축하합니다! 통과하셨습니다!

받은 학점 80% **최신 제출물 학점** 80% **통과 점수:** 80% 이상

7 시간 58 분 후에 과제를 다시 풀어보세요.

다음 항목으로 이동

1. Suppose your training examples are sentences (sequences of words). Which of the following refers to the s^{th} word in the r^{th} training example?

1 / 1점

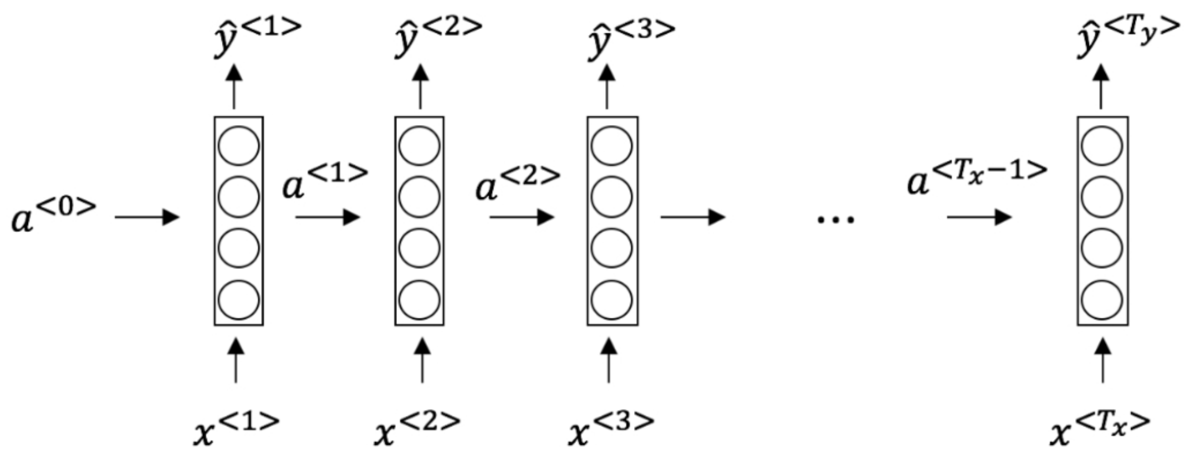
- ○ $x^{<r>(s)}$
- ○ $x^{(s)<r>}$
- ◉ $x^{(r)<s>}$
- ○ $x^{<s>(r)}$

더 보기

맞습니다
We index into the r^{th} row first to get to the r^{th} training example (represented by parentheses), then the s^{th} column to get to the s^{th} word (represented by the brackets).

2. Consider this RNN:

1 / 1점



True/False: This specific type of architecture is appropriate when Tx>Ty

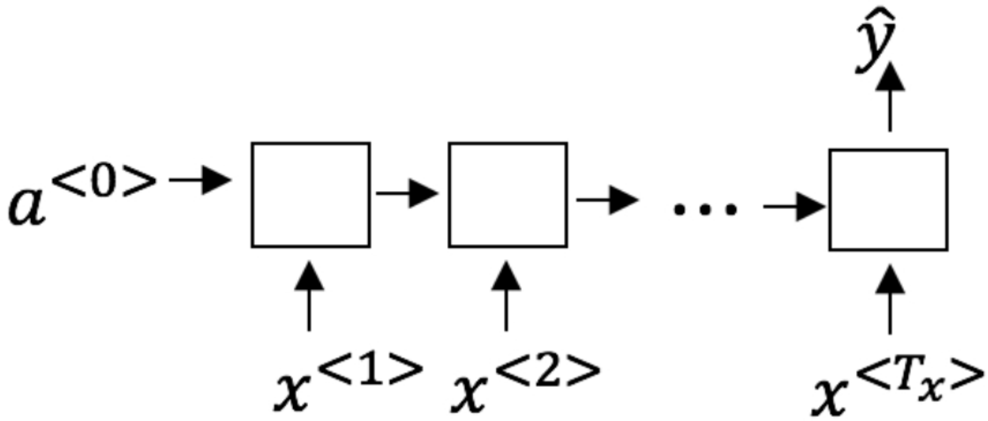
- ◉ False
- ○ True

더 보기

맞습니다
Correct! This type of architecture is for applications where the input and output sequence length is the same.

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

1 / 1점



- [ ] Speech recognition (input an audio clip and output a transcript)
- [x] Sentiment classification (input a piece of text and output a 0/1 to denote positive or negative sentiment)

✓ **Correct**
Correct!

- [ ] Image classification (input an image and output a label)
- [x] Gender recognition from speech (input an audio clip and output a label indicating the speaker's gender)

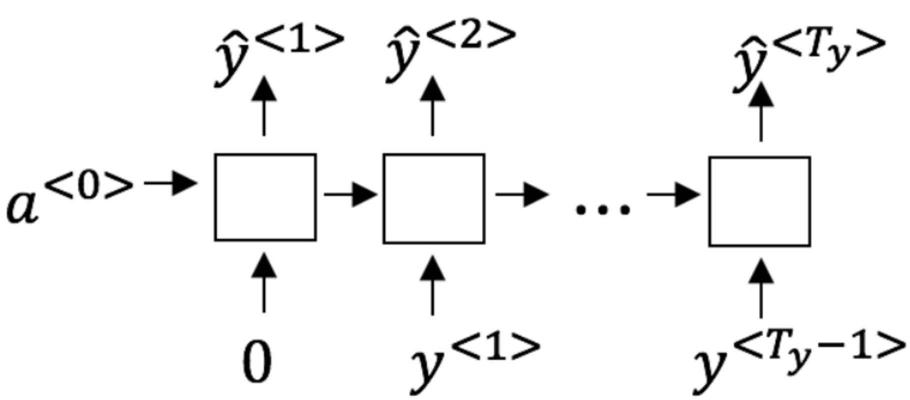
✓ **Correct**
Correct!

