2023 DNN EndSem Regular

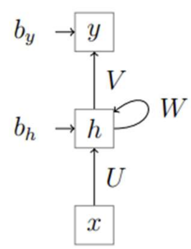
1. Compute the dimensions of W^Q, W^K, W^V, W^O if the input dimensions and attention dimensions are given as $d = 2048$ and the query and key dimensions are $d_k = 512$ and value dimensions are $d_v = 1024$ and the number of head $h = 4$. Compute the dimensions of the attention vector. [5]

   One mark each

$$\text{Dimensions of } W^Q = (h \times d) \times d_k = (4 \times 2048) \times 512$$
$$\text{Dimensions of } W^K = (h \times d) \times d_k = (4 \times 2048) \times 512$$
$$\text{Dimensions of } W^V = (h \times d) \times d_v = (4 \times 2048) \times 1024$$
$$\text{Dimension of attention vector} = d_v = 4 \times 1024$$
$$\text{Output Projection Matrix} W^O = h \times d_v \times d = 4 \times 1024 \times 2048$$

2. Design a single hidden layer recurrent neural network that outputs the moving sum of difference of two input real sequences. For example, left-to-right input sequences $p = \{0.6, 0.4, 0.3, 1.2, -1.2\}$ and $q = \{-1.2, 1.2, 0.3, 0.4, 0.6\}$ will produce the left-to-right output sequence $\{1.8, 1.0, 1.0, 1.8, 0\}$. All nodes use linear activation function $g(x)$, i.e., $g(x) = x$. There are two input unit corresponding to two input sequences and one output unit. Hidden states are denoted by h, as shown in the following figure. Note, at time t,

$$h_t = g(Wh_{t-1} + Ux - b_h) \text{ and } y_t = g(vh_t - b_y)$$



   (a) What is the minimum number of hidden nodes required? Justify [1]

   minimum number of hidden nodes = 1

   (b) Compute W, U, V, b_h and b_y. Only integer values are allowed. Show all computations. [5]

   One mark each

   $b_h = 0; b_y = 0; U = [1 - 1]; W = 1; V = 1$

3. Two historians approach you for your deep learning expertise. They want to classify images of historical objects into 3 classes depending on the time they were created: Antiquity ($y = 0$), Middle Ages ($y = 1$) and Modern Era ($y = 2$)

(A) Class: Antiquity

(B) Class: Middle Ages

(C) Class: Modern Era

2023 DNN EndSem Regular

You come up with a CNN classifier. For each layer, calculate the number of weights, number of biases and the size of the associated feature maps. [8]
The notation follows the convention:

- CONV-K-N denotes a convolutional layer with N filters, each them of size KxK, Padding and stride parameters are always 0 and 1 respectively.
- POOL-K indicates a KxK pooling layer with stride K and padding 0.
- FC-N stands for a fully-connected layer with N neurons.

Layer	Activation map dimensions	Number of weights	Number of biases
INPUT	128 x 128 x 3	0	0
CONV-9-32			
POOL-2			
CONV-5-64			
POOL-2			
CONV-5-64			
POOL-2			
FC-64			
FC-3			

One mark each

Layer	Activation map dimensions	Number of weights	Number of biases
INPUT	128 x 128 x 3	0	0
CONV-9-32	120 x 120 x 32	7776	32
POOL-2	60 x 60 x 32	0	0
CONV-5-64	56 x 56 x 64	51200	64
POOL-2	28 x 28 x 64	0	0
CONV-5-64	24 x 24 x 64	102400	64
POOL-2	12 x 12 x 64	0	0
FC-64	64	589824	64
FC-3	3	192	3

4. A deep learning researcher is designing an LSTM-based neural network for a natural language processing task. The researcher plans to use a single-layer LSTM with specific architectural parameters. The LSTM layer is expected to have an input size of 2^{10} and a hidden size of 2^8. Additionally, the researcher decides not to use bidirectionality in the LSTM layer. The expected output is a 2^{12} vector. Based on these specifications, calculate the total number of parameters required for this LSTM layer. Show Gate wise necessary calculations. [5]
One mark each

$$\begin{aligned}
\text{Parameters for cell state} &= n_i \times n_h + n_h \times n_h + n_h = 2^{10} \times 2^8 + 2^8 \times 2^8 + 2^8 \\
\text{Parameters for input gate} &= n_i \times n_h + n_h \times n_h + n_h = 2^{10} \times 2^8 + 2^8 \times 2^8 + 2^8 \\
\text{Parameters for forget gate} &= n_i \times n_h + n_h \times n_h + n_h = 2^{10} \times 2^8 + 2^8 \times 2^8 + 2^8 \\
\text{Parameters for output gate} &= n_i \times n_h + n_h \times n_h + n_h = 2^{10} \times 2^8 + 2^8 \times 2^8 + 2^8 \\
\text{Parameters for output projection} &= n_h \times n_o + n_o = 2^8 \times 2^{12} + 2^{12}
\end{aligned}$$

2023 DNN EndSem Regular

5. Consider a simplified version of the NiN architecture with one NiN block followed by global average pooling and a softmax output layer. Each NiN block consists of the following layers: 1x1 convolutional layer with 20 filters, three separate 3x3 convolutional layers, each with 50 filters, two separate 5x5 convolutional layers with 20 filters, ReLU activation function applied after each convolutional layer. Assume that the input to the NiN architecture is a greyscale image with dimensions of 256x256 pixels. Calculate the following:

   (a) The number of parameters (weights and biases) in one NiN block. [4]

   (b) The dimensions of the output feature map after the NiN block. [1]

   (c) The total number of parameters in the entire NiN architecture. [1]

   Rubric

   - Number of parameters in one NiN block:

$$
\begin{aligned}
\text{1x1 Convolutional layer } Weights + bias &= 1 \times 1 \times 1 \times 20 + 20 = 40 && [1] \\
\text{3x3 Convolutional layer } Weights + bias &= 3 \times 3 \times 20 \times 50 + 50 = 9050 \\
\text{For three 3x3 Convolutional layers } &= 3 \times 9050 = 27150 && [1] \\
\text{5x5 Convolutional layer } Weights + bias &= 5 \times 5 \times 20 \times 20 + 20 = 10020 \\
\text{For two 5x5 Convolutional layers } &= 2 \times 10020 = 20040 && [1] \\
\text{Total parameters in onw NiN block } &= 40 + 27150 + 20040 = 47230 && [1]
\end{aligned}
$$

- Dimensions of the output feature map after the NiN block: Assume that the convolutional layers use "same" padding. Output feature map after the NiN block will be of size 256x256. [1]
- Since the NiN architecture consists of one NiN block followed by global average pooling and a softmax output layer, the NiN block contains 47230 parameters. [1]