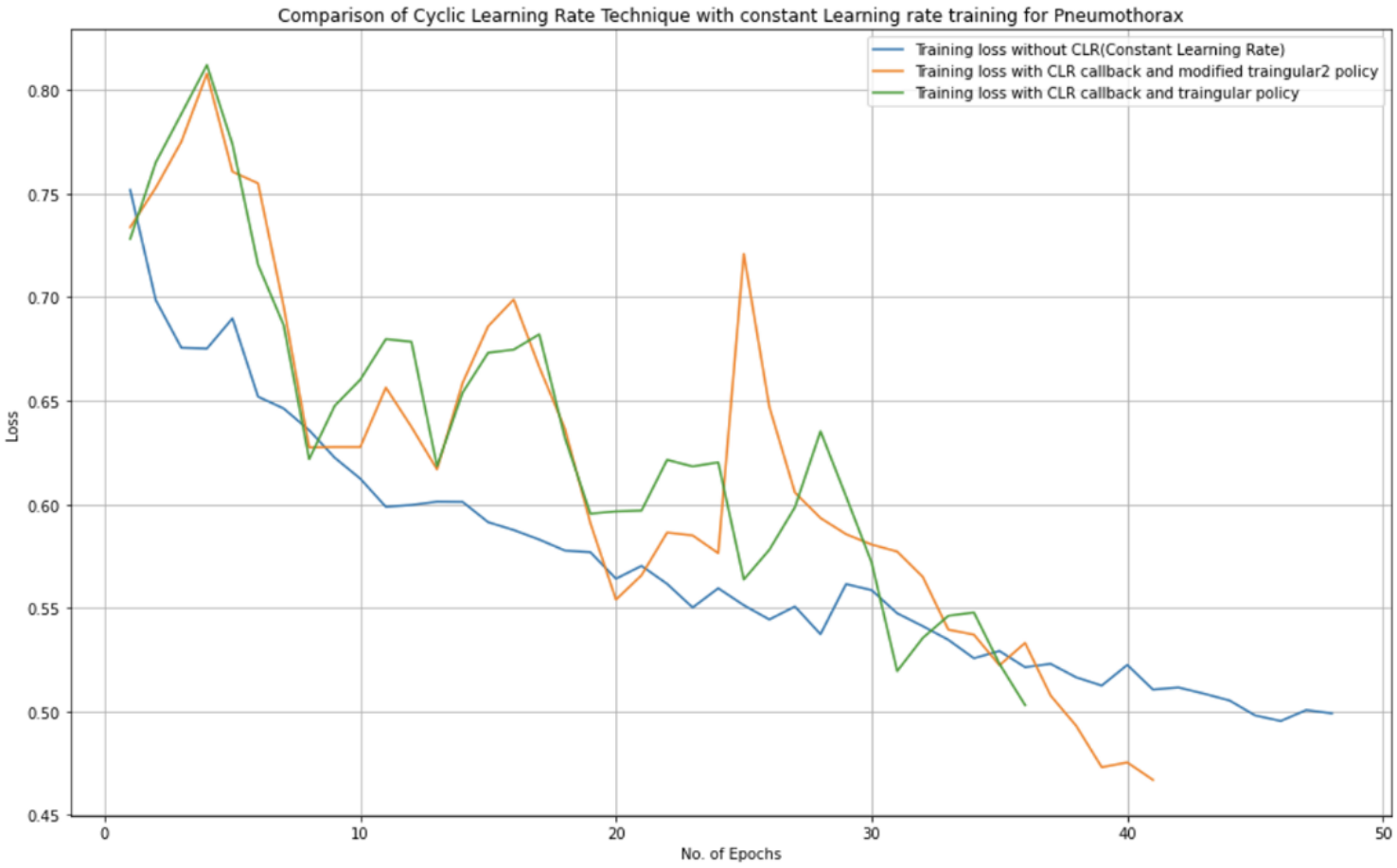

Figure 8

Loss for “Pneumothorax” model with constant learning rate, CLR with “triangular” policy and CLR with “modified triangular2” policy.

