The University of Texas at Austin
Department of Electrical and Computer Engineering

EE379K: Data Science Lab — Fall 2017

Lab Eight

Caramanis/Dimakis Due: Monday Nov 13th, 3:00pm 2017.

Problem 1. In this problem, we start working with the Keras library and follow the tutorial `https://blog.keras.io/how-convolutional-neural-networks-see-the-world.html`. We will observe what the deep neural networks are really learning in the images. First, read through the tutorial and try to understand the steps we are going to take.

1. Load the pre-trained VGG16 convolutional neural network model without the output layer. Check-out the summary of the model. How many layers have been used in the model? How many parameters needed to be optimized?

2. We will look into the behavior of a specific layer of the network. Select the 1st filter of the 3rd convolutional layer of the 5th block of the network as the filter of interest. You can look through the demo for this part. Feed a gray image with random noise to the filter and perform gradient ascent for 20 times with the learning rate $\mu = 1.0$. Plot the output image.

3. We can extend this part for all of the layers. Feed a gray image with random noise to the first convolutional layer of each block. Note that each layer has a different number of filters. Use the range of 64 filters and plot the 12 results that maximize the activation of the corresponding filter. How are the outputs evolving as we go up in the layers? Can you find any patterns? Comment on the results.

4. We do not have to feed a random image. Feed an actual image of your choosing to the filter we used in the first part. Again, use gradient ascent for 20 times and plot the output image.

5. Now, we will look into how the network is classifying the images. For this part, we also need the output layer of the network. Load VGG convolutional network with the output layer this time. We first use $output\_index = 65$ which corresponds to 'Sea Snake' class in ImageNet. Feed a gray image with random noise to the filter corresponding to the specified output index. Perform gradient ascent 20 times and report the loss at each step. How sure your network is about the image being a sea snake? Plot the output image. Does it look like a sea snake? Why did network decide that it is a sea snake with the corresponding probability? Comment on the results.

6. Change the output index we used to $output\_index = 18$, which corresponds to 'Magpie' class in ImageNet. Perform the same tasks as in part 5. Report the losses and plot the output image. How sure your network is about the image belonging to this class? Does the output look like a magpie? Comment on the results.

Problem 2. In this problem, we will work on how data-preprocessing and augmentation can improve the performance of the neural networks. For this task, we will use CIFAR-10 dataset. We mainly follow the tutorial in `https://www.tensorflow.org/tutorials/deep_cnn`. Read the tutorial thoroughly.

1. Train the model using training data in the CIFAR-10 dataset and $cifar10\_train.py$ provided in the tutorial. Note that this script uses the 'random flip' and 'random distortion' on training images to augment the training dataset. After the training, evaluate your model using test set

provided in CIFAR-10 and $cifar10\_eval.py$ provided in the tutorial. Report your results.

2. Now, remove 'random flip' pre-processing step. Train and evaluate your model. Report your results. What do you think was the effect of the 'random flip'?

3. Now, remove 'random distortion' pre-processing step. Train and evaluate your model. Report your results. What do you think was the effect of the 'random distortion'?

4. Comment on your findings. Did using pre-processing steps improve the overall performance? Why or why not pre-processing improved the performance of the model? Provide your insights.

5. What other pre-processing methods can be applied to the training or test set to improve the performance of the learning models? Propose a possible pre-processing method. You do not have to apply it.

Problem 3. (Starting your class project) Report what you are thinking of doing for your class project. Assemble your team and discuss which datasets you plan to use. If you plan to scrape some data, show some preliminary results. Discuss what you plan to do in your data analysis. Final project presentations will be on Monday 12/11 (last day of classes) 5-8pm.