Course Curriculum for M Tech in Artificial Intelligence and Data Science (AI & DS)

Computer Science and Engineering Department

NATIONAL INSTITUTE OF TECHNOLOGY DELHI

(An autonomous Institute under the aegis of Ministry of Education, Govt. of India.)

Department of Computer Science and Engineering

National Institute of Technology Delhi

1.1 About the Department

The Computer Science and Engineering Department was started in 2010 along with the foundation of NIT Delhi. Initially, only the Bachelor of Technology Programme was offered with the intake 30 which presently has been increased to 120. Now, apart from B. Tech., the department also offers Master of Technology in CSE, Artificial Intelligence and Data Science (AI & DS), MCA and Ph.D. programmes which cover a number of important areas of Computer Science and Engineering, e.g., Algorithms, Computer Networks, Data Warehousing and Data Mining, Software Engineering, Machine Learning, Image Processing, Web Technologies, Data Analytics, Complex Networks, Wireless Sensor Networks etc. We provide our students with a broad undergraduate and graduate curriculum based on the application and theoretical foundations of computer science. Our faculty and students participate in interdisciplinary research. The combination of these elements makes the department an especially exciting environment in which to study and work; an environment that serves us well in our goal of providing excellence in education, research, and discovery. The department envisions producing quality graduates, capable of leading the world in the technical realm. The department is equipped with the latest configuration and high computing system with hi-speed Internet facility, both wired as well as wi-fi. The Computer Science programs at this institute are dedicated to educate students and to advance research in computer and information technology. The department has all the facilities to carry out the related teaching and research work.

1.2 Vision

- To communicate quality Computer Science education for producing globally identifiable technocrats and entrepreneurs upholding sound ethics, profound knowledge, and innovative ideas to meet industrial and societal expectations.

1.3 Mission

- To impart value-based technical knowledge and skill relevant to Computer Science and Engineering through effective pedagogies and hands-on experience on the latest tools and technologies to maximize employability.
- To strengthen multifaceted competence in allied areas of Computer Science in order to nurture creativity and innovations to adapt the ever-changing technological scenario requiring communally cognizant solutions.
- To create an appetite for research that leads to pursuing a research career or higher education in contemporary and emerging areas of computer science.
- To inculcate the moral, ethical, and social ideals essential for prosperous nation building.

M. Tech. Artificial Intelligence and Data Science (AI & DS)

2.1 Preamble

M. Tech. Artificial Intelligence and Data Science (AI & DS): The objective of the M. Tech. program in Artificial Intelligence and Data Science (AI & DS) is to prepare students to undertake careers involving innovation and problem solving using computational techniques and technologies, or to undertake advanced studies for research careers. In order to give due importance to applied as well as theoretical aspects of computing, the curriculum for the M.Tech Artificial Intelligence and Data Science (AI & DS) program covers most of the foundational aspects of computing sciences and also develops in students the engineering skills for problem solving using computing sciences. The program offered at NIT Delhi is designed to equip students with a unique blend of skill sets that include:

- Life skills orientation.
- Predominantly practice-oriented approach with access to well-equipped and specialized laboratories, and supervised internship, projects, dissertation and Ph.D Thesis.
- Hands-on technical training.
- Business perspective, along with emphasis on innovation and entrepreneurship.
- Strong theoretical foundation for computer science and engineering.
- Hard and soft skills.
- Strong research environment.
- Participate in the R&D and industrial projects.

2.2 Salient Features

- Minimum Credits requirements for completion of MTech Artificial Intelligence and Data Science (AI & DS) program is 80.
- The Curriculum is based on the guidelines of National Education Policy (NEP) – 2020.
- The curriculum is designed to meet the prevailing and ongoing industrial requirements.
- The curriculum is flexible and offers Choice Based Credit System (CBCS).
- The curriculum inherits the Value based Education and offers Interdisciplinary/ Multidisciplinary Courses.
- The Curriculum offers Digital Pedagogy & Flipped Learning with adequate motivation for Entrepreneurship/ Startups.
- The curriculum aims at the Holistic Development of the students.
- Students can attend 2 MOOC/NPTEL/any online courses (as per department list).
- Students can do any number of courses from the other IITs/NITs/or any other CFTI institutes. There will be the provision of credit transfer as per NIT Delhi norms.
- A list of online courses is proposed by the department after mapping with the existing courses and respective mentors.

2.3 Program Outcomes (POs)

PO-1	Ability to apply knowledge to design and analyze complex engineering problems using appropriate analytical methods.
PO-2	Ability to independently carry out research /investigation and development work to solve practical problems.
PO-3	Ability to write and present a substantial technical report/document.
PO-4	Post Graduates should be able to demonstrate a degree of mastery over the area as per the specialization of the program. The mastery should be at a level higher than the requirements in the appropriate bachelor program.
PO-5	Post Graduates will show the understanding of technical communication and the impact of Engineering solutions on the society and also be aware of contemporary issues.

2.4 Program Educational Objectives (PEOs)

PEO-1	Post Graduates will establish themselves as influential professionals by solving real problems through computer science knowledge and with attention to teamwork, effective communication, critical thinking, and problem-solving skills.
PEO-2	Post Graduates will demonstrate their ability to adapt to a rapidly changing environment by learning and applying new skills and technologies.
PEO-3	Post Graduates shall drive scientific and societal advancement through technological innovation and entrepreneurship.
PEO-4	Post Graduates will be prepared for excellence and leadership roles along diverse career paths, encouraging professional ethics and active participation needed for a successful career.

2.5 Program Specific Outcomes (PSOs)

PSO-1	To be able to analyse, interpret and provide solutions to the advanced software tools for designing real-life computer science and engineering problems.
PSO-2	Ability to practice as an ethical Analytics Engineer or Researcher in the evolving disciplines of Computer Science and Engineering and its allied application domains by employing soft and project management skills learned through internships, project work, and collaborative projects with industry.

3.1 Semester wise Credit Structure

Credits							**Total**
Sl. No.	**Courses**	**1st Year**		**2nd Year**			
		1st Sem	**2nd Sem**	**3rd Sem**	**4th Sem**		
1	Program Core (PC)	11	11	-	-		22
2	Program Electives (PE)	7	7	3	3		20
3	Independent Study / Term Paper (IS-TP)	2	2	-	-		4
4	Seminar (SEM)			1	1		2
5	Thesis/Dissertation (TH-DIS)	-	-	16	16		32
Total		**20**	**20**	**20**	**20**		**80**

Course Scheme

Year	FIRST SEMESTER							SECOND SEMESTER					
	Course Code	**Course Name**	**L**	**T**	**P**	**C**		**Course Code**	**Course Name**	**L**	**T**	**P**	**C**
I	ADLM 5XX	Core 1	3	0	0	3		ADLM 5XX	Core 4	3	0	0	3
	ADBM 5XX	Core 2	3	0	2	4		ADBM 5XX	Core 5	3	0	2	4
	ADBM 5XX	Core 3	3	0	2	4		ADBM 5XX	Core 6	3	0	2	4
	ADBM 6XX	Elective 1	3	0	2	4		ADBM 6XX	Elective 3	3	0	2	4
	ADLM 6XX	Elective 2	3	0	0	3		ADLM 6XX	Elective 4	3	0	0	3
	ADPM 504	Independent Study -I /Term Paper- I	0	0	4	2		ADPM 554	Independent Study -II/ Term Paper -II	0	0	4	2
		Total Credits				**20**			**Total Credits**				**20**
Year	**THIRD SEMESTER**							**FOURTH SEMESTER**					
	Course Code	**Course Name**	**L**	**T**	**P**	**C**		**Course Code**	**Course Name**	**L**	**T**	**P**	**C**
II	ADPM 601	Dissertation I	-	-	-	16		ADPM 651	Dissertation II	-	-	-	16
	ADLM XXX	MOOC Course I	3	0	0	3		ADLM XXX	MOOC Course II	3	0	0	3
	ADPM 602	Seminar I	-	-	-	1		ADPM 652	Seminar II	-	-	-	1
		Total Credits				**20**			**Total Credits**				**20**

I. Core Courses

S. No.	Course Code	Course Name	L	T	P	C		S. No.	Course Code	Course Name	L	T	P	C
1	ADLM 501	Machine Learning	3	0	0	3		6	ADBM 551	Cloud Computing	3	0	2	4
2	ADBM 502	Advanced Data Structure and Algorithms	3	0	2	4		7	ADBM 552	Advanced Artificial Intelligence	3	0	2	4
3	ADBM 503	Distributed Databases	3	0	2	4		8	ADBM 553	Deep Neural Networks	3	0	2	4
4	ADLM 504	Natural Language Processing	3	0	0	3		9	ADLM 554	Optimization Techniques	3	0	0	3
5	ADBM 505	Data Mining and Warehouse	3	0	2	4		10	ADLM 555	Computer Vision and Pattern Recognition	3	0	0	3

II. Elective Courses

S. No.	Course Code	Course Name	L	T	P	C		S. No.	Course Code	Course Name	L	T	P	C
1	ADLM 601	Mining Large Networks	3	0	0	3		12	ADBM 612	Advanced Digital Image Processing	3	0	2	4
2	ADBM 602	Distributed and Federated ML	3	0	2	4		13	ADBM 613	Motion Analytics	3	0	0	3
3	ADLM 603	Large language Models	3	0	0	3		14	ADLM 614	Game Theory	3	1	0	4
4	ADLM 604	Bioinformatics	3	0	0	3		15	ADLM 615	Reinforcement Learning and Applications	3	0	0	3
5	ADLM 605	Multimodal AI	3	0	0	3		16	ADLM 616	Information Retrieval	3	0	0	3

6	ADLM 606	Speech Recognition	3	0	0	3		17	ADBM 617	Internet of Things	3	0	2	4
7	ADLM 607	AI for Healthcare	3	0	0	3		18	ADBM 618	Big data Analytics	3	0	0	3
8	ADLM 608	Prompt Engineering	3	0	0	3		19	ADLM 619	Introduction to Cyber Security	3	0	0	3
9	ADBM 609	Robotics and Automation	3	0	2	4		20	ADLM 620	High Performance Computing	3	0	0	3
10	ADLM 610	Generative AI	3	0	0	3		21	ADBM 621	Blockchain Technology	3	0	2	4
11	ADBM 611	Quantum Computing	3	0	2	4								

Course Title: Machine Learning

Course Code: ADLM 501

L-T-P: 3-0-0

Credits: 3

Pre-requisites:

Course Outcomes:

CO-1	Define the basic concepts of machine learning.
CO-2	Awareness about off-the shelf machine learning algorithms such as logistic regression, Random Forest, and Neural Networks.
CO-3	Explain overfitting, underfitting as well as real-world applications of classical machine learning techniques.
CO-4	Understanding of ensemble techniques.

Course Articulation Matrix:

COs	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2
CO1	3		1	1		2	1
CO2	3	3	2	1	1	3	2
CO3	1		2	1	3	3	2
CO4	3	2	2	3	1	3	1

1 - Slightly; 2 - Moderately; 3 - Substantially

Syllabus:

Unit I: Basic Mathematics and Introduction for Machine Learning (08 Hours)
Linear and convex optimization, eigen values, eigen vectors and vector spaces, introduction to machine learning, and empirical risk minimization.

Unit II: Regression and Bayes Classification (06 Hours)
Linear regression, logistic regression, gradient descent, and Bayes classifier.

Unit III: Classification (10 Hours)
Support vector machine, Decision Trees, Random Forest, and Principal Component Analysis.

Unit IV: Neural Networks (08 Hours)
Distributed Transaction Processing, Distributed Concurrency Control, Distributed DBMS Reliability.

Unit V: NoSQL Databases and its Types (08 Hours)
Different types of NoSQL Databases: Key-value Stores, Wide –column Stores, Document Stores, Graph Stores.

Learning Resources:

<u>Text Books:</u>

[T1] Shalev-Shwartz, Shai, and Shai Ben-David, *"Understanding Machine Learning: From Theory to Algorithms"*, Cambridge University Press, 2014.

<u>Reference Books:</u>

[R1] Christopher M. Bishop, *"Pattern recognition and machine learning"*, Springer, 2006.

[R2] Tom Mitchell, *"Machine Learning"*, Tata Mc Graw Hill, 1997.

[R3] Marc Peter Deisenroth, A. Aldo Faisal, and Cheng Soon Ong, *"Mathematics for machine learning"*, Cambridge University Press, 2020.

Course Title: Advanced Data Structure and Algorithms

Course Code: ADBM 502

L-T-P: 3-0-2

Credits: 4

Pre-requisites:

Course Outcomes:

CO-1	Analyze the complexity of algorithms and apply asymptotic notations.
CO-2	Study of nonlinear data structure and applications.
CO-3	Study of Amortized algorithms.
CO-4	Understand and apply greedy dynamic and divide and conquer.
CO-5	Study of randomized and approximation algorithms.

