Consider a1DCNNmodelfortimeseriesforecastingwithasequencelengthof100.The modelhasasingleconvolutionallayerwith64filters,eachofsize3,followedbyafully connected layerwith128neurons.Batchnormalizationisappliedafterconvolution layer.Calculatethetotalnumberoftrainableparametersandnon-trainableparameters in themodel.[3]
(b) ConsideraDeepNeuralNetworkwith3hiddenlayersfortimeseriesforecasting.The input sequencehasalengthof50,andthehiddenlayershave64,128,and256neurons, respectively.Calculatethetotalnumberoftrainableparametersinthemodel.Assume the outputsequencelength.[3]
(c) A datascientistwantstobuildaDeepLearningmodelusingasingleLSTMcell for forecastingatime-series.Thetrainingdatasethasseveralrecordswith15time steps each.Eachtimestepconsistsofthree-featurenormalizednumericdata.Iftotal learnable parametersinthismodelare180,whatisthedimensionalityoftheshort state? [3]
(Note: Youranswermustbesupportedbythecalculation/justification/reasoning for allpartsofthisquestion.Therearenomarksforthedirectanswerortheoutput of anytoolorprogramorfordirectscreenshotsorimagesoftextbookpages.)

ChatGPT

(a) Total Number of Trainable and Non-Trainable Parameters in 1DCNN Model:

Given:

- Sequence length: 100
- Single convolutional layer with:
  - 64 filters
  - Filter size: 3
- Fully connected layer with 128 neurons
- Batch normalization after the convolutional layer

Let's calculate the trainable and non-trainable parameters:

Convolutional Layer:

- Trainable parameters:
  $(3 \times 1 + 1) \times 64 = 256$ (weights + bias)
- Non-trainable parameters:
  - Batch normalization has 2 parameters per filter (scale and shift) and 64 filters.
    $2 \times 64 = 128$
- Total parameters in the convolutional layer:
  $256 + 128 = 384$

Fully Connected Layer:

- Trainable parameters:
  $(64 \times 128 + 1) = 8193$ (weights + bias)

Total trainable parameters in the model:
$384 + 8193 = 8577$

Non-trainable parameters:
128 (batch normalization parameters)

(b) Total Number of Trainable Parameters in Deep Neural Network:

Given:

- Input sequence length: 50
- Hidden layers with 64, 128, and 256 neurons
- Output sequence length is not provided

Let's calculate the trainable parameters:

Input to Hidden Layer 1:
$(50 \times 64 + 1) = 3201$ (weights + bias)

Hidden Layer 1 to Hidden Layer 2:
$(64 \times 128 + 1) = 8257$ (weights + bias)

Hidden Layer 2 to Hidden Layer 3:
$(128 \times 256 + 1) = 33024$ (weights + bias)

Total trainable parameters in the model:
$3201 + 8257 + 33024 = 44482$

(c) Dimensionality of the Short State in LSTM Model:

Given:

- Total learnable parameters: 180
- Each LSTM cell has:
  - 4 weight matrices (input gate, forget gate, output gate, cell state)
  - 4 bias vectors
  - Input dimensionality: 3
  - Hidden dimensionality: Unknown (short state dimensionality)

Let's denote the hidden dimensionality as h. Each weight matrix has dimensionality $3 \times h$ and each bias vector has dimensionality h.

Total parameters per LSTM cell:
$3 \times h + h \times h + 3 \times h = 7h$

Since each LSTM cell has $7h$ parameters and there are 180 total parameters, we can solve for h:

$7h = 180$
$h = \frac{180}{7} \approx 25.71$

Therefore, the dimensionality of the short state h in the LSTM model is approximately 26.