CS7150 (Spring 2025): Practice Exam

Remarks from Professor Jiang

- In addition to this practice exam, review the 3 in-class quizzes as well.
- The midterm will have 4 parts as in this practice exam.
- But the same question may be asked in a different way.
- So, instead of memorizing the answers, the most effective way of preparing for the midterm is to understand the concepts.
- The number of questions in each part may be different in each part in the midterm.
- **The midterm is closed book *i.e.*, no laptops, notes, textbooks, cheat sheet etc. during the exam.**
- Enough space will be provided for you to write down and derive your answers in the midterm.
- No matter how many points we have in total, it will be converted to 15 out of 100 toward your final grade of this course.

1 Part 1: Questions with multiple choices (2 points for each question)

1) Which of the following would you consider to be valid activation functions (performed in an element-wise manner) to train neural networks

   (a) $f(x) = \min(0, x)$

   (b) $f(x) = x + 1$

   (c) $f(x) = \begin{cases} \min(x, 0.1x), & \text{if } x \geq 0 \\ \min(x, 0.1x), & \text{if } x < 0 \end{cases}$

   (d) $f(x) = \begin{cases} \max(x, 0.1x), & \text{if } x \geq 0 \\ \min(x, 0.1x), & \text{if } x < 0 \end{cases}$

2) Which of the following technique(s) can be used to reduce overfitting?

   (a) Batch Normalization

   [✓ (b)] Dropout

   [✓ (c)] Data augmentation

   (d) Stochastic Gradient Descent (SGD) with momentum

3) What can you do if you see underfitting?

   (a) Increase the number of training data

   (b) Increase the number of model parameters

   (c) Use a large learning rate

   (d) Add residual connections in your model architecture

4) If you input image's shape is $16 \times 32 \times 32$ (16 is the number of channels and 32×32 is the spatial dimension), how many parameters are there in a single 3×3 convolution filter, including bias?

   (a) 9

   (b) 10

   (c) 145

   (d) 10,240

5) Which of the following are true about Batch Normalization (BN)?

   (a) BN is another way of doing Dropout

   (b) BN makes training faster

   (c) BN can only be used in the training phase

   (d) BN is a non-linear transformation that centers the output around the origin

6) Which of the following are true about a Convolution layer?

   (a) The number of weights depends on the number of channels (depth) of the input volume

   (b) The number of biases is equal to the number of filters

   (c) The total number of parameters is dependent on the stride

   (d) The total number of parameters is dependent on the padding

2 Part 2: Short answers (2 or 3 points for each question)

Answer each of the question **concisely** with 2-4 sentences.

1) Alice is designing a fully-connected neural network. But somehow she forgets to add non-linear activation functions in-between. Why may happen to the neural network?

△ 2) Name two scenarios where you may need to use the Sigmoid function in your network. < output
LSTM

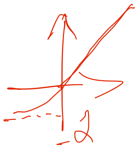
3) You come across the following activation function,

$$f(x) = \begin{cases} 1, & \text{if } x \geq 0, \\ 0, & \text{if } x < 0. \end{cases} \quad (1)$$

A friend highly recommends you to use it in your neural network together with the Stochastic Gradient Descent (SGD) optimizer plus momentum. Would you follow their advise? Why or why not?

△ 4) Recall the ReLU activation function is $ReLU(x) = \max(0, x)$. Consider an alternative to ReLU called Exponential Linear Unit (ELU).

$$f(x) = \begin{cases} x, & \text{if } x \geq 0, \\ \alpha(e^x - 1), & \text{if } x < 0, \end{cases} \quad \begin{matrix} 1 \\ \alpha e^x \end{matrix} \quad (0, ReLU) \quad (2)$$



where α is a scalar. It is a hyper parameter. i) Derive the gradient of ELU. ii) Name two advantages of ELU over ReLU.

} grad in neg
smoother

5) What is the definition of the cross-entropy loss?

6) Briefly explain why gradient vanishing problem may happen in training Convolutional Neural Networks and describe two methods to tackle it. Assume we are training the networks with SGD plue momentum.

7) Name two advantages of using the SGD optimizer with momentum over without momentum.

