USN 1RV22AI007

RV COLLEGE OF ENGINEERING®

(An Autonomous Institution affiliated to VTU)

V Semester B. E. Regular Examinations Feb/Mar-2025

Artificial Intelligence and Machine Learning

ARTIFICIAL NEURAL NETWORKS AND DEEP LEARNING

Time: 03 Hours ***Maximum Marks: 100***

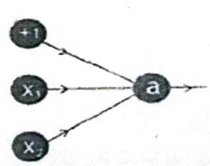
Instructions to candidates:

1. Answer all questions from Part A. Part A questions should be answered in first three pages of the answer book only.
2. Answer FIVE full questions from Part B. In Part B question number 2 is compulsory. Answer any one full question from 3 and 4, 5 and 6, 7 and 8, 9 and 10.

PART-A

<table>
<tr><th></th><th></th><th></th><th>M</th><th>BT</th><th>CO</th></tr>
<tr><td>1</td><td>1.1</td><td>Represent a non-linear neuron model using labeled diagram.</td><td>02</td><td>1</td><td>1</td></tr>
<tr><td></td><td>1.2</td><td>Let us assume we implement an AND function to a single neuron. Below tabular representation of an AND function:
<table>
<tr><th>X1</th><th>X2</th><th>X1 AND X2</th></tr>
<tr><td>0</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>0</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td>1</td></tr>
</table>
The activation function of our neuron is denoted as:

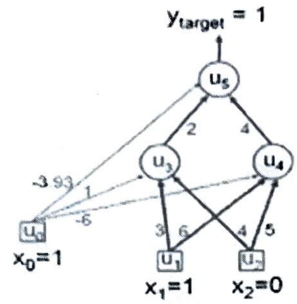
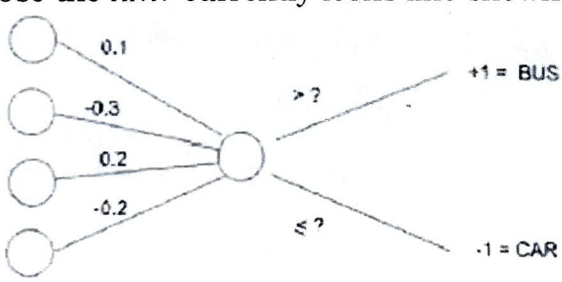
$$f(x) = \begin{cases} 0, & for\ x < 0 \\ 1, & for\ x \geq 0 \end{cases}$$



What should be the weights and bias? (Hint: For which values of $w1, w2$ and b does our neuron implement an AND function?)
a) $Bias = -1.5, w1 = 1, w2 = 1$
b) $Bias = 1.5, w1 = 2, w2 = 2$
c) $Bias = 1, w1 = 1.5, w2 = 1.5$
d) None of these</td><td>02</td><td>3</td><td>1</td></tr>
<tr><td></td><td>1.3</td><td>Examine the effectiveness of the ReLU layer compared to other activation functions (sigmoid and Tanh) in CNNs.</td><td>02</td><td>4</td><td>2</td></tr>
<tr><td></td><td>1.4</td><td>How does data augmentation influence overfitting and generalization in CNN models?</td><td>02</td><td>1</td><td>2</td></tr>
<tr><td></td><td>1.5</td><td>Justify why Long Short-Term Memory (LSTM) units should be utilized to address the vanishing gradient problem in RNNs</td><td>02</td><td>3</td><td>2</td></tr>
<tr><td></td><td>1.6</td><td>How would you analyze the application of Recurrent Neural Networks (RNNs) in automatic image captioning and what role do they play in generating textual descriptions of images?</td><td>02</td><td>2</td><td>2</td></tr>
<tr><td></td><td>1.7</td><td>Evaluate the challenges of Reinforcement Learning in real-world applications like self-driving cars and conversational systems and suggest possible solutions.</td><td>02</td><td>3</td><td>3</td></tr>
<tr><td></td><td>1.8</td><td>Analyze the role of Deep Reinforcement Learning in the development of AlphaGo and AlphaZero, highlighting their differences in human knowledge and self-learning.</td><td>02</td><td>4</td><td>3</td></tr>
</table>

1.9	Draw the architecture of the Neural Turing Machine and represent the components.	02	1	1
1.10	Identify the three major limitations of Neural Turing Machines.	02	1	1

