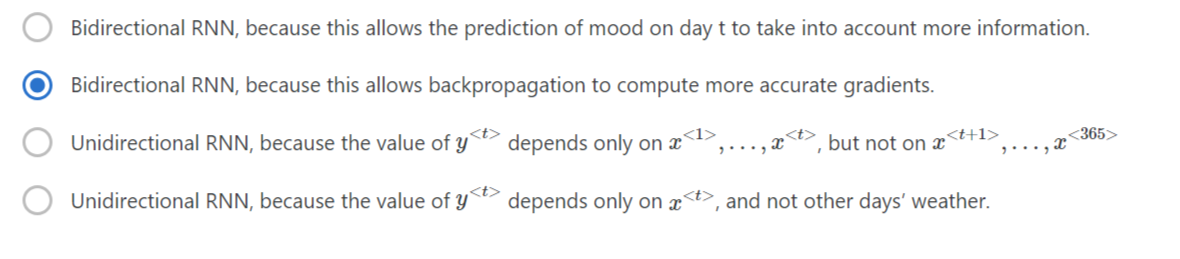
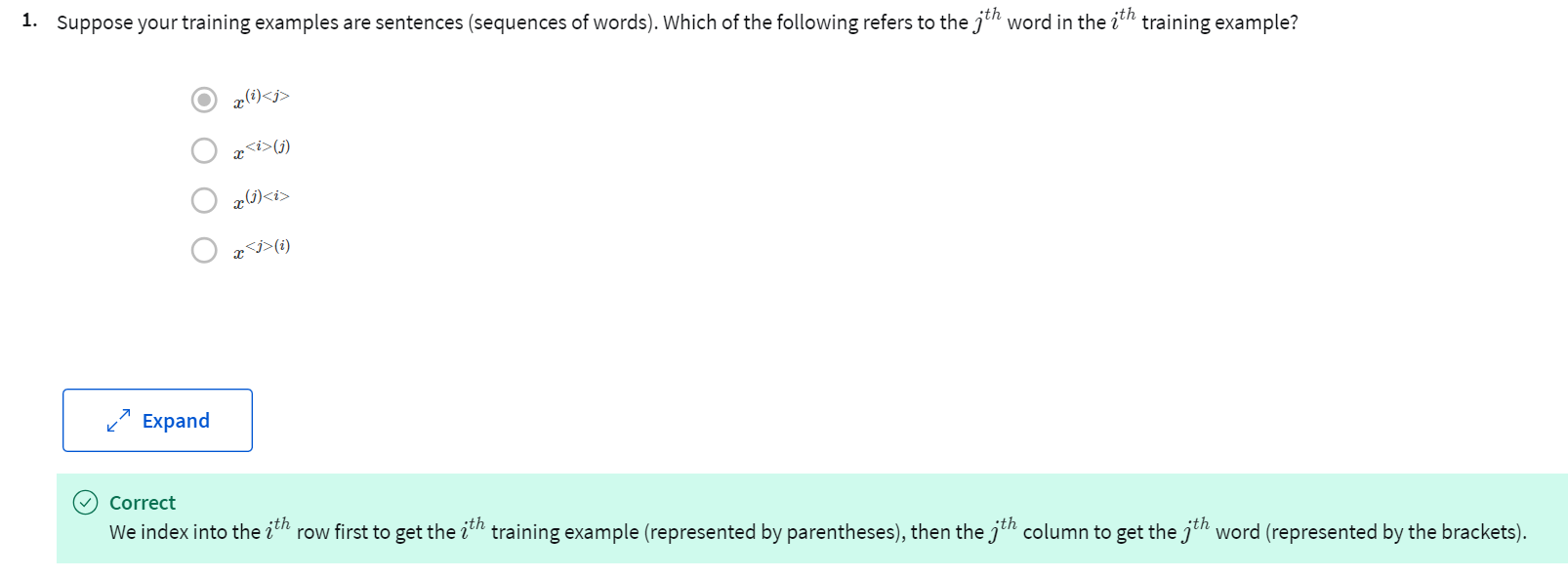
Please help me to answer the questions as below:

Question:

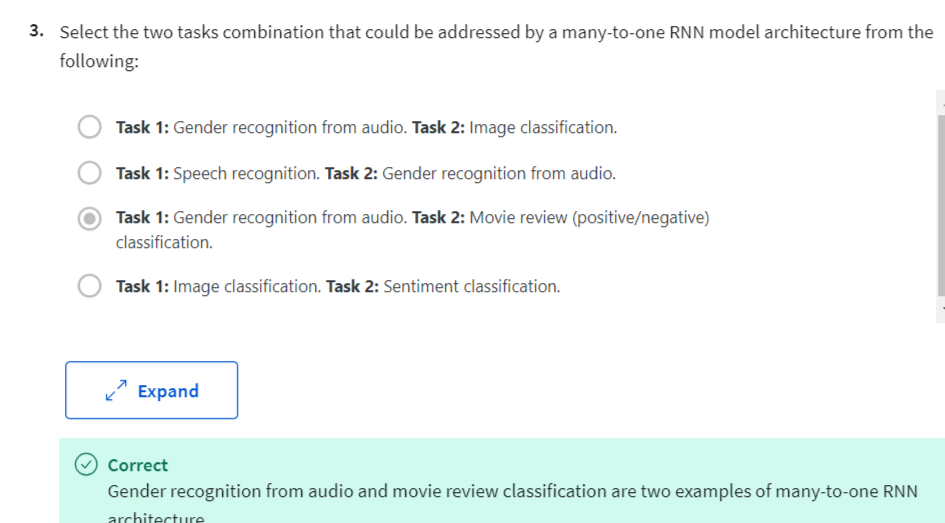
You have a pet dog whose mood is heavily dependent on the current and past few days’ weather. You’ve collected data for the past 365 days on the weather, which you represent as a sequence as x<1>,…,x<365>*x*<1>,…,*x*<365>. You’ve also collected data on your dog’s mood, which you represent as y<1>,…,y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

1. Bidirectional RNN, because this allows the prediction of mood on day t to take into account more information.
2. Bidirectional RNN, because this allows back propagations to compute more accurate gradients.
3. Unidirectional RNN because the value y<t> depends only on x<1> ….., X<t>, but not on x<t-1>, … x<305>
4. Unidirectional RNN because the value of y<t> depends only on x<t> and not other days’weather.





|  |
| --- |
|  |
|  |



|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

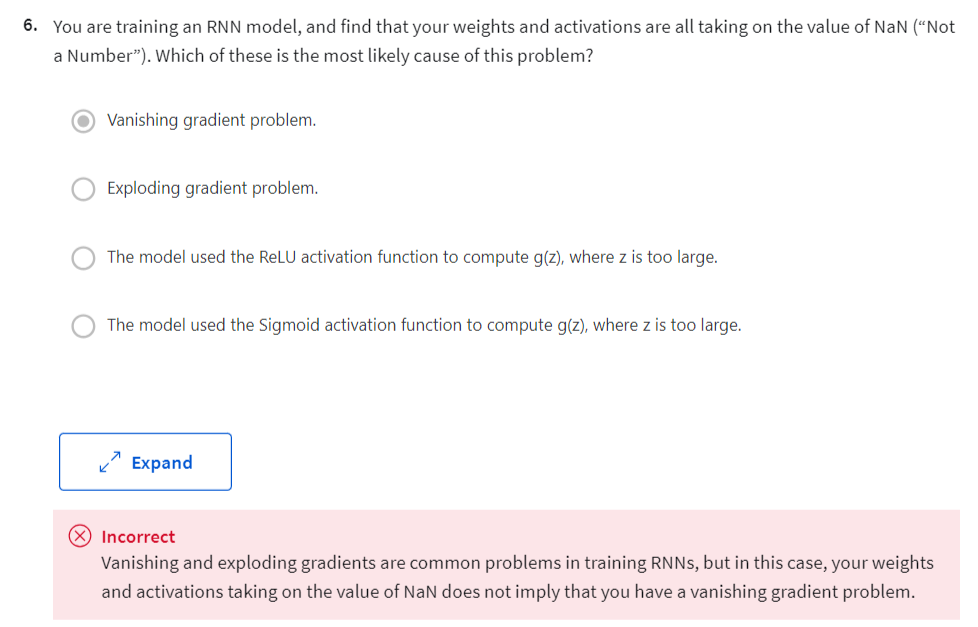
TRY: D

Q 5: Please help me to answer the questions as below:

Question:

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

1. (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.
2. (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.
3. (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 tie-step.
4. **(i) Use the probabilities output by the RNN to randomly sample a chosen word for that time-step y^<t>. (ii) Then pass this selected word to the next time-step.**



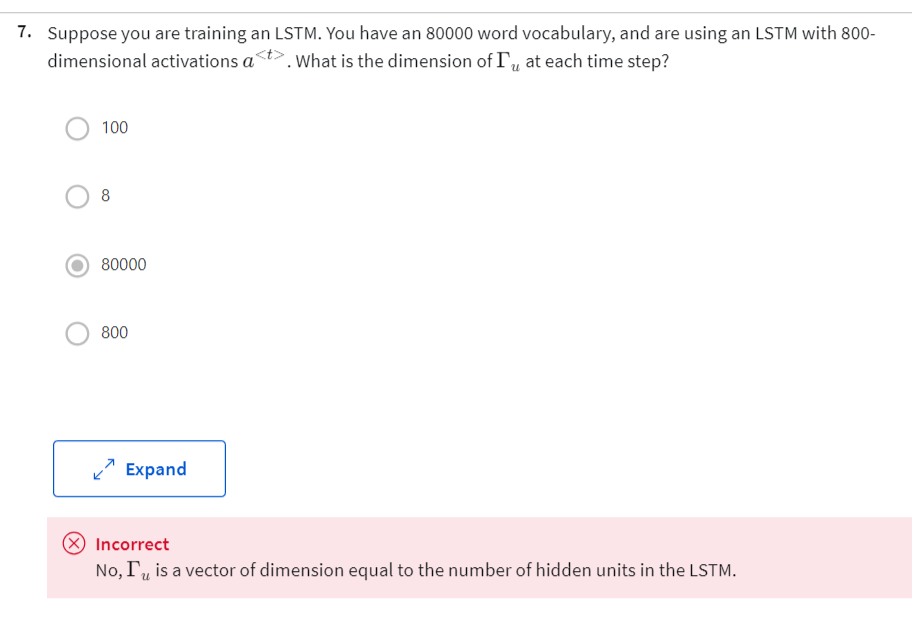
**TRY: B**

Q 6:

Please help me to answer the questions as below:

Question: You are training an RNN model, 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?

1. Vanishing gradient problem? Or
2. **Exploding gradient problem? Or**
3. The model used the RELU activation function to compute g(z), where z is too large? Or
4. The model used the Sigmoid activation function to compute g(z), where z is too large?



**TRY: D - 800**

Q 7:

Please help me to answer the questions as below:

Suppose you are training an LSTM. You have an 80000 word vocabulary, and are using an LSTM with 800-dimensional activations a<t>. What is the dimension of Γu at each time step?

1. 100? Or
2. 8? or
3. 80000? or
4. **800?**

|  |
| --- |
|  |
|  |

**TRY D**

Question 8

Please help me to answer the following question:

Here are the update equations for the GRU.

GRU

C~<t> = tanh(Wc[Γr \* c<t-1>, x<t>] + bc)

Γu = Sigma(Wu[c<t-1>, x<t>] + bu)

Γr = Sigma(Wr[c<t-1>, x<t>] + br)

c<t> = Γu \* c~<t> + (1 – Γu) \* c<t-1>

a<t> = c<t>

Alice proposes to simplify the GRU by always removing the Γu​. I.e., setting Γu = 0. Betty proposes to simplify the GRU by removing the Γr. I. e., setting Γr ​ = 1 always. Which of these models is more likely to work without vanishing gradient problems even when trained on very long input sequences?