Course Articulation Matrix:

COs	PO1	PO2	PO3	PO4	PO5	PO12	PSO1	PSO2
CO1	2	2	2	2	2	2	2	3
CO2	2	3	2	3	2	1	2	2
CO3	2	3	2	2	2	2	3	3
CO4	3	2	2	2	2	1	3	2

1 - Slightly; 2 - Moderately; 3 - Substantially

Syllabus:

Unit I: Introduction (8 Hours)
Introduction to Programming, Data Structure, Algorithms: Need of Algorithms Analysis, Steps in Algorithms Design, Performance of Algorithms-Asymptotic Analysis. Graphs Algorithms, Priority Queues, Skip List, Advance Tree: Heap, Splay Tree, B/B++, String and pattern matching algorithm.

Unit II: Amortized Analysis (10 Hours)
Amortized Analysis Dynamic tables, Aggregate Method, Accounting Method, Potential Method, Disjoint set union problem. Competitive Analysis and Online Algorithms -Move-To-Front (MTF)Method, Buy vs Rent Method, Lost cow problem, Secretary Problem.

Unit III: Probability analysis of Randomized algorithms (10 Hours)
Linearity of Expectation, Markov model and Markov inequality, Threshold phenomena in graph analysis Linear Programming Formulation of Problem, Simplex, Duality, Ellipsoid algorithm, Interior Points. Approximation Algorithms Type of algorithmic Problems, one way of coping with NP hardness, TSP, Vertex Cover

Unit IV: Advanced Design and Analysis Techniques (8 Hours) Greedy Method, Divide and Conquer, recurrence relation, substitution Method, Master Theorem, Dynamic

Programming, Knap Sack Problem. Parallel Algorithm and External Memory Algorithm.

Learning Resources:

Text Books:

[T1] J. Kleinberg and E. Tardos, *"Algorithm Design"*, Addison Wesley, 2005.
[T2] TH. Cormen, CE. Leiserson, RL. Rivest and C. Stein, *"Introduction to Algorithms"*, MIT Press, 2001.

Reference Books:

[R1] Aho, JE Hopcroft and J.D. Ullman, *"The Design and Analysis of Computer Algorithms"*, Addison Wesley, 1974.
[R2] S Sahni, *"Data Structures, Algorithms and Applications in C++"*, McGraw Hill, 2001.
[R3] M.T. Goodrich and R. Tamassia, *"Algorithm Design: Foundations, Analysis and Internet Examples"*, John Wiley & Sons, 2001.

Other Suggested Readings:

List of Experiments:

1. Study and implementation of Dijkstra's Algorithm and Bellman Ford
2. Study and implementation of Kahn's algorithm and Dinic's algorithm
3. Study and implementation of Ford–Fulkerson algorithm
4. Study and implementation of Prim's algorithm and Kruskal's algorithm
5. Study and implementation of basic operations associated with B+ Tree
6. Study and implementation of K Dimensional tree
7. Study and implementation of Rabin-Karp Algorithm
8. Study and implementation of KMP Algorithm and Union by rank algorithm
9. Study and implementation of Path compression
10. Study of Research paper

Course Title: Distributed Databases

Course Code: ADBM 503

L-T-P: 3-0-2

Credits: 4

Pre-requisites:

Course Outcomes:

CO-1	Study and understand distributed DBMS architecture and distributed database concepts.
CO-2	Identify and apply various stages of distributed query processing.
CO-3	Analyse and evaluate distributed transaction processing, distributed concurrency control, and distributed reliability.
CO-4	Learn NoSQL databases, their types and applicability in different domains.

Course Articulation Matrix:

COs	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2
CO1	3		1	1		2	1
CO2	3	3	2	1	1	3	2
CO3	1		2	1	3	3	2
CO4	2	2	2	3	1	3	1

1=addressed to small extent
2= addressed significantly
3= addressed strongly (major part of course)

Syllabus:

Unit I: Introduction to Distributed Databases (06 Hours)
Introduction to Distributed Databases, Promises of DDBSs, Design, Issues, Distributed DBMS Architecture.

Unit II: Distributed Database Design (08 Hours)
Distributed Database Design: Design strategies (Top-down, Bottom-up), Design Issues, Data Fragmentation (Horizontal, Vertical, Hybrid), Allocation and Replication.

Unit III: Distributed Query Processing (10 Hours)
Distributed Query Processing: Overview, Objectives, Layers, Query Decomposition, Data Localization, Distributed Query Optimization, Distributed Query Execution.

Unit IV: Transaction and Concurrency Control in Distributed Databases (08 Hours)
Distributed Transaction Processing, Distributed Concurrency Control, Distributed DBMS Reliability.

Unit V: NoSQL Databases and its Types (08 Hours)

Different types of NoSQL Databases: Key-value Stores, Wide –column Stores, Document Stores, Graph Stores.

Learning Resources:

Text Books:

[T1] Stefano Ceri, Giuseppe Pelagatti, *“Distributed Databases: Principles and Systems”*, Tata McGraw-Hill Education, Indian Edition, 15th Reprint 2018.

[T2] M. Tamer Ozsu, Patrick Valduriez, *“Principles of Distributed Database Systems”*, Springer, Fourth Edition.

[T3] Dan Sullivan, *“NoSQL for Mere Mortals”, Addison-Wesley Professional”*, Indian Edition, published 2015.

Reference Books:

[R1] Saeed K. Rahimi, Frank S. Haug, *“Distributed Database Management Systems: A Practical Approach”*, John Wiley & Sons, 2010.

[R2] Paulo Cortez, *“Professional NoSQL”*, Wiley, 2011.

[R3] Pramod Sadalage, Martin Fowler, *“NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence*”, Addison-Wesley, 2013.

[R4] Ivan Bayros, *“Administering Oracle”*, BPB Publications, 2006.

Other Suggested Readings:

List of Experiments:

1. Create two databases on single DBMS and design database to horizontal fragment and share the fragments from both databases.
2. Create two databases on single DBMS and design database to vertical fragment.
3. Create two databases on single DBMS and design database to hybrid fragment and share the fragments from both database and write single query for creating view.
4. Working with Database Link in Oracle: create a Database Link with UserName and Password and create a Database Link without UserName and Password.
5. Write the code to create a private database link that points to the remote database named Employee and retrieve information from Employee.
6. Write the code to create a public database link, pub_emp_link that points to the remote database named Employee and retrieve information from Employee.
7. Write the code to create a global database link using Oracle Net Manager.
8. Write a program to implement Lamport’s Logical Clock.
9. Case study on NoSQL.

Course Title: Natural Language Processing

Course Code: ADLM 504

L-T-P: 3-0-0

Credits: 3

Pre-requisites:

Course Outcomes:

CO-1	Understand the fundamental concepts of natural language processing, including models, ambiguity, processing paradigms, and phases of NLP along with text representation in computers.
CO-2	Apply linguistic resources and tools such as corpus, WordNet, Treebank, and Finite State Automata to analyze morphology and word recognition using probabilistic models like N-grams and HMM.
CO-3	Demonstrate the ability to perform Part-of-Speech tagging, statistical and probabilistic parsing, and handle challenges like unknown words and multi-word expressions.
CO-4	Evaluate semantic analysis techniques, Word Sense Disambiguation methods, and NLP applications such as sentiment analysis, summarization, and machine translation.

Course Articulation Matrix:

COs	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2
CO1	3	2			2	3	2
CO2	3	3	2	2	3	3	2
CO3	3	3	2	2	3	3	3
CO4	3	3	2	3	3	3	3

1=addressed to small extent
2= addressed significantly
3= addressed strongly (major part of course)

Syllabus:

Unit 1: Introduction to Natural Language Processing
Introduction Human languages, models, ambiguity, processing paradigms; Phases in natural language processing, applications. Text representation in computers.

Unit 2: Linguistic Resources and Statistical Foundations
Linguistics resources Introduction to corpus, elements in balanced corpus, Treebank, PropBank, WordNet, VerbNet etc. Regular expressions, Finite State Automata, word recognition, lexicon. Morphology, acquisition models, Finite State Transducer. N grams, smoothing, entropy, HMM, Maximum Entropy.

Unit 3: POS Tagging and Parsing Techniques
Part of Speech tagging Stochastic POS tagging, HMM, Transformation based tagging (TBL), Handling of unknown words, named entities, multi word expressions. Parsing Statistical Parsing, Probabilistic parsing.

Unit 4: Semantics and Text Analysis
Semantics Meaning representation, semantic analysis, lexical semantics, WordNet, Word Sense Disambiguation, Selectional restriction, machine learning approaches, dictionary-based approaches. Text Classification Sentiment Analysis.

Unit 5: NLP Applications
Applications of NLP Spell checking, text summarization, machine translation, chatbots, question answering, text generation, speech-based systems, named entity recognition, and topic modeling.

Learning Resources:

<u>Text Books:</u>

[T1] Daniel Jurafsky and James H Martin, *"Speech and Language Processing"*, Pearson Education, 2009.

<u>Reference Books:</u>

[R1] Siddiqui T., Tiwary U. S., *"Natural language processing and Information retrieval"*, OUP, 2008.

[R2] James A, *"Natural language Understanding"*, Pearson Education, 1994.

[R3] Bharati A., Sangal R., Chaitanya V., *"Natural language processing: a Paninian perspective"*, PHI, 2000.

Course Title: Data Mining and Warehousing

Course Code: ADBM 505

L-T-P: 3-0-2

Credits: 4

Pre-requisites:

Course Outcomes:

CO-1	Explain the concept and significance of Data Mining.
CO-2	Explore Recent Trends in Data Mining such as Web Mining, Spatial-Temporal Mining.
CO-3	Analyze different mining algorithms and clustering techniques for Data Analytics.
CO-4	Design and Develop a Data Warehouse for an organization.

Course Articulation Matrix:

COs	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2
CO1	3	1	2	1		2	2
CO2	3	2	2		1	2	2
CO3	2	3	2	2		1	2
CO4	3	2	3		2	2	2

1 - Slightly; 2 - Moderately; 3 - Substantially

Syllabus:

Unit I: Introduction to Data Mining and Data Warehouse (8 Hours)
Design Guidelines for Data Warehouse Implementation, Multidimensional Models, OLAP–Introduction, Characteristics, Architecture, Multidimensional view, Efficient Processing of OLAP Queries, OLAP Server Architecture, ROLAP versus MOLAP Versus HOLAP and Data Cube, Data Cube Operations, Data Cube Computation. Motivation for data mining, Introduction to data mining system, Data mining functionalities, KDD, Data object and attribute types, Statistical description of data, Issues and Applications.

Unit II: Machine Learning Concepts and Approaches (6 Hours)
Supervised Learning Framework, Concepts & Hypothesis, Training & Learning, Boolean Functions and Formulae, Monomials, Disjunctive Normal Form & Conjunctive Normal Form, A Learning Algorithm for Monomials.

Unit III: Data Preparation and Mining Association Rules (8 Hours)
Data cleaning, Data integration and transformation, Data reduction, Data discretization and Concept Hierarchy Generation, Data mining primitives. Frequent patterns, Market basket analysis, Frequent item sets, closed item sets, association rules, Types of association rule

(Single dimensional, multi-dimensional, multilevel, quantitative), Finding frequent item set (Apriori algorithm, FP growth), Generating association rules from frequent item set, Limitation and improving Apriori, From Association Mining to Correlation Analysis, Lift.

Unit IV: Classification and Prediction and Cluster Analysis (8 Hours)
Issues regarding Classification & Prediction, Classification by Decision Tree Induction, Bayesian Classification, Classification by Back Propagation, k-Nearest Neighbour Classifiers, Genetic Algorithms, Rough Set & Fuzzy Set Approaches.

Unit V: Cluster Analysis (6 Hours)
Types of Data in Clustering Analysis, Categorization of Major Clustering Methods, Hierarchical Methods, Density-based methods, Grid-based methods, Grid-based methods, Model-based Clustering Method.

Learning Resources:

Text Books:

[T1] Jiawei Han and Micheline Kamber, *"Data Mining Concepts and Techniques"*, Morgan Kaufmann, 2011.
[T2] Eibe Frank and Ian H. Witten, *"Data Mining: Practical Machine Learning Tools and Techniques"*, Morgan Kaufmann, Third Edition,2011.
[T3] Pang-Ning Tan Michael Steinbach Vipin Kumar, *"Introduction to Data Mining"*, Pearson, Second Edition, 2016.

Reference Books:

[R1] Abraham Sibertschatz, Henry F. Korth and S. Sudarshan, *"Database Concepts"*, McGraw Hill, Seventh Edition, 2019.

