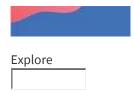
Skip to Main Content



- thadiboyina v m kishore
 - 1. Sequence Models
 - 2. Week 2
 - 3. Natural Language Processing & Word Embeddings

PreviousNext

- Introduction to Word Embeddings
- Learning Word Embeddings: Word2vec & GloVe
- Applications Using Word Embeddings
- Lecture Notes (Optional)
- Quiz

С

Quiz: Natural Language Processing & Word Embeddings

10 questions

Programming Assignments
 QUIZQuiz • 30 MIN30 minutes

Natural Language Processing & Word Embeddings

Submit your assignment

DUE DATEJun 14, 10:59 AM +04June 14, 10:59 AM +04 **ATTEMPTS**3 every 8 hours Try again **Receive grade**

TO PASS80% or higher

Grade

100%

View Feedback

We keep your highest score

Natural Language Processing & Word Embeddings

Graded Quiz • 30 min

Due Jun 14, 10:59 AM +04

Congratulations! You passed!

TO PASS 80% or higher

Keep Learning

GRADE

100%

Natural Language Processing & Word Embeddings

LATEST SUBMISSION GRADE

100%

1.

Question 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.



O

True

 \odot

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.

Question 2

What is t-SNE?



Œ

A non-linear dimensionality reduction technique

 \overline{a}

A linear transformation that allows us to solve analogies on word vectors

c
A supervised learning algorithm for learning word embeddings
C
An open-source sequence modeling library
Correct
Yes
3. Question 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 = 1y=1.

1 / 1 point

 \odot

True

 \circ

False

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.

Question 4

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

1 / 1 point

14 e_{boy} - e_{girl} \approx e_{sister} - e_{brother} eboy-egirl \approx esister-ebrother V e {boy} - e {girl} \approx e {brother} - e {sister} eboy-egirl≈ebrother-esister Correct Yes! V $e_{boy} - e_{boy} - e_{sister} - e_{sister} = e_{sister} - e_{sister}$ Correct Yes! e {boy} - e {brother} \approx e_{sister} - e_{girl} eboy-ebrother \approx esister-egirl 5. Question 5 Let EE be an embedding matrix, and let o $\{1234\}o_{1234}$ be a one-hot vector corresponding to word 1234. Then to get the embedding of word 1234, why don't we call $E * o_{1234}E*o_{1234}$ in Python? 1 / 1 point The correct formula is $E^T^* o_{1234}E^* o_{1234}$. O None of the above: calling the Python snippet as described above is fine. 0 This doesn't handle unknown words (<UNK>). \odot It is computationally wasteful. Correct

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

6.

Question 6

When learning word embeddings, we create an artificial task of estimating $P(\text{target} \mid \text{mid} \text{context})P(\text{target} \mid \text{context})$. 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.

1 / 1 point C
False
\odot
True
Correct
7. Question 7
In the word2vec algorithm, you estimate $P(t \setminus dc)$, where tt is the target word and cc is
a context word. How are tt and cc chosen from the training set? Pick the best answer.
1 / 1 point C
cc is a sequence of several words immediately before tt .
C
cc is the sequence of all the words in the sentence before tt .
C
cc is the one word that comes immediately before tt .
\odot
cc and tt are chosen to be nearby words.
Correct
8. Question 8
Suppose you have a 10000 word vocabulary, and are learning 500-dimensional word
embeddings. The word2vec model uses the following softmax function:
$P(t \mid c) = \frac{e^{\left(t \mid t^T e_c\right)} \left(sum_{t'=1}^{10000}\right)}{e^{\left(t \mid t'\right)^T e_c\right)} \left(sum_{t'=1}^{10000e\theta t'Tece\theta \theta t'Tece}\right)}$
Which of these statements are correct? Check all that apply.
1 / 1 point
\theta_t $ heta_t$ and e_c e_c are both 10000 dimensional vectors.
\theta_t $ heta$ t and e_c e c are both trained with an optimization algorithm such as Adam or gradient

descent.

Correct



\theta $t\theta t$ and e cec are both 500 dimensional vectors.

Correct

After training, we should expect \theta_t θt to be very close to e_cec when tt and cc are the same word.

9.

Question 9

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

 $\label{limin} $$ \min \sum_{i=1}^{10,000} \sum_{j=110,000} f(X_{ij}) (\theta_{ij}) (X_{ij}) (X_{$

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

1 / 1 point

V

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

Correct

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

V

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

Correct

\theta $i\theta_i$ and e_je_j should be initialized to 0 at the beginning of training.

V

\theta $i\theta_i$ and e je_i should be initialized randomly at the beginning of training.

Correct

10.

Question 10

You have trained word embeddings using a text dataset of m_1m_1 words. You are considering using these word embeddings for a language task, for which you have a separate labeled dataset of m_2m_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

0

 \odot

Correct