Question 1: Deep Learning Warm-up

2

Which of these techniques might help with overfitting?
(1 punto)

- [ ] Weight Decay
- [ ] Early Stopping
- [ ] Dropout
- [ ] ReLu and Leaky ReLu
- [ ] Stochastic Gradient Descend
- [ ] Xavier Initialization
- [ ] Batch Normalization

3

Which of these techniques might help with vanishing gradient?
(1 punto)

- [ ] Dropout
- [ ] ReLu and Leaky ReLu
- [ ] Early Stopping
- [ ] Xavier Initialization
- [ ] Stochastic Gradient Descend
- [ ] Weight Decay
- [ ] Batch Normalization

4

What is the dying neuron problem and how would you fix it?
(2 punti)

Inserisci la risposta

Indietro

Question 2: Convolutional Neural Networks

```
import tensorflow as tf
tfk = tf.keras
tfkl = tf.keras.layers

input_shape = (256, 256, 3);

# Build the neural network layer by layer
input_layer = tfkl.Input(shape=input_shape, name='Input')

conv1 = tfkl.Conv2D(filters=8, kernel_size=(5, 5), strides = (1, 1),padding = 'same',activ
ation = 'relu', name='conv1')(input_layer)
pool1 = tfkl.MaxPooling2D(pool_size = (2, 2), name='mp1')(conv1)

conv2 = tfkl.Conv2D(filters=32,kernel_size=(5, 5),strides = (2, 2),padding = 'same',activ
ation = 'relu', name='conv2')(pool1)
pool2 = tfkl.MaxPooling2D(pool_size = (2, 2), name='mp2')(conv2)

btchNorm = tfkl.BatchNormalization(name='batchNorm')(pool2) #this normalizes each
slice of the volume independently

conv3 = tfkl.Conv2D(filters=64,kernel_size=(3, 3),strides = (2, 2),padding = 'same',activ
ation = 'relu', name='conv3')(btchNorm)
pool3 = tfkl.MaxPooling2D(pool_size = (2, 2), name='mp3')(conv3)

flattening_layer = tfkl.Flatten(name='Flatten')(pool3)

dropout1 = tfkl.Dropout(0.3)(flattening_layer)
dense1 = tfkl.Dense(units=64, name='Dense1', activation='relu')(dropout1)

dropout2 = tfkl.Dropout(0.3)(dense1)
dense2 = tfkl.Dense(units=32, name='Dense2', activation='relu')(dropout2)

output_layer = tfkl.Dense(units=2, activation='linear', name='Output')(dense2)

# Connect input and output
model = tfk.Model(inputs=input_layer, outputs=output_layer, name='model')

# Consider now the execution of the following command
model.summary()
```

 # TO ANSWER THIS QUESTION, DOWNLOAD THE TXT FILE FROM HERE
 # https://www.dropbox.com/s/yj3gf8cfch4xjnh/2022_01_17_architecture.txt?dl=0 (https://www.dropbox.com/s/yj3gf8cfch4xjnh/2022_01_17_architecture.txt?dl=0)
#
 # THEN COPY THE TEMPLATE DIRECTLY IN THE ANSWER FIELD,
 # KEEP EVERYTHING ALIGNED WITH THE TEMPLATE

Fill in the missing numbers in the console output below. In particular, replace
 # - [ADD SIZES] with the tuple indicating the output shape. Remember that the first si
ze is None because is the minibatch size used during training
 # - [ADD PARAMS] add the formula for computing the parameters and the result (e.g.
8 x 2 x 4 = 64). When this is 0 you can just enter 0

Inserisci la risposta

(5 punti)

6

Consider the above model is compiled as follows.

model.compile(optimizer=tfk.optimizers.Adam(), loss='mse')

Consider also that you are given a dataset of car images together with all the labels ne
eded. What kind of tasks are compatible with the above network?

(1.5 punti)

- [ ] inference on whether the car is red or not
- [ ] inference on height of the driver and his income
- [ ] inference on academic degree of the car owner
- [ ] inference on the cost
- [ ] inference on the cost, the horse power and the maximum speed
- [ ] inference on the maximum speed and the colour
- [ ] inference on the brand and the model
- [ ] inference on its maximum speed

7

Assume you want to develop a system to automatically read the hours from your old w all clock as in the picture.

Now, assume you place a webcam in front of your wall clock, gather a lot of images wit h their acquisition time and you are ready to start training your CNN.

You don't have many images however, and you want your network to be robust to diff erent positioning of the webcam in front of the wall clock, different weather / light con ditions...

What kind of data augmentation **should be avoided** during training?

(1.5 punti)

- [ ] translation
- [ ] scaling of y axis
- [ ] rotation
- [ ] vertical flip
- [ ] noise addition
- [ ] scaling of x axis
- [ ] horizontal flip
- [ ] change in brightness
- [ ] image scaling

blur

8

Briefly describe what Siamese networks are and what they are used for.
(2 punti)

Inserisci la risposta

Indietro

Sequential Models

9

What is the goal of Word Embedding and what motivates it?
(1 punto)

Inserisci la risposta

10

Describe the Word2Vec network architecture.
(2 punti)

Inserisci la risposta

11

Can Word Embedding overfit? If yes, how can it be possible?
(1 punto)

Inserisci la risposta

12

How sentences are embedded in seq2seq modeling?

(1 punto)

Inserisci la risposta

13

How does the attention mechanism work?

(1 punto)

Inserisci la risposta

Indietro