DAYANANDA SAGAR UNIVERSITY

SCHOOL OF ENGINEERING

SCHEME & SYLLABUS FOR
BACHELOR OF TECHNOLOGY (B. Tech)

COMPUTER SCIENCE & ENGINEERING
(Artificial Intelligence & Machine Learning)
(3rd to 8th Semester)

Graduation Batch: 2025-2029

SCHEME - B.TECH – 2025-26 ONWARDS

I SEM - CHEMISTRY CYCLE

SL	COURSE CODE	COURSE TITLE	SCHEME OF TEACHING					Duration in Hrs Exam	Examination (Maximum Marks)		
			L	T	P	J	C		CIE	SEE	TM
1	25EN1101	LINEAR ALGEBRA AND DIFFERENTIAL EQUATIONS	3	0	0	0	3	3	60	40	100
2	25EN1102	OBJECT ORIENTED PROGRAMMING	2	1	2	0	4	3	60	40	100
3	25EN1103	COGNITIVE AND TECHNICAL SKILLS – I	0	0	4	0	2	1	100	-	100
4	25EN1104	ENGINEERING CHEMISTRY	2	0	2	0	3	3	60	40	100
5	25EN1105	INTRODUCTION TO MECHANICAL ENGINEERING	2	0	2	0	3	3	60	40	100
6	25EN1106	INTRODUCTION TO ELECTRICAL ENGINEERING	2	0	0	0	2	3	60	40	100
7	25EN1107	ENGINEERING MECHANICS	2	0	0	0	2	3	60	40	100
8	25EN1108	TECHNICAL ENGLISH	2	0	0	0	2	1	100	-	100
9	25EN1109	ENVIRONMENTAL SCIENCE	1	0	0	0	0	1	50	-	50
10	25EN1110	KANNADA KALI / MANASU	1	0	0	0	0	1	50	-	50
			17	1	10	0	21		660	240	900

L – Lecture, T – Tutorial, P – Practical, J – Project, C – No. of Credits, CIE – Continuous Internal Evaluation,
SEE- Semester End Examinations, TM – Total Marks.

SCHEME - B.TECH – 2025-26 ONWARDS

I SEM - PHYSICS CYCLE

SL	COURSE CODE	COURSE TITLE	SCHEME OF TEACHING					Duration in Hrs Exam	Examination (Maximum Marks)		
			L	T	P	J	C		CIE	SEE	TM
1	25EN1101	LINEAR ALGEBRA AND DIFFERENTIAL EQUATIONS	3	0	0	0	3	3	60	40	100
2	25EN1102	OBJECT ORIENTED PROGRAMMING	2	1	2	0	4	3	60	40	100
3	25EN1103	COGNITIVE AND TECHNICAL SKILLS – I	0	0	4	0	2	1	100	-	100
4	25EN1111	ENGINEERING PHYSICS	3	0	2	0	4	3	60	40	100
5	25EN1112	INTRODUCTION TO ELECTRONICS ENGINEERING	3	0	0	0	3	3	60	40	100
6	25EN1113	ENGINEERING GRAPHICS AND DESIGN THINKING	2	0	2	0	3	2	60	40	100
7	25EN1114	BIOLOGY FOR ENGINEERS	2	0	0	0	2	1	100	-	100
8	25EN1115	CONSTITUTION OF INDIA AND PROFESSIONAL ETHICS	1	0	0	0	0	1	50	-	50
			16	1	10	0	21		550	200	750

L – Lecture, T – Tutorial, P – Practical, J – Project, C – No. of Credits, CIE – Continuous Internal Evaluation,
SEE- Semester End Examinations, TM – Total Marks.

SCHEME - B.TECH – 2025-26 ONWARDS

II SEM - CHEMISTRY CYCLE

SL	COURSE CODE	COURSE TITLE	SCHEME OF TEACHING					Duration in Hrs. Exam	Examination (Maximum Marks)		
			L	T	P	J	C		CIE	SEE	TM
1	25EN1201	SINGLE AND MULTIVARIATE CALCULUS	3	0	0	0	3	3	60	40	100
2	25EN1202	C PROGRAMMING FOR PROBLEM SOLVING	2	1	2	0	4	3	60	40	100
3	25EN1203	COGNITIVE AND TECHNICAL SKILLS – II	0	0	4	0	2	1	100	-	100
4	25EN1204	ENGINEERING CHEMISTRY	2	0	2	0	3	3	60	40	100
5	25EN1205	INTRODUCTION TO MECHANICAL ENGINEERING	2	0	2	0	3	3	60	40	100
6	25EN1206	INTRODUCTION TO ELECTRICAL ENGINEERING	2	0	0	0	2	3	60	40	100
7	25EN1207	ENGINEERING MECHANICS	2	0	0	0	2	3	60	40	100
8	25EN1208	TECHNICAL ENGLISH	2	0	0	0	2	1	100	-	100
9	25EN1209	ENVIRONMENTAL SCIENCE	1	0	0	0	0	1	50	-	50
10	25EN1210	KANNADA KALI / MANASU	1	0	0	0	0	1	50	-	50
			17	1	10	0	21		660	240	900

L – Lecture, T – Tutorial, P – Practical, J – Project, C – No. of Credits, CIE – Continuous Internal Evaluation,
SEE- Semester End Examinations, TM – Total Marks.

SCHEME - B.TECH – 2025-26 ONWARDS

II SEM - PHYSICS CYCLE

SL	COURSE CODE	COURSE TITLE	SCHEME OF TEACHING					Duration in Hrs Exam	Examination (Maximum Marks)		
			L	T	P	J	C		CIE	SEE	TM
1	25EN1201	SINGLE AND MULTIVARIATECALCULUS	3	0	0	0	3	3	60	40	100
2	25EN1202	C PROGRAMMING FOR PROBLEM SOLVING	2	1	2	0	4	3	60	40	100
3	25EN1203	COGNITIVE AND TECHNICAL SKILLS – II	0	0	4	0	2	1	100	-	100
4	25EN1204	ENGINEERING PHYSICS	3	0	2	0	4	3	60	40	100
5	25EN1205	INTRODUCTION TO ELECTRONICS ENGINEERING	3	0	0	0	3	3	60	40	100
6	25EN1206	ENGINEERING GRAPHICS AND DESIGN THINKING	2	0	2	0	3	2	60	40	100
7	25EN1207	BIOLOGY FOR ENGINEERS	2	0	0	0	2	1	100	-	100
8	25EN1208	CONSTITUTION OF INDIA AND PROFESSIONAL ETHICS	1	0	0	0	0	1	50	-	50
			16	1	10	0	21		550	200	750

L – Lecture, T – Tutorial, P – Practical, J – Project, C – No. of Credits, CIE – Continuous Internal Evaluation,
SEE- Semester End Examinations, TM – Total Marks.

III SEM – CSE (ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

SL. NO	Course Type	Course Code	Course Name	Teaching Hours / Week				Examination			
				Lecture	Tutorial	Practical	Project	CIE Marks	SEE Marks	Total Marks	Credit
				L	T	P	J				
1	BSC	25AM2301	Probability & Statistics	3	0	0	0	60	40	100	03
2	IPCC	25AM2302	Data Structures	3	0	2	0	60	40	100	04
3	IPCC	25AM2303	Principles of Database Management	3	0	2	0	60	40	100	04
4	PCC	25AM2304	Artificial Intelligence	3	0	0	0	60	40	100	03
5	PCC	25AM2305	Computer Organisation and Architecture	3	0	0	0	60	40	100	03
6	PCC	25AM2306	AI For Embedded Systems	2	0	2	0	60	40	100	03
7	SEC	25AM2307	Skill Enhancement Course – I (Java Programming)	1	0	2	0	100	--	100	02
8	AEC	25AM2309	Cognitive and Technical Skills-III	--	--	--	--	--	--	--	Pass/Fail
			Total	20	0	06	0				**22**

IPCC-Integrated Professional Core Course, PCC-Professional Core Courses, SEC-Skill Enhancement Courses, BSC-Basic Science, L – Lecture, T – Tutorial, P – Practical, J –Project, C – No. Of Credits, AEC-Ability Enhancement Course

PROBABILITY AND STATISTICS

[As per Choice Based Credit System (CBCS)scheme]

SEMESTER – III

Course Code	:	25AM2301	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–J	:	3–0–0–0			

Course Learning Objectives:

This Course will enable students to:

1. **Apply** statistical principles and probability concepts to solve complex problems in real-world scenarios involving uncertainty and randomness.
2. **Evaluate** and select appropriate probability distributions and statistical techniques to analyze and interpret data accurately in various applications.
3. **Justify** the use of estimation methods and hypothesis testing techniques for drawing meaningful inferences about population parameters.
4. **Analyze** and interpret sample test results for different statistical relationships, such as means, variances, correlation coefficients, regression coefficients, goodness of fit, and independence, to make informed decisions.
5. **Identify** sample tests using appropriate statistical procedures to investigate the significance of observed data and communicate findings effectively.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I : Probability	**09 Hours**

Definitions of Probability, Addition Theorem, Conditional Probability, Multiplication Theorem, Bayes' Theorem of Probability

UNIT – II: Random Variables and their Properties and Probability Distributions	**09 Hours**
Discrete Random Variable, Continuous Random Variable, Joint Probability Distributions Their Properties, Probability Distributions: Discrete Distributions: Binomial, Poisson Distributions and their Properties; Continuous Distributions: Exponential, Normal, Distributions and their Properties.	
UNIT – III: Estimation and testing of hypothesis	**06 Hours**
Sample, Populations, Statistic, Parameter, Sampling Distribution, Standard Error, Un-Biasedness, Efficiency, Maximum Likelihood Estimator, Notion & Interval Estimation.	
UNIT – IV: Sample Tests-1	**07 Hours**
Large Sample Tests Based on Normal Distribution, Small Sample Tests : Testing Equality of Means, Testing Equality of Variances, Test of Correlation Coefficient	
UNIT – V: Sample Tests-2	**08 Hours**
Test for Regression Coefficient; Coefficient of Association, 2 – Test for Goodness of Fit, Test for Independence.	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	**Apply** the principles of probability to solve complex problemsin various real-world scenarios.	L2 & L3
2	**Solve** and **compare** different probability distributions, including discrete and continuous random variables, in order to make informed decisions and predictions.	L2 & L3
3	**Apply** statistical estimation techniques, such as maximum likelihood estimation and interval estimation, to draw meaningful inferences about population parameters from sample data.	L3
4	**Examine** hypothesis testing methods, including large and small sample tests, to assess the significance of observed dataand draw valid conclusions.	L4
5	**Analyze** statistical relationships and perform sample tests toassess the Equality of means in different populations, Correlation coefficients between variables to determine the strength and direction of the relationship. Independence of variables using appropriate statistical tests to assess theabsence of any relationship.	L4

Table: Mapping Levels of COs to POs / PSOs														
COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
CO1	3	2	2		2				1					
CO2	3	2	2		2				1					
CO3	3	2	2						1					
CO4	3	2	2		2				1					
CO5	3	2	2		2				1					

3: Substantial (High)2: Moderate (Medium) 1: Poor (Low)

TEXT BOOKS:

1. Probability & Statistics for Engineers and Scientists, Walpole, Myers, Myers, Ye. PearsonEducation.

REFERENCE BOOKS:

1. Probability, Statistics and Random Processes T. Veerarajan Tata McGraw – Hill
2. Probability & Statistics with Reliability, Queuing and Computer Applications, Kishor S. Trivedi,Prentice Hall of India ,1999

E-Resources:

1. https://nptel.ac.in/courses/106104233
2. https://nptel.ac.in/courses/117103067
3. https://nptel.ac.in/courses/103106120
4. https://www.coursera.org/learn/probability-intro#syllabus
5. https://nptel.ac.in/courses/111104073

Activity Based Learning
(Suggested Activities in Class)

1. Tools like Python programming, R programming can be used which helps student to developa skill to analyze the problem and providing solution.
2. Regular Chapter wise assignments/ Activity/Case studies can help students to have criticalthinking, developing an expert mind set, problem-solving and teamwork.

Following are Assignments/ Activities Can be carried out using either R programming language orPython Programming or excel solver.

1. There are n people gathered in a room. What is the probability that at least 2 of them will havethe same birthday? (Use excel solver, R Programming, Python Programming)
   a. Use simulation to estimate this for various n., and Produce Simulation Graph.
   b. Find the smallest value of n for which the probability of a match is greater than 0.5.

c. Explore how the number of trials in the simulation affects the variability of our estimates.

2. **Case Study 1: Customer Arrivals at a Coffee Shop**
    a. A coffee shop wants to analyze the number of customer arrivals during its morning rushhour (7:00 AM to 9:00 AM). The shop has been recording the number of customer arrivals every 15 minutes for the past month.
    b. Data: The data consists of the number of customer arrivals recorded at the coffee shop during each 15-minute interval for the past month.
    c. Here is a sample of the data:

Time Interval	**Customer Arrivals**
00 AM - 7:15 AM	**6**
15 AM - 7:30 AM	**4**
30 AM - 7:45 AM	**9**
45 AM - 8:00 AM	**7**
00 AM - 8:15 AM	**5**
15 AM - 8:30 AM	**8**
30 AM - 8:45 AM	**10**
45 AM - 9:00 AM	**6**

analyze the customer arrivals and determine the probability distribution that best fits thedata. Specifically, explore both discrete and continuous probability distributions,

including the binomial, Poisson, exponential, and normal distributions.

3. **Case Study 2:** Comparing the Performance of Two Groups
    a. Suppose you are a data analyst working for a company that manufactures a new energydrink. The marketing team conducted a promotional campaign in two different cities (City A and City B) to determine the effectiveness of the campaign in increasing sales. The sales data for a random sample of customers in each city was collected over a week.Your task is to compare the average sales between the two cities and test whether there is a significant difference in the variance of sales.
    b. **Data**: Let's assume the following sample data for the number of energy drinks sold in each city:

City A: [30, 28, 32, 29, 31, 33, 34, 28, 30, 32]
City B: [25, 24, 26, 23, 22, 27, 29, 30, 26, 24]

perform a two-sample t-test to test the equality of means and a test for equality of variances using Python's SciPy library.

4. **Case study 3:** testing independence between two categorical variables.
    a. Data: Sample of 100 employees, and each employee is classified as either Male or Female.They were asked to rate their job satisfaction on a scale of 1 to 5, where 1 represents low satisfaction and 5

represents high satisfaction. The data is as follows:

Employee	Gender	Job Satisfaction
1	Male	4
2	Female	3
3	Male	2
4	Female	5
...	...	...
100	Female	4

b. Test for independence between gender and job satisfaction, use the chi-squared test in R.

****

DATA STRUCTURES

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Course Code	:	25AM2302	**Credits**	:	04
Hours / Week	:	05 Hours	**Total Hours**	:	39(Th)+26(P) Hours
L–T–P-J	:	3–0–2-0			

Prerequisites:

Proficiency in a C programming language.

Course Objectives:

This Course will enable students to:

1. **Understand** the basic approaches for analyzing and designing data structures.
2. **Introduce** dynamic memory allocation and C language concepts required for building data structures.
3. **Develop** essential skills to construct data structures to store and retrieve data quickly and **efficiently**.
4. **Utilize** different data structures that support different sets of operations which are suitable for various applications.
5. **Explore & implement** how to insert, delete, search, and modify data in any data structure- Stack, Queues, Lists, Trees.
6. **Develop** applications using the available data structure as part of the course for mini project.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture methods, but different *types of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note taking, annotating, and roleplaying.
3. Show ***Video/Animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.

8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**
IINTRODUCTION: Introduction to Data Structure, Classification, C Structure and Union, C Pointers, Array Definition, Representation, Operations (Insertion, Deletion, Search and Traversal), Two/Multidimensional Arrays, Sparse matrix. TB1: 1.1, 2.2, 2.5 ; TB2: 1.1, 1.2,1.3.1-1.3.4; RB1: 5.1 – 5.12, 6.4	
UNIT – II	**08 Hours**
INTRODUCTION TO ADT: **Stack:** Definition, Array Representation of Stack, Operations on Stacks. **Applications of Stack:** Expression evaluation, Conversion of Infix to Postfix, Infix to Prefix Recursion, Tower of Hanoi **Queue:** Definition, Representation of Queues, Operations of Queues, Circular Queue. **Applications of Queue:** Job Scheduling, A Maze Problem TB1: 3.1,3.2, 3.3,3.4,3.5 ; TB2: 2.1, 2.2, 2.3, 3.2, 3.3	
UNIT – III	**08 Hours**
DYNAMIC DATA STRUCTURES: **Linked List:** Types, Representation of Linked Lists in Memory. Traversing, Searching, Insertion & Deletion from Linked List. Circular List, Doubly Linked List, Operations on Doubly Linked List (Insertion, Deletion, Traversal). **Applications:** Stack & Queue Implementation using Linked Lists. **Case Study**: Josephus problem. TB2: 4.2,4.3,4.5	
UNIT – IV	**08 Hours**
TREES: Basic Terminology, Binary Trees and their representation, Complete Binary Trees, Binary Search Trees, Threaded Binary Trees, Operations on Binary Trees (Insertion, Deletion, Search & Traversal). **Applications:** Expression Evaluation **Case Study:** Game Tree TB2: 5.5.3,5.5.4,5.6	
UNIT – V	**07 Hours**
Efficient Binary Search Trees: Optimal Binary Search Trees, AVL Trees, Red Black Trees, Splay Trees. **Case Study**: B Trees TB1: 10.1,10.2,10.3,10.4, 11.2	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Apply the concepts of pointers, arrays, structures, and unions to address real-world problems and implement the concept in C programming language.	L3
2	Utilize stacks and queue data structures to solve problems such as infix to postfix, infix to prefix conversions, the Towers of Hanoi puzzle, job scheduling and maze navigation.	L3
3	Implement and manipulate singly linked lists, doubly linked lists, and circular linked lists, executing operations such as insertion, deletion, and traversal.	L3
4	Understand the concepts of binary trees, binary search trees, and threaded binary trees, and their associated operations.	L2
5	Understand advanced binary tree structures includes optimal binary search trees, AVL trees, Red-Black trees, and Splay trees.	L2

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	1	2	3	-	3	-	-	-	-	-	-	-	2	2
CO2	1	2	3	-	3	-	-	-	-	-	-	-	2	2
CO3	1	2	3	-	3	-	-	-	-	-	-	-	2	2
CO4	-	1	2	-	3	-	-	-	-	-	-	-	2	2
CO5	-	1	2	-	-	-	-	-	-	-	-	-	2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS (TB):

1. Ellis Horowitz, Susan Anderson-Freed, and Sartaj Sahni, "Fundamentals of Data structures in C", 2nd Edition, Orient Longman, 2008.
2. A.M. Tannenbaum, Y Langsam, M J Augentien "Data Structures using C", 1st Edition, Pearson, 2019.

REFERENCE BOOKS:

1. Brian. W. Kernighan, Dennis. M. Ritchie, "The C Programming Language", 2nd Edition, Prentice-Hall, 1988.
2. Gilbert & Forouzan, "Data Structures: A Pseudo-code approach with C", 2nd Edition, Cengage Learning,2014.
3. Jean-Paul Tremblay & Paul G. Sorenson, "An Introduction to Data Structures with Applications", 2nd Edition, McGraw Hill, 2013.
4. R.L. Kruse, B.P. Learly, C.L. Tondo, "Data Structure and Program design in C", 5th Edition, PHI ,2009.

E-Resources:

1. https://nptel.ac.in/courses/106102064
2. https://www.coursera.org/learn/data-structures?specialization=data-structures-algorithms
3. https://www.udemy.com/topic/data-structures/free/
4. https://www.mygreatlearning.com/academy/learn-for-free/courses/data-structures
5. https://cse01-iiith.vlabs.ac.in/
6. https://kremlin.cc/k&r.pdf

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving using group discussion.
2. Role play E.g., Stack, Queue, etc.,
3. Demonstration of solution to a problem through programming.
4. Flip class activity E.g., arrays, pointers, dynamic memory allocation, etc.,

DATA STRUCTURES LABORATORY

Total Contact Hours: 26

Following are experiments to be carried out using either C programming language.

1. To Implement C programs to perform array operations.
2. To determine the validity of a 9x9 Sudoku board (application of 2-dimensional array).
3. To store, retrieve and update the elements in structures (structures and pointers to structures).
4. To implement stack using linked list.
5. To implement a queue data structure using a singly linked list.
6. To implement a singly linked list and its operations.
7. To implement a doubly linked list and its operations.
8. To create a circular queue using a circular linked list data structure
9. To implement binary tree traversal techniques.

OPEN-ENDED EXPERIMENTS

1. Design a web browser history tracker in C. Implement a stack data structure to keep track of visited URLs. Create functions to push new URLs onto the stack as users visit websites and pop URLs when users navigate backward in their browsing history.
2. Imagine you are responsible for designing a queue-based system to manage the queue of regular customers waiting to purchase cinema tickets at a popular movie theatre. Your system should ensure fair and efficient ticket sales for all customers. When a customer's arrive at the cinema, they join the queue. Each customer is represented by his name, age (for record-keeping), and number of tickets needed. When a customer reaches the front of the queue, they are served by the ticketing agent. Implement a ticket sale process where the agent provides the customer with the requested ticket(s). Initialize the total number of tickets and if the tickets are sold, then the ticketing agent should display a houseful message.

PRINCIPLES OF DATABASE MANAGEMENT

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Course Code	: 25AM2303	**Credits**	: 04
Hours / Week	: 05 Hours	**Total Hours**	: 39(Th)+26(P) Hours
L–T–P–J	: 3–0–2–0		

Course Learning Objectives:

This course will enable students to:

1. **Acquire** the concept of databases, Entity-Relationship Model and relational model for creating and designing databases for the real-world scenario.
2. **Develop** queries to extract data from the databases using a structured query language.
3. **Differentiate** SQL and NoSQL.
4. **Demonstrate** the operations on MongoDB, Database connectivity with front end and **Optimize** the Database design using Normalization Concepts.
5. **Understand** the importance of Transaction Management, Concurrency control mechanism and recovery techniques.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the student's understanding.

UNIT – I	**10 Hours**

INTRODUCTION TO DATABASE SYSTEMS :

Introduction, Characteristics of the Database Approach, Advantages of using DBMS Approach, Data Models, Schemas, Instances and Data Independence, Three Schema Architecture, various components of a DBMS.

(Text Book-1: Chapter 1: 1.1 to 1.4, 1.6, Chapter 2: 2.1,2.2, 2.4)

ENTITY-RELATIONSHIP MODEL:

Entity Types , Entity Sets , Attributes and Keys, Relationship types, Relationship Sets , Roles

and Structural Constraints; Weak Entity Types; ER Diagrams ***(Text Book-1: Chapter 7: 7.3, 7.4, 7.5, 7.7)***.	
UNIT – II	**07 Hours**
RELATIONAL MODEL : Relational Model Concepts, Relational Model Constraints and Relational Database Schemas, Update operations and Dealing with Constraint Violations. ***(Text Book-1: Chapter 3: 3.1 to 3.3).*** **SQL –THE RELATIONAL DATABASE STANDARD:** SQL Data Definition and Data types, Specifying constraints in SQL, Basic Queries in SQL-Data Definition Language in SQL, Data Manipulation Language in SQL; ***(Text Book-1: Chapter 4: 4.1 to 4.4).***	
UNIT – III	**08 Hours**
SQL –THE RELATIONAL DATABASE STANDARD: Additional Features of SQL; Views (Virtual Tables) in SQL; Database Programming Issues and Techniques ; ***(Text Book-1: Chapter 4: 4.5; Chapter 5: 5.1 to 5.4).*** **SQL AND NOSQL DATA MANAGEMENT:** Triggers, Database connectivity using Python, SQL vs NoSQL, Introduction to MongoDB, ***(Text Book-1: Chapter 5: 5.2,5.3)(Text Book-2 Chapter 1: 1.1 to 1.5***)	
UNIT – IV	**07 Hours**
NOSQL DATA MANAGEMENT: Data Types, Data Modelling, CRUD Operations. ***(Text Book-2 Chapter 1: 1.1 to 1.5***) **DATABASE DESIGN:** Design Guidelines, Functional Dependencies; Normal Forms Based on Primary Keys; General Definitions of Second and Third Normal Forms; Boyce-Codd Normal Form; ***(Text Book-1: Chapter 14: 14.1 to 14.5)***	
UNIT – V	**07 Hours**
TRANSACTION MANAGEMENT The ACID Properties; Transactions and Schedules; Concurrent Execution of Transactions; Concurrency Control Mechanisms; Error recovery methods. ***(Text Book-1: Chapter 20: 20.1 to 20.5, Chapter 21: 21.1 to 21.3, Chapter 22: 22.1 to 22.4***)	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	**Use** the basic concepts of database management system in the design and creating database blueprint using E-R model and relational model.	L3
2	**Formulate** SQL and NoSQL queries for building structure and unstructured databases	L3

3	**Demonstrate** database connectivity using vendor specific drivers	L3
4	**Apply** normalization techniques to design relational database management system	L3
5	**Adapt** Transaction Management, concurrency control and recovery management techniques in database management system.	L3

Mapping Levels of COs to POs / PSOs														
Cos				**Program Outcomes (POs)**									**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
	Engineering knowledge	Problem analysis	Design	Conduct investigations of complex problems	tool usage	The engineer and society	Environment and sustainability	Ethics	team work	Communication	Project management and finance	Life-long learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	3	2	-	-	-	-	-	2	2	-	-	2	3
CO2	3	2	1	-	3	-	-	-	2	2	-	-	2	3
CO3	2	2	2	-	3	-	-	-	2	2	-	-	2	3
CO4	3	1	2	-	1	-	-	-	2	2	-	-	2	3
CO5	2	1	-	-	-	-	-	-	2	2	-	-	2	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Elmasri and Navathe, "Fundamentals of Database Systems", Seventh Edition, Pearson Education, 2021.
2. P. J. Sadalage and M. Fowler, "NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence", First Edition, Pearson Education, Inc. 2012.

REFERENCE BOOKS:

1. Raghu Ramakrishnan and Johannes Gehrke, "Database Management Systems", Third Edition, McGraw-Hill, 2003.
2. Silberschatz, Korth and Sudharshan: "Database System Concepts", Seventh Edition, Mc-GrawHill, 2019.

3. C.J. Date, A. Kannan, S. Swamynatham: "An Introduction to Database Systems", Eight Edition, Pearson Education, 2012.

E-Resources:

1. http://nptel.ac.in/courses/106106093/
2. https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-830-database-systems-fall-2010/lecture-notes/
3. http://agce.sets.edu.in/cse/ebook/DBMS%20BY%20RAGHU%20RAMAKRISHNAN.pdf
4. http://iips.icci.edu.iq/images/exam/databases-ramaz.pdf
5. https://db-class.org/
6. https://www.w3schools.com/mongodb/

Activity Based Learning (Suggested Activities in Class)

1. Database designing and data extraction using group discussion.
2. Collaborative Activity is minor project development with a team of 4 students.

LABORATORY EXPERIMENTS

Total Contact Hours: 26 Following are experiments to be carried out using either oracle or mysql, Mongo Db .

1. Design any database with at least 3 entities and establish proper relationships between them. Draw suitable ER/EER diagrams for the system. Apply DCL and DDL commands.
2. Design and implement a database and apply at least 10 Different DML Queries for the following task.
   a. For a given input string display only those records which match the given pattern or a phrase in the search string. Make use of wild characters and like operators for the same. Make use of Boolean and arithmetic operators wherever necessary
3. Write SQL statements to join table and retrieve the combined information from tables.
4. Execute the Aggregate functions count, sum, avg, min, max on a suitable database. Make use of built in functions according to the need of the database chosen .
5. Retrieve the data from the database based on time and date functions like now(), date(), day(), time() etc., Use of group by and having clauses.
6. Write and execute database trigger. Consider row level and statement level triggers.
7. Write and execute program to perform operations on MongoDb Database.
8. Write and execute program to perform CRUD operations.

Open Ended Experiments

1. Consider the Table "employees", write a SQL query to remove all the duplicate emails of employees keeping the unique email with the lowest employee id,return employee id and unique emails .

table: employees

Employee id	Employee_name	Email id
101	Liam Alton	li.al@abc.com
102	Josh Day	jo.da@abc.com
103	Sean Mann	se.ma@abc.com

104	Evan Blake	ev.bl@abc.com
105	Toby Scott	jo.da@abc.com

2. A salesperson is a person whose job is to sell products or services. Consider the table "Sales" [given below]. Write a SQL query to find the top 10 salesperson that have made highest sale. Return their names and total sale amount.

Table: sales

TRANSACTION_ID|SALESMAN_ID|SALE_AMOUNT|

501	18	5200.00
502	50	5566.00
503	38	8400.00
599	24	16745.00
600	12	14900.00

Table: salesman

SALESMAN_ID	SALESMAN_NAME
11	Jonathan Goodwin
12	Adam Hughes
13	Mark Davenport
59	Cleveland Hart
60	Marion Gregory

ARTIFICIAL INTELLIGENCE

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Course Code	: 25AM2304	**Credits**	: 03
Hours / Week	: 03 Hours	**Total Hours**	: 39 Hours
L–T–P–S	: 3–0–0–0		

Course Learning Objectives:

This course will enable students to:

1. To provide a strong foundation of fundamental concepts in artificial intelligence.
2. To provide a basic exposition to different types of searching in Artificial intelligence.
3. To provide different knowledge representation, reasoning, and learning techniques.
4. To distinguish the different types of Experts Systems.
5. To design the Expert System based on the concepts of knowledge representation, searching and Reasoning on various applications using modernised AI tools.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the student's understanding.

UNIT – I	**08 Hours**

Introduction: Artificial Intelligence, AI Problems, AI Techniques: supervised learning, unsupervised learning, and reinforcement learning, Different Types of Agents: Simple reflex agents, Model-based reflex agents, Goal-based agents and Utility-based agents; Environment, Problem Space and Search, Defining the Problem as a State Space Search, Problem Characteristics.

Textbook 2: Chapter 1,2

UNIT – II	**08 Hours**

Basic Search Techniques: Solving problems by searching; issues in the design of search programs; uniform search strategies: Breadth first search, Depth first search, Depth limited search, Bidirectional search, Best First search. Textbook 3: Chapter 3, 4; Textbook 2: Chapter 2, 3
UNIT – III **08 Hours**
Special Search Techniques: Heuristic Search, greedy best first search, A* search, AO*Algorithm, Hill climbing search, Simulated Annealing search, Adversarial search, Minimax search, Alpha, beta pruning, Genetic Algorithm. Knowledge Representation: Procedural Vs Declarative Knowledge, Approaches to Knowledge Representation, Forward Vs Backward Reasoning. Textbook 1: Chapter 5; Textbook 2: Chapter 3, 6
UNIT – IV **07 Hours**
Symbolic Logic: Propositional Logic, First Order Predicate Logic: Representing Instance and isa Relationships. Reasoning: Introduction to Monotonic Reasoning and Non-Monotonic Reasoning. Statistical Reasoning: Bayes Theorem, Certainty Factors, Bayesian Networks, Dempster-Shafer Theory. Textbook 2: Chapter 7,8
UNIT – V **08 Hours**
Experts Systems: Overview of an Expert System, Structure of an Expert Systems, Different Types of Expert Systems: Rule Based, Model Based, Case Based and Hybrid Expert Systems, Knowledge Building System Tools, Expert System Shells and Fuzzy Expert systems. Textbook 1: Chapter 8,9; Textbook 2: Chapter 20,22

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	**Summarize** the basic concepts of Artificial Intelligence, AI principles, AI Task domains and applications.	L2
2	**Apply** the different types of searching and knowledge representations techniques to solve the AI problems.	L3
3	**Make use of** the different types of reasoning such as Monotonic Reasoning, Non-Monotonic Reasoning and Statistical Reasoning techniques for making analysis and predictions.	L4

4	**Differentiate** the various types of Experts Systems such as Rule Based, Model Based, Case Based, Hybrid and Fuzzy Expert Systems.	L4
5	**Design** the Expert System based on the concepts of knowledge representation, searching and Reasoning on various applications using modernised AI tools.	L5

Mapping Levels of COs to POs / PSOs														
COs				**Program Outcomes (POs)**									**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
	Engineering knowledge	Problem analysis	Design	Conduct investigations of complex problems	tool usage	The engineer and society	Environment and sustainability	Ethics	team work	Communication	Project management and finance	Life-long learning	Cognitive Outcome	Skill & Design Outcome
CO1	2				2	1			1	1			2	2
CO2	3	2			2	1			1	1			2	2
CO3	3	2			2	1			1	1			2	2
CO4	3	2			2	1			1	1			2	2
CO5	3	3	2		2	1			1	1			2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Artificial Intelligence, George F Luger, Sixth Edition, Pearson Education Publications, 2014.
2. Artificial Intelligence, Elaine Rich and Knight, McGraw-Hill Publications, 2010.

REFERENCE BOOKS:

1. Artificial Intelligence: A modern Approach, Russell and Norvig, Printice Hall, 2010.
2. Introduction To Artificial Intelligence & Expert Systems, Patterson, PHI.
3. Multi Agent systems- a modern approach to Distributed Artificial intelligence, Weiss. G, MIT Press.
4. Artificial Intelligence: A modern Approach, Russell and Norvig, Printice Hall.

E-Resources:

1. https://onlinecourses.nptel.ac.in/noc22_cs56/preview
2. https://onlinecourses.nptel.ac.in/noc23_ge40/preview

Activity Based Learning (Suggested Activities in Class)

1. Flipped Class Activity on Searching techniques.
2. Problem Solving and Discussion.
3. Role Play

COMPUTER ORGANIZATION AND ARCHITECTURE

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Course Code	: 25AM2305	**Credits**	: 03
Hours / Week	: 03 Hours	**Total Hours**	: 39 Hours
L–T–P-J	: 3–0–0-0		

Course Learning Objectives:

This Course will enable students to:

1. **Understand** the Architecture and programming of ARM microprocessor.
2. **Develop** program using Arm instruction set and appreciate the advanced features provided in the ARM
3. **Understand** the exception handling techniques.
4. **Study in** detail the concept of instruction level parallelism and concepts of pipelining.
5. **Understand** various cache memory mapping techniques and memory Organization.

Teaching-Learning Process

1. ***Lecture method*** along with traditional lecture method, different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *incorporating* brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Showing ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, asking Higher order Thinking questions in the class in the form of Quiz and writing programs with complex solutions.
6. Showing the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.

UNIT – I	**05 Hours**

An Overview of Computing Systems: History of Computers, The Computing Device,
The ARM7TDMI Programmers' Model:
Introduction, Data types, Processor Modes, Registers, Program Status Registers, The vector Table.
Assembler Rules and Directives: Structure of Assembly Language Modules, Registers, Directives and Macros. **Loads, Stores and Addressing:** LODS and STORES instructions, Operand Addressing , ENDIANNES ***Text Book-1: 1.1 to 1.3; 2.1 to 2.6 ; 4; 5.3, 5.4, 5.5***

UNIT – II	**05 Hours**

Constants and Literal Pools: The ARM Rotation Scheme, Loading Constants and address into Registers **Logic and Arithmetic:** Flags and their Use, Compare instructions, Data Processing Instructions **Loops and Branches:** Branching, Looping, Conditional Execution, Straight-Line Coding **Subroutines and Stacks:** Stack, Subroutines, Passing parameters to subroutines, The ARM APCS. ***(Text Book-1: 6.1 to 6.4; 7.1 to 7.4; 8.2 to 8.6; 10.1 to 10.5***
UNIT – III **05 Hours**
Mixing C and Assembly Language: Inline Assembler Embedded Assembler, Calling Between C and Assembly. **Exception Handling:** Interrupts, Error Conditions, Processor Exception Sequence, The Vector Table, Exception Handlers, Exception Priorities, Procedures for Handling Exceptions. ***(Text Book-1: 11.1 to 11.8; 14.1 to 14.4***
UNIT – IV **12 Hours**
Pipelining: Basic and Intermediate Concepts Introduction, The Major Hurdle of Pipelining, How Pipelining Implemented, What makes Pipelining hard to Implement, Extending the MIPS Pipeline to Handle Multicycle Operations, The MIPS R4000 Pipeline, Crosscutting Issues. ***Text Book-2: C.1 to C.7***
UNIT – V **12 Hours**
Memory Hierarchy: *Introduction, Cache Performance, Six basic cache Optimizations, Virtual Memory, Protection and examples of Virtual Memory, Fallacies and Pitfalls.* ***Text Book-2: B.1 to B.6***

4. Mini Project

****

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	**Apply** knowledge of the internal architecture and organization of ARM microprocessors to utilize their components and functionalities.	L3
2	**Apply** the instruction set of ARM Microprocessor by writing Assembly language programs.	L3
3	**Analyze and compare** the various exception handling techniques.	L4
4	**Examine** the concept of instruction-level parallelism and **analyze** the principles of Pipelining techniques.	L4
5	**Compare** and C**ontrast** memory hierarchy and its impact on computer cost/performance.	L4

Mapping Levels of COs to POs / PSOs

Cos				**Program Outcomes (POs)**									**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
	Engineering knowledge	Problem analysis	Design	Conduct investigations of complex problems	tool usage	The engineer and society	Environment and sustainability	Ethics	team work	Communication	Project management and finance	Life-long learning	Cognitive Outcome	Skill & Design Outcome
CO1	3		2										2	
CO2	3		3		1								2	
CO3	3	3	1										2	
CO4	3	3	1										2	
CO5	3	3	1										2	

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. William Hohl, "ARM Assembly Language", 2nd Edition, CRC Press, 2009.

2. John L Hennessy, David A Patterson, “Computer Architecture, A Quantitative Approach”, 5th Edition, Morgan Kaufmann publishers, 2012.

REFERENCE BOOKS:

1. David A Patterson, John L Hennessy, “Computer Organization and Design”, 4th Edition, Morgan Kaufmann publishers, 2010.
2. Steve Furber, “ARM System-on-chip Architecture”, 2nd Edition, Pearson Publications, 2000.
3. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, “Computer Organization”, 5th Edition, Tata McGraw Hill, 2002.

E-Resources:

1. https://www.udemy.com/topic/arm-cortex-m/
2. https://www.edx.org/school/armeducation
3. https://onlinecourses.nptel.ac.in/noc22_cs93/preview

Activity Based Learning (Suggested Activities in Class)

1. Mini project implementation using Assembly Language Programming.
2. Demonstration of solution to a problem through programming.

AI FOR EMBEDDED SYSTEMS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Subject Code	: 25AM2306	**Credits**	: 03
Hours / Week	: 03 Hours	**Total**	: 30 +15 Hours
L–T–P–J	: 2–0–2–0	**CIA + SEE**	: 60+40 Marks

Course Vision

To develop advanced knowledge in embedded system architecture, real-time software development, and intelligent edge systems enabling students to design optimized hardware–software integrated systems for modern AI-enabled applications.

Course Objectives:

This course will enable students to:

1. Understand advanced embedded system architecture and processor design.
2. Develop embedded firmware using advanced C and real-time techniques.
3. Apply communication protocols and hardware interfaces in embedded systems.
4. Design real-time embedded systems using RTOS concepts.
5. Integrate embedded systems with IoT and edge AI applications.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

a. **Lecture method** along with traditional lecture method, different type of teaching methods may be adopted to develop the course outcomes.

b. **Interactive Teaching:** incorporating brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.

c. Showing **Video/animation** films to explain functioning of various concepts.

d. Encourage **Collaborative** (Group Learning) Learning in the class.

e. To make **Critical thinking**, asking Higher order Thinking questions in the class in the form of Quiz and writing programs with complex solutions.

f. Showing the **different ways to solve** the same problem and encourage the students to come up with their own creative ways to solve them

MODULE – I	**06 Hours**

Advanced Embedded System Architecture

Topics: Fundamentals review: embedded systems vs general computing systems, Advanced embedded architectures, ARM Cortex-M architecture deep dive, Processor pipeline and instruction execution. **Applications:** Design and optimization of embedded architectures for high-performance systems, development of efficient ARM Cortex-M based applications, improving execution speed using pipelining techniques, real-time data processing in control systems, and architecture-level optimization for power and performance in embedded devices.	
MODULE – II	**06 Hours**
Embedded Hardware Design & Interfaces **Topics:** Memory hierarchy in embedded systems, Interrupt handling and exception management, Low-power embedded design techniques, Embedded processors and microcontrollers, Memory subsystems (Flash, EEPROM, SRAM), Peripheral interfaces, Sensor and actuator interfacing. **Applications:** Design and integration of memory subsystems in embedded devices, efficient interrupt handling for responsive systems, development of low-power hardware solutions, interfacing sensors and actuators for real-time applications, and building reliable embedded systems with optimized peripheral communication.	
MODULE – III	**06 Hours**
Embedded Software Design **Topics:** Communication Interfaces: I2C, SPI, UART, CAN, USB. Advanced communication systems: Ethernet for embedded systems, Wireless embedded interfaces, Industrial communication protocols Embedded hardware debugging tools: Logic analyzers, Oscilloscopes. **Applications:** Development of communication-driven embedded applications using I2C, SPI, UART, and CAN protocols, implementation of network-enabled embedded systems using Ethernet and wireless technologies, design of industrial communication solutions, and effective debugging and validation using hardware diagnostic tools.	
MODULE – IV	**06 Hours**

Real-Time Operating Systems (RTOS) **Topics:** Embedded programming using C, Compiler architecture and build stages, Memory management in embedded systems, Bitwise operations and register programming, Interrupt driven programming. Embedded software optimization techniques: Code size optimization, Power-aware programming, Real-time scheduling, Software design patterns in embedded systems. **Applications:** Development of real-time embedded applications using C, efficient memory and resource management in constrained systems, implementation of interrupt-driven and time-critical applications, optimization of embedded software for performance and power efficiency, and design of reliable systems using real-time scheduling and software design patterns.	
MODULE – V	**06 Hours**
Embedded IoT & Edge Intelligence **Topics:** Embedded systems in IoT architecture, Edge AI concepts: TinyML, Embedded cybersecurity: Secure boot, Device authentication, Firmware security. **Applications:** Design and deployment of IoT-enabled embedded systems, implementation of edge intelligence using TinyML techniques, development of secure embedded systems with device authentication and secure boot, enabling real-time data processing at the edge, and building scalable smart applications for connected environments.	

Course Outcome

Course Outcomes	Description	Bloom' s Taxonomy Level
At the end of the course, the student will be able to:		
1	**Understand** the key features of advanced embedded system architectures and processor designs.	L2
2	**Understand** how hardware interfaces and communication modules function within embedded systems.	L2
3	**Analyze** embedded firmware to identify optimization opportunities using advanced programming techniques.	L3
4	**Evaluate** real-time operating system concepts for suitability in different embedded applications.	L5
5	**Examine** intelligent embedded systems integrating IoT and edge AI solutions.	L4

Mapping Levels of COs to POs / PSOs

Table: Mapping Levels of COs to POs / PSOs													
CO's	**PO'S**											**PSO'S**	
COs / PSOs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO-1	3	2	1	-	2	-	-	2	1	-	-	1	1
CO-2	3	3	1	1	2	-	-	-	-	-	-	2	2
CO-3	3	3	2	1	3	-	-	-	-	-	-	1	2
CO-4	3	2	3	1	3	-	-	1	2	1	-	2	2
CO-5	3	3	3	2	3	2	1	2	2	2	1	2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Joseph Yiu," The Definitive Guide to ARM® Cortex®-M3 and Cortex®-M4 Processors", 3rd Edition, Newnes, 2013
2. K.V. Shibu," Introduction to Embedded Systems", 2nd Edition, McGraw Hill Education, 2017.

REFERENCE BOOKS:

1. James K. Peckol, "Embedded Systems: A Contemporary Design Tool", Wiley, 2009.
2. Raj Kamal, "Embedded Systems- Architecture, Programming and Design", 4th Edition, McGraw Hill Education, 2020.
3. Jonathan Valvano," Embedded Systems: Real-Time Operating Systems for ARM Cortex-M Microcontrollers", 2nd Edition, CreateSpace Independent Pub, 2012.

E-Resources : (Recommended Tools)

1. **IDE / Firmware development:** VS Code + PlatformIO, Arduino IDE, STM32CubeIDE, Keil uVision, MPLAB X, ESP-IDF.
2. **Simulation / prototyping:** Proteus, Tinkercad Circuits, Arm Mbed simulator.
3. **Debugging / analysis:** Logic analyzer, oscilloscope, JTAG/SWD debugger, Wireshark.
4. **AI-assisted development:** ChatGPT, GitHub Copilot, local LLM tools such as Ollama for privacy-sensitive work.
5. **Learning resources:** Arm learning resources, Coursera embedded systems courses, embedded systems reference libraries and tutorials.

Recommended Tools

1. Keil uVision
2. STM32CubeIDE
3. PlatformIO
4. VS Code
5. MATLAB / Simulink (Embedded Coder)
6. Proteus
7. TensorFlow Lite
8. FreeRTOS

RESEARCH INTEGRATION TEACHING

1. **Paper Review:** Students study recent research on embedded architectures, RTOS, IoT, TinyML, and edge AI.
2. **Case Studies:** Analyze real-world embedded systems such as smart devices, industrial controllers, and AI-enabled edge applications to connect theory with practice.
3. **Mini Presentation:** Present short reviews on emerging topics like low-power design, sensor interfacing, secure boot, and embedded AI deployment.
4. **AI-Assisted Research:** Use AI tools to summarize and compare recent papers on embedded firmware optimization, communication protocols, and intelligent embedded systems.
5. **Critical Reflection:** Identify research gaps, design limitations, reliability issues, and ethical concerns in autonomous and connected embedded systems.

Assignment 1: Embedded Architecture Analysis using AI Student prompt to AI tool:

"Explain the architecture of an embedded system using a microcontroller-based real-world example."

Students must:

1. Analyze the processor architecture and system components.
2. Identify the role of memory, interrupts, and peripherals.
3. Compare the embedded architecture with general-purpose computing systems.
4. Suggest suitable applications for the given architecture.

Assignment 2: Embedded Hardware Interface Design Student prompt to

AI tool: "Design a hardware interface for sensors and actuators in an

embedded system." Students must:

1. Analyze the interface requirements for sensors and actuators.
2. Evaluate memory and peripheral selection for the design.
3. Identify possible hardware constraints and low-power concerns.
4. Suggest improvements for reliability and efficiency.

Assignment 3: Embedded Communication and Debugging Student prompt to AI tool:

"Explain how I2C, SPI, UART, or CAN can be used in an embedded application."

Students must:

1. Compare the communication protocols and their use cases.
2. Evaluate the suitability of each protocol for a given embedded scenario.
3. Analyze debugging methods using logic analyzers and oscilloscopes.
4. Suggest the most appropriate communication method for the system.

Assignment 4: RTOS-Based Embedded Software Design Student prompt to AI tool:

"Develop an RTOS-based scheduling plan for a real-time embedded application."

Students must:

1. Analyze task priorities and timing constraints.
2. Evaluate the suitability of RTOS concepts for the application.
3. Identify scheduling and memory management issues.
4. Suggest code optimization techniques for real-time performance.

Assignment 5: IoT and Edge AI Embedded System Design Student prompt to AI tool:

"Design an intelligent embedded system integrating IoT and edge AI for a smart application." Students must:

1. Analyze the role of embedded systems in IoT architecture.
2. Evaluate the edge AI/TinyML approach for the application.
3. Identify cybersecurity, privacy, and reliability concerns.
4. Propose a human-centered and secure embedded solution.

LAB EXERCISES

S. No	Exercise Title	Aim / Objective	Tools	Hrs	CO
1	Design and Implementation of Smart Sensor Monitoring System using Embedded AI	To design an embedded system architecture integrating sensors with AI-based decision-making	Arduino / Raspberry Pi, Embedded C, Python (TinyML), Sensors	2h	CO1
2	Development of AI-Based Predictive Maintenance System using Sensor Data	To interface multiple sensors and analyze data for predictive maintenance	Arduino, Sensors (Vibration/Temp), Python	2h	CO2
3	Implementation of Temperature and Humidity Monitoring System with Anomaly Detection	To design a sensor-based monitoring system with anomaly detection logic	DHT11/DHT22 Sensor, Arduino, Python	2h	CO2
4	Design of Smart Parking System using Embedded Sensors and AI Logic	To implement sensor interfacing and real-time decision-making for parking management	Ultrasonic Sensors, Arduino, Embedded C	2h	CO2
5	Design of Smart Energy Monitoring and Optimization System	To develop communication-enabled embedded system for energy monitoring	Arduino, Energy Meter Sensors, IoT Platform	2h	CO3
6	Development of Voice-Controlled Embedded System using Edge AI	To implement communication interfaces and voice processing in embedded systems	Raspberry Pi, Microphone, Python, Speech Libraries	2h	CO3
7	Implementation of Autonomous Obstacle Avoidance Robot using Edge AI	To design a real-time embedded system with task scheduling and control logic	Arduino/Raspberry Pi, Ultrasonic Sensors, Motor Driver	2h	CO4
8	Development of Edge AI-Based Health Monitoring Device using TinyML	To implement TinyML models for real-time health monitoring on embedded devices	Arduino Nano 33 BLE Sense, TensorFlow Lite	2h	CO5

9	Implementation of Basic Image Classification on Embedded Device (TinyML)	To deploy and run image classification models on resource-constrained embedded systems	Raspberry Pi / Arduino, TensorFlow Lite	2h	CO5

Outcome of Assignments & Projects

Students will:

1. Develop AI-assisted embedded system design skills
2. Gain hands-on experience with real hardware constraints
3. Learn debugging, optimization, and validation of AI models on edge devices
4. Build industry-relevant intelligent embedded applications

ASSESSMENT SCHEME

Professional Core Courses (Theory)

Criteria	Weightage of Marks
Continuous Internal Assessment	30
Mid-Semester Examination (MSE)	30
Semester End Examination (SEE)	40
Total	**60 marks**

SKILL ENHANCEMENT COURSE-I
JAVA PROGRAMMING
[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Course Code	: 25AM2307	**Credits**	: 02
Hours / Week	: 03 Hours	**Total Hours**	: 26 Hours
L–T–P–S	: 1–0–2–0		

Course Learning Objectives:

This course will enable students to:

1. Understand the basic principles of object-oriented programming and their application in Java.
2. Demonstrate proficiency in writing and executing simple Java programs using fundamental programming constructs.
3. Apply problem-solving skills to design and implement Java programs that solve real-world computational problems.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Real-World Examples:*** *Use* ***real-world examples*** *and scenarios to demonstrate the practical relevance of Java programming concepts which enhances their understanding of how Java is used in real-world applications*
2. ***Interactive Coding Sessions:*** *Conduct* ***Interactive coding sessions*** *where students can code alongside the teacher or participate in coding challenges. This promotes active participation and helps students develop their coding skills*
3. ***Project-Based Learning****: Assign projects or mini-projects that require students to apply Java programming concepts and develop complete applications. This approach fosters independent learning, problem solving skills and a deeper understanding of Java Programming Principles.*
4. ***Active Learning Strategies:*** *Incorporate* ***active learning strategies*** *such as group discussions, problem-solving activities, case studies, and hands-on coding exercises. This allows students to actively engage with the material and apply their knowledge in practical scenarios*
5. ***Collaborative Learning:*** *Encourage students to work in pairs or small groups on programming tasks.* ***Collaborative learning*** *promotes peer-to-peer learning, fosters teamwork, and allows for the exchange of ideas and knowledge.*

6. ***Guest Speakers and Industry Connections:*** *Invite* ***guest speakers, industry professionals, or alumni*** *who work with Java programming to share their experiences and insights. This provides students with real-world perspectives and helps them understand the practical applications of Java programming*

UNIT – I	**06 Hours**
An Overview of Java: Object-Oriented Programming, A First Simple Program, A Second Short Program, Two Control Statements, Using Blocks of Code, Lexical Issues, The Java Class Libraries, **Data Types, Variables, and Arrays**: Java Is a Strongly Typed Language, The Primitive Types, Integers, Floating-Point Types, Characters, Booleans, A Closer Look at Literals, Variables, Type Conversion and Casting, Automatic Type Promotion in Expressions, Arrays ***(Text book-1: Ch 2, Ch 3).***	
UNIT – II	**05 Hours**
Operators: Arithmetic Operators, The Bitwise Operators, Relational Operators, Boolean Logical Operators, The Assignment Operator, The ? Operator, Operator Precedence, Using Parentheses, **Control Statements**: Java's Selection Statements, Iteration Statements, Jump Statements. ***(Text book 1: Ch 4, Ch 5).***	
UNIT – III	**05 Hours**
Introducing Classes: Class Fundamentals, Declaring Objects, Assigning Object Reference Variables, Introducing Methods, Constructors, The this Keyword, Garbage Collection, The finalize( ) Method, A Stack Class, **A Closer Look at Methods and Classes**: Overloading	
Methods, Using Objects as Parameters, A Closer Look at Argument Passing, Returning Objects, Recursion, Introducing Access Control, Understanding static ***(Text book 1: Ch 6, Ch 7).***	
UNIT – IV	**05 Hours**
Inheritance: Inheritance Basics, Using super, Creating a Multilevel Hierarchy, When Constructors Are Executed, Method Overriding, Dynamic Method Dispatch, Using Abstract Classes, Using final with Inheritance, The Object Class. Packages: **Packages**, Packages and member Access, Importing Packages, **Interfaces**, ***(Text book 1: Ch 8,Ch 9).***	
UNIT – V	**05 Hours**
Exception Handling: Exception-Handling Fundamentals, Exception Types, Uncaught Exceptions, Using try and catch, Multiple catch Clauses, Nested try Statements, throw, throws, finally, Java's Built-in Exceptions, Creating Your Own Exception Subclasses, Three Recently Added Exception Features, Using Exceptions. ***(Text book 1: Ch 10).***	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course, the student will be able to:		

1	Demonstrate an understanding of object-oriented programming concepts in Java.	L2
2	Apply different operators and control statements in Java to solve programming problems.	L3
3	Design and implement classes, methods, and constructors in Java to create reusable and modular code.	L4
4	Utilize inheritance, packages, and interfaces in Java to develop complex object-oriented programs.	L4
5	Handle exceptions effectively in Java programs and develop robust error-handling mechanisms.	L5

Mapping Levels of COs to POs / PSOs

				Program Outcomes (POs)								**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
Cos	Engineering knowledge	Problem analysis	Design	Conduct investigations of complex problems	tool usage	The engineer and society	Environment and sustainability	Ethics	team work	Communication	Project management and finance	Life-long learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	2				1				2			3	
CO2		3	2			2				2				3
CO3	2	2	3						2	2		2		2
CO4	2	2	3			1							2	
CO5		2		2	2	2								2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Herbert Schildt, Java The Complete Reference, 11th Edition, Tata McGraw Hill, 2019. (Chapters 2, 3, 4, 5, 6,7, 8, 9,10)

REFERENCE BOOKS:

1. H.M.Dietel and P.J.Dietel,Java, How to Program, 11th Edition, Pearson Education/PHI, 2017
2. E Balagurusamy, Programming with Java A primer, McGraw Hill Education,2019.

E-Resources:

1. https://onlinecourses.nptel.ac.in/noc22_cs47
2. https://www.geeksforgeeks.org/java/
3. https://www.codecademy.com/learn/learn-java

Activity Based Learning (Suggested Activities in Class)

1. **Code Walkthrough and Debugging**:
   a) Provide students with a pre-written Java program that contains bugs or logical errors. Divide the class into small groups or pairs.
   b) Instruct each group to identify and fix the errors in the code, encouraging them to use their knowledge of Java programming concepts.
   c) After a set time, have each group present their fixed code to the class and explain their debugging process.
2. **Mini Java Project:**
   a) Divide the class into small groups and assign each group a specific programming problem or scenario that requires a Java solution.
   b) Instruct each group to design and implement a Java program to solve the assigned problem, utilizing concepts learned in class.
   c) Provide guidance and support as needed while encouraging students to apply their creativity and problem-solving skills.