Other Suggested Readings:

List of Experiments:

1. Load Data from heterogenous sources including text files into a predefined warehouse schema.
2. Design a data mart for a bank to store the credit history of customers in a bank. Use this credit profiling to process future loan applications.
3. Feature Selection and Variable Filtering (For very large datasets).
4. Association Mining in large datasets.
5. Interactive Drill-Down, Rollup, Slice and Dice Operations.
6. Generalized E Mandk – Means Cluster Analysis.
7. Generalized Additive Models (GAM).
8. General Classification Regression Tress (GTrees)
9. General CHAID (Chi- square Automatic Interaction Detection) Models.
10. Interactive Classification and Regression Trees.
11. Goodness of Fit Computations.

Course Title: Cloud Computing

Course Code: ADBM 551

L-T-P: 3-0-2

Credits: 4

Pre-requisites:

Course Outcomes:

CO-1	Describe the fundamental concepts, architectures, and service models of cloud computing.
CO-2	Analyze virtualization technologies and resource management techniques used in cloud environments.
CO-3	Apply cloud platforms and services to deploy, manage, and scale distributed applications.
CO-4	Assess security, privacy, and reliability challenges in cloud computing systems.
CO-5	Evaluate emerging cloud technologies, research trends, and cloud–based solutions for complex computing problems.

Course Articulation Matrix:

COs	**POs & PSOs**						
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PSO1**	**PSO2**
CO1	3	1		1	2	1	1
CO2	3	2		2	1	3	2
CO3	3	2		2	1	2	2
CO4	3	2	1	2	1	3	2
CO5	2	3	1	2	1	2	3

1=addressed to small extent
2= addressed significantly
3= addressed strongly (major part of course)

Syllabus:

Unit I: Introduction to Cloud Computing (8 Hours)
Evolution of Distributed Computing, Grid Computing, Utility Computing, Cloud Computing Concepts, Characteristics, Cloud Reference Architecture, Service Models (IaaS, PaaS, SaaS), Deployment Models, Cloud Ecosystem and Applications.

Unit II: Virtualization and Resource Management (8 Hours)
Hypervisors, Server, Storage and Network Virtualization, Containerization, Docker, Resource Provisioning, Scheduling, Load Balancing, Data Center Architecture.

Unit III: Cloud Platforms and Services (8 Hours)
AWS, Microsoft Azure, Google Cloud Platform, Storage Services, Compute Services,

Database Services, SOA, Microservices, Cloud APIs.

Unit IV: Cloud Security and Reliability (7 Hours)
IAM, Authentication, Authorization, Data Security, Encryption, Compliance, Disaster Recovery, Fault Tolerance and Reliability.

Unit V: Advanced Cloud Computing (5 Hours)
Edge Computing, Fog Computing, Serverless Computing, Kubernetes, Multi-cloud Architectures, AI/ML in Cloud, Research Challenges and Future Trends.

Learning Resources:

Text Books:

[T1] Kai Hwang, Geoffrey C. Fox, and Jack J. Dongarra, *"Distributed and Cloud Computing: From Parallel Processing to the Internet of Things"*, Elsevier, First Edition, 2012

[T2] Rajkumar Buyya, Christian Vecchiola, S. Thamarai Selvi, "*Mastering Cloud Computing: Foundations and Applications Programming*", McGraw Hill Education, 1st Edition.

Reference Books:

[R1] Thomas Erl, Ricardo Puttini, Zaigham Mahmood, *"Cloud Computing: Concepts, Technology & Architecture"*, Pearson Education, 1st Edition 2013.

[R2] Barrie Sosinsky, *"Cloud Computing Bible"*, Wiley Publishing, Second Edition, 2011.

[R3] Dan C. Marinescu, *"Cloud Computing: Theory and Practice"*, Morgan Kaufmann, Second Edition, 2017.

Other Suggested Readings:

List of Experiments:

1. Virtual Machine Configuration using VirtualBox/VMware: To install, configure, and manage virtual machines using virtualization platforms such as VirtualBox or VMware and understand resource allocation in virtualized environments.
2. Cloud Instance Creation in AWS/Azure/GCP: To create, configure, and manage virtual machine instances on public cloud platforms and understand cloud resource provisioning.
3. Web Application Deployment on Cloud Platforms: To deploy and host a web application on a cloud platform and analyze the deployment process and service configurations.
4. Cloud Storage and Database Services: To explore cloud-based storage and database services and implement data storage, retrieval, and management in a cloud environment.
5. Docker Container Deployment: To create, deploy, and manage containerized applications using Docker and understand containerization concepts.
6. Kubernetes Orchestration: To deploy, manage, and scale containerized applications using Kubernetes and understand container orchestration mechanisms.
7. Identity and Access Management (IAM) Configuration: To configure user identities, roles, permissions, and access policies in cloud environments for secure resource management.
8. Cloud Monitoring and Performance Analysis: To monitor cloud resources and applications using cloud monitoring tools and evaluate system performance and resource utilization.
9. Load Balancing and Auto-Scaling Implementation: To implement load balancing and auto-scaling mechanisms for improving availability, scalability, and performance of cloud applications.

Course Title: Advanced Artificial Intelligence

Course Code: CSBM 552

L-T-P: 3-0-2

Credits: 4

Pre-requisites:

Course Outcomes:

CO-1	Understand the basic concepts of AI.
CO-2	Apply search strategies to solve AI problems.
CO-3	Apply knowledge representation and reasoning to solve real world AI Problems.
CO-4	Explore machine learning concepts and algorithms for real world applications.

Course Articulation Matrix:

COs	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2
CO1	3					2	2
CO2	2	3	3	3	3	3	3
CO3	2	2	3	3	3	3	3
CO4	2	2	3	3	3	3	3

1 - Slightly; 2 - Moderately; 3 - Substantially

Syllabus:

Unit I: Introduction and Automated Problem-Solving Agent (06 Hours)

What is Artificial Intelligence, History of AI, Possible Approaches in AI, Application Domains, and Modern AI, Areas Contributing to AI, Core Capabilities covered in this course, Automated Problem-Solving Agent: Intelligent Agent & Environment, Complex Problems and AI, Shannon number, Problem Representation in AI.

Unit II: Search Strategies and Logic Detection (10 Hours)

Search Strategies: Search introduction, Uninformed Search, Informed/Heuristic Search, Beyond Classical Search, Local Search, Problem Reduction, Adversarial Search, Constraint Satisfaction Problems. Logic and Deduction: Logical Agents, Propositional logic and Predicate Logic, Forward & Backward Chaining, Inferencing by Resolution Refutation.

Unit III: AI Planning (6 Hours)

AI Planning: AI Planning, Robot introduction and types, Steps in Robot Motion Planning, Graph-based Planning (Grassfire, Dijkstra & A*Algorithm), Graph Construction Methods and path planning in Configuration Space, Skeletonization [Visibility Graphs, Voronoi diagrams/Trapezoidal Decomposition, Cell decomposition [X-connected grids–lattice-based graphs], Collision Detection and Free space Sampling, Intruder Finding Problem, Probabilistic roadmaps (PRM)], Rapidly Exploring Random Trees (RRT).

Unit IV: Reasoning Under Uncertainty (6 Hours)
Quantifying Uncertainty, Basic of Probability, Probabilistic Reasoning, Bayes Net, Bayesian Network, Fuzzy Logic, Decisions Theory, Utility Function, Decision Network, Markov Decision Process, Probabilistic Reasoning overtime, Hidden Markov Model, Kalman filter, Markov Chain Monte Carlo.

Unit V: Learning from examples (8 Hours)
Reinforcement Learning, Learning Agent, Introduction to Machine Learning, Types of Machine Learning, learning from experience: Reinforcement Learning, Background, Model based and Model free learning, TD and Q Learning, RL Applications, Learning from Example, Supervised learning: Introduction, Naïve Bayes, Decision Tree, Perceptrons, Neural Network, Introduction to Deep Learning. AI Applications and Ethics, Computer Vision and Robotics, natural language understanding, AI in Healthcare, Ethics of AI.

Learning Resources:

Text Books:
[T1] Stuart Russell, Peter Norvig, *"Artificial Intelligence: A Modern Approach"*, Prentice Hall, Fourth edition,2020.

Reference Books:
[R1] Nils J. Nilsson, *"Artificial Intelligence: A New Synthesis"*, Morgan-Kaufmann, 1998.
[R2] Judea Pearl, *"Heuristics: Intelligent Search Strategies for Computer Problem Solving"*, Addison-Wesley Publishing Company, 1984.

Other Suggested Readings:

List of Experiments:

1. Introduction to Prolog programming.
2. Python Frameworks Tutorial (with Jupyter and Colab) and it's Data Structures.
3. Searching in graph-based problem space, exploring Uninformed search Techniques.
4. Exploring Informed search Techniques (Vacuum world and Maze Problem).
5. Exploring Uninformed and Informed search Techniques (PACMAN Search Space).
6. Multiagent in a search space.
7. Introduction Logical Agent and Knowledge representation using Prolog.
8. Reasoning Under Uncertainty using Bayesian Learning.
9. Reinforcement Learning using Q-Learning.
10. Introduction to Machine Learning and Python libraries for Data Analysis (Pandas, NumPy, Matplotlib).

Course Title: Deep Neural Networks

Course Code: ADBM 553

L-T-P: 3-0-2

Credits: 3

Pre-requisites:

Course Outcomes:

CO-1	Explain the fundamentals, evolution, and architectures of deep neural networks.
CO-2	Demonstrate DNN training using activation functions, loss functions, optimizers, and metrics.
CO-3	Analyze CNN, RNN, LSTM, autoencoder, and deep belief network architectures.
CO-4	Apply transfer learning, fine-tuning, and hyperparameter tuning for better model performance.
CO-5	Design, develop, evaluate, and present a DNN mini project using Python tools.

Course Articulation Matrix:

COs	PO-1	PO-2	PO-3	PO-4	PO-5	PSO-1	PSO-2
CO-1	3	1	1		1	1	3
CO-2	2	2	2	2	3	2	2
CO-3	3	3	2	2	2	2	3
CO-4	2	3	2	3	3	2	2
CO-5	2	3	3	3	3	3	2

1- Slightly; 2 - Moderately; 3 - Substantially

Syllabus:

Module-I: Foundations of Deep Neural Networks (6 hours)
AI, machine learning, ANN vs DNN; perceptron and multilayer perceptron; activations, losses, data representation, applications, advantages, and limitations.

Module-II: Training Deep Neural Networks (7 Hours)
Forward and backpropagation; gradient descent variants; SGD, RMSProp, Adam; learning rate, batch size, epochs; overfitting, dropout, L1/L2 regularization, batch normalization, early stopping, and metrics.

Module-III: Deep Learning Architectures (7 Hours)
CNN, RNN, LSTM/GRU, autoencoders, deep belief networks, and semantic hashing; architecture comparison, use cases, merits, and demerits.

Module-IV: Transfer Learning and Model Tuning (8 Hours)

Inductive, transductive, and unsupervised transfer learning; pre-trained models; feature extraction, fine-tuning, hyperparameter tuning, validation, and model selection.

Module-V: Applications and Emerging Trends (8 Hours)
Deep learning applications in computer vision, natural language processing, healthcare, education, cybersecurity, and scientific data analysis; domain-specific case studies; recent trends in deep learning, including large models and generative models; ethical issues, data privacy, bias, interpretability, model deployment, and computational challenge.

Learning Resources:

Text Books:
[T1] Seth Weidman, "*Deep Learning from Scratch*", O'Reilly Media, 1st ed., 2019.
[T2] Charu C. Aggarwal, "*Neural Networks and Deep Learning: A Textbook*", Springer, 1st ed., 2018.

Reference Books:
[R1] David Julian, "*Deep Learning with PyTorch Quick Start Guide*", Packt Publishing Limited, 1st ed., 2018.
[R2] Yuxi (Hayden) Liu, "*Hands-On Deep Learning Architectures with Python*", Packt Publishing Limited, 1st ed., 2019.
[R3] Ian Goodfellow, Yoshua Bengio, and Aaron Courville, "*Deep Learning*", MIT Press, 1st ed., 2016.

Other Suggested Readings:

List of Experiments:
1. Install Python/Jupyter and TensorFlow/Keras or PyTorch for deep learning experiments.
2. Implement perceptron and MLP models using sigmoid, tanh, ReLU, and softmax activation functions.
3. Train a feed forward DNN for classification or regression on a benchmark dataset.
4. Compare optimizers, learning rate, batch size, epochs, loss functions, and evaluation metrics.
5. Apply dropout, L1/L2 regularization, batch normalization, and early stopping.
6. Build a CNN for image classification using MNIST, Fashion-MNIST, or CIFAR-10.
7. Build an RNN/LSTM model for sequence, text, or time-series prediction.
8. Implement an autoencoder for reconstruction or dimensionality reduction.
9. Apply transfer learning and fine-tuning using a pre-trained CNN model.
10. Develop and present a mini project using a real-world dataset and a DNN model.

