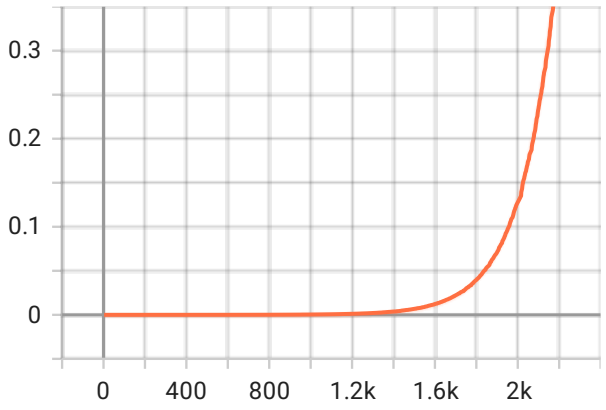
(Github: https://github.com/RishabhSheoran/Assignment6_Colossalai)

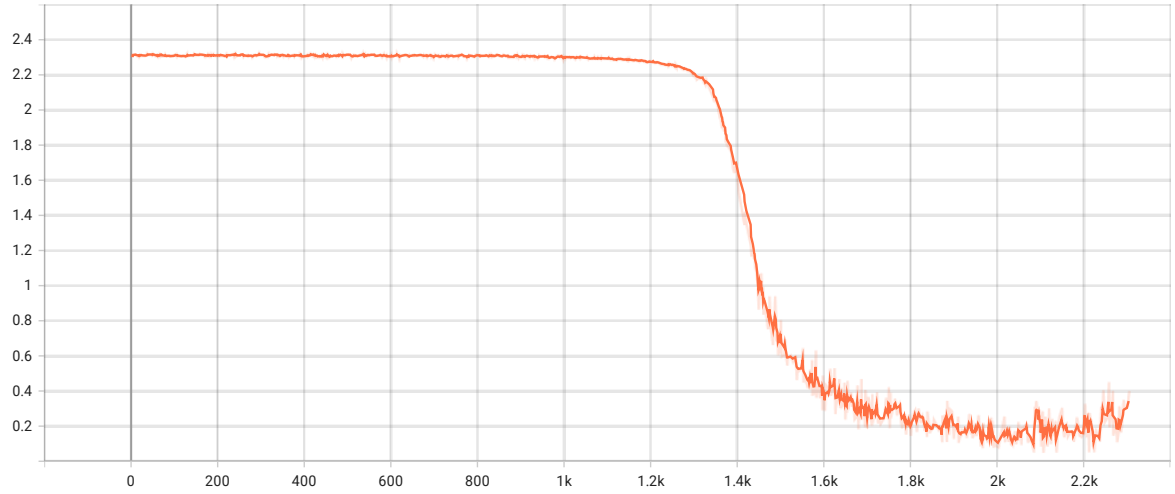
OPTIMIZER-1: SGD

LR Range Test:

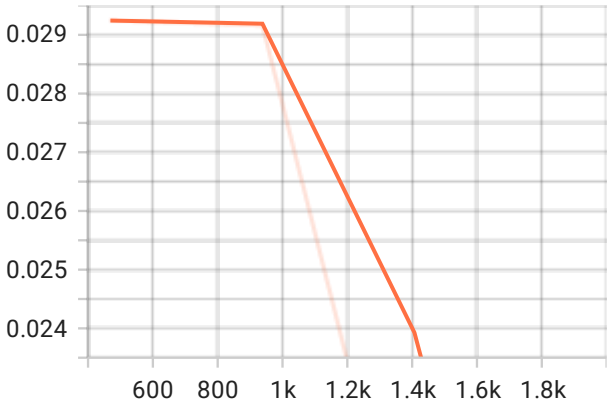
LR/train



Loss/train



Loss/test



We observe that the loss starts decreasing at a fast pace from 1.432k step and reaches the lowest at 2.084k step. We check the corresponding learning rates for the steps in the LR/train plot and choose the below 4 learning rates for the corresponding steps range (1.432k step, 2.084k step):

-lr: 0.0046401, step: 1.432k

-lr: 0.01271, step: 1.603k

-lr: 0.05044, step: 1.837k

-lr: 0.2161, step: 2.084k

Proposed LRs:

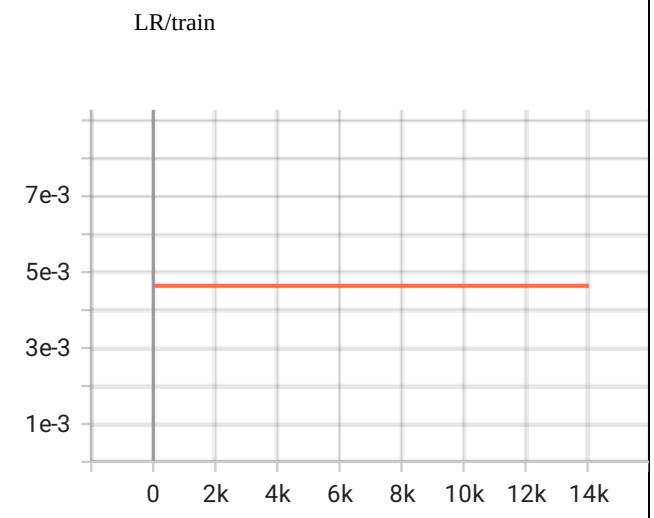
lr = [0.0046401, 0.01271, 0.05044, 0.2161]

We perform experiments with the proposed LRs using 2 learning rate schedulers to find the best lr value for the given optimizer:

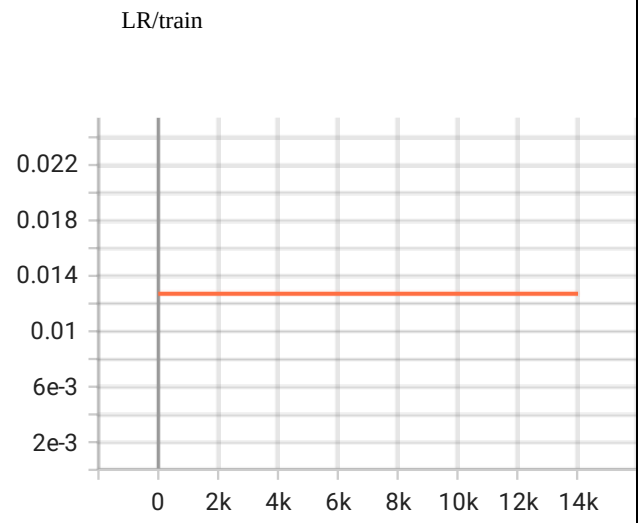
(a) Fixed learning rate
(b) STEPLR scheduler

(a) **Fixed learning rate:**
Th learning rate is fixed here.