****

IV SEM – ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

SL	Course Type	Course Code	Course Name	Teaching Hours / Week				Examination			
				Lecture	Tutorial	Practical	Project	CIE Marks	SEE Marks	Total Marks	Credit
				L	T	P	J				
1	BSC	25AM2401	Transform and Numerical Techniques	3	0	0	0	60	40	100	03
2	IPCC	25AM2402	Design and Analysis of Algorithms	3	0	2	0	60	40	100	04
3	IPCC	25AM2403	Computer Networks	3	0	0	0	60	40	100	03
4	PCC	25AM2404	Theory of Computation and System Software	3	1	0	0	60	40	100	04
5	IPCC	25AM3405	Machine Learning	3	0	2	0	60	40	100	04
6	PCC	25AM2406	Full Stack Development	3	0	0	0	60	40	100	03
7	SEC	25AM2407	Sensors, Systems & IoT Fundamentals	2	0	2	0	60	40	100	03
8	AEC	25AM2408	Cognitive and Technical Skills-IV	--	--	--	--	--	--	--	Pass/Fail
9	PCC	25AM2409	AI for Sustainability	0	0	0	2	100	--	100	01
			Total	19	1	06	0				**25**

IPCC-Integrated Professional Core Course, PCC-Professional Core Courses, SEC-Skill Enhancement Courses, PEC-Professional Elective Courses, BSC-Basic Science, AEC-Ability Enhancement Course, L – Lecture, T – Tutorial, P – Practical, J– Project, C – No. Of Credits

TRANSFORMS AND NUMERIAL TECHNIQUES

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Course Code	: 25AM2401	**Credits**	: 03
Hours / Week	: 03 Hours	**Total Hours**	: 39 Hours
L–T–P–J	: 3–0–0–0		

Course Learning Objectives:

This Course will enable students to:

1. **Apply** their knowledge of Laplace transforms and inverse Laplace transforms to proficiently solve linear ordinary differential equations with constant coefficients, facilitating the analysis and modelling of complex systems.
2. **Analyze** periodic functions using Fourier series, assessing the convergence properties and precision of the series expansion, thereby enhancing their ability to understand and manipulate periodic phenomena.
3. **Utilize** complex exponential form, Fourier transforms of basic functions, and Fourier sine and cosine transforms to solve problems involving Fourier integrals, developing proficiency in applying these techniques to various mathematical scenarios.
4. **Employ** numerical methods, including Euler's Method, Runge-Kutta 4th order, Adams-Bashforth, and Adams-Moulton Methods, to solve differential equations and effectively analyze dynamic systems, enabling them to model real-world phenomena and make accurate predictions**.**
5. **Apply** finite difference methods, including the Crank-Nicolson method and appropriate techniques for hyperbolic PDEs, to effectively solve different types of partial differential equations (PDEs) such as elliptic, parabolic, and hyperbolic equations, enhancing their problem-solving skills in the context of differential equations and their applications.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them. Discuss how every ***concept can***

be applied to the real world - and when that's possible, it helps improve the students' understanding	
UNIT – I: Laplace Transform and Inverse Laplace Transform	**09 Hours**
Laplace Transforms of Elementary functions (without proof), ***(Text Book-1: Chapter 6: 203 to 207).*** Laplace Transforms $e^{at}f(t), t^n f(t)$ and $\frac{f(t)}{t}$ of Periodic functions, Unit step function and impulse functions ***(Text Book-1: Chapter 6:208-230)***. Inverse Laplace Transforms- By the method of Partial Fractions, Logarithmic and Trigonometric functions, Convolution Theorem, Inverse Laplace transform using Convolution Theorem ***(Text Book-1: Chapter 6: 238)***. Solution to Differential Equations by Laplace Transform. ***(Text Book-1: Chapter 238-242)***.	
UNIT – II: Fourier Series	**09 Hours**
Periodic Functions, Trigonometric Series ***(Text Book-1: Chapter 11: 495).*** Fourier series Standard function, Functions of any Period 2L, Even and Odd functions, Half-range Expansions. ***(Text Book-1: Chapter 11: 483-492)*** Practical Harmonic analysis (calculate average power and RMS values of periodic waveforms)	
UNIT – III: Fourier Transform	**06 Hours**
Calculation of Fourier integrals using complex exponential form ***(Text Book-1: Chapter 11: 510).*** Fourier transform of basic functions ***(Text Book-1: Chapter 11: 510-516).*** Fourier sine and cosine transforms. ***(Text Book-1: Chapter 11: 518-522).***	
UNIT – IV: Numerical Methods for Solving Ordinary Differential Equations	**07 Hours**
Euler's Method-Basic principles of Euler's method for solving first-order ODEs ***(Text Book-1: Chapter 1:10-12).*** Runge-Kutta 4th order ***(Text Book-1: Chapter 21:904).*** Multistep Methods-Explanation of multistep methods (Adams-Bashforth, Adams–Moulton Methods) ***(Text Book-1: Chapter 21:911- 913).*** Second-Order ODE. Mass–Spring System (Euler Method, Runge–Kutta Methods) ***(Text Book-1: Chapter 21:916-918).***	
UNIT – V: Numerical Methods for Partial Differential Equations	**08 Hours**
Classification of PDEs (elliptic, parabolic, hyperbolic), ***(Text Book-1: Chapter 21:922-923).*** Finite Difference Methods (Laplace and Poisson Equations), Derivation of finite difference approximations ***(Text Book-1: Chapter 21:923-927).*** Crank–Nicolson Method ***(Text Book-1: Chapter 21:938-941).*** Method for Hyperbolic PDEs ***(Text Book-1: Chapter 21:943-945).***	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	**Apply** Laplace transforms and inverse Laplace transforms to solve linear ordinary differential equations with constant coefficients, demonstrating proficiency in system analysis and modelling.	L3
2	**Analyze** periodic functions using Fourier series and evaluate the convergence properties and precision of the series expansion.	L2 & L3
3	**Solve** problems involving Fourier integrals by applying complex exponential form, Fourier transforms of basic functions, and Fourier sine and cosine transforms.	L3
4	**Utilize** numerical methods such as Euler's Method, Runge-Kutta 4th order, Adams-Bashforth, and Adams-Moulton Methods to solve differential equations and analyze dynamic systems	L2 & L3
5	**Apply** finite difference methods, including the Crank-Nicolson method and appropriate techniques for hyperbolic PDEs, to solve various types of partial differential equations (PDEs) such as elliptic, parabolic, and hyperbolic equations.	L3

Mapping Levels of COs to POs / PSOs

				Program Outcomes (POs)								**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
Cos	Engineering knowledge	Problem analysis	Design	Conduct investigations of complex problems	tool usage	The engineer and society	Environment and sustainability	Ethics	team work	Communication	Project management and finance	Life-long learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	2	2	1					1					
CO2	3	2	2						1					
CO3	3	2	2	1					1					
CO4	3	2	2	1					1					
CO5	3	2	2	1					1					

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Erwin Kreyszig, Advanced Engineering Mathematics, 2015, 10th Edition, Wiley India.

REFERENCE BOOKS:

1. Higher Engineering Mathematics, B.S. Grewal, 2015, 43rd Edition, Khanna Publishers.
2. Higher Engineering Mathematics, John Bird, 2017, 6 th Edition, Elsevier Limited.

E-Resources:

1. https://nptel.ac.in/courses/111106139
2. https://nptel.ac.in/courses/111101164
3. https://nptel.ac.in/courses/111105038

****

DESIGN AND ANALYSIS OF ALGORITHMS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Course Code	:	25AM2402	**Credits**	:	04
Hours / Week	:	05 Hours	**Total Hours**	:	39(Th)+26(P) Hours
L–T–P–J	:	3–0–2–0			

Course Learning Objectives:

This Course will enable students to:

1. **Analyze** the non-recursive and recursive algorithms and to represent efficiency of these algorithms in terms of the standard Asymptotic notations.
2. **Acquire** the knowledge of Brute Force and Divide and Conquer techniques to design the algorithms and apply these methods in designing algorithms to solve a given problem.
3. **Master** the Decrease and Conquer, Transform and Conquer algorithm design techniques, and Time versus Space Trade-offs.
4. **Learn** Greedy method and dynamic programming methods and apply these methods in designing algorithms to solve a given problem.
5. **Understand** the importance of Backtracking and Branch and Bound algorithm design techniques to solve a given problem.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**

INTRODUCTION:
What is an Algorithm? Fundamentals of Algorithmic Problem Solving.
(Text Book-1: Chapter 1: 1.1 to 1.2)
FUNDAMENTALS OF THE ALGORITHMS EFFICIENCY:
Analysis Framework, Asymptotic Notations and Standard notations and common functions ***(Text Book-2: Chapter 3: 3.1, 3.2)***,
Mathematical Analysis of Non-recursive and Recursive Algorithms,
(Text Book-1: Chapter 2: 2.1, 2.3, 2.4,)

UNIT – II	**08 Hours**

BRUTE FORCE:
Background, Selection Sort, Brute-Force String Matching. TSP
(Text Book-1: Chapter 3: 3.1, 3.2)
DIVIDE AND CONQUER:
General method, Recurrences: The substitution method, The recursion-tree method, The master method.
(Text Book-2: Chapter 4: 4.4, 4.5),
Merge sort, Quick sort, Binary Search, Multiplication of large integers,
Case study: Strassen's Matrix Multiplication.
(Text Book-1: Chapter 4: 4.1 to 4.3, 4.5)

UNIT – III	**06 Hours**

DECREASE & CONQUER:
General method, Insertion Sort, Graph algorithms: Depth First Search, Breadth First Search, Topological Sorting
TRANSFORM AND CONQUER:
Case study: Heaps and Heap sort.

TIME AND SPACE TRADEOFFS:
Input Enhancement in String Matching: Horspool's algorithm, Hashing: Open and Closed hashing.
(Text Book-1: Chapter 5: 5.1 to 5.3, Chapter 6: 6.3 to 6.4, Chapter 7: 7.2 to 7.3)

UNIT – IV	**9 Hours**

GREEDY TECHNIQUE:
General method of Greedy technique, Single-Source Shortest Paths: General method, The Bellman-Ford algorithm, Single-Source Shortest Paths in DAGs, Dijkstra's Algorithm
(Text Book-2: Chapter 24: 24.1 to 24.3).
Minimum Spanning Trees: Prim's Algorithm, Optimal Tree problem: Huffman Trees;
Case study: Kruskal's Algorithm. Fractional Problem
(Text Book-1: Chapter 9: 9.1, 9.2, 9.4).
DYNAMIC PROGRAMMING:
General method, The Floyd-Warshall Algorithm, Johnson's algorithm for sparse graphs ***(Text Book-2: Chapter 25: 25.1 to 25.3)***,
The Knapsack problem ***(Text Book-1: Chapter 8: 8.4).***

UNIT – V	**08 Hours**

LIMITATIONS OF ALGORITHMIC POWER
P, NP and NP-complete problems ***(Text Book-1: Chapter 11: 11.3***)

BACKTRACKING:
General method, N-Queens problem, Subset-sum problem.
(Text Book-1: Chapter 12: 12.1)
BRANCH AND BOUND:
General method, Travelling Salesman problem, Approximation algorithms for TSP.
Case study: Knapsack Problem.
(Text Book-1: Chapter 12: 12.2, 12.3)

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	**Exemplify** the algorithm design techniques and standard Asymptotic notations. Analyze non-recursive and recursive algorithms to obtain worst- case running times of algorithms using asymptotic analysis	L3
2	**Interpret** the brute-force, divide-and-conquer paradigms and explain when an algorithmic design situation calls for it. Recite algorithms that employ this paradigm. Synthesize divide-and-conquer algorithms. Derive and solve recurrences describing the performance of divide-and-conquer algorithms.	L3
3	**Demonstrate** the Decrease and Conquer, Transform and Conquer algorithm design techniques and analyze the performance of these algorithms.	L3
4	**Identify and interpret** the greedy technique, dynamic-programming paradigm as to when an algorithmic design situation calls for it. Recite algorithms that employ this paradigm. Synthesize dynamic-programming algorithms and analyze them	L3
5	**Illustrate** the Backtracking, Branch and Bound algorithm design paradigms and explain when an algorithmic design situation calls for it. Recite algorithms that employ these paradigms. **Summarize** the limitations of algorithmic power.	L3

Mapping Levels of COs to POs / PSOs

Cos				Program Outcomes (POs)									PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2

	Engineering knowledge	Problem analysis	Design	Conduct investigations of complex problems	tool usage	The engineer and society	Environment and sustainability	Ethics	team work	Communication	Project management and finance	Life-long learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	3											2	2
CO2	3	3	2										2	2
CO3	3	3											1	2
CO4	3	3	2										2	2
CO5	3	3											2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Anany Levitin, "Introduction to the Design & Analysis of Algorithms", 2nd Edition, Pearson Education, 2011.
2. Thomas H. Cormen, Charles E.Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", 3rd Edition, PHI, 2014.

REFERENCE BOOKS:

1. Horowitz E., Sahni S., Rajasekaran S, "Computer Algorithms", Galgotia Publications, 2001.
2. R.C.T. Lee, S.S. Tseng, R.C. Chang & Y.T.Tsai, "Introduction to the Design and Analysis of Algorithms A Strategic Approach", Tata McGraw Hill, 2005.

E-Resources:

1. https://nptel.ac.in/courses/106/101/106101060/
2. http://cse01-iiith.vlabs.ac.in/
3. http://openclassroom.stanford.edu/MainFolder/CoursePage.php?course=IntroToAlgorithms
4. https://www.coursera.org/specializations/algorithms

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving and puzzles using group discussion. E.g., Fake coin identification, Cabbage puzzle, Konigsberg bridge puzzle etc.,
2. Demonstration of solution to a problem through programming.

LABORATORY EXPERIMENTS

Total Contact Hours: 26

Following are experiments to be carried out using either C programming language or Object-oriented programming language:

1. Apply divide and conquer method and Design a C program to implementation of Binary Search algorithm.
2. Sort a given set of n integer elements using Merge Sort method and compute its time complexity. Demonstrate this algorithm using Divide-and-Conquer method.
3. Sort a given set of n integer elements using Quick Sort method and compute its time complexity. Demonstrate this algorithm using Divide-and-Conquer method.
4. Incorporate the array data structure and demonstrate whether a given unweighted graph is connected or not using DFS method.
5. Implement the graph traversal technique using BFS method to print all the nodes reachable from a given starting node in an unweighted graph.
6. Compute the Transitive Closure for a given directed graph using Warshall's algorithm.
7. For a given weighted graph, construct an All-Pairs Shortest Paths problem using Floyd's algorithm and implement this algorithm to find the shortest distance and their shortest paths for every pair of vertices.
8. Implement 0/1 Knapsack problem using Dynamic Programming Memory Functions technique
9. Find Minimum Cost Spanning Tree for a given weighted graph using Prim's and Kruskal's algorithm.
10. From a given vertex in a weighted connected graph, determine the Single Source Shortest Paths using Dijkstra's algorithm.
11. Mini project proposal should be submitted and Implementation should be done based on the problem stated in the proposal

Open ended experiments

1. Implement Fractional Knapsack problem using Greedy Method.
2. Implement N-Queens problem using Backtracking technique.
3. implementation of Travelling Sales man problem using Dynamic programming

****

COMPUTER NETWORKS

[As per the Choice Based Credit System (CBCS) scheme]

SEMESTER - V

Course Code	**:** 26AMXXXX		**Credits**	**:**	03
Hours / Week	**:** 03		**Total Hours**	**:**	45
L–T–P–J	**:** 3–0–0–0		**CIE+SEE**	**:**	60+40 Marks

Course Vision:

To equip learners with strong foundational and practical knowledge of computer networks, enabling them to design, analyze, and manage modern communication systems effectively.

Course Objectives:

This course will enable students to:

1. Understand fundamental concepts and architectures of computer networks.
2. Analyze networking protocols and their functionality across layers.
3. Design and evaluate network systems for performance and scalability.
4. Apply concepts of routing, congestion control, and data transmission.
5. Explore modern networking technologies and applications.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. Lecture method along with traditional lecture method, different type of teaching methods may be adopted to develop the course outcomes.
2. Interactive Teaching: incorporating brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Showing Video/animation films to explain functioning of various concepts.
4. Encourage Collaborative (Group Learning) Learning in the class.
5. To make Critical thinking, asking Higher order Thinking questions in the class in the form of Quiz and writing programs with complex solutions.
6. Showing the different ways to solve the same problem and encouraging the students to come up with their own creative ways to solve them

MODULE – I	**09 Hours**

Topics: Network definition, components of networks, types of networks (LAN, MAN, WAN), network topologies, OSI reference model, TCP/IP model, transmission media (guided and unguided), Network performance metrics (bandwidth, delay, throughput), Ethernet Networks ***Applications:*** Internet, enterprise networks, wireless communication

MODULE – II	**09 Hours**

Topics: Framing techniques, error detection (CRC, checksum), error correction, flow control multiple access protocols (ALOHA, slotted ALOHA, CSMA/CD), switching techniques (circuit switching, packet switching), and MAC addressing.

Applications: LAN design- virtual local area network (VLAN), Spanning Tree Protocol (STP), link aggregation (LAG), and multi-chassis link aggregation (MLAG) , data communication systems.

MODULE – III	**09 Hours**

Topics: IP addressing (IPv4), basics of IPv6, subnetting, CIDR, routing algorithms (distance vector, link state), routing protocols (RIP, OSPF basics), packet forwarding, congestion control basics. ***Applications:*** Internet routing, large-scale networks.	
MODULE – IV	**09 Hours**
Topics: Transport layer services, UDP, TCP, reliable data transfer, flow control, congestion control, sliding window protocol. ***Applications***: Web services, real-time communication.	
MODULE – V	**09 Hours**
Topics: Application layer architecture, client-server model, peer-to-peer model, HTTP ,FTP, SMTP, DNS, Remote execution over SSH / APIs, network security basics (introduction to encryption, authentication) ***Applications:*** Web applications, email systems, distributed services.	

AI INTEGRATION & AUTHENTIC INTELLIGENCE REFLECTION (AI^2 Pedagogy)

AI INTEGRATION

- AI enables intelligent analysis of network performance and adaptive optimization of bandwidth, topology, and communication systems.
- AI enhances data transmission efficiency through smart error detection, flow control, and collision management techniques.
- AI improves routing and congestion control using predictive analytics for dynamic and large-scale network environments.
- AI optimizes transport layer performance by adapting TCP/UDP mechanisms for reliable and efficient data transfer.
- AI strengthens application layer services and security through intelligent automation, anomaly detection, and protocol optimization.

AUTHENTIC INTELLIGENCE REFLECTION

- Students develop foundational understanding and critical thinking in analyzing network structures and performance.
- Students enhance logical reasoning and problem-solving skills in data communication and LAN design.
- Students build analytical abilities to design and evaluate efficient routing and scalable network solutions.
- Students apply critical thinking to understand reliable data transfer and transport layer protocols.
- Students develop ethical awareness and practical skills in designing secure and efficient network applications.

Course Outcomes

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain networking fundamentals and layered architectures	L2
2	Analyze protocols across different network layers	L4
3	Apply routing and addressing techniques in networks	L3
4	Evaluate network performance using standard metrics	L5
5	Evaluate basic network solutions for real-world applications	L5

Mapping Levels of COs to POs / PSOs

Table: Mapping Levels of COs to POs / PSOs

	Program Outcomes (POs)											PSOs	
COS	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**
CO-1	3	2	1	1	2	–	–	–	–	–	–	2	1
CO-2	3	3	2	1	2	1	–	1	–	1	–	2	1
CO-3	3	3	3	2	2	–	–	2	–	–	–	2	1
CO-4	3	3	2	3	2	1	–	2	–	1	2	2	2
CO-5	3	2	3	2	3	1	–	2	1	1	2	2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

ASSESSMENT SCHEME

1. **Internal Assessment (IA)** refers to the Continuous Internal Assessment (CIA) with a weightage of 60%. It includes:
   - Mid Semester Examination (MSE): One descriptive examination to evaluate conceptual understanding of AI fundamentals and problem-solving approaches.
   - AI-Powered Assignment (APA): Students utilize AI tools for solving problems while demonstrating Authentic Intelligence through critical evaluation, validation of AI outputs, ethical reasoning, and responsible decision-making.

- Lab Record: Evaluation of regular laboratory work, observations, coding exercises, and maintenance of lab records.
- Lab Examination: Practical examination conducted to assess students' ability to design, implement, test, and analyse AI-based solutions using appropriate tools and techniques.

2. **External Assessment (EA)** refers to the Semester End Examination (SEE) with a weightage of 40%. This examination evaluates students' overall understanding of DBMS concepts, application skills, and their ability to analyze and solve problems effectively.

A. **Integrated Professional Core Course (Theory + Practical)**

Criteria	**Weightage of Marks**
Mid-Semester Examination (MSE)	30
AI-Powered Assignment (APA)	30
Total	60 Marks

TEXTBOOKS:

1. Computer Networks, Andrew S. Tanenbaum, David J. Wetherall, 5th Edition, 2010, Pearson.
2. Computer Networking: A Top-Down Approach James F. Kurose, Keith W. Ross, 8th Edition, 2021, Pearson.
3. Data Communications and Networking Behrouz A. Forouzan, 5th Edition, 2013, McGraw-Hill Education.

REFERENCE BOOKS:

1. Co-Intelligence: Living and Working with AI Ethan Mollick, 1st Edition, 2024, Penguin Publishing Group.
2. Computer Networks: A Systems Approach Larry L. Peterson, Bruce S. Davie, 6th Edition, 2021, Morgan Kaufmann.
3. Data and Computer Communications William Stallings, 10th Edition, 2013, Pearson.
4. Internetworking with TCP/IP Volume 1 Douglas E. Comer, 6th Edition, 2014, Pearson.

E-RESOURCES:

- Programming: Java (basic network simulation, socket programming), C/C++ (optional for protocol-level understanding).
- Network Analysis: Wireshark (packet capture and analysis)
- AI Tools: ChatGPT / AI assistants for protocol explanation, network analysis support, simulation guidance.

- Simulation Tools: NS-3, OMNeT++, Cisco Packet Tracer
- Network Administration with NVIDIA Onyx Switch System (3 hrs) | Path: VLAN, STP,LAG https://nvdam.widen.net/s/wlbgbqr7cj/nvidia-learning-training-course- catalog
- Introduction to Networking (1 hr) | Path: https://nvdam.widen.net/s/wlbgbqr7cj/nvidia-learning-training-course-catalog
- Network Ansible Essentials for Network Engineers (3 hrs) | Path:https://nvdam.widen.net/s/wlbgbqr7cj/nvidia-learning-training-course-catalog

RECOMMENDED TOOLS

1. Wireshark For packet analysis, protocol inspection, and network troubleshooting.
2. Cisco Packet Tracer For simulating network topologies, routing, and switching concepts.
3. GNS3 For advanced network emulation and real device simulation.
4. Ansible For network automation and configuration management.
5. NVIDIA Deep Learning Institute For industry-based networking courses (DPU, RDMA, automation).
6. Python (with libraries like Scapy) For network scripting, automation, and packet manipulation.

THEORY OF COMPUTATION and SYSTEM SOFTWARE

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Subject Code	: 25AM2404	**Credits**:	04
Hours / Week	: 04 Hours	**Total Hours**:	52 Hours
L–T–P–S	: 3–1–0–0		

Course Learning Objectives:

This course will enable students to:

1. Understand the concept of finite automata.
2. Explore Regular Grammar, Context-Free Language & Grammars and Pushdown Automata
3. Explain the working principles of Turing Machines as computational models and undecidability.
4. Explain the functions of assemblers, loaders, and the compilation process including compiler structure and phases.
5. Design and construct parsers using LL(1) and LR parsing techniques, perform grammar transformations, and handle syntax errors effectively.
6. Generate intermediate code using syntax-directed translation, create three-address code representations, and apply fundamental code optimisation techniques.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Real-World Examples:*** *Incorporate real-world examples and case studies that demonstrate the relevance and application of automata theory in areas such as natural language processing, compilers, and pattern recognition*.
2. ***Problem-Solving Sessions:*** *Organize problem-solving sessions where students can work through challenging problems together*.
3. ***Reflective Learning****: Encourage students to reflect on their problem-solving approaches and discuss the reasoning behind their solutions.*

UNIT – I	**11 Hours**
Part 1: Introduction to Finite Automata. The Central Concepts of Automata Theory, Deterministic Finite Automata, Non deterministic Finite Automata, An application, Finite Automata with Epsilon Transitions. Part II: Regular Expressions Concept, Relation with Finite Automata, Applications of Regular Expressions, Properties of Regular Languages.	
UNIT – II	**10 Hours**

Regular Language Context Free Grammars and Languages: Context Free Grammars, Parse Tree, Applications of Context Free Grammar, Ambiguity in Grammars and Languages, Properties of Context Free Languages, The Pumping Lemma for Context Free Languages, Closure Properties of Context Free Languages.	
UNIT – III	**10 Hours**
Definition of Pushdown Automata, Deterministic Pushdown Automata, The Languages of a PDA, , Introduction to Turing Machines: The Turing Machines, Undecidability A language that is not Recursively Enumerable, An Undecidable Problem That is RE, Post Correspondence Problem	
UNIT – IV	**8 Hours**
Introduction to System Software, ASSEMBLERS Introduction to System Software, ASSEMBLERS: Basic assembler functions: A simple assembler, Machine dependent assembler features, Machine independent assembler features: Literals, Symbol-defining statements, Expressions, Program blocks, Basic loader functions: Design of an Absolute Loader, A Simple Bootstrap Loader, Introduction: Language Processors, Structure of compiler, Phases of Compiler.	
UNIT – V	**13 Hours**
COMPILERS Part I: LEXICAL AND SYNTAX ANALYSIS: Role of lexical Analyzer, Specification of Tokens, Lexical Analyzer generator Lex. SYNTAX ANALYSIS I: Role of Parser, Syntax error handling, Error recovery strategies, Writing a grammar: Eliminating ambiguity, Left recursion, Left factoring. Part II: Top down parsing: Recursive Descent Parsing, First and follow, LL (1), –Bottom up parsing: Shift Reduce Parsing, Introduction to LR parsing Simple LR: Why LR Parsers, Items and LR0 Automaton, The LR Parsing Algorithm. SYNTAX-DIRECTED TRANSLATION: Syntax-Directed Definitions: Inherited and Synthesized Attributes, Evaluation orders for SDDs: Dependency graphs, Ordering the evaluation of Attributes,Three address code: Quadruples, Triples, indirect triples, Basic Blocks, Optimization of Basic Blocks, The Code Generation Algorithm, Peephole optimization	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Analyze and formulate finite Automata, Regular Expressions for Text Processing	L3
2	Demonstrate Understanding of Context-Free Grammars and Pushdown Automata	L3
3	Apply Turing Machines to explore Undecidability and Computational Limits	L4
4	Construct lexical analyzers and parsers (LL(1), LR) with grammar transformation capabilities.	L4
5	Implement syntax-directed translation, generate intermediate code, and perform basic code optimization.	L3

Mapping Levels of COs to POs / PSOs														
COs				Program Outcomes (POs)									PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
	Engineering Knowledge	Problem Analysis	Design & Developmen	Conduct Investigatio	Modern Tool Usage	The Engineer	Environmen t and	Ethics	Individual & Team Work	Communica tion	Project managemen	Life-long Learning	Cognitive Outcome	Skill & Design
CO1	3	3	1		2	2						1	2	1
CO2	3	3	2		2						1	1	3	2
CO3	3	3	2	2	3					2	1	3	3	3
CO4	3	2	1		3						1	2	2	2
CO5	3	2	3	3	3						2	3	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Esparza, Javier, and Michael Blondin. *Automata theory: An algorithmic approach*. MIT Press, 2023.
2. Pettorossi, Alberto. *Automata Theory and Formal Languages: Fundamental Notions, Theorems, and Techniques*. Springer Nature, 2022.
3. "System Software: An Introduction to Systems Programming" by Leland L. Beck and D.Manjula was the 3rd edition, published in 2020.
4. "Compilers: Principles, Techniques, and Tools" by Alfred V. Aho, Monica S. Lam, Ravi Sethi, and Jeffrey D. Ullman (commonly known as the "Dragon Book") 2nd Edition 2020.

REFERENCE BOOKS:

1. K.L.P. Misra and N. Chandrashekaran. Theory of Computer Science- Automata, Languages and Computation, 3rd Edn. PHI, New Delhi, 2007
2. Elaine Rich, Automata, Computability and Complexity, 1st Edition, Pearson education,2013
3. J.E. Hopcroft , R. Motwani, and J. D. Ullman, Introduction to Automata Theory, Languages and Computation, 3rd Edn. Pearson Education ,2013
4. Computation, 3rd Edn. PHI, New Delhi, 2007
5. Elaine Rich, Automata, Computability and Complexity, 1st Edition, Pearson education,2013
6. V. Raghavan, Principles of Compiler Design||, Tata McGraw Hill Education Publishers, 2010.

7. Keith D Cooper and Linda Torczon, Engineering a Compiler, Morgan Kaufmann Publishers Elsevier Science, 2004. 3. D.M.Dhamdhere, Systems Programming and operating systems, Second Revised edition, Tata McGraw Hill.

TEXTBOOKS:

E-Resources:

- https://archive.nptel.ac.in/courses/111/103/111103016/
- https://www.youtube.com/watch?v=58N2N7zJGrQ&list=PLBlnK6fEyqRgp46KUv4ZY69yXmpwKOIev
- https://www.geeksforgeeks.org/theory-of-computation-automata-tutorials/
- Compiler Design - Course (nptel.ac.in)

Activity Based Learning (Suggested Activities in Class)

Hands on activity, Quiz, Seminar

MACHINE LEARNING

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Subject Code : 25AM2405		**Credits**	: 04
Hours / Week	: 05 Hours	**Total Hours**	: 39(T)+26(P) Hours
L–T–P–S	: 3–0–2–0		

Course Learning Objectives:

This course will enable students to:

1. **Summarize** the basic concepts and different types of Machine Learning including Supervised learning, Unsupervised and Reinforcement learning Techniques.
2. **Explore** and analyze the mathematics behind Machine Learning algorithms to gain a solid understanding of probability density functions, basics of sampling theorem and estimation of the maximum likelihood.
3. **Make Use of** the Supervised Machine Learning techniques for solving appropriate real-world applications.
4. **Make Use of** the Unsupervised Machine Learning techniques , Feature Engineering and Dimensionality Reduction techniques for solving appropriate real-world applications.
5. **Evaluate** the performance of Machine Learning algorithms using appropriate metrics such as accuracy, precision, recall, FI Score and Make Use of the different Optimization and Regularization Techniques for improving the model performance.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the student's understanding.

UNIT – I	**6 Hours**

INTRODUCTION TO MACHINE LEARNING

Well-posed learning problems, Designing a Learning system. Introduction to AI, Machine learning, and Deep learning with applications. Types Of Learning: Supervised, Unsupervised, And Reinforcement Learning. Perspective and Issues in Machine Learning.

Classical Paradigm of Solving Learning Problems, The Learning Problems--Classes and Types of Learning, Fundamental of Statistical Learning And Its Framework. Introduction to Feature Representation and Extraction.

Text-2-Chapter 1

UNIT – II	**9 Hours**

MATHEMATICS FOR MACHINE LEARNING

Introduction To Statistics and Probability: Probability concepts - Axioms of probability, Notion of random variables, PMF, PDFs, CDFs. Two Random Variables, Pairs of Discrete Random Variables, Joint Probability, Conditional Probability, Bayes Theorem, Different Distributions, Univariate and Multivariate Gaussian Distribution, PDF, MLE, Motivation, Estimating Hypothesis Accuracy, Basics of the Sampling Theorem, General Approach For Deriving Confidence Intervals, Difference in the Error of Two Hypotheses, Comparing Learning Algorithms.

Text-2-Chapter 2

UNIT – III	**8 Hours**

SUPERVISED LEARNING

Introduction to Supervised Learning: Introduction to Classification, Naive Bayes Classification Binary and Multi-Class Classification, Decision Trees and Random Forest, Regression (Methods Of Function Estimation) -- Linear Regression And Nonlinear Regression, Logistic Regression, Introduction to Kernel Based Methods of Machine Learning: K-Nearest Neighborhood, Kernel Functions, SVM, Introduction to Ensemble-Based Learning.

Text-2-Chapter 2

UNIT – IV	**8 Hours**

UNSUPERVISED LEARNING

Introduction To Unsupervised Learning, Clustering (Hard and Soft Clustering) Hierarchal Clustering: K-Means, Fuzzy C-Means (FCM) Algorithm, Gaussian Mixture Models (GMM), Expectation Maximization Algorithm, Feature Engineering In Machine Learning, Dimensionality Reduction, Linear Discriminant Analysis And Principal Component Analysis.

Text-2-Chapter 3

UNIT – V	**8 Hours**

MODEL SELECTION

Machine Learning Model Validation - Confusion Matrix, Accuracy, Precision, F Score, Cost Function, Machine Learning Optimization Algorithms: Gradient Descent, Stochastic GD. Regularization: Normalization and Standardization, Overfitting, Underfitting, Optimal Fit, Bias, Variance, Cross-Validation.

Text-2-Chapter 5

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	**Describe** the basic concepts and different types of Machine Learning techniques including Supervised learning, Unsupervised learning and Reinforcement learning techniques and design Well Posed Learning system for solving the problem.	L2
2	Explore and **Analyze** the mathematics behind Machine Learning algorithms to gain a solid understanding of probability density functions, the basics of sampling theorem and estimating the maximum likelihood.	L4
3	**Apply** the learned concepts of machine learning to interpret the Supervised learning algorithms including regression and classification problems.	L3
4	**Apply** Unsupervised Machine Learning Algorithms such as Hard and Soft clustering and Feature Engineering and Dimensionality Reduction techniques for solving appropriate real-world applications.	L3
5	**Evaluate** the performance of Machine Learning algorithms using appropriate metrics such as accuracy, precision, recall, FI Score and Make Use of the different Optimization and Regularization Techniques for improving the model performance.	L5

Mapping Levels of COs to POs / PSOs														
Cos				**Program Outcomes (POs)**									**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	2	2	1	-	-	-	-	-	-	-	1	-	2	2
CO2	2	2	1	-	-	-	-	-	-	-	1	-	2	2
CO3	3	2	1	1	1	2	1	-	2	1	1	-	2	2

CO4	3	2	1	1	1	2	1	-	2	1	1	-	2	2
CO5	3	2	1	1	1	2	1	-	2	1	1	-	2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXTBOOKS:

1. Thomas M. Mitchell, Machine Learning, McGraw- Hill, Inc. New York, ISBN-13: 978-1259096952, 2017.
2. Andreas Muller, Introduction to Machine Learning with Python: A Guide for Data Scientists, O'reilly, 2016.

REFERENCE BOOKS:

1. Ethem Alpaydin, Introduction to Machine Learning (Adaptive Computation and Machine Learning series), The MIT Press; second edition, 2009.
2. U Dinesh Kumar Manaranjan Pradhan, Machine Learning Using Python, Wiley India Pvt. Ltd, 2019.
3. Mark Fenner, Machine Learning with Python for Everyone, Addison-Wesley Professional, 2019.

E-Resources:

1. https://machinelearningmastery.com
2. https://www.knuggets.com/
3. https://www.geeksforgeeks.org/machine-learning-projects/

MOOC's Courses:

1. "Introduction to Machine Learning", NPTEL
2. "Machine Learning for Engineering and Science applications", NPTEL

Activity-Based Learning (Suggested Activities in Class)

1. Workshops/Seminar (ML Projects Based on Python)
2. Quiz

LIST OF LABORATORY/PRACTICAL EXPERIMENTS ACTIVITIES TO BE CONDUCTED
1. Write a Program to Implement the Water-Jug problem using Python. 2. Apply Linear Regression on a given dataset and comment on their efficiency and performance. 3. •Implement linear regression using sklearn library. 4. •Split the obtained dataset into training and testing: 80-20, 70-30 ratio. 5. •Evaluate the model using metrics: Accuracy, Mean Square Error. 6. Apply logistic regression to the given dataset and evaluate the model using a confusion matrix. Download the dataset from the repository and import it as input. Build a Logistic Regression Model and train the dataset. 7. Apply Naïve Bayes Classifier to the given dataset and evaluate the model using a confusion matrix Precision, Recall, F score, and AUC-ROC curve. 8. Use K Nearest Neighbor technique on a given dataset and analyze the

performance by changing the value of K.

9. •Implement the KNN algorithm.
10. •Apply KNN model on the dataset and perform testing on unseen dataset.
11. •Change the value of K in KNN and analysis the performance of the model.
12. Build a machine learning model using the Decision Tree algorithm to predict whether a breast tumor is benign or malignant based on various characteristics of the tumor, improving diagnostic accuracy in medical applications.
13. Apply a Random Forest Classifier for classifying breast cancer tumors as benign or malignant based on features like radius and texture.
14. Use Support Vector Machine to perform the classification and regression on given dataset.
15. •Import the dataset and perform the classification using SVM.
16. •Evaluate the model's performance on testing dataset and validation dataset
17. •Use Sklearn's Grid Search CV method for fine tuning of hyper-parameters.
18. Apply K-Means Clustering on the collected dataset.
19. •Implement K-Means clustering using sklearn.
20. •Check for the best k-value.
21. Implement Principal Component Analysis (PCA) for dimensionality reduction and data visualization, demonstrating both from-scratch implementation and sklearn comparison on iris and digits datasets.
22. Implement both simple Perceptron and Multi-Layer Perceptron (MLP) neural networks, demonstrating binary classification with custom implementation and multi-class classification using TensorFlow/Keras.
23. Implement Byte Pair Encoding (BPE) for text processing, a tokenization technique that reduces vocabulary size by iteratively merging the most frequent character pairs in a corpus while retaining important sub word components.

FULL STACK DEVELOPMENT

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Course Code	: 25AM2406	**Credits**	: 03
Hours / Week	: 03 Hours	**Total Hours**	: 39 Hours
L–T–P–S	: 3–0–0–0		

Course Learning Objectives:

This course will enable students to:

1. To demonstrate the ability to design basic web pages using HTML and CSS
2. To utilize advanced CSS methods to design the web page, background and components.
3. To experiment with Bootstrap, Javascript and DOM for to create dynamic webpages.
4. To build web pages with the ReactJS and NodeJS as front end and back end.
5. To Model a complete web-page with mangoDB CRUD operations.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the student's understanding.

UNIT – I	**08 Hours**

Introduction to HTML: HTML syntax, HTML elements, HTML document structure, CSS selectors, properties, and values, Build your first web page, Semantics of HTML, Block and inline elements, Hyperlinks, Lists, Tables, Forms, Create a basic multi page website, Cascading effect, Advanced selectors, CSS resets, Positioning

UNIT – II	**08 Hours**

Introduction to CSS: CSS syntax ,Cascading effect, Specificity, Combining and layering selectors, Backgrounds and gradients, CSS resets, The box model, Positioning with floats, Creating a grid structure, Precise Positioning, Basic website clone.

UNIT – III	08 Hours
Bootstrap and JavaScript Basics: Introduction to Bootstrap, Bootstrap Basics ,Bootstrap Grids, Bootstrap Themes, Bootstrap CSS, Bootstrap JS, Fundamentals Of JavaScript, Fundamentals of jQuery, Fundamentals of Ajax Development, Document Object Model, DOM Manipulation, DOM Events, JavaScript Libraries (jQuery and Underscore), Simple HTML CSS JavaScript Project with AJAX.	
UNIT – IV	**08 Hours**
ReactJS and NodeJS Development: Introduction and Foundation, Node Projects, Working with shrink-wrap to lock the node modules versions, Working with asynchronous programming, Building a HTTP Server with Node.JS using HTTP APIs, React Components, React State and Props, React Event Handling, Routing in React React flux, Styling React.	
UNIT – V	**08 Hours**
MongoDB Development: Introduction to NoSQL databases, MongoDB A Database the Modern Web, CRUD Operations in MongoDB, Indexing and Aggregation, Replication and Sharding, Developing Java and Node JS Application with MongoDB. LAMP/WAMP/XAMP	

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs		
	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3
CO1	3	3	3		3				-						
CO2	3	3	3		3				-						
CO3	2	2	2		3	2			-		3	2	2	2	2
CO4	3	3	3		3	2			-		3	2	2	2	2
CO5	3	1	3		3	2			-		3	2	2	2	2

Text Books:

1. Riaz Ahmed-Full Stack Web Development For Beginners,2021
2. Chris Northwood - The Full Stack Developer: Your Essential Guide to the Everyday Skills
3. Edwin Ross Torres -Full Stack Web Development,2020

Reference Books:

1. Ahmed Bouchefra,- Full Stack Development with Angular and GraphQL, 2022
2. David Choi -Full-Stack React, TypeScript, and Node,2020.
3. Valerio De Sanctis-ASP.NET Core 5 and Angular,2021

E-Resources:

1.Manu Sharma, Full Stack Development with MongoDB, bpb publishers
2. Apress Modern Full-Stack Development by Zammetti

SENSORS, SYSTEMS & IoT FUNDAMENTALS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Subject Code	:	25AM2407	**Credits**	:	03
Hours / Week	:	04 Hours	**Total Hours**	:	30 + 30
L–T–P–S		2–0–2–0	**CIE+SEE**	:	60 + 40 Marks

Course Vision:

To develop skilled engineers with strong foundations in sensors, embedded systems, and IoT technologies for smart applications, while nurturing ethical and innovative professionals capable of designing AI-enabled connected solutions for societal needs.

Course Learning Objectives:

This course will enable students to:

1. To introduce the fundamental concepts of sensors and their role in modern intelligent systems.
2. To provide basic understanding of analog and digital sensing technologies and data acquisition.
3. To explain the architecture and components of embedded and sensing systems.
4. To develop knowledge of Internet of Things (IoT) architectures, communication protocols, and connectivity.
5. To develop AI-enabled IoT applications using Jetson Nano/ Raspberry Pi for real-time data processing and inference.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. **Lecture method** along with traditional lecture method, different type of teaching methods may be adopted to develop the course outcomes.
2. **Interactive Teaching:** incorporating brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Showing **Video/animation** films to explain functioning of various concepts.
4. Encourage **Collaborative** (Group Learning) Learning in the class.
5. To make **Critical thinking**, asking Higher order Thinking questions in the class in the form of Quiz and writing programs with complex solutions.
6. Showing the **different ways to solve** the same problem and encourage the students to come up with their own creative ways to solve them.

Course Description:

This course introduces the fundamentals of sensors, embedded systems, and Internet of Things (IoT) technologies for building intelligent and connected systems. It covers sensing principles, data acquisition, and embedded platforms such as Arduino and ESP32 for real-time data processing. Explore IoT architectures, communication protocols including MQTT, CoAP, and HTTP, and cloud-based data integration using modern IoT platforms.

The course further integrates Artificial Intelligence with IoT (AIoT), enabling students to design smart applications through data-driven decision-making. Hands-on learning is emphasized through the use of Jetson Nano/Raspberry Pi for image data collection, model training, and real-time inference. Applications span across smart homes, healthcare, agriculture, industrial automation, and intelligent transportation systems.

The course adopts an AI^2 pedagogy, encouraging students to critically evaluate AI-generated solutions, address ethical considerations, and develop reliable, real-world intelligent system.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Understand sensor principles, characteristics, and data acquisition techniques for intelligent systems.	L2
2	Apply embedded system concepts and edge devices for sensor interfacing and real-world applications	L3
3	Analyze IoT architectures, networking models, and communication protocols for connected systems.	L4
4	Apply IoT platforms and cloud technologies for data analytics and system integration.	L3
5	Develop AI-enabled IoT applications using Jetson Nano/ Raspberry Pi for data collection, model training, and real-time inference.	L3

Course Outcome:

Course Modules:

UNIT – I:	**06 Hours**

Introduction to Sensors & Physical Systems

Analog vs digital signals, Introduction to sensors and its characteristics (Analog and digital sensors, environmental sensors, motion sensors), data acquisition fundamentals, role of sensors in intelligent systems.

Applications: Environmental monitoring, smart homes, healthcare monitoring devices, robotics sensing systems, wearable technologies.

UNIT – II:	**06 Hours**

Embedded Systems & Edge Devices

Fundamentals of embedded systems, microcontrollers Arduino and ESP32 Architecture, sensor interfacing, edge computing basics. ***Applications:*** Industrial monitoring systems, smart agriculture, health monitoring systems, robotics, and automation.

UNIT – III:	**05 Hours**

IoT Architecture & Networking

Internet of Things fundamentals, IoT layered architecture, wireless sensor networks, IoT communication protocols: MQTT (Publish–subscribe model), CoAP (REST-based communication, designed for constrained devices), HTTP (Request–response model used in web-based IoT applications), connectivity technologies (WiFi, Bluetooth, Zigbee).

Applications: Smart homes, smart agriculture, industrial IoT monitoring systems, connected healthcare devices.

UNIT – IV:	**05 Hours**

IoT Platforms, Data Analytics & Cloud Integration

IoT cloud platforms, device management, cloud connectivity, IoT analytics platforms, introduction to AWS IoT, Azure IoT, and Google IoT services. ***Applications***: Smart logistics, predictive maintenance systems, connected supply chains, smart infrastructure monitoring.
UNIT – V: **08 Hours**
AI-Enabled IoT Systems & Real-Time Video Analytics Introduction to Jetson Nano/Raspberry Pi, Setup and configuration of Jetson Nano/ Raspberry Pi with camera interface, Image data collection for classification models, annotation of image data for regression tasks, Training of neural networks using collected datasets to build custom models, Deployment and real-time inference on Jetson Nano using trained models. ***Applications:*** Smart surveillance systems, intelligent transportation and traffic monitoring, industrial automation and defect detection, smart retail analytics, and healthcare monitoring systems.

Mapping Levels of COs to POs / PSOs

Mapping Levels of COs to POs / PSOs													
				Program Outcomes (POs)								**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**
COs	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Engineering Tool Usage	The Engineer and the world	Ethics	Individual & Collaborative Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	2	1	-	2	-	-	1	-	-	1	1	1
CO2	3	3	3	1	3	-	-	1	-	-	2	2	3
CO3	3	3	2	2	3	-	-	1	1	1	2	2	2
CO4	3	3	3	2	3	1	-	1	1	1	2	2	3
CO5	3	3	3	2	3	2	1	2	2	2	2	2	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

ASSESSMENT SCHEME:

1. **"Internal Assessment (IA)"** refers to the Continuous Internal Assessment (CIA) with a weightage of 60%. It includes:

a. **Mid Semester Examination (MSE):** One descriptive examination to evaluate conceptual understanding of AI fundamentals and problem-solving approaches.

b. **Continuous Assessment Tools (CAT – Six Components):** Includes activities such as quizzes, surprise tests, case studies, group discussions, project-based learning, seminars, industrial visits, reflection notes, model building, e-course certifications, or open book assignments.

c. **AI-Powered Assignment (APA):** Students utilize AI tools for solving problems while demonstrating Authentic Intelligence through critical evaluation, validation of AI outputs, ethical reasoning, and responsible decision-making.

2. **"External Assessment (EA)"** refers to the Semester End Examination (SEE) with

a weightage of 40%. This examination evaluates students' overall understanding of AI concepts, application skills, and their ability to analyze and solve problems effectively.

A. Integrated Professional Core Courses (Theory + Practical)

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT) every 2 weeks (Minimum 06 assessments including AI-powered Assignment)	20
Mid Semester Examination (MSE)	20
Lab Record	05
Lab Examination	15
Total	**60 marks**

Recommended Tools

- **Programming & Development**: Python, Arduino IDE, PlatformIO, VS Code
- **IOT Simulation & Development Platforms**: Cisco Packet Tracer (IoT Mode), Proteus /EWB, Tinkercad Circuits
- **IoT Communication & Integration Tools**: MQTT Explorer, Node-RED, Postman
- **Cloud & IoT Platforms**: ThingSpeak, AWS IoT, Azure IoT, Google Cloud IoT
- **AI& ML Integration Tools**: TensorFlow / TensorFlow Lite, Scikit-learn

LAB EXERCISES:

S.No	Exercise Title	Aim / Objective	Tools	Hrs	CO
MODULE I · Exercises 1–2 : Introduction to Sensors & Physical Systems (06 Hours)					
1	**Sensor Data Acquisition using Arduino**	Interface temperature/humidity sensors and acquire real-time data for monitoring applications.	Arduino IDE, Arduino Uno, DHT11/DHT22	2 h	**CO1**
2	**Analog vs Digital Sensor Analysis**	Compare outputs from analog and digital sensors and analyze signal behavior, accuracy, and noise characteristics.	Arduino IDE, Potentiometer, Digital Sensor Modules	2 h	**CO1**
MODULE II · Exercises 3–4 : Embedded Systems & Edge Devices (06 Hours)					

S.No	Exercise Title	Aim / Objective	Tools	Hrs	CO
3	**Embedded System Interfacing with ESP32**	Interface multiple sensors with ESP32 and display acquired sensor data through serial monitor.	ESP32, Arduino IDE	2 h	**CO2**
4	**Edge Computing Basics**	Perform thresholding, filtering, and local decision-making on embedded devices before cloud transmission.	ESP32, Arduino IDE	2 h	**CO2**
MODULE III · Exercises 5–6 : IoT Architecture & Networking (06 Hours)					
5	**IoT Communication using MQTT Protocol**	Publish and subscribe sensor data using MQTT broker for real-time IoT communication.	ESP32, MQTT Broker, Node-RED	2 h	**CO3**
6	**Wi-Fi Based Remote Monitoring**	Transmit sensor data wirelessly using ESP32 Wi-Fi and visualize through web dashboard.	ESP32, Wi-Fi Router, Web Dashboard	2 h	**CO3**
MODULE IV · Exercises 7–8 : IoT Platforms, Data Analytics & Cloud Integration (06 Hours)					
7	**IoT Cloud Integration**	Send sensor data to cloud platforms such as ThingSpeak / AWS IoT and visualize trends.	ESP32, ThingSpeak / AWS IoT	2 h	**CO4**
8	**Smart Analytics Dashboard**	Create dashboard for live sensor monitoring, alerts, and analytics visualization.	Node-RED, ThingSpeak, Cloud Tools	2 h	**CO4**
MODULE V · Exercises 9–12 : AI-Enabled IoT Systems & Real-Time Video Analytics (12 Hours)					
9	**Jetson Nano / Raspberry Pi Setup and Camera Interface**	Configure Jetson Nano/ Raspberry Pi and interface camera module for AI applications.	Jetson Nano/ Raspberry Pi, Camera, Ubuntu	2 h	**CO5**
10	**Image Data Collection using Jetson Nano/ Raspberry Pi**	Capture and store image datasets using camera interface for AI models.	Jetson Nano/ Raspberry Pi, Python, Camera	2 h	**CO5**
11	**AI Model Training on Jetson Nano/**	Train a simple image classification model using collected datasets.	Jetson Nano/ Raspberry Pi, TensorFlow /	2 h	**CO5**

S.No	Exercise Title	Aim / Objective	Tools	Hrs	CO
	Raspberry Pi		PyTorch		
12	**Real-Time Inference on Jetson Nano/ Raspberry Pi**	Deploy trained AI model and perform live object/image classification.	Jetson Nano/ Raspberry Pi, Camera, Python	2 h	**CO5**
13	**Mini Project**	Design and develop an integrated IoT system with sensors, cloud connectivity, and AI-based analytics for real-time monitoring and decision-making.	ESP32 / Arduino, Sensors, Jetson Nano / Raspberry Pi, MQTT, Cloud Platform, Python	6 h	**CO5**

TEXT BOOKS:

1. Xiao, P., Designing Embedded Systems and the Internet of Things (IoT) with ARM Mbed, 1st Edition, John Wiley & Sons, 2018
2. Bahga, A., & Madisetti, V., Internet of Things: A Hands-On Approach, 1st Edition, Universities Press, 2015
3. Cirani, S., Ferrari, G., Picone, M., & Veltri, L., Internet of Things: Architectures, Protocols and Standards, 1st Edition, John Wiley & Sons, 2019
4. Song, H., Internet of Everything: Key Technologies, Practical Applications and Security of IoT, World Scientific Publishing, 2023.

E-Resources : (Recommended Tools)

1. MIT OpenCourseWare
2. IEEE research articles
3. NVIDIA DLI Courses
4. Dimitrios Serpanos, Marilyn Wolf, Internet of Things (IoT) Systems: Architecture, Algorithms, Methodologies, Springer, 2018. https://link.springer.com/book/10.1007/978-3-319-69715-4
5. Chuankun Wu, Internet of Things Security, Springer Nature, 2021. https://link.springer.com/book/10.1007/978-981-16-1372-2
6. Sébastien Ziegler et al., Springer Handbook of Internet of Things, Springer, 2023. https://link.springer.com/book/10.1007/978-3-031-39650-2

AI FOR SUSTAINIBILITY

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Course Code	:	25A2408	**Credits**	:	01
Hours / Week	:	01 Hours	**Total Hours**	:	13 Hours
L–T–P-J	:	0–0–0-2			

Course Objectives:

This Course will enable students to:

1. **Understand** the fundamentals of sustainability, sustainable development goals (SDGs) and their challenges.
2. **Explore and analyze** real-world sustainability problems where AI and data-driven approaches can be applied effectively.
3. **Apply** machine learning/deep learning models to sustainability-related datasets for prediction, classification, or decision support.
4. **Develop** simple AI-based project solutions that contribute to sustainability.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different type of teaching methods may be adopted to develop the course outcomes.
2. ***Interactive Teaching***: Adopt the **Active learning** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

COURSE CONTENT:

The research topic proposed by both the guide and the student team should be approved by the project coordinator and department chairman to proceed further. A degree of industrial input and involvement will be encouraged, and can be facilitated through existing academic-industrial collaborations or by addressing specific topics that are of interest and addressing towards sustainability goals.

All projects will be closely supervised by the Project Guide with ongoing feedback and guidance at all stages of the project from conception to completion.

The following criteria will be checked by the department chairman to approve for the research proposal:

a. Department staff as course guide

1. Ability to provide research direction to the student in the chosen field of interest
2. Ability to design an appropriate research strategy and methodology to carry out the research by student
3. Ability to provide and evaluate the strong literature review document for the chosen research topic
4. Ability to train students on research paper / technical writing skills
5. Conduct reviews in regular time period and submit the evaluation to department chairman

b. Student Team

1. To be dedicated and committed to work on a new research topic by learning new technical skills
2. To have fair knowledge on what is product development or research topic
3. To have constant interaction with allocated guide by providing weekly updates
4. To be committed to complete the project and submitting the technical paper within the stipulated time framed by the university

Evaluation:
There will be CIA evaluation as well as the Semester end evaluation of the work done. It will be done by a committee of senior researchers of the Department.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	**Explain** the concept of sustainability, sustainable development goals (SDGs), and the relevance of artificial intelligence in addressing sustainability challenges.	L2
2	**Identify and analyze** real-world sustainability problems where AI-based solutions can be effectively applied.	L4
3	**Apply** machine learning models to sustainability-related datasets for prediction, classification, or decision-making.	L3
4	**Develop** a simple AI-driven project solution that addresses a selected sustainability challenge using available tools, datasets, or sensor data.	L4

Table: Mapping Levels of COs to POs / PSOs														
COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
CO1	3	1				2	3	2	3			1	2	1
CO2	2	3		1	3	2	3	1	3			1	1	2
CO3	2	3		1	3		2		3			1	2	2
CO4	2	2	1	1	2	2	2	2	3	3	2	2	2	2

****

V SEM – ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

SL	Course Type	Course Code	Course Name	Teaching Hours / Week				Examination			
				Lecture	Tutorial	Practical	Project	CIE Marks	SEE Marks	Total Marks	Credit
				L	T	P	J				
1	IPCC	25AM3501	Deep Learning	3	0	2	0	60	40	100	04
2	IPCC	25AM3502	System Software and Operating System	3	0	2	0	60	40	100	04
3	PCC	25AM3503	Natural Language Models	3	1	0	0	60	40	100	04
4	PCC	25AM3504	Image Processing and Computer Vision	3	0	0	0	60	40	100	03
5	HSMC	25AM3505	Innovation and Entrepreneurship	2	0	0	0	60	40	100	02
6	SEC	25AM3506	Skill Enhancement Course –III (Cloud Computing)	1	0	2	0	100	--	100	02
7	PEC	25AM35XX	Professional Elective Course -I	3	0	0	0	100	--	100	03
8	AEC	25AM3507	Cognitive and Technical Skills-V	-	-	-	-	--	--	--	Pass/Fail
			Total	18	1	6	0				**22**

HSMC-
Humanities and Social Sciences including Management Courses, IPCC-Integrated Professional Core Course, PCC-Professional Core Courses, SEC-Skill Enhancement Courses, PEC-Professional Elective Courses, AEC-Ability Enhancement Course, L – Lecture, T – Tutorial, P – Practical, J-Project, C – No. Of Credits

Professional Elective -1 (PEC-1) Offering										
Sl. No	**Course Type**	**Course Name**	**Teaching Hours / Week**				**Examination**			
			Lecture	**Tutorial**	**Practica**	**Project**	**CIE Marks**	**SEE Marks**	**Total Marks**	**CREDITS**
			L	**T**	**P**	**J**				
1	PEC	Accelerated Data Science	2	0	0	2	100	--	100	03
2	PEC	Ethics, Safety & Governance of Agentic AI	2	0	0	2	100	--	100	03
3	PEC	Intelligent Speech Processing Systems	2	0	0	2	100	--	100	03
4	PEC	Foundations of Robotics and Intelligent Systems	2	0	0	2	100	--	100	03
5	PEC	Explainable AI	2	0	0	2	100	--	100	03
6	PEC	Cognitive Linguistics	2	0	0	2	100	--	100	03

DEEP LEARNING

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Course Code	:	25AM3501	**Credits**	:	04
Hours / Week	:	05 Hours	**Total Hours**	:	39 (T) +26(p)
L–T–P-J	:	3–0–2-0			

Prerequisites:

Proficiency in Python programming language and probability concepts.

