축하합니다! 통과하셨습니다!

받은 학점 80% **최신 제출물 학점** 80% **통과 점수:** 80% 이상

다음 항목으로 이동 7 시간 58 분 후에 과제를 다시 풀어보세요.

1. Which of the following do you agree with? 1 / 1점

- (●) Face recognition requires K comparisons of a person's face.
- ( ) Face verification requires K comparisons of a person's face.
- ( ) Face recognition requires comparing pictures against one person's face.

더 보기

맞습니다
Correct, in face recognition we compare the face of one person to K to classify the face as one of those K or not.

2. Why do we learn a function $d(img1, img2)$ for face verification? (Select all that apply.) 1 / 1점

- [x] We need to solve a one-shot learning problem.

  Correct
  This is true as explained in the lecture.

- [ ] This allows us to learn to predict a person's identity using a softmax output unit, where the number of classes equals the number of persons in the database plus 1 (for the final "not in database" class).
- [x] This allows us to learn to recognize a new person given just a single image of that person.

  Correct
  Yes.

- [ ] Given how few images we have per person, we need to apply transfer learning.

더 보기

맞습니다
Great, you got all the right answers.

3. You want to build a system that receives a person's face picture and determines if the person is inside a workgroup. You have pictures of all the faces of the people currently in the workgroup, but some members might leave, and some new members might be added. To train a system to solve this problem using the triplet loss you must collect pictures of different faces from only the current members of the team. True/False? **1 / 1점**

- True
- False

더 보기

맞습니다
Correct. Although it is necessary to have several pictures of the same person, it is not absolutely necessary that all the pictures only come from current members of the team.

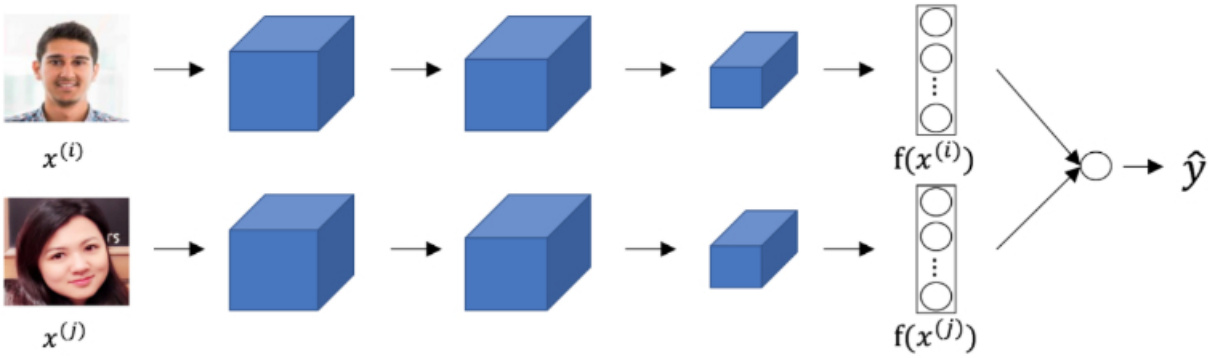
4. Which of the following is a correct definition of the triplet loss? Consider that $\alpha > 0$. (We encourage you to figure out the answer from first principles, rather than just refer to the lecture.) **0 / 1점**

- $max(||f(A) - f(P)||^2 - ||f(A) - f(N)||^2 + \alpha, 0)$
- $max(||f(A) - f(P)||^2 - ||f(A) - f(N)||^2 - \alpha, 0)$
- $$max(||f(A)-f(N)||^2 - ||f(A)-f(P)||^2 + \alpha, 0)$$
- $$max(||f(A)-f(N)||^2 - ||f(A)-f(P)||^2 - \alpha, 0)$$

더 보기

틀립니다
No, with this formula you would minimize $||f(A) - f(N)||^2$ instead of maximizing it.

5. Consider the following Siamese network architecture: 1 / 1점



The upper and lower neural networks have different input images, but have exactly the same parameters.

- False
- True

더 보기

맞습니다
Yes it is true, parameters are shared among these two networks.

6. Our intuition about the layers of a neural network tells us that units that respond more to complex features are more likely to be in deeper layers. True/False? 1 / 1점

- True
- False

더 보기

맞습니다
Correct. Neurons that understand more complex shapes are more likely to be in deeper layers of a neural network.

7. Neural style transfer uses images Content C, Style S. The loss function used to generate image G is composed of which of the following: (Choose all that apply.) 0 / 1점

- [x] $J_{content}$ that compares C and G.

  ✓ Correct
  Correct, in neural style transfer we are interested in the similarity between S and G, and the similarity between G and C.

- [ ] J_{style} that compares S and G.
- [ ] T that calculates the triplet loss between S, G, and C.
- [ ] J_{corr} that compares

더 보기

⊗ 틀립니다
You didn't select all the correct answers

8. In neural style transfer the content loss J_{cont} is computed as: 1 / 1점

$$J_{cont}(G, C) = \left\| a^{[l](C)} - a^{[l](G)} \right\|^2$$

Where $a^{[l](k)}$ is the activation of the l-th layer of a ConvNet trained for classification. We choose l to be a very high value to use compared to the more abstract activation of each image. True/False?

- ○ True
- ◉ False

더 보기

✓ 맞습니다
Correct. We don't use a very deep layer since this will only compare if the two images belong to the same category.

9. In neural style transfer, we can't use gradient descent since there are no trainable parameters. True/False? 1 / 1점

- (●) False
- ( ) True

더 보기

맞습니다
Correct. We use gradient descent on the cost function $J(G)$ and we update the pixel values of the generated image G.

10. You are working with 3D data. You are building a network layer whose input volume has size 32x32x32x16 (this volume has 16 channels), and applies convolutions with 32 filters of dimension 3x3x3x16 (no padding, stride 1). What is the resulting output volume? 1 / 1점

- (●) 30x30x30x32
- ( ) 30x30x30x16
- ( ) Undefined: This convolution step is impossible and cannot be performed because the dimensions specified don't match up.

더 보기

맞습니다
Correct, you have used the formula $\lfloor \frac{n^{[l-1]}-f+2\times p}{s} \rfloor + 1 = n^{[l]}$ over the three first dimensions of the input data.