## Congratulations! You passed!

0.8

Grade received 100% To pass 80% or higher

Go to next item

1.	Classification allows you to identify things.	s while siamese networks allow you to categorize	u to categorize 1 / 1 point					
	O True							
	<ul><li>False</li></ul>							
	<b>⊘</b> Correct							
	Correct.							
2.	Do the two subnetworks in a siam	ese netwo	rk share th	ne same pa	arameters?	1/1 point		
	Yes							
	○ No							
	<b>⊘</b> Correct							
	Correct.							
3.	When training a siamese network you expect to have the highest cos			s, which pa	airs of questions from the following questions do	1/1 point		
	you expect to have the highest co.	arre sirrita	, .					
	Is learning NLP useful for me to ge							
	What should I learn to get a job? (i	POSITIVE)						
	Where is the job? (NEGATIVE)							
	Anchor, Positive							
	Anchor, Negative							
	O Negative, Positive							
	⊘ Correct							
	Correct.							
4.	In the triplet loss function below, optimization during training?	will decrea	sing the h	yperparan	neter alpha from 0.5 to 0.2 require more, or less,	1/1 point		
	diff = $s(A, N) - s(A, P)$							
	$\mathcal{L}(A,P,N) = \max(diff + lpha,0)$							
	<ul><li>Less</li></ul>							
	O More.							
	<ul> <li>Correct</li> <li>Correct. Alpha is the margin</li> </ul>	, so the sm	aller it is t	he less vo	u have to optimize.			
	, ,			,	·			
5.	The orange square below correspondence	onds to the	similarity	score of c	juestion duplicates?	1/1 point		
		0.7	-0.6	-0.4				
		-0.6	0.4	0.1				
		-0.4	0.1	0.5				
	<ul><li>False</li></ul>							
	O True							
	Correct. They correspond to	non quest	tion duplic	cates.				
6.	What is the closest negative in this set of numbers assuming a duplicate pair similarity of 0.6? 1/1 point							
	[-0.9,-0.4,0.4, 0.8]							
	-0.9							
	○ -0.4 <b>③</b> 0.4							
	U							

signature.  No Yes  Correct Correct.  B. During training, you have to update the weights of each of the subnetworks independently.  False. True.  Correct Correct. You update the same weight.  The mean negative is defined as the closest off-diagonal value to the diagonal in each row (excluding the diagonal).  False True  Correct Correct Correct Correct Correct	○ Correct Correct.	
signature.  ● No  Yes  Correct Correct.  During training, you have to update the weights of each of the subnetworks independently.  False.  True.  Correct Correct.  True.  False  True  True  In the mean negative is defined as the closest off-diagonal value to the diagonal in each row (excluding the diagonal).  False  True  Correct Correct.  True  Correct Correct.  True  Correct Correct.  In what order are Siamese networks performed in lecture?  In the mean negative is defined as the closest off-diagonal value to the diagonal in each row (excluding the diagonal).  False  True  Correct Correct.  Correct Correct.  Correct Correct.  In the mean negative is defined as the closest off-diagonal value to the diagonal in each row (excluding the diagonal).  False  True  Correct Correct Correct.  Correct Correct.  Correct Correct.  1/31  Correct each input into an array of numbers  Feed arrays into your model  Run soft-max classifier for all classes  Take the arg-max of the probabilities  Convert each input into an array of numbers  Feed arrays into your model  Run soft-max classifier for all classes  Take the arg-max of the probabilities  Correct each input into an array of numbers  Feed arrays into your model  Correct each input into an array of numbers  Feed arrays into your model  Correct each input into an array of numbers  Feed arrays into your model  Correct each input into an array of numbers  Feed arrays into your model  Correct each input into an array of numbers  Feed arrays into your model  Correct each input into an array of numbers  Feed arrays into your model  Correct each input into an array of numbers  Feed arrays into your model  Correct each input into an array of numbers  Feed arrays into your model  Correct each input into an array of numbers  Feed arrays into your model  Correct each input into an array of numbers  Feed arrays into your model  Correct each input into an array of numbers  Feed arrays into your model  Correct each input into an array of numbers  Feed arrays into your model  Correct ea		
O Yes  O Correct Correct.  During training, you have to update the weights of each of the subnetworks independently.  False.  True.  O correct Correct. You update the same weight.  1/1  9. The mean negative is defined as the closest off-diagonal value to the diagonal in each row (excluding the diagonal).  False True  C correct Correct.  Correct Correct.  1/1  10. In what order are Siamese networks performed in lecture?  1/1  10. In what order are Siamese networks performed in lecture?  1/1  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare v1, v2 using cosine similarity 4. Test against a threshold  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run logistic regression classifier 4. Classify by using the probability  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run soft-max classifier for all classes 4. Take the arg-max of the probabilities  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run soft-max classifier for all classes 4. Take the arg-max of the probabilities  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare v1, v2 using euclidean distance		1/1p