Course Objectives:

This course will enable students to:

1. To understand the basic building blocks and general principles that allow one to design Deep learning algorithms
2. To become familiar with specific, widely used Deep learning networks
3. To introduce building blocks of Convolution neural network architecture
4. To learn to use deep learning tools and frameworks for solving real-life problems

Teaching-Learning Process (General Instructions)

1. Lecture method means it includes not only the traditional lecture method but also a different type of teaching method that may be adopted to develop the course outcomes.
2. Interactive Teaching: Adopt Active learning that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show Video/animation clips to explain the functioning of various concepts.
4. Encourage Collaborative (Group Learning) Learning in the class.
5. To make Critical thinking, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the student's understanding.

UNIT – I	**08 Hours**

INTRODUCTION TO DEEP LEARNING:

Introduction to Perceptron, Learning rules, Single-layer perceptron and its applications, Multilayer Feed forward Neural Network, neural networks training, activation functions, loss functions, Model Selection.

(Text 2: Chapters 2&3)

UNIT – II	**07 Hours**

COMPUTATION IN DEEP LEARNING:

Forward and Backward Propagation, Computational Graphs Layers, shallow neural network, deep neural network, Optimisation for Training Deep Models, Gradient Descent and its types, Regularisation techniques, Automatic Differentiation. (Text 1: Chapter 4)

UNIT – III	**08 Hours**

CONVOLUTIONAL NEURAL NETWORKS:

Convolutional Neural Networks (CNNs) - Biological inspiration, Mapping of Human Visual System and CNN. Convolution operation, Convolutional Layers, Padding and Stride, Subsampling - Pooling, Fully Connected CNN Layer, Transfer Learning, Overview of Alex Net, VGG, Google Net, Res Net, YOLO, Graph Neural Networks. (Text1: Chapter 7)

UNIT – IV	08 Hours
SEQUENCES TO MODEL: Basics of RNN, Types of RNN, RNN's For Sequence Modeling- Language Modeling, Back Propagation Through Time, LSTM Network, Bidirectional LSTMs, Applications of LSTM. (Text 1: Chapter 9, 10)	
UNIT – V	**08 Hours**
UNSUPERVISED DEEP LEARNING: Unsupervised Pretrained Networks (UPNs)- Autoencoders, Deep Belief Networks (DBNs), Generative Adversarial Networks (GANs), Restrictive Boltzmann Machines, Transformers. Deep Learning Applications in Healthcare and other areas (Case study) Text 1: Chapter 20	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Build an Image classifier model for applying the concept of single layer and multilayer NN and analyze activation and loss function with that model.	L2
2	Apply the mathematical concept of deep learning for the manipulation and preprocessing of data.	L3
3	Evaluate deep learning models applying optimization techniques to solve real-world problems and analyse the efficiency of the models.	L5
4	Build an image classifier model, applying CNN and evaluating associated hyperparameters.	L4, L5
5	Construct deep learning-based models for healthcare applications and compare effectivity of advanced networks.	L5

Mapping Levels of COs to POs / PSOs

				Program Outcomes (POs)								PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
COs	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	3	3	-	2	-	-	-	2	-	2	-	1	1
CO2	3	3	3	3	2	-	-	-	2	-	2	-	2	2
CO3	3	3	3	3	2	-	-	-	2	-	2	-	2	2
CO4	3	3	3	3	2	-	-	-	2	-	2	-	3	3
CO5	3	3	3	3	2	-	-	-	2	-	2	-	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

Lab Programs:

1. Build an Artificial Neural Network by implementing the Backpropagation algorithm and test the same using appropriate data sets. Vary the activation functions used and compare the results.
2. **Design** and build a comprehensive project leveraging various deep learning architectures to solve real-world problems effectively.
3. Analyze the performance of CNN on the image dataset.
   a. Load the dataset as input.
   b. Change the hyper-parameter (Varying with different convolutional and pooling layer) of classification model and analyze its performance.
   c. Evaluate and compare the model performance by using metrics – classification accuracy and Binary Cross Entropy Loss.
4. Design and implement a CNN model (with 2+ layers of convolutions) to classify multi category image datasets. Use the concept of padding and Batch Normalization while designing the CNN model. Record the accuracy corresponding to the number of epochs. Use the Fashion MNIST/MNIST/CIFAR10 datasets.
5. Design and implement a CNN model (with 4+ layers of convolutions) to classify multi category image datasets. Use the concept of regularization and dropout while designing the CNN model. Use the Fashion MNIST/MNIST/CIFAR10 datasets datasets. Record the Training accuracy and Test accuracy corresponding to the following architectures:

   i. Base Model
   ii. Model with L1 Regularization
   iii. Model with L2 Regularization
   iv. Model with Dropout
   v. Model with both L2 (or L1) and Dropout

6. Implement the standard VGG-16, Resnet architecture model to classify multi category image dataset and check the accuracy.
7. Implement the standard DenseNet architecture model to classify multi category image dataset and check the accuracy.
8. Implement and Compare RNN and LSTM for sentiment analysis on movie reviews/other dataset and tabulate the results.
9. Implement and Compare LSTM and Bidirectional LSTM for sentiment analysis on movie reviews/other dataset and tabulate the results.
10. Implement Generative Adversarial Networks to generate realistic Images. Use MNIST, Fashion MNIST or any human face datasets.
11. Implement Auto encoders for image denoising on MNIST, Fashion MNIST or any suitable dataset.

Open Ended Experiments:

1) “Smart Diagnosis" – Disease Prediction Using Medical Imaging.
2) "Smart Yield" – Predicting Crop Yield with Satellite and Environmental Data.
3) “Stock Sense" – Predicting Stock Market Trends Using Deep Learning.

Image Processing & Computer Vision Experiments:

1) Implement the different filtering techniques for noise removal based on spatial and frequency domains using OpenCV.
2) Write a program to implement the Image Compression technique such as Huffman Coding Algorithm.
3) Implement the Harris Corner Detector algorithm without the inbuilt Open CV() function.
4) Write a program to compute the SIFT feature descriptors of a given image.
5) Write a program to detect the specific objects in an image using HOG.

TEXT BOOKS:

1. Aston Zhang, Zack C. Lipton, Mu Li, Alex J. Smola, "Dive into Deep Learning", Amazon Science, 2020
2. Josh Patterson and Adan Gibson, "Deep Learning a Practitioners Approach", July 2018.
3. Ian Goodfellow, Yoshua Bengio, Aaron Courville, "Deep Learning", The MIT Press, 2016

REFERENCE BOOKS:

1. Tom Mitchell, Machine Learning, McGraw-Hill, 1997
2. François Chollet, "Deep Learning Python", Manning Publications, 2018
3. Neural Networks: A Comprehensive Foundation," S. Haykin, 2ndEd, Prentice Hall of India, 2003.

Activity-Based Learning (Suggested Activities in Class)

1. Assignments (in writing and doing forms on the aspects of syllabus content and outside the syllabus content. Shall be individual and challenging)
2. Student seminars (on topics of the syllabus and related aspects, individual activity)
3. Quiz (on topics where the content can be compiled by smaller aspects and data, Individuals or groups as teams)
4. Study projects (by very small groups of students on selected local real-time problems about the syllabus or related areas. The individual participation and contribution of students shall be ensured (team activity)

****

SYSTEM SOFTWARE AND OPERATING SYSTEM

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER - V

Course Code	: 25AM3502	**Credits**	: 03
Hours / Week	: 04	**Total Hours**	: 30+15
L–T–P–J	: `2–0–2–0	**CIE+SEE**	: 60+40 Marks

Course Vision:

To equip students with a strong foundation in operating system principles for designing and managing efficient, reliable, and scalable computing systems. The course aims to develop the ability to analyze and optimize system performance through effective resource management, process coordination, and memory handling, while integrating modern trends such as AI-assisted system optimization, ethical decision-making, and responsible use of computing technologies in real-world applications.

Course Objectives:

This course will enable students to:

1. **Understand** the fundamental concepts, structure, and functions of operating systems.
2. **Analyse** process management, threads, and CPU scheduling techniques.
3. **Apply** synchronisation mechanisms and deadlock handling techniques.
4. **Evaluate** memory management and virtual memory techniques.
5. **Analyse** file systems and disk scheduling methods for efficient storage management.

Teaching-Learning Process (General Instructions)

These are sample pedagogical methods, that the teacher can use to accelerate the attainment of the various course outcomes:

1. **Lecture method:** Along with traditional lecture method, different teaching methods may be adopted to explain OS concepts such as process management, memory management, and file systems.
2. **Interactive Teaching:** Incorporating brainstorming, discussions, group work, and focused problem-solving on topics like CPU scheduling, synchronization, and deadlocks.
3. **Use of Multimedia:** Showing videos/animation to explain working of OS concepts such as process execution, paging, and disk scheduling.
4. **Collaborative Learning:** Encourage group learning activities such as implementing scheduling algorithms and memory allocation techniques.
5. **Critical Thinking:** Ask higher-order thinking questions through quizzes and programming exercises on OS problems like deadlock handling and resource allocation.
6. **Multiple Problem-Solving Approaches:** Demonstrate different algorithms (FCFS, SJF, LRU, etc.) and encourage students to develop efficient and creative solutions.

MODULE – I	**06 Hours**

OS Fundamentals and System Structure

Topics: Introduction to Operating Systems, Computer System Organization & Architecture, Operating System Structure, Types of Operating Systems (Batch, Time-sharing, Real-time, Distributed), OS Services, System Calls, Virtual Machines.

Applications: Mobile operating systems, cloud platforms, embedded systems.

MODULE – II	**07 Hours**

Process Management

Topics: Process concept, Process states, Process Control Block (PCB), Threads and Multithreading, User vs Kernel Threads, Scheduling Criteria. Scheduling Algorithms: FCFS, SJF, SRTF, Round Robin, Priority Scheduling.

Applications: CPU scheduling in multitasking systems, cloud resource allocation.

MODULE – III	**07 Hours**

Process Synchronization and Deadlocks

Topics: Critical Section Problem, Peterson's Solution, Semaphores, Monitors, Classical

AI

Synchronization Problems. Deadlocks: Concept, Characterization, Prevention, Avoidance (Banker's Algorithm), Detection and Recovery. ***Applications:*** Multi-threaded applications, database transaction systems.	
MODULE – IV	**06 Hours**
Memory Management **Topics:** Contiguous Memory Allocation, Paging, Segmentation, Page Table Structure. Virtual Memory: Demand Paging, Page Replacement Algorithms (FIFO, LRU, Optimal), Frame Allocation, Thrashing. **Applications:** Virtual machines, cloud memory management.	
MODULE – V	**04 Hours**
File Systems and Disk Management **Topics:** File Concepts, Access Methods, Disk Scheduling Algorithms (FCFS, SSTF, SCAN, C-SCAN). ***Applications:*** Database storage systems, data centers.	

INTEGRATION & AUTHENTIC INTELLIGENCE REFLECTION:

- **AI Integration:**

  - Apply AI-based system monitoring and analysis tools to observe system performance, detect anomalies, and optimize resource utilization.
  - Use AI techniques to enhance CPU scheduling by predicting process behavior and improving scheduling efficiency in multitasking environments.
  - Implement AI-based approaches for deadlock detection and prevention, and analyze synchronization issues in concurrent systems.
  - Apply AI models to predict memory usage patterns and optimize memory allocation strategies in virtual memory systems.
  - Use AI-driven techniques to optimize disk scheduling and file system performance for efficient data access and storage management.

- **Authentic Intelligence Reflection:**

  - Critically evaluate the reliability and limitations of AI-based system monitoring tools and emphasize the need for human oversight in system decisions.
  - Assess fairness and efficiency in AI-assisted CPU scheduling, including risks such as starvation and bias in process allocation.
  - Analyze ethical issues in synchronization and deadlock handling, focusing on fairness in resource sharing and prevention of indefinite waiting.
  - Reflect on ethical and efficiency concerns in memory management, including resource wastage, equitable allocation, and trust in AI predictions.
  - Evaluate trade-offs between performance and fairness in disk scheduling and file systems, ensuring responsible and sustainable system design.

ASSESSMENT SCHEME :

1. **Internal Assessment (IA):** "Internal Assessment (IA)" refers to the Continuous Internal Assessment (CIA) with a weightage of 60%. It includes:

a. **Mid-Semester Examination (MSE):** One descriptive examination to evaluate conceptual understanding of OS concepts such as process management, memory management, synchronization, and file systems.

b. **Continuous Assessment Tools** (CAT – Six Components):
Includes activities such as:
   i. Quizzes and surprise tests on the OS concept
   ii. Problem-solving (CPU scheduling, deadlocks, memory allocation.
   iii. Case studies (real-world OS issues)
   iv. Group discussions (Linux vs Windows OS)
   v. Seminars on modern OS trends (cloud OS, distributed systems)
   vi. Open book assignments and mini case analysis

c. **AI-Powered Assignment (APA):** Students utilise AI tools for solving OS-related problems while demonstrating Authentic Intelligence through:
   i. Critical evaluation of AI outputs

ii. Identifying errors and inefficiencies
iii. Justifying improvements using OS concepts
iv. Ethical reasoning and responsible system design

2. External Assessment (EA)

"External Assessment (EA)" refers to the Semester End Examination (SEE) with a weightage of 40%.

This examination evaluates:

i. Conceptual understanding of OS principles , Problem-solving ability
ii. Application of scheduling, memory, and file system techniques.

Professional Core Courses (Theory)

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT) every 2 weeks (Minimum 06 assessments)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	10
Total	**60 marks**

AI-POWERED ASSIGNMENTS :

AI^2 Pedagogy: A structured approach where students use AI tools for OS problem-solving while developing Authentic Intelligence through evaluation and decision-making.

Assignment 1: AI-Based CPU Scheduling Analysis

Students prompt an AI tool:

"Generate a CPU scheduling algorithm (SJF / Round Robin)"

Students must:

1. Analyze correctness of scheduling
2. Evaluate waiting time and turnaround time
3. Suggest improvements

Assignment 2: Memory Management Evaluation

Students prompt an AI tool:

"Compare paging and segmentation techniques"

Students must:

1. Verify performance differences
2. Identify fragmentation issues
3. Evaluate real-world applicability

Assignment 3: Synchronization Debugging

Students use AI-generated faulty code for synchronization

Students must:

1. Identify the bug
2. Fix synchronization issues
3. Explain correctness

Assignment 4: OS Visualization Study

Students use AI tools to:

1. Visualize CPU scheduling (Gantt charts)
2. Analyze memory allocation
3. Compare algorithm performance

Assignment 5: System Performance Case Study

Students prompt AI:
"Analyze system performance using different scheduling and memory techniques"

Students must:

1. Compare CPU utilisation
2. Analyse efficiency
3. Suggest optimisation strategies

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain operating system concepts, structure, and services	L2
2	Analyze and apply process scheduling and threading models	L3
3	Implement synchronization and deadlock handling techniques	L3
4	Evaluate memory management and virtual memory techniques	L3
5	Analyze file systems and disk scheduling algorithms	L2

Mapping Levels of COs to POs / PSOs

Table: Mapping Levels of COs to POs / PSOs

COS	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO-1	3	2	-	–	1	-	-	-	-	–	-	2	1
CO-2	3	3	2	–	2	-	-	-	-	–	–	3	2
CO-3	3	3	2	2	2	-	-	1	1	–	–	3	2
CO-4	3	3	2	2	2	-	-	-	-	–	–	3	2
CO-5	3	3	2	2	2	-	-	-	-	-	-	3	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

PSO1: Solve complex system-level and computing problems by applying the principles of Operating Systems such as process management, CPU scheduling, synchronization, deadlock handling, memory management, and file systems along with knowledge from Artificial Intelligence, Machine Learning, and Software Engineering.
PSO2: Apply technical and research skills through laboratory exercises, projects, internships, and professional activities to design, develop, and optimize system-level programs and efficient computing solutions using modern tools, ensuring reliability, scalability, and sustainability for real-world applications.

TEXT BOOKS:

1. Abraham Silberschatz, Peter B. Galvin, Greg Gagne *Operating System Concepts*, 10th Edition, Wiley, 2018.

2. Andrew S. Tanenbaum, Herbert Bos. *Modern Operating Systems*, 4th Edition, Pearson, 2015.
3. William Stallings, *Operating Systems: Internals and Design Principles*, 9th Edition, Pearson, 2018.

REFERENCE BOOKS:

1. Remzi H. Arpaci-Dusseau, Andrea C. Arpaci-Dusseau , *Operating Systems: Three Easy Pieces*, 2018.
2. Christopher Negus, David Clinton, *Linux Bible*, 11th Edition, Wiley, 2025.

E-RESOURCES:

- AI Assistants for OS problem solving
- Jupyter Notebook / Google Colab for simulations
- OS Simulation Tools (CPU scheduling, memory visualization)
- GitHub Copilot for coding support
- Linux environment / Virtual machines

RECOMMENDED TOOLS

- C (GCC Compiler), Python 3.10+
- GitHub for version control and project development
- NumPy, Matplotlib (for analysis and visualization)
- Linux (Ubuntu) and Shell Scripting
- VirtualBox / VMware for OS experimentation

LAB EXERCISES (15 hrs)

S.No	Exercise Title	Aim / Objective	Tools & Dataset	Hrs	CO
MODULE I · Exercises 1–2 : OS Fundamentals and System Structure (02 Hours)					
1	**OS Environment + Process Creation**	Study Linux commands, system calls, and implement process creation using fork()	Linux / C	1 h	**CO1**
2	**AI-Based System Monitoring**	Monitor CPU & memory usage and detect anomalies using AI tools	Python, AI Tools	1 h	**CO1**
MODULE II · Exercises 3–4 : Process Management (03 Hours)					
3	**CPU Scheduling**	Implement FCFS, SJF, Round Robin, Priority and calculate waiting & turnaround time	C / Python	2 h	**CO2**
4	**AI vs Manual Scheduling**	Compare traditional scheduling with AI-based prediction approach	Python, AI Tools	1 h	**CO2**
MODULE III · Exercises 5–6 : Process Synchronization and Deadlocks (03 Hours)					
5	**Synchronization Problems**	Implement Producer-Consumer and Readers-Writers using semaphores	C / Python	1.5 h	**CO3**
6	**Deadlock + Debugging**	Implement Banker's Algorithm and analyze/debug synchronization issues	C / Python, AI Tools	1.5 h	**CO3**
MODULE IV · Exercises 7–8 : Memory Management (04 Hours)					
7	**Page Replacement Algorithms**	Implement FIFO, LRU, Optimal page replacement techniques	C / Python	2 h	**CO4**
8	**Memory Allocation + Analysis**	Implement First Fit, Best Fit, Worst Fit and analyze fragmentation	C / Python	2 h	**CO4**

S.No	Exercise Title	Aim / Objective	Tools & Dataset	Hrs	CO
MODULE V · Exercises 9–10 : File Systems and Disk Management (03 Hours)					
9	**Disk Scheduling Algorithms**	Implement FCFS, SSTF, SCAN, C-SCAN and analyze performance	C / Python	1.5 h	**CO5**
10	**OS Performance Analysis (Mini Case Study)**	Combine scheduling, memory, and disk concepts with AI-based analysis	Python, AI Tools	1.5 h	**CO5**

NATURAL LANGUAGE MODELS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Subject Code	: 25AM3503	**Credits**:	04
Hours / Week	: 04 Hours	**Total Hours**:	52 Hours
L–T–P–S	: 3–1–0–0		

Course Learning Objectives:

This course will enable students to:

1. To understand the algorithms available for the processing of linguistic information and computational properties of natural languages
2. To conceive basic knowledge on various morphological, syntactic and semantic NLP task
3. To understand machine learning techniques used in NLP,
4. To write programs in Python to carry out natural language processing

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the student's understanding.

UNIT – I	**08 Hours**
Past, present, and future of NLP; Classical problems on text processing; Necessary Math concepts for NLP; Regular expressions in NLP. Parts of Speech and Morphology, Phrase Structure, Semantics and Pragmatics, Corpus-Based Work: Getting Set Up, Looking at Text, Marked-up Data Text processing: lemmatization, stop word, tokenization, stemming, Spelling errors corrections–Minimum edit distance, Bayesian method	

UNIT – II	**08 Hours**
Words & Sentences, N-grams: Simple unsmoothed n-grams; smoothing, backoff, spelling correction using N-grams, Metrics to evaluate N-grams. Parts of Speech tagging: Word classes, POST using Brill's Tagger and HMMs; Information Extraction: Introduction to Named Entity Recognition and Relation Extraction WordNet and WordNet-based similarity measures, Concept Mining using Latent Semantic Analysis	

UNIT – III	**08 Hours**
Sequence to sequence & Language Modelling, Word embedding: skip-gram model, CBOW, GloVe, Language Modelling: Basic ideas, smoothing techniques, Language modeling with RNN and LSTM	

UNIT – IV	**08 Hours**

Case studies on Generative AIs in NLP: History of generative AI, ChatGPT technical overview, Generative pre-trained Transformer – 1, Generative pre-trained Transformer – 2, and Generative pre-trained Transformer – 3.
UNIT – V **07 Hours**
Advanced Topics and Hands-on Practices Python libraries supporting NLP; Hands-on Data collection - from social network platforms, pdfs, word files, JSON, HTML Parsing text using regular expression; scraping data from web; Text processing: convert to lowercase, remove punctuation, remove stop words, standardizing text, tokenising, stemming, lemmatising. Applications: Spam detection, consumer complaint classification, Semantic Analyser, Dialogue processing (Chatbots), Text summarization, Text Categorization.

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course, the student will be able to:		
1	Demonstrate an understanding of fundamental NLP concepts, including text processing techniques and classical problems in NLP.	L2
2	Analyze and evaluate different NLP methods and algorithms for tasks such as part-of-speech tagging and named entity recognition.	L4
3	Apply NLP techniques to real-world problems, such as spam detection and text summarization, using Python libraries.	L4
4	Compare and contrast advanced NLP models, such as language models using recurrent neural networks (RNNs) and generative pre-trained transformers (GPTs).	L5
5	Synthesize their knowledge of NLP concepts and evaluate the techniques to design and develop their own NLP applications, such as chatbots or text categorization systems.	L5

Mapping Levels of COs to POs / PSOs

COs				**Program Outcomes (POs)**									**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	3	2										2	1
CO2	3	3	2						2	2			2	1
CO3	3	2	1		2				2	2			2	1
CO4	3	2	2		2				2	2			2	1
CO5	3	2	2		2				2	2			2	1

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Sowmya Vajjala, Bodhisattwa Majumder, Anuj Gupta, “Harshit Surana, Practical Natural Language Processing: A Comprehensive Guide to Building Real-World Nlp Systems” - "O'Reilly Media, Inc.", 17 Jun 2020.
2. Transformers for Natural Language Processing: Build innovative deep neural network architectures for NLP with Python, PyTorch, TensorFlow, BERT, RoBERTa, and more - Denis Rothman, Packt Publishing Ltd, 2021
3. Daniel Jurafsky and James H. Martin. 2009. Speech and Language Processing: An Introduction to Natural Language Processing, Speech Recognition, and Computational Linguistics. 2nd edition. Prentice-Hall.
4. Tiwary, U. S., & Siddiqui, T. (2008). Natural language processing and information retrieval. Oxford University Press, Inc.

REFERENCE BOOKS:

1. Hands-on Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems - Aurélien Géron,Edition 2, O'Reilly Media, 2017.
2. Deep Learning for Natural Language Processing - Palash Goyal, Sumit Pandey, Karan Jain, Apress Berkeley, CA- 2018

E-Resources:

1. https://github.com/topics/nlp-models
2. https://devopedia.org/site-map/browse-articles/natural%20language%20processing
3. https://wisdomml.in/hidden-markov-model-hmm-in-nlp-python/
4. https://spotintelligence.com/2023/06/16/activation-function/
5. https://radimrehurek.com/gensim/models/word2vec.html

Activity Based Learning (Suggested Activities in Class)

1. Better Understanding the concept of Sampling and Semantic Role Labeling Quantization of Speech and using group discussion.

2. Collaborative Activity is minor project development with a team of 4 students.

IMAGE PROCESSING AND COMPUTER VISION

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Subject Code	: 25AM3504	**Credits**	: 04
Hours / Week	: 03 Hours	**Total Hours**	: 39 Hours
L–T–P–S	: 3–1–0–0		

Course Learning Objectives:

This course will enable students:

1. To understand the algorithms available for the processing of linguistic information and computational properties of natural languages.
2. To conceive basic knowledge on various morphological, syntactic, and semantic NLP tasks.
3. To understand and analyze the fundamental concepts of Computer Vision
4. To understand and analyze the fundamental concepts of Computer Vision.
5. To learn to use deep learning tools and framework for solving real-life problems related to images and signals.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainmentof the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every ***concept can be applied to the real world*** - and when that's possible,it helps improve the student's understanding.

UNIT – I	**08 Hours**

INTRODUCTION:

Digital Image Fundamentals: Elements of Visual Perception, A Simple Image Model, Sampling and Quantization, Image File Formats, Color Models, In spatial domain: Basic gray level transformations, Histogram processing, using arithmetic/Logic operations, smoothening spatial filters, Sharpening spatial filters. In Frequency domain: Introduction to the Fourier transform and frequency domain concepts, Frequency-domain filters: Low pass filter, High pass filter, Band pass filter, Sharpening frequency domain filters.

(Text1: Chapters 1, 2, 3)

UNIT – II	**08 Hours**

IMAGE SEGMENTATION:

Introduction, Detection of isolated points, line detection, Edge detection, Edge linking, Region-based segmentation- Region growing, split and merge technique, local processing, regional processing, Hough transform, Segmentation using Threshold.

COLOR IMAGE PROCESSING:

Color fundamentals, Color models, Color transformation, Smoothing and Sharpening, Color segmentation.

(Text1: Chapters 6, 10)

UNIT – III	08 Hours
MORPHOLOGICAL IMAGE PROCESSING: Erosion, dilation, opening, closing, Basic Morphological Algorithms: hole filling, connected components, thinning, skeleton. **FEATURE EXTRACTION:** Textural Features, Shape Features, Color Features. Image Compression: Introduction, coding Redundancy, Inter-pixel redundancy, image compression model, Lossy and Lossless compression, Huffman Coding. (Text1: Chapters 9, 8, 5)	
UNIT – IV	**08 Hours**
Introduction to Computer Vision, Camera Models and Calibration: Camera Projection Models – Orthographic, Affine, Perspective, Projective Geometry, Camera Internal and External Parameters, Lens Distortion Models, Local Feature Detectors and Descriptors: Hessian corner detector, Harris Corner Detector, LOG detector, DOG detector. (Text3: Chapters 1, 4, 5, 8)(Text4: Chapter 4)	
UNIT – V	**07 Hours**
Stereo vision, Epipolar Geometry, Rectification and Issues related to Stereo, SIFT, PCA-SIFT, SURF, HOG, and Image segmentation. (Text3: Chapters 11, 15) (Text4: Chapters 5, 8, 12)	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Comprehend different image representations such as grayscale, RGB and HSV	L2
2	Apply Image Enhancement Techniques including histogram equalization, contrast stretching and sharpening filters	L3
3	Implement spatial domain operations like convolution smoothing filters and edge detection	L4
4	Apply image segmentation techniques like thresholding, region growing and edge-based segmentation	L3
5	Evaluate real-world applications including medical imaging, satellite imagery, face recognition and video processing	L5

Mapping Levels of COs to POs / PSOs														
				Program Outcomes (POs)									PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
Cos	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	2	2											2	2
CO2	3	2	2		2								2	1
CO3	2	2		2									2	2
CO4	3	2		2	2								2	1
CO5	3	2	2		2				2				2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Richard Szeliski, "Computer Vision: Algorithms and Applications", Springer, 2nd Edition, 2022.
2. Manas Kamal Bhuyan, "Computer vision and Image Processing Fundamentals and Applications", © 2020 by Taylor & Francis Group.

REFERENCE BOOKS:

1. Tekalp A.M., Digital Video Processing, Prentice Hall (1995).
2. Simon Prince, Computer Vision: Models, Learning, and Inference, 2012.
3. Gonzalez, R.C., and Woods, R.E., Digital Image Processing. 4th edition. Pearson Education (2017).
4. Jain A.K., Fundamentals of Digital Image Processing, Prentice Hall (2015).
5. David Forsyth, Jean Ponce, Computer Vision: A Modern Approach, 2015.

E-Resources:

1. Image Processing/Open CV| Udemy
2. Introduction to Computer Vision and Image Processing | Coursera

Activity-Based Learning (Suggested Activities in Class)

1. The Applications of Computer vision using group discussion.
2. Collaborative Activity is minor project development with a team of 4 students.

****

INNOVATION AND ENTREPRENEURSHIP

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Subject Code	: 25AM3505	**Credits**	: 02
Hours / Week	: 02 Hours	**Total Hours**	: 26 Hours
L–T–P–S	: 2–0–0–0		

Course Learning Objectives:

This course will enable students:

1. Identify and analyze the factors that contribute to the process of successfully launching an entrepreneurial venture and managing a new business.
2. Learn the entrepreneurial process from idea generation to implementation.
3. Acquaint with special problems of starting new ventures, finding products and services, which can support new enterprises, and raising capital.
4. Discuss how to start own business and also to work in or with small business or are involved with entrepreneurship.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainmentof the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every ***concept can be applied to the real world*** - and when that's possible,it helps improve the student's understanding.

UNIT – I	**08 Hours**

Overview Of Entrepreneurship: The Entrepreneurial Perspective:

Nature and Development of Entrepreneurship. Defining Manager, Entrepreneur, Entrepreneurship and Entrepreneurship. Key Elements of Entrepreneurship. Personality Characteristics of Successful Entrepreneurs. Common Myths about Entrepreneurs. Ethics and Social Responsibility of Entrepreneurs. Types of Start-Up Firms. Process of New Ventu2re7 Creation. Role of Entrepreneurship in Economic Development. Emerging Trends and Issues in Entrepreneurship.

Case Study: Successful Entrepreneurs Narayana Murthy Infosys

UNIT – II	**08 Hours**

THE ENTREPRENEURIAL AND ENTREPRENEURIAL MIND:

The Entrepreneurial Process: Identify and Evaluate the Opportunity, Develop a Business Plan, Determine the Resources Required, Manage the Enterprise. Managerial Versus Entrepreneurial Decision Making: Strategic Orientation, Commitment to Opportunity, Commitment of Resources, Control of Resources, Management Structure, Entrepreneurial Venturing inside a Corporation, Causes for Interest in Entrepreneurship, Climate for Entrepreneurship, Entrepreneurial Leadership Characteristics.

Case study: How to develop effective Business Plan

UNIT – III	08 Hours
CREATIVITY AND BUSINESS IDEA: Identify and Recognizing Opportunities: Observing Trends and Solving Problems. Creativity: Concept,Components and Types of Creativity, Stages of Creative Process. Sources of New Venture Ideas.Techniques for Generating Ideas. Stages of Analyzing and Selecting the Best Ideas. Protecting the Idea:Intellectual Property Rights and its Components. Linking Creativity, Innovation and Entrepreneurship. Case study: Application of Design Thinking in New business ideas generation in particular sector (Health care, Water Saving, Energy saving)	
UNIT – IV	**08 Hours**
Preparing The Proper Ethical And Legal Foundation: Initial Ethical and Legal Issues Facing a New Firm, Establishing a Strong Ethical Culture, Choosing an attorney (Lawyer), Drafting a founder's agreement, Avoiding legal disputes, Choosing a form of business organization, Obtaining business licenses and permits, Choosing a Form of Business Ownership (Sole, Proprietorship, Partnership, Corporation & Limited Liability Company) Case study: Startup Law A to Z IP https://techcrunch.com/2019/02/25/startup-law-a-to-z-intellectual-property/	
UNIT – V	**07 Hours**
Managing Early Growth And Challenges Recruiting and Selecting Key Employees. Lenders and Investors. Funding Requirements: Sources of Personal Financing. Venture Capital. Commercial Banks. Sources of Debt Financing. Key Marketing Issues for New Ventures. Why marketing is critical for Entrepreneurs. Entrepreneurs face unique Marketing Challenges. Guerrilla Marketing. Business Growth: Nature of Business Growth, Planning for Growth, Reasons for Growth. Managing Growth: Knowing and Managing the Stages of Growth, Challenges of Growing a Firm. Strategies for Firms Growth: Internal and External Growth Strategies. Implications of Growth for the Firm and Entrepreneur. Entrepreneurial Skills and Strategies to Overcome Pressures On: Financial Resources (Financial Control, Managing Inventory and Maintaining Good Records). Human Resources, Management of Employees, Time Management. Case study: 9 ways to get startups funded https://www.quicksprout.com/how-to-get-your-startup-funded/	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Demonstrate knowledge of the key elements of the entrepreneurial process	L2
2	Employ strategies to generate new ideas for startups	L2
3	Outline how to protect IP legally	L2
4	Examine different ways of generating funding	L2
5	Explain organizing managing people, finance and customers	L2

Mapping Levels of COs to POs / PSOs														
Cos				Program Outcomes (POs)									PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	2	2							2		2		2	2
CO2	3	2	2		2				2		2		2	1
CO3	2	2		2					2		2		2	2
CO4	3	2		2	2				2		2		2	1
CO5	3	2	2		2				2		2		2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Dana, Léo-Paul, ed. *World encyclopedia of entrepreneurship*. Edward Elgar Publishing, 2021.
2. Barringer, Bruce R., and R. Duane Ireland Bruce Barringer. *Entrepreneurship: Successfully launching new ventures*. Pearson Education India, 2013.

REFERENCE BOOKS:

1. Soltanifar, Mariusz, Mathew Hughes, and Lutz Göcke. *Digital entrepreneurship: Impact on business and society*. Springer Nature, 2021.
2. Aulet, Bill. *Disciplined Entrepreneurship: 24 Steps to a Successful Startup, Expanded & Updated*. John Wiley & Sons, 2024.
3. Havinal, Veerabhadrappa. Management and entrepreneurship. New Age International, 2009.
4. Janakiram, B. *Management & Entrepreneurship*. Excel Books India, 2010.

E-Resources:

1. https://archive.nptel.ac.in/courses/110/106/110106141/
2. https://www.coursera.org/mastertrack/innovation-management-entrepreneurship-hec

Activity-Based Learning (Suggested Activities in Class)

1. Organizational improvement in startup's using group discussion.

CLOUD COMPUTING
(Skill Enhancement Course - III)

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Subject Code	:	25AM3506	**Credits**	:	02
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	1–0–2–0			

Course Learning Objectives:

This course will enable students to:

1. **Understand** various basic concepts related to cloud computing technologies.
2. **Contrast** various programming models used in cloud computing.
3. **Choose** appropriate cloud model for a given application.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainmentof the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and role-playing.
3. Show ***Video/animation*** films to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. Discuss how every ***concept can be applied to the real world*** - and when that's possible,it helps improve the student's understanding.

UNIT – I	**08 Hours**

INTRODUCTION TO CLOUD COMPUTING

Introduction, Cloud Computing at a Glance, The Vision of Cloud Computing, Defining a Cloud, A Closer Look, Cloud Computing Reference Model, Characteristics and Benefits, Challenges Ahead, Historical Developments, Distributed Systems, Virtualization, Web 2.0, Service- Oriented Computing, Utility-Oriented Computing

UNIT – II	**08 Hours**

TYPES OF CLOUDS

Types of Clouds, Public Clouds, Private Clouds, Hybrid Clouds, Community Clouds, Economicsof the Cloud, Open Challenges.

Cloud Definition, Cloud Interoperability and Standards Scalability and Fault ToleranceSecurity, Trust, and Privacy Organizational Aspects.

UNIT – III	**08 Hours**

CLOUD SERVICES AND PLATFORMS

Building Cloud Computing Environments, Application Development, Infrastructure and System Development.

Computing Platforms and Technologies, Amazon Web Services (AWS), Google AppEngine, Microsoft Azure, Hadoop, Force.com and Salesforce.com, Manjrasoft Aneka.

Cloud Computing Architecture-Introduction, Cloud Reference Model- Architecture, Infrastructure / Hardware as a Service, Platform as a Service, Software as a Service.

Text Book 1: Chapter 1, Chapter 4

UNIT – IV	**08 Hours**

AWS CLOUD APPLICATION PLATFORM

Introduction to AWS ecosystem-Introduction to AWS Elastic Beanstalk-Benefits of using AWS for application development- Supported platforms on Elastic Beanstalk (Java, Python, Node.js, etc.)-Installing AWS CLI & EB CLI-Configuring AWS credentials locally-Creating Elastic Beanstalk environment-Deployment options: CLI, Console, GitHub-Lifecycle of an application in Beanstalk-Modify app configuration-Update code and redeploy changes-Explore environment dashboard-Auto-scaling and load balancing-Monitoring tools: CloudWatch metrics, logs-Health monitoring in Elastic Beanstalk-Connecting Elastic Beanstalk with RDS (Relational Database Service)-Using S3 for static asset storage-IAM roles and security basics

Text Book 1

UNIT – V	**07 Hours**

CLOUD PLATFORMS IN INDUSTRY

Concurrent Computing: Thread Programming, Introducing Parallelism for Single MachineComputation.

Cloud Platforms in Industry, Amazon Web Services, Compute Services, Storage Services, Communication Services, Additional Services, Google AppEngine, Architecture and Core Concepts, Application Life-Cycle, Cost Model, Observations, Microsoft Azure, Azure CoreConcepts, SQL Azure, Windows Azure Platform Appliance.

Text Book 1: Chapter 9

LABORATORY EXPERIMENTS

1. Introduction to Cloud and Virtualization
Objective: Understand cloud computing and install a virtualization environment.

Tools: VirtualBox or VMware

Tasks:

Install VirtualBox

Create and manage a virtual machine (e.g., Ubuntu)

Understand virtualization vs cloud

2. Setting Up a Private Cloud using Eucalyptus/OpenStack (Simulation)
Objective: Simulate a private cloud environment.

Tools: Ubuntu + Eucalyptus (or DevStack for OpenStack)

Tasks:

Install Eucalyptus/DevStack

Launch a VM instance inside the private cloud

Access the dashboard

3. Creating a Virtual Machine in AWS / Azure / GCP
Objective: Deploy a VM in a public cloud.

Tools: AWS EC2 / Azure VM / GCP Compute Engine

Tasks:

Launch an EC2 instance

Connect via SSH

Install a web server (e.g., Apache or Nginx)

4. Deploying a Web Application on AWS Elastic Beanstalk
Objective: Understand PaaS by deploying a web app.

Tools: AWS Elastic Beanstalk, Python/Node.js app

Tasks:

Initialize EB environment

Deploy app using CLI or Console

Monitor deployment logs

5. Using AWS S3 for Object Storage
Objective: Store and retrieve data from cloud storage.

Tools: AWS S3

Tasks:

Create an S3 bucket

Upload and download files

Set permissions and enable static website hosting

6. Load Balancing and Auto Scaling
Objective: Explore scalability in cloud environments.

Tools: AWS Elastic Load Balancer + Auto Scaling Groups

Tasks:

Create multiple EC2 instances

Attach to Load Balancer

Setup auto-scaling rules

7. Monitoring and Logging with CloudWatch
Objective: Monitor performance and logs of cloud resources.

Tools: AWS CloudWatch

Tasks:

Set up basic alarms (CPU usage, memory)

View application logs

Create a dashboard

8. Working with Cloud Databases (RDS / Firebase)
Objective: Use cloud-based relational or NoSQL databases.

Tools: AWS RDS or Firebase

Tasks:

Launch an RDS instance (MySQL/PostgreSQL)

Connect from a web app

Perform CRUD operations

9. Identity and Access Management (IAM)
Objective: Manage user roles and permissions in cloud.

Tools: AWS IAM

Tasks:

Create users and groups

Assign policies (S3 access, EC2 control)

Test access restrictions

10. Final Mini Project – Host a Full-Stack Application on Cloud
Objective: Integrate all concepts in a real-world mini project.

Tasks:

Backend (Node.js/Python) + Frontend (HTML/CSS/JS)

Deployed on EC2 or Beanstalk

Use S3 for static content and RDS for database

Optional Tools & Simulators
CloudSim – For simulating cloud environments in research

Cisco Packet Tracer – For network simulation in cloud labs

Google Cloud Shell – In-browser terminal for experiments

Course Outcomes:

At the end of the course the student will be able to:

1. **Analyze** the principles and components of cloud computing, demonstrating a comprehensive understanding of its architecture and operation.
2. **Compare and contrast** cloud architecture, computing platforms, and technologies.
3. **Analyze** cloud computing architecture principles to design scalable and resilient systems, utilizing various platforms and technologies.
4. **Examine** the framework, cloud deployment mode, and services in Aneka to gain a comprehensive understanding of its functionalities.
5. **Apply** web services and search engines in the industry by utilizing cloud platforms such as Amazon Web Services and Google AppEngine to leverage compute, storage, and communication services.

Cos	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
	Engineeringknowledge	Problem analysis	Design	Conduct investigations of	tool usage	The engineer andsociety	Environment and sustainability	Ethics	teamwork	Communication	Life-long learning	Project managementa	Cognitive Outcome	Skill & Design Outcome
CO1	3	2	-	-	-	-	-	-	-	-	-	1		1
CO2	3		-	2	2	-	-	-	-	-	-	1		1
CO3	2	3	2		2	-	-	-	2	-	-	2		1
CO4	3	3	2	-	-	-	-	-	-	-	-	1		1
CO5	2	2	3	-	3	-	-	-	3	-	-	2		1

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. 1. Cloud Computing: From Beginning to End", Ray J. Rafaels Rajkumar Buyya, James Broberg, and Andrzej Goscinski, Packt,2022.
2. Cloud Computing: Principles and Paradigms" by Rajkumar Buyya, James Broberg, and Andrzej Goscinski, Wiley in 2020.

REFERENCE BOOKS:

1. Cloud Computing: Concepts, Technology & Architecture" by Thomas Erl, Ricardo Puttini, and Zaigham Mahmood, Pearson Education,2020.
2. Cloud Computing A Practical Approach, Anthony T Velte et.al, MC Graw Hill publications, 2014.
3. Cloud Computing Principles and Paradigms, Rajkumar Buyya et.al, WileyPublications, 2015.

Activity Based Learning (Suggestion Activities in Class)

- Presentation
- Collaborative Activity is minor project development with a group of 4 students.

Accelerated Data Science

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER-V

Course Code	: 25AM3508	**Credits**	: 03
Hours/Week	: 03	**Total Hours**	: 45
L–T–P–J	: 2–0–0–2	**CIE+SEE**	: 60+40 Marks

Course Vision:

To empower the ability to harness modern data science techniques and GPU-accelerated computing frameworks for solving real-world problems, while promoting ethical, scalable, and efficient data-driven decision-making aligned with industry standards. The course envisions developing skilled professionals who can integrate advanced analytics, machine learning, and high-performance computing to build impactful, responsible, and future-ready AI solutions across diverse domains.

Course Objectives:

This course will enable students to:

1. To introduce students to the fundamentals of data science and the role of GPU-accelerated computing using NVIDIA RAPIDS for efficient data processing.
2. To develop the ability to perform data collection, preprocessing, and ETL operations for handling real-world datasets.
3. To enable students to understand ethical considerations, bias, and responsible practices in data science and AI systems.
4. To equip learners with skills in data analysis, visualization, and interpretation using modern tools and techniques.
5. To provide foundational knowledge of scalable computing and machine learning methods for building high-performance data-driven applications.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that a teacher can use to accelerate the attainment of the various course outcomes:

1. **Lecture method:** Along with the traditional lecture method, different types of teaching methods may be adopted to develop the course outcomes.
2. **Interactive Teaching:** incorporating brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and role-playing.
3. Showing **video/animation** films to explain the functioning of various concepts.
4. Encourage **collaborative** (group learning) learning in the class.
5. To foster critical thinking by asking higher-order thinking questions in the class in the form of quizzes and writing programs with complex solutions.
6. Showing the **different ways to solve** the same problem and encourage the students to come up with their own creative ways to solve them.

UNIT– I	10 Hours
Introduction to data science and rapids **Topics:** Accelerating End-to-End Data Science Workflows, Fundamentals of Data Science, Data Science lifecycle and workflow, Introduction to NVIDIA RAPIDS framework, GPU vs CPU computing in data science, Overview of accelerated analytics. **Applications**: Analyzing and Visualizing Large Data using Accelerated Computing - Interactive data visualization, GPU-accelerated visualization (cuDF, RAPIDS), Scalable visualization tools (hvPlot, Datashader), Dashboard development (Plotly Dash, cuXfilter), Large dataset visualization techniques, Real-time and exploratory data analysis, Applications (customer, geospatial, embeddings).	
UNIT – II	**8 Hours**
Data Collection and Preprocessing (ETL) **Topics:** Data acquisition techniques Collecting Data, Data cleaning and preprocessing Scraping Data, ETL (Extract, Transform, Load) pipeline, handling missing and noisy data SQLite as Simple, Effective Storage, Feature engineering basics SQL Refresher - Beware of Missing Indexes **Applications:** Feature scaling and normalization techniques, Dimensionality reduction (PCA, UMAP concepts), Hyper parameter tuning (grid search, optimization basics), Model performance evaluation metrics, Handling imbalanced datasets, End-to-end ML pipeline optimization.	
UNIT – III	**9 Hours**
Data Ethics and Bias in Data Sets **Topics:** Ethical issues in data science, Bias in datasets and algorithms, Fairness and accountability, Responsible AI practices, Case studies on bias mitigation, Introduction to Data Pre-processing, Data Cleaning and Statistical Preprocessing, Data Cleaners: OpenRefine and Wrangler, Feature Selection: Introduction to Filter Methods, Feature Selection: Introduction to Model-based Methods, Feature Reduction: PCA **Applications:** AI-Based Recruitment System, Healthcare Prediction System, Loan Approval and Credit Scoring, Customer Recommendation Engine, Smart Crime Analysis System.	
UNIT – IV	**10 Hours**
Data Ethics and Reducing Bias in Data Sets **Topics:** Sources of Bias and Fairness Measures, Tools for Discovering and Interpreting Bias in Models, Challenges Faced by Underrepresented Groups, Data integration techniques, Exploratory Data Analysis (EDA), Statistical analysis methods, Data visualization techniques, Tools for analytics (RAPIDS, Python libraries). **Applications:** Fair Hiring and Resume Screening Systems, Inclusive Healthcare Risk Prediction, Fair Loan and Insurance Decision Systems, Customer Behavior Analytics and Personalization, Social Media Sentiment and Bias Detection.	
UNIT – V	**08 Hours**
Scalable Computing and Machine Learning **Topics:** Scalable and distributed computing concepts, GPU-accelerated computing frameworks, Introduction to Machine Learning, Neural Networks basics, Graph analytics fundamentals **Applications:** Large-Scale Fraud Detection Systems, Real-Time Image and Video	

Recognition, Smart Traffic and Route Optimization, Social Network Graph Analysis, Predictive Maintenance in Industry.

Course Outcome

Course Outcomes (COs)	Description	Bloom'sTaxonomy Level
At the end of the course the student will be able to:		
1	Explain GPU-accelerated data processing, visualization, and machine learning concepts.	L2
2	Apply machine learning algorithms and data processing techniques for real-world problems.	L3
3	Analyze large-scale datasets, clustering methods, and forecasting models.	L4
4	Evaluate graph-based AI systems, RAG pipelines, and distributed AI applications.	L5
5	Evaluate AI system performance, security threats, and deploy scalable AI solutions.	L5

Mapping Levels of Cos to POs/PSOs

Table:MappingLevelsofCOstoPOs/PSOs

COS	ProgramOutcomes(POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO-1	3	2	1	–	2	2	–	3	–	–	2	3	3
CO-2	2	2	1	–	2	2	–	2	–	–	–	2	2
CO-3	3	2	2	–	3	3	–	3	–	–	–	3	3
CO-4	3	2	3	2	3	3	–	3	2	–	–	3	3
CO-5	3	–	2	–	3	3	–	3	2	2	2	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI Integration:

1. AI is integrated through GPU-accelerated data science workflows using tools like RAPIDS and NVIDIA Triton Inference Server. Students apply AI models such as XGBoost and cuML

for large-scale data processing, visualization, and real-time analytics. Integration focuses on handling big data efficiently and deploying AI models in production environments.

2. AI is integrated through advanced feature engineering and optimization techniques using GPU-accelerated libraries like cuDF and cuML. Students implement clustering algorithms (K-Means, DBSCAN) and dimensionality reduction methods (PCA, UMAP) to improve model performance. AI integration emphasizes building efficient ML pipelines and improving predictive accuracy for real-world datasets.
3. This module integrates AI into domain-specific applications such as time series forecasting and financial modeling. Using GPU-based tools and optimization frameworks like cuOpt, students develop AI models for forecasting trends and optimizing portfolios. AI is applied to solve real-world problems in finance and analytics with high computational efficiency.
4. AI is integrated through advanced techniques like Graph AI and Retrieval-Augmented Generation (RAG). Students work with knowledge graphs using cuGraph and build intelligent systems combining LLMs and vector databases. Frameworks like LangChain enable development of AI-powered applications such as semantic search, recommendation systems, and question-answering pipelines.
5. This module integrates AI security and robustness through adversarial machine learning. Students analyze vulnerabilities in ML and LLM systems, including attacks like data poisoning and prompt injection. AI integration focuses on developing secure, reliable, and ethical AI systems through testing, evaluation, and red-teaming methodologies.

TEXT BOOKS:

1. Aurélien Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 3rd Edition, O'Reilly Media, 2023.
2. Wes McKinney, *Python for Data Analysis: Data Wrangling with Pandas, NumPy, and Jupyter*, 3rd Edition, O'Reilly Media, 2022.

REFERENCE BOOKS:

1. Cathy O'Neil & Rachel Schutt, *Doing Data Science: Straight Talk from the Frontline*, O'Reilly Media, 2023..
2. Chip Huyen, *Designing Machine Learning Systems: An Iterative Process for Production-Ready Applications*, O'Reilly Media, 2022.

E-RESOURCES:

1. https://www.nvidia.com/en-us/training/self-paced-courses/Data-Science
2. https://www.coursera.org/browse/data-science
3. https://www.udemy.com/topic/data-science/
4. https://developer.nvidia.com/data-science-teaching-kit-syllabus

RECOMMENDED TOOLS

Python, Jupyter Notebook, RAPIDS, cuDF, cuML, cuGraph, cuPy, Dask, Pandas, NumPy, XGBoost, Scikit-learn, Triton Inference Server, Bokeh, hvPlot, Datashader, Plotly Dash, cuXfilter, NetworkX, UMAP, ARIMA/SARIMA (statsmodels), cuOpt, cuVS, Milvus, OpenAI APIs, NVIDIA NIM, BioNeMo, LangChain.

Evaluation Scheme:

Internal (100 Marks – No End Semester Exam Recommended) Component Marks

Continuous Assessment (Activities) 30

Component	Marks
Design Challenge / Assignments	20
Prototype Development	20
IP Report / Patent Search	10
Final Project + Presentation	20

ETHICS, SAFETY & GOVERNANCE OF AGENTIC AI

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Course Code	:	25AM3509	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P-J	:	2–0–0-2			

Prerequisites:

Basic understanding of Artificial Intelligence, Machine Learning concepts, probability, and introductory programming (Python/C). Familiarity with data ethics and algorithmic decision-making is desirable.

Course Objectives:

This course will enable students to:

1. Understand core ethical theories and principles relevant to the development and operation of agentic and autonomous AI systems.
2. Analyze safety risks, alignment challenges, and potential failure modes in intelligent agents deployed in real-world environments.
3. Evaluate global AI governance frameworks, regulatory policies, and international standards guiding responsible AI adoption.
4. Assess the societal, legal, and economic impacts of agentic AI across critical application domains.
5. Design ethically aligned, safe, and accountable agentic AI solutions incorporating transparency, human oversight, and governance mechanisms.

Teaching-Learning Process (General Instructions)

These pedagogical approaches are designed to accelerate the attainment of course learning outcomes and promote deep conceptual understanding.

1. Lecture Method:

Includes structured lectures integrated with concept-driven discussions supported by contemporary research articles, global policy documents, and real-world case examples of agentic AI systems.

2. Interactive Teaching:

Active learning through guided discussions, ethical dilemma analysis, and problem-solving sessions focused on safety, alignment, and decision-making challenges in autonomous AI agents.

3. Visual Learning:

Use of documentaries, expert talks, system flow visualizations, and simulation videos to illustrate AI failures, governance mechanisms, and human–AI interaction dynamics.

4. Collaborative Learning:

Group-based evaluation of AI governance frameworks, regulatory case studies, and comparative analysis of international AI policies and standards.

5. Critical Thinking:

Faculty will pose higher-order questions encouraging students to reason about ethical trade-offs, accountability, risk mitigation, and long-term implications of agentic AI deployment.

6. Application-Oriented Learning:
Continuous emphasis on applying ethical principles, safety strategies, and governance models to real-world agentic AI use cases across domains such as healthcare, transportation, and public systems.

UNIT – I	**08 Hours**
Foundations of Ethics and Agentic AI Introduction to Agentic AI and autonomous systems; historical evolution of AI ethics; normative ethical theories (deontology, utilitarianism, virtue ethics); moral agency and responsibility in AI; human–AI interaction and value alignment; overview of ethical challenges in learning, reasoning, and acting agents. **Case Study**: Autonomous Decision-Making in Healthcare Triage Systems	
UNIT – II	**08 Hours**
Safety, Risk, and Alignment in Agentic Systems AI safety fundamentals; reward hacking and specification gaming; robustness, reliability, and adversarial risks; alignment problem and value misgeneralization; interpretability and explainability for agentic models; human-in-the-loop and human-on-the-loop systems. **Case Study**: Reward Hacking in a Reinforcement Learning Trading Agent	
UNIT – III	**08 Hours**
Governance, Law, and Policy Frameworks Global AI governance landscape; OECD AI principles; UNESCO AI ethics recommendations; EU AI Act; US AI Bill of Rights; data protection laws (GDPR concepts); accountability, transparency, and auditability; standards bodies (ISO/IEC, IEEE). **Case Study**: Facial Recognition AI and Regulatory Conflict	
UNIT – IV	**08 Hours**
Societal Impact and Responsible Deployment Bias, fairness, and discrimination in agentic AI; impact on employment and labor; AI in healthcare, finance, surveillance, and defense; misinformation and autonomous decision-making risks; sustainability and environmental cost of AI systems; cross-cultural and global south perspectives. **Case Study**: Algorithmic Hiring Agent and Workforce Discrimination	
UNIT – V	**07 Hours**
Future Directions and Trustworthy Agentic AI AI assurance and certification; ethical-by-design and safety-by-design methodologies; red-teaming and impact assessments; governance of autonomous agents and multi-agent systems; future risks (superintelligence, autonomous weapons); research frontiers and global cooperation.	

Case Study: Autonomous Weapons and Human Oversight

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain ethical theories and their application to agentic AI systems.	L2
2	Analyze safety risks and alignment challenges in autonomous AI systems	L4
3	Examine global AI governance frameworks, standards, and regulations	L4
4	Assess societal impacts and ethical trade-offs of agentic AI deployment	L3
5	Analyze approaches for responsible and trustworthy governance o agentic AI	L4

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	2	1	–	–	2	–	–	–	–	–	–	2	2
CO2	3	3	2	2	–	2	–	–	–	–	–	–	2	2
CO3	2	3	2	–	–	3	2	–	–	–	–	–	2	2
CO4	2	2	3	–	–	3	2	–	–	–	–	–	1	1
CO5	3	3	3	2	–	2	–	–	–	–	–	–	1	1

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS (TB):

1. Russell, S., & Norvig, P. *Artificial Intelligence: A Modern Approach* (4th ed.). Pearson, 2021.
2. IEEE Standards Association. *Ethically Aligned Design: A Vision for Prioritizing Human Well-being with Autonomous and Intelligent Systems*. IEEE, 2019.

