



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

Keep Learning

GRADE  
100%

### You have new skill scores!

Great job! Keep learning and making progress in your courses to increase your skill scores.

Applied Machine Learning

269



Computer Programming

153



Deep Learning

249



Machine Learning

317



Python Programming

234



Statistical Programming

166



## 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

- ☐ tokenize\_on\_text(sentences)
- ☐ fit\_to\_text(sentences)
- ☒ fit\_on\_texts(sentences)
- ☐ tokenize(sentences)

✓ Correct

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

1 / 1 point

- ☐ (None, 116, 124)
- ☐ (None, 120, 128)
- ☒ (None, 116, 128)
- ☐ (None, 120, 124)

✓ 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 for the vector representing the word 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

✓ Correct

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

1 / 1 point

- ☐ Categorical crossentropy
- ☐ Binary Gradient descent
- ☐ Adam
- ☒ Binary crossentropy

✓ Correct

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

1 / 1 point

- ☐ Specify the input layer of the Neural Network to expect different sizes with dynamic\_length
- ☐ Process them on the input layer of the Neural Network using the pad\_sequences property

- ☐ Make sure that they are all the same length using the `pad_sequences` method of the tokenizer
- ☒ Use the `pad_sequences` object from the `tensorflow.keras.preprocessing.sequence` namespace

✓ 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 the probability of prediction compounds, and thus increases overall
- ☒ Because the probability that each word matches an existing phrase goes down the more words you create
- ☐ It doesn't, the likelihood of gibberish doesn't change
- ☐ Because you are more likely to hit words not in the training set

✓ Correct

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

1 / 1 point

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

✓ 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 shuffle the words randomly
- ☐ They don't
- ☒ Values from earlier words can be carried to later ones via a cell state

✓ Correct