[[https://d3njjcbhbojbot.cloudfront.net/api/utilities/v1/imageproxy/https:/coursera-course-photos.s3.amazonaws.com/68/185e72894e4ab98b049f98ebbf126c/LongLogo200x50.jpg?auto=format%2Ccompress&dpr=1&w=256&h=32](https://www.coursera.org/)](https://www.coursera.org/)

Explore

Haut du formulaire



0

Bas du formulaire

[Skip to Main Content](https://www.coursera.org/learn/nlp-sequence-models/exam/e4bJR/recurrent-neural-networks/attempt#main)

* [Shopping cart: 1 item](https://www.coursera.org/payments/checkout?cartId=49833185)
* Florence Phuong NGUYEN

[Sequence Models](https://www.coursera.org/learn/nlp-sequence-models/home/welcome)

[Week 1](https://www.coursera.org/learn/nlp-sequence-models/home/week/1)

Recurrent Neural Networks

[Prev](https://www.coursera.org/learn/nlp-sequence-models/lecture/ehs0S/deep-rnns)

[Next](https://www.coursera.org/learn/nlp-sequence-models/notebook/X20PE/building-a-recurrent-neural-network-step-by-step)

* **Recurrent Neural Networks**
* **Practice questions**

[Purchase course to unlock this item.](https://www.coursera.org/learn/nlp-sequence-models/exam/e4bJR/recurrent-neural-networks)

**[Quiz:](https://www.coursera.org/learn/nlp-sequence-models/exam/e4bJR/recurrent-neural-networks)**[Recurrent Neural Networks](https://www.coursera.org/learn/nlp-sequence-models/exam/e4bJR/recurrent-neural-networks)

[10 questions](https://www.coursera.org/learn/nlp-sequence-models/exam/e4bJR/recurrent-neural-networks)

* **Programming assignments**

**QUIZ • 30 MIN**

**Recurrent Neural Networks**

**Submit your assignment**

**DUE**Feb 24, 8:59 AM CET

**ATTEMPTS**3 every 8 hours

Resume

**Receive grade**

**TO PASS**80% or higher

**Grade**

—

Recurrent Neural Networks

Graded Quiz • 30 min

**Due** Feb 24, 8:59 AM CET

**Recurrent Neural Networks**

**TOTAL POINTS 10**

1.Question 1

Suppose your training examples are sentences (sequences of words). Which of the following refers to the j^{th}*jth* word in the i^{th}*ith* training example?



