

1. What is a MaxPooling2D layer? What's it do?

- a. MaxPooling2D layer is a common component of Convolutional Neural networks that performs image processing tasks. It helps to make the model more robust to small variations in the position of features in the image. It also helps to reduce the computational cost of the model.

2. What's Adam?

- a. Adam is a parameter for convolutional networks used to optimize the model using the Adam algorithm, Adam optimization is a stochastic gradient descent method that is based on adaptive estimation of first-order and second-order moments.

3. What's the softmax function do?

- a. The Softmax function is used in the output layer of a Neural Network for multi-class classification problems. It takes the inputs and outputs a probability of it belonging to a particular class.

4. What is CategoricalCrossEntropy? What do we use it for?

- a. CategoricalCrossEntropy is a loss function that computes how well the model is performing calculating the difference between the predicted distribution and the true distribution of the classes.

5. In the CNN example, what does the Flatten layer do?

- a. In the CNN example the flatten layer takes the multi-dimensional output of the convolutional layers and transforms it into a single, long vector.

6. In the CNN example, what does the Dense layer do?

- a. **Dense layers** in a Neural network perform the final classification task, for example with each neuron in the dense layer it gets the input from all the neurons in the previous layer and applies weights and an activation function like sigmoid to produce an output.

7. In the CNN example, why does the height and width get smaller for each convolutional layer?

- a. In the CNN example the height and width gets smaller because, to afford to add more output channels in each Conv2D layer and the pooling layer are performing downsampling.

The convolutional layer uses filters that slide over the input image, performing calculations that extract features. Maxpooling2D for example reduces the size by selecting only a representative value from a small region of the feature map.

Part 3:

1. What does it mean to normalize the data? Where else have we seen normalization?

- a. Normalizing the data means to adjust the data values to a common scale, this is helpful. Normalization ensures that all features contribute equally to the model's learning process.
- 2. Why is it a problem that the Titanic data has different types and ranges?**
 - a. The Titanic data having different types and ranges is problematic because it has categorical data like, sex and class and numerical data like, age and fare. This is problematic for machine learning models sensitive to feature scales like models that use distance calculations for optimization like KNN's, linear Regression and Neural Networks.
- 3. Why did we not have to worry about this with the decision tree?**
 - a. We don't have to worry about this with decision trees because they don't use distance calculation for optimization instead, they use threshold so, when making a decision it considers a single feature at a time and determines a threshold value to split the data.
- 4. What is a one-hot vector?**
 - a. **One-hot vector** is a way to represent categorical data numerically, it makes a categorical value with a numerical value to a dictionary for example, a categorical feature "Passenger class" with possible values like "first class," "business class," "economic" and represent them like this.
"First class" - 0
"Business class" - 1
"Economic" - 2
- 5. The example that shows how to manually slice the feature dictionary uses yield instead of return. Why is this? What's the difference between them, and why would you want to use yield?**
 - a. In yield keyword within the slices function to create a generator. The difference between yield and return is that when a function encounters a yield statement, it temporarily pauses the execution and returns the yielded value. Return statements when encountered by a function terminate the function and send the returned value.

PART 4:

- 1. As we know, encoding is a particularly important part of working with neural networks. Explain how text is encoded for an RNN.
 - A. Text encoding is an important part of work with neural networks similar to do the project we did in class to cluster text, for neural networks encoding or text vectorization it creates a vocabulary of words from the training data and the it creates a sequence of indices representing the words in the text.

2. What does the Bidirectional layer do? What are the advantages and disadvantages of this approach? How does it compare to the way we processed sequence data with an HMM?
 - a. A bidirectional layer in the context of RNNs process in both forward and backward directions. It consists of two RNN layers, one for each direction. Some advantages are contextual awareness and improved performance, some disadvantages are computational cost and inability for real time comparison. The biggest difference with HMM is that HMM don't have backpropagation for training, unlike RNNs.
3. What is masking? Why do we need to use it in this example?
 - a. In the context of RNNs, padding is used to create batches of equal length, but for that to not influence the network learning process masking is used so some paddings are ignored during the learning process. In the model to predict sentiment analysis of movie reviews. Masking ensures that the padding tokens do not affect the sentiment prediction.