9/10 points (90%)

Quiz, 10 questions

✓ Congra	atulations! You passed!	Next I tem
~	1 / 1 points	
Then t	se you learn a word embedding for a vocabulary of 10000 he embedding vectors should be 10000 dimensional, so a I range of variation and meaning in those words.	
	True	
0	False	
Corr	ect	
✓	1 / 1 points	
2. What i	s t-SNE?	
	A linear transformation that allows us to solve analogie vectors	s on word
0	A non-linear dimensionality reduction technique	
Corr	ect	
	A supervised learning algorithm for learning word embe	eddings
	An open-source sequence modeling library	

9/10 points (90%)

Quiz, 10 questions

/

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.



Correct

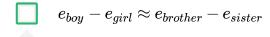
False



1/1 points

4.

Which of these equations do you think should hold for a good word embedding? (Check all that apply)



Correct

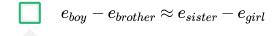
	e_{boy} $-$	e_{airl}	\approx	e_{sister}	_	$e_{brother}$
	oog	9010		000001		UI UUIIUI

9/10 points (90%)

Quiz, 10 questions



Correct



Un-selected is correct

Un-selected is correct



1/1 points

5.

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

0

It is computationally wasteful.

Correct

- $igcap ext{The correct formula is } E^T*e_{1234}.$
- This doesn't handle unknown words (<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 $P(taraet \mid context)$. It is okay if we do poorly on this artificial prediction 0 points (90%)

	1 (tar get content): it is only if we do poorly off this artificial prediction 1848 1840
Quiz, 10 questions	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.
	$igcup_c$ and t are chosen to be nearby words.
	Correct
	igcap c is the one word that comes immediately before t .
	c is a sequence of several words immediately before t .
	c is the sequence of all the words in the sentence before t .
	0/1 points

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

Natural Languager Processing & Word Embeddings

9/10 points (90%)

Quiz, 10 questions

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

 θ_t and e_c are both 10000 dimensional vectors.

This should not be selected

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

This should be selected

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

This should not be selected



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.

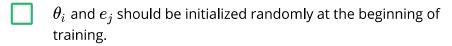
 $igcap heta_i$ and e_j should be initialized to 0 at the beginning of training.

Un-selected is correct

Natural Language Processing & Word Embeddings

9/10 points (90%)

Quiz, 10 questions





 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

 $m_1 \ll m_2$

♥ □ 9/10 points (90%)

Quiz, 10 questions