**x^{(i)<j>}*x*(*i*)<*j*>**



**x^{<i>(j)}*x*<*i*>(*j*)**



**x^{(j)<i>}*x*(*j*)<*i*>**

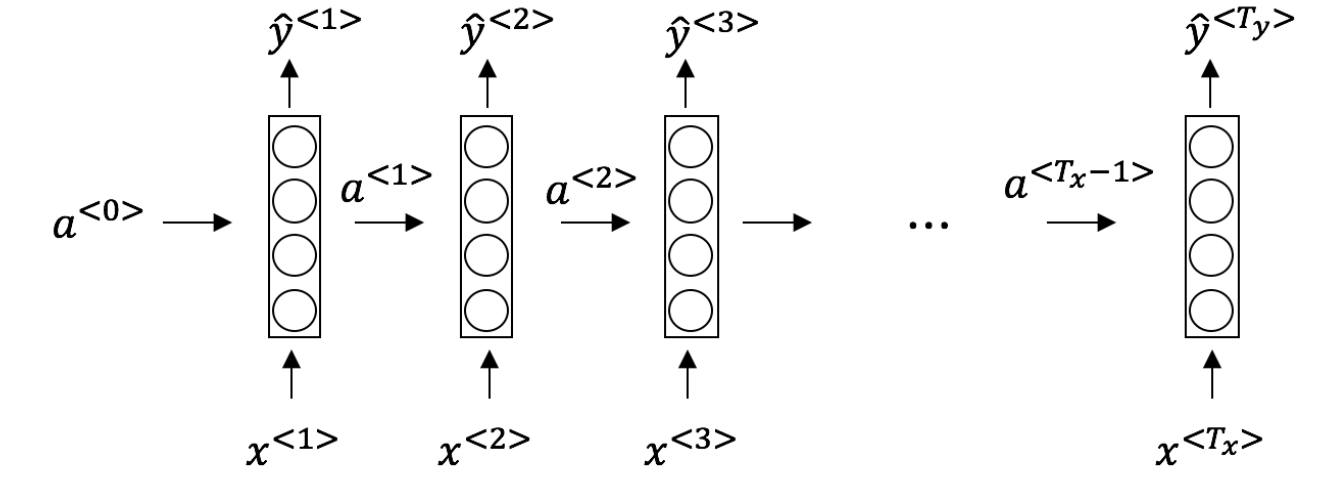


**x^{<j>(i)}*x*<*j*>(*i*)**

1 point

2.Question 2

Consider this RNN:



This specific type of architecture is appropriate when:



**T\_x = T\_y*Tx*​=*Ty*​**



**T\_x < T\_y*Tx*​<*Ty*​**



**T\_x > T\_y*Tx*​>*Ty*​**

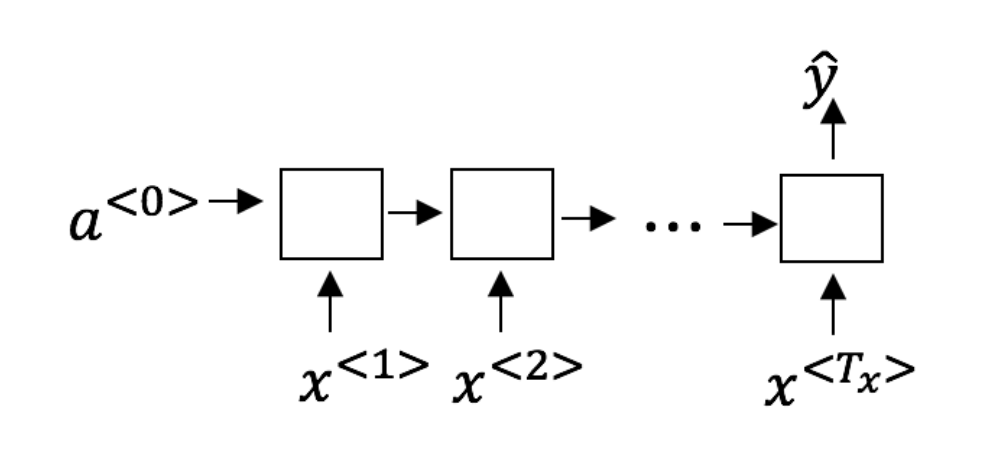


**T\_x = 1*Tx*​=1**

1 point

3.Question 3

To which of these tasks would you apply a many-to-one RNN architecture? (Check all that apply).





Speech recognition (input an audio clip and output a transcript)



Sentiment classification (input a piece of text and output a 0/1 to denote positive or negative sentiment)



Image classification (input an image and output a label)

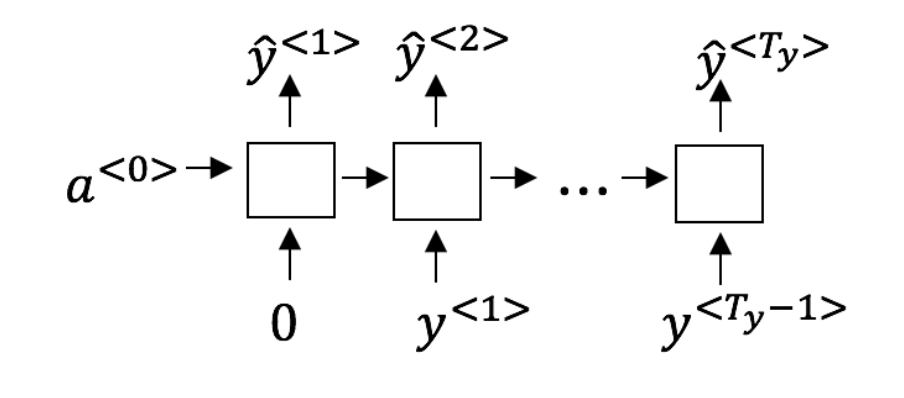


Gender recognition from speech (input an audio clip and output a label indicating the speaker’s gender)

1 point

4.Question 4

You are training this RNN language model.



