CS583-A: Deep Learning Name: May. 5, 2022

Final Exam

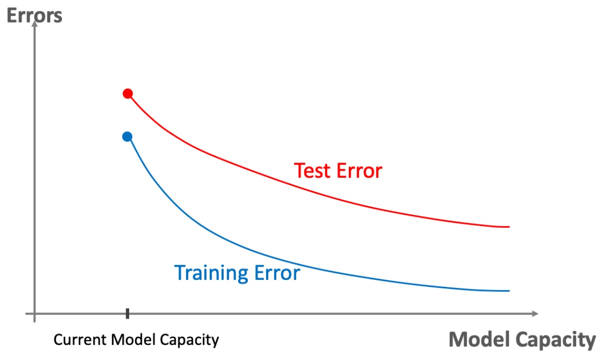
Final exam will be done individually. Use of partial or entire solutions obtained from others or online is strictly prohibited.

- There will be 8 pages in this exam (including this cover sheet)
- This is a **SEMI-OPEN BOOK** exam. You can only use our lecture slides. You **CANNOT** use other books or online materials.
- For **MULTIPLE CHOICE** questions, choose ALL options that apply. Selecting all of the correct options will get full credits. Partial credits will be given to correct but incomplete options. Answers containing any incorrect options will get 0 points.
- Work efficiently and independently.
- You have 150 minutes.
- Good luck!

Question	Topic	Max. score	Score
1	True/False	10	
2	Single/Multiple Choice	30	
3	Value Function	8	
4	RNN/LSTM	20	
5	Attention	14	
6	Short Answer Questions	18	
Total		100	

1. **True/False Question** (10 points)

   (a) (2 pts) **True** or **False**: As we increase the model capacity, the training error and test error vary as in the figure **below**. The current model is overfitting.



   (b) (2 pts) **True** or **False**: Let $f(\mathbf{x}; \mathbf{W})$ be a neural network which maps an image, $\mathbf{x}$, to a probability distribution over 10 classes. Let $\mathbf{W}^\star$ be the parameter matirx/tensor learned from a set of training samples. Let $\mathbf{x}^\star$ and $\mathbf{y}^\star$ be a pair of **real** image and **real** target. We want to generate a fake image $\tilde{\mathbf{x}}$ that makes the neural network err, i.e., having wrong prediction. We can compute $\tilde{\mathbf{x}}$ by:

$$\tilde{\mathbf{x}} \;=\; \text{argmax}_{\mathbf{x}}\, \mathsf{CrossEntropy}\big(\mathbf{y}^\star,\, f(\mathbf{x}; \mathbf{W}^\star)\big), \qquad \mathsf{s.t.}\ \|\mathbf{x} - \mathbf{x}^\star\|_1 \leq \delta.$$

   (c) (2 pts) **True** or **False**: The above attack in (b) is targeted attack.

   (d) (2 pts) **True** or **False**: A multi-layer neural network with *linear* activation functions is more expressive than a single-layer neural network with *linear* activation functions.

   (e) (2 pts) **True** or **False**:Let $p(x)$ and $q(x)$ be the probability density functions of two different Gaussian distributions. The KL-divergence can be used to measure the distance between two distributions and satisfies $\mathtt{KL}(p\|q) = \mathtt{KL}(q\|p)$.

2. **Single or Multiple Choices** (30 points)

(a) (3 pts) Which of the followings is **NOT** an unsupervised learning method?

A. Principal component analysis.
B. Autoencoder.
C. K-means clustering.
D. Polynomial regression.

(b) (3 pts) We want to train a recurrent neural network on a small dataset for sentiment analysis. Which of the followings helps alleviate overfitting?

A. Self-attention.
B. Stacked LSTM.
C. Bidirectional LSTM.
D. Pretrain the embedding layer on a large dataset.

(c) (3 pts) The input is a $20 \times 15 \times 3$ tensor. What is the number of $5 \times 4 \times 3$ patches in the input tensor? Here we set stride $= 1$ and no zero-padding.

A. 1×1.
B. 4×3.
C. 5×4.
D. 15×11.
E. 16×12.
F. 20×15.

(d) (3 pts) We want to learn a sequence-to-sequence model for machine translation. We hope the model will be less likely to "forget". Which of the following approaches is helpful?

A. In the decoder network, use bidirectional LSTM instead of LSTM.
B. In the encoder network, use bidirectional LSTM instead of LSTM.
C. In the encoder network, use stacked LSTM instead of LSTM.
D. In the decoder network, use stacked LSTM instead of LSTM.

(e) (3 pts) For training *denoising* autoencoder to remove noise from images. The training data should be:

A. Clean images as inputs, and clean images as targets.
B. Noisy images as inputs, and noisy images as targets.
C. Clean images as inputs, and noisy images as targets.
D. Noisy images as inputs, and clean images as targets.

(f) (3 pts) We want to train a GAN for generating fake images. Ideally, at the end of training, the discriminator's classification accuracy should be ____?____

A. almost 0%.
B. around 50%.
C. somewhere between 50% and 100%.
D. almost 100%.

(g) (3 pts) For a video game, the action space is {"left", "right", "up"}. The policy function predicts:

$$\pi(\text{"left"}|s_t) = 0.01, \qquad \pi(\text{"right"}|s_t) = 0.04, \qquad \pi(\text{"up"}|s_t) = 0.95.$$

Which of the following statements is correct?