Course Title: Optimization Techniques

Course Code: ADLM 554

L-T-P: 3-0-0

Credits: 3

Pre-requisites:

Course Outcomes:

CO-1	To understand the fundament of Linear Programming and Dynamic Programming.
CO-2	Enumerate fundamentals of Integer programming technique and apply different techniques to solve various optimization problems arising from engineering areas.
CO-3	Identify appropriate optimization method to solve complex problems involved in various industries.
CO-4	To understand the graphical, simplex, and analytical methods for making effective decision.

Course Articulation Matrix:

COs	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2
CO1	2					1	
CO2	1	1	2			1	2
CO3	1	2	1	2	2	2	2
CO4	2	2	2	2	1	2	

1=addressed to small extent
2= addressed significantly
3= addressed strongly (major part of course)

Syllabus:

Unit 1:
Preliminaries: Proofs, Vector Spaces and Matrices, Linear Transformations, Eigenvalues and Eigenvectors, Orthogonal Projections, Quadratic Forms, Matrix Norms, Concepts from Geometry, Elements of Calculus.

Unit 2:
Unconstrained Optimization: Basics of Set Constrained and Unconstrained Optimization, One-Dimensional Search Methods, Golden Section Search, Fibonacci Search, Newton's Method, Secant Method, Solving Ax = b.

Unit 3:
Linear Programming: Introduction to Linear Programming, Simplex Method, Duality.

Unit 4:
Nonlinear Constrained Optimization: Problems with Equality Constraints, Problems with

Inequality Constraints, Karush Kuhn Tucker Condition, Convex Optimization Problems.

Unit 5:

Algorithms for Constrained Optimization: Projections, Project gradient methods, Penalty methods.

Learning Resources:

Text Books:

[T1] Edwin K.P. Chong, Stanislaw H. Zak, *"An Introduction to Optimization"*, Wiley, Fourth Edition.

Reference Books:

[R1] Stephen Boyd, *"Convex Optimization"*, Lieven Vandenberghe, Third Edition.
[R2] Paulo Cortez, *"Modern Optimization with R (Use R)"*, Springer, 2014.

Course Title: Computer Vision and Pattern Recognition

Course Code: ADLM 555

L-T-P: 3-0-0

Credits: 3

Pre-requisites:

Course Outcomes:

CO-1	Apply mathematical modeling methods for low, intermediate, and high-level image processing tasks.
CO-2	Design a new algorithm to solve are cent of the art computer vision problem.
CO-3	Perform software experiments on the computer vision problems and compare their performance with the state of the art.
CO-4	Build a complete system to solve a computer vision problem.

Course Articulation Matrix:

COs	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2
CO1	1					1	
CO2	1	1	2			2	2
CO3	1	2	2	2	2	2	2
CO4	2	2	2	2	1	2	

1 - Slightly; 2 - Moderately; 3 - Substantially

Syllabus:

Unit I: Introduction to computer vision (4 Hours)
Human vision, Image formation: Geometric primitives and transformations, Photometric image formation, the digital camera, how machine sees and recognizes things, Applications, Mathematical foundations.

Unit II: Pattern Recognition (8 Hours)
Basics of pattern recognition, Design principles of pattern recognition system, Learning and adaptation, Pattern recognition approaches. Statistical Pattern Recognition: Bayesian Decision Theory, Classifiers, Normal density and discriminant functions, Dimension reduction methods– Principal Component Analysis (PCA), Fisher Linear discriminant analysis, Expectation maximization (EM).

Unit III: Convention computer vision and pattern recognition algorithms (10 Hours)
Object detection and segmentation e.g. Edge, texture, region, detection of sliding windows: Feature extraction, e.g. linear binary pattern, principal component analysis, Gabor filters, bags of features, Matching and recognition e.g. Bayesian classifier, support vector machine, fusion, Image alignment and stitching.

Unit IV: Motion estimation, Computational photography (6 Hours)
Photometric calibration, High dynamic range imaging, Super-resolution, denoising, and blur removal, Image matting and compositing, Texture analysis and synthesis.

Unit V: Deep learning for computer vision and pattern recognition (8 Hours)
Key components and basic architecture of deep neural network, Convolution neural network, Object detection using R-CNN, Segmentation using image-to-image neural network, Temporal processing, and recurrent neural network.

Learning Resources:

Text Books:
[T1] Stuart Russell, Peter Norvig, *"Artificial Intelligence: A Modern Approach"*, Prentice Hall, Fourth edition,2020.

Reference Books:
[R1] Nils J. Nilsson, *"Artificial Intelligence: A New Synthesis"*, Morgan-Kaufmann, 1998.
[R2] Judea Pearl, *"Heuristics: Intelligent Search Strategies for Computer Problem Solving"*, Addison-Wesley Publishing Company, 1984.

Other Suggested Readings:

List of Experiments:

1. Introduction to Prolog programming.
2. Python Frameworks Tutorial (with Jupyter and Colab) and it's Data Structures.
3. Searching in graph-based problem space, exploring Uninformed search Techniques.
4. Exploring Informed search Techniques (Vacuum world and Maze Problem).
5. Exploring Uninformed and Informed search Techniques (PACMAN Search Space).
6. Multiagent in a search space.
7. Introduction Logical Agent and Knowledge representation using Prolog.
8. Reasoning Under Uncertainty using Bayesian Learning.
9. Reinforcement Learning using Q-Learning.
10. Introduction to Machine Learning and Python libraries for Data Analysis (Pandas, NumPy, Matplotlib).

M. Tech. AI&DS (Elective) Syllabus

Course Title: Mining Large Networks

Course Code: ADLM 601

L-T-P: 3-0-0

Credits: 3

Pre-requisites:

Course Outcomes:

CO-1	Explain structural properties and generative models of real-world networks.
CO-2	Apply centrality, community detection, and link analysis algorithms to network data.
CO-3	Implement random walk, spectral, and deep graph learning methods for representation learning.
CO-4	Analyse influence propagation, misinformation, and cascade models in social networks.
CO-5	Design scalable graph mining pipelines for large datasets.

Course Articulation Matrix:

COs	**POs &PSOs**						
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PSO1**	**PSO2**
CO1	2	1		2		2	
CO2	3	1		2		3	1
CO3	3	2		3		3	1
CO4	2	2		3	2	2	1
CO5	3	3	1	3	1	3	2

1 = addressed to small extent
2 = addressed significantly
3 = addressed strongly (major part of course)

Syllabus:

Unit I: Foundations of Network Analysis (8 Hours)
Networks as graphs: directed, undirected, weighted, bipartite and multiplex networks; real-world network examples including social, biological, technological and information networks; scale, sparsity and dynamic nature of large networks; degree distribution and scale-free behaviour; clustering coefficient; path lengths and diameter; small-world phenomenon; degree assortativity; basic measures for analysing networks; Erdős–Rényi random graph model; Barabási–Albert preferential attachment model; Watts–Strogatz model; Kronecker graph

model and comparison with real networks; motivation for sampling large networks; node sampling and edge sampling; snowball sampling and random walk sampling; sampling bias and representativeness; adjacency matrix, adjacency list and edge list representations; sparse representations; NetworkX, iGraph, GraphX, Neo4j, SNAP and KONECT.

.

Unit II: Link Analysis and Node Importance (7 Hours)

Degree, betweenness, closeness, eigenvector, Katz, and harmonic centrality measures; PageRank formulation, random surfer model and power iteration method; Personalised PageRank; HITS algorithm and hub-authority analysis; SimRank algorithm; link farming and web spam behaviour; challenges in large-scale web ranking; MapReduce PageRank; Blocked PageRank; convergence analysis and dangling nodes.

Unit III: Community Detection (8 Hours)

Community structure, definition, and motivation; modularity and its limitations; overlapping and non-overlapping communities; graph partitioning using minimum cut, ratio cut and normalised cut; Laplacian matrix, graph spectrum and spectral clustering; community detection methods including Girvan–Newman, Louvain and Leiden algorithms; overlapping community detection using CPM, BIGCLAM and DEMON; evaluation metrics including NMI, conductance, coverage and ground-truth comparison.

Unit IV: Network Embedding and Representation Learning (9 Hours)

Motivation for embeddings and limitations of hand-crafted features; structural equivalence and homophily; matrix factorisation methods including Laplacian Eigenmaps, Graph Factorisation and GraRep; random walk-based methods including DeepWalk, Node2Vec, LINE and Metapath2Vec; graph neural networks including message passing, Graph Convolutional Networks (GCN), GraphSAGE and Graph Attention Networks (GAT); applications in node classification, link prediction and graph classification.

Unit V: Link Prediction and Recommender Systems on Graphs (5 Hours)

Link prediction problem formulation and train-test splits in graphs; heuristic methods including common neighbours, Jaccard similarity, Adamic–Adar and preferential attachment measures; embedding-based methods; matrix completion techniques; graph-based collaborative filtering using bipartite user-item graphs; handling cold-start using graph structure.

Unit VI: Information Propagation and Influence in Networks (5 Hours)

Diffusion and cascade models including Independent Cascade, Linear Threshold and SIR/SIS epidemic models; cascade behaviour and network effects; influence maximisation problem formulation; greedy approximation methods including submodularity and $(1-1/e)$ approximation concept; CELF, TIM and IMM algorithms; network-based outbreak modelling and early detection concepts; rumour models and fact-checking approaches.

Unit VII: Scalable Graph Mining and Emerging Topics (4 Hours)

Frequent subgraph mining using AGM, FSG and gSpan; applications in graph mining; dynamic and temporal networks; evolving graphs, streaming algorithms, and dynamic link prediction; heterogeneous information networks, meta-paths, HAN, and applications; network-based anomaly detection including structural anomalies and node/edge anomaly detection; fairness, privacy and ethics including bias in graph algorithms and ethical concerns in network analysis.

Learning Resources:

Text Books:

[T1] David Easley and Jon Kleinberg, "*Networks, Crowds, and Markets*", Cambridge University Press, 2010.
[T2] Jure Leskovec, Anand Rajaraman and Jeffrey D. Ullman, "*Mining of Massive Datasets*", Cambridge University Press, Third Edition, 2020.

Reference Books:

[R1] William L. Hamilton, "*Graph Representation Learning*", Morgan & Claypool Publishers, 2020.

Course Title: Distributed and Federated ML

Course Code: ADBM 602

L-T-P: 3-0-2

Credits: 4

Pre-requisites: NIL

Course Outcomes:

CO-1	Analyse distributed machine learning architectures and communication-efficient learning algorithms.
CO-2	Apply distributed optimisation techniques and federated learning algorithms for large-scale machine learning problems.
CO-3	Evaluate privacy-preserving and secure machine learning mechanisms in distributed settings.
CO-4	Design and implement federated learning systems considering heterogeneity, scalability, and robustness.
CO-5	Investigate current research challenges and emerging trends in distributed and federated AI systems.

Course Articulation Matrix:

COs	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2
CO1	2	2	2	2	2	2	3
CO2	2	3	2	3	2	2	2
CO3	2	3	2	2	2	3	3
CO4	3	2	2	2	2	3	2

1 - Slightly; 2 - Moderately; 3 - stantially

Syllabus:

Unit I: Foundations of Distributed Machine Learning (8 Hours)
Introduction to machine learning systems and distributed computing paradigms. Centralized versus distributed learning architectures. Challenges in large-scale machine learning. Data parallelism and model parallelism. Parameter server architecture. Distributed optimization techniques. Synchronous and asynchronous stochastic gradient descent. Communication protocols in distributed systems. Consensus mechanisms and gossip algorithms. Cloud, edge, and fog computing for machine learning applications.

Unit II: Federated Learning Fundamentals (10 Hours)
Introduction to federated learning and its applications. Cross-device and cross-silo federated learning frameworks. Federated optimization techniques. Federated stochastic gradient descent and Federated Averaging (FedAvg) algorithm. Aggregation methods in federated learning. Personalization and multi-task federated learning. Client selection strategies and participation

management. Communication-efficient federated learning. Handling statistical heterogeneity and non-IID data distributions. Performance evaluation metrics for federated systems.

Unit III: Privacy, Security and Trustworthy Federated Learning (10 Hours)
Privacy challenges in distributed machine learning. Differential privacy techniques for federated learning. Secure aggregation protocols. Homomorphic encryption and secure multi-party computation. Membership inference and model inversion attacks. Data poisoning and model poisoning attacks. Byzantine fault tolerance and robust aggregation methods. Fairness, accountability, transparency, and explainability in federated systems. Ethical and regulatory considerations in privacy-preserving machine learning

Unit IV: Advanced Topics in Distributed and Federated Learning (8 Hours)
Decentralized and peer-to-peer learning frameworks. Hierarchical federated learning architectures. Federated reinforcement learning. Federated graph neural networks. Distributed training of deep neural networks and foundation models. Edge intelligence and resource-constrained learning. Communication compression, quantization, and sparsification techniques. Green AI and energy-efficient distributed learning. Emerging research directions and open challenges in distributed and federated machine learning.