At the t^{th}*tth* time step, what is the RNN doing? Choose the best answer.



Estimating *P*(*y*<1>,*y*<2>,…,*y*<*t*−1>)



Estimating P(y^{<t>})*P*(*y*<*t*>)



Estimating *P*(*y*<*t*>∣*y*<1>,*y*<2>,…,*y*<*t*−1>)

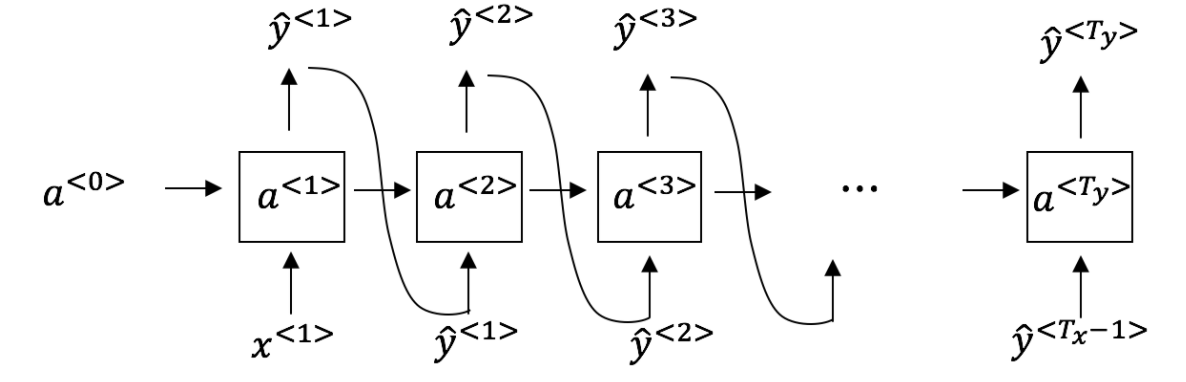


Estimating *P*(*y*<*t*>∣*y*<1>,*y*<2>,…,*y*<*t*>)

1 point

5.Question 5

You have finished training a language model RNN and are using it to sample random sentences, as follows:



What are you doing at each time step t*t*?



(i) Use the probabilities output by the RNN to pick the highest probability word for that time-step as *y*^<*t*>. (ii) Then pass the ground-truth word from the training set to the next time-step.



(i) Use the probabilities output by the RNN to randomly sample a chosen word for that time-step as *y*^<*t*>. (ii) Then pass the ground-truth word from the training set to the next time-step.



(i) Use the probabilities output by the RNN to pick the highest probability word for that time-step as *y*^<*t*>. (ii) Then pass this selected word to the next time-step.



(i) Use the probabilities output by the RNN to randomly sample a chosen word for that time-step as *y*^<*t*>. (ii) Then pass this selected word to the next time-step.

1 point

6.Question 6

You are training an RNN, and find that your weights and activations are all taking on the value of NaN (“Not a Number”). Which of these is the most likely cause of this problem?



Vanishing gradient problem.



Exploding gradient problem.



ReLU activation function g(.) used to compute g(z), where z is too large.



Sigmoid activation function g(.) used to compute g(z), where z is too large.

1 point

7.Question 7

Suppose you are training a LSTM. You have a 10000 word vocabulary, and are using an LSTM with 100-dimensional activations a^{<t>}*a*<*t*>. What is the dimension of Γ*u* at each time step?