A. "left" is the agent's action a_t.
B. "right" is the agent's action a_t.
C. "up" is the agent's action a_t.
D. Any of "left", "right", and "up" can be the agent's action a_t.

(h) (3 pts) Suppose we seek to predict housing price (a positive number) using a multi-layer perceptron. Which of the followings is the best choice for the activation function of the output layer?

A. No activation function (identity function).
B. ReLU.
C. Sigmoid function.
D. Softmax function.

(i) (3 pts) Let $\mathbf{h}_t$ be the hidden state of a SimpleRNN model. What is the range of the entries of $\mathbf{h}_t$? (Choose the most precise one from followings.)

A. In $(-\infty, +\infty)$.
B. In $(-1, 1)$.
C. In $(0, 1)$.
D. In $(0, +\infty)$.

(j) (3 pts) Which of the following functions can be used to control the agent?

A. Action-value function $Q_\pi(s, a)$.
B. Optimal action-value function $Q^\star(s, a)$.
C. State-value function $V_\pi(s)$.
D. Policy function $\pi(a|s)$.

3. **Value function** (8 pts) For a video game, the action space is $\{$“left”, “right”, “up”$\}$. The policy function predicts:

$$\pi(\text{“left”}|s_t) = 0.2, \qquad \pi(\text{“right”}|s_t) = 0.3, \qquad \pi(\text{“up”}|s_t) = 0.5.$$

The action-value function predicts:

$$Q_\pi(s_t, \text{“left”}) = 100, \qquad Q_\pi(s_t, \text{“right”}) = 200, \qquad Q_\pi(s_t, \text{“up”}) = 300.$$

Compute the state-value function $V_\pi(s_t)$.(Hint: $V_\pi(s_t) = \mathbb{E}_{a \sim \pi(\cdot|s_t)}\big[Q_\pi(s_t, a)\big]$.)

4. **RNN/LSTM** (20 pts) The following code builds a recurrent neural network. Based on the code, please answer the following questions (for number of parameters, you don't need to consider **bias** term). Please **write down the steps for full credits**.

```
1  from keras.models import Sequential
2  from keras.layers import Embedding, LSTM, Dense, Flatten
3
4  voc_size = 10000
5  shape_x = 50
6  seq_length = 100
7  shape_h = 40
8
9  model = Sequential()
10 model.add(Embedding(voc_size, shape_x, input_length=seq_length))
11 model.add(LSTM(shape_h, return_sequences=True))
12 model.add(LSTM(shape_h, return_sequences=True))
13 model.add(Flatten())
14 model.add(Dense(1, activation='sigmoid'))
15
16 model.summary()
```

(a) (3 pts) Line 10 is an embedding layer. What is the output shape of this layer?

(b) (3 pts) What is (roughly) the number of parameters in the embedding layer?

(c) (3 pts) Line 11 is the first LSTM layer. What is the output shape of this layer?

(d) (3 pts) What is (roughly) the number of parameters in the first LSTM layer?

(e) (3 pts) Line 12 is the second LSTM layer. What is the output shape of this layer?

(f) (3 pts) Line 13 is the Flatten layer. What is the output shape of this layer?

(g) (2 pts)What is (roughly) the number of parameters in the Flatten layer?

5. **Attention** (14 pts) We consider the single-head attention in the Transformer model. Let the three input matrices be $\mathbf{Q} \in \mathbb{R}^{20\times 100}$, $\mathbf{K} \in \mathbb{R}^{30\times 20}$, and $\mathbf{V} \in \mathbb{R}^{30\times 20}$. Let $\widetilde{\mathbf{Q}} = \mathbf{W}_q\mathbf{Q} \in \mathbb{R}^{10\times 100}$, $\widetilde{\mathbf{K}} = \mathbf{W}_k\mathbf{K} \in \mathbb{R}^{10\times 20}$, and $\widetilde{\mathbf{V}} = \mathbf{W}_v\mathbf{V} \in \mathbb{R}^{10\times 20}$. The output of the single-head attention layer is $\mathbf{C} = \widetilde{\mathbf{V}} \cdot \mathsf{softmax}\big(\widetilde{\mathbf{K}}^T \cdot \widetilde{\mathbf{Q}}\big)$.

(a) (3 pts) What is the total number of trainable parameters in this single-head attention layer.

(b) (3 pts) What is the output shape of the single-head attention layer.

(c) (4 pts) We want to build a multi-head attention by combining m single-head attention modules. To make sure the output shape is the same as the input, $\mathbf{Q}$, we must set $m =$________.

(d) (4 pts) What is the total number of trainable parameters in this multi-head attention.

6. **Short Answers** (18 pts) Please try to use your **OWN** words and understandings. Directly copy-paste from slides will get NO credit.

   (a) (6 pts) Describe the randomness in reinforcement learning, and where do they come from.

   (b) (6 pts) Describe the targeted attack and untargeted attack, and their difference.

   (c) (6 pts) Describe the difference between autoencoder and variational autoencoder (VAE). Can we train VAE with only generation loss (i.e., the L2 or cross entropy loss between generated data and training data)? Why or Why not.