MIT CSAIL

6.869/6.861 Advances in Computer Vision

Fall 2018

Miniplaces Challenge: Part2

Posted: Thursday, November 8, 2018 **Due:** Thursday 23:59, November 15, 2018

Please submit a report named ⟨your_kerberos⟩.pdf, as well as your code for Problem 2 in a zip file named ⟨your_kerberos⟩.zip. Do NOT upload the data/ folder onto the server.

Late Submission Policy: We do not accept late submissions. The submission deadline has a 50-minute soft cut-off; after midnight Thursday, submissions are penalized 2% per minute late.

Collaborators: You may do this problem set in groups of 2. However, each person should submit a separate report. Write the name of the student you collaborated with in the report. You may use the same code/report images.

You may want to train your networks on an AWS server, where you can take advantage of GPUs for faster training. We encourage you to use AWS, for this problem set and for your final project.

Problem 1 *Improving NN Performance*

For this problem set, we will be using the same MiniPlaces dataset as we did previously. The new evaluation criteria, however, will be based on Top-5 Error. As long as the target label is in one of your 5 predicted labels you generate for the image, that image will be marked as correctly classified.

We have heavily modified our code from Part I for this problem. The training portion has already been implemented for you, so running train.py will automatically start the training process and save a model after each epoch. However, the evaluation portion must be implemented. In particular, you must implement evaluating Top-1 Error (i.e. classification error) and Top-5 Error on the validation dataset.

After that, we'll take the networks you were introduced to in Mini-Places Part 1 and try and improve their performance using some common techniques.

Pick two of the techniques outlined below and implement them / modify your network.

- Add/Remove Dropout Layers

- Add Weight Decay and Change Learning Rate
- Change Initialization Techniques
- Add Skip Connections
- Change Optimizers/Add Schedulers

For each technique you implement, plot the top-5 error of your modified network against the top-5 error of the original network for both the training and validation sets. Try with a few different hyperparameter values! For example, if you chose to modify the learning rate, show what happens when you increase the learning rate and when you decrease the learning rate.

In your write up, explain each technique that you tried, and explain your results.

Problem 2 *Mini-Places Challenge*

First things first, visit http://miniplaces.csail.mit.edu/signup.html and follow the instructions to sign up with a team name. Ther submission server will email you back with a unique submission code, don't lose it!

This problem is open-ended. Using a combination of the techniques from above (or look up more NN architectures, be creative!), try to achieve the highest accuracy score you can on the test dataset. Upload your results on the test dataset to the submission server here, and then the server will report back with your test accuracy. Be wise with your submissions, as only one submission is allowed per hour.

Your test results that you upload to the website should be in a text file looking like this:
test/00000001.jpg 0 1 2 3 4
test/00000002.jpg 4 5 6 7 8
... (one for each image file)

To get full credit for this problem, you must achieve a baseline Top-5 Error of less than 30%. The top 25 teams will also receive 10% extra credit on this homework assignment. Top 5 teams will receive 20% extra credit.

For your report, make sure to describe the architecture of your NN, your hyperparameters used, as well as training details. Use a diagram if it makes your report clearer. Try to keep the report for this problem under a few paragraphs.