1



100



300

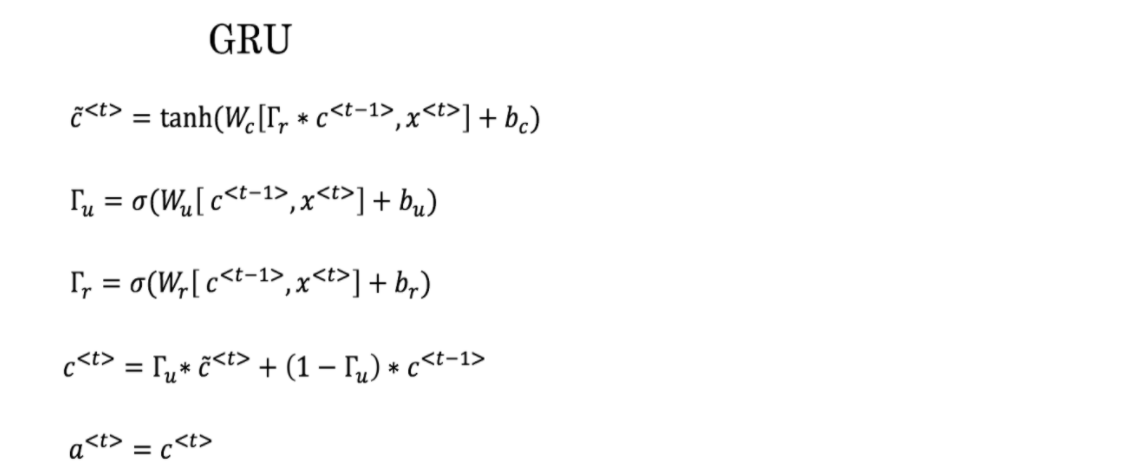


10000

1 point

8.Question 8

Here’re the update equations for the GRU.



Alice proposes to simplify the GRU by always removing the Γ*u*. I.e., setting Γ*u* = 1. Betty proposes to simplify the GRU by removing the Γ*r*. I. e., setting Γ*r* = 1 always. Which of these models is more likely to work without vanishing gradient problems even when trained on very long input sequences?



Alice’s model (removing Γ*u*), because if Γ*r*≈0 for a timestep, the gradient can propagate back through that timestep without much decay.



Alice’s model (removing Γ*u*), because if Γ*r*≈1 for a timestep, the gradient can propagate back through that timestep without much decay.



Betty’s model (removing Γ*r*), because if Γ*u*≈0 for a timestep, the gradient can propagate back through that timestep without much decay.