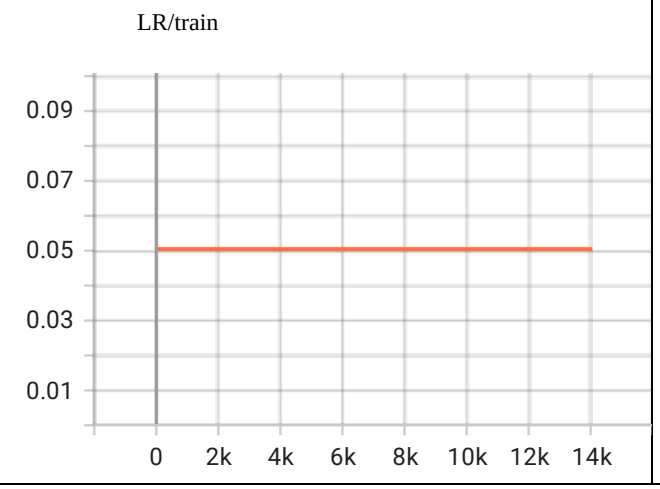
Lr = 0.0046401



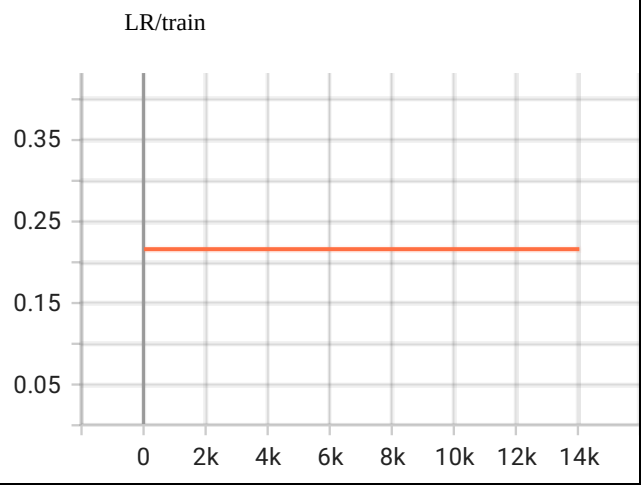
Lr = 0.01271



Lr = 0.05044

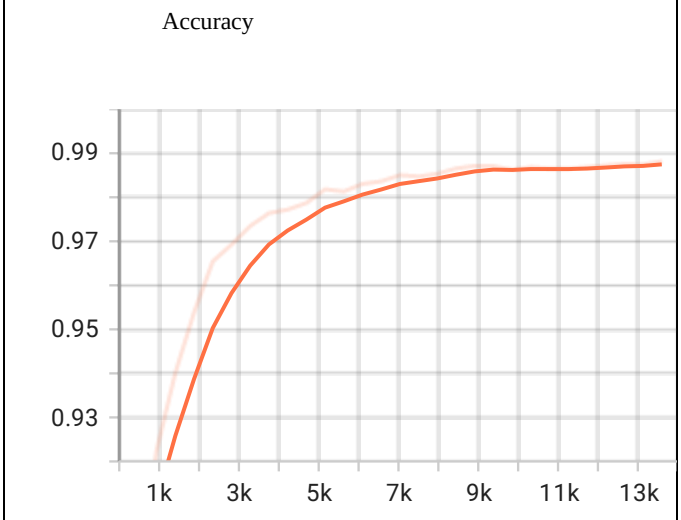


Lr = 0.2161

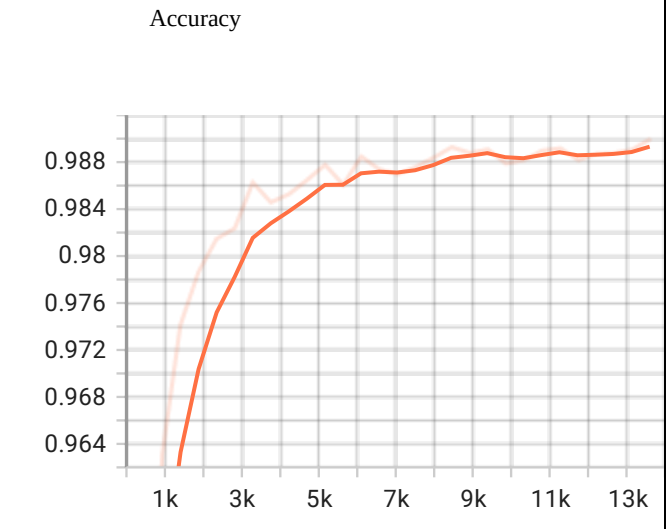


We analyse the Accuracy and LR/train plots for the proposed learning rates to choose the optimal lr. We get the maximum test accuracy when lr = 0.0127 and it also gives us the minimum loss. Another good candidate is lr=0.0046401 which gives us comparable test accuracy and training loss. The loss function is also smoother for lr=0.0046401 as the lr=0.0046401 is less than lr = 0.0127. However, **we choose lr = 0.0127 as the best lr for fixed learning rate** for its higher test accuracy and lower train loss. The test accuracy for lr=0.0127 is also increasing further as seen from the graph.