PART-B

2	a	The network weights have been initialized as shown in Fig 2a. Analyze the Multilayer perceptron back propagation algorithm for the initialized network by doing the following (use the sigmoid function):  Fig. 2a. i) Compute the output of hidden layers and output layer. ii) Calculate the error and delta values. iii) Find the new weights and show how the new error is reduced (Up to 2 iterations)	10	3	3
	b	Suppose we are training a neural network to learn a function that takes three integers as input, representing weight, fuel capacity and passenger numbers and outputs either "car" or "bus". Suppose the *ANN* currently looks like shown in Fig 2b.  Fig: 2b. i) Does the network predict a bus or a car for these inputs: (1, 10, 12, and 13)? ii) Suppose that the input (1, 10, 12, and 13) has been miscategorized by the perceptron. Using the perceptron learning rule, calculate the weight change for the weights in the network in light of this training network, if we use a learning rate of 0.1. iii) What does the re-trained network look like? iv) Does the re-trained network (feed-forward) correctly categorize the example (10, 12, 13)? v) Has the network over-corrected? This triple: (5, 7, 7) used to be (correctly) categorized as a car. Is it still correctly categorized?	06	3	3

3 a Given the input matrix and the kernel

1	0	1	1	0
0	0	0	1	1
1	0	0	0	1
0	1	1	1	0
1	1	0	1	0

Input Matrix

1	0	0
0	0	1
1	1	0

Kernel

i) Perform convolution with stride being 1 and 2.
ii) Apply Max-Pooling, Average-Pooling and Sum-Pooling to the results of the above convolutions.
iii) Visualize the flattened version of the pooled feature maps. 08 3 2

b Demonstrate the process of back propagation through convolutions with an example matrix calculation: (Considering : Input Image (3×3 matrix) : Input $= [[1,2,3]$, $[4,5,6]$, $[7,8,9]]$ Filter(Kernel) (2×2 Matrix) : Filter $= [[1,0], [0,-1]]$.
Compute the following:
i) Output feature map.
ii) Back propagation to update the weights: Assume the loss function L has been calculated and its gradient concerning the output is $\partial L / \partial$ Output $= [[1,1], [1,1]]$.
iii) Update the filter weights 08 3 2

OR

4 a Explain using *CNN* concepts, how object localization is effectively performed in an image, highlighting the key techniques, layers and processes involved in predicting both the object's class and its precise location within image. 08 2 2

b Consider the convolution neural network defined by the layers in the left column below. Fill in the shape of the output volume and number of parameters at each layer. You can write the shapes in the NumPy format (e.g., (128, 128, 3)).
Notation:
i) $CONV5 - N$ denotes a convolutional layer with N filters with height and width equal to 5. Padding is 2, and stride is 1.
ii) $POOL2$ denotes a 2×2 max- pooling layer with stride of 2 and 0 padding.
iii) $FC - N$ denoted a fully connected layer with N neurons.

Layer	*Activation Volume Dimensions*	*Number of parameters*
Input	32 * 32 * 1	0
CONV5 − 10		
POOL2		
CONV5 − 10		
POOL2		
FC 10		

08 3 2

5 a Analyze the different topologies of Recurrent Neural Networks (*RNNs*) and discuss how each topology is applied in real world scenarios. 08 4 3

	b	Examine the architecture of Gated Recurrent Unit (GRU) and Long Short- Term Memory ($LSTM$) models, derive the output equations for each layer and analyze the distinctions between the two.	08	4	2
		OR			
6	a	Compare the performance of basic $RNN, LSTMs$ and $GRUs$ in handling long-term dependencies in time-series forecasting. What are the strengths and weakness of each model?	08	2	2
	b	How does Backpropagation Through Time ($BPTT$) differ from regular backpropagation in feed-forward networks and what are the challenges faced when applying it to $RNNs$, particularly in terms of gradient explosion or vanishing gradients? Comparison in a tabular format.	08	2	2
7	a	Compare and contrast On-Policy and Off-Policy reinforcement learning methods, highlighting their differences with examples like $SARSA$ and Q-Learning.	08	2	2
	b	Discuss the impact of Deep Learning in training self- learning robots for the tasks like locomotion and visuomotor skills and how reinforcement learning enhances these tasks.	08	2	3
		OR			
8	a	Illustrate the working principle of Deep Q-Networks (DQN) in solving Atari games.	08	3	3
	b	Analyze the training process of Generative Adversarial Networks ($GANs$) and their role in generating image data.	08	4	2
9	a	With neat sketches and mathematical equations, explain the four stage process for creating the weight vectors to determine from where to read and write in Neural Turing machines. (Hint: Content-based addressing and location –based addressing)	08	2	1
	b	Critically evaluate the various training challenges of Generative Adversarial Networks ($GANs$) and suggest methods to stabilize GAN training.	08	4	1
		OR			
10	a	Design a strategy for applying conditional $GANs$ ($cGANs$) to generate image based on specific labels and discuss how this approach could be applied to image synthesis in fashion design.	08	4	4
	b	Describe the key parameters used to train Autoencoders and explain the differences between various types of Autoencoders.	08	2	1