Learning Resources:

Text Books:
[T1] Qiang Yang, Yang Liu, Yong Cheng, Yan Kang, Tianjian Chen, Han Yu, "*Federated Learning*".
[T2] Yuan Tang, "*Distributed Machine Learning Patterns*".

Reference Books:
[R1] Rajkumar Buyya, Anwesha Mukherjee, Sajal K Das, "*Federated Learning: Foundations and Applications*".
[R2] Yaochu Jin, Hangyu Zhu, Jinjin Xu, Yang Chen, "*Federated Learning: Fundamentals and Advances*".
[R3] Gauri Joshi, "*Optimization Algorithms for Distributed Machine Learning*".

Other Suggested Readings:

List of Experiments:

1. Study and implementation of distributed gradient descent using PyTorch Distributed.
2. Study and implementation of synchronous and asynchronous stochastic gradient descent.
3. Study and implementation of parameter server architecture for distributed training.
4. Study and implementation of Federated Averaging (FedAvg) algorithm.
5. Study and implementation of client selection strategies in federated learning.
6. Study and implementation of federated learning on non-IID datasets.
7. Study and implementation of differential privacy mechanisms in federated learning.
8. Study and implementation of secure aggregation protocols.
9. Study and implementation using TensorFlow Federated framework.
10. Study and implementation using Flower federated learning framework.

11. Performance evaluation of communication-efficient federated learning algorithms.
12. Comparative study of centralized, distributed, and federated machine learning approaches.
13. Implementation of decentralized learning algorithms.
14. Study and presentation of recent research papers in distributed and federated machine learning.

Course Title: Large Language Models

Course Code: ADLM 603

L-T-P: 3-0-0

Credits: 3

Pre-requisites: NIL

Course Outcomes:

CO-1	Explain the fundamental concepts, architectures, and training paradigms of Large Language Models, including transformers, tokenization, and foundation models.
CO-2	Analyze transformer-based architectures, pre-training methodologies, fine-tuning approaches, and alignment techniques employed in modern Large Language Models.
CO-3	Apply prompt engineering, embedding models, vector databases, and Retrieval-Augmented Generation (RAG) techniques to develop domain-specific LLM solutions.
CO-4	Design and develop LLM-powered applications and intelligent agents while addressing evaluation, deployment, security, ethical, and responsible AI considerations.

Course Articulation Matrix:

COs	**POs & PSOs**						
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PSO1**	**PSO2**
CO1	2	1	–	1	1	2	1
CO2	3	2	1	2	–	3	1
CO3	3	3	1	3	–	3	2
CO4	3	2	2	3	2	3	3

1 - Slightly; 2 - Moderately; 3 - Substantially

Syllabus:

Unit I: Foundations of Large Language Models
Introduction to Natural Language Processing, evolution of language models, statistical and neural language models, word embeddings, sequence modeling, attention mechanism, self-attention, transformer architecture, encoder-decoder models, tokenization techniques, positional encoding, foundation models, BERT, GPT, T5, and modern LLM families.

Unit II: Training and Fine-Tuning of LLMs
Pre-training objectives, self-supervised learning, transfer learning, scaling laws, instruction tuning, supervised fine-tuning, parameter-efficient fine-tuning (PEFT), LoRA, QLoRA, model

compression, quantization techniques, Reinforcement Learning from Human Feedback (RLHF), Direct Preference Optimization (DPO), model alignment, safety alignment, and post-training methods.

Unit III: Prompt Engineering and Retrieval-Augmented Generation
Prompt engineering principles, zero-shot, one-shot and few-shot prompting, chain-of-thought prompting, in-context learning, reasoning techniques, embeddings, semantic search, vector databases, retrieval systems, Retrieval-Augmented Generation (RAG), knowledge grounding, context engineering, hallucination mitigation, retrieval evaluation, and domain-specific knowledge integration.

Unit IV: LLM Agents and Application Development
LLM-based agents, tool use and function calling, agent architectures, planning and reasoning, autonomous workflows, multi-agent systems, conversational AI, question answering systems, code generation, multimodal LLMs, integration with external systems and APIs, enterprise applications, agent frameworks, and deployment architectures.

Unit V: LLM Evaluation, Deployment, and Responsible AI
Evaluation metrics for LLMs, benchmark datasets, hallucination detection, robustness testing, inference optimization, serving architectures, monitoring and observability, model deployment, edge deployment, security and privacy concerns, prompt injection attacks, bias, and fairness, explainability, governance, ethical considerations, and responsible AI practices.

Learning Resources:

<u>Text Books:</u>

[T1] Jay Alammar and Maarten Grootendorst, "*Hands-On Large Language Models*", O'Reilly Media, First Edition.
[T2] Lewis Tunstall, Leandro von Werra, and Thomas Wolf, "*Natural Language Processing with Transformers*", O'Reilly Media, First Edition.

<u>Reference Books:</u>

[R1] Sunny Chopra and Shubham Saboo, "*Generative AI Engineering with LLMs*", O'Reilly Media, First Edition.

Course Title: Bioinformatics

Course Code: ADLM 604

L-T-P: 3-0-0

Credits: 3

Pre-requisites: NIL

Course Outcomes:

CO-1	Understand fundamental concepts of bioinformatics, molecular biology, and biological databases to retrieve and analyze biological information.
CO-2	Analyze biological sequences using pairwise and multiple sequence alignment techniques and evaluate their statistical significance.
CO-3	Apply computational methods for database similarity searching, motif and domain prediction, gene prediction, and phylogenetic analysis.
CO-4	Understand functional genomics approaches and apply machine learning techniques for the analysis and interpretation of biological datasets.

Course Articulation Matrix:

COs	**POs & PSOs**						
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PSO1**	**PSO2**
CO1	2	1	-	2	1	2	1
CO2	3	2	-	2	-	3	1
CO3	3	3	-	2	-	3	2
CO4	3	3	2	3	2	3	3

1 - Slightly; 2 - Moderately; 3 - Substantially

Syllabus:

Unit I (6 Hours)
Introduction: Bioinformatics, goals, scope, applications and limitations, Basic cell architecture, The structure, content, and scale of deoxyribonucleic acid (DNA), Genes and proteins, Central dogma, Importance of proteins, Gene and cell regulation, Biological Databases, Information Retrieval from Biological Databases.

Unit II (8 Hours)
Sequence Alignment: Pair-wise sequence alignment, Sequence homology versus sequence similarity, Sequence similarity versus sequence identity, Methods for sequence alignment, Statistical significance of sequence alignment. Multiple Sequence Alignment: Scoring function, Exhaustive algorithms, Heuristic algorithms, practical Issues.

Unit III (14 Hours)
Database similarity searching: Basic Local Alignment Search Tool (BLAST), FASTA,

Comparison of FASTA and BLAST.
Protein motifs and domain prediction: Identification of motifs and domains in multiple sequence alignment, Motif and domain databases using regular expressions, Motif and Domain Databases Using statistical models Gene Prediction: Gene prediction in prokaryotes and in eukaryotes.
Phylogenetics: Terminology, Gene phylogeny versus species phylogeny, Phylogenetic tree construction, Distance – based methods, Character- based methods, Phylogenetic tree evaluation.

Unit IV (8 Hours)
Functional Genomics: Sequence-based approaches, Microarray-based approaches, Comparison of SAGE, and DNA microarrays Case studies for machine learning techniques-based analysis of biological datasets.

Learning Resources:

<u>Text Books:</u>
[T1] J. Xiong, "*Essential bioinformatics*", Cambridge University Press, 1st edition, 2006.
[T2] E. Keedwell, and A. Narayanan, "*Intelligent bioinformatics: The application of artificial intelligence techniques to bioinformatics problems*", John Wiley & Sons, 1st edition, 2005.
[T3] J.M. Claverie, and C. Notredame, "*Bioinformatics for dummies*", John Wiley & Sons, 2nd edition, 2007.

<u>Reference Books:</u>
[R1] S. Mitra, S. Datta, T. Perkins, and G. Michailidis, "*Introduction to machine learning and bioinformatics*", CRC Press, 1st edition, 2008.
[R2] Z.R. Yang, "*Machine learning approaches to bioinformatics*", World scientific, 1st edition, 2010.
[R3] Y.Q. Zhang, and J.C. Rajapakse, "*Machine learning in bioinformatic*", Wiley, 1st edition, 2009.

Course Title: Multimodal AI

Course Code: ADLM 605

L-T-P: 3-0-0

Credits: 3

Pre-requisites: NIL

Course Outcomes:

CO-1	Apply multimodal representation learning techniques and architectures to process and integrate heterogeneous data from multiple modalities.
CO-2	Analyze multimodal applications and generative models for tasks such as image captioning, visual question answering, multimodal dialogue, and content generation.
CO-3	Understand the design, capabilities, and limitations of foundational multimodal large language models for complex reasoning and decision-making tasks.
CO-4	Understand the emerging trends, and advanced applications of multimodal AI in solving real-world interdisciplinary problems.

Course Articulation Matrix:

COs	**POs & PSOs**						
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PSO1**	**PSO2**
CO1	2	1	-	2	1	2	1
CO2	3	2	-	2	-	3	1
CO3	3	3	-	2	-	3	2
CO4	3	3	2	3	2	3	3

1=addressed to small extent
2= addressed significantly
3= addressed strongly (major part of course)

Syllabus:

Unit I (10 Hours)
Introduction to Multimodal Artificial Intelligence, challenges in multimodal learning, multimodal data representations involving text, images, audio, video, and sensor data. Cross-modal representation learning, joint embedding models, multimodal fusion strategies, attention mechanisms, multimodal transformers, neural modular networks, and techniques for integrating heterogeneous data sources. Evaluation metrics and performance assessment of multimodal systems.

Unit II (10 Hours)
Principles and applications of multimodal learning in image captioning, video description. Introduction to multimodal generative modeling, variational autoencoders, generative

adversarial networks, diffusion models, text-to-image synthesis, image-to-text generation, text-to-video generation, and evaluation methodologies for generative multimodal systems.

Unit III (8 Hours)
Evolution of foundation models and multimodal large language models, vision-language pre-training techniques, multimodal instruction tuning, alignment strategies, and architectures of contemporary multimodal LLMs.

Unit IV (8 Hours)
Applications of multimodal AI in healthcare, robotics, intelligent transportation systems, scientific discovery, and smart environments. Emerging research directions, open challenges, and future trends in multimodal artificial intelligence.

Learning Resources:

<u>Text Books:</u>
[T1] Li Deng and Yang Liu, "*Deep Learning in Natural Language Processing*", Springer, 2018.
[T2] Ian Goodfellow, Yoshua Bengio, and Aaron Courville, "*Deep Learning*", MIT Press, 2016.

<u>Reference Books:</u>
[R1] Koehn P., "*Neural Machine Translation*", Cambridge University Press, 2020.

Course Title: Speech Recognition

Course Code: ADLM 606

L-T-P: 3-0-0

Credits: 3

Pre-requisites:

Course Outcomes:

CO-1	Analyze speech signals and extract relevant acoustic features for speech processing applications.
CO-2	Understand statistical and machine learning approaches used in speech recognition systems.
CO-3	Apply hidden Markov models, language models, and deep neural networks for automatic speech recognition.
CO-4	Design and implement speech recognition systems for real-world applications.
CO-5	Evaluate and investigate state-of-the-art speech recognition technologies and research developments.

Course Articulation Matrix:

COs	**POs & PSOs**						
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PSO1**	**PSO2**
CO1	3	2	1	2	1	3	2
CO2	3	3	2	2	1	3	2
CO3	3	3	2	3	1	3	3
CO4	3	3	2	3	2	3	3
CO5	2	3	3	3	3	1	3

1=addressed to small extent
2= addressed significantly
3= addressed strongly (major part of course)

Syllabus:

Unit I: Fundamentals of Speech Processing (8 Hours)
Introduction to speech and spoken language technologies. Speech production and perception mechanisms. Characteristics of speech signals. Time-domain and frequency-domain representation of speech. Sampling, quantization, and digital speech processing. Short-time analysis of speech signals. Fourier Transform and Short-Time Fourier Transform. Linear Predictive Coding (LPC). Cepstral analysis. Mel-frequency cepstral coefficients (MFCCs). Speech databases and annotation techniques.