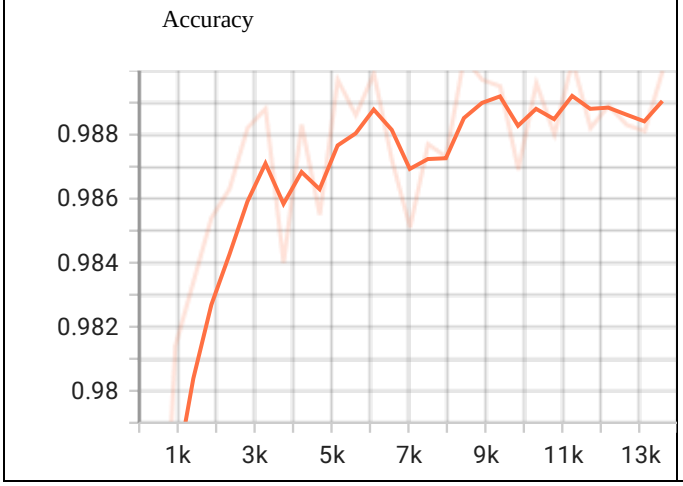
Lr = 0.0046401



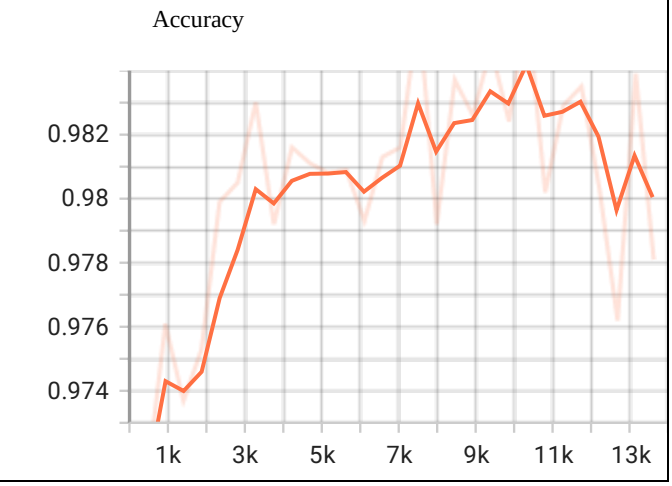
Lr = 0.01271



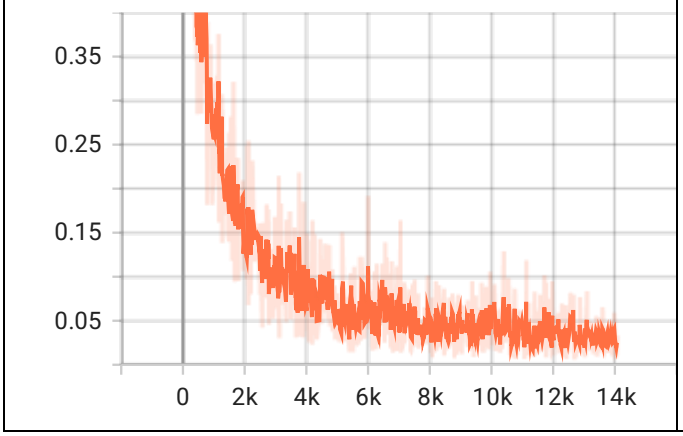
Lr = 0.05044



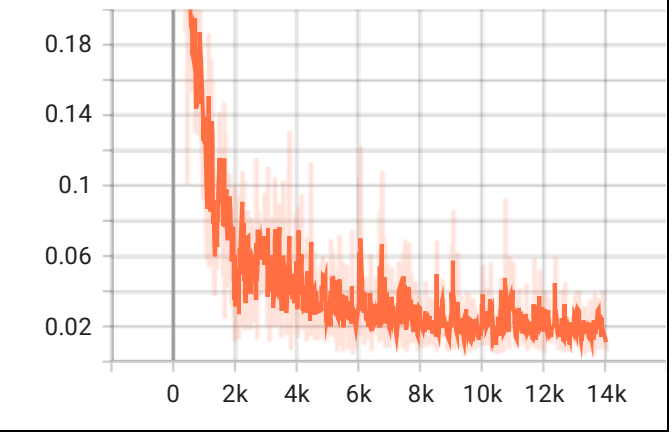
Lr = 0.2161



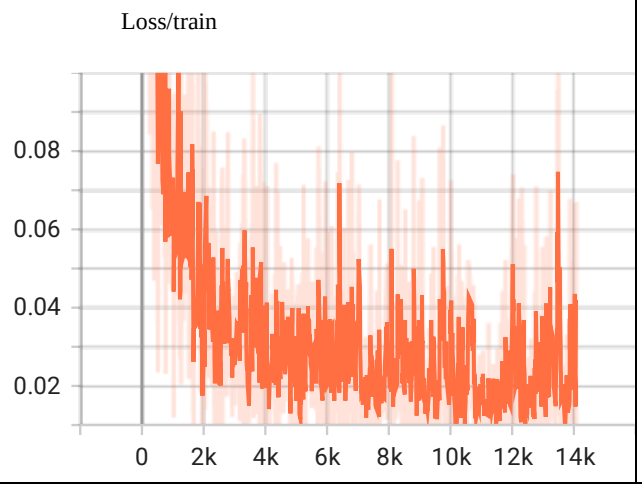
Lr = 0.0046401



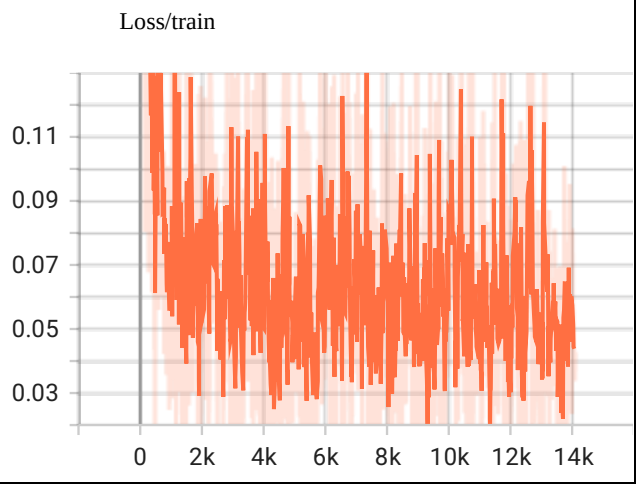
Lr = 0.01271



Lr = 0.05044



Lr = 0.2161



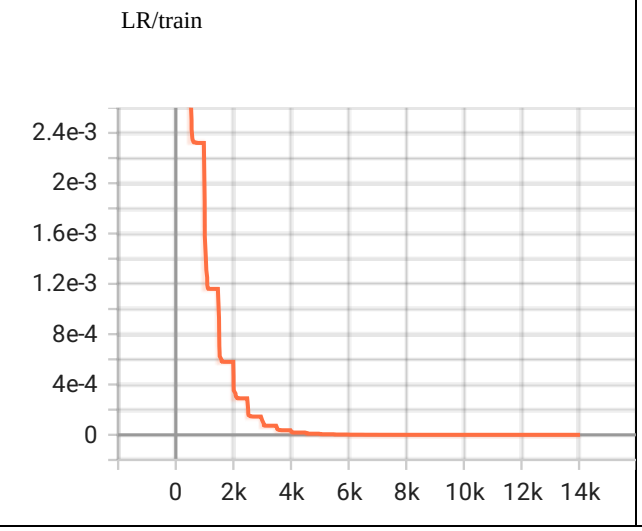
(b) **STEPLR scheduler:**

The STEPLR scheduler updates(decreases) the learning rate after a fixed number of steps. We use Pytorch's StepLR scheduler with the below parameters:

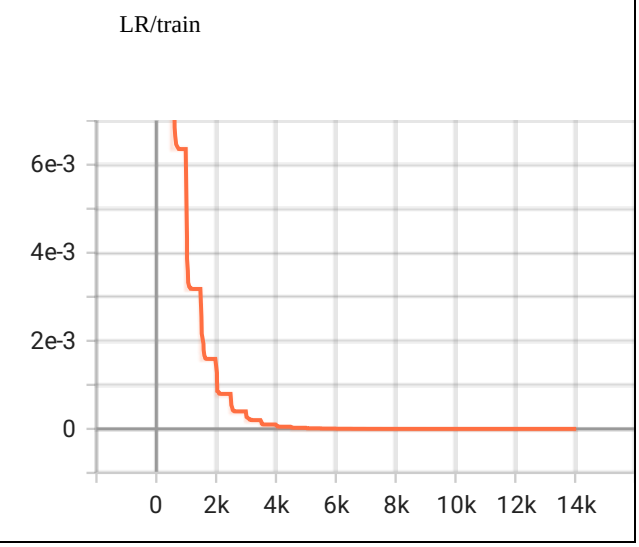
```
torch.optim.lr_scheduler.StepLR(optimizer, step_size=500, gamma=0.5, last_epoch=
- 1, verbose=False)
```

This will decrease the learning rate by half after every 500 steps.