8. During training, you have to update the weights of each of the subnetworks independently.		
<ul> <li>False.</li> <li>True.</li> <li>✓ correct Correct. You update the same weight.</li> <li>1/11</li> <li>a. The mean negative is defined as the closest off-diagonal value to the diagonal in each row (excluding the diagonal).</li> <li>False</li> <li>True</li> <li>✓ correct Correct.</li> <li>1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare v1, v2 using cosine similarity 4. Test against a threshold</li> <li>1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run logistic regression classifier 4. Classify by using the probability</li> <li>1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run soft-max classifier for all classes 4. Take the arg-max of the probabilities</li> <li>1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run soft-max classifier for all classes 4. Take the arg-max of the probabilities</li> <li>1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare v1, v2 using euclidean distance</li> </ul>	⊙ Correct	
<ul> <li>False.</li> <li>True.</li> <li>Correct         Correct. You update the same weight.</li> </ul> <li>1/11         Same an negative is defined as the closest off-diagonal value to the diagonal in each row (excluding the diagonal).</li> <li>False</li> <ul> <li>True</li> <li>Correct</li> <li>Correct</li> <li>Correct.</li> </ul> <li>10. In what order are Siamese networks performed in lecture?</li> <li>1/11         <ul> <li>Correct</li> <li>Convert each input into an array of numbers</li> <li>Feed arrays into your model</li> <li>Compare v1, v2 using cosine similarity</li> <li>Convert each input into an array of numbers</li> <li>Feed arrays into your model</li> <li>Run logistic regression classifier</li> <li>Classify by using the probability</li> <li>Convert each input into an array of numbers</li> <li>Feed arrays into your model</li> <li>Run soft-max classifier for all classes</li> <li>Take the arg-max of the probabilities</li> <li>Convert each input into an array of numbers</li> <li>Feed arrays into your model</li> <li>Compare v1, v2 using euclidean distance</li> </ul> </li>		
True.  ○ Correct Correct. You update the same weight.  9. The mean negative is defined as the closest off-diagonal value to the diagonal in each row (excluding the diagonal).  ● False ○ True  ○ Correct Correct.  10. In what order are Siamese networks performed in lecture?  1 Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare v1, v2 using cosine similarity 4. Test against a threshold  ○ 1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run logistic regression classifier 4. Classify by using the probability  ○ 1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run soft-max classifier for all classes 4. Take the arg-max of the probabilities  ○ 1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run soft-max classifier for all classes 4. Take the arg-max of the probabilities  ○ 1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare v1, v2 using euclidean distance	. During training, you have to update the weights of each of the subnetworks independently.	1/1p
Ocrrect Correct. You update the same weight.  1/11  3. The mean negative is defined as the closest off-diagonal value to the diagonal in each row (excluding the diagonal).  False True  Correct Correct.  1/11  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare v1, v2 using cosine similarity 4. Test against a threshold  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run logistic regression classifier 4. Classify by using the probability  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run logistic regression classifier 4. Classify by using the probability  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run soft-max classifier for all classes 4. Take the arg-max of the probabilities  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare v1, v2 using euclidean distance		
9. The mean negative is defined as the closest off-diagonal value to the diagonal in each row (excluding the diagonal).   ● False  ○ True  ○ Correct  Correct.  1/11  ● 1. Convert each input into an array of numbers  2. Feed arrays into your model  3. Compare v1, v2 using cosine similarity  4. Test against a threshold  ○ 1. Convert each input into an array of numbers  2. Feed arrays into your model  3. Run logistic regression classifier  4. Classify by using the probability  ○ 1. Convert each input into an array of numbers  2. Feed arrays into your model  3. Run logistic regression classifier  4. Classify by using the probability  ○ 1. Convert each input into an array of numbers  2. Feed arrays into your model  3. Run soft-max classifier for all classes  4. Take the arg-max of the probabilities  ○ 1. Convert each input into an array of numbers  2. Feed arrays into your model  3. Convert each input into an array of numbers  2. Feed arrays into your model  3. Convert each input into an array of numbers  2. Feed arrays into your model  3. Convert each input into an array of numbers  2. Feed arrays into your model  3. Convert each input into an array of numbers  2. Feed arrays into your model  3. Compare v1, v2 using euclidean distance		