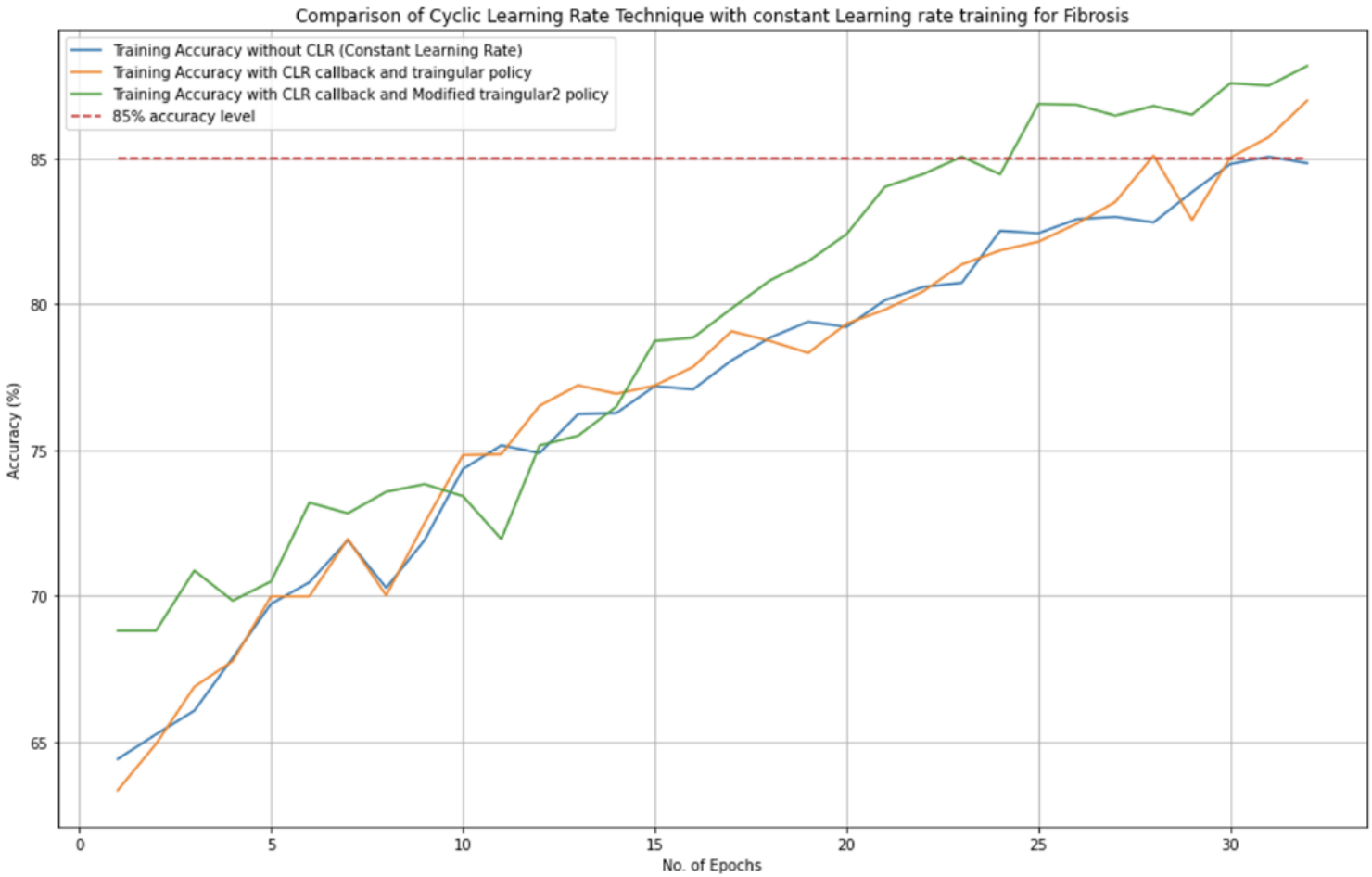

Figure 9

Accuracy plot comparing “Fibrosis” binary classifier with constant learning rate, CLR with “triangular” policy and CLR with “modified triangular2” policy.