3. Bostrom, N. *Superintelligence: Paths, Dangers, Strategies*. Oxford University Press, 2014.

REFERENCE BOOKS:

1. O'Neil, C. Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy. Crown Publishing Group, 2016.
2. Floridi, L. The Ethics of Artificial Intelligence: Principles, Challenges, and Opportunities. Oxford University Press, 2023.
3. AI Now Institute. Annual AI Impact Report. New York University, 2022.
4. Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., & Mané, D. Concrete Problems in AI Safety. arXiv / OpenAI, 2016.
5. Mittelstadt, B., Allo, P., Taddeo, M., Wachter, S., & Floridi, L. The Ethics of Algorithms: Mapping the Debate. Big Data & Society, 2016.
6. Crawford, K. Atlas of AI: Power, Politics, and the Planetary Costs of Artificial Intelligence. Yale University Press, 2021.
7. European Commission. Ethics Guidelines for Trustworthy AI. Publications Office of the European Union, 2019.
8. OECD. Artificial Intelligence in Society. OECD Publishing, 2019.

E-Resources:

1. https://oecd.ai
2. https://www.unesco.org/en/artificial-intelligence

Activity Based Learning (Suggested Activities in Class)

1. Case study analysis of real-world AI failures.
2. Policy debate role-play (regulator, developer, civil society).
3. Ethical impact assessment of an agentic AI application.
4. Flipped classroom discussions on AI governance documents.

Evaluation Scheme :

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

Intelligent Speech Processing Systems

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER - V

Course Code	**:** 25AMS3510	**Credits**	**:** 03
Hours / Week	**:** 03	**Total Hours**	: 45(15Hrs Project)
L–T–P–J	**: 2**–0–0–2	**CIE**	**:** 100 Marks

Course Vision:

This course develops the ability to design, analyse, and evaluate intelligent speech processing and recognition systems by integrating speech signal processing, probabilistic modelling, deep learning, and responsible AI for real-world spoken language applications.

Course Objectives:

Students will:

1. Understand the fundamentals of speech production, perception, and automatic speech recognition pipelines.
2. Apply speech signal processing and feature extraction techniques for speech analysis.
3. Analyze acoustic and language modelling approaches used in conventional ASR systems.
4. Evaluate deep learning and end-to-end methods for modern speech processing applications.
5. Design intelligent speech-enabled solutions with appropriate ethical and engineering judgment.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. Lecture method along with traditional lecture method, different type of teaching methods may be adopted to develop the course outcomes.
2. Interactive Teaching: incorporating brainstorming, discussion, group work, focused listening, problem formulation, note-taking, and classroom demonstrations.
3. Showing video/animation clips and speech demos to explain speech generation, signal analysis, and recognition workflows.
4. Encourage collaborative learning through mini case studies, waveform analysis, and toolkit-based exercises.
5. To make critical thinking, asking higher order thinking questions in the class in the form of quiz, interpretation of speech outputs, and model comparison.
6. Showing different ways to solve the same speech processing problem and encourage the students to come up with creative and responsible solutions.

MODULE – I	**08 Hours**

Foundations of Intelligent Speech Processing

Topics: Introduction to speech processing and speech-based AI systems, speech production mechanism, speech perception, characteristics of speech signals, speech representation, stages of intelligent speech processing systems, and challenges in speech understanding.

Applications: Voice assistants, spoken dialogue systems, dictation systems, and assistive communication tools.

MODULE – II	**09 Hours**

Speech Signal Processing and Feature Extraction

Topics: Sampling and quantization, pre-emphasis, framing, windowing, short-time analysis, time-frequency representation, spectrogram analysis, feature extraction methods such as MFCC, LPC, PLP, and filter-bank energies.

Applications: Keyword spotting, speaker interfaces, embedded speech systems, and robust front-end design.

MODULE – III	08 Hours
Acoustic and Language Modelling **Topics:** Acoustic modelling concepts, Hidden Markov Models, Gaussian Mixture Models, Baum-Welch training, Viterbi decoding, pronunciation lexicons, N-gram language models, and decoding strategies in conventional ASR systems. **Applications:** IVR systems, command-and-control interfaces, transcription engines, and speech analytics pipelines.	
MODULE – IV	**10 Hours**
Deep Learning for Speech Processing **Topics:** Limitations of traditional ASR, deep neural networks for speech, Transformer concepts in speech processing, self-supervised learning, end-to-end ASR, CTC, and attention-based speech models. **Applications:** Real-time transcription, multilingual speech interfaces, mobile ASR, and cloud speech services.	
MODULE – V	**10 Hours**
Evaluation, Multilingual Speech Applications, and Emerging Trends **Topics:** Speaker-dependent and speaker-independent systems, noise-robust speech recognition, multilingual and low-resource speech processing, evaluation metrics such as WER, SER, and CER, speech APIs and toolkits, ethics, privacy, fairness, and recent research trends. **Applications:** Healthcare transcription, customer support automation, educational technology, accessibility systems, and domain-specific spoken language solutions.	

AI Integration:

- AI tools for waveform, spectrogram, and speech feature understanding; AI-assisted explanation of speech pipelines.
- Classical ASR modelling support using AI for comparing HMM, GMM, lexicon, and language model workflows.
- Deep learning based speech models, speech embeddings, and transformer-based ASR prototyping tools.
- Pretrained speech APIs, toolkit benchmarking environments, and multilingual speech processing platforms.
- Evaluation of intelligent speech systems using standard datasets, recognition metrics, and responsible AI validation.

Authentic Intelligence:

- Limitations of machine speech perception compared with human auditory perception and context understanding.
- Bias due to accents, languages, dialects, and data imbalance in speech datasets.
- Challenges in noisy speech, multilingual speech processing, and over-reliance on AI-generated modelling suggestions.
- Ethical concerns in speech technologies including privacy, inclusion, fairness, transparency, and deployment risks.
- Students evaluate societal impact, data sensitivity, and practical trustworthiness of intelligent speech systems.

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Understand the fundamentals of speech production, perception, and speech signal characteristics.	L2
2	Apply speech signal processing and feature extraction techniques for speech analysis.	L3
3	Analyze acoustic and language modelling approaches used in conventional ASR systems.	L4
4	Evaluate deep learning and end-to-end speech processing models for recognition tasks.	L5
5	Design intelligent speech-enabled applications for multilingual and real-world environments.	L5

Mapping Levels of COs to POs / PSOs

Table: Mapping Levels of COs to POs / PSOs

COS	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO-1	3	2	1	–	1	–	–	–	–	–	1	1	2
CO-2	3	3	2	2	2	–	2	2	–	1	1	2	3
CO-3	3	3	3	2	2	–	1	1	–	1	1	2	3
CO-4	3	2	3	3	3	–	2	2	–	2	2	3	3
CO-5	3	3	3	3	3	1	1	–	–	2	2	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Jurafsky, D., & Martin, J. H., Speech and Language Processing, 3rd Edition, Pearson,
2. Rabiner, L. R., & Juang, B.-H., Fundamentals of Speech Recognition, Pearson, year..
3. Quatieri, T. F., Discrete-Time Speech Signal Processing, 3rd Edition, Pearson.

REFERENCE BOOKS:

1. Yu, D., & Deng, L., Automatic Speech Recognition: A Deep Learning Approach, Springer.
2. Deng, L., & Yu, D., Deep Learning for Speech Recognition and Language Processing, Springer.
3. O'Shaughnessy, D., Speech Communications: Human and Machine, 2nd Edition, Wiley-IEEE Press.

E-RESOURCES:

1. Research papers from IEEE, Springer, Elsevier, and ACL Anthology related to speech processing and ASR.
2. NPTEL / SWAYAM courses on Speech Processing, Pattern Recognition, and Machine Learning.
3. Official documentation of Librosa, Kaldi, SpeechBrain, torchaudio, and Hugging Face speech models.

RECOMMENDED TOOLS

- Programming: Python, Jupyter Notebook
- Libraries: Librosa, torchaudio, SpeechBrain, Kaldi, Hugging Face Transformers
- Data Processing: NumPy, SciPy, pandas
- Visualization: Matplotlib, waveform and spectrogram analysis tools
- AI Tools: ChatGPT, code assistants, speech API sandboxes, and model benchmarking platforms

Evaluation Scheme :

Criteria	**Weightage of Marks**
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

****

FOUNDATIONS OF ROBOTICS AND INTELLIGENT SYSTEMS

Professional Elective

Subject Code	:	25AM3511	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 Hours
L–T–P–S	:	2–0–0–2			

Course Learning Objectives:

Students will:

- Understand the principles and capabilities of foundation models in AI systems
- Analyze limitations of traditional robotics pipelines and the need for integrated intelligence
- Apply language and vision models for robotic perception and interaction
- Examine how foundation models enable reasoning and decision-making
- Design robotic applications integrating multimodal AI models
- Evaluate reliability, safety, and ethical implications of AI-driven robotic systems

Course Description :

To enable students to understand and apply foundation models (LLMs, VLMs, and Vision-Language-Action models) in robotic systems for perception, reasoning, and task-level decision-making, representing a paradigm shift from traditional modular pipelines to integrated intelligent systems.

This course emphasizes how foundation models transform robotic autonomy by enabling language understanding, cross-modal reasoning, and generalization across environments, moving beyond rigid perception–planning–control architectures toward flexible, intelligent robotic behavior.

Course Outcome	Description	Bloom's Taxonomy Level
CO1	Analyze and describe the principles and role of foundation models in robotics	L4
CO2	Apply language and vision models for perception and interaction tasks	L3
CO3	Analyze and utilize foundation models for reasoning and decision-making in robotics	L4
CO4	Develop and integrate foundation models into robotic subsystems	L5
CO5	Design and implement a complete multimodal robotic system	L6

UNIT – I : Foundations of Robotics Paradigm Shift	**09 Hours**

Evolution of robotics systems, modular vs integrated architectures, introduction to foundation models (LLMs, VLMs, VLAs), representation learning, limitations of classical robotics Applications: Natural language-based robot control, generalization across environments	
UNIT – II : Language and Vision Models for Robotics	**09 Hours**
Vision-Language Models (VLMs), multimodal embeddings, grounding language in perception, semantic scene understanding, perception-language alignment, linking perception outputs to actionable representations for robotic tasks. Applications: Language-guided object detection, scene interpretation, robot perception enhancement.	
UNIT – III : Reasoning and Decision-Making with Grounded Robotics Actions	**09 Hours**
Task decomposition, instruction following, prompt-based planning, reasoning strategies, grounding reasoning into executable robotic actions, decision-making under uncertainty, action abstraction and task-level planning. Applications: High-level task planning with executable robot actions, instruction-driven embodied agents.	
UNIT – IV : Robotics Applications using Foundation Models	**09 Hours**
Overview of applications including manipulation using vision-language-action models, semantic navigation, human-robot interaction, learning from demonstration, action execution constraints, embodied decision-making, case studies (RT-2, SayCan, PaLM-E) Applications: Language-driven manipulation, context-aware navigation, assistive and service robotics	
UNIT – V : System Integration, Evaluation, and Future Robotics Intelligence	**09 Hours**
System architectures for AI-driven robotics, ROS integration with foundation models, deployment challenges (latency, compute), evaluation of foundation model performance in robotics, failure case analysis (hallucination, grounding errors, safety risks), future trends ***Applications:*** End-to-end robotic systems, cloud vs edge AI in robotics	

Mapping Levels of COs to POs / PSOs					
COs				**Program Outcomes (POs)**	**PSOs**

	1	2	3	4	5	6	7	8	9	10	11	1	2
CO1	3	3	1	1	1	-	1	-	-	-	3	3	1
CO2	3	2	2	2	3	-	1	1	1	-	2	3	3
CO3	3	3	2	2	2	-	2	1	1	-	3	3	2
CO4	2	2	3	2	3	-	1	2	1	1	2	2	3
CO5	2	3	3	2	3	-	1	2	1	1	3	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI Integration & Authentic Intelligence Reflection (AI^2 Pedagogy)

AI Integration:

Module-1: Use of pretrained AI models in robotics workflows, API-based model access

Module-2: Integration of VLMs with robotic perception pipelines, transforming sensor data into actionable representations.

Module-3: LLM-based planners, prompt engineering for robotics tasks, integration of reasoning outputs with robotic execution pipelines.

Module-4: Combining perception, reasoning, and action in embodied systems

Module-5: Model orchestration, system-level integration strategies

Authentic Intelligence Reflection:

Module-1: Model bias, lack of interpretability, over-reliance on pretrained systems

Module-2: Visual misinterpretation, dataset bias, perception uncertainty, and challenges in grounding perception to real-world actions.

Module-3: Hallucination risks, mismatch between reasoning and physical execution, reliability and safety concerns in embodied decision-making.

Module-4: Real-world uncertainty, safety in human environments, embodiment limitations

Module-5: Accountability, trust, and real-world failure consequences

Mini Project (Mandatory)

Students will design a robotics system utilizing foundation models.

Examples:

- Language-guided object manipulation in simulation
- Vision-language-based navigation system
- Instruction-following robotic assistant

Mandatory Requirement:

All projects must demonstrate grounding of model outputs into robotic actions (simulation or real-world).

Students must analyze:

- System design and architecture
- AI model integration
- Task reasoning capability
- Robustness and limitations
- Ethical and safety considerations

Evaluation Focus

- System Design Clarity
- Level of Autonomy Achieved
- Grounding Fidelity
- Robustness Under Uncertainty
- Failure Handling and Recovery

Recommended Tools

- Python
- OpenAI / HuggingFace APIs
- Integration with ROS-based pipelines (optional but strongly encouraged)
- Simulation tools (Gazebo, PyBullet, Isaac Sim)

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

AI Tool Usage for Teaching and Learning

AI tools will be used in this course as structured pedagogical support to enhance understanding, system design, and problem-solving.

Instructor Use (AI as Co-Teacher)

- Generate alternative explanations, examples, and case studies
- Demonstrate system workflows, debugging strategies, and design variations

Student Use (AI as Teaching Assistant)

- Concept clarification and understanding
- Code assistance and debugging (with explanation)
- System design exploration and documentation support

Guidelines for Use

- All AI-assisted work must be understood and explainable by the student

- Blind use of AI-generated content is not acceptable
- Students must critically evaluate AI outputs and identify errors or limitations

Evaluation Integration

- Viva and project discussions will assess individual understanding
- Students may be asked to explain or modify AI-assisted solutions

Suggested Reading

1. Russell, S., Norvig, P., Artificial Intelligence: A Modern Approach, 4th Edition, Pearson, 2020
2. Goodfellow, I., Bengio, Y., Courville, A., Deep Learning, MIT Press, 2016
3. Thrun, S., Burgard, W., Fox, D., Probabilistic Robotics, MIT Press, 2005

Research Papers:

1. Recent survey papers on foundation models in robotics (e.g., embodied AI surveys)
2. Google RT-2: Vision-Language-Action Models
3. SayCan: Language to Robot Action Grounding
4. PaLM-E: Embodied Multimodal Models
5. Recent publications from ICRA, RSS, CoRL (Post-2020)

****

Explainable AI

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER - V

Course Code	:	25AM3512	**Credits**	:	03
Hours / Week	:	03	**Total Hours**	:	45
L–T–P–J	:	2–0–0–2	**CIE+SEE**	:	60+40 Marks

Course Vision:

To enable learners to develop transparent, interpretable, and trustworthy AI systems by understanding and applying Explainable AI techniques in real-world engineering applications.

Course Objectives:

This course will enable students to:

1. To understand the fundamental concepts, need, and challenges of Explainable AI.
2. To explore various taxonomy of XAI methods, including model-agnostic and model-specific approaches.
3. To study and apply interpretable machine learning models for transparent decision-making.
4. To analyze explainability techniques in deep learning models using visualization and attribution methods.
5. To implement and evaluate XAI techniques using tools and real-world case studies.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. Lecture method along with traditional lecture method, different type of teaching methods may be adopted to develop the course outcomes.
2. Interactive Teaching: incorporating brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Showing Video/animation films to explain functioning of various concepts.
4. Encourage Collaborative (Group Learning) Learning in the class.
5. To make Critical thinking, asking Higher order Thinking questions in the class in the class in the form of Quiz and writing programs with complex solutions.
6. Showing different ways to solve the same problem and encouraging the students to come up with their own creative ways to solve them

MODULE – I : Introduction to Explainable AI	**09 Hours**

Black Box Problem in AI, History of AI & Explainability, Need for Explainability: Trust, Transparency, Accountability, Definitions, Goals, and Scope of XAI, Science of Interpretable Machine Learning, Motivation, Challenges, and Trade-offs of Explainable AI, Human Factors in Explainability, Interpreting Interpretability.

Engineering Applications: Healthcare diagnosis systems, transparent credit scoring in financial systems, AI-based recommendation systems with user trust

MODULE – II: Advanced Explainability Methods	**09 Hours**

Model-Agnostic Approaches: LIME, SHAP, Integrated Gradients. Model-Specific Approaches: Rule-based Explanation, Tree Feature Importance. What-if Analysis: Counterfactual Explanations, Reasoning-based: Causal Inference and User-facing: Interactive Explanations and Visualizations.

Engineering Applications: Explainable Credit Scoring System (Finance), Healthcare Diagnosis Support System,Counterfactual Explanation in Insurance Systems

MODULE – III: Interpretable vs Explainable Algorithms	09 Hours
Interpretable vs. Explainable Algorithms, Ensemble-Based-Boosted Rulesets, Explainable, Boosting Machines (EBM), RuleFit, Skope-Rules, Iterative Random Forests (iRF), Decision Tree, Based-Optimal Classification Trees, Optimal Decision Trees, Scoring System Engineering Applications: High-stakes decision systems like healthcare, finance, and risk assessment	
MODULE – IV:Applications and Tools for Explainable AI	**09 Hours**
Applications, Tools and Libraries, Intrinsic, Perturbation- LIME, Occlusion, RISE, Prediction, Difference Analysis, Meaningful Perturbation, Maximization, Class Model Visualization, Saliency Maps, DeepLIFT, DeepSHAP, Deconvolution, Guided Backpropagation, Integrated Gradients, Layer-Wise Relevance Propagation,Excitation,CAM. **Engineering Applications: Computer vision, medical imaging, and autonomous systems**	
MODULE – V:Practical Implementation and Evaluation of Explainable AI	**09 Hours**
Hands-on implementation of XAI libraries (e.g., LIME, SHAP). Explainability metrics and evaluation methods. Trade-offs between model performance and explainability. Case studies of XAI applications in real-world scenarios. Engineering Applications: Applied in real-world AI deployment and model validation systems	

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain the importance of interpretability and transparency in AI systems.	L2
2	Differentiate between various XAI methods such as model-agnostic, model-specific, and causal approaches.	L4
3	Apply interpretable machine learning models for solving real-world problems	L3
4	Analyze and interpret deep learning models using advanced XAI techniques	L4
5	Evaluate and implement explainable AI solutions considering performance and interpretability trade-offs.	L5

Mapping Levels of COs to POs / PSOs

Mapping Levels of COs to POs / PSOs					
COs				Program Outcomes (POs)	PSOs

	1	2	3	4	5	6	7	8	9	10	11	1	2
CO1	2	2	-	-	1	-	-	2	-	-	-	1	2
CO2	2	3	-	2	1	-	-	2	-	-	-	2	2
CO3	3	3	2	2	2	-	-	1	-	-	-	3	3
CO4	3	3	2	3	2	-	-	1	-	-	-	3	3
CO5	3	3	3	3	2	1	1	3	1	1	1	3	3

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

PSO1: Design and build systems using analytical, logical, and problem-solving skills.
PSO2: Develop intelligent/AI-enabled systems for real-world applications.

AI INTEGRATION & AUTHENTIC INTELLIGENCE REFLECTION:

- **AI Integration:**

1. Understanding transparency and interpretability in AI-driven decision systems.
2. Application of model-agnostic and model-specific XAI techniques for explaining machine learning models.
3. Use of interpretable machine learning models for transparent prediction and decision-making.
4. Explainability techniques for deep learning models using visualization and attribution methods.
5. Evaluation, deployment, and ethical integration of explainable AI in real-world systems.

- **Authentic Intelligence Reflection:**

1. Reflect on the importance of transparency and trust in AI systems and their impact on human decision-making.
2. Analyze how different explanation techniques influence understanding, fairness, and accountability of AI models.
3. Evaluate the balance between model accuracy and interpretability in real-world problem-solving scenarios.
4. Interpret deep learning model behaviors using visualization techniques and reflect on their reliability
5. Assess ethical implications, biases, and societal impact of deploying explainable AI systems.

TEXT BOOKS:

1. Samek, W., Montavon, G., Vedaldi, A., Hansen, L. K., & Müller, K.-R. (2019). Explainable AI: Interpreting, Explaining and Visualizing Deep Learning. Springer.
2. Molnar, C. (2020). Interpretable Machine Learning: A Guide for Making Black Box Models Explainable. Independently published.

REFERENCE BOOKS:

1. Raschka, S., & Mirjalili, V. (2019). Python Machine Learning: Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow 2 (3rd ed.). Packt Publishing.
2. Mayuri Mehta, Vasile Palade, Indranath Chatterjee, "Explainable AI: Foundations, Methodologies and Applications", Springer, 2023.

E-RESOURCES:

1. https://nvdam.widen.net/s/wlbgbqr7cj/nvidia-learning-training-course-catalog
2. https://www.coursera.org/learn/explainable-ai-xai

RECOMMENDED TOOLS

- Python, OpenCV
- TensorFlow / PyTorch
- Scikit-learn
- Jupyter Notebook

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment “Or” Mini Project*	50
Total	**100 marks**

COGNITIVE LINGUISTICS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Course Code	:	25AM3513	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45
L–T–P-J	:	2–0–0-2			

Course Vision

Course Objectives:

1. To introduce students to the historical development, core assumptions, and main schools of thought within cognitive linguistics as an alternative to formal models of language.
2. To develop students' understanding of key theoretical frameworks in cognitive linguistics, including frame semantics, construction grammar, conceptual metaphor theory, and prototype theory.
3. To enable students to analyse linguistic data using usage-based, functional, and cognitive approaches, linking grammar to cognition, categorization, and language use.
4. To train students to critically read, interpret, and discuss primary scholarly works in cognitive linguistics, fostering independent thinking and methodological awareness.
5. To cultivate students' ability to design and present small-scale research projects in cognitive linguistics, culminating in oral presentations and a written term paper.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture methods, but different *types of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note taking, annotating, and roleplaying.
3. Show ***Video/Animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.

6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

MODULE– I	**09 Hours**
INTRODUCTION AND ORIGINS OF COGNITIVE LINGUISTICS: Cognitive linguistics and other cognitive sciences. Formal, functional and cognitive linguistics; historical development and core assumptions.	
MODULE – II	**09 Hours**
Grammar, Cognition and Usage-Based Model: Grammar and cognition; Talmy's cognitive representations. Language as human behaviour, language acquisition, language as a complex adaptive system (usage-based model).	
MODULE – III	**09 Hours**
Frame Semantics and Construction Grammar: Frame semantics: frames and Fillmore's case grammar. Construction grammar: constructions as elements of grammar; Fillmore, Langacker; construction grammar vs cognitive grammar.	
MODULE – IV	**09 Hours**
Conceptual Metaphor and Conceptual Blending: Lakoff & Johnson: metaphor as a cognitive mechanism. Fauconnier: conceptual blending and its applications in meaning construction.	
MODULE – V	**09 Hours**
Prototype Theory, Typology and Applications: Prototype theory: Rosch, Berlin & Kay; prototype and basic-level categories; applications in psychology, psychiatry, cultural anthropology. Typology and cognitive linguistics: Corbett, language universals, future directions; student project presentations as integrative activity.	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Understand the historical development and core assumptions of cognitive linguistics and distinguish it from formal and functional approaches to language.	**L2**
2	Apply concepts of grammar–cognition interfaces and usage-based models to analyse how language structure emerges from use and cognitive processes.	**L3**
3	Analyze linguistic data using frame semantics and construction grammar to identify how meaning and grammatical patterns are organised in speakers' minds.	**L4**
4	Analyze conceptual metaphors and conceptual blending in authentic language examples to reveal underlying cognitive mechanisms of meaning construction.	**L4**
5	Evaluate and design small-scale studies that use prototype theory and typological perspectives to explain categorization and linguistic universals in cognitive linguistics.	**L5**

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
CO1	2	1	0	0	0	1	0	1	0	1	0	2	1	1
CO2	2	2	1	1	1	1	0	1	0	1	0	2	1	2
CO3	2	3	1	2	1	1	0	1	1	2	1	2	2	2
CO4	1	2	1	2	1	2	1	2	1	3	1	2	2	2
CO5	2	3	2	3	2	2	1	2	2	3	2	3	2	3

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

- CO3 and CO5 have higher mapping to PO2 and PO4 because students analyse data and synthesize conclusions from research-like tasks and term projects.
- CO4 and CO5 map more strongly to PO10 and PO12 due to presentations, term papers, and emphasis on continuous engagement with scholarly literature.

TEXTBOOKS (TB):

1. Evans, Vyvyan, & Green, Melanie. Cognitive Linguistics.
   - A comprehensive, up-to-date general introduction to cognitive linguistics covering cognitive semantics, construction grammar, metaphor, blending, and comparisons with formal approaches.
   - Intended as a primary textbook for undergraduate and postgraduate courses, with clear structure aligning well with units on frames, constructions, metaphor, and typology.
2. Dąbrowska, Ewa, & Divjak, Dagmar (eds.). Handbook of Cognitive Linguistics. De Gruyter, 2015.
   - State-of-the-art handbook with chapters on usage-based grammar, frame semantics, construction grammar, metaphor, prototypes, and typology by leading scholars.researchportal.northumbria+1
   - Suitable as an advanced text to support student projects and term papers, giving research-level depth beyond the core textbook.
3. Dąbrowska, Ewa (ed.). The Cambridge Handbook of Cognitive Linguistics. Cambridge University Press, 2017.
   - Provides a thorough overview of methods, key concepts, and new subfields in cognitive linguistics, including construction grammar, conceptual mappings, acquisition, corpus studies, and diachrony.
   - Well-suited as a reference and complementary reading for units on grammar–cognition, metaphor/blending, and prototype theory.

REFERENCE BOOKS:

- Tomasello, Michael. A Natural History of Human Thinking. Harvard University Press, 2014 (often used post-2015).
- Bybee, Joan. Frequency of Use and the Organization of Language. Oxford University Press, 2006 (still a key reference in 2015–2025 syllabi).

E-Resources:

NPTEL courses

- **Language, Culture and Cognition: An Introduction (IIT Guwahati)** – Introduces relations between language, cognition and culture explicitly from a cognitive linguistics perspective; videos and materials are free, exam is paid only if needed.
- **Bilingualism: A Cognitive and Psycholinguistic Perspective (NPTEL)** – Explores cognitive and psycholinguistic aspects of bilingualism, useful for units on cognition and categorisation.onlinecourses.
- **Applied Linguistics (NPTEL)** – Focus on applications of linguistic theory; relevant for connecting cognitive and applied perspectives.
- **Introduction to Cognitive Psychology / Cognition and its Computation / Introduction to Basic Cognitive Processes (NPTEL)** – Background on cognitive processes, attention, memory and computation supporting cognitive linguistics foundations.

Audit-mode MOOCs

- **Coursera – Linguistics & Cognitive Science courses** – Several linguistics and cognitive science courses can be audited free (access to videos and readings; certificate is paid).
- **FutureLearn – Free Linguistics Courses** – Short, fully online linguistics courses (often free for the duration of the course) that cover language, communication and sometimes language & mind topics.
- **Alison – Cognitive Processing and Language Acquisition** – Free course on cognitive processing and language acquisition, useful as a supplementary psycholinguistics resource.

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving using group discussion.
2. Role play E.g., Stack, Queue, etc.,
3. Demonstration of solution to a problem through programming.
4. Flip class activity E.g., arrays, pointers, dynamic memory allocation, etc.,

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

VI SEM – ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

SL	Course Type	Course Code	Course Name	Teaching Hours / Week				Examination			
				Lecture	Tutorial	Practical	Project	CIE Marks	SEE Marks	Total Marks	Credit
				L	T	P	J				
1	IPCC	25AM3601	Agentic AI	3	0	2	0	60	40	100	03
2	IPCC	25AM3602	Gen AI and Prompt Engineering	3	0	2	0	60	40	100	04
3	PCC	25AM3603	Designing MLOps for Enterprises	3	0	0	0	60	40	100	03
4	PEC	25AM36XX	Professional Elective Course – II	3	0	0	0	100	--	100	03
5	PEC	25AM36XX	Professional Elective Course – III	3	0	0	0	100	--	100	03
6	OEC	25OEXXXX	Open Elective – I	3	0	0	0	100	--	100	03
7	PROJ	25AM3604	Minor Project	0	0	0	4	--	100	100	02
8	AEC	25AM3605	Cognitive and Technical Skills-VI	-	-	-	-	--	--	--	P/F
			Total	18	0	4	4				**22**

IPCC-Integrated Professional Core Course, AEC-Ability Enhancement Course,PCC-Professional Core Courses, PEC-Professional Elective Courses, OEC-Open Elective Courses, PROJ-Project Work, L – Lecture, T – Tutorial, P – Practical, J– Project, C – No. Of Credits

Professional Elective Courses –II (PEC-II) Offering										
Sl. No	Course Type	Course Name	Lecture	Tutoria	Practic	Project	CIE Marks	SEE Marks	Total Marks	CREDITS
			L	T	P	J				
1	PEC	Accelerated Computing	2	0	0	2	100	--	100	03
2	PEC	LLM Based Agent Frameworks	2	0	0	2	100	--	100	03
3	PEC	Visual Computing and Graphics	2	0	0	2	100	--	100	03
4	PEC	Robot Operating Systems (ROS)	2	0	0	2	100	--	100	03
5	PEC	AI for Autonomous Vehicles & Smart Mobility	2	0	0	2	100	--	100	03
6	PEC	Conversational AI and spoken Language Understanding	2	0	0	2	100	--	100	03

Professional Elective Courses –III (PEC-III) Offering										
Sl. No	Course Type	Course Name	Lectur	Tutori	Practi	Projec	CIE Marks	SEE Marks	Total Marks	CREDITS
			L	T	P	J				
1	PEC	Edge AI and Robotics	2	0	0	2	100	--	100	03
2	PEC	Multi agent AI Systems	2	0	0	2	100	--	100	03
3	PEC	Intelligent Computational Imaging and Reconstruction	2	0	0	2	100	--	100	03
4	PEC	Autonomous Robotic Systems	2	0	0	2	100	--	100	03
5	PEC	AI for Healthcare	2	0	0	2	100	--	100	03
6	PEC	Building AI Applications with Large Language Models	2	0	0	2	100	--	100	03

Open Elective-1 (OE-1) Offering										
Sl. No	**Course Type**	**Course Name**	**Lecture**	**Tutorial**	**Practica**	**Project**	**CIE Marks**	**SEE Marks**	**Total Marks**	
			L	**T**	**P**	**J**				
1	OEC	Industrial Robotics	2	0	0	2	100	--	100	03
2	OEC	Multimodal and Multilingual Language Perception	2	0	0	2	100	--	100	03
3	OEC	AI for Autonomous Vehicles and Smart Mobility	2	0	0	2	100	--	100	03
4	OEC	AI For Content Generation	2	0	0	2	100	--	100	03

AGENTIC AI

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER - VI

Course Code	:	25AM3601	**Credits**	:	04
Hours / Week	:	05	**Total Hours**	:	45+30
L–T–P–J	:	3–0–2–0	**CIE+SEE**	:	60+40 Marks

Course Vision:

To cultivate engineers capable of designing, implementing, and responsibly deploying AI agents that reason, plan, use tools, and collaborate — integrating modern agentic frameworks, NVIDIA DLI certified learning pathways, and sound judgment about human oversight in autonomous systems.

Course Objectives:

This course will enable students to:

6. To understand the foundations of agentic AI: agent taxonomies, perception-reasoning-action loops, and environments.
7. To apply reasoning patterns (ReAct, Reflexion, Chain-of-Thought) and build self-correcting memory-augmented agents.
8. To implement tool use, retrieval-augmented generation, and agent orchestration with LangChain, LangGraph, CrewAI.
9. To design multi-agent systems with role specialization, delegation, and supervisor-worker architectures.
10. To evaluate agent safety, alignment, and production deployment with human-in-the-loop oversight and benchmarking.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. **Lecture method** along with traditional lecture method, different type of teaching methods may be adopted to develop the course outcomes.
2. **Interactive Teaching:** incorporating brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Showing **Video/animation** films to explain functioning of various concepts.
4. Encourage **Collaborative** (Group Learning) Learning in the class.
5. To make **Critical thinking**, asking Higher order Thinking questions in the class in the form of Quiz and writing programs with complex solutions.
6. Showing the **different ways to solve** the same problem and encourage the students to come up with their own creative ways to solve them

MODULE – I	**09 Hours**

Foundations of Agentic AI

Topics: Evolution from rule-based systems to LLM-powered agents; defining agency (autonomy, reactivity, proactivity, social ability); perception-reasoning-action loop; environment types (observable, deterministic, static/dynamic); agent taxonomies (reflex, model-based, goal-based, utility-based, learning); structured outputs and JSON mode for reliable tool interfaces; local LLM inference setup (complemented by DLI for NIM-specific deployment).

Applications: Virtual assistants, autonomous coding agents, customer support bots, educational tutoring systems.

MODULE – II	**09 Hours**

Agent Architectures & Reasoning Patterns

Topics: Advanced reasoning frameworks: ReAct (Reasoning + Acting) implementation from scratch; Chain-of-Thought and Tree-of-Thought prompting; Reflexion and self-critique loops —

agents evaluating their own outputs; memory systems (short-term working, long-term vector stores, episodic); state and context window management; cognitive architectures (BDI model, ACT-R parallels in LLM agents); code generation and self-debugging agent patterns. (Prompt engineering fundamentals, LangChain LCEL, structured outputs, and basic tool invocation are covered in depth via the 8-hour DLI course.) **Applications:** Research assistants, code generation and review agents, document analysis pipelines, automated debugging tools.	
MODULE – III	**09 Hours**
Tool Use, RAG & Orchestration **Topics:** Production-grade patterns beyond the DLI course: error handling and retry strategies; circuit breakers for unreliable tools; advanced structured output schemas with Pydantic; function and tool calling API patterns across providers (OpenAI, Anthropic, open-source); RAG evaluation discussion — when retrieval helps vs. hurts. (RAG architecture fundamentals, vector databases, LangGraph state machines, dialog management, and base RAG evaluation metrics are covered in depth via the 8-hour DLI RAG Agents course.) **Applications:** Enterprise search agents, data analysis pipelines with tool chains, automated report generation, domain-specific Q&A systems.	
MODULE – IV	**09 Hours**
Multi-Agent Systems & Collaboration **Topics:** Multi-agent communication protocols and message passing; role-based agent design (planner, researcher, writer, critic, executor); cooperative vs competitive strategies; debate and consensus mechanisms for improved reasoning; agent-to-agent delegation; supervisor/worker patterns; hierarchical orchestration with LangGraph; swarm architectures; game-theoretic foundations; multimodal agents reasoning across text, images, and tool outputs. **Applications:** Multi-agent software development teams (planner + coder + tester); collaborative research agents; supply chain optimization; automated content pipelines.	
MODULE – V	**09 Hours**
Agent Safety, Evaluation & Deployment **Topics:** Agent safety and alignment — sandboxing, permission models, guardrails, policy enforcement; evaluation metrics for agents (task completion, efficiency, safety violations); benchmarking frameworks (AgentBench, SWE-bench, WebArena); human-in-the-loop design patterns; interrupt/resume primitives for supervised execution; deployment architecture (containerization, monitoring, observability with LangSmith); red-teaming agents; regulatory considerations (EU AI Act Article 14 human oversight, NIST AI RMF). (Inference sizing and cost-latency trade-offs covered in depth via DLI.) **Applications:** Production AI assistants, autonomous DevOps agents, healthcare decision support, financial advisory agents, compliance agents.	

Authentic Intelligence :

- Module 1 — Foundations: When should an agent defer to a human? Students discuss the limits of autonomous decision-making and the critical distinction between capability and trustworthiness — an agent that CAN perform a task is not one that SHOULD do it unsupervised in all contexts.
- Module 2 — Reasoning Patterns: Hallucination detection and self-correction — when does the agent know it doesn't know? Students evaluate the reliability of agent self-assessment through Reflexion experiments, and reflect on new failure modes introduced when self-critique itself becomes unreliable.
- Module 3 — Tool Use, RAG & Orchestration: Grounding vs. hallucination — students evaluate when retrieval genuinely improves agent outputs versus when it introduces noise, stale information, or misleading sources that sound authoritative but are factually wrong.
- Module 4 — Multi-Agent Systems: Emergent behavior risks — students analyze when multi-agent approaches amplify individual agent failures through cascading errors versus when they improve reliability through debate, redundancy, and consensus mechanisms; they design safeguards accordingly.
- Module 5 — Safety & Deployment: The alignment problem in agentic AI — ensuring agents act in the user's true interest, not just their stated instructions. Students design trust boundaries, identify decision points where human judgment must remain non-negotiable, and evaluate regulatory

compliance requirements.

AI Integration:

- Module 1 — Foundations: Students set up local LLM inference using Ollama and NVIDIA NIM, compare model variants on reasoning tasks, and use the DLI NIM course to understand microservice deployment patterns for production agent backends.
- Module 2 — Reasoning Patterns: Students complete DLI Prompt Engineering labs on LangChain LCEL, structured outputs, and tool-use patterns; classroom then implements a ReAct agent from scratch to expose the mechanics beneath framework abstractions and identify failure modes.
- Module 3 — Tool Use, RAG & Orchestration: Students complete the DLI Building RAG Agents lab covering RAG architecture, LangGraph state machines, dialog management, and retrieval evaluation; classroom focuses on productionizing results with error handling, circuit breakers, and advanced output schemas.
- Module 4 — Multi-Agent Systems: Students build CrewAI multi-agent workflows with role specialization (planner, researcher, writer, critic) and implement a hierarchical supervisor-routing system with LangGraph; they quantitatively measure end-to-end task quality against a single-agent baseline.
- Module 5 — Safety & Deployment: Students implement human-in-the-loop agents with interrupt/resume primitives, red-team a deployed agent with prompt injection and unauthorized tool-use attempts, and instrument observability with LangSmith; DLI Inference Sizing course provides cost-latency trade-off analysis.

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain foundations of agentic AI including agent taxonomies, perception-reasoning-action loops, and environment classifications.	L2
2	Apply reasoning patterns (ReAct, Reflexion, CoT) and implement memory-augmented agents from scratch and with frameworks.	L4
3	Analyze and implement tool use, RAG, and agent orchestration with LangChain, LangGraph, and CrewAI.	L4
4	Design multi-agent systems with role specialization, delegation patterns, and supervisor-worker architectures.	L3
5	Evaluate agent safety, alignment, and production deployment with HITL oversight, benchmarking, and regulatory awareness.	L5

Mapping Levels of COs to POs / PSOs

Table: Mapping Levels of COs to POs / PSOs

COS	Program Outcomes (POs)											PSOs		
	1	2	3	4	5	6	7	8	9	10	11	1	2	3
CO-1	3	2	1	–	1	–	–	1	1	–	–	3	2	1
CO-2	3	3	3	2	3	–	–	–	–	–	2	3	3	3
CO-3	3	3	3	–	3	1	–	–	2	1	2	3	3	3

CO-4	3	2	2	3	1	2	1	–	–	–	3	3	3	2
CO-5	3	3	3	–	3	2	–	3	2	–	3	3	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

PSO-1: Design and build robotic systems for different applications using analytical, logical, and problem-solving skills.
PSO-2: Develop AI-powered software systems to control and operate robotic systems.
PSO-3: Apply robotics and AI to solve real-world problems in a variety of domains, including industrial robotics, service robots, exoskeletons, surgical robots, delivery vehicles, autonomous vehicles, and crewless micro aerial vehicles

TEXT BOOKS:

1. Stuart Russell & Peter Norvig, *Artificial Intelligence: A Modern Approach*, 4th Edition, Pearson, 2021.
2. Michael Wooldridge, *An Introduction to MultiAgent Systems*, 2nd Edition, Wiley, 2009.

REFERENCE BOOKS:

1. Chip Huyen, *AI Engineering: Building Applications with Foundation Models*, O'Reilly Media, 2025.
2. Shunyu Yao et al., *ReAct: Synergizing Reasoning and Acting in Language Models*, ICLR 2023.
3. Noah Shinn et al., *Reflexion: Language Agents with Verbal Reinforcement Learning*, NeurIPS 2023.

E-RESOURCES:

1. https://langchain-ai.github.io/langgraph/ (LangGraph Documentation)
2. https://microsoft.github.io/autogen/ (AutoGen Documentation)
3. https://www.deeplearning.ai/short-courses/ (Multi AI Agent Systems with CrewAI)
4. https://lilianweng.github.io/posts/2023-06-23-agent/ (Lilian Weng: LLM Powered Autonomous Agents)
5. https://www.anthropic.com/engineering/building-effective-agents (Anthropic: Building Effective Agents)

NVIDIA DLI: Introduction to NVIDIA NIM Microservices (2 hrs, FREE) — NIM microservice architecture, deployment patterns, API standards, and how to build and scale AI applications using NVIDIA-optimized model endpoints. Reinforces Module 1 deployment context. URL: https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-FX-23+V1
NVIDIA DLI: Building LLM Applications With Prompt Engineering (8 hrs, $90) — Iterative prompt development, LangChain LCEL and runnables, chat messages, few-shot prompting, structured outputs with Pydantic, and basic LLM agents with tool use. Foundational for Module 2. URL: https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-FX-12+V2
NVIDIA DLI: Building RAG Agents With LLMs (8 hrs, FREE) — RAG architecture and workflow, LangGraph state machines for dialog management, embedding-based similarity search, document reasoning agents, and retrieval evaluation metrics. Core to Module 3. URL: https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-FX-15+V1
NVIDIA DLI: Sizing LLM Inference Systems (3 hrs, $30) — LLM inference performance metrics (latency, throughput, TTFT), tensor parallelism, in-flight batching, KV-cache optimization, and cost estimation using NVIDIA NIM. Supports Module 5 deployment. URL: https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-FX-18+V1

RECOMMENDED TOOLS

Python 3.10+, LangChain, LangGraph, CrewAI, AutoGen
LLM Inference: Ollama, OpenAI API, Anthropic API, NVIDIA NIM
Vector Stores: ChromaDB, FAISS, Pinecone
Observability: LangSmith, OpenTelemetry, Weights & Biases
Vector Stores: ChromaDB, FAISS, Pinecone — chromadb.com

LAB EXERCISES

S.No	Exercise Title	Aim / Objective	Tools & Dataset	Hrs	CO
MODULE I · Exercises 1–3 : Agentic AI Foundations (06 Hours)					
1	**Agent Development Environment Setup**	Configure Python agent stack (LangChain, Ollama); run local LLM inference; test structured JSON output for tool interfaces.	Python, LangChain, Ollama **Dataset:** *Local LLM (Llama 3, Mistral, Qwen)*	2 h	**CO1**
2	**Single-Turn Tool-Calling Agent**	Build a basic agent that calls calculator and web search tools; test structured output validation and error recovery.	Python, LangChain, Tavily API **Dataset:** *Multi-step calculation tasks*	2 h	**CO1**
3	**ReAct Agent from Scratch**	Implement the ReAct loop without frameworks; observe Thought-Action-Observation traces; identify hallucinated tools and infinite loops.	Raw OpenAI/Anthropic API, Python **Dataset:** *Multi-hop factual questions*	2 h	**CO1**
MODULE II · Exercises 4–5 : Reasoning Patterns & Memory (04 Hours)					
4	**Chain-of-Thought vs Tree-of-Thought**	Implement CoT and ToT on the same benchmark (GSM8K subset); measure accuracy vs cost and latency trade-offs.	LangChain, Python **Dataset:** *UCI Adult Income Dataset*	2 h	**CO1**
5	**Memory-Augmented Agent**	Add ChromaDB vector memory to an agent; evaluate long-conversation consistency; test Reflexion self-critique on reasoning errors.	LangChain, ChromaDB **Dataset:** *UCI Adult Income Dataset*	2 h	**CO1**
MODULE III · Exercises 6–7 : RAG & Orchestration (04 Hours)					
6	**Production-Style RAG Pipeline**	Build a RAG pipeline with dense + sparse retrieval, query transformation, and re-ranking; measure MRR, NDCG, faithfulness.	LangChain, ChromaDB, rerankers **Dataset:** *Enterprise document corpus*	2 h	**CO2**
7	**LangGraph Stateful Agent with Error Handling**	Design a stateful LangGraph workflow with conditional routing; implement error handling and retry with circuit breakers to handle tool failures gracefully.	LangGraph, Pydantic **Dataset:** *Customer service / triage scenarios*	2 h	**CO2**
MODULE IV · Exercises 8–10 : Multi-Agent Systems (06 Hours)					
8	**CrewAI Role-Based Multi-Agent System**	Build a CrewAI team with planner, researcher, writer, and critic agents; measure end-to-end task quality vs. single-agent baseline.	CrewAI, Python **Dataset:** *Research report generation tasks*	2 h	**CO2**
9	**Multi-Agent Debate for Reasoning**	Two agents argue opposing positions; a judge agent synthesizes; measure accuracy improvement over single-agent on reasoning benchmarks.	LangChain, Python **Dataset:** *GSM8K subset, logic puzzles*	2 h	**CO3**
10	**Supervisor-Worker Orchestration**	Build a supervisor agent that dynamically routes to specialized worker agents (coder, researcher, validator); evaluate routing accuracy.	LangGraph, Python **Dataset:** *Mixed task stream (code, search, math)*	2 h	**CO4**
MODULE V · Exercises 11–12 : Safety & Capstone (10 Hours)					

S.No	Exercise Title	Aim / Objective	Tools & Dataset	Hrs	CO
11	**Agent Red-Team & Guardrail Design**	Red-team a deployed agent with prompt injection and unauthorized tool-use attempts; design and test guardrails; quantify safety improvement.	LangChain, guardrails-ai, Python **Dataset:** *Custom adversarial prompt suite*	4 h	CO4
12	**Production-Grade Agent with HITL (Capstone)**	Design a production agent: LangGraph state machine → interrupt/resume primitives → guardrails → observability (LangSmith) → cost tracking → regulatory documentation.	Full stack: LangGraph, LangSmith, guardrails-ai, OpenTelemetry, NVIDIA NIM **Dataset:** *Student-chosen enterprise workflow domain*	6 h	CO5

CoE (Centre of Excellence) Utilization Mapping for Lab Exercises

DSU AI & Robotics CoE — Dayananda Sagar University | Semester IV | Year 2: Core Systems (2026–2027) | NVIDIA DLI Integration

CoE Note: Lab exercises are aligned with the DSU AI & Robotics CoE Year 2 (Semester IV) Core Systems milestone under the 2026–2029 Blueprint. NVIDIA AI/DLI Lab (DGX) supports 40 students/batch. NVIDIA Jetson Nano = 15–20 units (₹6.36L investment). All NVIDIA DLI courses listed are FREE as per the March 2026 NVIDIA DLI learning path catalogue.

GEN AI AND PROMPT ENGINEERING

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	:	25AM3602	**Credits**	:	04
Hours / Week	:	03 Hours	**Total Hours**	:	39 (T)+26 (P)
L–T–P–S	:	3–0–2–0			

Course Learning Objectives:

This course will enable students to:

1. **Understand foundational concepts** of Generative AI, including transformer architectures, attention mechanisms, embeddings, and the distinction between language models and foundational models.
2. **Master prompt engineering and model evaluation** including zero-shot and few-shot learning, defensive prompting strategies, evaluation metrics, and model selection criteria.
3. **Implement advanced customization techniques** including parameter-efficient finetuning (PEFT), RLHF, and Retrieval-Augmented Generation (RAG) systems based on specific use case requirements.
4. **Design and deploy AI agent systems** with planning, memory, and tool-use capabilities, utilizing frameworks like LangChain for autonomous decision-making and multi-agent orchestration.
5. **Apply AI engineering principles** for practical implementation, monitoring, and deployment of generative AI solutions in production environments.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the student's understanding.

UNIT – I	**06 Hours**

Introduction to Generative AI: Introduction to Generative AI, Definition and scope of Generative AI, Hierarchy of Generative AI, Overview of generative models and their applications, Importance of Generative AI in various domains, Ethical considerations and challenges.

UNIT – II	**09 Hours**

Understanding Foundational Models: Introduction to Embeddings, Basic Encoder/decoder architectures, Simple Attention Mechanisms, Transformer Architecture, Multi-Head Attention Mechanisms, GPT architecture and its significance, Dataset Engineering Fundamentals, Hands-

on Implementation.
UNIT – III **07 Hours**
Prompt Engineering and Evaluating Foundational Models: Introduction to Prompt Engineering, In-Context Learning: Zero-Shot and Few-Shot, Prompt Engineering Best Practices, Defensive Prompt Engineering, Language Modeling Metrics, Challenges of Evaluating Foundational Models, Evaluation Criteria, Model Selection, Designing an Evaluation Pipeline
UNIT – IV **09 Hours**
Finetuning and RAG: Overview of Finetuning, Memory Bottlenecks and Computational Considerations, Full Finetuning vs Parameter Efficient Finetuning (PEFT), Reinforcement Learning with Human Feedback (RLHF) Fundamentals, Introduction to Retrieval Augmented Generation (RAG), Finetuning vs RAG, Retrieval Algorithms and Vector Databases, RAG Pipeline optimization and Chunking Strategies, Hybrid approaches (Combining fine-tuning with RAG), Model Merging Techniques and Multitask considerations
UNIT – V **09 Hours**
AI Agents and Practical Implementation: Introduction to AI agents and autonomous systems, Core components of intelligent agents (Planning, Memory, Tool Use, Goal-directed behavior), Types of AI agents (Reflex agents, Goal-based agents, Utility-based agents, Learning agents), Agent orchestration and multi-agent systems, Introduction to frameworks (LangChain, AutoGen, CrewAI, LangGraph), Retrieval-Augmented Generation (RAG) agents, Chain of thought reasoning and multi-step decision making, Agent deployment strategies and monitoring, AI Engineering Architecture and Monitoring, Case studies and hands-on implementation, Responsible AI Development

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Explain foundational concepts of Generative AI including transformer architectures and embeddings.	L2
2	Apply prompt engineering techniques and evaluate model performance using appropriate metrics.	L3
3	Implement finetuning and RAG techniques to customize foundational models for specific tasks.	L3
4	Design AI agent systems with planning and memory capabilities using modern frameworks.	L4
5	Create and deploy generative AI solutions with proper monitoring and engineering practices.	L6

Mapping Levels of COs to POs / PSOs					
Cos				**Program Outcomes (POs)**	**PSOs**

	1	2	3	4	5	6	7	8	9	10	11	12	1	2
	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	2	1	2	2	1		1	1	2		3	2	2
CO2	2	3	2	3	3	1		2	1	2	1	3	1	3
CO3	2	2	3	2	3	2	1	2	2	1	1	2	2	3
CO4	2	3	3	2	3	2	1	2	3	2	2	2	2	3
CO5	3	2	3	1	3	3	2	3	2	3	3	3	2	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

Text Books:

1. Foster, David. *Generative deep learning*. O'Reilly Media, Inc., 2022.
2. Dhamani, Numa. *Introduction to Generative AI*. Simon and Schuster, 2024.
3. Chip Huyen. *AI Engineering: Building Applications with Foundation Models*. O'Reilly Media, Inc., 2024.

Reference Books:

1. Mastering AI Agents: A comprehensive guide for evaluating AI agents: *Pratik Bhavsar*
2. Babcock, Joseph, and Raghav Bali. *Generative AI with Python and TensorFlow 2: Create images, text, and music with VAEs, GANs, LSTMs, Transformer models*. Packt Publishing Ltd, 2021
3. Alto, Valentina. *Modern Generative AI with ChatGPT and OpenAI Models: Leverage the capabilities of OpenAI's LLM for productivity and innovation with GPT3 and GPT4*. Packt Publishing Ltd, 2023. de Albuquerque, Victor Hugo C., Pethuru Raj, and Satya Prakash Yadav, eds. *Toward Artificial General Intelligence: Deep Learning, Neural Networks, Generative AI*. Walter de Gruyter GmbH & Co KG, 2023.

E-Resources:

1. https://jalammar.github.io/illustrated-transformer/
2. https://www.3blue1brown.com/lessons/gpt
3. https://poloclub.github.io/transformer-explainer/
4. https://www.datacamp.com/blog/what-is-prompt-engineering-the-future-of-ai-communication
5. https://www.promptengineering4u.com/learning/techniques/template-based-prompting#h.2n56pv37pv0c

Activity Based Learning (Suggested Activities in Class)

1. Hands-on prompt engineering with real LLMs.
2. Model evaluation sessions using different metrics.
3. AI agent development project using modern frameworks.
4. Mini Project

LIST OF LABORATORY/PRACTICAL EXPERIMENTS ACTIVITIES TO BE CONDUCTED
1. Implement Word Embeddings using the Skip-gram model and explore semantic similarity. 2. Build a Simple Autoencoder for image reconstruction and dimensionality reduction. 3. Implement a Tiny GAN for generating handwritten digits using adversarial training. 4. Design and implement a Variational Autoencoder (VAE) for probabilistic image generation. 5. Develop a Character-Level RNN for text generation using the Shakespeare dataset. 6. Implement a Baby Transformer model from scratch for sequence prediction tasks. 7. Perform GPT-2 prompting and analyse text generation under different decoding strategies. 8. Build a Semantic Search engine using sentence embeddings and FAISS. 9. Fine-tune a pre-trained DistilBERT model for sentiment classification on the IMDB dataset. 10. Generate images from text prompts using Stable Diffusion and analyze quality–speed tradeoffs. 11. Implement a Mini Retrieval-Augmented Generation (RAG) system integrating retrieval and generation. 12. Develop a Multi-Modal Search system using CLIP embeddings for text–image alignment and retrieval.

****

DESIGNING MLOPS FOR ENTERPRISES

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	: 25AM3603	**Credits**:	03
Hours / Week	: 3 Hours	**Total Hours**:	39 Hours
L–T–P–S	: 2–0–0–2		

Course Learning Objectives:

This course will enable students to:

1. To impart knowledge on production-level challenges of ML models
2. To provide comprehension of various activities involved in the development, deployment, and monitoring of ML models
3. To familiarize the principles of MLOps and different platforms

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** clips to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the student's understanding.

UNIT – I	**07 Hours**
Introduction to DevOps: SDLC, Virtualization: Containers, Container Orchestration Systems, Cloud platforms, CI/CD: Continuous Integration – Configuration Management, Deployment and Delivery phases, Continuous monitoring, Continuous Testing	
UNIT – II	**08 Hours**
Basic Concepts: Evolution of MLOps, Data-centric AI, ML Development Lifecycle, MLOps Approach, Features of MLOps, ML Data Lifecycle in Production, MLOps maturity levels, ML artifacts, MLOps workflows.	
UNIT – III	**08 Hours**
Machine Learning Pipelines and automation: CI/CD for Machine Learning, ML model serving, Data pipelines, Data drift, ML pipelines: Data ingestion, Feature engineering, Hyperparameter optimization, testing and packaging.	
UNIT – IV	**08 Hours**
Model in MLOps : Model management: Model deployment and monitoring, feedback, orchestration pipelines for ML workflows, ML security, Real-time Streaming ML models, Deployment on edge devices, Automated ML.	
UNIT -V	**08 Hours**
Case Studies on MLOps best practices: **Netflix:** Enhancing Content Recommendations with MLOps, **Uber:** Demand Forecasting with MLOps, **Airbnb**: Search Ranking Models, **Intuit**: Fraud Detection and Prevention, **NASA:** Satellite	

Image Analysis.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Explain and assess the effectiveness and scalability of an end-to-end machine learning system in a real-world scenario.	L2
2	Recall various metrics used to evaluate machine learning model performance	L2
3	Design and integrate comprehensive automated systems that encompass ML pipelines, CI/CD, data processing, model serving, and drift detection.	L3, L4
4	Analyze the challenges and benefits of implementing MLOps in an organization.	L4
5	Evaluate the applicability and effectiveness of MLOps best practices from case studies in your own or another organization.	L5

Mapping Levels of COs to POs / PSOs

Cos				Program Outcomes (POs)									PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	2	2	2	2	-	2	-	-	2	1	-	2	2	2
CO2	2	2	2	2	-	2	-	-	2	1	-	2	2	2
CO3	2	2	2	2	-	2	-	-	2	1	-	2	2	2
CO4	2	2	2	2	-	2	-	-	2	1	-	2	2	2
CO5	2	1	-	2	-	-	-	2	-	1	-	-	2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Alla, Sridhar, and Suman Kalyan Adari. *Beginning MLOps with MLFlow*. Apress, 2021.
2. Rao, Dattaraj. *Keras to Kubernetes: The Journey of a Machine Learning Model to Production*. John Wiley & Sons, 2019.
3. Sculley, David, et al. "Machine learning: The high interest credit card of technical debt." (2014).
4. Jez Humble, David Farley. Continuous Delivery.,2011

REFERENCE BOOKS:

1. Treveil, Mark, Nicolas Omont, Clément Stenac, Kenji Lefevre, Du Phan, Joachim Zentici, Adrien Lavoillotte, Makoto Miyazaki, and Lynn Heidmann. *Introducing MLOps*. O'Reilly Media, 2020.
2. Burkov, Andriy. *Machine Learning Engineering.* True Positive Inc. , 2020.
3. Ameisen, Emmanuel. *Building Machine Learning Powered Applications.* O'Reilly Media, 2020..