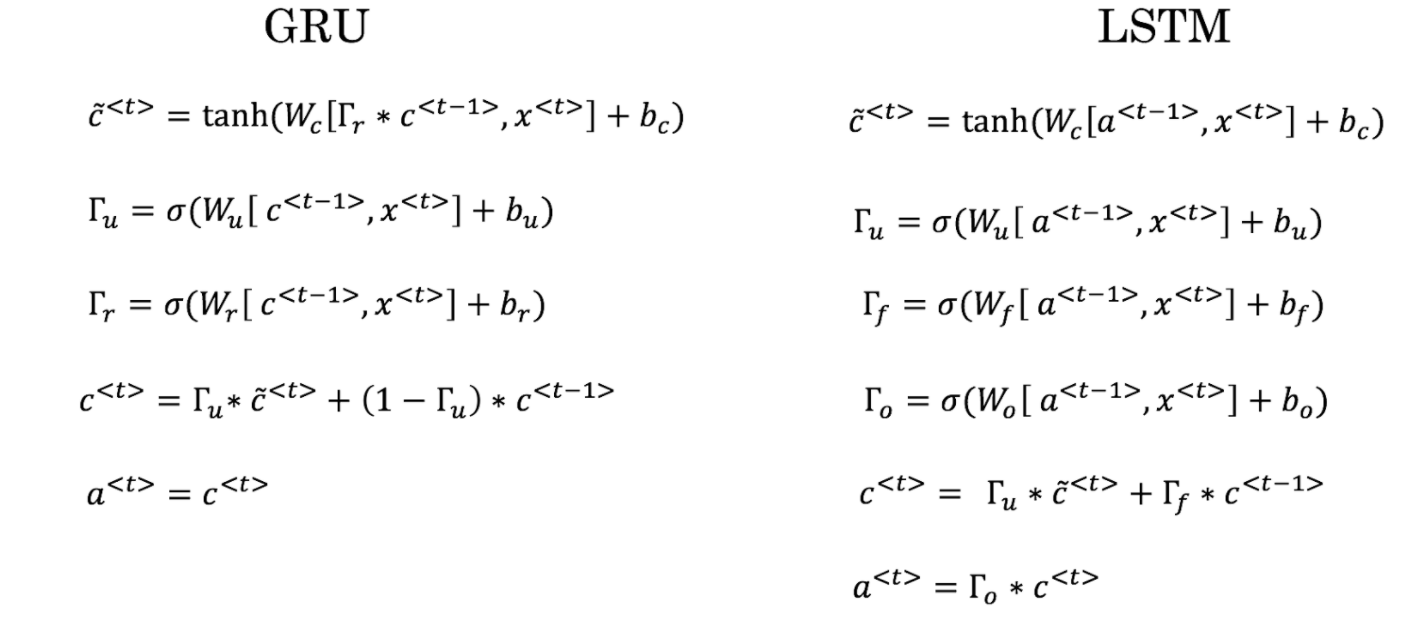


Betty’s model (removing Γ*r*), because if Γ*u*≈1 for a timestep, the gradient can propagate back through that timestep without much decay.

1 point

9.Question 9

Here are the equations for the GRU and the LSTM:



From these, we can see that the Update Gate and Forget Gate in the LSTM play a role similar to \_\_\_\_\_\_\_ and \_\_\_\_\_\_ in the GRU. What should go in the the blanks?



Γ*u* and 1−Γ*u*



Γ*u* and Γ*r*



1−Γ*u* and Γ*u*



Γ*r* and Γ*u*

1 point

10.Question 10

You have a pet dog whose mood is heavily dependent on the current and past few days’ weather. You’ve collected data for the past 365 days on the weather, which you represent as a sequence as *x*<1>,…,*x*<365>. You’ve also collected data on your dog’s mood, which you represent as *y*<1>,…,*y*<365>. You’d like to build a model to map from x \rightarrow y*x*→*y*. Should you use a Unidirectional RNN or Bidirectional RNN for this problem?



Bidirectional RNN, because this allows the prediction of mood on day t to take into account more information.



Bidirectional RNN, because this allows backpropagation to compute more accurate gradients.



Unidirectional RNN, because the value of y^{<t>}*y*<*t*> depends only on *x*<1>,…,*x*<*t*>, but not on *x*<*t*+1>,…,*x*<365>



Unidirectional RNN, because the value of y^{<t>}*y*<*t*> depends only on x^{<t>}*x*<*t*>, and not other days’ weather.

1 point

[[https://d3njjcbhbojbot.cloudfront.net/api/utilities/v1/imageproxy/https:/coursera-course-photos.s3.amazonaws.com/68/185e72894e4ab98b049f98ebbf126c/LongLogo200x50.jpg?auto=format%2Ccompress&dpr=1&w=256&h=32](https://www.coursera.org/)](https://www.coursera.org/)

Explore

Haut du formulaire



0

Bas du formulaire

[Skip to Main Content](https://www.coursera.org/learn/nlp-sequence-models/exam/nIlU0/natural-language-processing-word-embeddings/attempt#main)

* [Shopping cart: 1 item](https://www.coursera.org/payments/checkout?cartId=49833185)
* Florence Phuong NGUYEN

[Sequence Models](https://www.coursera.org/learn/nlp-sequence-models/home/welcome)

[Week 2](https://www.coursera.org/learn/nlp-sequence-models/home/week/2)

Natural Language Processing & Word Embeddings

[Prev](https://www.coursera.org/learn/nlp-sequence-models/lecture/zHASj/debiasing-word-embeddings)

[Next](https://www.coursera.org/learn/nlp-sequence-models/notebook/5NrJ6/operations-on-word-vectors-debiasing)