Unit II: Classical Automatic Speech Recognition (7 Hours)

Architecture of automatic speech recognition systems. Feature extraction techniques. Dynamic Time Warping (DTW). Pattern classification methods. Hidden Markov Models (HMMs). Gaussian Mixture Models (GMMs). Acoustic modeling. Pronunciation lexicons. Statistical language models. N-gram language modeling. Decoding and search algorithms. Evaluation metrics and word error rate analysis

Unit III: Deep Learning for Speech Recognition (8 Hours)

Introduction to deep learning for speech processing. Feedforward neural networks. Convolutional neural networks for speech recognition. Recurrent neural networks and Long Short-Term Memory networks. Attention mechanisms. Sequence-to-sequence models. Connectionist Temporal Classification (CTC). Transformer-based speech recognition. Self-supervised speech representation learning. End-to-end speech recognition systems. Speech recognition toolkits and frameworks.

Unit IV: Advanced Topics and Applications (7 Hours)

Robust speech recognition in noisy environments. Speaker recognition and speaker verification. Multilingual and code-mixed speech recognition. Speech emotion recognition. Speech enhancement and speech separation. Spoken dialogue systems. Speech synthesis and text-to-speech systems. Large language models and speech technologies. Edge deployment of speech recognition systems. Emerging trends and current research challenges in speech technologies.

Learning Resources:

<u>Text Books:</u>

[T1] Daniel Jurafsky and James H. Martin, "*Speech and Language Processing*", Pearson, Third Edition.

[T2] Lawrence Rabiner and Biing-Hwang Juang, "*Fundamentals of Speech Recognition",* Pearson, 1993.

<u>Reference Books:</u>

[R1] Frederick Jelinek, "*Statistical Methods for Speech Recognition*", MIT Press, 1998.

[R2] Bernard Gold, Nelson Morgan, "*Speech and Audio Signal Processing*", Wiley, 2000.

Course Title: AI for Healthcare

Course Code: ADLM 607

L-T-P: 3-0-0

Credits: 3

Pre-requisites: Machine Learning, Deep Learning, Probability and Statistics, Python Programming, Basic Knowledge of Healthcare Systems.

Course Outcomes:

CO-1	Analyze heterogeneous healthcare datasets.
CO-2	Design machine learning models for clinical prediction tasks.
CO-3	Develop deep learning models for medical imaging and bio-signal analytics.
CO-4	Apply foundation models and generative AI in healthcare applications.

Course Articulation Matrix :

COs	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2
CO1	3	2	1	2	1	3	2
CO2	3	3	2	2	1	3	2
CO3	2	2	2	1	2	2	3
CO4	2	3	3	3	2	2	3

1 - Slightly; 2 - Moderately; 3 - Substantially

Syllabus:

Unit I: Foundations of AI in Healthcare (8Hours)
Introduction to Healthcare AI: - Evolution of AI in medicine, Precision medicine, Data-driven healthcare, Learning healthcare systems. Healthcare Data Ecosystem-Electronic Health Records (EHR), Electronic Medical Records (EMR), Clinical notes, medical images, Biosignals, Genomic data, Wearable sensor data. Healthcare Data Standards -HL7, FHIR, DICOM, SNOMED CT, ICD Coding Systems. Healthcare Analytics-Descriptive analytics, Predictive analytics, Prescriptive analytics.

Unit II: Foundation Models, Generative AI, and Future Healthcare (12Hours)
Large Language Models in Healthcare - Clinical language models, medical question answering, Clinical summarization, medical report generation. Foundation Models for Healthcare -Med-PaLM, BioGPT, ClinicalBERT, GatorTronMedSAM. Generative AI Applications -Synthetic patient data generation, medical image synthesis, Drug discovery support, Clinical trial

optimization. AI for Precision Medicine- Genomics and AI, Personalized treatment planning, Digital twins in healthcare Trustworthy Healthcare AI -AI ethics, Explainability, Bias mitigation, Privacy and security, Responsible AI. Regulatory and Legal Frameworks, Emerging Topics, Supervised Learning in Healthcare, Unsupervised Learning, Explainable AI, Clinical Decision Support Systems, Case Studies- Diabetes prediction, cardiovascular disease risk, Cancer prognosis.

Unit III: Deep Learning and Medical Intelligence (10Hours)
Medical Image Analysis -X-ray analysis, CT scan analysis, MRI analysis, Histopathology imaging. Deep Learning Architectures-CNN, U-Net, ResNet, Vision Transformers. Biosignal Analytics - ECG, EEG, EMG, PPG. Multimodal Healthcare Learning- Clinical text + imaging, Imaging + genomics, EHR + sensor fusion. Federated Learning-Distributed healthcare AI, Privacy-preserving learning, Case Studies-Diabetic retinopathy detection, Brain tumour classification, Arrhythmia detection.

Unit IV: Foundation Models, Generative AI, and Future Healthcare (10Hours)
Large Language Models in Healthcare - Clinical language models, medical question answering, Clinical summarization, medical report generation. Foundation Models for Healthcare - Med-PaLM, BioGPT, ClinicalBERT, GatorTron, MedSAM. Generative AI Applications -Synthetic patient data generation, medical image synthesis, Drug discovery support, Clinical trial optimization.AI for Precision Medicine -Genomics and AI, Personalized treatment planning, Digital twins in healthcare. Trustworthy Healthcare AI, Regulatory and Legal Frameworks, Emerging Topics -Agentic Healthcare Systems Autonomous Clinical Assistants AI-powered Smart Hospitals.

Learning Resources:

Text Books:

[T1] Adam Bohr & Kaveh Memarzadeh, *"Artificial Intelligence in Healthcare"*, Academic Press, Elsevier.1st Edition, 2020.
[T2] Topol E., *"Deep Medicine: How Artificial Intelligence Can Make Healthcare Human Again"*, Basic Books, Hachette Book Group, 1st Edition, 2019.
[T3] Eric Topol, "*Deep Medicine*", 2019.
[T4] Robert Hoyt, Robert Muenchen, "*Introduction to Biomedical Data Science*", 2023.

Reference Books:

[R1] Topol, E. (2024), "*AI in Medicine: Current Trends and Future Directions*".
[R2] Nature Medicine Special Issue on Foundation Models in Healthcare (2024–2025).
[R3] WHO Guidance on Ethics and Governance of AI for Health.
[R4] Recent IEEE Transactions on Medical Imaging publications.
[R5] Recent ACM Computing Surveys on Healthcare AI.
[R6] NeurIPS, ICML, ICLR, AAAI and MICCAI proceedings on Medical AI.
[R7] Google DeepMind and Microsoft Research publications on Clinical Foundation Models.

Course Title: Prompt Engineering

Course Code: ADLM 608

L-T-P: 3-0-0

Credits: 3

Pre-requisites: Artificial Intelligence, Machine Learning, Natural Language Processing, Python Programming.

Course Outcomes:

CO-1	Design prompts for various AI applications.
CO-2	Apply advanced prompting techniques for reasoning and problem solving.
CO-3	Evaluate prompt quality using quantitative and qualitative metrics.
CO-4	Develop automated prompt optimization frameworks.

Course Articulation Matrix :

COs	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2
CO1	3	2	1	2	1	3	2
CO2	3	3	2	2	1	3	2
CO3	2	2	2	1	2	2	3
CO4	2	3	3	3	2	2	3

1 - Slightly; 2 - Moderately; 3 - Substantially

Syllabus:

UNIT I: Foundations of Prompt Engineering (07 Hours)
Introduction-Evolution of Prompt Engineering, Prompt Engineering vs Fine-Tuning, Prompting as In-Context Learning, Human-AI Interaction Paradigms. Prompt Fundamentals - Prompt Anatomy, Instructions, Context, Constraints, Output Specifications. Prompt Types-Direct Prompting, Instruction Prompting, Completion Prompting, Conversational Prompting. In-Context Learning-Zero-Shot Learning, One-Shot Learning, Demonstration-Based Learning. Prompt Design Principles -Clarity, Specificity, Modularity, Reusability, Scalability.

UNIT II: Advanced Prompting Techniques (10 Hours)
Reasoning-Oriented Prompting -Chain-of-Thought Prompting, Self-Consistency Prompting, Least-to-Most Prompting, Stepwise Reasoning. Structured Prompting-Role-Based Prompting, Persona Prompting, XML Prompting, JSON Prompting, Schema-Constrained Prompting. Decomposition Techniques-Problem Decomposition, Task Decomposition, Multi-Step Prompting, Recursive Prompting. Verification-Oriented Prompting- Self-Verification,

Reflection Prompting, Critique-and-Revise Prompting, Iterative Refinement. Domain-Specific Prompt Design -Scientific Research, Education, Healthcare, Software Development

UNIT III: Prompt Optimization and Evaluation (10 Hours)
Prompt Evaluation Frameworks-Accuracy Assessment, Robustness Evaluation, Hallucination Measurement, Consistency Analysis. Human-Centered Evaluation-User Satisfaction Metrics, Explainability Metrics, Usability Assessment. Automated Prompt Engineering- Search-Based Optimization, Evolutionary Prompt Optimization, Gradient-Based Prompt Tuning, Reinforcement Learning for Prompt Optimization. Benchmarking-Prompt Benchmarks, Comparative Evaluation, Statistical Analysis. Prompt Lifecycle Management-Prompt Versioning, Prompt Documentation, Prompt Testing Pipelines.

UNIT IV: Multimodal Prompting, Security and Emerging Research (10 Hours)
Multimodal Prompt Engineering -Text-to-Text Prompting, Text-to-Image Prompting, Vision-Language Prompting, Document Prompting. Prompt Security-Prompt Injection Attacks, Jailbreaking Techniques, Adversarial Prompting, Defense Mechanisms. Responsible Prompt Engineering -Bias Mitigation, Fairness, Transparency, Accountability. Emerging Research Topics-Automatic Prompt Engineering (APE), Meta-Prompting, Prompt Programming. Research Case Studies Prompt Engineering in Education, Research Workflow, Software Engineering, Healthcare

Learning Resources:

Text Books:

[T1] Schulhoff, S. et al. "*The Prompt Report: A Systematic Survey of Prompting Techniques*", Technical Monograph, Version 2025.
[T2] White J., Hays S., Fu Q., Spencer-Smith, J., and Schmidt, D.C., "*Prompt Engineering for Generative AI*", O'Reilly Media, 1st Edition, 2024.

Reference Books:

[R1] Mollick, Ethan, "*Co-Intelligence: Living and Working with AI*", Portfolio, Penguin Random House, 1st Edition, 2024.
[R2] Russell, Stuart and Norvig, Peter, "*Artificial Intelligence: A Modern Approach*", Pearson, 4th Edition, 2021.
[R3] Murphy, Kevin P., "*Probabilistic Machine Learning: Advanced Topics*" MIT Press, 2023.

Other Suggested Readings:

1. A Comprehensive Taxonomy of Prompt Engineering Techniques for Large Language Models, 2026.
2. Prompt Engineering and In-Context Learning: A Comprehensive Survey of Techniques, Applications and Future Directions, 2025.
3. A Survey of Automatic Prompt Engineering: An Optimization Perspective, 2025.
4. Prompt Engineering in Higher Education: A Systematic Review to Help Inform Curricula, 2025.
5. Beyond Miracle Prompts: Prompt Engineering as Academic Competency, 2026.

Course Title: Robotics and Automation

Course Code: ADBM 609

L-T-P: 3-0-2

Credits: 4

Pre-requisites:

Course Outcomes:

CO-1	To understand robotics fundamentals, robot architectures, workspaces, and subsystems.
CO-2	To analyze robotic sensors, perception techniques, and computer vision methods.
CO-3	To apply kinematics and dynamics for robot motion, navigation, and path planning.
CO-4	To understand intelligent automation techniques and AI applications in robotics.
CO-5	To design and evaluate autonomous robotic and automation systems for real-world applications.

Course Articulation Matrix:

COs	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2
CO1	2	1		2		2	
CO2	2	2		2	1	2	1
CO3	3	2		3	1	3	1
CO4	2	3		3	2	2	2
CO5	3	3	1	3	2	3	3

1=addressed to small extent
2= addressed significantly
3= addressed strongly (major part of course)

Syllabus:

Unit I: Introduction to Robotics (6 Hours)
Types of robots, Degrees of freedom of robots, Robot configurations and concept of workspace, Overview of robot subsystems, Mechanisms and transmission, End effectors and Different types of grippers, vacuum, and other methods of gripping. Applications of robots, specifications of different industrial robots.

Unit II: Drive systems and Sensors (6 Hours)
Drive system- hydraulic, pneumatic, and electric systems. Sensors in robot – Touch sensors, Tactile sensor, Proximity and range sensors, Robotic vision sensor, Force sensor, Light sensors, Pressure sensors.

Unit III: Kinematics and Dynamics of Robots (9 Hours)
2D, 3D Transformation, Scaling, Rotation, Translation, Homogeneous coordinates, multiple

transformation, Simple problems. Matrix representation, Forward and Reverse Kinematics of Three Degree of Freedom, Homogeneous Transformations, Inverse kinematics of Robot, Robot Arm dynamics, D-H representation of robots, Basics of Trajectory Planning.