8) Bob made the following statement: a fully-connected layer is essentially equivalent to a convolution layer. Is it correct? Why or why not?

△ 9) Name three advantages of using Convolutional Neural Networks over Fully-connected Neural Networks to process images. 1. spatial structure 2. arbitratry reso 3. equivariance w/ translation

10) What are the hyper parameters of a Convolution layer?

11) Name one way of achieving downsampling in Convolutional Neural Networks (CNNs)? Why is downsampling essential in CNNs (name two reasons)?

12) Name three data augmentation techniques that we can use to train neural networks to recognize different breeds of dogs.

△ 13) Suppose the shape of input to a Convolution layer is $C_{in} \times H \times W$. Instead of using a regular convolution layer, one can use group-based convolution, where C_{out} kernels are used, each kernel has the shape of $K \times K$, and the convolution operations are divided into G groups. What is the number of parameters? What may happen if we increase G?

14) You have a dataset $\mathcal{D}_1$ with 1 million labeled samples for classification and another dataset $\mathcal{D}_2$ with 100 labeled samples. Your friend trains a deep neural network with randomly initialized weights (from scratch) on $\mathcal{D}_2$. You decide to train a model on $\mathcal{D}_1$ first and then do transfer learning on $\mathcal{D}_2$ then. State one problem your friend may likely to find with their approach. Explain how your approach solves the same problem.

$$\frac{C_{out}}{G} \times \left(\frac{C_{in}}{\cancel{G}} \times K \times K + 1\right) \cancel{G}$$

15) What is the problem, even in a very shallow network, with initializing weights to all zeros?

16) What are the advantages of Recurrent Neural Networks over Convolutional Nueral Networks for processing text?

▷ 17) What are the advantages of Transformer over Recurrent Neural Networks for processing text?

① deep

② recurrence

▷ 18) Can Transformer be used to process images? Explain your answer.

▷ 19) What are the hyper parameters of a Transformer model?

heads

stacks

dropout

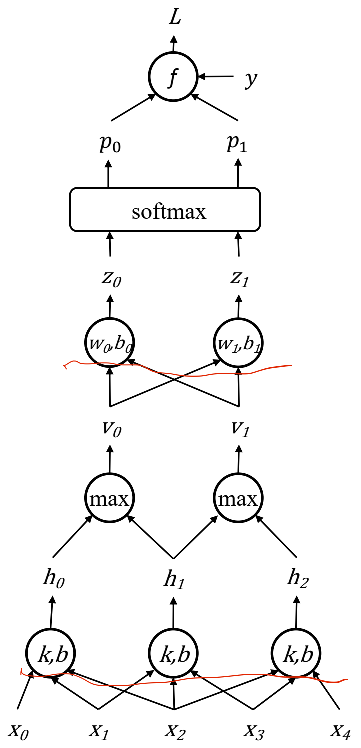
W^Q, W^K, W^V

$$\text{Softmax}\left(\frac{QK^T}{\sqrt{d}}\right) V$$

output dimension

hidden dimension

3 Part 3: Gradient Backpropagation (the point for each question may vary between 2 and 5 points)



$$L = -\log p_y$$

$$p_i = \frac{e^{z_i}}{\sum_j e^{z_j}}$$

$$\begin{bmatrix} z_0 \\ z_1 \end{bmatrix} = \begin{bmatrix} w_{00} & w_{01} \\ w_{10} & w_{11} \end{bmatrix} \begin{bmatrix} v_0 \\ v_1 \end{bmatrix} + \begin{bmatrix} b_0 \\ b_1 \end{bmatrix}$$

$$\begin{bmatrix} v_0 \\ v_1 \end{bmatrix} = \begin{bmatrix} \max(h_0, h_1, 0) \\ \max(h_1, h_2, 0) \end{bmatrix}$$

$$\begin{bmatrix} h_0 \\ h_1 \\ h_2 \end{bmatrix} = \begin{bmatrix} k_0 & k_1 & k_2 & 0 & 0 \\ 0 & k_0 & k_1 & k_2 & 0 \\ 0 & 0 & k_0 & k_1 & k_2 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} b \\ b \\ b \end{bmatrix}$$