Activity Based Learning (Suggested Activities in Class)

1. Assignments (in writing and doing forms on the aspects of syllabus content and outside the syllabus content. Shall be individual and challenging)
2. Student seminars (on topics of the syllabus and related aspects (individual activity)
3. Quiz (on topics where the content can be compiled by smaller aspects and data (Individuals or groups as teams))
4. Study projects (by very small groups of students on selected local real-time problems pertaining to syllabus or related areas. The individual participation and contribution of students shall be ensured (team activity))

****

ACCELERATED COMPUTING

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER - VI

Course Code	:	25AM3606	**Credits**	:	03
Hours / Week	:	03	**Total Hours**	:	45
L–T–P–J	:	2–0–0–2	**CIE+SEE**	:	60 + 40 Marks

Course Vision:

This course introduces students to accelerated computing using GPUs, focusing on parallel programming concepts with CUDA. It covers memory hierarchy, performance optimization, parallel computation patterns, and multi-GPU systems, enabling students to design high-performance computing applications.

Course Objectives:

This course will enable students to:

1. Understand GPU architecture and CUDA programming fundamentals
2. Apply parallel programming concepts using CUDA APIs
3. Analyze memory hierarchy and optimize data access
4. Design efficient parallel algorithms and computation patterns
5. Implement scalable GPU applications using advanced CUDA features.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. **Lecture method** along with traditional lecture method, different type of teaching methods may be adopted to develop the course outcomes.
2. **Interactive Teaching:** incorporating brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Showing **Video/animation** films to explain functioning of various concepts.
4. Encourage **Collaborative** (Group Learning) Learning in the class.
5. To make **Critical thinking**, asking Higher order Thinking questions in the class in the form of Quiz and writing programs with complex solutions.
6. Showing the **different ways to solve** the same problem and encourage the students to come up with their own creative ways to solve them.

UNIT – I	**09 Hours**

Introduction to CUDA

Topics: Introduction to Heterogeneous Parallel Computing – portability and scalability - Overview of Accelerated Computing -GPU vs CPU Architecture Introduction to CUDA Programming Model- CUDA API Basics (Host and Device Code)- CUDA Threads, Blocks, and Grids -Data Parallelism Concepts

Applications: Vector addition, image processing, scientific simulations.

UNIT – II	**09 Hours**

Memory and Data Locality

Topics: Memory Hierarchy in CUDA, Global Memory, Shared Memory, Registers, Memory,

Allocation and Data Transfer, Memory Coalescing, Data Locality Optimization Techniques, Unified Memory Concepts. **Applications:** Matrix operations, numerical computing, real-time data processing.	
UNIT – III	**09 Hours**
Thread Execution and Performance Optimization **Topics:** Thread Execution Model, Warp Execution and Divergence, Thread Scheduling, Performance Bottlenecks, Optimization Techniques, Profiling CUDA Programs. **Applications:** AI pipeline optimization, HPC workloads, real-time analytics systems.	
UNIT – IV	**09 Hours**
Parallel Computation Patterns **Topics:** Common Parallel Patterns, Reduction, Scan (Prefix Sum), Stencil Computations, Algorithm Design for Parallelism, Load Balancing, Synchronization Mechanisms. **Applications:** Large-scale data processing, simulation systems, scientific computing workflows.	
UNIT – V	**09 Hours**
Advanced CUDA and Multi-GPU Systems **Topics:** Efficient Host–Device Data Transfer, CUDA Streams and Asynchronous Execution, Dynamic Parallelism, Multi-GPU Programming, CUDA Libraries (cuBLAS, cuDNN, etc.), Introduction to OpenACC, MPI, OpenCL **Applications:** Deep learning training, autonomous systems, big data analytics, real-time inference systems.	

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain the fundamentals of parallel computing, GPU architecture, and CUDA programming models.	L2
2	Apply CUDA programming and parallel algorithms to develop high-performance computing solutions.	L3
3	Analyze performance bottlenecks and optimize applications using profiling and GPU optimization techniques.	L4
4	Create scalable and efficient GPU-accelerated applications using modern C++ and parallel computing frameworks.	L5
5	Evaluate real-world problems and implement accelerated computing solutions for AI, data science, and HPC applications.	L5

Mapping Levels of COs to POs / PSOs

Table: Mapping Levels of COs to POs / PSOs													
COS	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO-1	3	2	2	–	2	2	–	2	–	–	2	2	2
CO-2	3	3	3	–	3	3	–	2	–	–	–	3	3
CO-3	3	2	3	–	3	3	–	3	–	–	–	3	3
CO-4	3	2	3	2	3	3	–	3	2	–	–	3	3 -
CO-5	3	2	3	–	3	3	–	3	2	2	2	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI Integration

1. GPU acceleration for faster model training; parallel data preprocessing; efficient handling of large-scale datasets.
2. Accelerating neural network computations; parallel feature extraction; optimizing training pipelines.
3. Optimizing deep learning workflows; identifying slow stages in ML pipelines; improving GPU utilization.
4. Integration of C++ acceleration in AI systems; improving inference speed; scalable deployment of models.
5. Full GPU-based AI workflow optimization; deployment-ready accelerated models; performance-aware AI system design.

Authentic Intelligence

1. Understanding computational efficiency vs energy consumption; responsible use of high-performance resources; scalability considerations in real-world systems.
2. Designing efficient algorithms with minimal resource wastage; balancing speed and accuracy; avoiding redundant computations.
3. Interpreting performance trade-offs; optimizing without compromising correctness; responsible benchmarking practices.
4. Writing portable and maintainable parallel code; ensuring system reliability; efficient resource utilization at scale.
5. Designing efficient and sustainable AI systems; evaluating trade-offs in performance vs cost; end-to-end optimization for real-world deployment.

TEXT BOOKS:

1. Yoshida, K., Miwa, S., Yamaki, H., & Honda, H. (2024). Analyzing the impact of CUDA versions on GPU applications. *Parallel Computing*, *120*, 103081.

2. Boopathi, S., Pandey, B. K., & Pandey, D. (2023). Advances in artificial intelligence for image processing: techniques, applications, and optimization. In *Handbook of research on thrust technologies' effect on image processing* (pp. 73-95). IGI Global Scientific Publishing.
3. Nemirovsky, M., & Tullsen, D. (2022). *Multithreading architecture*. Springer Nature.

REFERENCE BOOKS:

1. Sinadjan, L. (2025). GPU Acceleration for Faster Evolutionary Spatial Cyclic Game Systems. *arXiv preprint arXiv:2508.16639*.
2. Zhao, Q. (2025). *Novel Profiling Techniques for Emerging Architectures and Applications*. North Carolina State University.

E-RESOURCES:

https://developer.nvidia.com/accelerated-computing-teaching-kit-syllabus

1. https://developer.nvidia.com/cuda/toolkit/
2. https://www.nvidia.com/en-in/training/
3. https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-FX-07+V1
4. https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-DS-01+V1
5. https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-FX-05+V1
6. https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-FX-05+V1

RECOMMENDED TOOLS

- Use CUDA by Example for lab/practical understanding
- Keep others as reference books for depth and evaluation.

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

LLM BASED AGENT FRAMEWORKS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER-VII

Course Code	:	25AM3607	**Credits**	:	03
Hours/Week	:	03	**Total Hours**	:	45
L–T–P–J	:	2–0–0–2	**CIE+SEE**	:	60+40 Marks

Course Vision:

To empower learners with strong foundations and practical expertise in Large Language Models and agent-based AI systems, enabling them to design, analyze, and deploy intelligent, scalable, and efficient AI solutions for real-world applications.

Course Objectives:

This course will enable students to:

1. To understand the architecture and working principles of Large Language Models and transformers
2. To master practical techniques for building AI applications including prompt engineering and RAG
3. To implement agentic workflows and fine-tune models using parameter-efficient techniques
4. To design evaluation methodologies and optimize models for production deployment
5. To develop end-to-end AI engineering skills for real-world LLM applications.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. **Lecture method** along with traditional lecture method, different type of teaching methods may be adopted to develop the course outcomes.
2. **Interactive Teaching:** incorporating brainstorming, discussing, group work, focused listening, formulating questions, note taking, annotating, and role playing.
3. Showing **Video/animation** films to explain functioning of various concepts.
4. Encourage **Collaborative** (Group Learning) Learning in the class.
5. To make **Critical thinking**, asking higher order Thinking questions in the class in the form of Quiz and writing programs with complex solutions.
6. Showing the **different ways to solve** the same problem and encourage the students to come up with their own creative ways to solve them

MODULE–I	**09 Hours**

INTRODUCTION TO AI ENGINEERING AND FOUNDATION MODELS:

Topics: The Rise of AI Engineering: From language models to large language models, from LLMs to foundation models, AI engineering vs ML engineering, the AI engineering stack. **Understanding Foundation Models**: Training data (Speech processing and domain-specific models), model size considerations, post-training (supervised fine-tuning, preference fine-tuning), sampling fundamentals and strategies, test-time compute, structured outputs, the probabilistic/deterministic AI. **Planning AI Applications**: Foundation model use cases (coding, writing, education, conversational bots, information aggregation, workflow automation), use case evaluation, setting expectations, milestone planning, cost and latency considerations, model build vs buy decisions, model econometrics.

MODULE – II	09 Hours
TRANSFORMER ARCHITECTURE AND LLM FUNDAMENTALS **Topics: Working with Text Data**: Understanding word embeddings, tokenizing text (byte pair encoding), converting tokens to token IDs, adding special context tokens, data sampling with sliding window, creating token embeddings, encoding word positions. **Attention Mechanisms**: Exploring LCS (Long Common Sequences), capturing data dependencies with attention, self-attention mechanism (with and without trainable weights), computing attention weights, causal attention and masking future words, dropout in attention, multi-head attention. **Transformer Architecture:** Introduction to transformer architecture, encoder-decoder models, GPT architecture deep-dive. Implementing GPT Components: Layer normalization, feed-forward networks with GELU activations, shortcut connections, connecting attention and linear layers in transformer blocks, coding the GPT model, text generation mechanisms.	
MODULE – III	**09 Hours**
RETRIEVAL-AUGMENTED GENERATION & MODEL ORCHESTRATION: **Topics: Prompt Engineering**: Introduction to prompting, in-context learning (zero-shot and few-shot), system prompts vs user prompts, context length and efficiency. **Best Practices**: Writing clear instructions, providing sufficient context, breaking complex tasks into subtasks, chain-of-thought prompting, iterative prompt development, evaluating prompt engineering tools, organizing and versioning prompts. **Defensive Prompting**: Proprietary prompts and reverse engineering, jail breaking and prompt injection attacks, information extraction risks, defenses against attacks. **RAG Fundamentals**: RAG architecture and workflow, when to use RAG vs fine-tuning, RAG limitations and challenges. Retrieval Systems: Dense retrieval (semantic search with embeddings), sparse retrieval (BM25, TF-IDF), hybrid retrieval strategies, vector databases and indexing (Pinecone, Weaviate, Chroma). **Retrieval Optimization**: Query transformation and expansion, re-ranking strategies, chunk size optimization, metadata filtering, evaluation metrics for retrieval (recall, MRR, NDCG), RAG beyond text (multimodal RAG).	
MODULE – IV	**09 Hours**
AGENT SYSTEMS AND FINE-TUNING TECHNIQUES: **Topics: Agent Systems**: Agent overview and architecture, agentic workflows vs traditional pipelines, when to use agents. Agent Components: Tools and tool use (web search, calculators, databases), planning strategies (ReAct, tree-of-thoughts, chain-of-thought), agent memory (short-term and long-term), agent orchestration frameworks (LangChain, AutoGPT, AutoGEN). **Agent Challenges**: Failure modes and debugging, agent evaluation strategies, cost and latency optimization. **Fine-tuning Fundamentals**: When to fine-tune (vs RAG vs prompt engineering). **Parameter-Efficient Fine-Tuning (PEFT)**: LoRA (Low-Rank Adaptation) theory and implementation, QLoRA for memory efficiency.	
MODULE – V	**09 Hours**
MODEL EVALUATION, INFERING AND DEPLOYMENT: **Topics: Evaluation Methodology**: Challenges of evaluating foundation models, language modeling metrics (entropy, cross-entropy, perplexity, bits-per-character), exact evaluation methods, functional correctness, similarity measurements (lexical and semantic), introduction to embeddings. **AI as a Judge**: Why and how to use AI as judge, limitations, comparative evaluation and ranking models. **Evaluation Criteria**: Domain-specific capability, generation capability, instruction-following capability, cost and latency evaluation. **Designing Evaluation Pipelines**: Model selection workflow, navigating public benchmarks, evaluating all components in a system,	

creating evaluation guidelines, defining evaluation methods and data. **Inference Optimization**: Understanding inference optimization, inference overview, performance metrics (latency, throughput, TTFT, cost), AI accelerators (GPUs, TPUs). **Model Optimization**: Quantization (post-training and quantization-aware), pruning, knowledge distillation. **Inference Service Optimization**: Batching strategies, caching mechanisms, and model serving frameworks.

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain the architecture of transformers and GPT models, including attention mechanisms.	L2
2	Implement prompt engineering techniques and build RAG systems with retrieval optimisation	L3
3	Design and implement agentic workflows and apply parameter-efficient fine-tuning (LoRA, PEFT).	L5
4	Develop comprehensive evaluation methodologies and pipelines for AI systems.	L5
5	Optimize LLM inference and deploy production-ready AI applications with monitoring.	L5

Mapping Levels of COs to POs/PSOs

Table: Mapping Levels of COs to POs/PSOs

COS	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO-1	3	2	–	1	2	–	–	–	–	–	–	2	1
CO-2	2	3	2	2	3	–	–	–	1	–	–	3	2
CO-3	2	3	3	2	3	–	–	–	2	1	–	3	3
CO-4	2	3	2	3	2	–	–	–	1	1	–	2	3
CO-5	2	3	3	3	3	1	–	–	2	2	1	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI Integration

AI is integrated in this unit through exploratory and application-based learning using tools such as ChatGPT and Google Gemini. Students interact with large language models to understand real-world AI applications, probabilistic outputs, and foundational concepts. Case studies on foundation models are discussed, and learners identify domain-specific AI use cases in areas like education, automation, and conversational systems, enabling conceptual clarity and practical exposure.

This unit incorporates AI integration through hands-on implementation and experimentation with transformer models using libraries such as Hugging Face Transformers. Students perform tokenization, embedding generation, and attention visualization through coding exercises. They also experiment with pre-trained models to understand text generation and internal working of GPT architectures, thereby strengthening their foundational and analytical skills.

This unit integrates AI by developing intelligent agent systems using frameworks such as LangChain and AutoGPT. Students build agentic workflows capable of using tools, planning, and managing memory. Additionally, they explore parameter-efficient fine-tuning concepts such as LoRA through guided experiments, and analyse real-world challenges in agent performance and debugging.

AI integration in this unit emphasises the evaluation and deployment of AI systems in practical environments. Students apply evaluation metrics, use AI-as-a-judge techniques, and analyze model performance in terms of latency and cost. They also gain exposure to deploying AI models using APIs and cloud platforms, and perform optimisation techniques such as quantization and caching. A mini project involving end-to-end AI application deployment ensures industry readiness and hands-on expertise.

TEXT BOOKS:

1. Chip Huyen, AI Engineering: Building Applications with Foundation Models, O'Reilly ,Media, 2025.

REFERENCE BOOKS:

1. Sebastian Raschka, Build a Large Language Model (From Scratch), Manning Publications, 2025.
2. Jay Alammar and Maarten Grootendorst, Hands-On Large Language Models, O'Reilly Media, 2024.
3. Lewis Tunstall, Leandro von Werra, and Thomas Wolf, Natural Language Processing with Transformers, O'Reilly Media, 2022.
4. Chip Huyen, Designing Machine Learning Systems, O'Reilly Media, 2022.
5. Ashish Vaswani et al., Attention Is All You Need, NeurIPS, 2017.
6.

E-RESOURCES:

- Hugging Face NLP Course: https://huggingface.co/learn/nlp-course
- DeepLearning.AI Courses on LLMs, RAG, and Agents: https://www.deeplearning.ai/short-courses/
- Stanford CS224N: Natural Language Processing with Deep Learning: https://web.stanford.edu/class/cs224n/
- LangChain Documentation: https://python.langchain.com/
- OpenAI API Documentation: https://platform.openai.com/docs
- Papers with Code – Transformers: https://paperswithcode.com/methods/category/transformers
- AI Engineering Resources by Chip Huyen: https://github.com/chiphuyen/aie-book

RECOMMENDED TOOLS

Python, Jupyter Notebook, Google Colab, Hugging Face Transformers, LangChain, OpenAI API, Chroma, Pinecone, PyTorch, Weights & Biases.

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

VISUAL COMPUTING AND GRAPHICS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER - VI

Course Code	:	25AM3608	**Credits**	:	03
Hours / Week	:	03	**Total Hours**	:	45
L–T–P–J	:	3–0–0–0	**CIE+SEE**	:	60+40 Marks

Course Vision:

This course develops an understanding of modern computer graphics by integrating concepts from rendering pipelines, GPU programming, real-time ray tracing, and neural rendering, enabling students to design and analyze advanced visual computing systems.

Course Objectives:

Students will:

1. Understand core concepts of computer graphics and rendering pipelines
2. Analyze geometric transformations and GPU-based rendering techniques
3. Explore lighting, shading, and material models for realistic rendering
4. Evaluate real-time ray tracing and optimization techniques
5. Assess integration of AI/ML in modern graphics systems.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. Lecture method combined with interactive discussions
2. Brainstorming, group activities, and case-based learning
3. Demonstrations using audio-visual tools and datasets
4. Problem-solving sessions and analytical exercises
5. Encouraging critical thinking through real-world scenarios and design challenges

UNIT – I	**08 Hours**

Graphics Pipeline and GPU Fundamentals

Topics:

Introduction to computer graphics, applications in visualization and game engines, graphics rendering pipeline, rasterization vs ray tracing, GPU architecture fundamentals, memory hierarchy, parallel processing, graphics APIs (OpenGL, Vulkan), mathematical foundations (vectors, matrices, coordinate systems).

Applications: Game engines, simulation systems, visualization platforms, real-time rendering systems.

UNIT – II	**09 Hours**

Geometric Transformations and Shader Programming

Topics: 2D and 3D transformations (translation, rotation, scaling), homogeneous coordinates, viewing transformations, projection systems (orthographic and perspective), camera systems, GPU pipeline execution, shader programming (vertex and fragment shaders), texture mapping basics. **Applications:** 3D modeling systems, animation pipelines, AR/VR environments, interactive graphics applications.	
UNIT – III	**06 Hours**
Lighting, Shading and Rendering Techniques **Topics:** Lighting models (ambient, diffuse, specular), Phong and Blinn-Phong shading, physically-based rendering (PBR), normal mapping, shadow mapping, ray tracing fundamentals, BVH acceleration, global illumination basics. **Applications:** Realistic rendering systems, visual effects, gaming graphics, cinematic visualization.	
UNIT – IV	**11 Hours**
Advanced Rendering and Neural Graphics **Topics:** Real-time ray tracing, hybrid rendering, denoising techniques, rendering optimization (culling, LOD), neural rendering basics, NeRF, AI-based denoising, deep learning in graphics, 3D visualization for AI applications. **Applications:** Next-gen game engines, film production, AI-based rendering systems, immersive visualization technologies.	
UNIT – V	**11 Hours**
Advanced Rendering and Neural Graphics **Topics:** Real-time ray tracing, hybrid rendering, denoising techniques, rendering optimization (culling, level-of-detail), neural rendering basics, NeRF, AI-based denoising, deep learning in graphics, 3D visualization for AI applications. **Applications:** Next-generation game engines, film production, AI-based rendering systems, immersive visualization technologies.	

AI Integration:

- AI-assisted visualization and rendering optimization tools
- AI-supported shader generation and transformation modeling
- AI-based lighting and rendering enhancement tools
- Neural rendering models, AI-based denoising, NeRF systems.

Authentic Intelligence:

- Limitations of automated rendering systems
- Errors in shader design and transformation inaccuracies

- Trade-offs between realism and performance
- Ethical use of AI-generated graphics and synthetic content.

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain graphics pipeline and GPU architecture concepts	L3
2	Analyze transformations, viewing systems, and shader programming	L4
3	Apply lighting models and shading technique	L3
4	Evaluate real-time ray tracing and optimization methods	L5
5	Evaluate neural rendering and AI integration in graphics	L5

Mapping Levels of COs to POs / PSOs

Table: Mapping Levels of COs to POs / PSOs

COS	Program Outcomes (POs)											PSO	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO-1	3	1	–	–	–	–	–	–	–	–	–	3	2
CO-2	3	3	2	2	2	–	–	–	–	–	–	3	3
CO-3	2	3	2	2	2	–	–	–	–	–	–	3	3
CO-4	2	3	2	2	3	–	–	–	1	–	–	3	3
CO-5	1	2	2	3	3	–	–	–	2	1	–	3	3

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

TEXT BOOKS:

1. Shirley, Peter, Michael Ashikhmin, and Steve Marschner. *Fundamentals of Computer Graphics*. **4th Edition. A K Peters/CRC Press, 2021.**

2. Akenine-Möller, Tomas, Eric Haines, and Naty Hoffman. *Real-Time Rendering*. **4th Edition. CRC Press, 2018.**
3. Shreiner, Dave and Bill Licea-Kane. *OpenGL Programming Guide: The Official Guide to Learning OpenGL*. **9th Edition. Addison-Wesley, 2013.**

REFERENCE BOOKS:

1. Foley, James D., Andries van Dam, Steven K. Feiner, and John F. Hughes. *Computer Graphics: Principles and Practice*. **3rd Edition. Addison-Wesley, 2013.**
2. Buss, Samuel R. *3D Computer Graphics: A Mathematical Introduction with OpenGL*. **1st Edition, 2003.**

RESOURCES:

- Khronos Group OpenGL Documentation: https://www.khronos.org/opengl/
- Vulkan API Specifications: https://www.khronos.org/vulkan/
- NVIDIA Developer Blog – Ray Tracing, DLSS, and Neural Rendering: https://developer.nvidia.com/blog/
- SIGGRAPH Papers and Course Materials: https://kesen.realtimerendering.com/
- High-Performance Graphics Conference: https://www.highperformancegraphics.org/
- Global Illumination Compendium: https://people.cs.kuleuven.be/~philip.dutre/GI/
- NPTEL Computer Graphics: https://nptel.ac.in/courses/106106090/

RECOMMENDED TOOLS

- **Programming:** C++ / Python
- **Graphics APIs:** OpenGL / Vulkan
- **Frameworks:** Unity / Unreal Engine (conceptual exposure)
- **Libraries:** OpenGL Shading Language (GLSL)
- **AI Tools:** ChatGPT, AI-based rendering tools

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

ROBOT OPERATING SYSTEMS (ROS)

Professional Elective – II

SEMESTER - VI

Subject Code	:	25AM3609	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 Hours
L–T–P–S	:	2–0–0–2			

Course Vision:

To develop the ability to design, implement, and integrate robotic systems using ROS middleware by combining communication, perception, and control components. The course aims to provide a strong system-level understanding of robotic software architecture, enabling students to build scalable and modular robotic applications aligned with modern autonomous and intelligent systems.

Course Learning Objectives:

Students will:

1. Understand ROS architecture and communication mechanisms for robotic systems.
2. Develop ROS-based programs using nodes, topics, and services.
3. Implement sensor integration and data handling in robotic applications.
4. Analyze and debug robotic systems using ROS tools and visualization frameworks.
5. Build and simulate robotic systems using ROS and simulation environments.

Course Outcome	Description	Bloom's Taxonomy Level
CO1	Analyze and describe ROS architecture and communication mechanisms.	L4
CO2	Develop ROS-based programs using nodes, topics, and services.	L5
CO3	Evaluate sensor integration and data processing in ROS-based systems.	L5
CO4	Analyze and debug robotic systems using ROS visualization and diagnostic tools.	L4
CO5	Design and implement a basic robotic system using ROS in simulation or hardware.	L5

UNIT – I : Introduction to ROS and Robotic Middleware	**07 Hours**

Evolution of robotic middleware, ROS ecosystem, computation graph, nodes, topics, services, ROS architecture, ROS file system, basic ROS setup.

Applications: Autonomous robots, service robots

UNIT – II : ROS Programming and Communication	**07 Hours**
ROS workspace and package creation, publishing and subscribing, message types, services, ROS client libraries (Python), inter-node communication. Applications: Sensor data communication, robot control pipelines	
UNIT – III : Sensor Integration and Data Handling	**08 Hours**
Integration of sensors (camera, LiDAR – conceptual), data acquisition in ROS, message handling, basic data processing, introduction to sensor-based decision pipelines. Applications: Perception-enabled robotic systems	
UNIT – IV : ROS Tools, Simulation, and System Integration	**08 Hours**
ROS debugging tools (rosnode, rqt_graph, rosout, rqt_console), RViz visualization, rosbag data logging, introduction to Gazebo simulation, basic robot simulation, teleoperation, introduction to navigation concepts in ROS. Applications: Autonomous navigation systems, warehouse robots	
UNIT – V: NVIDIA DLI: ROS 2 and Isaac Sim	**15 Hours**
Getting Started with Issac Sim - Building Your First Robot in Isaac Sim, Ingesting Robot Assets and Simulating Your Robot in Isaac Sim, Synthetic Data Generation for Perception Model Training in Isaac Sim, Developing Robots With Software-in-the-Loop (SIL) in Isaac Sim, Leveraging ROS 2 and Hardware-in-the-Loop in Isaac Si Getting Started With Isaac ROS - An Introduction to AI-Based Robot Development With Isaac ROS Accelerating ROS 2 Workloads With NVIDIA GPU-Powered Libraries and AI Models -	

Mapping Levels of COs to POs / PSOs													
COs				**Program Outcomes (POs)**								**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**

CO1	3	2	1	-	2	-	-	-	-	-	1	3	1
CO2	3	3	2	1	3	-	-	1	-	-	2	3	3
CO3	3	3	2	2	3	-	-	1	-	-	2	3	3
CO4	3	3	2	2	3	-	1	2	1	1	2	3	3
CO5	3	3	3	2	3	-	1	2	2	1	2	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low) AI**

Integration & Authentic Intelligence Reflection (AI^2 Pedagogy)

AI Integration:

ROS as middleware for AI-enabled robotic systems
Integration of perception modules into ROS
Handling perception data in ROS pipelines
Simulation-based development for AI-driven robotics

Authentic Intelligence Reflection:

- Role of human supervision in robotic autonomy
- Reliability and robustness in communication systems
- Handling noisy and uncertain sensor data
- Simulation-to-real challenges and validation

Mini Project (Mandatory)

Students will design and implement a ROS-based robotic system (simulation or hardware-based).

Minimum Requirement (Mandatory for all)

- ROS-based system with:
  - Sensor input
  - Inter-node communication
  - Basic control or decision logic

Example Problem Statements:

- ROS-based obstacle avoidance robot
- Sensor-driven mobile robot simulation
- ROS-based teleoperation with feedback
- Basic perception-integrated robotic system

Advanced Track (Optional / High-grade band)

- Integration with SLAM or navigation stack
- Multi-node system with modular design

Evaluation Focus:

- System design and architecture
- Correct ROS implementation
- Integration quality
- Debugging and validation

Recommended Tools

- ROS2
- Gazebo
- RViz
- Python
- Ubuntu Linux environment

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

AI Tool Usage for Teaching and Learning

AI tools will be used in this course as structured pedagogical support to enhance understanding, system design, and problem-solving.

Instructor Use (AI as Co-Teacher)

- Generate alternative explanations, examples, and case studies
- Demonstrate system workflows, debugging strategies, and design variations

Student Use (AI as Teaching Assistant)

- Concept clarification and understanding
- Code assistance and debugging (with explanation)
- System design exploration and documentation support

Guidelines for Use

- All AI-assisted work must be understood and explainable by the student
- Blind use of AI-generated content is not acceptable
- Students must critically evaluate AI outputs and identify errors or limitations

Evaluation Integration

- Viva and project discussions will assess individual understanding
- Students may be asked to explain or modify AI-assisted solutions

Suggested Reading

Text Books

1. Quigley, Morgan; Gerkey, Brian; Smart, William D., Programming Robots with ROS, 1st Edition, O'Reilly Media, 2015.
2. Joseph, Lentin, Robot Operating System for Absolute Beginners, 1st Edition, Apress, 2018.

Reference Books

1. Fairchild, Carol & Harman, Thomas L., ROS Robotics by Example, 2nd Edition, Packt Publishing, 2017.
2. Martinez & Fernandez, Mastering ROS for Robotics Programming, 3rd Edition, Packt, 2021.
3. Bassa, Marco Matteo, ROS2 Best Practices, 1st Edition, 2023.

E-Resources

1. https://docs.nvidia.com/learning/physical-ai/getting-started-with-isaac-sim/latest/index.html
2. https://docs.nvidia.com/learning/physical-ai/getting-started-with-isaac-ros/latest/index.html
3. https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-OV-02+V1

****

AI FOR AUTONOMOUS VEHICLES & SMART MOBILITY

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	:	25AM3610	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 Hours
L–T–P–S		2–0–0-2	**CIE + SEE**	:	60 + 40 Marks

Course Learning Objectives:

This course will enable students to:

1. Understand AI-driven architecture of autonomous vehicles and smart mobility systems.
2. Apply machine learning and deep learning techniques for perception, prediction, and decision-making.
3. Analyze localization, planning, and control using AI-based approaches.
4. Develop AI-enabled solutions for real-world smart mobility applications.
5. Evaluate safety, ethics, and societal impacts of AI in transportation systems.

Course Vision

AI for Autonomous Vehicles & Smart Mobility introduces students to the integration of artificial intelligence with modern transportation systems. The course covers perception using deep learning, decision-making using machine learning models, and intelligent path planning for autonomous navigation.

Course Outcome

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Explain AI-based architecture of autonomous vehicles	L2
2	Apply ML/DL models for perception and prediction tasks	L3
3	Analyze AI-driven localization and path planning techniques	L4
4	Develop intelligent mobility solutions using AI	L3
5	Analyze ethical, safety, and societal implications of AI systems	L4

Course Modules :

Module I: AI in Smart Mobility & Autonomous Systems	**09 Hours**

Evolution of mobility, levels of automation, AV architecture, Role of AI & ML in daily transportation, introduction to AI in transportation, intelligent transportation systems, AI pipeline (data → model → decision).

Applications: Self-driving cars, smart cities, ride-sharing platforms.

Module II : AI-Based Perception Systems	**09 Hours**

Sensors (Camera, LiDAR, RADAR), sensor fusion, Computer Vision basics for autonomous vehicles, image processing, Convolutional Neural Networks (CNNs), object detection (YOLO, SSD), semantic segmentation. ***Applications:*** Lane detection, obstacle detection, and pedestrian recognition under different environmental scenario.

UNIT – III : AI for Localization & Prediction	**09 Hours**

Simultaneous Localization and Mapping (SLAM), neural networks, supervised learning: regression & classification in mobility systems, time-series prediction, traffic forecasting. ***Applications:*** GPS enhancement, traffic prediction, recommendations, and route optimization.
UNIT – IV : AI-Based Planning, Control & Connectivity **09 Hours**
Path planning (Dijkstra, A*), reinforcement learning, decision-making models, trajectory planning, V2X communication, Edge AI in smart mobility, 5G mobility**,** Real-life examples: Smart traffic signals, emergency vehicle prioritization. ***Applications***: Autonomous navigation, smart traffic signals, connected vehicles, V2G, G2V.
UNIT – V : AI Safety, Ethics & Future Mobility **09 Hours**
Safety challenges in autonomous vehicles, Explainable AI (XAI), safety in AI systems, bias in ML models, ethical decision-making, legal frameworks, sustainability. Future trends: Electric vehicles, shared mobility, AI-based smart city ecosystems. ***Applications:*** Autonomous cars, delivery robots, smart city ecosystems, and personalized recommendations.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO1	3	2	1	-	2	-	1	-	-	-	2	1	1
CO2	3	3	2	1	3	-	-	-	-	-	2	2	2
CO3	3	3	3	2	3	-	-	-	-	-	2	2	2
CO4	3	3	3	2	3	1	1	2	1	-	2	2	3
CO5	3	3	2	2	2	2	3	1	1	1	3	2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI INTEGRATION & AUTHENTIC INTELLIGENCE REFLECTION:

- **AI Integration:**
  - Module-1: AI-driven mobility system analysis
  - Module-2: Deep learning for perception (CNN models)
  - Module-3: ML models for prediction and forecasting
  - Module-4: Reinforcement learning for navigation
  - Module-5: Explainable AI and ethical AI systems.

- **Authentic Intelligence Reflection:**
  - Module-1: Evaluate reliability of AI perception models
  - Module-2: Identify bias in datasets
  - Module-3: Analyze safety-critical decisions
  - Module-4: Reflect on ethical dilemmas
  - Module-5: Assess sustainability in AI mobility.

RESEARCH INTEGRATION TEACHING

1. Paper review on autonomous driving datasets (KITTI, Waymo, TiAND).

2. Case studies on smart mobility systems.
3. Mini presentations on emerging AI mobility technologies.
4. AI-assisted research using tools for analysis.
5. Critical evaluation of safety and ethical issues.

ASSESSMENT SCHEME:

1. **"Internal Assessment (IA)"** refers to the Continuous Internal Assessment (CIA) with a weightage of 60%. It includes:
    a. **Mid Semester Examination (MSE):** One descriptive examination to evaluate conceptual understanding of AI fundamentals and problem-solving approaches.
    b. **Continuous Assessment Tools (CAT – Six Components):** Includes activities such as quizzes, surprise tests, case studies, group discussions, project-based learning, seminars, industrial visits, reflection notes, model building, e-course certifications, or open book assignments.
    c. **AI-Powered Assignment (APA):** Students utilize AI tools for solving problems while demonstrating Authentic Intelligence through critical evaluation, validation of AI outputs, ethical reasoning, and responsible decision-making.
2. **"External Assessment (EA)"** refers to the Semester End Examination (SEE) with a weightage of 40%. This examination evaluates students' overall understanding of AI concepts, application skills, and their ability to analyze and solve problems effectively.

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

AI-POWERED ASSIGNMENTS:

AI^2 Pedagogy: It is a structured teaching-learning approach where students use AI tools for problem-solving while simultaneously developing Authentic Intelligence through critical evaluation, ethical reasoning, and responsible decision-making.

Assignment 1: AI Perception Model Evaluation

Students prompt an AI tool:

"" Generate code for lane detection using CNN."

Students must:

1. Analyze accuracy
2. dataset limitations
3. improve performance

Assignment 2: Traffic Prediction Model Verification

- Students ask AI:

  " Compare ML models (Linear Regression vs Neural Networks)."

- Students must verify:

  1. Evaluate accuracy and edge cases

Assignment 3: AI Debugging Challenge

- Debug faulty object detection or prediction model
- Students must:

  1. Identify the bug
  2. Fix the code
  3. Explain the error

Assignment 4: Mobility Data Visualization

- Students use AI visualization tools to:

  1. Analyze real-world traffic datasets

Assignment 5: AI-Based Path Planning Study

1. Compare classical algorithms vs AI-based approaches

Suggested Reading:

Primary Textbooks

1. Mahajan, Rupali A., Rajesh Dey, Parikshit N. Mahalle, and Vivek S. Deshpande, AI-Driven Transportation Systems: Real-Time Applications & Related Technologies. Springer, 2025.
2. Houbing Song, Sabina Jeschke, et al., Smart Transportation: AI Enabled Mobility and Autonomous Driving. CRC Press, 2021.
3. Dayyala, N. M. Kottayi, and R. B. Mallick, Machine Learning in Transportation: Applications with Examples and Codes. De Gruyter, 2024.

Additional Resources

1. IEEE research articles
2. Hussein T. Mouftah, Melike Erol-Kantarci, Sameh Sorour, onnected and Autonomous Vehicles in Smart Cities. CRC Press, 2020.
3. Dominique Paret and Hassina Rebaine, Autonomous and Connected Vehicles: Network Architectures from Legacy Networks to Automotive Ethernet. Wiley, 2022.
4. Y. Wang, Z. Cui, and R. Ke, Machine Learning for Transportation Research and Applications. Elsevier, 2023.

E-Resources : (Recommended Tools)

- Python/OpenCV
- Tensorflow/ PyTorch
- Scikit-learn
- CARLA Simulator, SUMO Traffic Simulator
- Jupyter Notebook

Activity Based Learning (Suggested Activities in Class)

1. Group discussion: Discuss different AI techniques used in autonomous vehicles such as perception, localization, and path planning to understand real-world mobility challenges.
2. Collaborative Activity: Develop a mini project in teams of 4 students, such as a smart traffic system, lane detection model, or traffic prediction system using AI techniques.

Edge AI and Robotics

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER-VI

Course Code	:	25AM3612	**Credits**	:	03
Hours/Week	:	04	**Total Hours**	:	45
L–T–P–J	:	2–0–0–2	**CIE+SEE**	:	60+40 Marks

Course Vision:

To equip learners with knowledge and practical skills in Edge Artificial Intelligence and Robotics for designing intelligent, autonomous, and real-time systems that solve industrial and societal challenges.

Course Objectives:

This course will enable students to:

1. Understand the fundamentals of Edge AI, embedded systems, and robotics.
2. Learn sensor integration, data acquisition, and real-time decision-making at the edge.
3. Apply machine learning and computer vision techniques in robotic systems.
4. Develop autonomous robotic applications using Edge AI platforms.
5. Explore industrial and smart automation use cases using robotics and AI.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that a teacher can use to accelerate the attainment of the various course outcomes:

1. **Lecture method:** Along with the traditional lecture method, different types of teaching methods may be adopted to develop the course outcomes.
2. **Interactive Teaching:** incorporating brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and role-playing.
3. Showing **video/animation** films to explain the functioning of various concepts.
4. Encourage **collaborative** (group learning) learning in the class.
5. To foster critical thinking by asking higher-order thinking questions in the class in the form of quizzes and writing programs with complex solutions.
6. Showing the **different ways to solve** the same problem and encourage the students to come up with their own creative ways to solve them.

UNIT– I	**10 Hours**

Introduction to Edge AI

Topics: AI Introduction, AI Hardware, Introduction to Jetson Nano, Edge, AIoT, Big Data, Introduction to Parallel Programming, Machine Learning Fundamentals.

Applications: Smart surveillance systems using real-time video analytics, AI-powered IoT devices for home and industrial automation, Edge-based healthcare monitoring and wearable devices, Predictive maintenance using sensor data analytics, Autonomous drones and robots with onboard intelligence, Real-time traffic monitoring and smart city applications, GPU-accelerated data processing for large-scale analytics.

UNIT – II	**08 Hours**

Vision Deep Neural Networks (DNNs)

Topics: Computer Vision, Image Classification, Object Detection, Semantic Segmentation, Pose Estimation

Applications: Face recognition and biometric attendance systems, Autonomous vehicle pedestrian and lane detection, medical image diagnosis and anomaly detection, Smart retail product recognition and inventory tracking, Human pose estimation for fitness and healthcare monitoring, Industrial defect detection in manufacturing, Intelligent

surveillance and activity monitoring.

UNIT – III	09 Hours

Diversity, Ethics, and Security

Topics: Diversity and Bias in Datasets, Ethics and Security, Fairness, Accountability, and Transparency in AI, Privacy Preservation and Data Protection, Adversarial Attacks and AI Model Robustness, Explainable AI (XAI) for Decision Making, Cybersecurity Risks in Edge Devices.

Applications: Fair AI-based hiring and recruitment systems, Secure facial recognition and access control systems, Privacy-preserving healthcare analytics, Fraud detection with transparent decision models, Robust autonomous systems against adversarial attacks, Secure smart home and IoT edge devices, Ethical decision-making in robotics and surveillance systems.

UNIT – IV	09 Hours

Autonomous Robotics

Topics: Agent Architectures, Teaching Machines to Act, Introduction to ROS2, Robot Navigation, Path Planning Algorithms, Obstacle Detection and Avoidance.

Applications: Self-driving delivery robots in campuses and warehouses, Autonomous drones for inspection and surveillance, Smart warehouse robots for material handling, Robot vacuum cleaners with intelligent navigation, Agricultural robots for crop monitoring and spraying, Search-and-rescue robots in hazardous environments, Hospital service robots for medicine and food delivery.

UNIT – V	09 Hours

Reinforcement Learning

Topics: Reinforcement Learning Fundamentals, Markov Decision Process (MDP), Q-Learning and Deep Q-Networks (DQN), Reward Functions and Policy Optimization, Exploration vs Exploitation Strategies, Applications of Reinforcement Learning in Robotics.

Applications: Robot path planning in dynamic environments, Self-learning game-playing agents, Warehouse robots for task optimization, Traffic signal control using adaptive learning, Energy management in smart grids, Autonomous drone navigation and control, Industrial robot arm motion optimization.

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain the concepts of Edge AI, robotics, and embedded intelligence.	L2
2	Analyze sensors, actuators, and communication systems in robotic platforms.	L3
3	Develop AI models for real-time robotic perception and control.	L4
4	Implement autonomous robotic solutions using edge devices.	L5
5	Evaluate industrial applications, ethical issues, and future trends in Edge AI robotics.	L5

Mapping Levels of Cos to POs/PSOs

Table: MappingLevelsofCOstoPOs/PSOs													
COS	**Program Outcomes(POs)**											**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**
CO-1	3	2	-	–	2	2	–	3	–	–	2	2	-
CO-2	2	3	2	–	3	2	–	2	–	–	–	3	2
CO-3	2	3	3	2	3	3	–	3	–	–	–	3	3
CO-4	2	3	3	3	3	3	–	3	2	–	–	3	3
CO-5	2	2	-	2	2	3	–	3	2	2	2	2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

1. AI is integrated through Edge AI platforms and embedded computing devices such as NVIDIA Jetson Nano and Raspberry Pi. Students learn to deploy lightweight AI models for real-time decision-making directly on robotic hardware. Integration focuses on low-latency processing, offline intelligence, and efficient resource utilization.
2. AI is integrated through robotic sensors, actuators, and IoT communication systems using tools like Arduino IDE and ROS. Students build intelligent sensing systems for environment monitoring and robot control. AI integration emphasizes smart perception, data acquisition, and autonomous response mechanisms.
3. AI is integrated through computer vision and machine learning techniques using OpenCV, TensorFlow Lite, and YOLO models. Students develop systems for object detection, face recognition, and robotic vision. Integration focuses on enabling robots to interpret and react to visual information in real time.
4. This module integrates AI through autonomous navigation and reinforcement learning using Gazebo and ROS. Students implement path planning, obstacle avoidance, and self-learning robotic agents. AI is applied to create adaptive robots capable of operating in dynamic environments.
5. AI is integrated into industrial and service robotics through collaborative robots, predictive maintenance, and smart automation using NVIDIA Isaac platforms and cloud-edge systems. Students explore secure, ethical, and scalable robotic AI solutions. Integration focuses on Industry 4.0, healthcare robotics, and future intelligent autonomous systems.

TEXT BOOKS:

1. Murphy, R. R. (2019). *Introduction to AI robotics*. MIT press.
2. Vermesan, O., & Marples, D. (2024). *Advancing Edge Artificial Intelligence: System Contexts*. River Publishers.

REFERENCE BOOKS:

1. Pickover, C. A. (2024). *Artificial intelligence: an illustrated history: from medieval robots to neural networks*. Hachette UK.
2. Khang, A. (Ed.). (2024). *Medical Robotics and AI-Assisted Diagnostics for a High-Tech Healthcare Industry*. IGI global.

E-RESOURCES:

1. https://developer.nvidia.com/edge-ai-robotics-teaching-kit-syllabus
2. https://docs.ros.org/
3. https://opencv.org/university/free-opencv-course/?srsltid=AfmBOorQQWQA-Dwg0eibzbCI4LbZ5moQ6Mkrn5016jUvFYBJTKKbQLDD

4. https://www.edx.org/learn/embedded-systems
5. https://www.mathworks.com/solutions/electrical-computer-engineering/robotics.html

RECOMMENDED TOOLS

- Python
- OpenCV
- TensorFlow Lite
- ROS
- Raspberry Pi
- NVIDIA Jetson Nano
- Gazebo
- MATLAB

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

MULTI AGENT AI SYSTEM

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER-VI

Course Code	:	25AM3613	**Credits**	:	03
Hours/Week	:	03	**Total Hours**	:	45
L–T–P–J	:	2–0–0–2	**CIE+SEE**	:	60+40 Marks

Course Vision:

To develop learners' understanding of intelligent and multi-agent systems, enabling them to analyze, design, and apply autonomous agent-based solutions for complex real-world problems with consideration of coordination, communication, and ethical implications.

Course Objectives:

This course will enable students to:

1. Understanding the principles behind intelligent agents and agent-oriented systems.
2. Knowledge of architectural models enabling autonomous agent behaviour.
3. Familiarity with cooperation, coordination, and communication in multi-agent societies.
4. Awareness of ethical implications and governance challenges of autonomous agents.
5. Capability to conceptualize and build agent-based solutions for practical problems.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. **Lecture method** along with traditional lecture method, different type of teaching methods may be adopted to develop the course outcomes.
2. **Interactive Teaching:** incorporating brainstorming, discussing, group work, focused listening, formulating questions, note taking, annotating, and role playing.
3. Showing **Video/animation** films to explain functioning of various concepts.
4. Encourage **Collaborative** (Group Learning) Learning in the class.
5. To make **Critical thinking**, asking higher order Thinking questions in the class in the form of Quiz and writing programs with complex solutions.
6. Showing the **different ways to solve** the same problem and encourage the students to come up with their own creative ways to solve them

MODULE–I	**09 Hours**

Intelligent Agents vs Conventional Expert Systems

Topics: Agents and objects, agents versus expert systems, agents versus distributed systems, characteristics of intelligent agents, agent environments and environment properties, typical application domains of agent systems.

MODULE – II	**09 Hours**

Intelligent Agent Architectures

Topics: Abstract architectures for intelligent agents, task environments and agent functions, design of intelligent agents, reasoning agents and BDI architecture, reactive agents and subsumption architecture, hybrid agents such as PRS architecture, Multi-layered agent architectures including InterRAP.

MODULE – III	**09 Hours**

Multi-Agent Systems and Interaction Models

Topics: Definition and characteristics of multi-agent systems, classification of multi-agent interactions, cooperative versus non-cooperative agents, zero-sum and non-zero-sum interactions, Prisoner's Dilemma, Axelrod's experiments on cooperation, self-interested agents and strategic interaction.

MODULE – IV	**09 Hours**

Agent AI, Communication, Coordination and Ethics **Topics:** Principles of agent communication, speech acts and communication semantics, agent communication languages, Knowledge Query and Manipulation Language (KQML), Knowledge Interchange Format (KIF), FIPA framework and interaction protocols, ontologies for agent communication, ethical issues in autonomous agent communication, accountability and transparency in agent interactions .	
MODULE – V	**09 Hours**
Cooperative Multi-Agent Systems and Ethical Framework **Topics:** Cooperative Distributed Problem Solving (CDPS), Partial Global Planning, coordination and coherence in agent societies, negotiation and argumentation among agents, auction-based multi-agent systems, ethical risks in multi-agent decision-making, governance and societal impact of agent-based systems, real-world applications of intelligent agents and multi-agent systems .	

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain the concepts of intelligent agents and applications suitable for agent-oriented solutions.	L2
2	Describe agent architectures and mechanisms enabling autonomous intelligent behaviour.	L3
3	Analyze interaction and coordination strategies in multi-agent systems.	L4
4	Evaluate ethical, social, and governance challenges in agent-based systems.	L4
5	Design and develop a basic intelligent or multi-agent system for a real-world problem	L3

Mapping Levels of COs to POs/PSOs

Table: Mapping Levels of COs to POs/PSOs													
COS	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO-1	3	2	–	1	1	–	–	–	–	–	–	2	1
CO-2	2	3	2	2	2	–	–	–	1	–	–	2	2
CO-3	2	3	3	2	2	–	–	–	2	1	–	3	2
CO-4	2	2	1	2	1	2	1	2	1	1	–	2	2
CO-5	2	3	3	2	2	2	1	2	2	2	1	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI Integration:

AI integration in this unit is achieved through conceptual exploration and interactive demonstrations using AI-based tools and simulators. Students analyze real-world intelligent agent applications such as chatbots, recommendation systems, and autonomous assistants. They compare traditional expert systems with modern AI agents through case studies and hands-on interactions, enabling them to understand agent characteristics, environments, and practical application domains..

This unit integrates AI through modeling and simulation of various agent architectures such as BDI, reactive, and hybrid systems. Students engage in practical exercises to design simple agents and observe behavior differences across architectures. Tools and frameworks for agent development are introduced to demonstrate subsumption and layered architectures, helping learners visualize decision-making and autonomy in intelligent systems.

AI integration focuses on simulating multi-agent environments where students observe cooperation, competition, and strategic interactions. Practical activities include modeling scenarios like Prisoner's Dilemma and analyzing agent strategies. Simulation tools are used to study coordination and interaction patterns, enabling students to understand real-world applications such as distributed systems, traffic management, and economic systems.

In this unit, AI integration is achieved through implementation and analysis of agent communication protocols using standard frameworks. Students explore communication languages and simulate agent interactions using message-passing systems. Ethical considerations such as transparency, accountability, and bias in autonomous agents are discussed through case studies, encouraging critical thinking about responsible AI deployment.

AI integration in this unit emphasizes real-world application development and problem-solving using cooperative multi-agent systems. Students design small-scale agent-based solutions for distributed problem solving, negotiation, and coordination scenarios. Practical exposure includes analyzing real-world systems such as smart grids, autonomous vehicles, and e-commerce agents, along with evaluating ethical risks and governance challenges in multi-agent decision-making.

TEXT BOOKS:

1. Michael Wooldridge, *An Introduction to MultiAgent Systems*, 2nd Edition, Wiley, 2009.
2. Stuart J. Russell and Peter Norvig, *Artificial Intelligence: A Modern Approach*, 4th Edition, Pearson, 2021.

REFERENCE BOOKS:

1. François Chollet, *Deep Learning with Python*, 2nd Edition, Manning Publications, 2021.
2. Aurélien Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 3rd Edition, O'Reilly Media, 2022.

E-RESOURCES:

- FIPA Agent Standards : https://www.fipa.org
- MIT OpenCourseWare – Multi-Agent Systems : https://ocw.mit.edu
- NVIDIA AI Resources: https://developer.nvidia.com/ai
- Coursera – Multi-Agent Systems & AI Courses: https://www.coursera.org
- Stanford Online AI Courses: https://online.stanford.edu

RECOMMENDED TOOLS

Python, Jupyter Notebook, Google Colab, JADE, SPADE, TensorFlow, PyTorch, NetLogo.

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

INTELLIGENT COMPUTATIONAL IMAGING AND RECONSTRUCTION

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER - VI

Course Code	:	25AM3614	**Credits**	:	03
Hours / Week	:	05	**Total Hours**	:	45+30
L–T–P–J	:	2–0–0–2	**CIE+SEE**	:	60+40 Marks

Course Vision:

To develop the ability to model, analyze, and design advanced imaging systems by integrating physical models, computational methods, and AI-driven techniques in **Computational Imaging**, enabling efficient and intelligent visual data reconstruction.

Course Objectives:

This course will enable students to:

1. Understand image formation and computational imaging principles.
2. Apply inverse problem solving and optimization techniques.
3. Analyze advanced imaging methods such as HDR and super-resolution.
4. Explore modern imaging modalities and sensing systems.
5. Design AI-based imaging solutions for real-world applications.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. **Lecture method** along with traditional lecture method, different type of teaching methods may be adopted to develop the course outcomes.
2. **Interactive Teaching:** incorporating brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Showing **Video/animation** films to explain functioning of various concepts.
4. Encourage **Collaborative** (Group Learning) Learning in the class.
5. To make **Critical thinking**, asking Higher order Thinking questions in the class in the form of Quiz and writing programs with complex solutions.
6. Showing the **different ways to solve** the same problem and encourage the students to come up with their own creative ways to solve them

MODULE – I	**09 Hours**

Foundations of Computational Imaging

Introduction to **Computational Imaging**, imaging system overview, image formation models, radiometry and photometry, sampling and quantization, linear shift-invariant systems, convolution and correlation operations, frequency domain analysis using the **Fourier Transform**.

Applications: Image acquisition and enhancement in medical imaging, computational photography, remote sensing systems, and scientific imaging applications.

MODULE – II	**09 Hours**

Inverse Problems and Optimization in Imaging

Forward and inverse imaging problems, ill-posed problems and their challenges, regularization techniques such as **Tikhonov Regularization**, L1 regularization, and total variation, Bayesian estimation methods, **Maximum A Posteriori (MAP) Estimation** and **Minimum Mean Square Error (MMSE) Estimation**, optimization techniques including **Gradient Descent**, proximal methods, and **Alternating Direction Method of Multipliers (ADMM)**

Applications: Image reconstruction in medical imaging (CT/MRI), deblurring and denoising in photography, super-resolution imaging, and restoration in scientific and remote sensing applications.

MODULE – III	**09 Hours**

Advanced Computational Imaging Techniques

Advanced techniques in **Computational Imaging** including computational photography, high dynamic range imaging using **HDR Imaging**, super-resolution imaging, image deblurring and denoising methods, compressive sensing and sparse representations, coded aperture imaging, and phase retrieval techniques.

Applications: Enhancement of photographic quality in mobile cameras, medical image reconstruction, astronomical imaging, surveillance systems, and high-resolution imaging in scientific research.

MODULE – IV	**09 Hours**

Advanced Imaging Systems and Modalities

Advanced imaging techniques in **Computational Imaging** including light field imaging, lensless and coded imaging systems, computational microscopy, basics of computational tomography such as **Computed Tomography (CT)** and **Magnetic Resonance Imaging (MRI)**, hyperspectral imaging, time-of-flight and depth imaging, and event-based cameras.

Applications: 3D scene reconstruction, biomedical imaging and diagnostics, remote sensing, autonomous systems, and advanced scientific imaging applications.

.

MODULE – V	**09 Hours**

Learning-Based Computational Imaging

Topics: Learning-based approaches in **Computational Imaging**, deep learning techniques for imaging, convolutional neural networks (**Convolutional Neural Networks (CNNs)**) for image reconstruction, plug-and-play priors, physics-informed neural networks, end-to-end imaging system design, self-supervised and unsupervised learning methods for imaging, and ethical and societal considerations in intelligent imaging systems.

Applications: AI-based image reconstruction in medical imaging, intelligent denoising and super-resolution, automated visual inspection systems, and smart imaging solutions in scientific and industrial domains.

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Understand image formation and computational imaging principles.	L2
2	Apply inverse problem solving and optimization techniques	L4
3	Analyze advanced imaging methods such as HDR and super-resolution	L4
4	Explore modern imaging modalities and sensing systems.	L3
5	Design AI-based imaging solutions for real-world applications.	L5

- **AI Integration:**
  - Module-1: Integration of computational imaging with AI techniques for image reconstruction, denoising, and enhancement using machine learning-based models.
  - Module-2: Use of AI and deep learning models for solving inverse imaging problems, data-driven regularization techniques, and intelligent optimization for faster and more accurate image reconstruction.
  - Module-3: Use of deep learning models for super-resolution, AI-based denoising and deblurring, and intelligent reconstruction techniques for compressive sensing and advanced imaging systems.
  - Module-4: Integration of advanced imaging systems with AI for depth estimation, hyperspectral data analysis, medical image interpretation, and real-time intelligent vision systems.
  - Module-5: Advanced use of deep learning models for end-to-end imaging pipelines, integration of physics-based models with AI, and intelligent data-driven reconstruction techniques.