Unit IV: Automation (6 Hours)
Automation concepts, Product cycle, Manufacturing functions, Types and degree of automation, Technical, Economic and Human factors in automation. Intelligent Automation (IA), Evolution of commercial AI tools, Unit Continuous Optimization Manufacturing (UCOM), Sensing and reasoning, Neural networks, Fuzzy logic systems, Genetic algorithms, and Hybrid intelligent systems for automation.

Unit V: Autonomous Systems and Applications (9 Hours)
Implementation of autonomous navigation and path planning algorithms, Localization and Mapping concepts, Real-time robotic decision making, Robot Operating System (ROS) overview, Applications of autonomous robots in manufacturing, healthcare, transportation and smart environments, Design considerations for automation, Cost-benefit analysis, Case studies and project-based applications.

Learning Resources:

Text Books:

[T1] John J. Craig, *"Introduction to Robotics: Mechanics and Control*", Pearson, 4th Edition.
[T2] Peter Corke, *"Robotics, Vision and Control: Fundamental Algorithms in MATLAB® and Python"*, Springer, Third Edition.

Reference Books:

[R1] Kevin M. Lynch, Frank C. Park, *"Modern Robotics: Mechanics, Planning, and Control"*, Cambridge University Press, First Edition.
[R2] Stuart Russell, Peter Norvig, *"Artificial Intelligence: A Modern Approach"*, Pearson, Fourth Edition.

List of Experiments

1. Study and implementation of Forward and Inverse Kinematics of a Robotic Manipulator.
2. Study and simulation of Denavit-Hartenberg (D-H) Parameter-based Robot Modeling.
3. Study and implementation of Robot Path Planning using Dijkstra's and An Algorithms.
4. Study and implementation of Autonomous Navigation using Rapidly exploring Random Trees (RRT).
5. Study and simulation of Obstacle Detection and Avoidance Techniques for Mobile Robots.
6. Study and implementation of Simultaneous Localization and Mapping (SLAM) using Robot Simulation Tools.
7. Study and implementation of Visual Odometry and Pose Estimation using Camera Data.
8. Study and implementation of Object Detection and Tracking for Robotic Vision Applications.
9. Study and implementation of Fuzzy Logic and Neural Network based Intelligent Robot Control Systems.
10. Study and simulation of Autonomous Robot Navigation and Multi-Robot Coordination using ROS/Gazebo.

Course Title: Generative AI

Course Code: ADLM 610

L-T-P: 3-0-0

Credits: 3

Pre-requisites:

Course Outcomes:

CO-1	Understand the fundamental concepts and real-world applications of Artificial Intelligence and Generative Al tools and platforms.
CO-2	Apply prompt engineering techniques to design optimized and context-aware prompts for generative tasks in various Al platforms.
CO-3	Analyze the effectiveness of generative Al in domains such as education, office automation, marketing, and healthcare through specific use cases.
CO-4	Evaluate and compare various generative Al tools for content generation, code assistance, media creation, and automation.
CO-5	Design Al-based workflows for real-world applications using tools like ChatGPT, GitHub Copilot, Jasper, and others.

Course Articulation Matrix:

COs	PO-1	PO-2	PO-3	PO-4	PO-5	PSO-1	PSO-2
CO-1	1	1	2	1	1	2	2
CO-2	1	2	2	1	1	2	2
CO-3	2	1	1	1	1	2	2
CO-4	2	1	1	1	2	2	2
CO-5	1	1	1	2	2	2	2

1 - Slightly; 2 - Moderately; 3 - Substantially

Syllabus:

Module-1
Foundations and Fundamentals of Generative AI: Introduction to Artificial Intelligence: Definitions, Foundations, Scope, and Applications, Implications of Al across industries, Fundamentals of Generative Al: Concept, Evolution, Capabilities, Applications of Generative Al in various domains, Introduction to Generative Al platforms: Chat GPT (OpenAI), Gemini (Google), GitHub Copilot (Microsoft).

Module-II
Prompt Engineering and Prompt Optimization Techniques: Basics of Prompt Engineering, Types of Prompts: Declarative, Interrogative, Instructional, Advanced Prompt Engineering:

Chain-of-Thought Prompting, Zero-shot and Few-shot Prompting, Prompt Optimization Techniques, Use Cases and Practical Examples in Coding, Writing, and Customer Interaction.

Module-III
Generative AI Applications in Workplace and Education: Al in Office Productivity: Email Writing, Report Drafting, Minutes of Meeting (MoM), Proposal Writing, Summarization, and Task Automation, Generative Al in Education: Al for Content Creation, Summarization, and Explanation, Curriculum Design using Al Tools, Al in Assessment, Feedback, and Visualization (e.g., Mindmaps, Infographics).

Module-IV
Domain-Specific Applications of Generative AI: Generative Al in Digital Marketing: Content Strategy for YouTube, Facebook, Instagram, X (Twitter), LinkedIn and Al in Professional Branding, Email Marketing Automation and Alin E-Commerce; Generative Al in Healthcare and Agriculture: Diagnosis Assistance, Patient Data Summarisation, Crop Monitoring, Yield Prediction using Al.

Module-V
Creative and Technical Aspects of Generative AI: Al-Assisted Programming (e.g., Code Generation, Auto Debugging using Copilot, GPT-4), Generative Al in Creative Domains: Image & Design Generation (DALL E, Midjourney), Music Composition with Al Tools, Al in Gaming & Animation K e.g., Scenario Generation, NPC Scripting), Customer Interaction Automation: Chatbots, Virtual Assistants, and Voice Al; Exploration of Popular Al Tools & Platforms: Runway ML, Jasper, Synthesia, Lumens, Bard, etc.

Learning Resources:

Text Books:
[T1] Emerson, J., "*Ripples of generative AI: Navigating Al's impact on society*", Al Press (2023).

Reference Books:
[R1] Pandit R., *"Generative Al (Hardcover)"*, Notion Press (2024).
[R2] OpenAI. (n.d.). OpenAl, from https://openai.com/
[R3] Google. (nd.). Gemini, from https://gemini.google.com/

Other Suggested Readings:

Microsoft. (n. d.). Copilot, from
https://copilot.microsoft.com/chats/VkLP7TbPcVtu4dD.jpDV Tv
SWAYAM/NPTEL/MOOCs Certification:

1. https://onlinecourses.swayam2.ac.in/mb25_mg135/preview

2. https://onlinecourses, swayam2.ac.in/nou25 ge85/preview.

Course Title: Quantum Computing

Course Code: ADBM 611

L-T-P: 3-0-2

Credits: 4

Pre-requisites: Linear Algebra, Probability and Statistics, Design and Analysis of Algorithms

Course Outcomes:

CO1	Explain the fundamental principles of quantum mechanics, quantum information, and quantum computation.
CO2	Analyze quantum states, quantum gates, quantum circuits, and quantum communication protocols.
CO3	Apply quantum algorithms to solve computational problems and evaluate their complexity advantages over classical algorithms.
CO4	Examine quantum cryptographic protocols, quantum noise models, and quantum error correction techniques.
CO5	Design and implement quantum programs using Qiskit and assess emerging applications of quantum computing.

Course Articulation Matrix:

COs	PO-1	PO-2	PO-3	PO-4	PO-5	PSO-1	PSO-2
CO-1	3	2	1	1	1	2	1
CO-2	3	3	2	2	2	3	2
CO-3	3	3	3	2	3	3	3
CO-4	2	3	2	3	2	3	3
CO-5	2	2	3	3	3	3	3

1=addressed to small extent
2= addressed significantly
3= addressed strongly (major part of course)

Syllabus:

Module-I: Mathematical Foundations and Quantum Mechanics Basics (8 Hours)
Introduction to Quantum Computing, Limitations of Classical Computing, Quantum Advantage, Complex Numbers, Vector Spaces, Linear Operators, Tensor Products, Eigenvalues and Eigenvectors, Dirac Notation, Hilbert Spaces, Quantum States, Superposition Principle, Quantum Measurement Postulates, Bloch Sphere Representation.

Module-II: Quantum Information and Quantum Circuits (8 Hours)

Qubits and Multi-Qubit Systems, Entanglement, Bell States, Density Matrix Formalism, Quantum Gates: Pauli Gates, Hadamard Gate, Phase Gates, CNOT Gate, Toffoli Gate, Fredkin Gate, Reversible Computation, Quantum Circuit Model, Universal Gate Sets, No-Cloning Theorem, Quantum Teleportation, Superdense Coding.

Module-III: Quantum Algorithms and Computational Complexity (8 Hours)
Deutsch Algorithm, Deutsch-Jozsa Algorithm, Bernstein-Vazirani Algorithm, Simon's Algorithm, Quantum Fourier Transform (QFT), Quantum Phase Estimation, Grover's Search Algorithm, Shor's Factoring Algorithm, Complexity Classes: P, NP, BPP, BQP, Quantum Computational Advantage.

Module-IV: Quantum Communication, Cryptography and Error Correction (8 Hours)
Quantum Communication Fundamentals, Quantum Channels, BB84 Protocol, E91 Protocol, Security of Quantum Key Distribution, Quantum Threats to Classical Cryptography, Introduction to Post-Quantum Cryptography, Quantum Noise Models, Decoherence, Bit Flip Code, Phase Flip Code, Shor Code, Surface Codes, Fault-Tolerant Quantum Computing.

Module-V: Quantum Programming and Emerging Applications (8 Hours)
IBM Quantum Platform, Qiskit Framework, Quantum Circuit Simulation, Execution on Quantum Hardware, Variational Quantum Eigensolver (VQE), Quantum Approximate Optimization Algorithm (QAOA), Quantum Machine Learning Fundamentals, Quantum Image Processing, Quantum Optimization, Quantum Internet, National Quantum Mission, Current Research Trends and Future Directions.

Learning Resources:

Text Books:

[T1] Michael A. Nielsen and Isaac L. Chuang, "*Quantum Computation and Quantum Information*", 10th Anniversary Edition, Cambridge University Press, 2010**.**

[T2] Jack D. Hidary, "*Quantum Computing: An Applied Approach*", 2nd Edition, Springer, 2021**.**

Reference Books:

[R1] Chris Bernhardt, "*Quantum Computing for Everyone*", MIT Press, 2019**.**

[R2] Maria Schuld and Francesco Petruccione, "*Machine Learning with Quantum Computers*", 2nd Edition, Springer, 2021**.**

[R3] John Watrous, "*The Theory of Quantum Information*", Cambridge University Press, 2018**.**

[R4] Ronald de Wolf, "*Quantum Computing: Lecture Notes*", QuSoft, 2023**.**

[R5] Hiu Yung Wong, "*Quantum Computing Architecture and Hardware for Engineers: Step by Step*", 2024**.**

[R6] Olivier Ezratty, "*Understanding Quantum Technologies 2024*", 2024 Edition**.**

Lab Experiments:

1. Introduction to IBM Quantum Experience and Qiskit.
2. Creation and Measurement of Single-Qubit States.
3. Implementation of Basic Quantum Gates.
4. Bell State Generation and Entanglement Analysis.
5. Quantum Teleportation Protocol.
6. Superdense Coding Implementation.
7. Deutsch-Jozsa Algorithm.
8. Bernstein-Vazirani Algorithm.

9. Grover's Search Algorithm.
10. Quantum Fourier Transform.
11. BB84 Quantum Key Distribution Protocol.
12. Quantum Error Correction Codes.
13. Variational Quantum Eigensolver (VQE).
14. Quantum Approximate Optimization Algorithm (QAOA).
15. Mini Project using Qiskit.

Course Title: Advanced Digital Image Processing

Course Code: ADBM 612

L-T-P: 3-0-2

Credits: 4

Pre-requisites:

Course Outcomes:

CO-1	Explain advanced image processing concepts, image transforms, and mathematical techniques used for image representation and analysis. **(L2)**
CO-2	Analyze image enhancement, restoration, and segmentation techniques for improving image quality and extracting meaningful information. **(L4)**
CO-3	Apply feature extraction, image registration, and image representation methods to solve real-world image processing problems. **(L3)**
CO-4	Design and develop advanced image processing solutions for applications such as medical imaging, remote sensing, computer vision, and intelligent image analysis systems. **(L6)**

Course Articulation Matrix:

COs / POs & PSOs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PSO1**	**PSO2**
CO1	2	1	–	1	1	2	1
CO2	3	2	1	2	–	3	1
CO3	3	3	1	3	–	3	2
CO4	3	2	2	3	1	3	3

1=addressed to small extent
2= addressed significantly
3= addressed strongly (major part of course)

Syllabus:

Unit I: Fundamentals and Image Transform Techniques
Review of digital image fundamentals, image sampling and quantization, image models, image transforms and frequency domain analysis, Discrete Fourier Transform (DFT), Fast Fourier Transform (FFT), Discrete Cosine Transform (DCT), Walsh-Hadamard Transform, Karhunen-Loeve Transform (KLT), Wavelet Transform, multiresolution analysis, image representation in transform domains.

