False

True

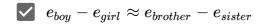
b/U	Natural Language Processing & Word Embeddings Coursera	
1.	Suppose you learn a word embedding for a vocabulary of 10000 words. Then the embedding vectors should be 10000 dimensional, so as to capture the full range of variation and meaning in those words.	1 / 1 poin
	○ True	
	False	
	✓ Correct	
	The dimension of word vectors is usually smaller than the size of the vocabulary. Most common sizes for word vectors range between 50 and 400.	
2.	What is t-SNE?	1 / 1 poin
	An open-source sequence modeling library	
	A linear transformation that allows us to solve analogies on word vectors	
	A supervised learning algorithm for learning word embeddings	
	A non-linear dimensionality reduction technique	
	✓ Correct	
	Yes	
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	2 1/1 poin
	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!	_
	I'm bummed my cat is ill. 0	
	Really enjoying this!	
	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

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 classify the sentence as a "1".

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

1 / 1 point



Correct

Yes!

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

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

Correct

Yes!

$$oxed{e} e_{boy} - e_{girl} pprox e_{sister} - e_{brother}$$

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?

1 / 1 point

- None of the above: calling the Python snippet as described above is fine.
- It is computationally wasteful.
- The correct formula is $E^T * o_{1234}$.
- This doesn't handle unknown words (<UNK>).

Correct

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

- 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.
 - False
 - True
 - ✓ Correct
- 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.

1 / 1 point

- $\bigcirc c$ is a sequence of several words immediately before t.
- $\bigcirc \ c$ is the one word that comes immediately before t.
- $\bigcirc \ c$ is the sequence of all the words in the sentence before t.
- lacktriangledown c and t are chosen to be nearby words.
 - ✓ Correct
- 8. Suppose you have a 10000 word vocabulary, and are learning 500-dimensional word embeddings. The word2vec model uses the following softmax function:

1 / 1 point

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

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

- lacksquare θ_t and e_c are both 500 dimensional vectors.
 - Correct
- lacksquare and e_c are both trained with an optimization algorithm such as Adam or gradient descent.

16/06/2021

Correct

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

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

1 / 1 point

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

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

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

✓ Correct

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

✓ Correct

lacksquare 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.

- 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 circumstances would you expect the word embeddings to be helpful?

1 / 1 point

- $m_1 >> m_2$
- $\bigcap m_1 \ll m_2$