Mapping Levels of COs to POs / PSOs

Table: Mapping Levels of COs to POs / PSOs

COS	Program Outcomes (POs)											PSOs		
	1	2	3	4	5	6	7	8	9	10	11	1	2	3
CO-1	3	2	2	–	2	2	–	3	–	–	2	3	3	3
CO-2	3	3	2	–	3	3	–	3	–	–	–	3	3	3
CO-3	3	2	2	–	3	3	–	3	–	–	–	3	3	3
CO-4	3	2	3	2	3	3	–	3	2	–	–	3	3	3
CO-5	3	–	2	–	3	3	–	3	2	2	2	3	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Veeraraghavan – Computational Imaging
2. Szeliski – Computer Vision

REFERENCE BOOKS:

E-RESOURCES:

1. MIT OpenCourseWare
2. Stanford Algorithms Lectures
3. IEEE research articles

RECOMMENDED TOOLS

Python 3.10+, NumPy, Pandas, Scikit-learn, PyTorch, Hugging Face Transformers
IBM AI Fairness 360 (AIF360) — https://aif360.mybluemix.net
Aequitas Bias Audit Toolkit — http://aequitas.dssg.io
SHAP — https://shap.readthedocs.io | LIME — https://github.com/marcotcr/lime
OpenAI Gymnasium (RL) — https://gymnasium.farama.org

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

AUTONOMOUS ROBOTIC SYSTEMS

Professional Elective

Subject Code	:	25AM3615	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 Hours
L–T–P–S	:	2–0–0–2			

Course Learning Objectives:

Students will:

1. Understand the architecture and design principles of autonomous robotic systems.
2. Apply perception and localization techniques under real-world sensing uncertainty.
3. Analyze trade-offs between accuracy, computation, and robustness in autonomy pipelines.
4. Design planning and navigation strategies under dynamic and constrained environments.
5. Integrate multiple subsystems into a cohesive autonomous system.
6. Evaluate system performance under uncertainty, noise, and failure conditions.

Course Description :

To enable students to design, implement, and evaluate autonomous robotic systems under uncertainty by integrating perception, localization, planning, and system-level decision-making frameworks for real-world deployment.

Course Outcome	Description	Bloom's Taxonomy Level
CO1	Analyze and describe the architecture of autonomous robotic systems	L4
CO2	Apply perception techniques for environment understanding under uncertainty	L3
CO3	Analyze and implement localization and mapping methods	L4
CO4	Design planning and decision-making strategies for autonomous navigation	L6
CO5	Develop and evaluate an integrated autonomous robotic system	L6

UNIT – I : Perception for Autonomous Systems	**03 Hours**

Sensors for robotics (LiDAR, camera, IMU), data acquisition, feature extraction, object detection (classical + deep learning overview), basic sensor fusion concepts. **Applications:** Mobile robots, autonomous vehicles, service robotics	
NVIDIA DLI Self-Paced Courses Mapped to Module-1	**06 Hours**
Getting Started with Deep Learning - Module: Image classification using convolutional neural networks Building Real-Time Video AI Applications - Modules: Object detection in images and video streams, Deep learning inference workflows for visual data Extension: - Application of deep learning-based perception in robotic systems - Integration of object detection outputs into robotic perception pipelines - Role of real-time perception in autonomous systems	
UNIT – II : Localization and Mapping	**09 Hours**
Coordinate frames, odometry, probabilistic localization (Kalman Filter, Particle Filter concepts), mapping basics, occupancy grid maps, SLAM overview (visual and LiDAR-based). Nvidia DLI: Getting Started with Deep Learning - Modules: Feature extraction using convolutional neural networks, Learned visual representations for image-based tasks **Applications:** Indoor navigation, warehouse robotics, mapping unknown environment	
UNIT – III : Planning and Decision Making	**09 Hours**
Graph-based planning methods (A*, Dijkstra), sampling-based planning (RRT, PRM), and hierarchical decision-making frameworks incorporating cost functions and constraints. **Applications:** Robot navigation, path optimization, exploration tasks	
UNIT – IV : Navigation and Execution	**09 Hours**
Navigation stack architecture (ROS-based), global vs local planning, trajectory generation, execution constraints, obstacle avoidance in dynamic environments, system limitations. Nvidia DLI: Building Real-Time Video AI Applications - Modules: Real-time video inference workflows, Processing visual data streams for object detection **Applications:** Autonomous mobile robots, drone navigation	
UNIT – V : System Integration and Evaluation	**09 Hours**

System-level architecture, integration of perception–localization–planning–execution, ROS pipelines, simulation-to-real transfer, evaluation metrics and benchmarking, failure case analysis, case studies (autonomous vehicles, warehouse robots). Nvidia DLI: Building Real-Time Video AI Applications - End-to-end video AI pipelines - Model inference and deployment workflows **Applications:** End-to-end autonomous systems

Mapping Levels of COs to POs / PSOs													
COs				**Program Outcomes (POs)**								**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**
CO1	3	3	1	-	1	-	-	-	-	-	1	3	1
CO2	3	3	2	2	3	-	-	1	-	-	2	3	3
CO3	3	3	2	3	2	-	-	1	-	-	2	3	3
CO4	3	3	3	2	3	-	1	2	1	1	2	3	3
CO5	3	3	3	2	3	1	1	2	2	1	2	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI Integration & Authentic Intelligence Reflection (AI^2 Pedagogy)

AI Integration:

Module-1: Vision-based perception using deep learning models

Module-2: Learning-based localization (overview only)

Module-3: Reinforcement learning for decision-making (conceptual exposure)

Module-4: Learning-based control (high-level overview)

Module-5: Modular AI pipelines in robotics

Authentic Intelligence Reflection:

Module-1: Limitations of perception systems, uncertainty and noise handling

Module-2: Uncertainty modeling, failure modes in localization

Module-3: Trade-offs between optimality and computational efficiency

Module-4: Safety, robustness, and real-world deployment constraints

Module-5: System reliability, ethical deployment, human-in-the-loop systems.

Mini Project (Mandatory)

Students will design and implement a basic autonomous robotic system (simulation or hardware-based).

Minimum Requirement (Mandatory for all)

- Autonomous navigation with:
- Perception input
- Basic planning
- Obstacle avoidance

Advanced Track (Optional / High-grade band)

- SLAM-based navigation OR
- Dynamic environment decision-making OR
- Vision-based autonomy

Evaluation Focus

- System design clarity
- Level of autonomy achieved
- Robustness under uncertainty
- Failure handling and recovery

Recommended Tools

- ROS / ROS2
- Gazebo / Webots
- Python / C++
- OpenCV
- RViz
- SLAM packages (e.g., Gmapping, Cartographer)

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

AI Tool Usage for Teaching and Learning

AI tools will be used in this course as structured pedagogical support to enhance understanding, system design, and problem-solving.

Instructor Use (AI as Co-Teacher)

- Generate alternative explanations, examples, and case studies
- Demonstrate system workflows, debugging strategies, and design variations

Student Use (AI as Teaching Assistant)

- Concept clarification and understanding
- Code assistance and debugging (with explanation)
- System design exploration and documentation support

Guidelines for Use

- All AI-assisted work must be understood and explainable by the student
- Blind use of AI-generated content is not acceptable
- Students must critically evaluate AI outputs and identify errors or limitations

Evaluation Integration

- Viva and project discussions will assess individual understanding
- Students may be asked to explain or modify AI-assisted solutions

Suggested Reading

1. Thrun, S., Burgard, W., Fox, D., Probabilistic Robotics, 2005, MIT Press.
2. LaValle, S. M., Planning Algorithms, 2006, Cambridge University Press.
3. Siciliano, B., Khatib, O., Springer Handbook of Robotics, 2016, Springer.
4. Corke, P., Robotics, Vision and Control, 2017, Springer.
5. Selected recent IEEE/ICRA/IROS research papers on autonomous systems
6. https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-FX-01+V1
7. https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-IV-01+V1

****

AI FOR HEALTHCARE

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER –VI

Course Code	: 25AM3616	**Credits**	: 3
Hours / Week	: 03 Hrs	**Total Hours**	: 45 Hrs
L–T–P-J	: 2–0–0-2		

Prerequisites:

Basic understanding of healthcare systems and/or introductory computing concepts.

Course Objectives:

5. To introduce artificial intelligence concepts relevant to healthcare
6. To understand clinical data types and AI-driven decision support
7. To explore machine learning models used in medical applications
8. To analyze ethical, legal, and governance issues in medical AI
9. To develop skills for evaluating and validating AI solutions in healthcare

Teaching-Learning Process (General Instructions)

7. **Case-based learning**
8. **Demonstrations of AI tools in healthcare**
9. **Clinical scenario discussions**
10. **Assignments and quizzes**
11. **Mini-project / case study**

UNIT – I	**7 Hours**

IINTRODUCTION: Introduction to AI in Healthcare

Overview of AI in medicine, Evolution of intelligent systems in healthcare, Role of AI in clinical workflows, Intelligent systems in clinical workflows, Landmark AI applications in medicine, Smart vs traditional healthcare systems

UNIT – II	**8 Hours**

Clinical Data & Machine Learning Foundations

Healthcare data sources and types, Data acquisition standards and challenges, Supervised, unsupervised, and reinforcement learning, ML pipeline: features, Feature extraction and dimensionality reduction, models, evaluation metrics.

UNIT – III	**10 Hours**

AI Applications in Healthcare and case studies

Screening and early detection, Diagnosis, grading, and subtyping , prognosis and outcome prediction, Surgery, drug delivery, and precision medicine, Treatment planning and intervention, ICU, ER, and hospital resource management, Patient and hospital management system, Bio-signals, medical images, text, molecular data.

UNIT – IV	**08 Hours**
Advanced AI & Emerging Trends Deep learning in healthcare, Generative AI and foundation models, AI-powered decision support systems, Wearables, chatbots, and remote monitoring, , Tree-based, ML, and deep learning models, Model interpretation and inference	
UNIT – V	**07 Hours**
Ethics, Governance & Responsible AI Ethical challenges and bias, Privacy, security, and data protection, AI as Software as Medical Device (SaMD), Regulatory frame works and governance frameworks	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
CO1	Understand AI fundamentals in healthcare	L2
CO2	Analyze clinical data using ML concepts	L4
CO3	Apply AI models to healthcare use-cases	L3
CO4	Evaluate ethical and regulatory issues	L4
CO5	Assess AI tools for real-world deployment	L5

COs COs	**Program Outcomes (POs)**												**PSOs**	
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	3	2	–	–	2	–	–	–	–	–	–	1	2	1
CO2	2	3	–	2	3	–	–	–	–	–	–	1	3	2
CO3	–	2	3	3	2	–	–	–	–	–	–	2	3	3
CO4	2	2	–	–	3	2	1	3	–	–	–	2	2	2
CO5	–	2	3	2	3	–	–	–	2	1	–	2	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS (TB):

1. S. Dash, B. Shickel, M. A. Bray, and S. Saria, Machine Learning for Healthcare, MIT Press, Cambridge, 2021
2. E. Topol, Deep Medicine: How Artificial Intelligence Can Make Healthcare Human Again, Basic Books, New York, 2019.
3. A. Holzinger, Explainable AI in Healthcare, Springer Nature, Cham, 2022.
4. K. D. Kulkarni and P. G. Bhat, Artificial Intelligence and Machine Learning in Healthcare, Academic Press (Elsevier), 2020.

REFERENCE BOOKS:

5. Rajkomar, A., Dean, J., & Kohane, I., Machine Learning in Medicine, Cambridge University Press, 2019.
6. J. Jiang, J. Zhi, A. Ni, & D. Shen, Advanced Methods and Tools for ECG Data Analysis, Springer, 2019 — *(for AI in signal processing)*
7. Biomedical Informatics: Computer Applications in Health Care and Biomedicine, Springer, 2018.
8. L. L. Popescu, Deep Learning and Convolutional Neural Networks for Medical Imaging and Clinical Informatics, IGI Global, 2021.
9. K. Friston et al., Statistical Parametric Mapping: The Analysis of Functional Brain Images, Academic Press, 2011 — *(For neuroimaging AI)*
10. D. S. Char, N. Shah, and D. Magnus, Implementing Machine Learning in Healthcare: From Ethical Challenges to Legal Frameworks, Cambridge University Press, 2020.
11. G. Litjens et al., Deep Learning for Medical Image Analysis, Elsevier, 2017.

E-Resources:

- IEEE, springer, mdp, science direct
- https://ocw.mit.edu/courses/hst-921-information-technology-in-the-health-care-system-of-the-future-spring-2009/
- https://www.nlm.nih.gov/
- https://www.who.int/
- https://ocw.mit.edu/search/?q=Health%20care

Activity Based Learning (Suggested Activities in Class)

1. Case study analysis
2. Group discussions
3. Mini-project / seminar
4. Quiz.,

Link of References of colleges/universities used for preparing the syllabus

1. https://web.stanford.edu/class/archive/biods/biods220/biods220.1222/?
2. https://smhs.gwu.edu/course/artificial-intelligence-applications-healthcare
3. https://apollouniversity.edu.in/wp-content/uploads/2024/07/Msc-HIA-TOTAL-SYLLABUS.pdf?
4. https://www.eicta.iitk.ac.in/courses/artificial-intelligence-in-healthcare
5. https://www.nptelprep.in/courses/106106238/materials

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20

Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

BUILDING TRANSFORMER-BASED NATURAL LANGUAGE PROCESSING APPLICATIONS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Course Code	:	25AM3617	**Credits**	:	3
Hours / Week	:	3	**Total Hours**	:	45(15 hrs for Project)
L–T–P-J	:	2–0–0-2	**CIE**	:	100 MARKS

Prerequisites:

Proficiency in a C programming language.

Course Objectives:

1. To provide a strong foundation in optimization concepts relevant to Natural Language Processing tasks.
2. To enable students to formulate NLP problems as mathematical optimization problems.
3. To study and apply classical and modern optimization algorithms used in NLP models.
4. To analyze optimization challenges in deep learning and transformer-based NLP architectures.
5. To develop the ability to select, implement, and evaluate efficient optimization strategies for large-scale NLP systems.

Teaching-Learning Process (General Instructions)

1. Classroom lectures using presentations and board work to explain theoretical concepts and mathematical foundations.
2. Proble m-solving sessions and numerical examples to strengthen understanding of optimization techniques in NLP.
3. Case studies and discussion of recent research papers related to optimization in modern NLP models.
4. Hands -on demonstrations, assignments, and mini-projects using Python-based NLP and deep learning frameworks.
5. Continuous assessment through quizzes, seminars, and student presentations to encourage self-learning and critical thinking.

MODULE– I	7 hours

Fundamentals of NLP and Optimization:Introduction to NLP tasks: text classification, sequence labeling, machine translation ,Mathematical formulation of NLP problems ,Review of linear algebra, probability, and information theory for NLP,Loss functions in NLP: cross-entropy, negative log-likelihood,Convex and non-convex optimization problems.

MODULE – II	7 hours

Classical Optimization Techniques for NLP: Gradient Descent and its variantsBatch-Stochastic, and Mini-batch Gradient Descent,First-order optimization methods,Second-order methods:

Newton's Method, Quasi-Newton (BFGS, L-BFGS)Optimization in traditional NLP models -Logistic Regression,Conditional Random Fields (CRF),Maximum Entropy models.

MODULE – III	7 hours

Optimization in Deep Learning for NLP: Backpropagation and computational graphs,Vanishing and exploding gradients,Advanced optimizers: Momentum, Adagrad, RMSProp, Adam, AdamW,Regularization techniques: dropout, weight decay, early stopping,Optimization for word embeddings: Word2Vec, GloVe.

MODULE – IV	7 hours

Optimization for Sequence Models and Transformers: Optimization in RNNs, LSTMs, and GRUs,Attention mechanisms and optimization aspects,Transformer architecture and training challenges,Learning rate scheduling and warm-up, strategies,Fine-tuning vs pre-training optimization.

MODULE – V	7 hours

Advanced and Scalable Optimization Techniques: Large-scale and distributed optimization,Gradient clipping and normalization,Mixed precision training,Hyperparameter optimization methods,Case studies from real-world NLP systems.

Course Outcomes:

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Formulate NLP problems as optimization tasks	C6
2	Apply classical optimization techniques to NLP models	C3
3	Implement and analyze optimization strategies in deep NLP models	C4
4	Optimize transformer-based language models	C5
5	Evaluate optimization methods for scalable NLP applications	C5

Mapping Levels of COs to POs / PSOs

Table: Mapping Levels of COs to POs / PSOs													
COS	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO-1	3	1	–	–	–	–	–	–	–	–	–	1	2
CO-2	3	3	2	2	2	–	–	–	–	–	–	–	2
CO-3	2	3	2	2	2	–	–	–	–	–	–	–	3
CO-4	2	3	2	2	3	–	–	–	1	–	–	–	3
CO-5	1	2	2	3	3	–	–	–	2	1	–	2	2

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

TEXT BOOKS (TB):

1 .Jurafsky, D., & Martin, J. H., *Speech and Language Processing*, 3rd Edition (Draft), Pearson.

2. Goodfellow, I., Bengio, Y., & Courville, A., *Deep Learning*, MIT Press.

REFERENCE BOOKS:

1.Boyd, S., & Vandenberghe, L., *Convex Optimization*, Cambridge University Press. 2.Goldberg, Y., *Neural Network Methods for Natural Language Processing*, Morgan & Claypool.

3.Murphy, K. P., *Machine Learning: A Probabilistic Perspective*, MIT Press.

4.Bottou, L., Curtis, F. E., & Nocedal, J., *Optimization Methods for Large-Scale Machine Learning*, SIAM Review.

E-Resources:

https://blog.truegeometry.com/tutorials/education/15970c4fdbca1fc48c5b412d94e798c0/JSON_TO_ARTCL_Optimization_in_Natural_Language_Processing_in_context_of_Optimiza.html?utm_source=chatgpt.com

Activity Based Learning (Suggested Activities in Class)

1. **Real-world problem solving using group discussion**– Students analyze real NLP problems (text classification, sentiment analysis, machine translation) and discuss suitable optimization strategies.
2. **Role play activity**– Role play of optimizer components such as gradient descent, learning rate scheduler, loss function, and model parameters to understand optimization flow.
3. **Demonstration of solution through programming**– Instructor-led and student-driven demonstrations using Python (PyTorch/TensorFlow) to show optimization techniques like SGD, Adam, and learning rate tuning in NLP models.
4. **Flipped classroom activity**– Students study assigned research papers or video lectures on optimization methods (e.g., AdamW, transformer fine-tuning) and discuss insights

Professional Elective Courses Offering (PEC-IV)											
SL	**Course Type**	**Course Code**	**Course Name**	Teaching Hours / Week				Examination			
				Lecture	**Tutorial**	**Practical**	**Project**	CIE Marks	SEE Marks	Total Marks	Credits
				L	T	P	J				
1	PEC	XXX	Deep Learning for Science and Engineering	2	0	0	2	100	--	100	03
2	PEC	XXX	Ethics, Safety & Governance of Agentic AI	2	0	0	2	100	--	100	03
3	PEC	XXX	Intelligent Video Surveillance Systems	2	0	0	2	100	--	100	03
4	PEC	XXX	Foundation Models for Robotics	2	0	0	2	100	--	100	03
5	PEC	XXX	Blockchain for Data Security	2	0	0	2	100	--	100	03
6	PEC	XXX	Multi modal and Multilingual Language Perception	2	0	0	2	100	--	100	03

DEEP LEARNING FOR SCIENCE AND ENGINEERING

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Course Code	:	25AM4702	**Credits**	:	03
Hours / Week	:	03	**Total Hours**	:	45
L–T–P–J	:	2–0–0–2	**CIE+SEE**	:	60 + 40 Marks

Course Vision:

This course aims to develop a strong foundation in deep learning and its application to scientific and engineering problems. It focuses on integrating data-driven models with physical principles, enabling learners to solve complex real-world problems using scalable, efficient, and intelligent computational techniques.

Course Objectives:

This course will enable students to:

1. Understand fundamental concepts of deep learning and neural networks
2. Gain proficiency in Python and deep learning frameworks
3. Apply deep learning techniques to scientific and engineering problems
4. Explore advanced topics such as neural operators and uncertainty quantification
5. Implement scalable solutions using distributed and high-performance computing.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. **Lecture method** along with traditional lecture method, different type of teaching methods may be adopted to develop the course outcomes.
2. **Interactive Teaching:** incorporating brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Showing **Video/animation** films to explain functioning of various concepts.
4. Encourage **Collaborative** (Group Learning) Learning in the class.
5. To make **Critical thinking**, asking Higher order Thinking questions in the class in the form of Quiz and writing programs with complex solutions.
6. Showing the **different ways to solve** the same problem and encourage the students to come up with their own creative ways to solve them.

UNIT – I	**09 Hours**

Fundamentals of Deep Learning

Topics: Introduction to Deep Learning, A Primer on Python, NumPy, SciPy and Jupyter Notebooks, Deep Learning Networks, A Primer on TensorFlow, PyTorch and JAX, Training and Optimization, Neural Network Architectures

Applications: Image Classification and Object Detection, Chatbots and Text Analysis, Speech Recognition Systems, Deep Learning Model Development, Data Analysis with Python Tools

UNIT – II	**09 Hours**

Neural Differential Equations

Topics: Discovering Differential Equations, Physics-Informed Neural Networks (PINNs)- Part I, Physics-Informed Neural Networks (PINNs)- Part II

Applications: Solving Differential Equations in Engineering Systems, Fluid Dynamics and Aerodynamics Simulation, Heat Transfer and Thermal Analysis, Structural Health Monitoring and Material Modelling, Climate Modelling and Environmental Prediction

UNIT – III	**09 Hours**

Neural Operators

Topics: Introduction to Deep Neural Operators, Mathematical Foundations, Fourier Transform and Inverse Fourier Transform, Deep Operator Networks, Fourier Neural Operators, Learning

Operators for Differential Equations, Implementation using Python and Deep Learning Frameworks **Applications:** Solving Partial Differential Equations in scientific computing, Fluid Flow and Turbulence Modelling, Weather and Climate Prediction, Material Science and Structural Simulation, Real-time Engineering Simulations and Digital Twins	
UNIT – IV	**09 Hours**
Data and Uncertainty **Topics:** Introduction to Multi-Fidelity Data in Machine Learning, Multi-Fidelity Modelling Techniques, Gaussian Process Regression, Multi-Fidelity Deep Neural Networks, Introduction to Uncertainty Quantification, Bayesian Neural Networks, Neural Methods for Uncertainty Estimation. **Applications:** Engineering Design Optimization with Limited Data, Risk Assessment and Decision Making under Uncertainty, Climate and Weather Forecasting with Confidence Intervals, Medical Diagnosis with Uncertainty Estimation, Surrogate Modelling for Complex Simulations	
UNIT – V	**09 Hours**
Scalable Scientific Machine Learning **Topics:** Introduction to Scalable Scientific Machine Learning, Multi-GPU and Distributed Training Concepts, Message Passing Interface for Parallel Computing, Horovod for Distributed Deep Learning, Domain Decomposition Methods, NVIDIA Physics NeMo Core and Framework Overview, Physics NeMo Symbolic for Solving Differential Equations. **Applications:** Large-Scale Scientific Simulations using Multi-GPU Systems, High-Performance Computing for Physics-Based Modelling, Real-Time Fluid Dynamics and Wave Simulations, Parallel Training of Deep Learning Models for Big Data, Digital Twin Development for Complex Engineering Systems	

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain the fundamentals of deep learning and scientific machine learning	L2
2	Apply Python and deep learning frameworks to build models	L3
3	Analyse differential equations using PINNs and neural operators	L4
4	Evaluate model performance with uncertainty quantification methods	L5
5	Design scalable deep learning systems using distributed computing.	L5

Mapping Levels of COs to POs / PSOs

Table: Mapping Levels of COs to POs / PSOs

COS	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2

CO-1	3	2	2	–	2	2	–	2	–	–	2	2	2
CO-2	3	3	3	–	3	3	–	2	–	–	–	3	3
CO-3	3	2	3	–	3	3	–	3	–	–	–	3	3
CO-4	3	2	3	2	3	3	–	3	2	–	–	3	3
CO-5	3	2	3	–	3	3	–	3	2	2	2	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI Integration

1. Application of deep learning models in scientific problem solving
2. Integration of AI with physics-based simulations
3. Automated model training and optimization
4. AI-driven predictive analytics for engineering systems
5. Deployment of intelligent, scalable AI systems

Authentic Intelligence

1. Real-world case studies from science and engineering domains
2. Hands-on implementation using industry tools
3. Problem-solving using real datasets
4. Ethical considerations in AI applications
5. Project-based experiential learning.

TEXT BOOKS:

1. *Deep Learning* – Ian Goodfellow, Yoshua Bengio, Aaron Courville
2. *Pattern Recognition and Machine Learning* – Christopher M. Bishop
3. *Scientific Machine Learning* – Course Notes / Technical Resources

REFERENCE BOOKS:

1. *Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow* – Aurélien Géron
2. *Machine Learning: A Probabilistic Perspective* – Kevin P. Murphy

E-RESOURCES:

https://developer.nvidia.com/teaching-kits/science-and-engineering-syllabus

1. https://www.tensorflow.org/
2. https://pytorch.org/tutorials/
3. https://ocw.mit.edu/courses/6-036-introduction-to-machine-learning-fall-2020/
4. NVIDIA Deep Learning Institute (DLI)
5. Deep Learning with PyTorch (Coursera)
6. TensorFlow 2 Deep Learning Specialization

RECOMMENDED TOOLS

- Python (NumPy, SciPy, Jupyter Notebook)
- TensorFlow / PyTorch / JAX

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

INDUSTRIAL ROBOTICS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	:	25OE3618	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	2–0–0–2			

Course Learning Objectives:

This course will enable students to:

1. Understand the configuration space with specific reference to roboticmotion
2. Understand different types of kinematics used in industrial robotics.
3. To understand motion planning in industrial robotics.
4. Understand Computational Motion Planning and Mobility
5. Understand the concept of grasping and manipulation.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the student's understanding.

UNIT – I :	**08 Hours**

Introduction Configuration Space: Foundations of Robot Motion-Degrees of Freedom of a Rigid Body-Degrees of Freedom of a Robot-Configuration Space Representation Configuration and Velocity Constraints-Task Space and Workspace.

Rigid-Body Motions: Introduction to Rigid-Body Motions-Rotation Matrices-Angular Velocities-Homogeneous Transformation Matrices.

UNIT – II:	**08 Hours**

Forward Kinematics: Forward Kinematics Example- Velocity Kinematics and Statics: Introduction to Velocity Kinematics and Statics-Space Jacobian-Body Jacobian, Inverse Kinematics: Inverse Kinematics of Open Chains.

UNIT – III:	**08 Hours**

Kinematics of Closed Chains: Dynamics of Open Chains- Lagrangian Formulation of Dynamics-Understanding the Mass Matrix, Newton-Euler Inverse Dynamics, Trajectory Generation: Point-to-Point Trajectories-Polynomial Via Point Trajectories-
Time-Optimal Time Scaling.

UNIT – IV:	**07 Hours**

Motion Planning: Overview of Motion Planning, Robot Control, Control System Overview- Error Response-Linear Error Dynamics-First-Order Error Dynamics- Second-Order Error Dynamics-Motion Control with Velocity Inputs - Motion Control with Torque or Force Inputs.	
UNIT – V:	**08 Hours**
Grasping and Manipulation: First-Order Analysis of a Single Contact Contact Types: Rolling, Sliding, and Breaking-Multiple Contacts, Force Closure-Duality of Force and Motion Freedoms, Omnidirectional Wheeled Mobile Robots- Controllability of Wheeled Mobile Robots	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course, the student will be able to:		
1	Recall foundational concepts of robot motion and configuration space.	L1, L2
2	Interpret rigid-body motions and transformation matrices.	L2
3	Apply forward kinematics to solve problems related to robot motion.	L3
4	Analyze the dynamics of open chains using Lagrangian and Newton-Euler formulations.	L4
5	Evaluate motion planning and control systems for robots, including error dynamics and motion control inputs.	L5

Mapping Levels of COs to POs / PSOs

Cos				**Program Outcomes (POs)**									**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
CO1	3	3	2	-	1	-	-	-	2	2	-	2	2	2
CO2	3	3	2	-	1	-	-	-	2	2	-	2	2	2
CO3	3	2	1	-	1	-	-	-	2	2	-	2	1	1
CO4	3	1	-	-	1	-	-	-	2	2	-	2	2	2
CO5	2	1	3	-	1	-	-	-	2	2	-	2	1	1

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Yan, Lili, and Gene M. Grossman. Robots and AI: A new economic era. Taylor & Francis, 2023.
2. Niku, Saeed B. Introduction to robotics: analysis, control, applications. John Wiley & Sons, 2020.

REFERENCE BOOKS:

1. Groover M. P, "Industrial Robotics", TataMcGraw-Hill, 1 st Edition, 2013
2. Richard D. Klafter, "Robotic Engineering", Prentice Hall, 1st Edition, 2013.
3. Fu K S, "Robotics", McGraw-Hill, 1st Edition, 2013.
4. Spong, M. W., Hutchinson, S., & Vidyasagar, M. (2020). Robot Modeling andControl, 2nd Edition. Wiley, ISBN: 978-1-119-52404-5.
5. Peter cork, Robotics, 2017, Vision and Control (2nd ed.), springer tracts inadvanced Robotics.
6. Simon J.D. Prince,Computer Vision: Models, Learning, and Inference, Cambridge University press
7. Ghosal, A. (2015). Robotics: Fundamental concepts and analysis. Oxford: OxfordUniversity Press. ISBN: 978-0-195-67391-3.

E-Resources:

1. https://www.doc.ic.ac.uk/~ajd/Robotics/RoboticsResources/lecture1.pdf
2. http://opencourses.emu.edu.tr/course/view.php?id=32
3. https://www.researchgate.net/publication/277712686_Introduction_to_Robotics_class_notes_UG_level

Activity Based Learning (Suggested Activities in Class)

1. Group discussion.
2. Projects on Computer graphics & User interface design.
3. Quiz
4. Assignment

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

****

MULTI MODAL AND MULTILINGUAL LANGUAGE PERCEPTION

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER - VI

Course Code	: 25OE3619	**Credits**	: 03
Hours / Week	: 03	**Total Hours**	: 45
L–T–P–J	: 2–0–0–2	**CIE+SEE**	: 60+40 Marks

Course Vision:

This course develops an understanding of multimodal and multilingual language perception by integrating concepts from natural language processing, computer vision, and speech technologies, enabling students to design and evaluate intelligent systems that process diverse linguistic and sensory inputs.

Course Objectives:

Students will:

1. Understand strengths, limitations, and risks of AI-generated content
2. Develop advanced prompting and content structuring strategies
3. Design AI-assisted content workflows and systems
4. Analyses content performance using platform insights
5. Evaluate ethical and authenticity concerns.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. **Lecture method** along with traditional lecture method, different type of teaching methods may be adopted to develop the course outcomes.
2. **Interactive Teaching:** incorporating brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Showing **Video/animation** films to explain functioning of various concepts.
4. Encourage **Collaborative** (Group Learning) Learning in the class.
5. To make **Critical thinking**, asking Higher order Thinking questions in the class in the form of Quiz and writing programs with complex solutions.
6. Showing the **different ways to solve** the same problem and encourage the students to come up with their own creative ways to solve them

UNIT – I	**08 Hours**

Foundations of Multimodal and Multilingual Perception

Topics: Introduction to language perception (human vs machine), multimodal communication, modalities (text, speech, vision, gesture), multilingualism and language diversity, challenges in

multilingual AI, overview of multimodal AI systems, data sources and preprocessing basics.

Applications: Speech assistants, multilingual translation systems, human-computer interaction, accessibility technologies.

UNIT – II	**09 Hours**

Multilingual Language Processing

Topics: Multilingual NLP fundamentals, language typology and scripts, tokenization challenges, cross-lingual embeddings, multilingual language models, transfer learning, low-resource languages, language identification, code-mixing and code-switching.

Applications: Machine translation systems, multilingual chatbots, cross-lingual information retrieval, global content processing systems.

UNIT – III	**06 Hours**

Multimodal Representation Learning

Topics: Multimodal data representation, feature extraction (text, speech, images), joint embedding spaces, alignment techniques, fusion methods (early, late, hybrid), attention mechanisms, transformer architectures for multimodal learning.

Applications: Image captioning, visual question answering, emotion recognition, multimodal sentiment analysis.

UNIT – IV	**11 Hours**

Multimodal and Multilingual Models & Applications

Topics: Vision-language models, speech-text systems, multimodal machine translation, pretrained multimodal models, zero-shot and few-shot learning, evaluation metrics, bias and fairness, privacy concerns, emerging trends.

Applications: AI assistants (multilingual), accessibility tools (speech-to-text, captioning), intelligent tutoring systems, cross-modal search engines.

UNIT – V	**11 Hours**

Applications, Evaluation, and Emerging Trends

Topics: Multimodal sentiment analysis, emotion recognition, visual question answering, multilingual chat-bots, evaluation metrics, ethical issues, privacy concerns, future trends.
Applications: Assistive technologies, AI-driven communication systems, real-time translation tools.

AI Integration:

- AI tools for multimodal data understanding, language detection systems.
- Multilingual NLP models, AI-based translation tools.

- Transformer-based multimodal models, embedding visualization tools.
- Pretrained multimodal AI systems, evaluation and benchmarking tools.
- Evaluation of AI systems using standard datasets and performance metrics for multimodal and multilingual applications.

Authentic Intelligence:

Limitations of machine perception vs human perception.
Bias in multilingual datasets and fairness issues.
Challenges in multimodal fusion and misinterpretation risks.
Ethical concerns in AI systems, privacy, inclusivity, and reliability.
Students evaluate ethical implications, privacy concerns, and societal impact of AI-driven systems.

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Understand fundamentals of multimodal and multilingual language perception.	L1
2	Analyze multilingual language processing techniques	L2
3	Apply multimodal representation learning methods	L2
4	Evaluate multimodal and multilingual AI models	L4
5	Design applications using multimodal and multilingual perception systems.	L3

Mapping Levels of COs to POs / PSOs

Table: Mapping Levels of COs to POs / PSOs

COS	Program Outcomes (POs)											PSOs		
	1	2	3	4	5	6	7	8	9	10	11	1	2	3
CO-1	3	3	1	–	2	–	–	–	–	–	–	1	2	1
CO-2	3	3	2	–	–	–	–	–	–	–	1	1	3	2
CO-3	3	3	3	2	2	–	–	–	–	2	1	3	3	3

CO-4	3	2	2	3	3	–	–	–	–	–	1	2	2	2
CO-5	3	3	3	3	3	–	–	–	–	–	2	3	3	3

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

TEXT BOOKS:

1. Jurafsky, D., & Martin, J. H., Speech and Language Processing, 3rd Edition, Pearson.
2. Baltrusaitis, T. et al., Multimodal Machine Learning: A Survey and Taxonomy, IEEE, 2019.
3. Eisenstein, J., Introduction to Natural Language Processing, MIT Press, 2019.

REFERENCE BOOKS:

1. Goldberg, Y., Neural Network Methods for Natural Language Processing, Morgan & Claypool, 2017.
2. Poria, S., Cambria, E., & Bajpai, R., Multimodal AI: From Theory to Practice, CRC Press, 2022.
3. Lewis, M. et al., Multilingual Language Models and Cross-lingual NLP, Cambridge University Press, 2020.

E-RESOURCES:

1. Research papers from IEEE, ACL, and Springer
2. NPTEL courses on NLP and AI
3. Hugging Face documentation (transformers & multimodal models).

RECOMMENDED TOOLS

- Programming: Python (NLP and ML libraries)
- Libraries: TensorFlow / PyTorch, Hugging Face Transformers
- Data Processing: NLTK, SpaCy
- Visualization: Matplotlib, Seaborn
- AI Tools: ChatGPT, multilingual AI platform

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

AI FOR AUTONOMOUS VEHICLES & SMART MOBILITY

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	:	25OE3620	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 Hours
L–T–P–S		2–0–0–2	**CIE + SEE**	:	60 + 40 Marks

Course Learning Objectives:

This course will enable students to:

1. Understand AI-driven architecture of autonomous vehicles and smart mobility systems.
2. Apply machine learning and deep learning techniques for perception, prediction, and decision-making.
3. Analyze localization, planning, and control using AI-based approaches.
4. Develop AI-enabled solutions for real-world smart mobility applications.
5. Evaluate safety, ethics, and societal impacts of AI in transportation systems.

Course Description:

AI for Autonomous Vehicles & Smart Mobility introduces students to the integration of artificial intelligence with modern transportation systems. The course covers perception using deep learning, decision-making using machine learning models, and intelligent path planning for autonomous navigation. Explore how AI enables vehicles to sense environments, predict traffic patterns, and make real-time decisions. Topics include computer vision, neural networks, reinforcement learning, and AI-based optimization for mobility systems.

The course emphasizes real-world applications such as self-driving cars, smart traffic systems, and connected mobility platforms, along with ethical and safety considerations in AI-driven transportation.

Course Outcome:

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Explain AI-based architecture of autonomous vehicles	L2
2	Apply ML/DL models for perception and prediction tasks	L3
3	Analyze AI-driven localization and path planning techniques	L4
4	Develop intelligent mobility solutions using AI	L3
5	Analyze ethical, safety, and societal implications of AI systems	L4

Course Modules :

UNIT – I : AI in Smart Mobility & Autonomous Systems	**09 Hours**

Evolution of mobility, levels of automation, AV architecture, Role of AI & ML in daily transportation, introduction to AI in transportation, intelligent transportation systems, AI pipeline (data → model → decision).
Applications: Self-driving cars, smart cities, ride-sharing platforms.

UNIT – II : AI-Based Perception Systems	**09 Hours**

Sensors (Camera, LiDAR, RADAR), sensor fusion, Computer Vision basics for autonomous vehicles, image processing, Convolutional Neural Networks (CNNs), object detection (YOLO, SSD), semantic segmentation. ***Applications:*** Lane detection, obstacle detection, pedestrian recognition.

UNIT – III : AI for Localization & Prediction	**09 Hours**

Localization (SLAM) and Mapping, supervised learning: regression & classification in mobility systems, neural networks, time-series prediction, traffic forecasting, ***Applications:*** GPS

enhancement, traffic prediction, route optimization.
UNIT – IV : AI-Based Planning, Control & Connectivity **09 Hours**
Path planning (Dijkstra, A*), reinforcement learning, decision-making models, trajectory planning, V2X communication, Edge AI in smart mobility, 5G mobility**,** Real-life examples: Smart traffic signals, emergency vehicle prioritization. ***Applications***: Autonomous navigation, smart traffic signals, connected vehicles.
UNIT – V : AI Safety, Ethics & Future Mobility **09 Hours**
Safety challenges in autonomous vehicles, Explainable AI (XAI), safety in AI systems, bias in ML models, ethical decision-making, legal frameworks, sustainability. Future trends: Electric vehicles, shared mobility, AI-based smart city ecosystems. ***Applications:*** Autonomous cars, delivery robots, smart city ecosystems.

Table: Mapping Levels of COs to POs / PSOs

	Program Outcomes (POs)											PSOs	
COs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**
CO1	3	2	1	-	2	-	1	-	-	-	2	1	1
CO2	3	3	2	1	3	-	-	-	-	-	2	2	2
CO3	3	3	3	2	3	-	-	-	-	-	2	2	2
CO4	3	3	3	2	3	1	1	2	1	-	2	2	3
CO5	3	3	2	2	2	2	3	1	1	1	3	2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI INTEGRATION & AUTHENTIC INTELLIGENCE REFLECTION:

- AI Integration:
  - Module-1: AI-driven mobility system analysis
  - Module-2: Deep learning for perception (CNN models)
  - Module-3: ML models for prediction and forecasting
  - Module-4: Reinforcement learning for navigation
  - Module-5: Explainable AI and ethical AI systems.

- **Authentic Intelligence Reflection:**
  - Module-1: Evaluate reliability of AI perception models
  - Module-2: Identify bias in datasets
  - Module-3: Analyze safety-critical decisions
  - Module-4: Reflect on ethical dilemmas
  - Module-5: Assess sustainability in AI mobility.

RESEARCH INTEGRATION TEACHING

1. Paper review on autonomous driving datasets (KITTI, Waymo, TiAND).
2. Case studies on smart mobility systems.
3. Mini presentations on emerging AI mobility technologies.
4. AI-assisted research using tools for analysis.
5. Critical evaluation of safety and ethical issues.

ASSESSMENT SCHEME:

1. **"Internal Assessment (IA)"** refers to the Continuous Internal Assessment (CIA) with a weightage of 60%. It includes:
   a. **Mid Semester Examination (MSE):** One descriptive examination to evaluate conceptual understanding of AI fundamentals and problem-solving approaches.
   b. **Continuous Assessment Tools (CAT – Six Components):** Includes activities such as quizzes, surprise tests, case studies, group discussions, project-based learning, seminars, industrial visits, reflection notes, model building, e-course certifications, or open book assignments.
   c. **AI-Powered Assignment (APA):** Students utilize AI tools for solving problems while demonstrating Authentic Intelligence through critical evaluation, validation of AI outputs, ethical reasoning, and responsible decision-making.
2. **"External Assessment (EA)"** refers to the Semester End Examination (SEE) with a weightage of 40%. This examination evaluates students' overall understanding of AI concepts, application skills, and their ability to analyze and solve problems effectively.

A. Professional Core Courses (Theory)

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT) every 2 weeks (Minimum 06 assessments)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	10
Total	**60 marks**

AI-POWERED ASSIGNMENTS:

AI^2 Pedagogy: It is a structured teaching-learning approach where students use AI tools for problem-solving while simultaneously developing Authentic Intelligence through critical evaluation, ethical reasoning, and responsible decision-making.

Assignment 1: AI Perception Model Evaluation

Students prompt an AI tool:

"Generate code for lane detection using CNN."

Students must:

1. Analyze accuracy
2. dataset limitations
3. improve performance

Assignment 2: Traffic Prediction Model Verification

- Students ask AI:

  " Compare ML models (Linear Regression vs Neural Networks)."
- Students must verify:
  1. Evaluate accuracy and edge cases

Assignment 3: AI Debugging Challenge

- Debug faulty object detection or prediction model
- Students must:
  1. Identify the bug
  2. Fix the code
  3. Explain the error

Assignment 4: Mobility Data Visualization

- Students use AI visualization tools to:
  1. Analyze real-world traffic datasets

Assignment 5: AI-Based Path Planning Study

1. Compare classical algorithms vs AI-based approaches

Suggested Reading:

Primary Textbooks

1. Mahajan, Rupali A., Rajesh Dey, Parikshit N. Mahalle, and Vivek S. Deshpande, AI-Driven Transportation Systems: Real-Time Applications & Related Technologies. Springer, 2025.
2. Houbing Song, Sabina Jeschke, et al., Smart Transportation: AI Enabled Mobility and Autonomous Driving. CRC Press, 2021.
3. Dayyala, N. M. Kottayi, and R. B. Mallick, Machine Learning in Transportation: Applications with Examples and Codes. De Gruyter, 2024.

Additional Resources

1. IEEE research articles
2. Hussein T. Mouftah, Melike Erol-Kantarci, Sameh Sorour, onnected and Autonomous Vehicles in Smart Cities. CRC Press, 2020.
3. Dominique Paret and Hassina Rebaine, Autonomous and Connected Vehicles: Network Architectures from Legacy Networks to Automotive Ethernet. Wiley, 2022.
4. Y. Wang, Z. Cui, and R. Ke, Machine Learning for Transportation Research and Applications. Elsevier, 2023.

E-Resources : (Recommended Tools)

- Python/OpenCV
- Tensorflow/ PyTorch
- Scikit-learn
- CARLA Simulator, SUMO Traffic Simulator
- Jupyter Notebook

Activity Based Learning (Suggested Activities in Class)

1. Group discussion: Discuss different AI techniques used in autonomous vehicles such as perception, localization, and path planning to understand real-world mobility challenges.
2. Collaborative Activity: Develop a mini project in teams of 4 students, such as a smart traffic system, lane detection model, or traffic prediction system using AI techniques.

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

****

AI FOR CONTENT CREATION

SEMESTER – VI

Subject Code	:	25OE3621	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 Hours
L–T–P–S	:	2–0–0–2			

Course Learning Objectives:

Students will:

1. Understand limitations and challenges of AI-generated content
2. Develop advanced prompting and content strategy skills
3. Design scalable AI-assisted content systems
4. Analyze content performance using platform-level insights
5. Evaluate ethical, societal, and authenticity concerns in AI content

Course Description:

AI for Content Creation is an advanced, application-driven course that focuses on building intelligent, differentiated content using AI systems rather than basic tool usage. The course explores why most AI-generated content fails to stand out, and how human creativity, strategic prompting, and system-level thinking can be used to create impactful and high-performing content.

Students will learn to design content pipelines that transform a single idea into multiple formats, analyze how digital platforms prioritize and rank content, and understand how to optimize engagement through structure, hooks, and storytelling.

The course also examines the risks of AI-generated content, including hallucination, misinformation, bias, and copyright issues. Through practical exercises and case-based learning, students will develop the ability to critically assess AI outputs and build responsible, high-quality content systems.

By the end of the course, students will be able to integrate AI tools with human intelligence to create scalable, ethical, and high-impact digital content.

Course Outcome:

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Explain strengths and limitations of AI-generated content	L2
2	Apply advanced prompting and content structuring techniques	L3
3	Design AI Assisted content workflows and systems	L3
4	Analyze content performance and platform behaviour	L4
5	Critically evaluate ethical and authenticity issues in AI-generated content	L4

Course Modules :

UNIT I: Understanding AI Content and Limitations	**10 Hours**
AI content generation: capabilities and limitations, hallucination and misinformation, AI sameness problem, human vs AI creativity, content differentiation strategies.	
UNIT – II : Advanced Prompting and Content Structuring	**08 Hours**
Prompt engineering vs prompt strategy, context stacking, iterative prompting, role-based prompting, structuring content for clarity, engagement, and impact.	
UNIT – III : AI-Assisted Content Systems	**09 Hours**

Content pipelines (idea → script → multi-format), content repurposing, personal brand voice creation, human-AI collaboration workflows.
UNIT – IV : Platform Algorithms and Content Optimization **09 Hours**
Content ranking systems (YouTube, social media), engagement metrics (CTR, retention, watch time), hook design, storytelling patterns, content psychology.
UNIT – V : Ethics, Authenticity and Future of AI Content **09 Hours**
AI misinformation, deepfakes, copyright and ownership, bias and fairness, AI governance, future of creators (augmentation vs replacement).

Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigation	Modern Tool Usage	The Engineer and Environment and		Ethics	Individual & Team Work	Communica-tion	Project management	Cognitive Outcome	Skill & Design
CO1	3	2	1	1	2	1		2	1	1		2	1
CO2	3	3	2	1	3	1		1	2	2	1	3	2
CO3	2	3	3	2	3	2	1	1	2	2	2	2	3
CO4	3	2	3	2	3	1	1	2	2	3	2	3	3
CO5	3	2	2	2	3	1	1	3	1	2	2	2	2

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

AI INTEGRATION & AUTHENTIC INTELLIGENCE REFLECTION:

AI Integration

- AI-generated content analysis
- Advanced prompt experimentation
- AI-assisted content pipeline design
- AI-based performance analysis
- AI tools for content verification

Authentic Intelligence Reflection

- Evaluation of incorrect AI outputs
- Comparison of prompting strategies
- Human vs AI contribution analysis

- Algorithmic impact analysis
- Ethical evaluation of AI-generated media

RESEARCH INTEGRATION TEACHING

Students review at least one topic:

- Generative AI and content authenticity
- AI bias in media
- Deepfake detection
- Algorithmic influence on content

Students present a short research review.

ASSESSMENT SCHEME :

1. **"Internal Assessment (IA)"** refers to the Continuous Internal Assessment (CIA) with a weightage of 60%. It includes:
    a. **Mid Semester Examination (MSE):** One descriptive examination to evaluate conceptual understanding of AI fundamentals and problem-solving approaches.
    b. **Continuous Assessment Tools (CAT – Six Components):** Includes activities such as quizzes, surprise tests, case studies, group discussions, project-based learning, seminars, industrial visits, reflection notes, model building, e-course certifications, or open book assignments.
    c. **AI-Powered Assignment (APA):** Students utilize AI tools for solving problems while demonstrating Authentic Intelligence through critical evaluation, validation of AI outputs, ethical reasoning, and responsible decision-making.
2. **"External Assessment (EA)"** refers to the Semester End Examination (SEE) with a weightage of 40%. This examination evaluates students' overall understanding of AI concepts, application skills, and their ability to analyze and solve problems effectively.

A. Professional Core Courses (Theory)

Criteria	Weightage of Marks
Hackathon, Continuous Assessment Tool (CAT)	30
Mid-Semester Examination (MSE)	20
AI-powered Assignment "Or" Mini Project*	10
Total	**60 marks**

AI-POWERED ASSIGNMENTS:

AI^2 Pedagogy: It is a structured teaching-learning approach where students use AI tools for problem-solving while simultaneously developing Authentic Intelligence through critical evaluation, ethical reasoning, and responsible decision-making.

- AI Content Evaluation
- Prompt Optimization

- AI Debugging Challenge
- Performance Analysis
- Ethical Case Study

Text Books

1. Taulli, Tom – *Artificial Intelligence Basics: A Non-Technical Introduction*
2. Handley, Ann – *Everybody Writes: Your Go-To Guide to Creating Ridiculously Good Content*
3. Miller, Donald – *Building a StoryBrand: Clarify Your Message So Customers Will Listen*

Reference Books:

1. Berger, Jonah – *Contagious: Why Things Catch On*
2. Schaefer, Mark W. – *The Content Code: Six Essential Strategies for Igniting Your Content, Your Marketing, and Your Business*
3. Vaynerchuk, Gary – *Crush It!: Why NOW Is the Time to Cash In on Your Passion*
4. OpenAI Documentation – Generative AI Concepts and Applications
5. Google AI Documentation – Responsible AI and Content Systems
6. IEEE Research Articles – Generative AI, Bias, and Content Authenticity

E-Resources: (Recommended Tools)

1. MDN Web Docs (for content structure and digital publishing basics)
2. OpenAI & Anthropic Documentation
3. Canva AI Learning Resources
4. YouTube Creator Academy
5. HubSpot Content Marketing Resources

Activity Based Learning (Suggested Activities in Class)

- Mini projects
- Content creation exercises
- Case studies
- Group discussions

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

CONVERSATIONAL AI AND SPOKEN LANGUAGE UNDERSTANDING

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	: 25AM3611	**Credits**:	03
Hours / Week	: 04 Hours	**Total Hours**:	45 Hours
L–T–P–S	: 2–0-0–2		

Prerequisites

Fundamentals of Artificial Intelligence, Machine Learning concepts, Probability and Statistics, Natural Language Processing. Familiarity with the Python programming language is desirable. Prior exposure to optimization or deep learning and Neural Network concepts will be an added advantage.

Course Learning Objectives:

This course will enable students to:

1. Understand classical and modern Conversational AI architectures, including LLM-based systems.
2. Apply NLP and deep learning techniques for language understanding and response generation.
3. Design intelligent dialogue systems using retrieval, generation, and reinforcement learning.
4. Develop, evaluate, and deploy scalable conversational AI applications.
5. Analyse ethical, societal, and research challenges in advanced conversational systems.

UNIT – I : Foundations of Conversational AI	**8 Hours**
Evolution of conversational systems, Types of conversational systems: task-oriented, open-domain, hybrid, Core components of the system, Understanding Conversational AI vs Chatbots vs Virtual Assistants, Applications of conversational systems, Challenges in conversational systems.	
UNIT – II : Natural Language Understanding & Representation Learning	**8 Hours**
Text preprocessing and linguistic features, Intent classification and entity recognition, Machine learning approaches for NLU, Word embeddings: Word2Vec, GloVe, FastText, Contextual embeddings: BERT, RoBERTa, Transformer architecture for NLU, Context handling, Ambiguity resolution	
UNIT – III : Dialogue Management, Generation & LLMs Dialogue systems: task-oriented vs open-domain, Dialogue state tracking, Rule-based dialogue management, Machine learning-based dialogue management, Natural Language Generation (NLG): template-	**8 Hours**

based and model-based approaches, Introduction to Large Language Models, Prompt engineering and in-context learning, Retrieval-Augmented Generation (RAG) for grounded conversations, Handling errors, fallbacks and mitigation strategies.	
UNIT – IV : Advanced Conversational AI Systems	**8 Hours**
Knowledge-grounded dialogue systems, Conversational Information Retrieval, Emotion-aware and empathetic conversational agents, Multimodal Conversational AI (text, speech, vision), Reinforcement Learning and RLHF for dialogue optimization, Explainable Conversational AI (XAI)	
UNIT – V : Deployment, Evaluation, Ethics & Future Trends	**7 Hours**
Evaluation metrics, human evaluation of conversational agents, deployment architectures and APIs, MLOps for conversational AI, security issues, future research directions and open challenges	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course, the student will be able to:		
1	Explain classical and modern conversational AI architectures	L2
2	Apply NLP and transformer-based models for language understanding	L3
3	Design dialogue systems using retrieval, generation, and RL	L4
4	Implement and evaluate conversational AI applications	L3
5	Analyze ethical, societal, and research challenges in conversational AI	L5

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	3	2	1	2	1							2	2	1
CO2	2	3	2	3	2							2	3	2
CO3	2	3	3	3	2				1	1	_	2	3	3
CO4	1	2	3	3	3				2	2	1	3	3	3
CO5	1	2	2	3	2	1	1	3	2	2	2	3	2	3

3: Substantial (High)2: Moderate (Medium) 1: Poor (Low)

TEXT BOOKS:

- Romil Rawat, Rajesh Kumar Chakrawarti, Sanjaya Kumar Sarangi, Piyush Vyas, Mary Sowjanya Alamanda, Kotagiri Srividya, Krishnan Sakthidasan Sankaran "Conversational Artificial Intelligence", Scrivener Publishing, January 2024.
- Michael McTear , Marina Ashurkina , "Transforming Conversational AI", February 2024.

REFERENCE BOOKS:

- "Effective Conversational AI" by Andrew Freed, Cari Jacobs, Eniko Rozsa, Foreword by Jesús Mantas, April 2025.
- "Designing Bots: Creating Conversational Experiences" by Amir Shewat, June 2017.
- The Definitive Guide to Conversational AI with Dialogflow and Google Cloud by Lee Boonstra, 2021.

E-Resources:

- Conversational AI: LLM Driven Development for Beginners – course by udemy
- Getting started with Conversational AI - https://www.youtube.com/watch?v=0sbY7jgI
- Conversational AI (2024): A Comprehensive Guide - https://www.gupshup.ai/resources/guide/conversational-ai-comprehensive-guide/

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

MINOR PROJECT

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	:	25AM3604	**Credits**	:	02
Hours / Week	:	02	**Total Hours**	:	26 Hours
L–T–P–S	:	0–0–0–4			

Course Learning Objectives:

This course will enable students to:

1. **To identify** key research questions within a field to carry out research in a team.
2. **To identify** and summarize the literature review of the relevant field.
3. **To demonstrate** relevant referencing and inculcate new skills in various aspects of academic writing.
4. **To demonstrate** the knowledge and understanding of writing the publication/report.
5. **To showcase** the strong evidence on the clarity of the argument, understanding of the selected domain area and presentation of its technical information.
6. **To detail description** of the process of carrying out the independent research in written document along with results and conclusions with reference to the existing literature.
7. **To analyze** and synthesize the new research findings.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. Lecture method means it includes not only the traditional lecture method but a different type of teaching method that may be adopted to develop the course outcomes.
2. Interactive Teaching: Adopt Active learning that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show Video/animation films to explain the functioning of various concepts.
4. Encourage Collaborative (Group Learning) Learning in the class.
5. To make Critical thinking, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the student's understanding.

COURSE CONTENT:

The research topic proposed by both the guide and the student team should be approved by the department chairman to proceed further. A degree of industrial input and involvement will be encouraged, and can be facilitated through existing academic-industrial collaborations or by addressing specific topics that are of interest to industrial partners.

All projects will be closely supervised by the Project Guide with ongoing feedback and guidance at all stages of the project from conception to completion.