더 보기

맞습니다
Great, you got all the right answers.

4. You are training this RNN language model.

1 / 1점



At the t^{th} time step, what is the RNN doing?

- ○ Estimating $P(y^{<1>}, y^{<2>}, \ldots, y^{<t-1>})$
- ○ Estimating $P(y^{<t>})$

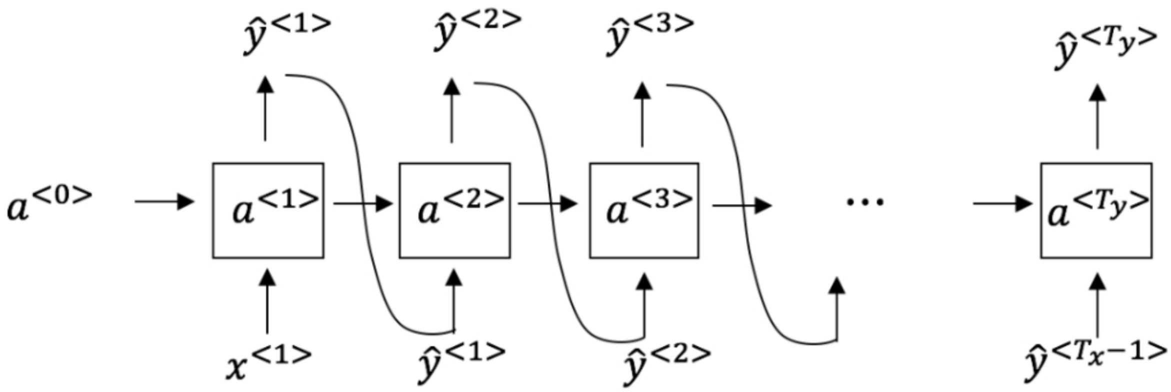
Estimating $P(y^{<t>} \mid y^{<1>}, y^{<2>}, \ldots, y^{<t-1>})$

더 보기

맞습니다
Yes, in a language model we try to predict the next step based on the knowledge of all prior steps.

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

1 / 1점



What are you doing at each time step t?

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

더 보기

맞습니다

6. True/False: If you are training an RNN model, and find that your weights and activations are all taking on the value of NaN ("Not a Number") then you have a vanishing gradient problem.

0 / 1점

- True
- False

더 보기

틀립니다
No, vanishing and exploding gradients are common problems in training RNNs, but in the case of this problem, your weights and activations taking on the value of NaN does not imply that you have a vanishing gradient problem.

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

1/1점

- ○ 1
- ◉ 100
- ○ 300
- ○ 10000

더 보기

맞습니다
Correct, Γ_u is a vector of dimension equal to the number of hidden units in the LSTM.

8. True/False: In order to simplify the GRU without vanishing gradient problems even when training on very long sequences you should remove the Γ_r i.e., setting $\Gamma_r = 1$ always.

1/1점

- ◉ True
- ○ False

더 보기

맞습니다
If $\Gamma_u \approx 0$ for a timestep, the gradient can propagate back through that timestep without much decay. For the signal to backpropagate without vanishing, we need $c^{<t>}$ to be highly dependent on $c^{<t-1>}$.

9. Here are the equations for the GRU and the LSTM:

1/1점

GRU

$$\tilde{c}^{<t>} = \tanh(W_c[\Gamma_r * c^{<t-1>}, x^{<t>}] + b_c)$$

$$\Gamma_u = \sigma(W_u[\, c^{<t-1>}, x^{<t>}] + b_u)$$

$$\Gamma_r = \sigma(W_r[\, c^{<t-1>}, x^{<t>}] + b_r)$$

$$c^{<t>} = \Gamma_u * \tilde{c}^{<t>} + (1 - \Gamma_u) * c^{<t-1>}$$

$$a^{<t>} = c^{<t>}$$

LSTM

$$\tilde{c}^{<t>} = \tanh(W_c[a^{<t-1>}, x^{<t>}] + b_c)$$

$$\Gamma_u = \sigma(W_u[\, a^{<t-1>}, x^{<t>}] + b_u)$$

$$\Gamma_f = \sigma(W_f[\, a^{<t-1>}, x^{<t>}] + b_f)$$

$$\Gamma_o = \sigma(W_o[\, a^{<t-1>}, x^{<t>}] + b_o)$$

$$c^{<t>} = \Gamma_u * \tilde{c}^{<t>} + \Gamma_f * c^{<t-1>}$$

$$a^{<t>} = \Gamma_o * \tanh c^{<t>}$$

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 blanks?

- ◉ Γ_u and $1 - \Gamma_u$

- Γ_u and $1-\Gamma_u$
- Γ_u and Γ_r
- $1-\Gamma_u$ and Γ_u
- Γ_r and Γ_u

더 보기

맞습니다
Yes, correct!

10. True/False: You would use unidirectional RNN if you were building a model map to show how your mood is heavily dependent on the current and past few days' weather.

0 / 1점

- True
- False

더 보기

틀립니다
Your mood is contingent on the current and past few days' weather, not on the current, past, AND future days' weather.