## Congratulations! You passed!

Grade received 100%

Latest Submission Grade 100%

To pass 80% or higher

Go to next item

**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 / 1 point

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

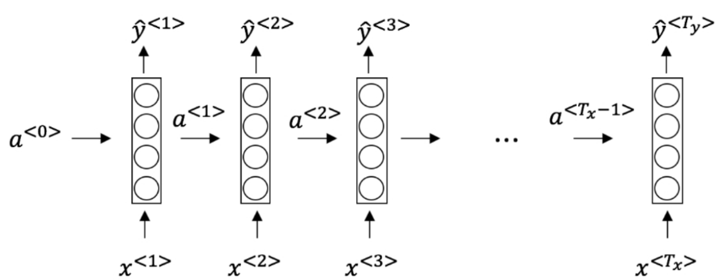
36:21

Expand

**Correct**
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 / 1 point




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

- (selected) True
- False

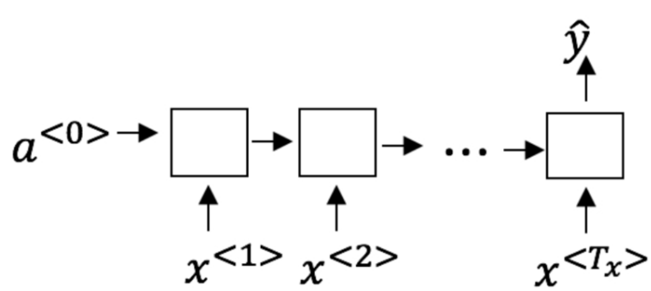
36:20

Expand

**Correct**
It is appropriate when the input sequence and the output sequence have the same length or size.

**3.** To which of these tasks would you apply a many-to-one RNN architecture?

1 / 1 point



- [ ] Image classification (input an image and output a label)
- [x] Music genre recognition
- [x] Language recognition from speech (input an audio clip and output a label indicating the language being spoken)
- [ ] Speech recognition (input an audio clip and output a transcript)

36:19

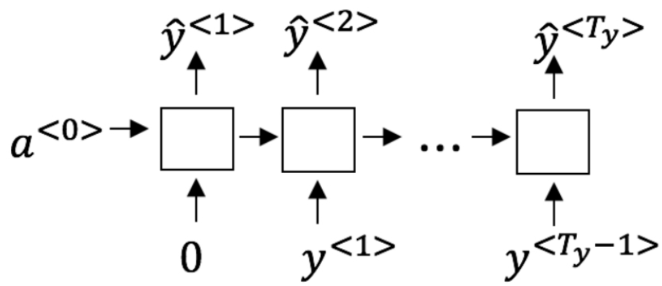
Expand

**Correct**

Great, you got all the right answers.

**4.** Using this as the training model below, answer the following:

**1 / 1 point**




True/False: At the $t^{th}$ time step the RNN is estimating $P(y^{<t>})$

- ( ) True
- (•) False

36:19

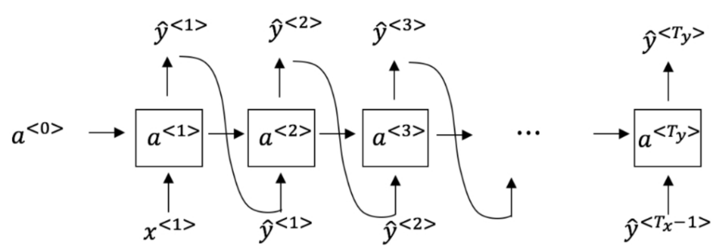
Expand

**Correct**

No, in a training model we try to predict the next steps 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 / 1 point**




True/False: In this sample sentence, step t uses the probabilities output by the RNN to pick the highest probability word for that time-step. Then it passes the ground-truth word from the training set to the next time-step.

- (•) False
- ( ) True

36:18

Expand

**Correct**

The probabilities output by the RNN are not used to pick the highest probability word and the ground-truth word from the training set is not the input to the next time-step.

truth word from the training set is not the input to the next time-step.