The following criteria will be checked by the department chairman to approve for the research proposal:

a. Department staff as course guide
   1. Ability to provide research direction to the student in the chosen field of interest
   2. Ability to design an appropriate research strategy and methodology to carry out the research by student
   3. Ability to provide and evaluate the strong literature review document for the chosen research topic
   4. Ability to train students on research paper / technical writing skills
   5. Conduct reviews in regular time period and submit the evaluation to department chairman

b. Student Team
   1. To be dedicated and committed to work on a new research topic by learning new technical skills
   2. To have fair knowledge on what is product development or research topic
   3. To have constant interaction with allocated guide by providing weekly updates
   4. To be committed to complete the project and submitting the technical paper within the stipulated time framed by the university

Evaluation:

There will be CIA evaluation as well as the Semester end evaluation of the work done. Itwill be done by a committee of senior researchers of the Department.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Identify and Select an Appropriate Research Problem	L1
2	Explain and Summarize Relevant Literature	L2
3	Compare and Critically Analyze Relevant Research Papers	L3
4	Construct a Research Model and Perform Evaluation	L4
5	Create and Draft Publications or Demonstrations	L5

Mapping Levels of COs to POs / PSOs														
Cos				**Program Outcomes (POs)**									**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
CO1	3	2	2	1	1	1	1	1	2	2	3	2	2	1
CO2	2	2	3	2	2	1	1	1	2	2	3	2	1	2
CO3	3	3	3	2	3	2	2	1	2	2	2	2	2	2
CO4	2	3	2	3	2	1	2	2	2	2	2	3	2	2
CO5	1	1	2	1	2	1	1	1	3	3	2	2	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

****

UNIT – I	07 Hours
Foundations of Cognitive Science and Cognitive Systems: Foundations of Cognitive Science, Computational Cognitive Science, Human-Centred Computing, Cognitive Neuroscience. Interdisciplinary nature: psychology, neuroscience, computer science, linguistics, philosophy. Core mental functions: perception, attention, memory, language, reasoning, and decision making. Information-processing view of the mind, symbolic vs connectionist vs embodied/situated cognition.	
UNIT – II	**07 Hours**
Cognitive Architectures and Knowledge Representation Classical cognitive architectures: ACT-R, SOAR – goals, production rules, declarative vs procedural memory (conceptual coverage, not heavy implementation). Knowledge representation for cognitive systems: symbols, frames, semantic networks, schemas, scripts. Mental models, working memory and long-term memory, chunking and cognitive load; implications for system design. Reasoning and decision-making: rule-based reasoning, heuristics and biases, bounded rationality.	
UNIT – III	**08 Hours**
Computational Cognitive Science and Learning in Cognitive Systems: Computational Cognitive Science: goals, modelling behavioural and neural data. Probabilistic/Bayesian models of cognition: basics of Bayesian inference, perception as Bayesian estimation, decision making under uncertainty. Basics of reinforcement learning (RL) as a model of learning and decision-making; links to neural basis of learning and memory. Supervised and unsupervised learning in cognitive systems; relation to human learning, generalization, and concept formation. Model evaluation: goodness-of-fit, predictive accuracy, comparison of models (symbolic vs probabilistic vs neural).	
UNIT – IV	**08 Hours**
Human-Centered Computing, HCI and Interaction in Cognitive Systems: Human Centered Computing (HCC) principles: user needs, mental models, affordances, feedback, and error prevention. Human–Computer Interaction basics: interaction paradigms, input/output modalities, cognitive aspects of interface design. Attention, perception, and memory constraints in UI design (e.g., visual search, change blindness, working memory limits). Experiment design and analysis in HCI/cognitive systems: within/between-subject designs, basic statistics & data analysis and inference. Evaluation of interactive cognitive systems (conversational agents, recommender systems, assistive interfaces).	
UNIT – V	**09 Hours**
Cognitive Neuroscience, Ethics, and Applications of Cognitive Systems: Cognitive Neuroscience: how the brain enables the mind. Neural basis of key functions: perception, attention, learning and memory, language, emotion, and motivation. Applications of cognitive systems: decision-support systems, educational technologies, cognitive tutors, healthcare and mental health, social robotics. Ethics and societal implications: privacy, autonomy, bias, transparency, explainability in cognitive systems; responsible AI guidelines.	

VII SEM – ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

SL	Course Type	Course Code	Course Name	Teaching Hours / Week				Examination			
				Lecture	Tutorial	Practical	Project	CIE Marks	SEE Marks	Total Marks	Credit
				L	T	P	J				
1	PROJ	25AM4701	Capstone Project Phase-I	0	0	0	8	100	--	100	04
2	PEC	25AM47XX	Professional Elective Course – IV	3	0	0	0	100	--	100	03
3	PEC	25AM47XX	Professional Elective Course – V/ MOOC	3	0	0	0	100	--	100	03
4	OEC	25OEXXXX	Open Elective – II	3	0	0	0	100	--	100	03
			Total	9	0	0	8				**13**

IPCC-Integrated Professional Core Course, PCC-Professional Core Courses, SEC-Skill Enhancement Courses, PEC-Professional Elective Courses, OEC-Open Elective Courses, PROJ-Project Work, L – Lecture, T – Tutorial, P – Practical, J–Project, C – No. Of Credits

SL	Course Type	Course Name	Teaching Hours / Week				CIE Marks	SEE Marks	Total Marks	Credits
			Lecture	Tutorial	Practica	Project				
			L	T	P	J				
1	PEC	Deep Learning for Science and Engineering	2	0	0	2	100	-	100	03
2	PEC	Autonomous Agentic AI Systems	2	0	0	2	100	-	100	03
	PEC	Intelligent Video Surveillance Systems	2	0	0	2	100	-	100	03
3	PEC	Foundation Models for Robotics	2	0	0	2	100	-	100	03
4	PEC	AI for Agriculture	2	0	0	2	100	-	100	03
5	PEC	Multimodal and Multilingual Language Perception	2	0	0	2	100	-	100	03

SL	Course Type	Course Name	Teaching Hours / Week				CIE Marks	SEE Marks	Total Marks	Credits
			Lecture	Tutorial	Practica	Project				
			L	T	P	J				
1	PEC	NPTEL/MOOC	2	0	0	2	100	-	100	03
2	PEC	Operationalizing Agentic AI Systems	2	0	0	2	100	--	100	03
3	PEC	Multi-modal interaction and speech interfaces	2	0	0	2	100	--	100	03
4	PEC	Human-Centric Automation and Industry 5.0	2	0	0	2	100	--	100	03
5	PEC	Blockchain for Data Security	2	0	0	2	100	--	100	03
6	PEC	Building Transformer-Based Natural Language Processing Applications	2	0	0	2	100	--	100	03

SL	Course Type	Course Name	Teaching Hours / Week				CIE Marks	SEE Marks	Total Marks	Credits
			Lecture	Tutorial	Practica	Project				
			L	T	P	J				
1	OEC	Human Centric Automation and Industry 5.0	2	0	0	2	100	--	100	03
2	OEC	Conversational AI and Spoken Language Understanding	2	0	0	2	100	--	100	03
3	OEC	AI for Psychology	2	0	0	2	100	--	100	03
4	OEC	GenAI and LLM (Non-CSE Students)	2	0	0	2	100	--	100	03

CAPSTONE PROJECT PHASE - I

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Subject Code	:	25AM4701	**Credits**	:	04
Hours / Week	:	04	**Total Hours**	:	42 Hours
L–T–P–J	:	0–0–0–8			

Course Learning Objectives:

This course will enable students to:

1. **To identify** key research questions within a field to carry out research in a team.
2. **To identify** and summarize the literature review of the relevant field.
3. **To demonstrate** relevant referencing and inculcate new skills in various aspects of academic writing.
4. **To demonstrate** the knowledge and understanding of writing the publication/report.
5. **To showcase** the strong evidence on the clarity of the argument, understanding of the selected domain area and presentation of its technical information.
6. **To provide a detailed description of the process of carrying out the independent research in a written document, along with results and conclusions,** with reference to the existing literature.
7. **To analyse and synthesise** the new research findings.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

7. Lecture method means it includes not only the traditional lecture method but a different type of teaching method that may be adopted to develop the course outcomes.
8. Interactive Teaching: Adopt Active learning that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
9. Show Video/animation films to explain the functioning of various concepts.
10. Encourage Collaborative (Group Learning) Learning in the class.
11. To make Critical thinking, ask at least three Higher-order Thinking questions in the class.
12. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the student's understanding.

COURSE CONTENT:

1. The research topic proposed by both the guide and the student team should be approved by the department chairman to proceed further.
2. A degree of industrial input and involvement will be encouraged, and can be facilitated through existing academic-industrial collaborations or by addressing specific topics that are of interest to industrial partners.
3. The problem statement should be big enough to be carried out in two phases over the two semesters i.e., VII and VIII semesters in the VI year.
4. All projects will be closely supervised by the Project Guide with ongoing feedback and guidance at all stages of the project from conception to completion.
5. The following criteria will be checked by the department chairman to approve for theresearch proposal:

a. Department staff as course guide
   1. Ability to provide research direction to the student in the chosen field of interest
   2. Ability to design an appropriate research strategy and methodology to carry out the research by student
   3. Ability to provide and evaluate the strong literature review document forthe chosen research topic
   4. Ability to train students on research paper / technical writing skills
   5. Conduct reviews in regular time period and submit the evaluation to department chairman

b. Student Team
   1. To be dedicated and committed to work on a new research topic by learning new technical skills
   2. To have fair knowledge on what is product development or research topic
   3. To have constant interaction with allocated guide by providing weekly updates
   4. To be committed to complete the project and submitting the technical paper within the stipulated time framed by the university

Evaluation:

1. Phase-1 comprises of Literature Survey, Problem identification, Objectives and Methodology.
2. There will be CIA evaluation (Project reviews) done by a committee of senior faculty of the Department based on the rubrics
3. Additionally, there will be a Semester end evaluation of the work done that would include an internal Faculty and an external academic expert

COURSE CONTENT:

1. The research topic proposed by both the guide and the student team should be approved by the department chairman to proceed further.
2. A degree of industrial input and involvement will be encouraged, and can be facilitated through existing academic-industrial collaborations or by addressing specific topics that are of interest to industrial partners.
3. The problem statement should be big enough to be carried out in two phases over the two semesters i.e., VII and VIII semesters in the VI year.
4. All projects will be closely supervised by the Project Guide with ongoing feedback and guidance at all stages of the project from conception to completion.
5. The following criteria will be checked by the department chairman to approve for theresearch proposal:

b. Department staff as course guide

1. Ability to provide research direction to the student in the chosen field of interest
2. Ability to design an appropriate research strategy and methodology to carry out the research by student
3. Ability to provide and evaluate the strong literature review document forthe chosen research topic
4. Ability to train students on research paper / technical writing skills
5. Conduct reviews in regular time period and submit the evaluation to department chairman

b. Student Team

1. To be dedicated and committed to work on a new research topic by learning new technical skills
2. To have fair knowledge on what is product development or research topic
3. To have constant interaction with allocated guide by providing weekly updates
4. To be committed to complete the project and submitting the technical paper within the stipulated time framed by the university

Evaluation:

1. Phase-1 comprises of Literature Survey, Problem identification, Objectives and Methodology.
2. There will be CIA evaluation (Project reviews) done by a committee of senior faculty of the Department based on the rubrics
3. Additionally, there will be a Semester end evaluation of the work done that would include an internal Faculty and an external academic expert

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Identify and Select an Appropriate Research Problem	L1
2	Explain and Summarize Relevant Literature	L2

3	Compare and Critically Analyze Relevant Research Papers	L3
4	Construct a Research Model and Perform Evaluation	L4
5	Create and Draft Publications or Demonstrations	L5

Mapping Levels of COs to POs / PSOs														
COs				**Program Outcomes (POs)**									**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
CO1	3	2	2	1	1	1	1	1	2	2	3	2	2	1
CO2	2	2	3	2	2	1	1	1	2	2	3	2	1	2
CO3	3	3	3	2	3	2	2	1	2	2	2	2	2	2
CO4	2	3	2	3	2	1	2	2	2	2	2	3	2	2
CO5	1	1	2	1	2	1	1	1	3	3	2	2	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

****

DEEP LEARNING FOR SCIENCE AND ENGINEERING

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Course Code	:	25AM4702	**Credits**	:	03
Hours / Week	:	03	**Total Hours**	:	45
L–T–P–J	:	2–0–0–2	**CIE+SEE**	:	60 + 40 Marks

Course Vision:

This course aims to develop a strong foundation in deep learning and its application to scientific and engineering problems. It focuses on integrating data-driven models with physical principles, enabling learners to solve complex real-world problems using scalable, efficient, and intelligent computational techniques.

Course Objectives:

This course will enable students to:

6. Understand fundamental concepts of deep learning and neural networks
7. Gain proficiency in Python and deep learning frameworks
8. Apply deep learning techniques to scientific and engineering problems
9. Explore advanced topics such as neural operators and uncertainty quantification
10. Implement scalable solutions using distributed and high-performance computing.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

7. **Lecture method** along with traditional lecture method, different type of teaching methods may be adopted to develop the course outcomes.
8. **Interactive Teaching:** incorporating brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
9. Showing **Video/animation** films to explain functioning of various concepts.
10. Encourage **Collaborative** (Group Learning) Learning in the class.
11. To make **Critical thinking**, asking Higher order Thinking questions in the class in the form of Quiz and writing programs with complex solutions.
12. Showing the **different ways to solve** the same problem and encourage the students to come up with their own creative ways to solve them.

UNIT – I	**09 Hours**

Fundamentals of Deep Learning

Topics: Introduction to Deep Learning, A Primer on Python, NumPy, SciPy and Jupyter Notebooks, Deep Learning Networks, A Primer on TensorFlow, PyTorch and JAX, Training and Optimization, Neural Network Architectures

Applications: Image Classification and Object Detection, Chatbots and Text Analysis, Speech Recognition Systems, Deep Learning Model Development, Data Analysis with Python Tools

UNIT – II	**09 Hours**

Neural Differential Equations

Topics: Discovering Differential Equations, Physics-Informed Neural Networks (PINNs)- Part I, Physics-Informed Neural Networks (PINNs)- Part II

Applications: Solving Differential Equations in Engineering Systems, Fluid Dynamics and Aerodynamics Simulation, Heat Transfer and Thermal Analysis, Structural Health Monitoring and Material Modelling, Climate Modelling and Environmental Prediction

UNIT – III	**09 Hours**

Neural Operators

Topics: Introduction to Deep Neural Operators, Mathematical Foundations, Fourier Transform and

Inverse Fourier Transform, Deep Operator Networks, Fourier Neural Operators, Learning Operators for Differential Equations, Implementation using Python and Deep Learning Frameworks **Applications:** Solving Partial Differential Equations in scientific computing, Fluid Flow and Turbulence Modelling, Weather and Climate Prediction, Material Science and Structural Simulation, Real-time Engineering Simulations and Digital Twins
UNIT – IV **09 Hours**
Data and Uncertainty **Topics:** Introduction to Multi-Fidelity Data in Machine Learning, Multi-Fidelity Modelling Techniques, Gaussian Process Regression, Multi-Fidelity Deep Neural Networks, Introduction to Uncertainty Quantification, Bayesian Neural Networks, Neural Methods for Uncertainty Estimation. **Applications:** Engineering Design Optimization with Limited Data, Risk Assessment and Decision Making under Uncertainty, Climate and Weather Forecasting with Confidence Intervals, Medical Diagnosis with Uncertainty Estimation, Surrogate Modelling for Complex Simulations
UNIT – V **09 Hours**
Scalable Scientific Machine Learning **Topics:** Introduction to Scalable Scientific Machine Learning, Multi-GPU and Distributed Training Concepts, Message Passing Interface for Parallel Computing, Horovod for Distributed Deep Learning, Domain Decomposition Methods, NVIDIA Physics NeMo Core and Framework Overview, Physics NeMo Symbolic for Solving Differential Equations. **Applications:** Large-Scale Scientific Simulations using Multi-GPU Systems, High-Performance Computing for Physics-Based Modelling, Real-Time Fluid Dynamics and Wave Simulations, Parallel Training of Deep Learning Models for Big Data, Digital Twin Development for Complex Engineering Systems

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain the fundamentals of deep learning and scientific machine learning	L2
2	Apply Python and deep learning frameworks to build models	L3
3	Analyze differential equations using PINNs and neural operators	L4
4	Evaluate model performance with uncertainty quantification methods	L5
5	Design scalable deep learning systems using distributed computing.	L5

Mapping Levels of COs to POs / PSOs

Table: Mapping Levels of COs to POs / PSOs													
COS	**Program Outcomes (POs)**											**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**
CO-1	3	2	2	–	2	2	–	2	–	–	2	2	2
CO-2	3	3	3	–	3	3	–	2	–	–	–	3	3
CO-3	3	2	3	–	3	3	–	3	–	–	–	3	3
CO-4	3	2	3	2	3	3	–	3	2	–	–	3	3
CO-5	3	2	3	–	3	3	–	3	2	2	2	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI Integration

- Application of deep learning models in scientific problem solving
- Integration of AI with physics-based simulations
- Automated model training and optimization
- AI-driven predictive analytics for engineering systems
- Deployment of intelligent, scalable AI systems

Authentic Intelligence

- Real-world case studies from science and engineering domains
- Hands-on implementation using industry tools
- Problem-solving using real datasets
- Ethical considerations in AI applications
- Project-based experiential learning.

TEXT BOOKS:

- *Deep Learning* – Ian Goodfellow, Yoshua Bengio, Aaron Courville
- *Pattern Recognition and Machine Learning* – Christopher M. Bishop
- *Scientific Machine Learning* – Course Notes / Technical Resources

REFERENCE BOOKS:

- *Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow* – Aurélien Géron
- *Machine Learning: A Probabilistic Perspective* – Kevin P. Murphy

E-RESOURCES:

https://developer.nvidia.com/teaching-kits/science-and-engineering-syllabus

7. https://www.tensorflow.org/
8. https://pytorch.org/tutorials/
9. https://ocw.mit.edu/courses/6-036-introduction-to-machine-learning-fall-2020/

10. NVIDIA Deep Learning Institute (DLI)
11. Deep Learning with PyTorch (Coursera)
12. TensorFlow 2 Deep Learning Specialization

RECOMMENDED TOOLS

- Python (NumPy, SciPy, Jupyter Notebook)
- TensorFlow / PyTorch / JAX

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

AUTONOMOUS AGENTIC AI SYSTEMS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER VII

Subject Code	: 25AM4703	**Credits**:	03
Hours / Week	: 03 Hours	**Total Hours**:	39 Hours
L–T–P–S	: 2–0–0–2		

Course Learning Objectives:

This course will enable students to:

1. **Understand** the fundamental principles of classical planning, search algorithms, and heuristic design for automated problem solving.
2. **Analyze** and formulate sequential decision-making problems using Markov Decision Processes and related frameworks.
3. **Apply** dynamic programming, Monte Carlo methods, and temporal-difference learning techniques for value-based reinforcement learning.
4. **Integrate** planning and learning approaches for model-based reinforcement learning and online decision-making.
5. **Develop** policy gradient methods and actor-critic algorithms for continuous control and high-dimensional decision spaces.
6. **Evaluate** planning and decision-making systems with respect to optimality, sample efficiency, scalability, and real-world deployment considerations.

Teaching-Learning Process (General Instructions):

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Real-World Case Studies:*** *Use real-world applications from robotics, game playing, resource management, and autonomous systems to demonstrate the practical relevance of planning and decision-making algorithms.*
2. ***Problem-Solving and Design Exercises:*** *Conduct problem-solving sessions where students formulate planning domains, design heuristics, and analyze algorithmic trade-offs for sequential decision problems.*
3. ***Hands-on Learning****: Organize laboratory sessions involving implementation and evaluation of planning algorithms, MDPs, and reinforcement learning agents in simulated environments.*
4. ***Reflective and Research-Based Learning:*** *Encourage students to study recent research papers on planning and RL, reflect on algorithmic advances, limitations, and design choices through presentations and discussions.*

UNIT – I	**8 Hours**

Classical Planning & Search:

Planning problem formulation: States, actions, goals, and plans. STRIPS and PDDL representation languages. State-space search: Forward and backward search. Uninformed search: BFS, DFS, iterative deepening. Informed search and heuristics: A* algorithm, greedy best-first search. Heuristic design: Relaxed planning graphs, delete-relaxation, pattern databases. Planning graphs

and GraphPlan algorithm. Hierarchical Task Network (HTN) planning.	
UNIT – II	**8 Hours**
Markov Decision Processes & Dynamic Programming: Sequential decision-making under uncertainty. Markov Decision Processes: States, actions, transition dynamics, rewards, policies. Bellman equations and optimality principles. Value iteration and policy iteration algorithms. Convergence properties and computational complexity. Partially Observable MDPs (POMDPs): Belief states and belief-space planning. Extensions: Discounting, finite vs infinite horizons, episodic vs continuing tasks.	
UNIT – III	**8 Hours**
Value-Based Reinforcement Learning: Model-free reinforcement learning framework. Monte Carlo methods: First-visit and every-visit MC, exploring starts. Temporal-Difference learning: TD(0), TD(λ), eligibility traces. Q-learning and SARSA algorithms. Function approximation in RL: Linear methods, feature engineering. Deep Q-Networks (DQN): Experience replay, target networks, double DQN. Rainbow and distributional RL. Multi-step bootstrapping and n-step methods.	
UNIT – IV	**7 Hours**
Policy Gradient & Actor-Critic Methods: Policy-based reinforcement learning. Policy gradient theorem and REINFORCE algorithm. Baseline functions and variance reduction. Actor-critic architectures: A2C, A3C. Advanced policy gradient methods: Proximal Policy Optimization (PPO), Trust Region Policy Optimization (TRPO). Deterministic policy gradients: DDPG, TD3. Soft Actor-Critic (SAC) and maximum entropy RL. Continuous action spaces and control problems.	
UNIT – V	**8 Hours**
Model-Based RL, Planning & Advanced Topics: Model-based reinforcement learning: Learning forward models, Dyna architecture. Monte Carlo Tree Search (MCTS) and AlphaGo/AlphaZero. Integrated planning and learning: Model Predictive Control with learned models. Multi-agent reinforcement learning: Nash equilibria, cooperative and competitive settings. Hierarchical reinforcement learning: Options framework, feudal networks. Exploration strategies: ε-greedy, UCB, Thompson sampling, curiosity-driven exploration. Sample efficiency, transfer learning, and sim-to-real transfer. Safety and robustness in planning and RL systems.	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Analyze planning problems and design effective heuristics for state-space search algorithms.	L3
2	Formulate and solve sequential decision problems using MDPs and	L4

	dynamic programming methods.	
3	Implement and compare value-based reinforcement learning algorithms including deep Q-learning.	L3
4	Apply policy gradient and actor-critic methods to continuous control and high-dimensional problems.	L4
5	Critically evaluate model-based planning, exploration strategies, and deployment considerations in real-world RL systems.	L3

Mapping Levels of COs to POs / PSOs														
COs				**Program Outcomes (POs)**									**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
CO1	3	3	2	2	2						2		3	
CO2	3	3	2	2	2						2		3	2
CO3	3	3	2	3	3						2		2	3
CO4	3	2	2	3	3						2		2	2
CO5	3	2	3	3	2	1	1	2	2		1		3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, 4th Edition, 2020.
2. Richard S. Sutton and Andrew G. Barto, Reinforcement Learning: An Introduction, 2nd Edition, 2018.
3. Malik Ghallab, Dana Nau, and Paolo Traverso, Automated Planning: Theory and Practice, 2004.

REFERENCE BOOKS:

Dimitri P. Bertsekas, Dynamic Programming and Optimal Control, 4th Edition, 2017.

Csaba Szepesvári, Algorithms for Reinforcement Learning, 2010, Morgan & Claypool.

Mykel J. Kochenderfer, Decision Making Under Uncertainty: Theory and Application, 2015, MIT Press.

E-Resources:

- NPTEL – Reinforcement Learning (IIT Madras) https://nptel.ac.in/courses/106106143
- OpenAI Spinning Up in Deep RL https://spinningup.openai.com/
- David Silver's RL Course (UCL/DeepMind)

https://www.youtube.com/playlist?list=PLqYmG7hTraZDMOYHWgPebj2MfCFzFObQ

- UC Berkeley CS188 – Introduction to AI
  https://inst.eecs.berkeley.edu/~cs188/
- Stanford CS221 – AI: Principles and Techniques
  https://stanford-cs221.github.io/
- DeepMind x UCL Deep RL Course
  https://www.deepmind.com/learning-resources/reinforcement-learning-lecture-series
- Gymnasium (OpenAI Gym) Documentation
  https://gymnasium.farama.org/
- Stable-Baselines3 RL Algorithms
  https://stable-baselines3.readthedocs.io/

Activity-Based Learning (Suggested Activities in Class):
Hands-on activity, Quiz, Seminar.

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

INTELLIGENT VIDEO SURVEILLANCE SYSTEMS

SEMESTER – VII

Subject Code	:	25AM4704	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 Hours
L–T–P–S	:	2–0–0–2			

Course Learning Objectives:

This course will enable students to:

1. Understand the fundamental components, processes, and architectures involved in video analytics and surveillance systems.
2. Apply various techniques for foreground extraction, background modeling, segmentation, and object tracking in video data.
3. Familiarize with different classifiers and learning methods including neural networks, deep learning models, fuzzy logic, Bayesian methods, and HMMs used for video-based analysis.
4. Explore the role of video analytics in security applications such as human action recognition, behavioral analysis, crowd monitoring, and perimeter security.
5. Examine the use of video analytics in business intelligence, traffic monitoring, and advanced driver assistance systems for decision-making and real-world deployment.

Course Description:

Video Analysis and Surveillance focus on developing intelligent systems that can interpret and analyze video data for real-world applications. The course introduces core concepts such as video analytics pipelines, pre-processing, feature extraction (PCA, SIFT, HOG), segmentation, and object tracking. It also covers classification techniques ranging from classical machine learning to deep learning models like CNNs, RNNs, and LSTMs for spatio-temporal analysis, along with performance evaluation methods.

The course emphasizes practical applications in security, surveillance, traffic monitoring, and business intelligence, including object detection, behavior analysis, crowd monitoring, and anomaly detection. Students gain hands-on experience in designing video analytics systems while exploring advanced topics such as edge computing and AI-powered surveillance, with a focus on ethical and responsible use of these technologies.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Explain the fundamental components and processes of video analytics including pre-processing, feature extraction, and classification pipelines.	L2
2	Apply foreground extraction, segmentation, and tracking techniques to analyze video data effectively.	L3
3	Compare and use classification approaches such as neural networks, deep learning models, fuzzy logic, Bayesian methods, and HMMs for video-based analysis.	L3
4	Evaluate video analytics approaches to recommend suitable solutions for security, business intelligence, and traffic monitoring applications.	L3
5	Design and develop intelligent video surveillance systems addressing real-world challenges while considering ethical and privacy implications.	L3

Course Outcome:

Course Modules:

UNIT – I: Video Analytic Components	**09 Hours**
Introduction to Video Analytics, Need for Video Analytics, Overview of video Analytics systems, Video analytics pipeline and architecture, Pre-processing Techniques, Feature space representation (PCA, FLD, SIFT, HOG), Classifier design for video data, Feature selection and dimensionality reduction. ***Applications:*** Surveillance and security systems, Traffic monitoring and accident detection, Facial recognition systems, Industrial quality inspection, Human activity recognition, Video search and retrieval.	
UNIT – II: Foreground Extraction & Segmentation	**09 Hours**
Background Modeling-Background estimation techniques, Averaging-based methods, Gaussian Mixture Model (GMM), Optical Flow-based approaches, Image Segmentation-Region-based segmentation, Region splitting and merging, Watershed segmentation, Morphological Operations-Erosion and Dilation, Advanced Tracking. ***Applications:*** Object detection and tracking in surveillance, Motion detection in video analytics, medical image segmentation, Autonomous vehicle navigation, Industrial defect detection, Crowd and activity monitoring.	
UNIT – III: Classifiers for Video Analysis	**09 Hours**
Classical Machine Learning Approaches, Bayesian classifier, Decision trees and ensemble methods, Fuzzy Classifier, Deep Learning Approaches, Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs) for temporal modeling, LSTM networks, Probabilistic Models, Hidden Markov Models (HMM) based classifier, Conditional Random Fields (CRFs), Performance Evaluation. ***Applications:*** Action and activity recognition, Video-based anomaly detection, Speech and gesture recognition, Emotion recognition from video, Autonomous driving perception systems, Smart surveillance and behavior analysis.	
UNIT – IV: Video Analytics for Security Applications	**09 Hours**
Object Detection and Recognition, Human behavioral analysis, Gait recognition, Human action recognition, Pose estimation, Crowd Analysis, Crowd density estimation, anomaly detection in crowds, Security Systems, Perimeter security and intrusion detection, Smart city surveillance applications. ***Applications:*** Smart surveillance systems, Intrusion and perimeter security, Crowd monitoring and management, Biometric identification (gait and pose), Suspicious activity detection, Smart city safety applications.	
UNIT – V: Business Intelligence, Traffic Monitoring & Advanced Topics	**09 Hours**
Business Intelligence Applications, Video Analysis for retail analytics, Customer behaviour analysis, Queue management systems, Traffic Monitoring, Traffic flow analysis, Speed estimation, Traffic violation detection, Advanced Driver Assistance Systems (ADAS), Video-based rendering and scene reconstruction, Activity recognition and summarization, 3D video surveillance, Edge computing for surveillance, Future trends: AI-powered surveillance, federated learning. ***Applications:*** Retail analytics and customer behavior analysis, Queue management systems, Traffic monitoring and violation detection, ADAS in smart vehicles, 3D surveillance and scene reconstruction, Edge-based real-time video analytics.	

Mapping Levels of COs to POs / PSOs													
COs				**Program Outcomes (POs)**								**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**
CO1	3	2	1	2	1							2	1
CO2	2	3	2	3	2							3	2
CO3	2	3	3	3	2				1	1	_	3	3
CO4	1	2	3	3	3				2	2	1	3	3
CO5	1	2	2	3	2	1	1	3	2	2	2	2	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI INTEGRATION & AUTHENTIC INTELLIGENCE REFLECTION:

- **AI Integration:**
  - Module-1: AI-based video understanding systems, automated feature extraction tools, AI-driven video classification models.
  - Module-2: AI-based background subtraction models, motion detection algorithms, intelligent object segmentation tools.
  - Module-3: Deep learning models (CNN, RNN, LSTM) for video classification, AI-based activity recognition systems, probabilistic sequence modeling tools.
  - Module-4: AI-powered surveillance systems, human behavior analysis models, crowd monitoring and anomaly detection systems.
  - Module-5: AI-driven retail analytics, smart traffic monitoring systems, ADAS and edge-based video analytics solutions.

- **Authentic Intelligence Reflection:**
  - Module-1: Students analyze limitations of video analytics systems, challenges in feature extraction, and the need for human interpretation in complex scenarios.
  - Module-2: Students evaluate accuracy issues in segmentation and tracking, challenges in dynamic environments, and robustness of motion detection techniques.
  - Module-3: Students assess model performance, overfitting issues, and limitations of deep learning models in real-time video applications.
  - Module-4: Students examine ethical concerns in surveillance, privacy issues, and biases in behavior and facial recognition systems.
  - Module-5: Students evaluate reliability of AI in traffic and retail analytics, challenges in edge deployment, and ethical implications of large-scale video monitoring.

RESEARCH INTEGRATION TEACHING

1. Paper review of recent research in video analytics and AI
2. Case studies on real-world surveillance and intelligent systems
3. Mini presentations on emerging topics in deep learning and computer vision
4. AI-assisted research using tools for literature review and comparison
5. Critical analysis of research gaps, limitations, and ethical considerations

ASSESSMENT SCHEME:

1. **"Internal Assessment (IA)"** refers to the Continuous Internal Assessment (CIA) with a weightage of 60%. It includes:
    a. **Mid Semester Examination (MSE):** One descriptive examination to evaluate conceptual understanding of AI fundamentals and problem-solving approaches.
    b. **Continuous Assessment Tools (CAT – Six Components):** It may include: Surprise test (Computer-based), Quiz, Project based learning, Reflection Note on Guest Lecture, Industrial visit, E-course certification, Building models, Group discussion, Case study, Seminar, or Open Book Assignment.
    c. **AI-Powered Assignment (APA):** Use of AI tools for problem-solving while simultaneously developing Authentic Intelligence through critical evaluation, ethical reasoning, and responsible decision-making.
2. **"External Assessment (EA)"** refers to the Semester End Examination (SEE) with a weightage of 40%. This examination evaluates students' overall understanding of AI concepts, application skills, and their ability to analyze and solve problems effectively.

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

AI-POWERED ASSIGNMENTS:

AI^2 Pedagogy: It is a structured teaching-learning approach where students use AI tools for problem-solving while simultaneously developing Authentic Intelligence through critical evaluation, ethical reasoning, and responsible decision-making.

Assignment 1: AI-Based Video Classification Analysis

Students prompt an AI tool:
"Classify objects or activities in a given video dataset."

Students must:
1. Analyze classification results
2. Identify misclassifications and limitations
3. Suggest improvements

Assessment focuses on human judgment.

Assignment 2: Video Segmentation & Motion Detection Evaluation

Students ask AI:
"Explain background subtraction and motion detection techniques."
Students must verify:

1. Accuracy of segmentation methods
2. Performance in dynamic environments
3. Limitations in real-world scenarios

Assignment 3: AI-Based Tracking & Debugging Challenge

AI generates object tracking output/code
Students must:

1. Identify tracking errors or inconsistencies
2. Debug or improve the method
   Explain the issues and corrections

Assignment 4: Feature Extraction & Visualization

Students use AI tools to:

1. Extract features (HOG, SIFT, etc.) from video frames
2. Visualize feature representations
3. Analyze effectiveness of features

Assignment 5: Video Analytics System Evaluation

Students prompt AI:
"Design a video surveillance system for anomaly detection."

Students must compare:

1. Accuracy vs computational cost
2. Real-time feasibility
3. Ethical and privacy concerns

Suggested Reading:

1. Patil, Lalit V., et al. – Video Analytics, Nirali Prakashan, 2023.
2. Tekalp, A. Murat – Digital Video Processing, Pearson, 2015.

E-Resources: (Recommended Tools)

- Jones, Graeme A., et al. – Video-Based Surveillance Systems: Computer Vision and Distributed Processing, Springer
- Dey, Nilanjan, et al. – Applied Video Processing in Surveillance and Monitoring Systems, IGI Global
- Shan, Caifeng, et al. – Intelligent Video Surveillance Systems, Wiley
- Programming: Python (OpenCV, basic video processing), MATLAB (optional)
- Visualization: Python libraries (Matplotlib, Seaborn), video visualization tools
- AI Tools: ChatGPT / AI assistants, computer vision libraries (OpenCV, TensorFlow, PyTorch)
- Platforms: Video analytics frameworks, annotation tools (CVAT, Labeling)

Activity Based Learning (Suggested Activities in Class)

1. Collaborative Activity is minor project development with a team of 4 students.

****

FOUNDATION MODELS FOR ROBOTICS
Professional Elective-VII

Subject Code	:	25AM4705	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 Hours
L–T–P–S	:	2–0–0–2			

Course Learning Objectives:

Students will:

- Understand the principles and capabilities of foundation models in AI systems
- Analyze limitations of traditional robotics pipelines and the need for integrated intelligence
- Apply language and vision models for robotic perception and interaction
- Examine how foundation models enable reasoning and decision-making
- Design robotic applications integrating multimodal AI models
- Evaluate reliability, safety, and ethical implications of AI-driven robotic systems

Course Description :

To enable students to understand and apply foundation models (LLMs, VLMs, and Vision-Language-Action models) in robotic systems for perception, reasoning, and task-level decision-making, representing a paradigm shift from traditional modular pipelines to integrated intelligent systems.

This course emphasizes how foundation models transform robotic autonomy by enabling language understanding, cross-modal reasoning, and generalization across environments, moving beyond rigid perception–planning–control architectures toward flexible, intelligent robotic behavior.

Course Outcome	Description	Bloom's Taxonomy Level
CO1	Analyze and describe the principles and role of foundation models in robotics	L4
CO2	Apply language and vision models for perception and interaction tasks	L3
CO3	Analyze and utilize foundation models for reasoning and decision-making in robotics	L4
CO4	Develop and integrate foundation models into robotic subsystems	L5
CO5	Design and implement a complete multimodal robotic system	L6

UNIT – I : Foundations of Robotics Paradigm Shift	**09 Hours**
Evolution of robotics systems, modular vs integrated architectures, introduction to foundation models (LLMs, VLMs, VLAs), representation learning, limitations of classical robotics Applications: Natural language-based robot control, generalization across environments	
UNIT – II : Language and Vision Models for Robotics	**09 Hours**
Vision-Language Models (VLMs), multimodal embeddings, grounding language in perception, semantic scene understanding, perception-language alignment, linking perception outputs to actionable representations for robotic tasks. Applications: Language-guided object detection, scene interpretation, robot perception enhancement.	
UNIT – III : Reasoning and Decision-Making with Grounded Robotics Actions	**09 Hours**
Task decomposition, instruction following, prompt-based planning, reasoning strategies, grounding reasoning into executable robotic actions, decision-making under uncertainty, action abstraction and task-level planning. Applications: High-level task planning with executable robot actions, instruction-driven embodied agents.	
UNIT – IV : Robotics Applications using Foundation Models	**09 Hours**
Overview of applications including manipulation using vision-language-action models, semantic navigation, human-robot interaction, learning from demonstration, action execution constraints, embodied decision-making, case studies (RT-2, SayCan, PaLM-E) Applications: Language-driven manipulation, context-aware navigation, assistive and service robotics	
UNIT – V : System Integration, Evaluation, and Future Robotics Intelligence	**09 Hours**
System architectures for AI-driven robotics, ROS integration with foundation models, deployment challenges (latency, compute), evaluation of foundation model performance in robotics, failure case analysis (hallucination, grounding errors, safety risks), future trends ***Applications:*** End-to-end robotic systems, cloud vs edge AI in robotics	

Mapping Levels of COs to POs / PSOs

COs				Program Outcomes (POs)								PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO1	3	3	1	1	1	-	1	-	-	-	3	3	1
CO2	3	2	2	2	3	-	1	1	1	-	2	3	3
CO3	3	3	2	2	2	-	2	1	1	-	3	3	2
CO4	2	2	3	2	3	-	1	2	1	1	2	2	3
CO5	2	3	3	2	3	-	1	2	1	1	3	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI Integration & Authentic Intelligence Reflection (AI^2 Pedagogy)

AI Integration:

Module-1: Use of pretrained AI models in robotics workflows, API-based model access

Module-2: Integration of VLMs with robotic perception pipelines, transforming sensor data into actionable representations.

Module-3: LLM-based planners, prompt engineering for robotics tasks, integration of reasoning outputs with robotic execution pipelines.

Module-4: Combining perception, reasoning, and action in embodied systems

Module-5: Model orchestration, system-level integration strategies

Authentic Intelligence Reflection:

Module-1: Model bias, lack of interpretability, over-reliance on pretrained systems

Module-2: Visual misinterpretation, dataset bias, perception uncertainty, and challenges in grounding perception to real-world actions.

Module-3: Hallucination risks, mismatch between reasoning and physical execution, reliability and safety concerns in embodied decision-making.

Module-4: Real-world uncertainty, safety in human environments, embodiment limitations

Module-5: Accountability, trust, and real-world failure consequences

Mini Project (Mandatory)

Students will design a robotics system utilizing foundation models.

Examples:

- Language-guided object manipulation in simulation
- Vision-language-based navigation system
- Instruction-following robotic assistant

Mandatory Requirement:

All projects must demonstrate grounding of model outputs into robotic actions (simulation or real-world).

Students must analyze:

- System design and architecture
- AI model integration
- Task reasoning capability
- Robustness and limitations
- Ethical and safety considerations

Evaluation Focus

- System Design Clarity
- Level of Autonomy Achieved
- Grounding Fidelity
- Robustness Under Uncertainty
- Failure Handling and Recovery

Recommended Tools

- Python
- OpenAI / HuggingFace APIs
- Integration with ROS-based pipelines (optional but strongly encouraged)
- Simulation tools (Gazebo, PyBullet, Isaac Sim)

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

AI Tool Usage for Teaching and Learning

AI tools will be used in this course as structured pedagogical support to enhance understanding, system design, and problem-solving.

Instructor Use (AI as Co-Teacher)

- Generate alternative explanations, examples, and case studies
- Demonstrate system workflows, debugging strategies, and design variations

Student Use (AI as Teaching Assistant)

- Concept clarification and understanding
- Code assistance and debugging (with explanation)
- System design exploration and documentation support

Guidelines for Use

- All AI-assisted work must be understood and explainable by the student
- Blind use of AI-generated content is not acceptable
- Students must critically evaluate AI outputs and identify errors or limitations

Evaluation Integration

- Viva and project discussions will assess individual understanding
- Students may be asked to explain or modify AI-assisted solutions

Suggested Reading

- Russell, S., Norvig, P., Artificial Intelligence: A Modern Approach, 4th Edition, Pearson, 2020
- Goodfellow, I., Bengio, Y., Courville, A., Deep Learning, MIT Press, 2016
- Thrun, S., Burgard, W., Fox, D., Probabilistic Robotics, MIT Press, 2005
    - **Research Papers:**
- Recent survey papers on foundation models in robotics (e.g., embodied AI surveys)
- Google RT-2: Vision-Language-Action Models
- SayCan: Language to Robot Action Grounding
- PaLM-E: Embodied Multimodal Models
- Recent publications from ICRA, RSS, CoRL (Post-2020)

****

AI FOR AGRICULTURE

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Subject Code	:	25AM4706	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 Hours
L–T–P–S		2–0–0–2	**CIE + SEE**	:	100 Marks

Course Learning Objectives:

This course will enable students to:

1. Understand the role of Artificial Intelligence, Machine Learning, and Computer Vision in modern agriculture.
2. Apply AI techniques for crop monitoring, yield prediction, and precision farming.
3. Analyze agricultural data from sensors, drones, and weather systems for intelligent decision-making.
4. Develop AI-enabled smart farming solutions using IoT, robotics, and automation tools.
5. Evaluate sustainability, efficiency, and future opportunities of AI-driven agriculture systems.

Course Description:

AI in Agriculture introduces the application of Artificial Intelligence technologies in transforming traditional farming into smart and sustainable agriculture. The course covers machine learning models for crop prediction, computer vision for disease detection, IoT sensors for soil and weather monitoring, and robotics for automated farming operations.

The Course explore how AI supports precision agriculture, optimizes resources, improves productivity, and reduces environmental impact. Topics include predictive analytics, deep learning, drones, autonomous tractors, agricultural data mining, and future trends in smart farming ecosystems.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Explain fundamentals of AI, ML, and computer vision in agriculture	L2
2	Apply AI models for agricultural data analysis and prediction	L3
3	Analyze crop disease, pest, and soil-health detection systems	L4
4	Develop AI-based smart farming solutions using IoT and robotics	L3
5	Analyze sustainability and efficiency of AI-enabled agriculture systems	L4

Course Outcome:

Course Modules :

UNIT – I : Introduction to AI in Agriculture	**09 Hours**

Evolution of agriculture technology, challenges in conventional farming, need for AI in agriculture, overview of AI, ML and data analytics, smart farming concepts, digital agriculture, AI pipeline (data → model → decision). ***Applications:*** Smart irrigation, crop advisory systems, farm monitoring.

UNIT – II : Precision Agriculture & Computer Vision	**09 Hours**

Precision agriculture concepts, Site-Specific Crop Management (SSCM), Variable Rate Technology (VRT), computer vision basics, image acquisition, crop disease detection, pest identification, weed recognition, crop maturity detection using CNN models ***Applications:*** Disease diagnosis, automated

spraying, crop quality grading.
UNIT – III : AI, IoT & Robotics in Farming **09 Hours**
IoT sensors for soil moisture, pH, temperature and humidity, sensor networks in farms, drones for surveillance and spraying, autonomous tractors, harvesting robots, robotics in greenhouse farming, AI + IoT integration. ***Applications:*** Smart irrigation, autonomous farming machines, precision spraying.
UNIT – IV : Agricultural Data Analytics & Prediction **09 Hours**
Agricultural datasets, data preprocessing, classification, clustering, regression, yield prediction models, weather forecasting, crop recommendation systems, satellite image analytics, decision support systems. ***Applications***: Yield estimation, crop planning, market forecasting.
UNIT – V : Sustainability, Future Trends & Smart Ecosystems **09 Hours**
Climate-smart agriculture, sustainable farming practices, AI for water management, carbon-efficient farming, blockchain in agriculture, future of agri-tech startups, ethical issues, challenges in rural deployment, future intelligent farming ecosystems. ***Applications:*** Smart villages, climate-resilient farming, agri-business intelligence.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO1	3	2	1	-	2	-	1	-	-	-	2	1	1
CO2	3	3	2	1	3	-	-	-	-	-	2	2	2
CO3	3	3	3	2	3	-	-	-	-	-	2	2	2
CO4	3	3	3	2	3	1	1	2	1	-	2	2	3
CO5	3	3	2	2	2	2	3	1	1	1	3	2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI INTEGRATION & AUTHENTIC INTELLIGENCE REFLECTION:

- **AI Integration:**
  - Module 1: AI-based farm decision systems
  - Module 2: Deep learning for crop disease detection
  - Module 3: IoT and robotics in agriculture
  - Module-4: ML for crop yield and weather prediction
  - Module-5: Sustainable AI farming ecosystems.

- **Authentic Intelligence Reflection:**
  - Module-1: Evaluate reliability of AI crop prediction models
  - Module-2: Identify bias in agricultural datasets
  - Module-3: Analyze ethical impact of farm automation
  - Module-4: Reflect on sustainability of AI farming tools

- Module-5: Assess rural adoption challenges.

RESEARCH INTEGRATION TEACHING

1. Case study review on smart farming startups.
2. Paper review on crop disease detection datasets.
3. Seminar on AI in Indian agriculture.
4. Mini presentations on drones and robotics.
5. Critical analysis of sustainable agriculture systems.

ASSESSMENT SCHEME:

1. **"Internal Assessment (IA)"** refers to the Continuous Internal Assessment (CIA) with a weightage of 60%. It includes:
    a. **Mid Semester Examination (MSE):** One descriptive examination to evaluate conceptual understanding of AI fundamentals and problem-solving approaches.
    b. **Continuous Assessment Tools (CAT – Six Components):** Includes activities such as quizzes, surprise tests, case studies, group discussions, project-based learning, seminars, industrial visits, reflection notes, model building, e-course certifications, or open book assignments.
    c. **AI-Powered Assignment (APA):** Students utilize AI tools for solving problems while demonstrating Authentic Intelligence through critical evaluation, validation of AI outputs, ethical reasoning, and responsible decision-making.
2. **"External Assessment (EA)"** refers to the Semester End Examination (SEE) with a weightage of 40%. This examination evaluates students' overall understanding of AI concepts, application skills, and their ability to analyze and solve problems effectively.

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

AI-POWERED ASSIGNMENTS:

AI^2 Pedagogy: It is a structured teaching-learning approach where students use AI tools for problem-solving while simultaneously developing Authentic Intelligence through critical evaluation, ethical reasoning, and responsible decision-making.

Assignment 1: Crop Disease Detection

Prompt AI tool to generate disease classification model using CNN.

Students must verify dataset quality and accuracy.

Assignment 2: Yield Prediction Study

Compare regression models for crop yield prediction.
Students evaluate performance metrics.

Assignment 3: Smart Irrigation Design

Use AI tool to generate IoT irrigation logic.
Students refine practical implementation.

Assignment 4: Drone Data Analytics

Use AI visualization tools for crop aerial data analysis.

Assignment 5: Sustainable Farming Evaluation

Compare AI farming solutions based on cost, water usage, and efficiency

Suggested Reading:

Primary Textbooks

1. Köse, U., Prasath, V. B., Mondal, M. R. H., Podder, P., & Bharati, S. (2024). *Artificial Intelligence and Smart Agriculture Technology*. Auerbach Publications. ISBN 978-1032289069.
2. Singh, R., Gehlot, A., Prajapat, M. K., & Singh, B. (2020). *Artificial Intelligence in Agriculture*. New India Publishing Agency. ISBN 978-9389547924.

Additional Resources

1. Russell, S. J., & Norvig, P. (2022). *Artificial Intelligence: A Modern Approach* (4th ed.). Pearson Education India. ISBN 978-9356063570.
2. Shelke, Suresh (2025). *AI for Agriculture*. Current Publications. ISBN 978-9349728219
3. IEEE research articles

E-Resources : (Recommended Tools)

- Python/OpenCV
- Tensorflow/ PyTorch
- Scikit-learn
- Kaggle Agricultural Datasets
- Drone Mapping Software

Activity Based Learning (Suggested Activities in Class)

1. Group discussion on AI applications in crop monitoring, irrigation, and disease detection.
2. Collaborative mini project in teams of 4 students such as smart irrigation, pest detection, or crop yield prediction system.

****

MULTIMODAL AND MULTILINGUAL LANGUAGE PERCEPTION

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Subject Code	: 25AM4707	**Credits**:	03
Hours / Week	: 04 Hours	**Total Hours**:	45 Hours
L–T–P–S	: 2–0–0–2		

Course Learning Objectives:

This course will enable students to:

1. Understand the fundamentals of multimodal and multilingual language perception.
2. Analyze how language is processed across text, speech, vision, and audio modalities
3. Explore computational models for multilingual and cross-lingual understanding
4. Apply deep learning techniques for multimodal language perception tasks
5. Evaluate real-world applications involving multilingual and multimodal AI systems

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Real-World Examples:*** *Multimodal AI systems used in translation, speech assistants, and vision-language tasks*
2. ***Problem-Solving Sessions:*** *Designing models for multilingual and multimodal perception*
3. ***Reflective Learning****: Discussions on bias, inclusivity, and ethical challenges in multilingual AI*

UNIT – I	**10 Hours**

Foundations of Multimodal and Multilingual Language Perception

Introduction to human and machine language perception, multimodal communication, modalities in language perception (text, speech, vision, gesture), multilingualism and language diversity, challenges in multilingual AI, overview of multimodal AI systems, applications in real-world scenarios.

UNIT – II	**10 Hours**

Multilingual Language Processing

Multilingual NLP fundamentals, language typology and scripts, tokenization and representation challenges, cross-lingual word embeddings, multilingual language models, transfer learning across languages, low-resource languages, language identification and code-mixing.

UNIT – III	**11 Hours**

Multimodal Representation Learning

Multimodal data representations, feature extraction from text, speech, and images, joint embedding spaces, alignment of multimodal data, fusion techniques (early, late, hybrid fusion), attention mechanisms, transformers for multimodal perception.

UNIT – IV	10 Hours
Multimodal and Multilingual Models Vision-language models, speech-text multimodal systems, multilingual speech recognition, multimodal machine translation, pretrained multimodal models, zero-shot and few-shot learning, bias and fairness in multimodal and multilingual models.	
UNIT – V	**11 Hours**
Applications, Evaluation, and Emerging Trends Multimodal sentiment analysis, emotion recognition, visual question answering, multilingual chatbots and assistants, accessibility technologies, evaluation metrics for multimodal and multilingual systems, ethical issues, privacy concerns, future research directions.	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Understand fundamentals of multimodal and multilingual language perception	L2
2	Analyze multilingual language processing techniques	L3
3	Apply multimodal representation learning methods	L4
4	Evaluate multimodal and multilingual AI models	L5
5	Design applications using multimodal and multilingual perception systems	L4

Mapping Levels of COs to POs / PSOs														
COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	3	3	1			2						1	2	1
CO2	3	3	2								1	1	3	2
CO3	3	3	3	2	2					2	1	3	3	3
CO4	3	2	2	3	3						1	2	2	2
CO5	3	3	3	3	3						2	3	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Jurafsky, D., & Martin, J. H. *Speech and Language Processing*. Pearson, 3rd Edition, 2023.
2. Baltrusaitis, T., Ahuja, C., & Morency, L. "Multimodal Machine Learning." *IEEE Signal Processing Magazine*, 2019.
3. Eisenstein, J. *Natural Language Processing*. MIT Press, 2022.

REFERENCE BOOKS:

1. Goldberg, Y. *Neural Network Methods for Natural Language Processing*. Morgan & Claypool, 2017.
2. Poria, S., et al. Multimodal AI: From Theory to Practice. Springer, 2022.
3. Lewis, M., et al. "Multilingual Language Models." *ACL Proceedings*, 2020.

E-Resources:

- https://nptel.ac.in/courses/106105235
- https://ieeexplore.ieee.org/document/10953521/
- https://www.geeksforgeeks.org/MLP-tutorials/
- MLP - Course (nptel.ac.in)

Activity-Based Learning (Suggested Activities in Class)
Hands-on activity, Quiz, Seminar

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

OPERATIONALIZING AGENTIC AI SYSTEMS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER-VII

Course Code	:	25AM4709	**Credits**	:	03
Hours/Week	:	04	**Total Hours**	:	45
L–T–P–J	:	2–0–0–2	**CIE+SEE**	:	100 Marks

Course Vision:

To enable learners to apply Artificial Intelligence techniques for understanding, analyzing, and influencing human behavior and social systems, fostering the development of responsible, ethical, and human-centered AI solutions for real-world societal challenges.

Course Objectives:

This course will enable students to:

1. Understand the fundamental principles of artificial intelligence as applied to social and behavioral science domains.
2. Analyze human behavior, social interactions, and decision-making processes using AI-driven models.
3. Apply machine learning, natural language processing, and network analysis techniques to real-world social and behavioural data.
4. Evaluate AI-based social systems with respect to fairness, ethics, privacy, interpretability, and societal impact.
5. Develop the ability to design responsible, human-centered AI solutions for behavioral prediction, social intervention, and policy support.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. **Lecture Method**: Includes structured lectures integrated with concept-driven discussions supported by contemporary research articles, global policy documents, and real-world case examples of agentic AI systems.
2. **Interactive Teaching**: Active learning through guided discussions, ethical dilemma analysis, and problem-solving sessions focused on safety, alignment, and decision-making challenges in autonomous AI agents.
3. **Visual Learning**: Use of documentaries, expert talks, system flow visualizations, and simulation videos to illustrate AI failures, governance mechanisms, and human–AI interaction dynamics.
4. **Collaborative Learning**: Group-based evaluation of AI governance frameworks, regulatory case studies, and comparative analysis of international AI policies and standards.
5. **Critical Thinking**: Faculty will pose higher-order questions encouraging students to reason about ethical trade-offs, accountability, risk mitigation, and long-term implications of agentic AI deployment.
6. **Application-Oriented Learning**: Continuous emphasis on applying ethical principles, safety strategies, and governance models to real-world agentic AI use cases across domains such as healthcare, transportation, and public systems.

UNIT–I	**09 Hours**

AI FOR SOCIAL AND BEHAVIORAL SCIENCE

Topics: Overview of social and behavioral sciences. Scope and objectives of computational social science. Role of artificial intelligence in understanding human behavior and social systems. Types of social and behavioral data: surveys, census data, digital traces, sensor data, and social media data. Human-centered AI and behavioral modeling. Differences between traditional statistical methods and AI-driven approaches. Challenges in social data analysis: bias, subjectivity, uncertainty, causality, and interpretability.

Case Study: AI-based analysis of public behavior during large-scale social or societal events.

UNIT– II	09 Hours

AI for Behavioral Analysis

Topics: Machine learning paradigms for social and behavioral data analysis. Supervised and unsupervised learning for behavioral prediction. Feature extraction and representation of social signals. Time-series and sequential modeling of human activities. Anomaly detection and behavioral profiling. Introduction to affective computing and emotion recognition. Evaluation metrics for behavioral models and fairness-aware machine learning.

Case Study: Predicting student engagement and dropout patterns using educational and interaction data.

UNIT– III	09 Hours

Natural Language Processing for Social Intelligence

Topics: Text as a primary source of social data. Natural language processing pipeline for social and behavioral analysis. Sentiment analysis, emotion detection, and opinion mining. Stance detection and topic modeling. Discourse and narrative analysis. Detection of misinformation, fake news, and hate speech. Applications and limitations of large language models in social science research.

Case Study: Public opinion analysis using social media data during elections or public debates.

UNIT– IV	09 Hours

Social Network and Group Behavior Analytics

Topics: Fundamentals of social network theory. Graph representation of social systems. Network measures such as centrality, cohesion, and connectivity. Community detection and influence modeling. Information diffusion, rumor spreading, and social contagion. Homophily, echo chambers, and polarization. Multi-agent systems for modeling collective and group behavior.

Case Study: Modeling information diffusion and influence propagation on online social platforms.

UNIT– V	09 Hours

Agent AI Systems for Social Impact

Topics: Ethical challenges in AI-driven social systems. Algorithmic bias and fairness in behavioral models. Privacy, surveillance, and data protection issues. Explainable and accountable AI for social decision-making. Applications of AI in governance, education, healthcare, and mental health. Behavioral nudging, persuasive AI, and social impact assessment. Regulatory frameworks and future directions in responsible and human-centric AI.

Case Study: Ethical evaluation of AI-based predictive policing and social risk assessment systems.

