9/10 points (90.00%)

Quiz, 10 questions

<b>~</b>	Congratulations! You passed!	Next Item
<b>/</b>	1/1 points	
	se you learn a word embedding for a vocabulary of 10000 words. Then the embe sional, so as to capture the full range of variation and meaning in those words.	edding vectors should be 1000
	True	
0	False	
	ect dimension of word vectors is usually smaller than the size of the vocabulary. Mo d vectors ranges between 50 and 400.	st common sizes for
2. What is	1 / 1 points s t-SNE?	
	A linear transformation that allows us to solve analogies on word vectors	
0	A non-linear dimensionality reduction technique	
<b>Corr</b> Yes	ect	
	A supervised learning algorithm for learning word embeddings	
	An open-source sequence modeling library	



9/10 points (90.00%)

Ouiz 10 questions 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
es, 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 embedding? (Check all that apply)

$$igcup_{boy} - e_{girl} pprox e_{brother} - e_{sister}$$

### Correct

Yes!

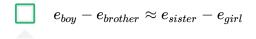
$$igcep_{boy} - e_{girl} pprox e_{sister} - e_{brother}$$

**Un-selected** is correct

$pprox e_{qirl} - e_{si}$	$e_{boy}-e_{brother}$	ı
$pprox e_{girl} - e_s$	$e_{boy}-e_{brother}$	Į

9/10 points (90.00%)

Quiz, 10 questions



**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?

O It is computationally wasteful.

#### Correct

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

The correct formula is $E^Tst o_{1234}.$
This doesn't handle unknown words ( <unk>).</unk>
None of the above: calling the Python snippet as described above is fine.



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 set of word embeddings.

0

True

Correct

False

9/10 points (90.00%)

Quiz, 10 questions

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 is the sequence of all the words in the sentence before t.
- $igcup_c$  and t are chosen to be nearby words.

Correct

( )	c is the one word that comes immediately before $t$
	o is the one from a mar commodition, some c



1/1 points

8.

Suppose you have a 10000 word vocabulary, and are learning 500-dimensional word embeddings. The word2vec model uses the following 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.

 $igcup_t$  and  $e_c$  are both 500 dimensional vectors.

Correct

 $heta_t$  and  $e_c$  are both 10000 dimensional vectors.

**Un-selected** is correct

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

Correct

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

# Natural Language Processing & Word Embeddings

9/10 points (90.00%)

Quiz, 10 questions



0/1 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.

 $heta_i$  and  $e_j$  should be initialized to 0 at the beginning of training.

### This should not be selected

The variables should not be initialized to 0 at the beginning of training.

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

### This should be selected

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

### Correct

 $oxed{igcap}$  The weighting function f(.) must satisfy f(0)=0.

### Correct

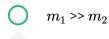
The weighting function helps prevent learning only from extremely common word pairs. It is not necessary that it satisfies this function.



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  $n_1 = n_2 = n_3 = n_3$ 



Correct