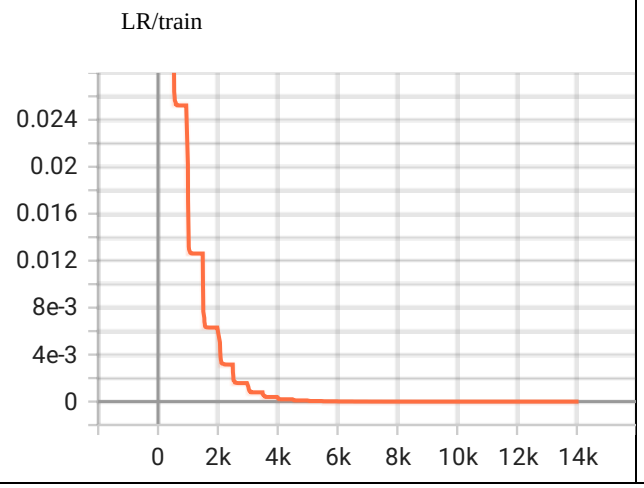
Lr = 0.0046401



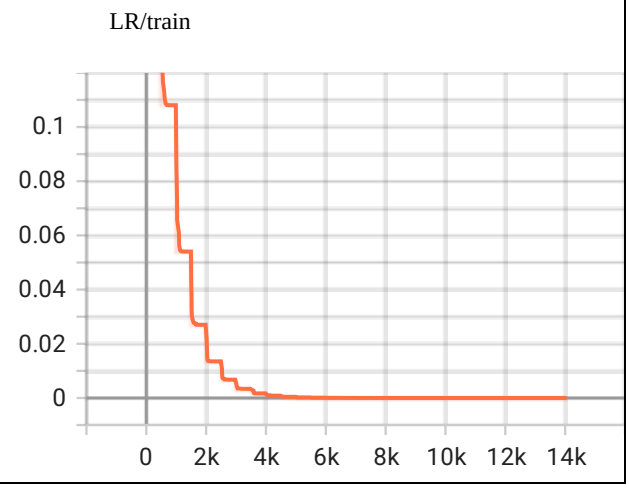
Lr = 0.01271



Lr = 0.05044

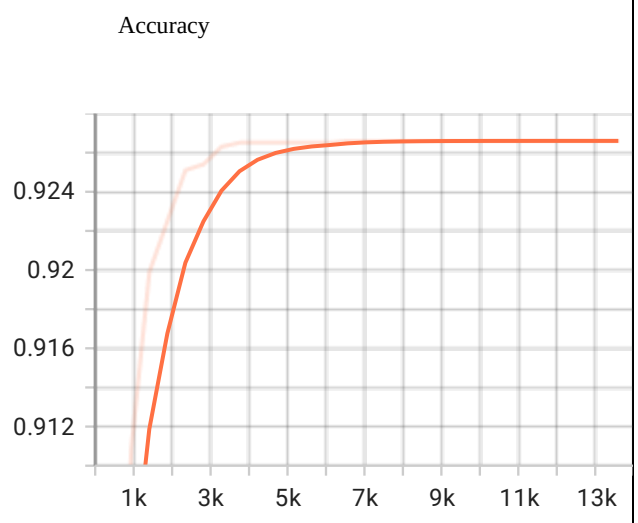


Lr = 0.2161

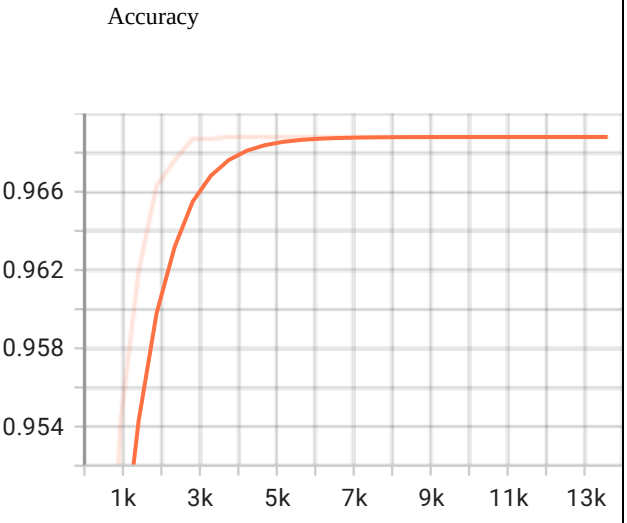


We analyse the Accuracy and LR/train plots for the proposed learning rates to choose the optimal lr. We get the maximum test accuracy when lr = 0.2161. We also get a smoother and lower loss for it as compared to the plots of other proposed learning rates. **We choose lr = 0.2161 as the optimal learning rate for STEPLR for SGD.**

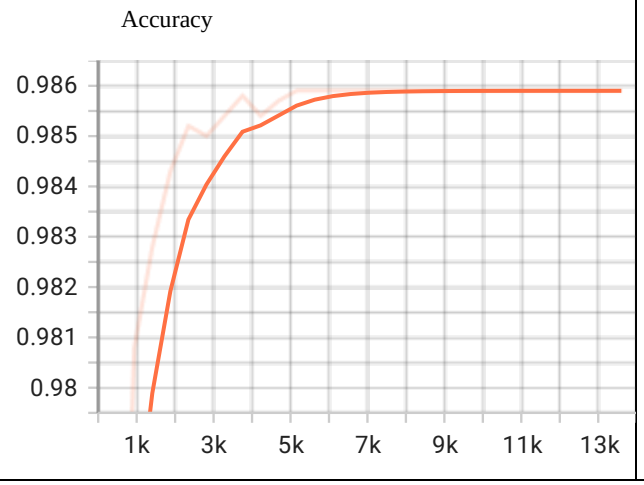
Lr = 0.0046401



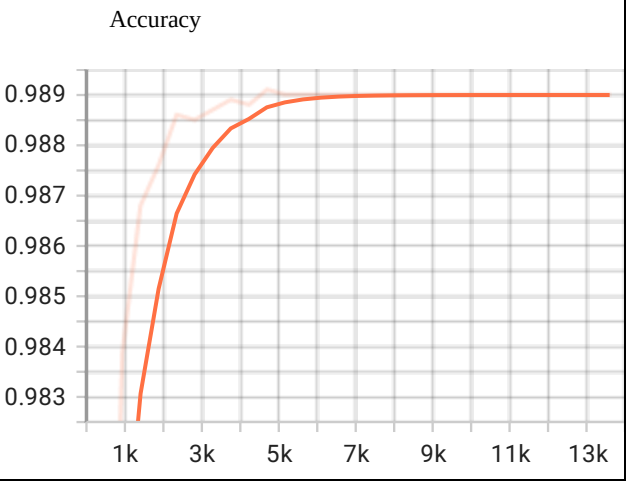
Lr = 0.01271



Lr = 0.05044

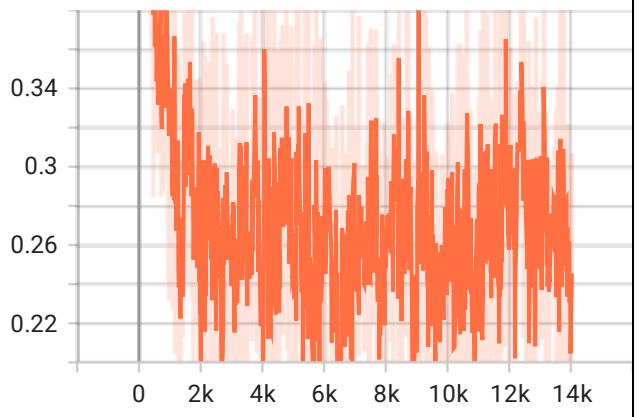


Lr = 0.2161



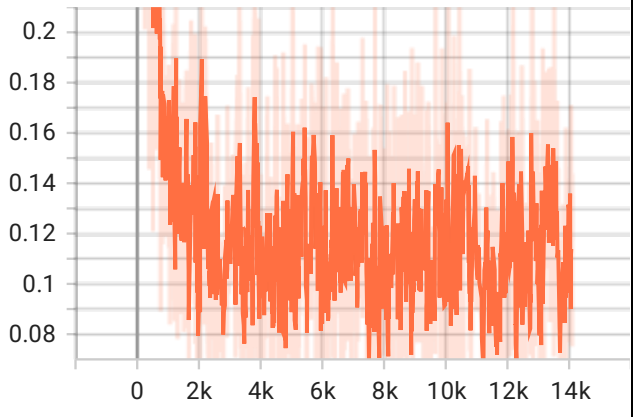
Lr = 0.0046401

Loss/train



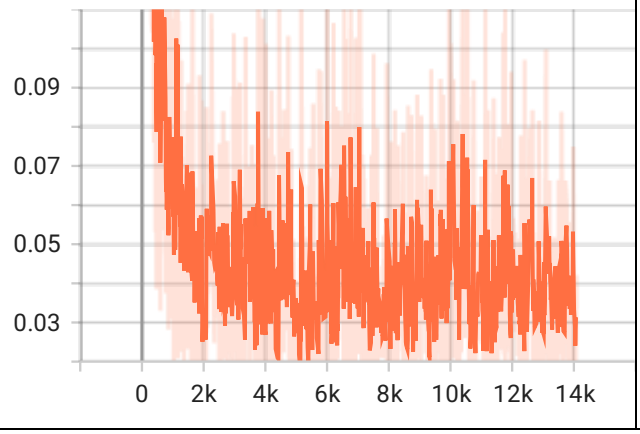
Lr = 0.01271

Loss/train



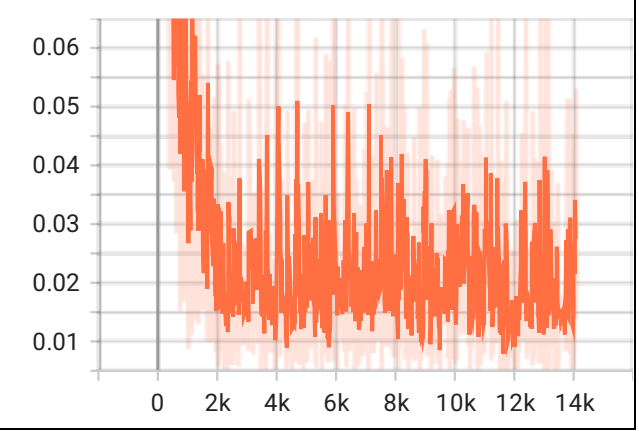
Lr = 0.05044

Loss/train



Lr = 0.2161

Loss/train

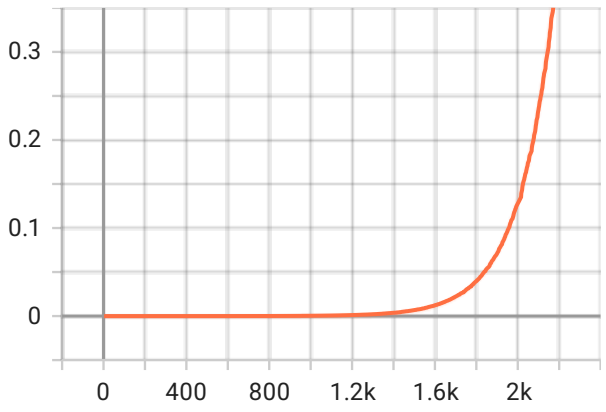


I would choose a STEPLR scheduler with an initial learning rate of 0.2161 to obtain the best results between the two schedulers for SGD.