diagonal).  False  True  Correct  Correct.  1/11  1. Convert each input into an array of numbers  2. Feed arrays into your model  3. Compare v1, v2 using cosine similarity  4. Test against a threshold  1. Convert each input into an array of numbers  2. Feed arrays into your model  3. Run logistic regression classifier  4. Classify by using the probability  1. Convert each input into an array of numbers  2. Feed arrays into your model  3. Run soft-max classifier for all classes  4. Take the arg-max of the probabilities  1. Convert each input into an array of numbers  2. Feed arrays into your model  3. Run soft-max classifier for all classes  4. Take the arg-max of the probabilities  1. Convert each input into an array of numbers  2. Feed arrays into your model  3. Compare v1, v2 using euclidean distance		
diagonal).  False  True  Correct  Correct.  1/11  1. Convert each input into an array of numbers  2. Feed arrays into your model  3. Compare v1, v2 using cosine similarity  4. Test against a threshold  1. Convert each input into an array of numbers  2. Feed arrays into your model  3. Run logistic regression classifier  4. Classify by using the probability  1. Convert each input into an array of numbers  2. Feed arrays into your model  3. Run soft-max classifier for all classes  4. Take the arg-max of the probabilities  1. Convert each input into an array of numbers  2. Feed arrays into your model  3. Run soft-max classifier for all classes  4. Take the arg-max of the probabilities  1. Convert each input into an array of numbers  2. Feed arrays into your model  3. Compare v1, v2 using euclidean distance		
True  Correct Correct.  10. In what order are Siamese networks performed in lecture?  1/11  ■ 1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare v1, v2 using cosine similarity 4. Test against a threshold  □ 1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run logistic regression classifier 4. Classify by using the probability  □ 1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run soft-max classifier for all classes 4. Take the arg-max of the probabilities  □ 1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run soft-max classifier for all classes 4. Take the arg-max of the probabilities  □ 1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare v1, v2 using euclidean distance		1/1p
Correct Correct.  10. In what order are Siamese networks performed in lecture?  1/1  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare v1, v2 using cosine similarity 4. Test against a threshold  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run logistic regression classifier 4. Classify by using the probability  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run soft-max classifier for all classes 4. Take the arg-max of the probabilities  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Convert each input into an array of numbers 2. Feed arrays into your model 3. Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare v1, v2 using euclidean distance		
10. In what order are Siamese networks performed in lecture?  1/11  1. Convert each input into an array of numbers  2. Feed arrays into your model  3. Compare v1, v2 using cosine similarity  4. Test against a threshold  1. Convert each input into an array of numbers  2. Feed arrays into your model  3. Run logistic regression classifier  4. Classify by using the probability  1. Convert each input into an array of numbers  2. Feed arrays into your model  3. Run soft-max classifier for all classes  4. Take the arg-max of the probabilities  1. Convert each input into an array of numbers  2. Feed arrays into your model  3. Compare v1, v2 using euclidean distance		
1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare v1, v2 using cosine similarity 4. Test against a threshold  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run logistic regression classifier 4. Classify by using the probability  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run soft-max classifier for all classes 4. Take the arg-max of the probabilities  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Convert each input into an array of numbers 2. Feed arrays into your model 3. Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare v1, v2 using euclidean distance		