$$\mathcal{L} = -\log P_1$$
$$= -\log \frac{e^{z_1}}{e^{z_0}+e^{z_1}}$$
$$= -\log e^{z_1} + \log(e^{z_0}+e^{z_1})$$
$$= -z_1 + \log(e^{z_0}+e^{z_1})$$

$$\frac{\partial L}{\partial z_0} = \frac{e^{z_0}}{e^{z_0}+e^{z_1}}$$
$$= P_0$$

$$\frac{\partial L}{\partial z_1} = -1 + \frac{e^{z_1}}{e^{z_0}+e^{z_1}}$$
$$= -1 + P_1 = -P_0$$

Figure 1: Computational graph a simple 1D convolutional network.

Consider the simple 1D convolutional network show in Fig. 1, where all the variables are scalars. Suppose the ground-truth label $y = 1$.

1) Does this network have any non-linearity activation functions?

2) What are the learnable parameters?

3) Determine $\frac{\partial L}{\partial z_0}$ and $\frac{\partial L}{\partial z_1}$.

$$\frac{\partial L}{\partial W_{00}} = \frac{\partial L}{\partial z_0} \cdot \frac{\partial z_0}{\partial W_{00}} = P_0 \cdot V_0$$

4) Determine $\frac{\partial L}{\partial w_{00}}$, $\frac{\partial L}{\partial w_{01}}$, and $\frac{\partial L}{\partial b_1}$.

5) Suppose $\frac{\partial L}{\partial v_0} = \delta_0$ and $\frac{\partial L}{\partial v_1} = \delta_1$, determine $\frac{\partial L}{\partial h_0}$, $\frac{\partial L}{\partial h_1}$, and $\frac{\partial L}{\partial h_2}$.

6) Suppose $\frac{\partial L}{\partial h_0} = \delta_0$, $\frac{\partial L}{\partial h_1} = \delta_1$, and $\frac{\partial L}{\partial h_2} = \delta_2$, determine $\frac{\partial L}{\partial k_0}$, $\frac{\partial L}{\partial k_1}$, $\frac{\partial L}{\partial k_2}$, and $\frac{\partial L}{\partial b}$.

$$\frac{\partial L}{\partial h_0} = \frac{\partial L}{\partial V_0} \cdot \frac{\partial V_0}{\partial h_0} = \delta_0 \cdot \begin{cases} 1 , & h_0 > h_1 > 0 \\ 0 & \text{otherwise} \end{cases}$$

$$\frac{\partial L}{\partial k_0} = \frac{\partial L}{\partial h_0} \cdot \frac{\partial h_0}{\partial k_0} + \frac{\partial L}{\partial h_1} \cdot \frac{\partial h_1}{\partial k_0} + \frac{\partial L}{\partial h_2} \cdot \frac{\partial h_2}{\partial k_0}$$

$$= \delta_0 \cdot x_0 + \delta_1 x_1 + \delta_2 x_2$$

4 Part 4: Convolution Arithmetic (each row is about 1 or 2 points)

Consider the convolutional neural network defined by the layers in the left column below. Fill in the shape of the output volume and the number of parameters at each layer. You can write the shapes in the format of 3 × 128 × 64 (3 being the channel dimension, 128 being the height, and 64 being the width).

Notation:

- CONVx-y-N denotes a convolutional layer with N filters and kernel height and width equal to x. Padding is y, and stride is 1.
- POOL-n denotes a $n \times n$ max-pooling layer with stride of 2 and 0 padding.
- FLATTEN flattens its input.
- FC-N denotes a fully-connected layer with N neurons/output.

Layer	Output Volume Dimension	Number of Parameters
Input	$3 \times 32 \times 32$	0
CONV5 − 0 − 16	16×28×28	16×(3×5×5+1)
ReLU	16×28×28	0
POOL-2	16×14×14	0
Batch Normalization	16×14+14	32
CONV3 − 1 − 32	32×14×14	32×(16×3×3+1)
ReLU	32×14×14	0
POOL-2	32×7×7	0
FLATTEN	K	0
FC-10	10	10×(K+1)