* **Introduction to Word Embeddings**
* **Learning Word Embeddings: Word2vec & GloVe**
* **Applications using Word Embeddings**
* **Practice questions**

[Purchase course to unlock this item.](https://www.coursera.org/learn/nlp-sequence-models/exam/nIlU0/natural-language-processing-word-embeddings)

**[Quiz:](https://www.coursera.org/learn/nlp-sequence-models/exam/nIlU0/natural-language-processing-word-embeddings)**[Natural Language Processing & Word Embeddings](https://www.coursera.org/learn/nlp-sequence-models/exam/nIlU0/natural-language-processing-word-embeddings)

[10 questions](https://www.coursera.org/learn/nlp-sequence-models/exam/nIlU0/natural-language-processing-word-embeddings)

* **Programming assignments**

**QUIZ • 30 MIN**

**Natural Language Processing & Word Embeddings**

**Submit your assignment**

**DUE**Mar 2, 8:59 AM CET

**ATTEMPTS**3 every 8 hours

Resume

**Receive grade**

**TO PASS**80% or higher

**Grade**

—

Natural Language Processing & Word Embeddings

Graded Quiz • 30 min

**Due** Mar 2, 8:59 AM CET

**Natural Language Processing & Word Embeddings**

**TOTAL POINTS 10**

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.



True



False

1 point

2.Question 2

What is t-SNE?



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



A non-linear dimensionality reduction technique



A supervised learning algorithm for learning word embeddings



An open-source sequence modeling library

1 point

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 = 1*y*=1.



True



False

1 point

4.Question 4

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



e\_{boy} - e\_{girl} \approx e\_{brother} - e\_{sister}*eboy*​−*egirl*​≈*ebrother*​−*esister*​



e\_{boy} - e\_{girl} \approx e\_{sister} - e\_{brother}*eboy*​−*egirl*​≈*esister*​−*ebrother*​



e\_{boy} - e\_{brother} \approx e\_{girl} - e\_{sister}*eboy*​−*ebrother*​≈*egirl*​−*esister*​



e\_{boy} - e\_{brother} \approx e\_{sister} - e\_{girl}*eboy*​−*ebrother*​≈*esister*​−*egirl*​

1 point

5.Question 5

Let E*E* 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?



It is computationally wasteful.



The correct formula is E^T\* o\_{1234}*ET*∗*o*1234​.



This doesn’t handle unknown words (<UNK>).



None of the above: calling the Python snippet as described above is fine.

1 point

6.Question 6

When learning word embeddings, we create an artificial task of estimating *P*(*target*∣*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.



True



False

1 point

7.Question 7

In the word2vec algorithm, you estimate *P*(*t*∣*c*), where t*t* is the target word and c*c* is a context word. How are t*t* and c*c* chosen from the training set? Pick the best answer.



c*c* is the one word that comes immediately before t*t*.



c*c* is a sequence of several words immediately before t*t*.



c*c* and t*t* are chosen to be nearby words.



c*c* is the sequence of all the words in the sentence before t*t*.

1 point

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*∣*c*)=*eθTtec*∑10000*t*′=1*eθTt*′*ec*

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



\theta\_t*θt*​ and e\_c*ec*​ are both 500 dimensional vectors.



\theta\_t*θt*​ and e\_c*ec*​ are both 10000 dimensional vectors.



\theta\_t*θt*​ and e\_c*ec*​ are both trained with an optimization algorithm such as Adam or gradient descent.



After training, we should expect \theta\_t*θt*​ to be very close to e\_c*ec*​ when t*t* and c*c* are the same word.

1 point

9.Question 9

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

min∑10,000*i*=1∑10,000*j*=1*f*(*Xij*)(*θTiej*+*bi*+*b*′*j*−*logXij*)2

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



\theta\_i*θi*​ and e\_j*ej*​ should be initialized to 0 at the beginning of training.



\theta\_i*θi*​ and e\_j*ej*​ should be initialized randomly at the beginning of training.