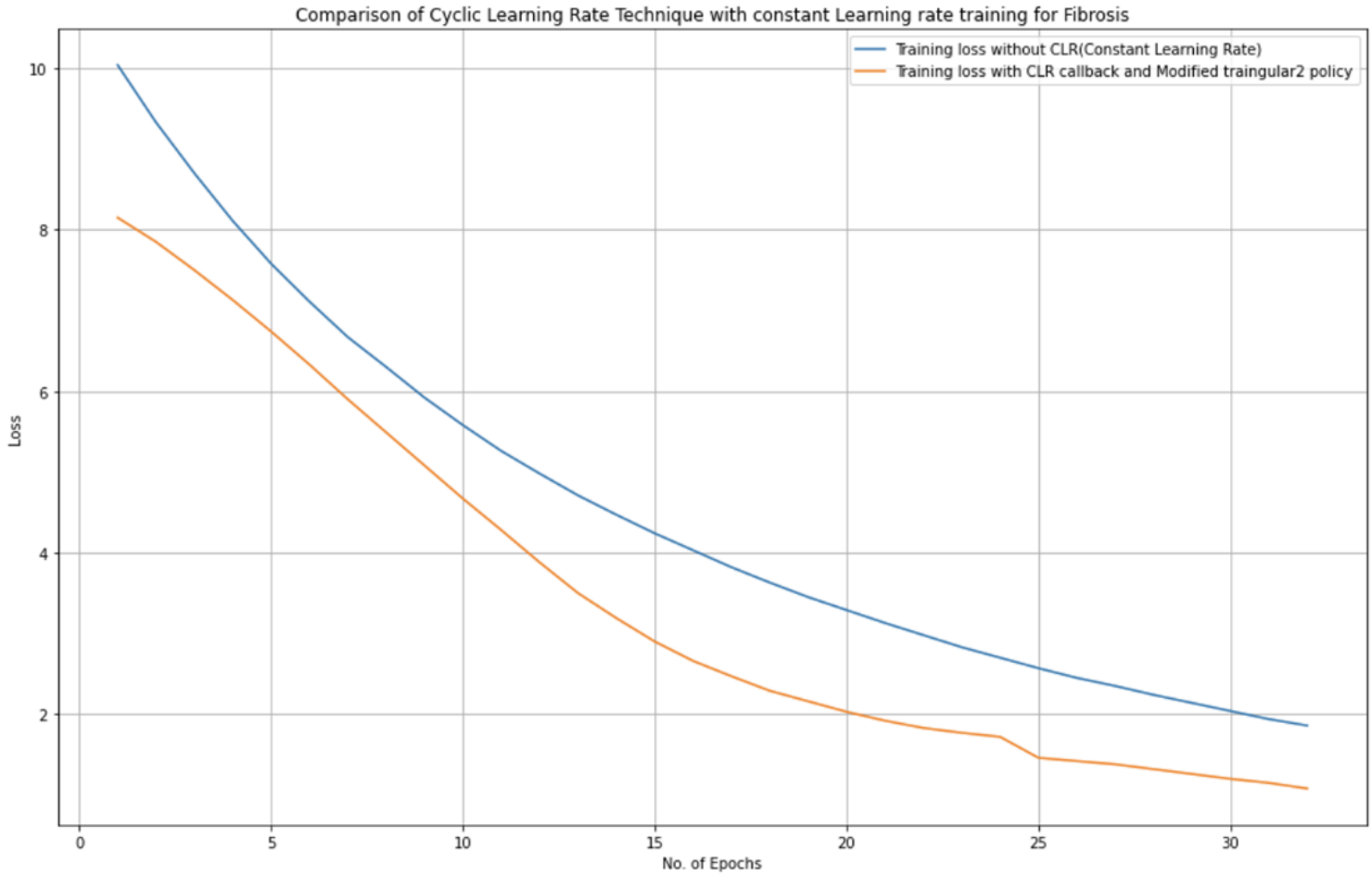

Figure 10

Loss for “Fibrosis” binary classifier with constant learning rate and CLR with a “modified Triangular2” policy.