USN | | | | | | | | | | |

Internal Assessment Test III –JULY 2024

Sub	Data Science and Its Applications					SubCode:	21AD62	Branch:	AIML	
Date	29.7.2024	Duration:	90 minutes	Max Marks:	50	SEC	VI-A		OBE	

	Scheme and Solutions	**MARKS**	**CO**	**RBT**
1a)	Explain feedforward neural network with a neat diagram. Also Discuss perceptron in detail and how it can be used to solve AND, OR NOT gates. **Answer: -** Feedforward network with diagram (4 Marks) • Simplest type of ANN • Information flows in one diagram • It consists of multiple layers of interconnected input layers, hidden layers and output layers. • No feedback loops Perceptron(3Marks) • Simplest form of a neural network • Single layer neural network • Input values,weights, bias, netsum, activation function+Explaination Solve AND, OR, NOT(3Marks) • Assign weights and bias and solve AND, OR and NOT	10	CO4	L2
2a)	Discuss the role of backpropagation algorithm in training neural networks. **Answer: -** Backpropagation algorithm (6 Marks) • Create a feedforward network with input units, hidden units and output units • Initialize weights with random values • Until termination condition repeat • Propagate input forward through the network • For each network unit and hidden unit calculate the error term. • Update the weights	6	CO4	L2
2b)	Explain the activation functions i) ReLU ii) tanh **Answer:-** ReLU (2Marks) The ReLU function is defined as: • f(x)=max(0,x) • Here's a breakdown of how it works: • If the input xxx is greater than 0, the function returns xxx. • If the input xxx is less than or equal to 0, the function returns 0. Tanh (2Marks) The tanh function is defined as: $\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$ It ouputs a value between -1 and 1.	4	CO4	L2

3a)	Write a python program to compute the Softmax and cross entropy in deep learning.
	Answer:-
	
	def softmax(logits):
	e_logits = np.exp(logits - np.max(logits))
	return e_logits / e_logits.sum(axis=0, keepdims=True)
	def cross_entropy_loss(predictions, labels):
	predictions (np.ndarray): Softmax probabilities for each class.
	labels (np.ndarray): True labels (one-hot encoded).
	loss = -np.sum(labels * np.log(predictions))
	return loss
		7	CO4	L3

3b)	Explain tensors in deep learning. **Answer:-** Tensor-3M **What is a Tensor?** A tensor is a multi-dimensional array that can hold data of various types (e.g., integers, floats). It can be thought of as an extension of vectors and matrices to more dimensions. • **0-D Tensor (Scalar)**: A single numerical value (e.g., 5). • **1-D Tensor (Vector)**: A one-dimensional array of values (e.g., [1, 2, 3, 4]). • **2-D Tensor (Matrix)**: A two-dimensional array of values (e.g., a table of numbers or a grid). • **3-D Tensor**: A three-dimensional array (e.g., a stack of matrices, like in a color image with width, height, and color channels). • **N-D Tensor**: Generalization to N dimensions (e.g., batches of images, sequences of data, etc.).	3	CO4	L2
4a)	Discuss n-gram models and word cloud. Write a python program to generate word cloud. **Answer:-** n-gram models(4Marks) An N-gram model predicts the probability of a word (or token) given the previous N−1N-1N−1 words in the sequence. It generalizes to various values of NNN: • **Unigram Model (N=1)**: Considers only the individual words. The probability of a word is independent of previous words. • **Bigram Model (N=2)**: Considers the previous word. The probability of a word depends on the immediately preceding word. • **Trigram Model (N=3)**: Considers the two previous words. The probability of a word depends on the preceding two words. • **N-Gram Model (N>3)**: Considers the N−1N-1N−1 preceding words. Word cloud (3Marks) A word cloud, also known as a tag cloud, is a visual representation of text data where the size of each word indicates its frequency or importance within the text.+Example Python code (3Marks)	10	CO5	L3
5a)	Explain user based collaborative filtering. **Answer:-** User-based collaborative filtering is a popular recommendation system technique used to suggest items (such as movies, products, or songs) based on the preferences of similar users. Steps • Data collection • Cosine Similarity • Recommendations	5	CO5	L2

5b)	Differentiate between ANN with RNN. **Answer:-** **Artificial Neural Networks (ANNs)**: • **Feedforward Structure**: In a standard ANN, information flows in one direction—from input to output—through hidden layers. Each layer consists of neurons (nodes) that are fully connected to the next layer. • **No Memory**: ANNs do not have a mechanism to maintain or remember information from previous inputs. They process each input independently of others. **Recurrent Neural Networks (RNNs)**: • **Feedback Connections**: RNNs have connections that loop back on themselves. This allows them to maintain a state and process sequences of inputs. They can maintain a form of memory by using hidden states that carry information from one time step to the next. • **Sequential Processing**: RNNs are designed to handle sequential data and learn patterns over time, making them suitable for tasks where the order of inputs matters, such as time series prediction or natural language processing.	5	CO5	L2

6a)	Explain the different measures of centrality in network analysis. Write a python code to implement page rank.
	Answer:-
	Explain each type -2Marks each
	• Degree Centrality
	• Betweenness Centrality
	• Eigen vector Centrality
	• Closeness centrality
	Page rank python Program -2Marks
	import networkx as nx
	import matplotlib.pyplot as plt
	
	# Create a directed graph
	G = nx.DiGraph()
	
	# Add nodes and edges to the graph
	edges = [
	('A', 'B'), ('A', 'C'), ('B', 'C'), ('C', 'A'),
	('C', 'D'), ('D', 'E'), ('E', 'D'), ('D', 'B')
	]
	G.add_edges_from(edges)
	
	# Compute PageRank
	pagerank = nx.pagerank(G, alpha=0.85)
	
	# Print PageRank values
	print("PageRank Values:")
	for node, rank in pagerank.items():
	print(f"Node {node}: {rank:.4f}")
		10	CO5	L3