**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.

**1 / 1 point**

- False
- True

36:17

Expand

**Correct**
Vanishing and exploding gradients are common problems in training RNNs, but in this case, your weights and activations taking on the value of NaN implies you have an exploding gradient problem.

**7.** Suppose you are training an LSTM. You have a 50000 word vocabulary, and are using an LSTM with 500-dimensional activations $a^{<t>}$. What is the dimension of $\Gamma_u$ at each time step?

**1 / 1 point**

- 500
- 200
- 50000
- 5

36:17

Expand

**Correct**
Correct, $\Gamma_u$ is a vector of dimension equal to the number of hidden units in the LSTM.

**8.** Sarah proposes to simplify the GRU by always removing the $\Gamma$u. I.e., setting $\Gamma$u = 0. Ashely proposes to simplify the GRU by removing the $\Gamma$r. I. e., setting $\Gamma$r= 1 always. Which of these models is more likely to work without vanishing gradient problems even when trained on very long input sequences?

**1 / 1 point**

GRU

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

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

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

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

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

- Sarah's model (removing $\Gamma_u$), because if $\Gamma_r \approx 1$ for a timestep, the gradient can propagate back through that timestep without much decay.
- Ashely's model (removing $\Gamma_r$), because if $\Gamma_u \approx 0$ for a timestep, the gradient can propagate back through that timestep without much decay.
- Ashely's model (removing $\Gamma_r$), because if $\Gamma_u \approx 1$ for a timestep, the gradient can propagate back through that timestep without much decay.
- Sarah's model (removing $\Gamma_u$), because if $\Gamma_r \approx 0$ for a timestep, the gradient can propagate back through that timestep without much decay.

36:16

Expand

**Correct**
Yes. 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 / 1 point

| GRU | LSTM |
|---|---|
| $\tilde{c}^{<t>} = \tanh(W_c[\Gamma_r * c^{<t-1>}, x^{<t>}] + b_c)$ | $\tilde{c}^{<t>} = \tanh(W_c[a^{<t-1>}, x^{<t>}] + b_c)$ |
| $\Gamma_u = \sigma(W_u[\, c^{<t-1>}, x^{<t>}] + b_u)$ | $\Gamma_u = \sigma(W_u[\, a^{<t-1>}, x^{<t>}] + b_u)$ |
| $\Gamma_r = \sigma(W_r[\, c^{<t-1>}, x^{<t>}] + b_r)$ | $\Gamma_f = \sigma(W_f[\, a^{<t-1>}, x^{<t>}] + b_f)$ |
| $c^{<t>} = \Gamma_u * \tilde{c}^{<t>} + (1 - \Gamma_u) * c^{<t-1>}$ | $\Gamma_o = \sigma(W_o[\, a^{<t-1>}, x^{<t>}] + b_o)$ |
| $a^{<t>} = c^{<t>}$ | $c^{<t>} = \Gamma_u * \tilde{c}^{<t>} + \Gamma_f * c^{<t-1>}$ |
| | $a^{<t>} = \Gamma_o * c^{<t>}$ |

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?

- (●) $\Gamma_u$ and $1 - \Gamma_u$
- ( ) $\Gamma_u$ and $\Gamma_r$
- ( ) $1 - \Gamma_u$ and $\Gamma_u$
- ( ) $\Gamma_r$ and $\Gamma_u$

36:15

Expand

Correct
Yes, correct!

**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 $x^{<1>}, \ldots, x^{<365>}$. You've also collected data on your dog's mood, which you represent as $y^{<1>}, \ldots, y^{<365>}$. You'd like to build a model to map from $x \rightarrow y$. Should you use a Unidirectional RNN or Bidirectional RNN for this problem?

1 / 1 point

- ( ) 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 $y^{<t>}$ depends only on $x^{<1>}, \ldots, x^{<t>}$, but not on $x^{<t+1>}, \ldots, x^{<365>}$
- ( ) Unidirectional RNN, because the value of $y^{<t>}$ depends only on $x^{<t>}$, and not other days' weather.

36:14

Expand

Correct
Yes!