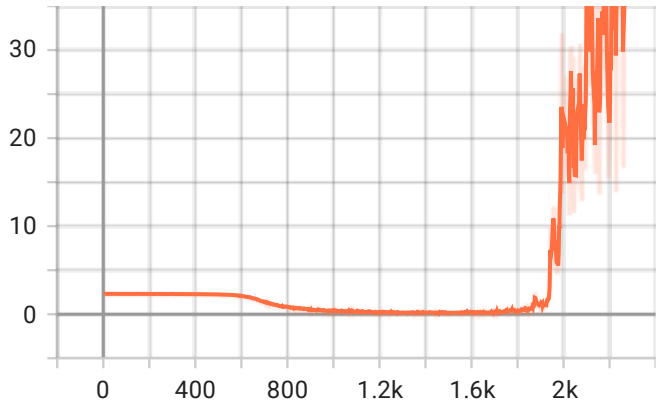
OPTIMIZER-2: ADAM

LR Range Test:

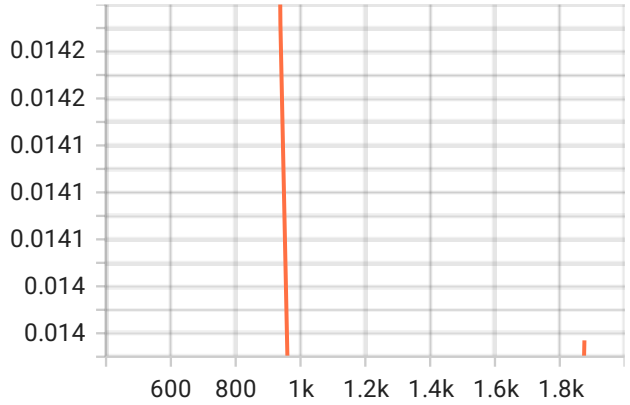
LR/train



Loss/train



Loss/test



We observe that the loss starts decreasing at a fast pace from 702 step and reaches the lowest at 1.46k step. We check the corresponding learning rates for the steps in the LR/train plot and choose the below 4 learning rates for the corresponding steps range (702 step, 1.46k step):

-lr: 6.291e-5, step: 702

-lr: 2.0318e-4, step: 901

-lr: 6.6401e-4, step: 1.102k

-lr: 5.4723e-3, step: 1.46k

Proposed LRs:

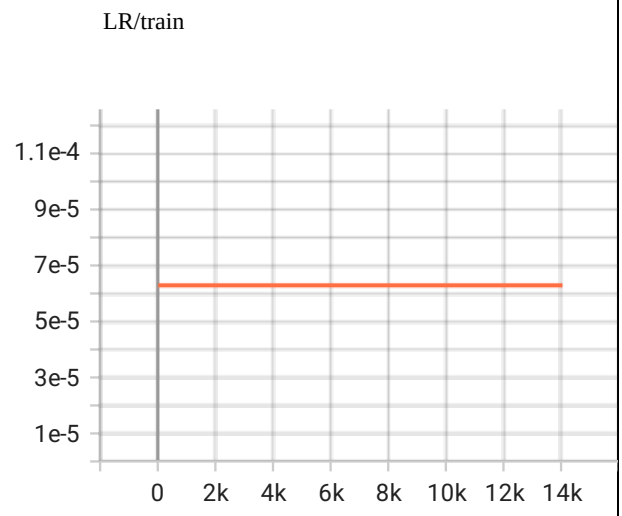
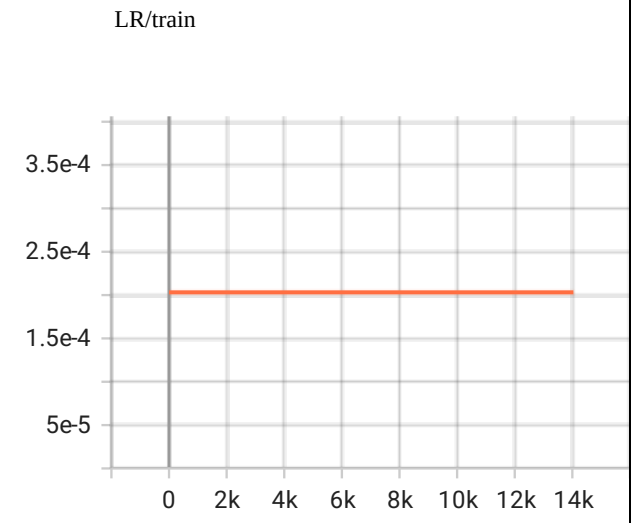
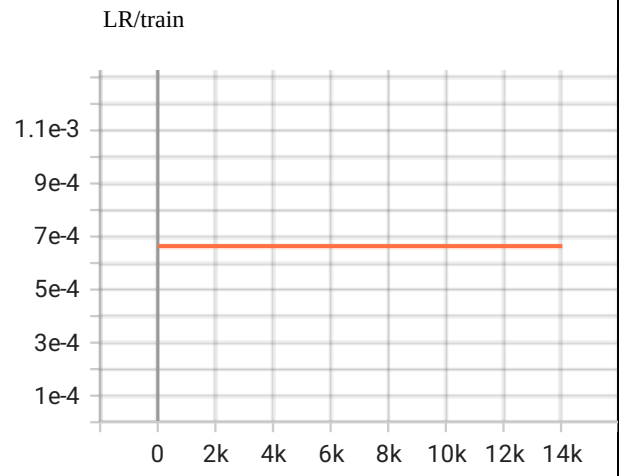
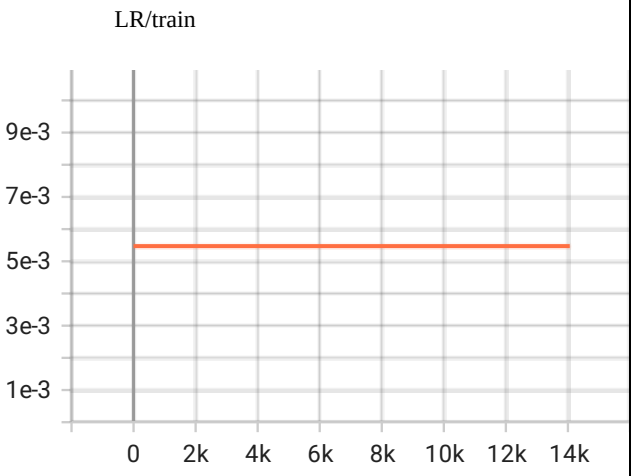
lr = [6.291e-5, 2.0318e-4, 6.6401e-4, 5.4723e-3]

We perform experiments with the proposed LRs using 2 learning rate schedulers to find the best lr value for the given optimizer:

(a) Fixed learning rate
(b) STEPLR scheduler

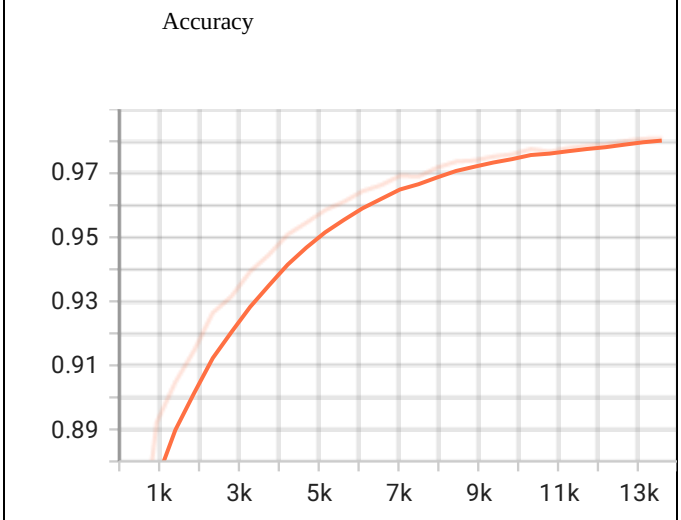
(a) Fixed learning rate:

The learning rate will be fixed here.