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



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

1 point

10.Question 10

You have trained word embeddings using a text dataset of m\_1*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*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*1​ >> m\_2*m*2​



m\_1*m*1​ << m\_2*m*2​

1 point

[[https://d3njjcbhbojbot.cloudfront.net/api/utilities/v1/imageproxy/https:/coursera-course-photos.s3.amazonaws.com/68/185e72894e4ab98b049f98ebbf126c/LongLogo200x50.jpg?auto=format%2Ccompress&dpr=1&w=256&h=32](https://www.coursera.org/)](https://www.coursera.org/)

Explore

Haut du formulaire



0

Bas du formulaire

[Skip to Main Content](https://www.coursera.org/learn/nlp-sequence-models/exam/4CCc4/sequence-models-attention-mechanism/attempt#main)

* [Shopping cart: 1 item](https://www.coursera.org/payments/checkout?cartId=49833185)
* Florence Phuong NGUYEN

[Sequence Models](https://www.coursera.org/learn/nlp-sequence-models/home/welcome)

[Week 3](https://www.coursera.org/learn/nlp-sequence-models/home/week/3)

Sequence models & Attention mechanism

[Prev](https://www.coursera.org/learn/nlp-sequence-models/lecture/ATXdW/conclusion-and-thank-you)

[Next](https://www.coursera.org/learn/nlp-sequence-models/notebook/npjGi/neural-machine-translation-with-attention)

* **Various sequence to sequence architectures**
* **Speech recognition - Audio data**
* **Conclusion**
* **Practice questions**

[Purchase course to unlock this item.](https://www.coursera.org/learn/nlp-sequence-models/exam/4CCc4/sequence-models-attention-mechanism)

**[Quiz:](https://www.coursera.org/learn/nlp-sequence-models/exam/4CCc4/sequence-models-attention-mechanism)**[Sequence models & Attention mechanism](https://www.coursera.org/learn/nlp-sequence-models/exam/4CCc4/sequence-models-attention-mechanism)

[10 questions](https://www.coursera.org/learn/nlp-sequence-models/exam/4CCc4/sequence-models-attention-mechanism)

* **Programming assignments**

**QUIZ • 30 MIN**

**Sequence models & Attention mechanism**

**Submit your assignment**

**DUE**Mar 9, 8:59 AM CET

**ATTEMPTS**3 every 8 hours

Resume

**Receive grade**

**TO PASS**80% or higher

**Grade**

—

Sequence models & Attention mechanism

Graded Quiz • 30 min

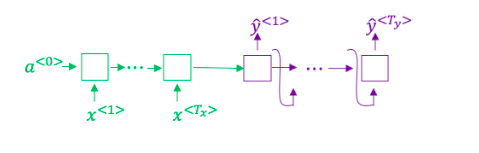
**Due** Mar 9, 8:59 AM CET

**Sequence models & Attention mechanism**

**TOTAL POINTS 10**

1.Question 1

Consider using this encoder-decoder model for machine translation.



This model is a “conditional language model” in the sense that the encoder portion (shown in green) is modeling the probability of the input sentence x*x*.



True



False

1 point

2.Question 2

In beam search, if you increase the beam width B*B*, which of the following would you expect to be true? Check all that apply.



Beam search will run more slowly.



Beam search will use up more memory.



Beam search will generally find better solutions (i.e. do a better job maximizing *P*(*y*∣*x*))



Beam search will converge after fewer steps.

1 point

3.Question 3

In machine translation, if we carry out beam search without using sentence normalization, the algorithm will tend to output overly short translations.



True



False

1 point

4.Question 4

Suppose you are building a speech recognition system, which uses an RNN model to map from audio clip x*x*to a text transcript y*y*. Your algorithm uses beam search to try to find the value of y*y* that maximizes *P*(*y*∣*x*).

On a dev set example, given an input audio clip, your algorithm outputs the transcript *y*^= “I’m building an A Eye system in Silly con Valley.”, whereas a human gives a much superior transcript y^\* =*y*∗= “I’m building an AI system in Silicon Valley.”

According to your model,

*P*(*y*^∣*x*)=1.09∗10−7

*P*(*y*∗∣*x*)=7.21∗10−8

Would you expect increasing the beam width B to help correct this example?