1. Alice’s model (removing Γu), because if Γr going to be zero for a timestep, the gradient can propagate back through that timestep without much decay.
2. Alice’s model (removing Γu), because if Γr going to be 1 for a timestep, the gradient can propagate back through that timestep without much decay.
3. Betty’s model (removing Γr), because if Γu going to be zero for a timestep, the gradient can propagate back through that timestep without much decay.
4. **Betty’s model (removing Γr), because if Γu going to be 1 for a timestep, the gradient can propagate back through that timestep without much decay.**

|  |
| --- |
|  |
|  |

**TRY a**

Please help me to answer the following question:

Here are the equations for the GRU and the LSTM:

GRU Equations as below:

c~<t> = tanh(Wc[Γr \* c<t-1>, x<t>] + bc)

Γu = Sigma(Wu[c<t-1>, x<t>] + bu)

Γr = Sigma(Wr[c<t-1>, x<t>] + br)

c<t> = Γu \* c~<t> + (1 – Γu) \* c<t-1>

a<t> = c<t>

LSTM Equations:

c~<t> = tanh(Wc[Γr \* c<t-1>, x<t>] + bc)

Γu = Sigma(Wu[c<t-1>, x<t>] + bu)

Γf = Sigma(Wf[c<t-1>, x<t>] + bf)

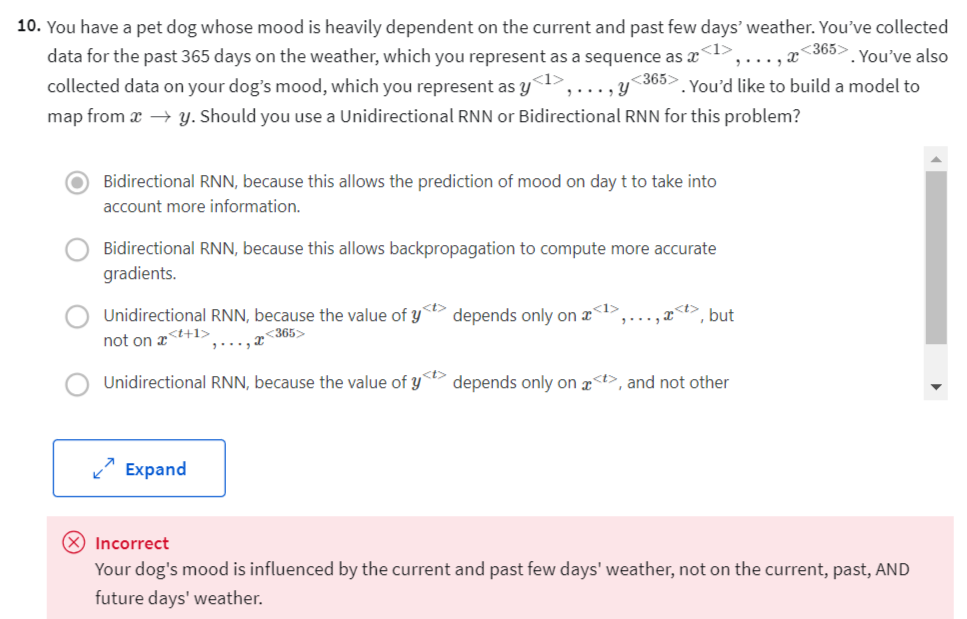
Γo = Sigma(Wo[c<t-1>, x<t>] + bo)

c<t> = Γu \* c~<t> + Γf \* c<t-1>

a<t> = Γo \* tanhc<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?

1. **Γu and 1- Γu? Or**
2. Γu and Γr? Or
3. 1- Γu and Γu? Or
4. Γr and Γu?



**TRY C**

Please help me to answer the following question:

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→y. Should you use a Unidirectional RNN or Bidirectional RNN for this problem?

1. Bidirectional RNN, because this allows the prediction of mood on day t to take into account more information? OR
2. Bidirectional RNN, because this allows backpropagation to compute more accurate gradients? or
3. **Unidirectional RNN, because the value of y<t> depends only on x<1>, ….., x<t>, but not on x<t+1>, …., x<365> ? or**
4. Unidirectional RNN, because the value of y<t> depends only on x<t>, and not other day’s weather?