1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare v1, v2 using cosine similarity 4. Test against a threshold  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run logistic regression classifier 4. Classify by using the probability  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run soft-max classifier for all classes 4. Take the arg-max of the probabilities  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Convert each input into an array of numbers 2. Feed arrays into your model 3. Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare v1, v2 using euclidean distance		
2. Feed arrays into your model 3. Compare v1, v2 using cosine similarity 4. Test against a threshold  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run logistic regression classifier 4. Classify by using the probability  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run soft-max classifier for all classes 4. Take the arg-max of the probabilities  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare v1, v2 using euclidean distance	D. In what order are Siamese networks performed in lecture?	1/1p
<ol> <li>Compare v1, v2 using cosine similarity</li> <li>Test against a threshold</li> <li>Convert each input into an array of numbers</li> <li>Feed arrays into your model</li> <li>Run logistic regression classifier</li> <li>Classify by using the probability</li> <li>Convert each input into an array of numbers</li> <li>Feed arrays into your model</li> <li>Run soft-max classifier for all classes</li> <li>Take the arg-max of the probabilities</li> <li>Convert each input into an array of numbers</li> <li>Feed arrays into your model</li> <li>Convert each input into an array of numbers</li> <li>Feed arrays into your model</li> <li>Compare v1, v2 using euclidean distance</li> </ol>		
4. Test against a threshold  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run logistic regression classifier 4. Classify by using the probability  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run soft-max classifier for all classes 4. Take the arg-max of the probabilities  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare v1, v2 using euclidean distance		
1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run logistic regression classifier 4. Classify by using the probability  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run soft-max classifier for all classes 4. Take the arg-max of the probabilities  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare v1, v2 using euclidean distance		
2. Feed arrays into your model 3. Run logistic regression classifier 4. Classify by using the probability  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run soft-max classifier for all classes 4. Take the arg-max of the probabilities  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare x1, x2 using euclidean distance		
4. Classify by using the probability  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run soft-max classifier for all classes 4. Take the arg-max of the probabilities  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare v1, v2 using euclidean distance		
1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Run soft-max classifier for all classes 4. Take the arg-max of the probabilities  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare v1, v2 using euclidean distance	3. Run logistic regression classifier	
2. Feed arrays into your model 3. Run soft-max classifier for all classes 4. Take the arg-max of the probabilities  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare v1, v2 using euclidean distance	4. Classify by using the probability	
3. Run soft-max classifier for all classes 4. Take the arg-max of the probabilities  1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare v1, v2 using euclidean distance	Convert each input into an array of numbers	
<ol> <li>Take the arg-max of the probabilities</li> <li>Convert each input into an array of numbers</li> <li>Feed arrays into your model</li> <li>Compare ν1, ν2 using euclidean distance</li> </ol>	2. Feed arrays into your model	
1. Convert each input into an array of numbers 2. Feed arrays into your model 3. Compare v1, v2 using euclidean distance		
<ol> <li>Feed arrays into your model</li> <li>Compare ν1, ν2 using euclidean distance</li> </ol>	4. Take the arg-max of the probabilities	
3. Compare $\nu 1, \nu 2$ using euclidean distance	Convert each input into an array of numbers	
	2. Feed arrays into your model	
4. Test against a threshold		
	4. Test against a threshold	