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain AI principles and architectures used in social and behavioral science applications.	L2
2	Analyze human behavior, social interactions, and decision-making processes using AI-driven models	L3
3	Apply machine learning, natural language processing, and network analysis techniques for social and behavioral data analysis.	L3
4	Examine social dynamics, group behavior, and collective decision-making using computational and AI-based approaches.	L4
5	Evaluate real-world AI-driven social systems with respect to fairness, privacy, ethical impact, and societal outcomes.	L4

Mapping Levels of COs to POs/PSOs

Table: Mapping Levels of COs to POs/PSOs

COs	Program Outcomes (POs)											PSOs	
CO 1	3	2	–	1	1	–	–	–	–	–	–	2	1
CO 2	2	3	2	2	2	–	–	–	1	–	–	3	2
CO 3	2	3	3	2	3	–	–	–	2	1	–	3	2
CO 4	2	3	2	3	2	–	–	–	2	1	–	2	2
CO 5	2	3	2	3	2	2	1	2	2	2	1	2	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

PSO-1: Apply concepts of **Artificial Intelligence, Machine Learning, and data analytics** to model, analyze, and interpret social and behavioral data for real-world applications..

PSO-2: Evaluate and design **ethical, fair, and human-centered AI systems** addressing societal challenges with considerations for privacy, bias, and policy implications.

AI Integration:

UNIT 1: AI integration in this unit is achieved through analysis of real-world social datasets and AI-driven behavioral models using tools such as Python and Google Colab. Students explore

social data sources like surveys and social media, applying AI techniques to understand human behavior and societal patterns. Case studies involving large-scale events help learners connect theoretical concepts with practical applications.

UNIT 2: This unit integrates AI through implementation of machine learning models for predicting and analyzing human behavior. Students use libraries such as Scikit-learn and TensorFlow to perform classification, clustering, and anomaly detection. Practical exercises include behavioral prediction and fairness-aware model evaluation using real datasets.

UNIT 3: AI integration focuses on analyzing textual data from social platforms using NLP techniques. Students implement sentiment analysis, opinion mining, and topic modeling using tools like NLTK and spaCy. They also explore large language model capabilities for social data interpretation, including detecting misinformation and hate speech.

UNIT 4: This unit integrates AI through graph-based modeling of social systems using tools such as NetworkX. Students analyze network structures, community detection, and information diffusion. Simulation of multi-agent interactions helps understand collective behavior and influence propagation in social environments.

UNIT 5: AI integration in this unit emphasizes evaluation and design of responsible AI systems. Students analyze ethical issues such as bias, fairness, and privacy using real-world case studies. Tools for explainability and fairness assessment are introduced, enabling learners to evaluate AI systems in domains like healthcare, governance, and public policy.

TEXT BOOKS:

1. Russell, S., & Norvig, P. Artificial Intelligence: A Modern Approach. Pearson, 2021.
2. Salganik, M. J. Bit by Bit: Social Research in the Digital Age. Princeton University Press, 2020.
3. Aggarwal, C. C. Social Network Data Analytics. Springer, 2022.
4. Jurafsky, D., & Martin, J. Speech and Language Processing. Pearson, 2023.

REFERENCE BOOKS:

1. Mitchell, T. M. Machine Learning. McGraw-Hill, 2017.
2. Jurafsky, D., & Martin, J. Speech and Language Processing. Pearson, 2023.
3. Aggarwal, C. C. Machine Learning for Text. Springer, 2018.
4. Barocas, S., Hardt, M., & Narayanan, A. Fairness and Machine Learning. MIT Press, 2023.
5. Salganik, M. J. Bit by Bit: Social Research in the Digital Age. Princeton University Press, 2020.
6. Pentland, A. Social Physics: How Social Networks Can Make Us Smarter. Penguin Books, 2015.
7. Lazer, D. et al. Computational Social Science. Science, 2009.

E-RESOURCES:

1. NVIDIA AI Learning Resources – https://developer.nvidia.com/ai
2. Coursera AI & Data Science Courses – https://www.coursera.org
3. NPTEL AI Courses – https://nptel.ac.in
4. Stanford Online AI Programs – https://online.stanford.edu
5. Hugging Face NLP & LLM Resources – https://huggingface.co/learn

RECOMMENDED TOOLS

Python, Jupyter Notebook, Google Colab, Scikit-learn, TensorFlow, NLTK, spaCy, NetworkX.

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

MULTIMODAL INTERACTION AND SPEECH INTERFACES

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER - VI

Course Code	:	25AM4710	**Credits**	:	03
Hours / Week	:	03	**Total Hours**	:	45
L–T–P–J	:	2–0–0–2	**CIE+SEE**	:	60+40 Marks

Course Vision:

This course develops an in-depth understanding of multimodal speech interfaces by integrating speech processing, machine learning, and human-computer interaction, enabling students to design intelligent systems that combine speech with visual, gesture, and contextual modalities.

Course Objectives:

Students will:

1. Understand principles of speech processing and multimodal interaction
2. Analyze speech-based and multimodal interface architectures
3. Design systems integrating speech with other modalities
4. Apply machine learning and deep learning in multimodal systems
5. Evaluate usability, ethics, and real-world applications of multimodal interfaces

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. Lecture method combined with interactive discussions
2. Brainstorming, group activities, and case-based learning
3. Demonstrations using audio-visual tools and datasets
4. Problem-solving sessions and analytical exercises
5. Encouraging critical thinking through real-world scenarios and design challenges

MODULE – I	**08 Hours**

Foundations of Multimodal Speech Interfaces

Topics:

Introduction to Human–Computer Interaction (HCI), evolution of multimodal interfaces, modalities (speech, text, vision, gesture, touch), characteristics of speech as a modality, multimodal interaction models, advantages and challenges of multimodal systems, overview of speech-enabled systems.

Applications: Voice assistants, smart devices, conversational agents, assistive technologies.

MODULE – II	**09 Hours**

Speech Signal Processing and Recognition

Topics: Speech production and perception, digital representation of speech signals, feature extraction techniques (MFCC, LPC, PLP), noise reduction and speech enhancement, speech datasets and corpora, automatic speech recognition (ASR) architecture, acoustic, language, and pronunciation models, HMM, GMM, DNN-based ASR, Text-to-Speech (TTS) systems, evaluation metrics (WER, MOS).

Applications: Speech recognition systems, voice-controlled interfaces, transcription systems.

MODULE – III	**06 Hours**

Multimodal Representation and Fusion

Topics: Multimodal data representation, feature extraction across modalities (speech, text, vision), fusion strategies (early, late, hybrid), joint embedding spaces, alignment techniques, attention mechanisms, multimodal dialogue systems, context awareness, user modeling, ambiguity resolution. **Applications:** Multimodal assistants, emotion-aware systems, interactive AI systems.	
MODULE – IV	**11 Hours**
Machine Learning for Multimodal Interfaces **Topics:** Supervised and unsupervised learning for multimodal data, deep learning models (CNN, RNN, Transformers), attention-based multimodal fusion, transfer learning, multimodal embeddings, pre-trained models, real-time processing challenges, system optimization. **Applications:** Speech-enabled AI systems, intelligent interfaces, adaptive systems.	
MODULE – V	**11 Hours**
Applications, Evaluation, and Ethical Considerations **Topics:** Applications in healthcare, education, automotive, and smart environments, usability testing and user experience evaluation, accessibility in multimodal systems, privacy and security issues, bias in speech systems, ethical considerations, fairness and inclusivity, emerging trends in multimodal speech interfaces. **Applications:** Healthcare assistants, educational tools, autonomous systems, accessibility technologies..	

AI Integration:

AI-based speech recognition and synthesis systems
Deep learning models for multimodal fusion
Transformer-based architectures for speech and multimodal tasks
Pretrained speech and multimodal models
Evaluation using benchmark datasets and performance metrics

Authentic Intelligence:

Limitations of speech recognition in diverse environments
Bias in speech datasets and accent variability
Challenges in multimodal fusion and interpretation
Ethical concerns in voice-based systems (privacy, surveillance)
Reliability and trust in intelligent multimodal systems

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Understand fundamentals of speech processing and multimodal interaction	L1
2	Analyze speech recognition and multimodal interface techniques	L2
3	Apply multimodal fusion and machine learning methods	L3

4	Evaluate multimodal speech systems using performance metrics	L4
5	Design and assess real-world multimodal speech applications	L3

Mapping Levels of COs to POs / PSOs

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO-1	3	1	–	–	–	–	–	–	–	–	–	1	2	1
CO-2	3	3	2	2	2	–	–	–	–	–	–	–	2	2
CO-3	2	3	2	2	2	–	–	–	–	–	–	–	3	2
CO-4	2	3	2	2	3	–	–	–	1	–	–	–	3	3
CO-5	1	2	2	3	3	–	–	–	2	1	–	2	2	3

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

TEXT BOOKS:

1. Jurafsky, D., & Martin, J. H., *Speech and Language Processing*, Pearson
2. Oviatt, S. et al., *The Handbook of Multimodal–Multisensor Interfaces*, ACM

REFERENCE BOOKS:

1. Shneiderman, B. et al., *Designing the User Interface*, Pearson
2. Huang, X., Acero, A., & Hon, H., *Spoken Language Processing*, Prentice Hall

E-RESOURCES:

NPTEL courses on Speech Processing and AI
Lecture materials from IITs on Multimodal Interaction
Research papers from IEEE, ACM, and Springer

RECOMMENDED TOOLS

- Programming: Python
- Libraries: TensorFlow, PyTorch
- Speech Tools: Kaldi, Librosa
- NLP Tools: NLTK, SpaCy
- Visualization: Matplotlib

HUMAN-CENTRIC AUTOMATION AND INDUSTRY 5.0
Professional Elective-VII

Subject Code	:	25AM4711	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 Hours
L–T–P–S	:	2–0–0–2			

Course Learning Objectives:

Students will:

1. Understand principles of human-centric automation and Industry 5.0
2. Analyze human-robot interaction and collaboration models
3. Apply safety, ergonomics, and human factors in system design
4. Design automation systems with humans in the loop
5. Evaluate real-world systems for usability, safety, and effectiveness

Course Description :

This course focuses on the design and deployment of human-centric automation systems with humans in the loop.

It complements autonomous and AI-driven robotics by addressing real-world deployment challenges involving human interaction, safety, and usability.

Students will develop the ability to design deployable, human-aware automation systems that integrate human capabilities into system-level decision-making.

Course Outcome	Description	Bloom's Taxonomy Level
CO1	Analyze and describe the principles of Industry 5.0 and human-centric automation	**L4**
CO2	Analyze human-robot interaction mechanisms and interaction modeling approaches	**L4**
CO3	Apply safety, ergonomics, and human factors in system design	**L3**
CO4	Design human-in-the-loop automation systems	**L6**
CO5	Develop or evaluate real-world automation systems	**L5**

UNIT – I : Industry 5.0 and Human-Centric Automation — **09 Hours**

Practical limitations of purely autonomous robotic systems in real-world deployment, limitations of full automation, role of humans in automation, human-centric design principles, introduction to collaborative systems

Applications: Smart manufacturing, assistive robotics, semi-autonomous industrial systems

UNIT – II : Human-Robot Interaction (HRI)	09 Hours
Types of interaction (physical, cognitive), collaboration modes (coexistence, cooperation, collaboration), interaction modeling and human intent inference, interfaces (gesture, voice, wearable), shared autonomy. Applications: Collaborative robots (cobots), teleoperation, assistive robotic systems	
UNIT – III : Safety, Ergonomics, and Human Factors	**09 Hours**
Safety in Human-in-the-Loop systems, risk assessment, introduction to safety standards, ergonomics, usability and cognitive load, design trade-offs between safety, efficiency, and usability Applications: Industrial robot safety, exoskeletons, workspace design	
UNIT – IV : Human-in-the-Loop Automation Systems	**09 Hours**
Design and analysis of Human-in-the-Loop control architectures, system architectures with humans in the loop, decision sharing, supervisory control, adaptive automation, failure handling. Applications: Semi-autonomous vehicles, industrial supervision systems, AI-assisted control	
UNIT – V : Industrial Applications and Deployment	**09 Hours**
Human-centric manufacturing, service robotics, Overview of deployment challenges in real environments, system integration, validation and testing, evaluation metrics for human-centric systems, human performance vs system performance trade-offs, failure case analysis in Human-in-the-Loop systems. Applications: Industrial automation, healthcare robotics, logistics systems	

Mapping Levels of COs to POs / PSOs

COs				Program Outcomes (POs)								PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO1	2	2	-	-	-	2	2	-	-	-	2	2	-
CO2	2	3	2	2	-	2	2	2	-	-	2	2	2
CO3	2	3	3	2	-	3	3	-	-	-	2	2	3
CO4	2	3	3	2	2	3	2	2	2	1	3	3	3
CO5	2	3	3	3	2	3	3	2	2	-	3	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI Integration & Authentic Intelligence Reflection (AI^2 Pedagogy)

AI Integration:

Module-1: Adaptive automation, AI-assisted decision support
Module-2: Intent recognition, learning from demonstration
Module-3: Predictive safety systems, human monitoring
Module-4: Reinforcement learning with human feedback, adaptive policies
Module-5: Adaptive workflows, real-time decision systems

Authentic Intelligence Reflection:

Module-1: Trust in automation, ethical implications of human replacement vs augmentation
Module-2: Transparency, explainability, human trust
Module-3: Ethical responsibility in safety-critical systems, inclusivity
Module-4: Accountability, human override mechanisms
Module-5: Societal impact, sustainability, workforce considerations

Mini Project (Mandatory)

Projects must explicitly demonstrate Human-in-the-Loop interaction and its measurable impact on system performance.

Examples

- Collaborative robot workflow
- Human-in-the-loop control system
- Ergonomic assistive robotic system

Minimum Requirement (Mandatory for all)

- System Definition
- Human–System Interaction Design
- System Architecture
- Safety and Risk Analysis
- Ergonomics and Usability
- Implementation / Simulation
- Failure Handling
- Evaluation Metrics

Evaluation Focus

- Clarity of System Design
- Quality of Human Integration
- Safety and Risk Consideration
- Usability and Ergonomic Design
- Technical Implementation / Simulation
- Failure Handling and Robustness
- Real-World Relevance

Recommended Tools

- Simulation tools (Gazebo, Webots)
- ROS (optional)
- Human-in-the-loop simulation frameworks
- Visualization tools

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

AI Tool Usage for Teaching and Learning

AI tools will be used in this course as structured pedagogical support to enhance understanding, system design, and problem-solving.

Instructor Use (AI as Co-Teacher)

- Generate alternative explanations, examples, and case studies
- Demonstrate system workflows, debugging strategies, and design variations

Student Use (AI as Teaching Assistant)

- Concept clarification and understanding
- Code assistance and debugging (with explanation)
- System design exploration and documentation support

Guidelines for Use

- All AI-assisted work must be understood and explainable by the student
- Blind use of AI-generated content is not acceptable
- Students must critically evaluate AI outputs and identify errors or limitations

Evaluation Integration

- Viva and project discussions will assess individual understanding
- Students may be asked to explain or modify AI-assisted solutions

Suggested Reading

1. European Commission – Industry 5.0: Towards a Sustainable, Human-Centric and Resilient European Industry, 2021
2. ISO 10218 & ISO/TS 15066 – Collaborative Robot Safety Standards

3. Goodrich & Schultz – Human-Robot Interaction: A Survey, Foundations and Trends in HCI
4. Recent IEEE / Springer / Elsevier papers on Human-Robot Interaction and Industry 5.0

****

BLOCKCHAIN FOR CYBERSECURITY AND DATA PRIVACY

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Subject Code	: 25AM4712	**Credits**:	03
Hours / Week	: 03 Hours	**Total Hours**:	xx Hours
L–T–P–S	: 2–0–0–2		

Course Vision:

To equip learners with the knowledge of blockchain technology to enhance cybersecurity and ensure secure, transparent data privacy solutions.

To enable the design of decentralized systems that protect data integrity, prevent cyber threats, and build trust in digital environments.

Course Learning Objectives:

This course will enable students to:

1. Understand the fundamentals of blockchain technology and distributed ledger systems.
2. Analyze cryptographic mechanisms used in blockchain for secure data management
3. Explore blockchain-based solutions for cybersecurity challenges
4. Apply blockchain techniques to ensure data privacy, integrity, and access control
5. Evaluate real-world blockchain applications in secure systems and privacy-preserving architectures

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Real-World Examples:*** *Incorporate real-world examples and case studies that demonstrate the relevance and application of blockchain deployments in cybersecurity, healthcare, and finance.*
2. ***Problem-Solving Sessions:*** *Organize problem-solving sessions where students can work through challenging problems together.*
3. ***Reflective Learning****: Encourage students to reflect on their problem-solving approaches and discuss the reasoning behind their solutions.*

UNIT – I	**9 Hours**

Introduction to Blockchain

Blockchain fundamentals, history and evolution, centralized vs decentralized systems, peer-to-peer networks, distributed ledger technology, block structure, blockchain types: public, private, consortium blockchains.

Application: *Using blockchain to enable secure, transparent transactions and decentralized data management across industries like finance, healthcare, and supply chains.*	
UNIT – II	**9 Hours**
Cryptography Foundations for Blockchain Security Cryptographic hash functions, digital signatures, public key infrastructure (PKI), symmetric and asymmetric encryption, key management, cryptographic security assumptions in blockchain systems. ***Application:*** Using cryptographic techniques like hashing, encryption, and digital signatures to secure blockchain transactions and protect data integrity.	
UNIT – III	**9 Hours**
Blockchain for Cybersecurity Cybersecurity threats and attack models, blockchain-based identity management, secure authentication and authorization, distributed trust models, blockchain for intrusion detection systems, DDoS mitigation using blockchain ***Application:*** Using blockchain to enhance cybersecurity through decentralized identity management, secure authentication, and protection against threats like DDoS and intrusions.	
UNIT – IV	**9 Hours**
Data Privacy and Privacy-Preserving Blockchain Techniques Data privacy challenges in distributed systems, GDPR and regulatory compliance, privacy issues in public blockchains, anonymization and pseudonymization. **Application:** Applying privacy-preserving techniques in blockchain to protect sensitive data while ensuring compliance with regulations like GDPR.	
UNIT – V	**9 Hours**
Applications, Smart Contracts, and Emerging Trends Smart contracts and security issues, secure smart contract design, vulnerabilities and attacks, blockchain in healthcare data privacy, financial data security, supply chain security, IoT security using blockchain, scalability challenges, future trends and research directions in blockchain security. **Application:** Using smart contracts and blockchain to secure healthcare, finance, supply chains, and IoT systems while addressing vulnerabilities and scalability challenges.	

AI Integration:

- Integrating AI with blockchain for intelligent data validation, fraud detection, and automated decision-making in decentralized systems.
- Integrating AI to enhance cryptographic security through anomaly detection, key management optimization, and threat prediction.

- Integrating AI with blockchain for intelligent threat detection, anomaly analysis, and automated response in distributed security systems.
- Integrating AI with blockchain to enhance data anonymization, detect privacy risks, and ensure secure handling of personal data in distributed systems.
- Integrating AI with blockchain to detect smart contract vulnerabilities, enhance security, and optimize performance in emerging decentralized applications.

Authentic Intelligence:

- Ensuring trust, transparency, and data integrity through decentralized consensus, immutable ledgers, and ethically aligned AI-driven processes.
- Ensuring trustworthy and secure systems by applying robust cryptographic principles, safeguarding keys, and maintaining transparency and integrity in data protection.
- Ensuring secure, transparent, and trustworthy cybersecurity solutions through decentralized trust, strong authentication, and ethical AI-driven monitoring.
- Ensuring ethical and trustworthy data usage through strong privacy mechanisms, regulatory compliance, and transparent handling of user information.
- Ensuring secure, reliable, and future-ready systems through robust smart contract design, continuous monitoring, and ethical, scalable blockchain innovations.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Understand blockchain architecture and distributed ledger principles	L2
2	Analyze cryptographic techniques used in blockchain security	L3
3	Apply blockchain concepts to address cybersecurity threats	L4
4	Evaluate privacy-preserving mechanisms in blockchain systems	L5
5	Design secure and privacy-aware blockchain-based applications	L4

Mapping Levels of COs to POs / PSOs													
COs				Program Outcomes (POs)								PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO1	3	3	1			2						2	1
CO2	3	3	2								1	3	2
CO3	3	3	3	2	2					2	1	3	3
CO4	3	2	2	3	3						1	2	2
CO5	3	3	3	3	3						2	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Narayanan, A., Bonneau, J., Felten, E., Miller, A., & Goldfeder, S. *Bitcoin and Cryptocurrency Technologies*. Princeton University Press, 2023.
2. Antonopoulos, A. M., & Wood, G. *Mastering Ethereum*. O'Reilly Media, 2022.
3. Drescher, D. *Blockchain Basics*. Apress, 2021.

REFERENCE BOOKS:

1. Crosby, M., et al. "Blockchain Technology: Beyond Bitcoin." *Applied Innovation Review*, 2020.
2. Bashir, I. *Mastering Blockchain*. Packt Publishing, 2023.
3. Christidis, K., & Devetsikiotis, M. "Blockchains and Smart Contracts for the Internet of Things." *IEEE Access*, 2019.

E-Resources:

- https://nptel.ac.in/courses/106105235
- https://ieeexplore.ieee.org/document/10953521/
- https://www.geeksforgeeks.org/blockchain-tutorials/
- blockchain - Course (nptel.ac.in)

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

BUILDING TRANSFORMER-BASED NATURAL LANGUAGE PROCESSING APPLICATIONS
[As per Choice Based Credit System (CBCS) scheme]
SEMESTER – VII

Course Code	**:**25AM4713	**Credits**	**:**	**3**
Hours / Week	**:** 3 Hrs	**Total Hours**	**:**	**45**
L–T–P-J	**:** 2-0-0-2			

Prerequisites:
Proficiency in a C programming language.

Course Objectives:

1. To provide a strong foundation in optimization concepts relevant to Natural Language Processing tasks.
2. To enable students to formulate NLP problems as mathematical optimization problems.
3. To study and apply classical and modern optimization algorithms used in NLP models.
4. To analyze optimization challenges in deep learning and transformer-based NLP architectures.
5. To develop the ability to select, implement, and evaluate efficient optimization strategies for large-scale NLP systems.

Teaching-Learning Process (General Instructions)

1. Classroom lectures using presentations and board work to explain theoretical concepts and mathematical foundations.
2. Problem-solving sessions and numerical examples to strengthen understanding of optimization techniques in NLP.
3. Case studies and discussion of recent research papers related to optimization in modern NLP models.
4. Hands-on demonstrations, assignments, and mini-projects using Python-based NLP and deep learning frameworks.
5. Continuous assessment through quizzes, seminars, and student presentations to encourage self-learning and critical thinking.

UNIT – I	(6–7 hours)

Fundamentals of NLP and Optimization:Introduction to NLP tasks: text classification, sequence labeling, machine translation ,Mathematical formulation of NLP problems ,Review of linear algebra, probability, and information theory for NLP,Loss functions in NLP: cross-entropy, negative log-likelihood,Convex and non-convex optimization problems.

UNIT – II	(6–7 hours)

Classical Optimization Techniques for NLP: Gradient Descent and its variantsBatch-Stochastic, and Mini-batch Gradient Descent,First-order optimization methods,Second-order methods: Newton's Method, Quasi-Newton (BFGS, L-BFGS)Optimization in traditional NLP models -Logistic Regression,Conditional Random Fields (CRF),Maximum Entropy models.

UNIT – III	(6–7 hours)

Optimization in Deep Learning for NLP:Backpropagation and computational graphs,Vanishing and exploding gradients,Advanced optimizers: Momentum, Adagrad, RMSProp, Adam, AdamW,Regularization techniques: dropout, weight decay, early stopping,Optimization for word embeddings: Word2Vec, GloVe.	
UNIT – IV	(6–7 hours)
Optimization for Sequence Models and Transformers:Optimization in RNNs, LSTMs, and GRUs,Attention mechanisms and optimization aspects,Transformer architecture and training challenges,Learning rate scheduling and warm-up, strategies,Fine-tuning vs pre-training optimization.	
UNIT – V	(6–7 hours)
Advanced and Scalable Optimization Techniques: Large-scale and distributed optimization,Gradient clipping and normalization,Mixed precision training,Hyperparameter optimization methods,Case studies from real-world NLP systems.	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
1	Formulate NLP problems as optimization tasks	C6
2	Apply classical optimization techniques to NLP models	C3
3	Implement and analyze optimization strategies in deep NLP models	C4
4	Optimize transformer-based language models	C5
5	Evaluate optimization methods for scalable NLP applications	C5

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1														
CO2														
CO3														
CO4														
CO5														

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS (TB):

1 .Jurafsky, D., & Martin, J. H., *Speech and Language Processing*, 3rd Edition (Draft),

Pearson.
2. Goodfellow, I., Bengio, Y., & Courville, A., *Deep Learning*, MIT Press.
REFERENCE BOOKS:
1.Boyd, S., & Vandenberghe, L., *Convex Optimization*, Cambridge University Press.
2.Goldberg, Y., *Neural Network Methods for Natural Language Processing*, Morgan & Claypool.
3.Murphy, K. P., *Machine Learning: A Probabilistic Perspective*, MIT Press.
4.Bottou, L., Curtis, F. E., & Nocedal, J., *Optimization Methods for Large-Scale Machine Learning*, SIAM Review.

E-Resources:
https://blog.truegeometry.com/tutorials/education/15970c4fdbca1fc48c5b412d94e798c0/JSON_TO_ARTCL_Optimization_in_Natural_Language_Processing_in_context_of_Optimiza.html?utm_source=chatgpt.com

Activity Based Learning (Suggested Activities in Class)
1.Real-world problem-solving using group discussion– Students analyze real NLP problems (text classification, sentiment analysis, machine translation) and discuss suitable optimization strategies.
2.Role play activity– Role play of optimizer components such as gradient descent, learning rate scheduler, loss function, and model parameters to understand optimization flow.
3.Demonstration of solution through programming– Instructor-led and student-driven demonstrations using Python (PyTorch/TensorFlow) to show optimization techniques like SGD, Adam, and learning rate tuning in NLP models.
4.Flipped classroom activity– Students study assigned research papers or video lectures on optimization methods (e.g., AdamW, transformer fine-tuning) and discuss insights in class.

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

HUMAN-CENTRIC AUTOMATION AND INDUSTRY 5.0

SEMESTER – VII

Subject Code	:	25OE4714	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 Hours
L–T–P–S	:	2–0–0–2			

Course Learning Objectives:

This course will enable students to:

1. Analyse the transition from Industry 4.0 to Industry 5.0
2. Evaluate human–machine collaboration models in modern industries
3. Apply AI, robotics, and IoT concepts in human-centric automation
4. Critically assess ethical, social, and sustainability challenges

Course Description:

This course provides an advanced understanding of Industry 5.0, emphasising the integration of human intelligence with smart automation systems. Students explore collaborative robotics, AI-driven industrial systems, digital twins, and human-in-the-loop frameworks. The course focuses on designing systems that enhance human capabilities, ensure sustainability, and maintain ethical standards. Case studies, industry practices, and research-oriented learning prepare students for leadership roles in future industries.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Analyse concepts and frameworks of Industry 5.0 and human-centric auton	L3
2	Evaluate enabling technologies for human–machine collaboration	L4
3	Design and analyse human-centric automated systems	L4
4	Assess ethical, societal, and sustainability issues in Industry 5.0	L4

Course Outcome:

Course Modules :

UNIT – I : Industry 5.0 – Concepts and Frameworks	**09 Hours**

Evolution from Industry 1.0 to 5.0, comparison of Industry 4.0 and 5.0, human-centric design principles, resilience and sustainability, industrial frameworks and standards.
Applications: Smart factories, resilient manufacturing systems, mass personalisation. Focus on worker-centric production and customisation.

UNIT – II : Human–Machine Collaboration Systems	**09 Hours**

Collaborative robots (COBOTS), human-in-the-loop systems, augmented intelligence, ergonomics, safety, and cognitive automation. **Applications:** Cobots in assembly lines, assistive robotics, adaptive workplaces. Focus on productivity with safety and comfort.

UNIT – III : Advanced Enabling Technologies	**09 Hours**

AI in industrial automation, Industrial IoT (IIoT), digital twins, edge computing, cyber-physical systems, and real-time analytics. **Applications:** Predictive maintenance, intelligent manufacturing, and real-time monitoring. Focus on data-driven industrial decisions.

UNIT – IV : Ethics, Sustainability	**09 Hours**
Ethical AI, workforce transformation, reskilling and upskilling, sustainability, circular economy, policies and governance. **Applications**: Green manufacturing, sustainable supply chains, responsible automation.	
UNIT – V : Human-Centric Industrial Design & Innovation	**09 Hours**
Human-centered system design, human-AI collaboration frameworks, adaptive automation, innovation in Industry 5.0 systems. **Applications:** Personalized production systems, human-robot teamwork, smart industrial design.	

Mapping Levels of COs to POs / PSOs

COs		Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	CO5	3	3	3	2	3	2	1	2	2	2	1	3	2
CO2	CO5	3	3	3	2	3	2	1	2	2	2	1	3	2
CO3	CO5	3	3	3	2	3	2	1	2	2	2	1	3	2
CO4	CO5	3	3	3	2	3	2	1	2	2	2	1	3	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI INTEGRATION & AUTHENTIC INTELLIGENCE REFLECTION:

- **AI Integration:**
  - AI-based industrial analytics
  - Human-in-the-loop AI systems
  - Predictive maintenance models
  - AI-driven sustainability optimization.

- **Authentic Intelligence Reflection:**
  - Understanding human role in automation
  - Evaluating trust and safety in AI systems
  - Analyzing risks of over-automation
  - Ethical responsibility and sustainability awareness

RESEARCH INTEGRATION TEACHING

- Paper Review on Industry 5.0 and human–robot collaboration
- Case Studies on smart factories and automation systems
- Mini Presentation on emerging industrial technologies
- AI-assisted research analysis
- Critical reflection on ethical and industrial challenges

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

ASSESSMENT SCHEME:

Internal Assessment (60 Marks)

- Continuous Assessment Tools (CAT) – 20
- Mid-Semester Examination (MSE) – 30
- AI-powered Assignment / Mini Project – 10

External Assessment (40 Marks)

- Semester End Examination (SEE)

AI-POWERED ASSIGNMENTS:

Assignment 1: Industry 5.0 System Analysis

- Analyze a smart factory using AI tools
- Evaluate human involvement

Assignment 2: Human-in-the-Loop Design

- Design a collaborative human-AI system
- Justify decision-making process

Assignment 3: Predictive Maintenance Evaluation

- Analyze AI-generated outputs
- Evaluate reliability and risks

Assignment 4: Ethical Automation Case Study

- Study real-world automation issues
- Suggest responsible solutions

Suggested Reading :
Primary Textbooks

- Liermann, Volker – The Impact of Digital Transformation and FinTech on Finance Professionals

- Boukherouaa, E. et al. – Powering the Digital Economy: AI in Finance

Additional Resources

- Bazarbash, Majid – Fintech in Financial Inclusion
- Ng, Jeffrey & Shah, Subhash – Hands-On Artificial Intelligence for Banking
- Dixon, Matthew F. et al. – Machine Learning in Finance
- Choi, Paul Moon Sub & Huang, Seth H. – FinTech with AI, Big Data, and Blockchain
- Chishti, Susanne – The AI Book

E-Resources : (Recommended Tools)

- ChatGPT / AI simulation tools
- IoT platforms (conceptual)
- Python / Excel
- Visualization tools

Activity Based Learning (Suggested Activities in Class)

- Group discussion on human–machine collaboration
- Mini project on smart industrial system design

CONVERSATIONAL AI AND SPOKEN LANGUAGE UNDERSTANDING

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER - VII

Course Code	:	25OE4715	**Credits**	:	02
Hours / Week	:	02	**Total Hours**	:	45
L–T–P–J	:	2–0–0–2	**CIE+SEE**	:	100 Marks

Course Vision:

To empower learners with the knowledge and practical skills to design intelligent conversational systems capable of understanding and responding to human language through text and speech, while fostering innovation, ethical responsibility, and research-oriented thinking in the field of Conversational AI and Spoken Language Understanding.

Course Learning Objectives:

This course will enable students to:

1. **Explain the fundamental concepts** of conversational AI, dialogue systems, and spoken language understanding, including system architectures and types of conversational agents.
2. **Apply natural language processing (NLP) techniques** such as tokenization, word embeddings, intent classification, and named entity recognition for building conversational systems.
3. **Analyze and implement speech processing techniques**, including automatic speech recognition (ASR), feature extraction (e.g., MFCC), and speech-to-text systems.
4. **Design and develop conversational agents** (chatbots and voice assistants) using machine learning and deep learning models such as sequence-to-sequence and transformer architectures.
5. **Evaluate the performance and limitations** of conversational AI systems using appropriate metrics, error analysis, and real-world testing scenarios.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. **Lecture method** along with traditional lecture method, different type of teaching methods may be adopted to develop the course outcomes.
2. **Interactive Teaching:** incorporating brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Showing **Video/animation** films to explain functioning of various concepts.
4. Encourage **Collaborative** (Group Learning) Learning in the class.
5. To make **Critical thinking**, asking Higher order Thinking questions in the class in the form of Quiz and writing programs with complex solutions.
6. Showing the **different ways to solve** the same problem and encourage the students to come up with their own creative ways to solve them

MODULE – I	**09 Hours**

Introduction to Conversational AI

Introduction to Conversational AI, Need for Conversational Systems, Types of dialogue systems (task-oriented and open-domain), Architecture of conversational systems, Rule-based vs AI-based systems, Components of conversational AI (NLU, Dialogue Management, NLG). ***Applications:***

Virtual assistants (Alexa, Siri), customer support chatbots, smart assistants, conversational interfaces in smart devices	
MODULE – II	**09 Hours**
Natural Language Processing for Conversations Text preprocessing and tokenization, Word embeddings (Word2Vec, GloVe), Named Entity Recognition (NER), Intent classification and slot filling, Text classification, Introduction to transformer-based NLP models. ***Applications:*** Chatbots, search engines, recommendation systems, automated helpdesk systems.	
MODULE – III	**09 Hours**
Spoken Language Understanding Topics Speech signal processing basics, Automatic Speech Recognition (ASR), Feature extraction (MFCC), Speech-to-text and text-to-speech systems. ***Applications:*** Voice assistants, speech-enabled apps, accessibility tools.	
MODULE – IV	**09 Hours**
Dialogue Management and Response Generation Dialogue state tracking, Rule-based vs ML-based dialogue management, Sequence-to-sequence models, Transformer models (BERT, GPT) ***Applications:*** Customer service automation, interactive AI tutors	
MODULE – V	**09 Hours**
Context-aware conversations, Multimodal conversational AI, Reinforcement learning in dialogue systems, Ethics, bias, and fairness in AI ***Applications:*** Healthcare chatbots, autonomous agents, AI companions	

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	**Explain and differentiate** the fundamental concepts, architectures, and types of conversational AI and dialogue systems.	L2
2	**Apply natural language processing techniques** such as tokenization, word embeddings, named entity recognition, intent classification, and slot filling for language understanding tasks.	L4
3	**Analyze and implement speech processing systems** using techniques such as feature extraction (MFCC), automatic speech recognition (ASR), and speech-to-text conversion.	L4
4	**Design and develop intelligent conversational agents** using machine learning and deep learning models, including sequence-to-sequence and transformer-based architectures.	L3
5	**Evaluate conversational AI systems** in terms of performance, usability, ethical considerations, bias, and real-world deployment challenges.	L5

Mapping Levels of COs to POs / PSOs

Table: Mapping Levels of COs to POs / PSOs													
COS	**Program Outcomes (POs)**											**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**
CO-1	3	2	–	–	1	–	–	–	–	–	–	2	1
CO-2	3	3	2	–	3	–	–	–	–	–	–	3	2
CO-3	3	3	2	2	3	–	–	–	1	–	–	3	2
CO-4	3	3	3	2	3	–	–	–	1	1	–	3	3
CO-5	3	3	2	2	2	1	1	3	1	1	–	2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

PSO-1: Design and build robotic systems for different applications using analytical, logical, and problem-solving skills.
PSO-2: Develop AI-powered software systems to control and operate robotic systems.

TEXT BOOKS:

1. Daniel Jurafsky & James H. Martin – *Speech and Language Processing, 3rd Edition (Draft), 2023*
2. Jacob Eisenstein – *Introduction to Natural Language Processing, 2019*

REFERENCE BOOKS:

1. Speech and Language Processing – Daniel Jurafsky & James H. Martin, 3rd Edition, 2023. A comprehensive textbook covering NLP, speech recognition, dialogue systems, and language models.
2. Introduction to Natural Language Processing – Jacob Eisenstein, 2019. Covers core NLP concepts such as text processing, embeddings, classification, and machine learning for language tasks.
3. Spoken Language Understanding – Wiley, 2011. Focuses on speech recognition, semantic understanding, and spoken dialogue systems.
4. A Pragmatic Guide to Conversational AI – 2022. Practical guide for designing chatbots using popular frameworks and conversational workflows.

E-RESOURCES:

1. Stanford NLP Course (Online)
2. HuggingFace Documentation and Tutorials
3. IEEE Research Articles on Conversational AI
4. Google AI Blog (Conversational Systems)
5. OpenAI Documentation (LLMs and Chatbots)

RECOMMENDED TOOLS

1. **Hugging Face Transformers** – For using pretrained NLP models such as BERT, GPT-style models, and speech models.
2. **NLTK** – For tokenization, stemming, text preprocessing, and NLP fundamentals.
3. **spaCy** – For Named Entity Recognition, dependency parsing, and industrial NLP pipelines.
4. **Google Speech-to-Text API** – For speech recognition and spoken language experiments.
5. **OpenAI Whisper** – For automatic speech recognition and multilingual transcription.
6. **NVIDIA NeMo Toolkit** – For speech AI, ASR, and conversational model training.
7. **TensorFlow / PyTorch** – For developing deep learning models for NLP and speech tasks.
8. **Jupyter Notebook** – For experimentation, coding, and model demonstrations.

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

****

AI FOR PSYCHOLOGY
SEMESTER -VII

Subject Code	:	25OE4716	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 Hours
L–T–P–S	:	2–0–0–2			

Course Learning Objectives:

This course will enable students to:

1. Understand basic concepts of Artificial Intelligence in psychological contexts.
2. Explore applications of AI in cognition, behavior, and mental health systems.
3. Analyze human–AI interaction and behavioral changes.
4. Evaluate ethical and societal issues of AI in psychology.

Course Description :

This course introduces students to the intersection of Artificial Intelligence and Psychology, focusing on how AI technologies influence human cognition, behavior, and mental health. It covers fundamental AI concepts and their applications in cognitive science, behavioral analysis, and psychological research.

Students explore areas such as human-AI interaction, emotional AI, and AI-driven mental health tools. The course emphasizes real-world applications like chatbots for therapy, behavioral prediction systems, and digital mental health platforms. A unique AI^2 pedagogy is followed, where students actively use AI tools and critically evaluate their outputs. The course highlights biases, ethical risks, and limitations of AI in sensitive psychological contexts.

Through case studies, discussions, and research-based learning, students develop both technical understanding and human-centered thinking. The course prepares students to responsibly use AI in psychological and social domains.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Explain fundamental AI concepts relevant to psychology	L2
2	Identify AI applications in cognitive science and mental health	L3
3	Analyze behavioral changes due to human–AI interaction	L4
4	Evaluate ethical and societal implications of AI in psychology	L4

Course Outcome:

Course Modules :

UNIT – I : Introduction to AI in Psychology	**09 Hours**
Basics of Artificial Intelligence, history and evolution, role of AI in psychology, introduction to cognitive science and computational thinking. **Applications**: AI-based cognitive modeling, behavior prediction systems.	
UNIT – II : I in Cognitive Science and Mental Health	**09 Hours**
AI applications in cognition, perception, memory, and learning; AI-based mental health tools, therapy chatbots, emotion detection systems. **Applications:** Digital mental health platforms, AI-based diagnosis support.	
UNIT – III : Human–AI Interaction and Behavioural Analysis	**09 Hours**

Human-AI interaction, user behavior analysis, emotional AI, social robots, trust and dependency on AI systems. **Applications:** Virtual assistants, human behavior tracking, adaptive interfaces.
UNIT – IV : Ethical and Social Implications of AI in Psychology **09 Hours**
Ethical AI, privacy concerns, bias in AI systems, psychological risks of AI, societal impact of AI in mental health. **Applications:** Responsible AI in therapy, ethical AI-based decision systems.
UNIT – V : AI for Psychological Research and Future Trends **09 Hours**
AI in psychological research, data-driven behavioral studies, future of AI in psychology, human-AI collaboration in mental health. **Applications:** AI-assisted research, personalized mental health solutions.

Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	3	2	1	1	2							2	2	1
CO2	3	3	2	1	2	1		1		1		2	2	1
CO3	3	2	1	2	2			2				2	2	1
CO4	3	3	2	2	2	1		2		1	2	2	2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI INTEGRATION & AUTHENTIC INTELLIGENCE REFLECTION:

- **AI Integration:**
  - AI-based behavioral analysis tools
  - Emotion recognition systems
  - Mental health chatbots
  - AI-driven cognitive modeling
- **Authentic Intelligence Reflection:**
  - Understanding human emotions vs AI predictions
  - Evaluating trust in AI-based mental health systems
  - Identifying risks of over-dependence on AI
  - Ethical use of AI in sensitive psychological data

RESEARCH INTEGRATION TEACHING

- Paper review on AI in mental health and cognitive science
- Case studies on AI therapy systems and behavioral analytics
- Mini presentations on emotional AI and social robotics
- AI-assisted research and comparison
- Critical reflection on ethical and psychological challenges

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

A. AI-POWERED ASSIGNMENTS:

AI^2 Pedagogy: It is a structured teaching-learning approach where students use AI tools for problem-solving while simultaneously developing Authentic Intelligence through critical evaluation, ethical reasoning, and responsible decision-making.

Assignment 1: AI-Based Behavior Analysis
- Use AI to analyze human behavior patterns
- Evaluate accuracy and limitations

Assignment 2: Mental Health Chatbot Evaluation
- Analyze AI chatbot responses
- Identify emotional accuracy and bias

Assignment 3: Human-AI Interaction Study
- Study behavioral changes using AI tools
- Evaluate trust and dependency

Assignment 4: Ethical Case Study
- Analyze ethical issues in AI-based mental health systems
- Suggest responsible solutions

Suggested Reading:

1. Artificial Intelligence: A Modern Approach, Stuart Russell and Peter Norvig, Prentice Hall, Third Edition, 2009.
2. Human-Centered AI, Ben Shneiderman, Oxford University Press, 2022.
3. Thinking, Fast and Slow, Daniel Kahneman, Farrar, Straus and Giroux, 2011.
4. Artificial Intelligence in Behavioral and Mental Health Care, David D. Luxton, Academic Press, 2016.
5. The Cambridge Handbook of Artificial Intelligence, Keith Frankish and William M. Ramsey, Cambridge University Press, 2014.
6. Affective Computing, Rosalind W. Picard, MIT Press, 1997.
7. The Oxford Handbook of Cognitive Science, Susan F. Chipman, Oxford University Press, 2013.

E-Resources : (Recommended Tools)

- ChatGPT / AI tools

- Basic Python (optional)
- Survey & behaviour analysis tools
- Visualization tools

Activity Based Learning (Suggested Activities in Class)

- Group discussion on AI impact on human behavior
- Case study on AI in mental health systems

****

GENERATIVE AI / LLM

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER -

Course Code	:	25OE4717	**Credits**	:	03
Hours / Week	:	05	**Total Hours**	:	45
L–T–P–J	:	3–0–2–0	**CIE+SEE**	:	

Course Vision:

To develop skilled learners who can design, build, and deploy intelligent AI systems using modern Generative AI, Large Language Models (LLMs), and agent-based architectures, while fostering the ability to apply AI solutions to real-world problems with efficiency, scalability, and ethical awareness.

Course Objectives:

This course will enable students to:

1. Explain the concepts of Generative AI, Agentic AI, and Transformer architectures
2. Understand how LLMs process and generate data.
3. Apply prompt engineering, Build basic LLM-powered applications using modern frameworks
4. Implement Retrieval-Augmented Generation (RAG) systems ,Develop AI agents for multi-step reasoning tasks.
5. Understand AI deployment pipelines ,Use tools like NVIDIA NIM, Docker, and APIs for scalable systems.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. **Lecture method** along with traditional lecture method, different type of teaching methods may be adopted to develop the course outcomes.
2. **Interactive Teaching:** incorporating brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Showing **Video/animation** films to explain functioning of various concepts.
4. Encourage **Collaborative** (Group Learning) Learning in the class.
5. To make **Critical thinking**, asking Higher order Thinking questions in the class in the form of Quiz and writing programs with complex solutions.
6. Showing the **different ways to solve** the same problem and encourage the students to come up with their own creative ways to solve them

UNIT – I	**09 Hours**

Introduction to Transformer-Based Natural Language Processing and Generative AI

Topics: Agentic AI concepts ,Generative AI fundamentals ,Transformer-based LLMs (NER, classification, QA).

Applications: Chatbots, Text summarisation tools, Email/Content generation, Basic AI assistants.

UNIT – II Word Embeddings, Tokens, and NLP	**09 Hours**

LLM Development & Prompt Engineering

Topics: Natural Language Processing and Language Models [NCA-GENL], Tokens and Tokenization [NCA-GENL], Word Embeddings [NCA-GENL], [NCA-GENM]

Applications: Resume generators ,ChatGPT-like apps ,Q&A systems, Code assistants.

UNIT – III Large Language Models and the Transformer	09 Hours
RAG & Agent Systems **Topics:** Retrieval-Augmented Generation (RAG) ,Vector databases , AI agents (multi-step reasoning) ,Research agents **Applications:** Document search systems ,Chatbots with memory ,Legal/medical assistants Research automation tools.	
UNIT – IV Multimodal Learning and its Applications	**09 Hours**
NVIDIA NIM™ Microservices **Topics:** NVIDIA NIM microservices ,Model deployment pipelines ,Inference optimization Latency vs throughput, LLM Architecture Variants [NCA-GENL], [NCA-GENM] Scaling Laws in Training LLMs [NCA-GENL] Training Data for Larger LLMs [NCA-GENL], [NCA-GENM **Applications:** Production AI systems ,Real-time chatbots , AI-powered web apps Enterprise AI deployment.	
UNIT – V Diffusion Models in Generative AI	**09 Hours**
Advanced AI, Optimization & Specialization **Topics:** Agentic AI, NVIDIA Nemotron reasoning models and test-time scaling, Agentic workflows and human-in-the-loop systems, Pruning, Knowledge Distillation,Quantization Model Deployment and Benchmarking, Image Generation and GANs [NCA-GENL], [NCA-GENM] ets and Diffusion Models [NCA-GENL], [NCA-GENM] P and Latent Diffusion [NCA-GENL], [NCA-GENM] **Applications:** Image generation (AI art) ,Healthcare AI (BioNeMo) , Data augmentation Personalized AI systems.	
AUTHENTIC INTELLIGENCE 1. Understanding how models "think" using attention ,Choosing right model for right task ,Identifying bias & hallucination risks. 2. Designing effective prompts (few-shot, chain-of-thought) Understanding user intent vs model output , Building usable AI products (not just demos). 3. Combining search + reasoning ,Reducing hallucinations using RAG 4. Designing multi-step AI workflows. 5. Thinking in terms of scalability & performance , Cost vs performance trade-offs, Designing production-ready AI systems 6. Optimizing models for real-world constraints , Applying AI in domain-specific problems , Understanding efficiency vs accuracy trade-offs **AI INTEGRATION** **1.** Transformer models (BERT, GPT-like) ,Basic LLM APIs ,Python + NLP libraries **2.** LangChain ,Hugging Face ,PyTorch ,Prompt templates **3.** LangChain + LangGraph ,Vector DBs (Milvus, Redis) , APIs + document loaders 4. NVIDIA RAG Blueprints. 5. NVIDIA NIM , Docker, Kubernetes , FastAPI , Monitoring tools (Grafana, Prometheus) 6. NVIDIA NeMo ,Diffusion pipelines ,Model optimization tools MLFlow, Docker	

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Understand the fundamentals of Generative AI, LLMs, and transformer architectures.	L2
2	Apply prompt engineering and develop LLM-based applications using modern frameworks	L4
3	Design and implement RAG systems and AI agents for real-world problem solving.	L4
4	Analyze and deploy scalable AI systems using NVIDIA tools and infrastructure.	L3
5	Evaluate and optimize AI models using advanced techniques like compression, synthetic data, and multi-modal learning.	L5

Mapping Levels of COs to POs / PSOs

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)											PSOs		
	1	2	3	4	5	6	7	8	9	10	11	1	2	3
CO-1	3	2	2	–	2	2	–	3	–	–	2	3	3	3
CO-2	3	3	2	–	3	3	–	3	–	–	–	3	3	3
CO-3	3	2	2	–	3	3	–	3	–	–	–	3	3	3
CO-4	3	2	3	2	3	3	–	3	2	–	–	3	3	3
CO-5	3	–	2	–	3	3	–	3	2	2	2	3	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Coleman, J. S. (2005). *Introducing speech and language processing*. Cambridge university press.
2. Rothman, D., & Gulli, A. (2021). *Transformers for natural language processing* (Vol. 384). Packt Publishing.

REFERENCE BOOKS:

1. Krishnamoorthy, C. S., & Rajeev, S. (2018). *Artificial intelligence and expert systems for engineers*. CRC press.
2. Rothman, D., & Gulli, A. (2021). *Transformers for natural language processing* (Vol. 384). Packt Publishing.

E-RESOURCES:

1. NVIDIA DLI Courses (Official Catalog)

2. Hugging Face Course – https://huggingface.co/course
3. LangChain Docs – https://docs.langchain.com
4. OpenAI / LLM APIs Documentation
5. Papers with Code – https://paperswithcode.com
6. https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-FX-07+V1
7. https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-FX-09+V1

RECOMMENDED TOOLS

Python 3.10+ ,PyTorch ,Hugging Face Transformers ,LangChain / LangGraph ,NVIDIA NIM Docker, FastAPI ,Vector DB (FAISS / Redis)

Evaluation Scheme: CIE:

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

VIII SEM- ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

SL	Course Type	Course Code	Course Name	Teaching Hours / Week				Examination			
				Lecture	Tutorial	Practical	Project	CIE Marks	SEE Marks	Total Marks	Credit
				L	T	P	J				
1	PROJ	25AM4801	Capstone Project Phase-II	0	0	0	24	60	40	100	10
2	INT	25AM4802	Internship	0	0	0	6	100	--	100	03
			Total	0	0	0	30				**13**

PROJ-Project Work, INT-Internship, L – Lecture, T – Tutorial, P – Practical, J – Project, C – No. Of Credits

CAPSTONE PROJECT PHASE – II

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VIII

Subject Code	:	25AM4801	**Credits**	:	10
Hours / Week	:	10	**Total Hours**	:	156 Hours
L–T–P–J	:	0–0–0–24			

Course Learning Objectives:

This course will enable students to:

1. **To identify** key research questions within a field to carry out research in a team.
2. **To identify** and summarize the literature review of the relevant field.
3. **To demonstrate** relevant referencing and inculcate new skills in various aspects of academic writing.
4. **To demonstrate** the knowledge and understanding of writing the publication/report.
5. **To showcase** the strong evidence on the clarity of the argument, understanding of the selected domain area and presentation of its technical information.
6. **To detail description** of the process of carrying out the independent research in written document along with results and conclusions with reference to the existing literature.
7. **To analyze** and synthesize the new research findings.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. Lecture method means it includes not only the traditional lecture method but a different type of teaching method that may be adopted to develop the course outcomes.
2. Interactive Teaching: Adopt Active learning that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show Video/animation films to explain the functioning of various concepts.
4. Encourage Collaborative (Group Learning) Learning in the class.
5. To make Critical thinking, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the student's understanding.

COURSE CONTENT:

1. The research topic proposed by both the guide and the student team should be approved by the department chairman to proceed further.
2. A degree of industrial input and involvement will be encouraged, and can be facilitated through existing academic-industrial collaborations or by addressing specific topics that are of interest to industrial partners.
3. The problem statement should be big enough to be carried out in two phases over the two semesters i.e., VII and VIII semesters in the VI year.
4. All projects will be closely supervised by the Project Guide with ongoing feedback and guidance at all stages of the project from conception to completion.
5. The following criteria will be checked by the department chairman to approve for theresearch proposal:

a. Department staff as course guide
   1. Ability to provide research direction to the student in the chosen field of interest
   2. Ability to design an appropriate research strategy and methodology to carry out the research by student
   3. Ability to provide and evaluate the strong literature review document for the chosen research topic
   4. Ability to train students on research paper / technical writing skills
   5. Conduct reviews in regular time period and submit the evaluation to department chairman

b. Student Team
   1. To be dedicated and committed to work on a new research topic by learning new technical skills
   2. To have fair knowledge on what is product development or research topic
   3. To have constant interaction with allocated guide by providing weekly updates
   4. To be committed to complete the project and submitting the technical paper within the stipulated time framed by the university

Evaluation:

1. The problem statement selected in Capstone Project Phase - I (VII semester) will be carried in the VIII semester.
2. Phase 2 comprises of the detailed design, implementation, and testing results during the internal and external review.
3. Each Project team needs to submit the technical paper or patent or participate in hackathons and project exhibitions as well as apply for various state and national funding agencies within the stipulated time frame by the university
4. There will be CIA evaluation (Project reviews) done by a committee of senior faculty of the Department.
5. Additionally, there will be a Semester end evaluation of the work done that would include an internal Faculty and an external academic expert.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		

1	Identify and Select an Appropriate Research Problem	L1
2	Explain and Summarize Relevant Literature	L2
3	Compare and Critically Analyze Relevant Research Papers	L3
4	Construct a Research Model and Perform Evaluation	L4
5	Create and Draft Publications or Demonstrations	L5

Mapping Levels of COs to POs / PSOs

Cos	Program Outcomes (POs)											PSOs	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	2	2	1	1	1	1	1	2	2	3	2	2	1
CO2	2	2	3	2	2	1	1	1	2	2	3	2	1	2
CO3	3	3	3	2	3	2	2	1	2	2	2	2	2	2
CO4	2	3	2	3	2	1	2	2	2	2	2	3	2	2
CO5	1	1	2	1	2	1	1	1	3	3	2	2	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

****

RESEARCH INTERNSHIP/ INDUSTRY INTERNSHIP/IN-HOUSE INTERNSHIP

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VIII

Subject Code	:	25AM4802	**Credits**	:	03
Hours / Week	:	-	**Total Hours**	:	78 Hours
L–T–P–S	:	0–0–0–6			

Course Learning Objectives:

This course will enable students to:

1. To expose students to the industrial environment
2. To create competent professionals for the industry.
3. To provide possible opportunities to learn, understand and sharpen the real time technical/managerial skills required at the job
4. To work on a problem assigned by a mentor at industry, prepare action plan and complete within time limit
5. To learn, create/prepare report for Project/research as used in industry with productive and efficient way
6. To strengthen industry-institute linkage and increase employability of the students

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. Lecture method means it includes not only the traditional lecture method but a different type of teaching method that may be adopted to develop the course outcomes.
2. Interactive Teaching: Adopt Active learning that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show Video/animation films to explain the functioning of various concepts.
4. Encourage Collaborative (Group Learning) Learning in the class.
5. To make Critical thinking, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the student's understanding.

COURSE CONTENT:

1. The course includes a 16 weeks of on-job training on current industry-relevant problem through supervised self-learning approach.
2. The internship is an individual activity.
3. The student should obtain approval from the chairman/supervisor to pursue.
4. A student shall submit a brief proposal about the work to be carried out in the internship, to a coordinator within 3 weeks, after starting the internship.
5. A comprehensive report is required to be prepared and submit to the department at the end of the semester.
6. A certificate shall be attached with this report duly signed by the competent authority of the industry for the successful completion of the internship.
7. An attendance report shall also be attached with this report.
8. The CIA evaluation will be done by faculty mentor or Industry Supervisor.
9. There is no SEE Exam for this course.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Comprehend the modern tools used in the field of AIML and engineering for product development.	L2
2	Demonstrate ethical conduct and professional accountability while working in a team for the benefit of society	L3
3	Analyze the resources requirement and planning to facilitate the project success	L4
4	Assess and adapt technical skills on industry environment	L5
5	Apply the modern industry practice for internship	L3

Mapping Levels of COs to POs / PSOs													
Cos				Program Outcomes (POs)								PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	2	2	1	1	1	1	1	2	2	3	2	2	1
CO2	2	2	3	2	2	1	1	1	2	2	3	2	1	2
CO3	3	3	3	2	3	2	2	1	2	2	2	2	2	2
CO4	2	3	2	3	2	1	2	2	2	2	2	3	2	2
CO5	1	1	2	1	2	1	1	1	3	3	2	2	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

****