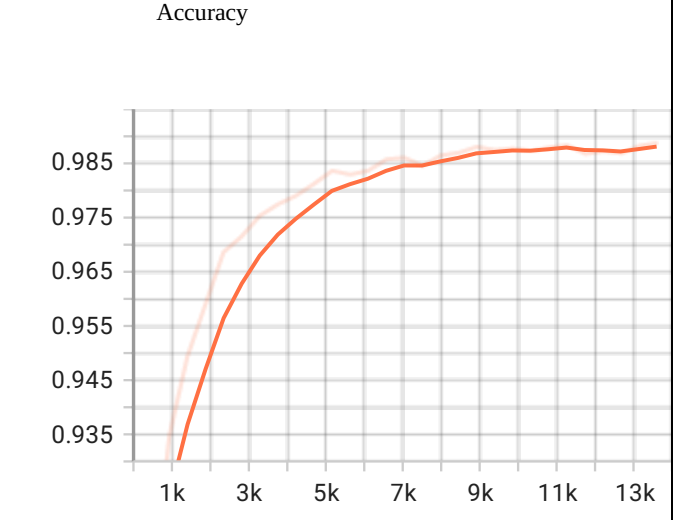
Lr = 6.291e-5 	Lr = 2.0318e-4 
Lr = 6.6401e-4 	Lr = 5.4723e-3 

We analyse the Accuracy and LR/train plots for the proposed learning rates to choose the optimal lr. We get the maximum test accuracy when lr = 2.0318e-4. We also get a smoother and lower loss for it as compared to the plots of other proposed learning rates. **We choose lr = 2.0318e-4 as the optimal learning rate for fixed learning rate for Adam.**
LR=6.291e-5 is another good candidate for optimal learning rate because of its smooth test accuracy curve and loss function.

