True/False: Suppose you learn a word embedding for a vocabulary of 60000 words. Then the embedding vectors could be 60000 dimensional, so as to capture the full range of variation and meaning in those words. 1 point

○ False

○ True

2.True/False: t-SNE is a linear transformation that allows us to solve analogies on word vectors. 1 point

○ False

○ True

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. 1 point

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 $y = 1$.

○ False

○ True

4.Which of these equations do you think should hold for a good word embedding? (Check all that apply) 1 point

□ $e_{man} - e_{woman} \approx e_{queen} - e_{king}$

□ $e_{man} - e_{king} \approx e_{woman} - e_{queen}$

□ $e_{man} - e_{woman} \approx e_{king} - e_{queen}$

□ $e_{man} - e_{king} \approx e_{queen} - e_{woman}$

5.True/False: The most computationally efficient formula for Python to get the embedding of word 1021, if C is an embedding matrix, and o_{1021} is a one-hot vector corresponding to word 1021, is $C^T * o_{1021}$. 1 point

○ False

○ True

6.When learning word embeddings, words are automatically generated along with the surrounding words. 1 point

○ True

○ False

7.True/False: In the word2vec algorithm, you estimate *P(t* | *c)*, where *t* is the target word and *c* is a context word. *t* and *c* are chosen from the training set to be nearby words. 1 point

○ True

○ False

8.Suppose you have a 10000 word vocabulary, and are learning 500-dimensional word embeddings. The word2vec model uses the following softmax function: 1 point

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

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

□ θ_t and e_c are both 500 dimensional vectors.

□ θ_t and e_c are both 10000 dimensional vectors.

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

□ After training, we should expect θ_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 point

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

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

□ Theoretically, the weighting function $f(.)$ must satisfy $f(0) = 0$.

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

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

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

10.You have trained word embeddings using a text dataset of t_1 words. You are considering using these word embeddings for a language task, for which you have a separate labeled dataset of t_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 point

○ When t_1 is equal to t_2

○ When t_1 is larger than t_2

○ When t_1 is smaller than t_2