No, because *P*(*y*∗∣*x*)≤*P*(*y*^∣*x*) indicates the error should be attributed to the RNN rather than to the search algorithm.



No, because *P*(*y*∗∣*x*)≤*P*(*y*^∣*x*) indicates the error should be attributed to the search algorithm rather than to the RNN.



Yes, because *P*(*y*∗∣*x*)≤*P*(*y*^∣*x*) indicates the error should be attributed to the RNN rather than to the search algorithm.



Yes, because *P*(*y*∗∣*x*)≤*P*(*y*^∣*x*) indicates the error should be attributed to the search algorithm rather than to the RNN.

1 point

5.Question 5

Continuing the example from Q4, suppose you work on your algorithm for a few more weeks, and now find that for the vast majority of examples on which your algorithm makes a mistake, *P*(*y*∗∣*x*)>*P*(*y*^∣*x*). This suggest you should focus your attention on improving the search algorithm.



True.

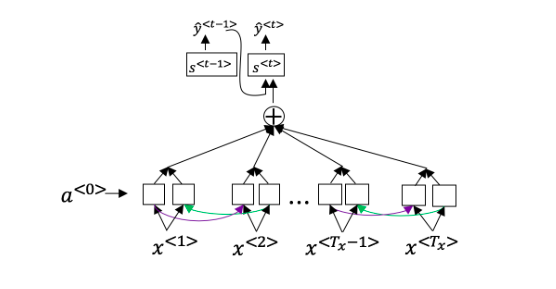


False.

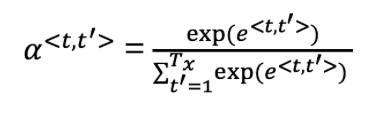
1 point

6.Question 6

Consider the attention model for machine translation.



Further, here is the formula for *α*<*t*,*t*′>.



Which of the following statements about *α*<*t*,*t*′> are true? Check all that apply.



We expect *α*<*t*,*t*′> to be generally larger for values of *a*<*t*′> that are highly relevant to the value the network should output for y^{<t>}*y*<*t*>. (Note the indices in the superscripts.)



We expect *α*<*t*,*t*′> to be generally larger for values of a^{<t>}*a*<*t*> that are highly relevant to the value the network should output for *y*<*t*′>. (Note the indices in the superscripts.)



∑*tα*<*t*,*t*′>=1 (Note the summation is over t*t*.)



∑*t*′*α*<*t*,*t*′>=1 (Note the summation is over *t*′.)

1 point

7.Question 7

The network learns where to “pay attention” by learning the values *e*<*t*,*t*′>, which are computed using a small neural network:

We can't replace s^{<t-1>}*s*<*t*−1> with s^{<t>}*s*<*t*> as an input to this neural network. This is because s^{<t>}*s*<*t*> depends on *α*<*t*,*t*′> which in turn depends on *e*<*t*,*t*′>; so at the time we need to evalute this network, we haven’t computed s^{<t>}*s*<*t*> yet.



True



False

1 point

8.Question 8

Compared to the encoder-decoder model shown in Question 1 of this quiz (which does not use an attention mechanism), we expect the attention model to have the greatest advantage when:



The input sequence length T\_x*Tx*​ is large.



The input sequence length T\_x*Tx*​ is small.

1 point

9.Question 9

Under the CTC model, identical repeated characters not separated by the “blank” character (\_) are collapsed. Under the CTC model, what does the following string collapse to?

\_\_c\_oo\_o\_kk\_\_\_b\_ooooo\_\_oo\_\_kkk



cokbok



cookbook



cook book



coookkboooooookkk

1 point

10.Question 10

In trigger word detection, x^{<t>}*x*<*t*> is:



Features of the audio (such as spectrogram features) at time t*t*.



The t*t*-th input word, represented as either a one-hot vector or a word embedding.



Whether the trigger word is being said at time t*t*.



Whether someone has just finished saying the trigger word at time t*t*.

1 point