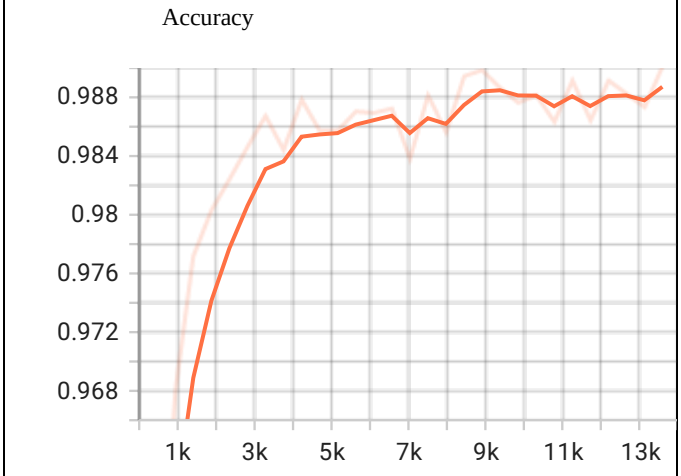
Lr = 6.291e-5



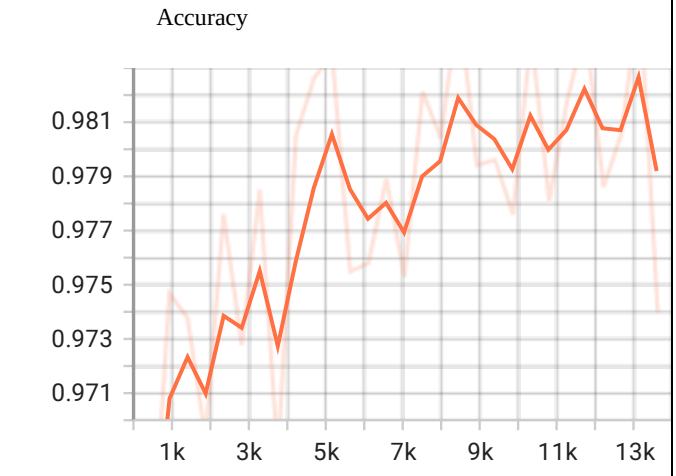
Lr = 2.0318e-4



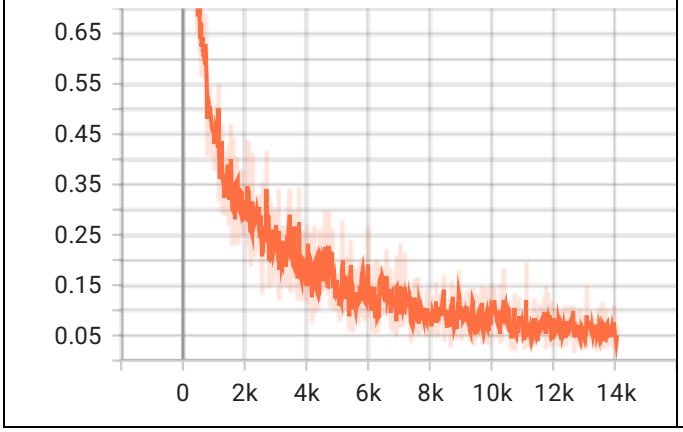
Lr = 6.6401e-4



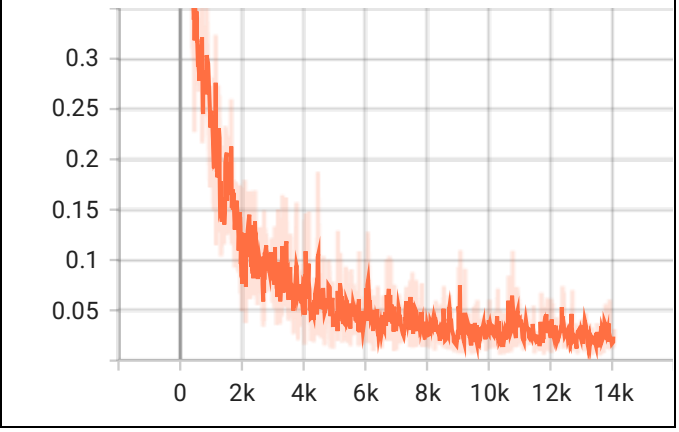
Lr = 5.4723e-3



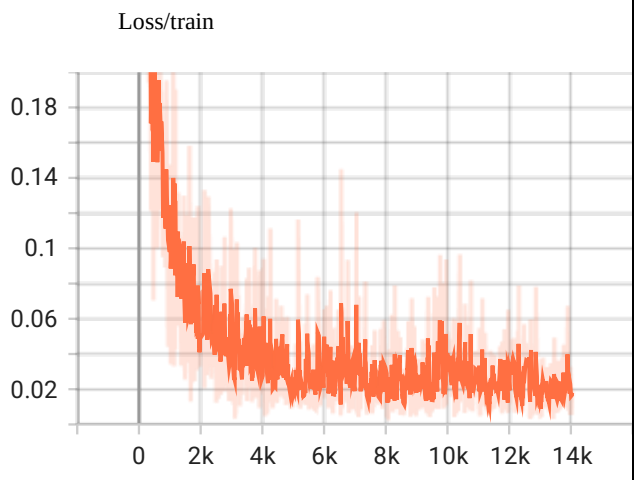
Lr = 6.291e-5



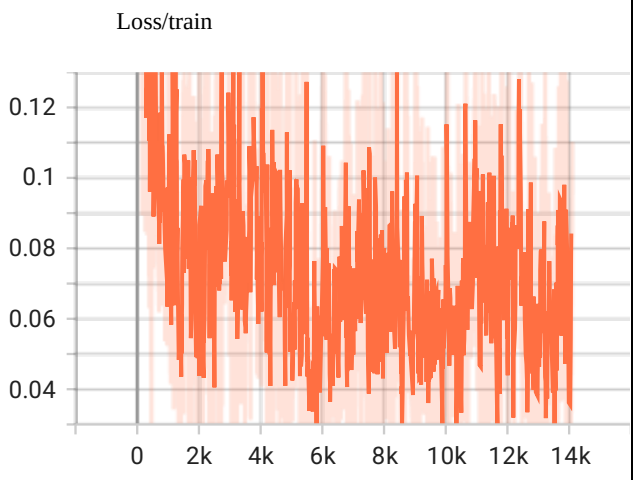
Lr = 2.0318e-4



Lr = 6.6401e-4



Lr = 5.4723e-3



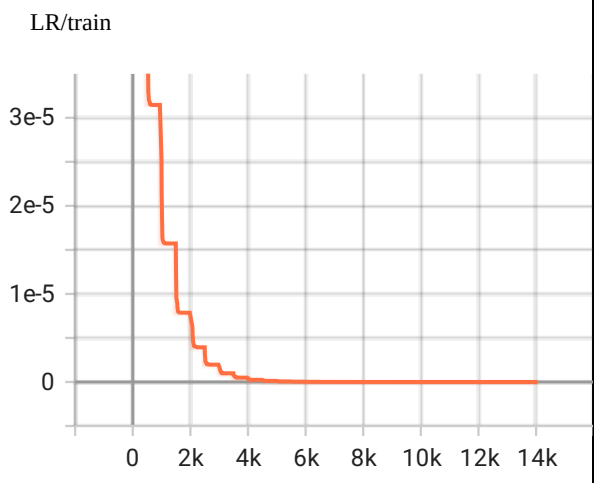
(b) STEPLR scheduler:

The STEPLR scheduler updates(decreases) the learning rate after a fixed number of steps. We use Pytorch's StepLR scheduler with the below parameters:

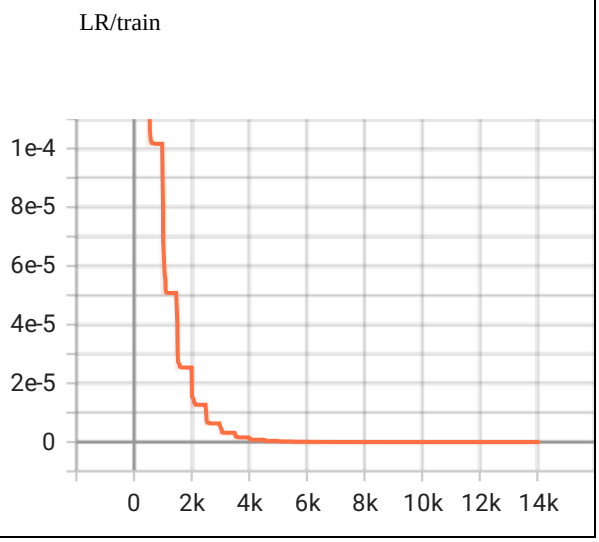
```
torch.optim.lr_scheduler.StepLR(optimizer, step_size=500, gamma=0.5, last_epoch=- 1, verbose=False)
```

This will decrease the learning rate by half after every 500 steps.

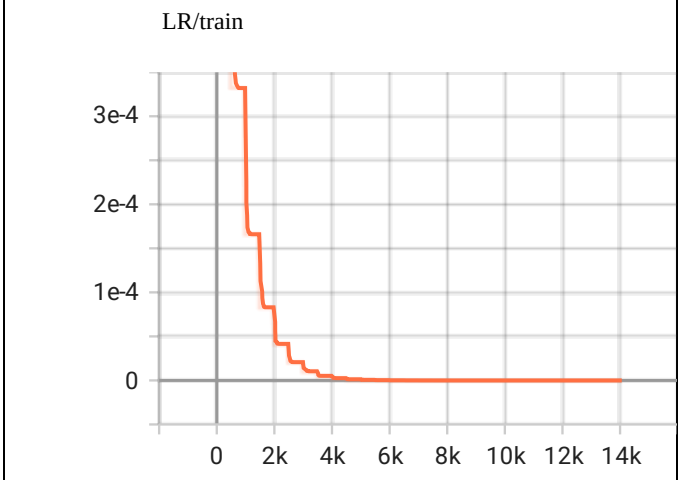
Lr = 6.291e-5



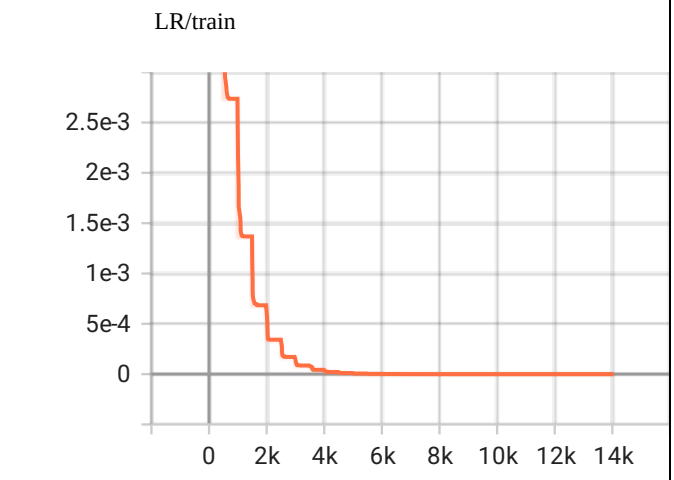
Lr = 2.0318e-4



Lr = 6.6401e-4

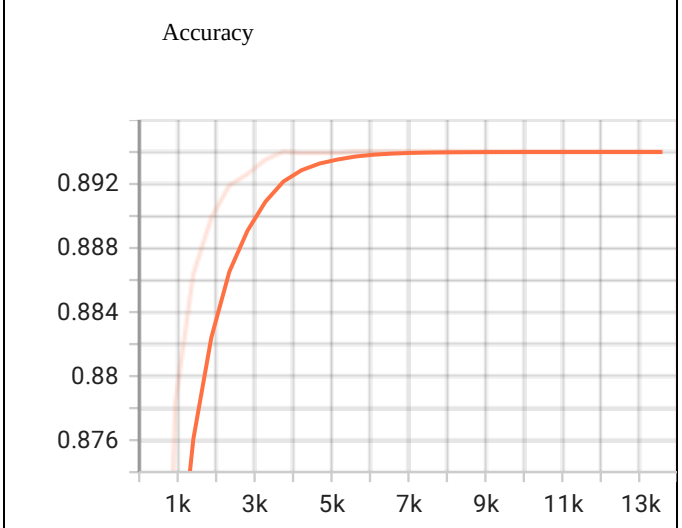


Lr = 5.4723e-3

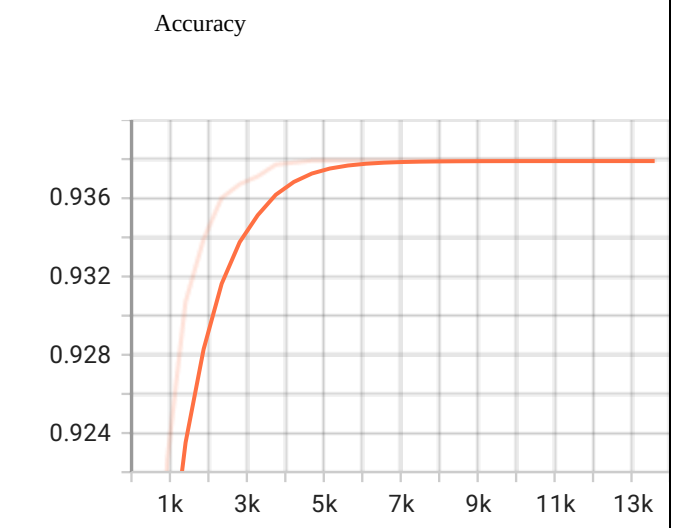


We analyse the Accuracy and LR/train plots for the proposed learning rates to choose the optimal lr. We get the maximum test accuracy when lr = 5.4723e-3. We also get a smoother and lower loss for it as compared to the plots of other proposed learning rates. **We choose lr = 5.4723e-3 as the optimal learning rate for STEPLR scheduler for Adam.**

