DeepLearning_Lesson4: Image Classification with CNN

Please don't forget to submit your feedback after the class. This helps a lot in increasing effectiveness of the course.
Use the following link to submit your feedback:
https://docs.google.com/forms/d/e/1FAIpQLSdmJkDgBMxr4qv73c9y5k1jtky44-sMmOI1v1jFtNEbUJ6H9A/viewform

Lesson Overview:
In this lesson, we are going to discuss Image classification with CNN.

Use Case Description:
Image Classification with CNN
1. Training the model
2. Evaluating the model

Programming elements:
1. About CNN
2. Hyperparameters of CNN
3. Image classification with CNN

Source Code:
https://umkc.box.com/s/0jkz2eljon8v374xgy1f6b4ooni0bx3t

In class programming:
1. Follow the instruction below and then report how the performance changed.(apply all at once)

- Convolutional input layer, 32 feature maps with a size of 3×3 and a rectifier activation function.
- Dropout layer at 20%.
- Convolutional layer, 32 feature maps with a size of 3×3 and a rectifier activation function.
- Max Pool layer with size 2×2.
- Convolutional layer, 64 feature maps with a size of 3×3 and a rectifier activation function.
- Dropout layer at 20%.
- Convolutional layer, 64 feature maps with a size of 3×3 and a rectifier activation function.
- Max Pool layer with size 2×2.
- Convolutional layer, 128 feature maps with a size of 3×3 and a rectifier activation function.
- Dropout layer at 20%.
- Convolutional layer,128 feature maps with a size of 3×3 and a rectifier activation function.
- Max Pool layer with size 2×2.
- Flatten layer.
- Dropout layer at 20%.
- Fully connected layer with 1024 units and a rectifier activation function.
- Dropout layer at 20%.
- Fully connected layer with 512 units and a rectifier activation function.
- Dropout layer at 20%.
- Fully connected output layer with 10 units and a softmax activation function

Did the performance change?
2. Visualize the graph and loss with TensorBoard
3. predict the first 4 image of the test data. Then, print the actual label for those 4 images (label means the probability associated with them) to check if the model predicted correctly or not

**MNIST dataset: is a data set

ICP Submission Guidelines (for In Class students):

1. ICP Submission is in pairs of two students.
2. Once completed, must be presented to TA or Instructor before the completion of the class
3. Submission after class is considered as a late submission. (Check the late submission policy in the syllabus)
4. ICP Code with brief explanation should be pushed to GitHub. Submit GitHub link through the Feedback Form: https://docs.google.com/forms/d/e/1FAIpQLSdmJkDgBMxr4qv73c9y5k1jtky44-sMmOI1v1jFtNEbUJ6H9A/viewform

Online Submission Guidelines (for Online students):

1. Submit your source code and documentation to GitHub and represent the work through wiki page properly (submit your screenshots as well. The screenshot should have both the code and the output)
2. Comment your code appropriately
3. Video Submission (2 – 3 min video showing the demo of the ICP, with brief voice over on the code explanation)
4. Submission after class is considered as a late submission. (Check the late submission policy in the syllabus)
5. Use the following Google link to submit your ICP # (GitHub wiki page link for ICP #): https://docs.google.com/forms/d/e/1FAIpQLSdmJkDgBMxr4qv73c9y5k1jtky44-sMmOI1v1jFtNEbUJ6H9A/viewform

Evaluation Criteria:

1. Completeness of Features
2. Code Quality (https://en.wikipedia.org/wiki/Best_coding_practices)
3. Time
4. Feedback Submission

Note: *Cheating, plagiarism, disruptive behavior and other forms of unacceptable conduct are subject to strong sanctions in accordance with university policy. See detailed description of university policy at the following URL: https://catalog.umkc.edu/special-notices/academic-honesty/*