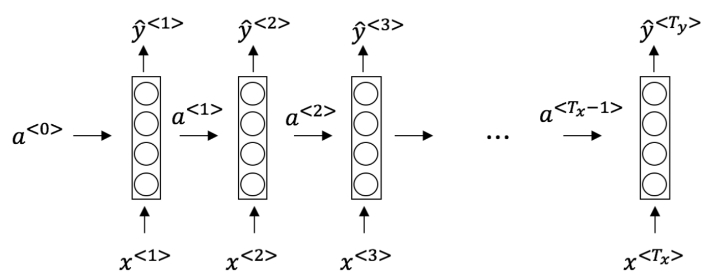
Week 1: Recurrent Neural Networks

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

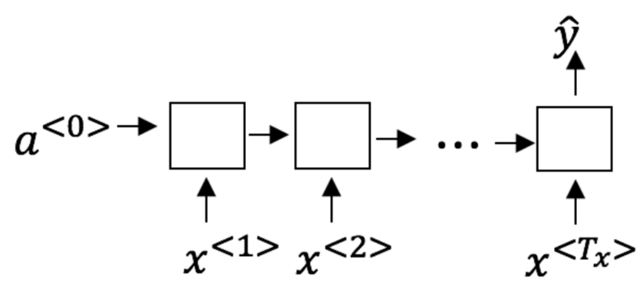
   Ans: $x^{(i)<j>}$

2. Consider this RNN. This type of architecture is appropriate when,



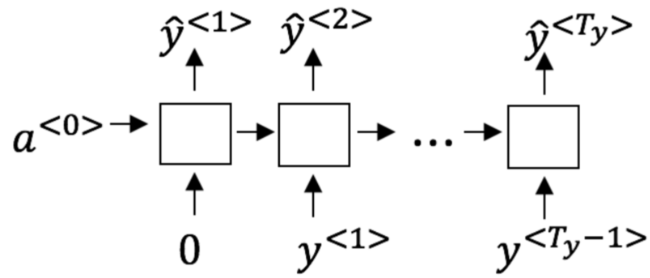
   Ans: $T_x = T_y$

3. What tasks can you apply this many-to-one RNN architecture?



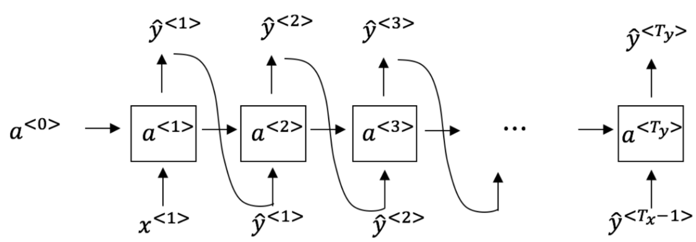
   Ans: Sentiment classification, Gender identification

4. You are training this model. What is it doing in the t^{th} step?



Ans: Estimating $P(y^{<t>}|y^{<t-1>}, y^{<t-2>} \dots y^{<1>})$

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



Ans: Use probabilities output to randomly sample a chosen word, and then pass this selected word to the next time step.

6. Suppose you find that your weights and activations are all taking on the value of NaN. Which of these is the most likely cause of this problem?

Ans: Exploding gradients

7. Suppose you are training a LSTM. You have a 10,000 word vocabulary, and are using an LSTM with 100-dimensional activations $a^{<t>}$. What is the dimension of Γ_u at each time step?

Ans: 100

8. Given equations are for update in GRU, Alice proposes to simplify the GRU by always removing the Γ_u, i.e. setting $\Gamma_u = 1$. Betty proposes to simplify by removing the Γ_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?

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

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

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

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

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

Ans: Betty's model (setting $\Gamma_r = 1$) because if $\Gamma_u \approx 0$ for a time-step, the gradient can propagate without much decay.

9. Givenare the equations for GRU, LSTM. Update and Forget gates play a role similar to?

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

Ans: Γ_u and $1 - \Gamma_u$

10. You have a pet dog whose mood is dependent on the current and past few days' weather. You have 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 $x \rightarrow$. Should you use a Unidirectional or Bidirectional RNN ?

Ans: Unidirectional, because y depends only on current and past values