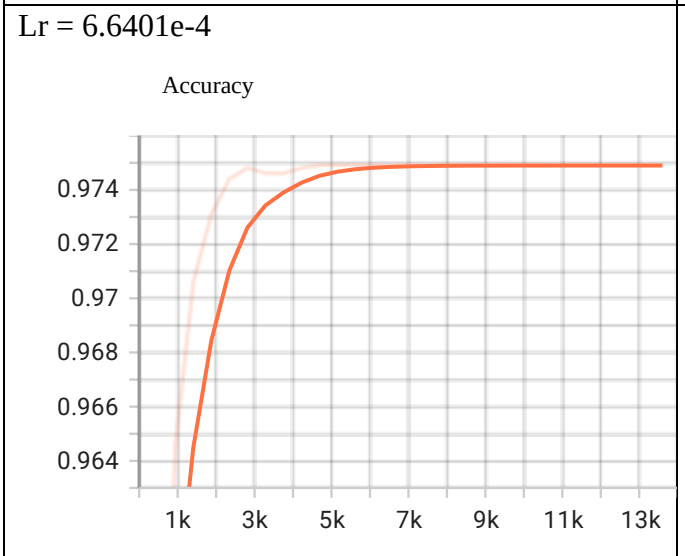
Lr = 6.291e-5



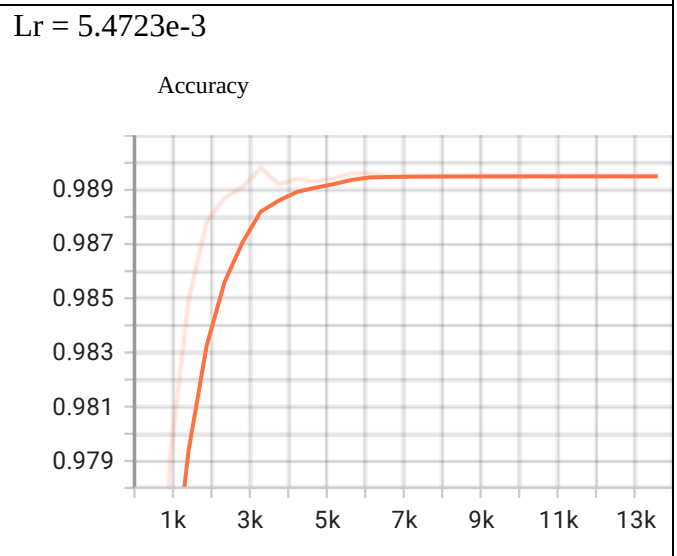
Lr = 2.0318e-4



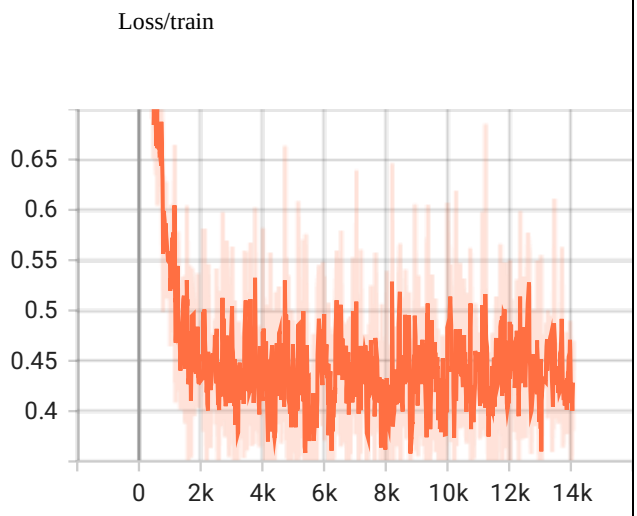
Lr = 6.6401e-4



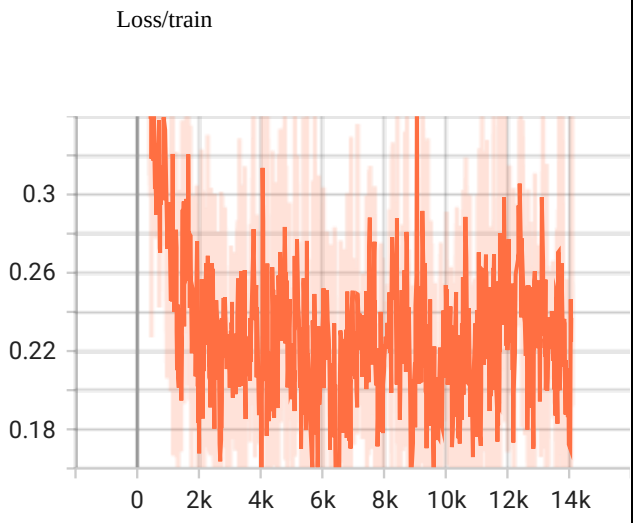
Lr = 5.4723e-3



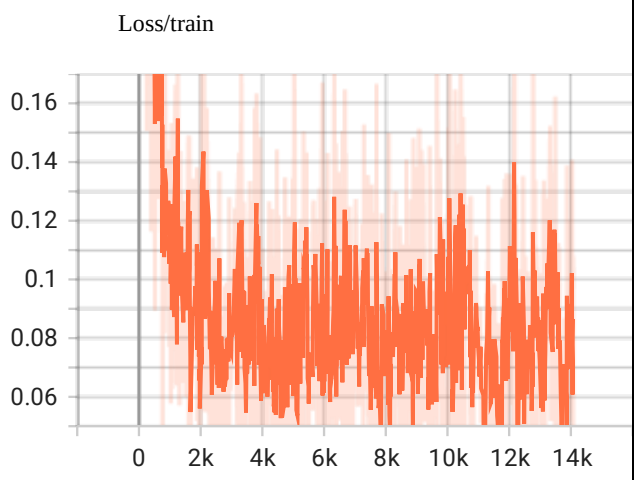
Lr = 6.291e-5



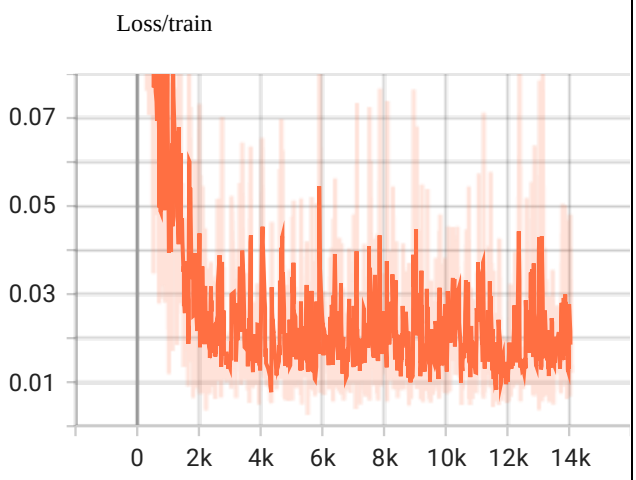
Lr = 2.0318e-4



Lr = 6.6401e-4



Lr = 5.4723e-3



I would choose a STEPLR scheduler with an initial learning rate of 5.4723e-3 to obtain the best results between the two schedulers for Adam optimizer.