Applied AI Course

Follow 337 Followers About

FAQ's about attention assignment

Applied AI Course Jul 7, 2020 · 4 min read

Before going through this assignment, you have to be comfortable with how encoder-decoder models work. You can check this nice blog.

Task-2 queries:

1. Do not define methods, layers in the call method for any class, You have to create layers or methods from the __init__ or build method and use them in the call method.

2. How to initialize encoder states?

Ans: Initialize the state_h and state_c values with all zeros.

3. How to initialize decoder_states?

> Initialize decoder LSTM states with previous states. This is very important else your model will not train properly.

For a sample decoder input <start> Hi how are you doing <end>

In decoder				
Time step	Input to decoder	Hidden State	Cell state	Output
1	<start>	Encoder final Hidden state	Encoder final cell state	Decoder output-t1,hidden state-t1, cell state -t1
2	Hi	Hidden state-t1	Cell state-t1	Decoder output-t2,hidden state-t2, cell state -t2
3	How	Hidden state-t2	Cell state-t2	Decoder output-t3,hidden state-t3, cell state -t3
4	Are	Hidden state-t3	Cell state-t3	Decoder output-t4,hidden state-t4, cell state -t4

Fig: Decoder LSTM for a sample input

> *For initializing the states, you have an inital_state attribute in LSTM, please check the documentation.*

4. What is teacher forcing?

If you are having Decoder input and output as same, Say I want to predict Hindi to English.

English sentence is — — <Start> Hi How are you <End>

So at the first-time step, you will pass <Start> and you expect your model to predict ***Hi***, not ***<Start>***.

If you want the model to predict the same input as output then why do you need such a complex network?. So your decoder output will be a one-time step ahead of decoder input.

5. How to train your model?

There are many ways to train your model. Say you are translating Hindi to English

1. Method -1

Encoder input should be — <start> Hindi sentence <end>

Decoder input should be — <start> what is your name?

Decoder output should be — what is your name? <end>

model.fit([encoder_input,decoder_input],decoder_output)

2. Method -2

Encoder input should be — <start> Hindi sentence <end>

Decoder input should be — <start> what is your name? <end>

model.fit([encoder_input,decoder_input[:,:-1]],decoder_output[:,1:])

att_units is used in general and concate scoring function, that is just used in matrix multiplication.

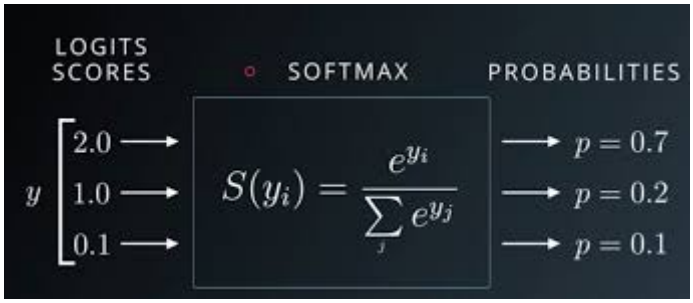
7. What is the custom loss function mentioned in the assignment notebook?

We have a lot of padded zeros for each data point, we do not want to consider the loss for this padded zeros. i.e. We do not worry about what will the model predict in this padded zeros. Please consider the loss mentioned in the blog.

8. Why should we not have an activation unit as softmax in my final dense layer(In one step decoder)?

```
loss_object = tf.keras.losses.SparseCategoricalCrossentropy(
    from_logits=True, reduction='none')
```

As we mentioned, from_logits=True this is the output before softmax.



Logits vs probabilities.

Please remove the activation unit as softmax for your final dense layer in your one_step_decoder.

9. Why you should not apply one more dense layer at the end of the encoder-decoder Model?

Your output from the decoder is coming from a dense layer(keep the same, do not apply softmax as activation unit), so no need to apply one more dense layer to the decoder output. Return the decoder output.

If you apply dense layer then calculate the parameters for that dense layer will be (vocabsize*vocabsize). If vocab size is close to 2000 then you will be having 4M

10. Getting shape error at the end of the epoch, what can be the reason?

What is happening is say you have 130 datapoints and your batch size is 64.

→ 1 st batch 64 points will be used

→ 2 nd batch 64 points will be used

→ 3 rd batch 2 points will be used. In this case, my batch size is 2, which is creating the problem

While creating a dataset you have to use drop_remainder=True as mentioned in this blog. — So 3rd batch will not be run

or

Initialize the hidden states of encoder-LSTM based on the current batch size.

11. Getting no gradients available error?

```
   ValueError: No gradients provided for any variable:
'encoder_decoder_2/encoder_3/embedding_layer_encoder/embedding
encoder_decoder_2/encoder_3/Encoder_LSTM/kernel:0',
encoder_decoder_2/encoder_3/Encoder_LSTM/recurrent_kernel:0',
encoder_decoder_2/encoder_3/Encoder_LSTM/bias:0',
encoder_decoder_2/decoder_5/one_step_decoder_8/embedding_lay
```

Change this line:

from → all_output.write(i,output)

to → all_output=all_output.write(i,output)

(Here all_output is a tensorArray)

11. Any improvements apart from the assignment instructions to improve the predictions?

12. Any problem when running the reference notebook with GPU?

Try this notebook, if you are facing any issues with the reference notebook.

13. What is the difference between Bahdanu's attention and Luong's attention?

Ans: Both of these are attention-based models, with different architecture. Please refer to this link.

Unlisted