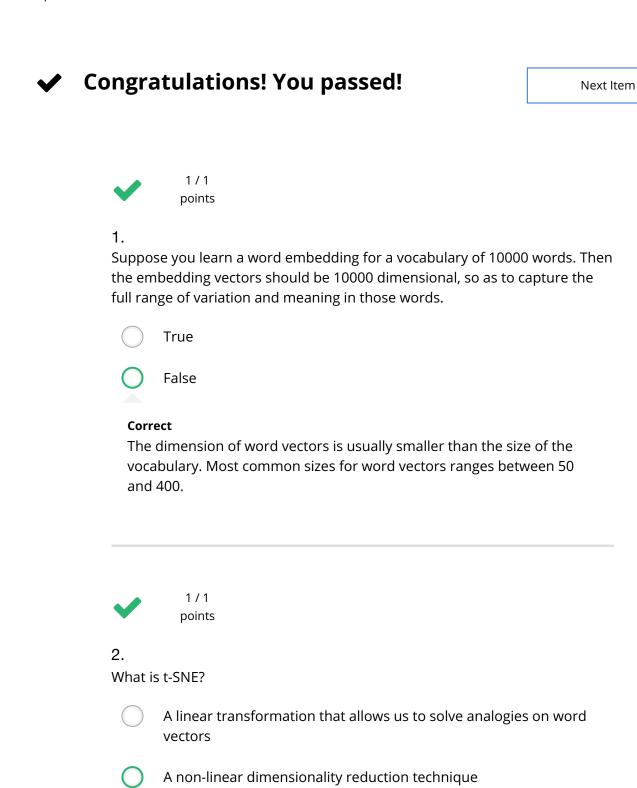
Natural Language Processing & Word Embeddings

10/10 points (100%)

Quiz, 10 questions



https://www.coursera.org/learn/nlp-sequence-models/exam/nIIUO/natural-language-processing-word-embeddings and the sequence-models of th

Correct Yes

A supervised learning alg	orithm for l	learning word	embeddings
7 Caper Visea rearring and	,011	icai i iii g ii a	c

Natural Language Processing & Word Empheddings

10/10 points (100%)

Quiz, 10 questions



1/1 points

3.

Suppose you download a pre-trained word embedding which has been trained on a huge corpus of text. You then use this word embedding to train an RNN for a language task of recognizing if someone is happy from a short snippet of text, using a small training set.

x (input text)	y (happy?)
I'm feeling wonderful today!	1
I'm bummed my cat is ill.	0
Really enjoying this!	1

Then even if the word "ecstatic" does not appear in your small training set, your RNN might reasonably be expected to recognize "I'm ecstatic" as deserving a label y=1.



True

Correct

Yes, word vectors empower your model with an incredible ability to generalize. The vector for "ecstatic would contain a positive/happy connotation which will probably make your model classified the sentence as a "1".

False



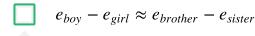
1/1 points

4.

Which of these equations do you think should hold for a good word Natural Language Processing to Word Embeddings

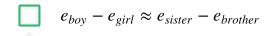
10/10 points (100%)

Quiz, 10 questions



Correct

Yes!



Un-selected is correct



Correct

Yes!



Un-selected is correct



1/1 points

5.

Let E be an embedding matrix, and let o_{1234} be a one-hot vector corresponding to word 1234. Then to get the embedding of word 1234, why don't we call $E*o_{1234}$ in Python?



It is computationally wasteful.

Correct

Yes, the element-wise multiplication will be extremely inefficient.

- The correct formula is $E^T * o_{1234}$.
- This doesn't handle unknown words (<UNK>).

4/29/2018	Coursera Online Courses From Top Universities. Join for Free Coursera				
	None of the above: calling the Python snippet as described above				
Natural Lan	guage ^{isfin} ocessing & Word Embeddings	10/10 points (100%)			
Quiz, 10 questions					
	1 / 1 points				
	6.				
	When learning word embeddings, we create an artificial task of estimating $P(target \mid context)$. It is okay if we do poorly on this artificial prediction				
	task; the more important by-product of this task is that we learn a useful se of word embeddings.	et			
	True				
	True				
	Correct				
	False				
	1/1 points				
	7.				
	In the word2vec algorithm, you estimate $P(t\mid c)$, where t is the target word and c is a context word. How are t and c chosen from the training set? Pick the best answer.				
	c is a sequence of several words immediately before t .				
	c and t are chosen to be nearby words.				
	Correct				
	$igcap c$ is the sequence of all the words in the sentence before $\it t$.				

 \boldsymbol{c} is the one word that comes immediately before \boldsymbol{t} .



1/1 points 8.

Suppose you have a 10000 word vocabulary, and are learning 500-Natural Language Parocessing Mard Fanted ings the following

10/10 points (100%)

Quiz, 10 questions

softmax function:

$$P(t \mid c) = \frac{e^{\theta_t^T e_c}}{\sum_{t'=1}^{10000} e^{\theta_t^T e_c}}$$

Which of these statements are correct? Check all that apply.

 $heta_t$ and e_c are both 500 dimensional vectors.

Correct

 θ_t and e_c are both 10000 dimensional vectors.

Un-selected is correct

 θ_t and e_c are both trained with an optimization algorithm such as Adam or gradient descent.

Correct

After training, we should expect θ_t to be very close to e_c when t and c are the same word.

Un-selected is correct



1/1 points

9.

Suppose you have a 10000 word vocabulary, and are learning 500-dimensional word embeddings. The GloVe model minimizes this objective:

$$\min \sum_{i=1}^{10,000} \sum_{j=1}^{10,000} f(X_{ij}) (\theta_i^T e_j + b_i + b_j' - log X_{ij})^2$$

Which of these statements are correct? Check all that apply.



 θ_i and e_i should be initialized to 0 at the beginning of training.

Natural Language Processing & Word Embeddings

10/10 points (100%)

Quiz, 10 questions

 θ_i and e_j should be initialized randomly at the beginning of training.

Correct

 X_{ij} is the number of times word i appears in the context of word j.

Correct

The weighting function f(.) must satisfy f(0) = 0.

Correct

The weighting function helps prevent learning only from extremely common word pairs. It is not necessary that it satisfies this function.



1/1 points

10.

You have trained word embeddings using a text dataset of m_1 words. You are considering using these word embeddings for a language task, for which you have a separate labeled dataset of m_2 words. Keeping in mind that using word embeddings is a form of transfer learning, under which of these circumstance would you expect the word embeddings to be helpful?



 $m_1 >> m_2$

Correct

 $m_1 \ll m_2$

Natural Language Processing & Word Embeddings

10/10 points (100%)

Quiz, 10 questions

↑ ♥ □