✓ **Congratulations! You passed!**
TO PASS 80% or higher

Keep Learning

GRADE
100%

Week 4 Quiz

LATEST SUBMISSION GRADE

100%

1. What is the name of the method used to tokenize a list of sentences?   `1 / 1 point`

   ○ fit_to_text(sentences)

   ○ tokenize_on_text(sentences)

   ○ tokenize(sentences)

   ⦿ fit_on_texts(sentences)

   ✓ **Correct**

2. If a sentence has 120 tokens in it, and a Conv1D with 128 filters with a Kernal size of 5 is passed over it, what's the output   `1 / 1 point`
   shape?

   ○ (None, 116, 124)

   ○ (None, 120, 128)

   ○ (None, 120, 124)

   ⦿ (None, 116, 128)

   ✓ **Correct**

3. What is the purpose of the embedding dimension?   `1 / 1 point`

   ○ It is the number of letters in the word, denoting the size of the encoding

   ○ It is the number of dimensions required to encode every word in the corpus

   ○ It is the number of words to encode in the embedding

   ⦿ It is the number of dimensions for the vector representing the word encoding

   ✓ **Correct**

4. IMDB Reviews are either positive or negative. What type of loss function should be used in this scenario?   `1 / 1 point`

   ○ Adam

   ⦿ Binary crossentropy

   ○ Categorical crossentropy

   ○ Binary Gradient descent

   ✓ **Correct**

5. If you have a number of sequences of different lengths, how do you ensure that they are understood when fed into a   `1 / 1 point`
   neural network?

   ○ Make sure that they are all the same length using the pad_sequences method of the tokenizer

   ○ Process them on the input layer of the Neural Network using the pad_sequences property

   ⦿ Use the pad_sequences object from the tensorflow.keras.preprocessing.sequence namespace

   ○ Specify the input layer of the Neural Network to expect different sizes with dynamic_length

   ✓ **Correct**

6. When predicting words to generate poetry, the more words predicted the more likely it will end up gibberish. Why?   `1 / 1 point`

   ○ Because you are more likely to hit words not in the training set

   ⦿ Because the probability that each word matches an existing phrase goes down the more words you create

   ○ Because the probability of prediction compounds, and thus increases overall

   ○ It doesn't, the likelihood of gibberish doesn't change

   ✓ **Correct**

7. What is a major drawback of word-based training for text generation instead of character-based generation? **1 / 1 point**

  ◯ There is no major drawback, it's always better to do word-based training

  ◯ Character based generation is more accurate because there are less characters to predict

  ◯ Word based generation is more accurate because there is a larger body of words to draw from

  ⦿ Because there are far more words in a typical corpus than characters, it is much more memory intensive

  ✓ **Correct**

8. How does an LSTM help understand meaning when words that qualify each other aren't necessarily beside each other in a sentence? **1 / 1 point**

  ◯ They load all words into a cell state

  ◯ They don't

  ⦿ Values from earlier words can be carried to later ones via a cell state

  ◯ They shuffle the words randomly

  ✓ **Correct**