Unit II: Image Enhancement and Restoration

Spatial and frequency domain enhancement techniques, histogram processing, contrast enhancement, image smoothing and sharpening, noise models, image degradation models, inverse filtering, Wiener filtering, constrained least squares restoration, blind deconvolution, regularization techniques, and image quality assessment measures.

Unit III: Image Segmentation and Feature Extraction
Thresholding methods, edge-based segmentation, region-based segmentation, clustering techniques, watershed transformation, active contour models, feature detection, corner detection, texture analysis, shape descriptors, feature extraction using statistical, structural, and transform-based methods.

Unit IV: Image Registration and Image Representation
Image registration fundamentals, geometric transformations, feature-based registration, intensity-based registration, multimodal image registration, image matching techniques, image representation methods, boundary representation, region representation, chain codes, shape analysis, object recognition fundamentals.

Unit V: Advanced Topics and Applications
Image compression techniques, predictive and transform coding, vector quantization, image fusion, color image processing, hyperspectral image analysis, medical image processing, remote sensing image analysis, computer vision applications, deep learning for image processing, emerging trends in image analysis.

Learning Resources:

<u>**Text Books:**</u>

[T1] Rafael C. Gonzalez and Richard E. Woods, *"Digital Image Processing"*, Pearson Education, Fourth Edition.
[T2] Anil K. Jain, *"Fundamentals of Digital Image Processing"*, Pearson Education, Second Edition.

<u>**Reference Books:**</u>

[R1] R. J. Schalkoff, *"Digital Image Processing and Computer Vision"*, Wiley, First Edition.

List of Experiments:
1. Image Enhancement Using Histogram Equalization and Contrast Limited Adaptive Histogram Equalization (CLAHE)
2. Image Denoising Using Spatial Domain and Frequency Domain Filtering Techniques
3. Image Restoration Using Inverse Filtering and Wiener Filtering Techniques
4. Image Compression Using Discrete Wavelet Transform (DWT)
5. Edge Detection and Boundary Extraction Using Gradient and Laplacian Operators
6. Morphological Image Processing for Object Extraction and Shape Analysis
7. Image Segmentation Using Global, Adaptive, and Otsu Thresholding Methods
8. Region-Based Image Segmentation Using Region Growing and Watershed Algorithms
9. Clustering-Based Image Segmentation Using K-Means and Fuzzy C-Means Algorithms
10. Texture Feature Extraction and Analysis Using Gray Level Co-occurrence Matrix (GLCM)

Course Title: Motion Analytics

Course Code: ADBM 613

L-T-P: 3-0-0

Credits: 3

Pre-requisites:

Course Outcomes:

CO-1	Explain fundamentals of motion, video data formation, spatio-temporal representations and motion cues used in analytics
CO-2	Apply motion estimation, background modelling, optical flow and object tracking methods for dynamic scene analysis.
CO-3	Analyse trajectories, human pose, activities, and events using machine learning and deep learning models.
CO-4	Design and evaluate motion analytics pipelines for surveillance, sports, healthcare, traffic, and industrial AI applications.

Course Articulation Matrix:

COs	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2
CO1	3	2	1		1	2	1
CO2	3	3	2		2	3	2
CO3	3	3	3	2	2	3	3
CO4	3	3	3	2	3	3	3

1=addressed to small extent
2= addressed significantly
3= addressed strongly (major part of course)

Syllabus:

Unit I: Foundations of Motion Analytics (7 Hours)
Introduction to motion analytics; video data acquisition and representation; frames, pixels, colour spaces and temporal sampling; camera models and calibration basics; motion cues in image sequences; spatio-temporal filtering; feature points, corners, edges, and descriptors; datasets and annotation formats for motion-based AI applications.

Unit II: Motion Detection and Estimation (7 Hours)
Frame differencing, temporal differencing, and background subtraction; Gaussian mixture models and adaptive background modelling; foreground segmentation and morphological processing; image pyramids and multi-scale motion; optical flow fundamentals; Lucas-Kanade and Horn-Schunck methods; dense and sparse flow; motion vectors, block matching and performance measures.

Unit III: Object Tracking and Trajectory Analytics (8 Hours)
Single object and multiple object tracking; Kalman filter and particle filter; Mean Shift/CamShift; correlation filters; detection-based tracking; SORT/DeepSORT concepts; data association, occlusion handling and re-identification; trajectory smoothing, clustering and similarity measures; motion patterns, speed, direction, dwell time and path analytics.

Unit IV: Human Motion, Pose and Activity Analysis (7 Hours)
Human detection and pose estimation; skeleton and key-point based motion representation; gait and gesture analysis; action and activity recognition; handcrafted features such as HOG, HOF and motion history images; machine learning classifiers for motion data; introduction to CNN, 3D CNN, LSTM/GRU and transformer-based video models.

Unit V: Applications, Evaluation and Deployment (7 Hours)
Motion anomaly and event detection; crowd, traffic, sports, healthcare, security, and industrial monitoring applications; real-time video analytics pipeline using OpenCV/deep learning frameworks; edge/cloud deployment issues; latency, throughput, and computational complexity; privacy, ethics, and bias; evaluation metrics including accuracy, precision, recall, F1-score, MOTA/MOTP, ID-switches and mAP.

Learning Resources:

<u>**Text Books:**</u>

[T1] Richard Szeliski, *"Computer Vision: Algorithms and Applications",* Springer Nature, Second Edition, 2022.
[T2] David A. Forsyth and Jean Ponce, *"Computer Vision: A Modern Approach"*, Pearson Education / Prentice Hall, Second Edition, 2012.

<u>**Reference Books:**</u>

[R1] Richard Hartley and Andrew Zisserman, "*Multiple View Geometry in Computer Vision*", Cambridge University Press, Second Edition, 2004.

Course Title: Game Theory

Course Code: ADLM 614

L-T-P: 3-1-0

Credits: 4

Pre-requisites:

Course Outcomes:

CO-1	Understanding of the basic concept of game theory.
CO-2	Understanding of Nash Equilibrium.
CO-3	Ability to design the Prisoner’s Dilemma.
CO-4	Ability to design Pricing and Resource Allocation.

Course Articulation Matrix:

COs	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2
CO1	2					2	
CO2	3	1	2			1	2
CO3	2	1	2	2	2	2	2
CO4	2	2	1	2	1	2	

1=addressed to small extent
2= addressed significantly
3= addressed strongly (major part of course)

Syllabus:

Unit 1
Introduction: What is game theory, the theory of rational choice, interacting decision makers

Unit 2
Nash Equilibrium: Theory Strategic games; the Prisoner's Dilemma, Bach or Stravinsky, Brass Paradox, Matching Pennies, the stag Hunt, Nash Equilibrium, Best response functions, Dominated actions, Equilibrium in single population.
Mixed Strategy Equilibrium Introduction, Mixed strategy Nash equilibrium, the formation of players' beliefs, finding all mixed strategy Nash equilibria, games in which each player has a continuum of actions.

Unit 3
Extensive Games with Perfect Information Introduction, Strategies and outcomes, Nash equilibrium, Subgame perfect equilibrium, Finding subgame perfect equilibria of finite horizon games, Stackelberg's model of duopoly.
Repeated Games: The Prisoner’s Dilemma, General Result

Unit 4
Bargaining as an extensive game, Illustration: trade in a market, Nash's axiomatic model, Relation between strategic and axiomatic models

Unit 5
Incentives and Pricing in Communications Networks: Large Networks, Pricing and Resource Allocation, Alternative Pricing, and Incentive Approaches.
Learning Resources:

<u>**Text Books:**</u>
[T1] Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay V. Vazirani., *"Algorithmic Game Theory",* Cambridge University Press, 2007.
[T2] Osborne, M. J., *"An introduction to game theory"*, Oxford Univ. Press.

<u>**Reference Books:**</u>
[R1] Robert Gibbons, "*Game Theory for Applied Economists"*, Princeton University Press, 1992.

List of Experiments:
1. Study and implementation of Forward and Inverse Kinematics of a Robotic Manipulator.
2. Study and simulation of Denavit-Hartenberg (D-H) Parameter-based Robot Modeling.
3. Study and implementation of Robot Path Planning using Dijkstra's and An Algorithms. *
4. Study and implementation of Autonomous Navigation using Rapidly exploring Random Trees (RRT).
5. Study and simulation of Obstacle Detection and Avoidance Techniques for Mobile Robots.
6. Study and implementation of Simultaneous Localization and Mapping (SLAM) using Robot Simulation Tools.
7. Study and implementation of Visual Odometry and Pose Estimation using Camera Data.
8. Study and implementation of Object Detection and Tracking for Robotic Vision Applications.
9. Study and implementation of Fuzzy Logic and Neural Network based Intelligent Robot Control Systems.
10. Study and simulation of Autonomous Robot Navigation and Multi-Robot Coordination using ROS/Gazebo.

Course Title: Reinforcement Learning and Applications

Course Code: ADLM 615

L-T-P: 3-0-0

Credits: 3

Pre-requisites:

Course Outcomes:

CO-1	Explain the fundamental concepts, mathematical foundations, and learning paradigms of reinforcement learning. **(L2)**
CO-2	Analyze value-based, policy-based, and model-based reinforcement learning algorithms for solving sequential decision-making problems. **(L4)**
CO-3	Apply classical and deep reinforcement learning techniques to develop intelligent agents for complex environments. **(L3)**
CO-4	Design and implement reinforcement learning solutions for real-world applications while evaluating their performance and scalability. **(L6)**

Course Articulation Matrix:

COs	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2
CO1	3	2	1	2	1	3	2
CO2	3	3	2	2	1	3	2
CO3	2	2	2	1	2	2	3
CO4	2	3	3	3	2	2	3

1=addressed to small extent
2= addressed significantly
3= addressed strongly (major part of course)

Syllabus:

Unit I: Fundamentals of Reinforcement Learning
Introduction to machine learning paradigms, supervised, unsupervised and reinforcement learning, reinforcement learning framework, agents and environments, rewards and returns, exploration and exploitation, Markov Decision Processes (MDP), Bellman equations, finite and infinite horizon problems, dynamic programming, policy evaluation and policy improvement.

Unit II: Model-Free Reinforcement Learning
Monte Carlo methods, Temporal Difference (TD) learning, SARSA algorithm, Q-Learning, Expected SARSA, eligibility traces, TD(λ), on-policy and off-policy learning, convergence properties, function approximation techniques, practical implementation issues.

Unit III: Deep Reinforcement Learning

Introduction to deep learning for RL, neural network function approximation, Deep Q-Networks (DQN), experience replay, target networks, Double DQN, Dueling DQN, Deep Deterministic Policy Gradient (DDPG), Trust Region Policy Optimization (TRPO), Proximal Policy Optimization (PPO), actor-critic methods.

Unit IV: Advanced Reinforcement Learning Techniques

Policy gradient methods, stochastic and deterministic policies, model-based reinforcement learning, multi-agent reinforcement learning, hierarchical reinforcement learning, inverse reinforcement learning, imitation learning, transfer learning in RL, safe reinforcement learning, exploration strategies.

Unit V: Reinforcement Learning Applications

Reinforcement learning in robotics, autonomous vehicles, game playing systems, recommender systems, finance and algorithmic trading, resource allocation, wireless networks, smart grids, healthcare applications, industrial automation, reinforcement learning for large-scale decision-making systems, ethical considerations and future trends.

Learning Resources:

<u>Text Books:</u>

[T1] Richard S. Sutton and Andrew G. Barto, *"Reinforcement Learning: An Introduction"*, MIT Press, Second Edition.

[T2] Maxim Lapan, *"Deep Reinforcement Learning Hands-On"*, Packt Publishing, Third Edition.

<u>Reference Books:</u>

[R1] Csaba Szepesvári, *"Algorithms for Reinforcement Learning"*, Morgan & Claypool Publishers, First Edition.

Course Title: Information Retrieval

Course Code: ADLM 616

L-T-P: 3-0-0

Credits: 3

Pre-requisites: Database management system, Data structures

Course Outcomes:

CO-1	Analyze and implement indexing structures, query processing techniques, and efficient retrieval mechanisms.
CO-2	Apply retrieval models and evaluate the effectiveness of information retrieval systems using standard performance measures.
CO-3	Design document classification and clustering solutions using machine learning and statistical approaches.
CO-4	Develop and analyze web-scale retrieval systems using crawling, indexing, ranking, and link analysis techniques.

Course Articulation Matrix:

COs	PO-1	PO-2	PO-3	PO-4	PO-5	PSO-1	PSO-2
CO-1	3	2	1	3	1	3	1
CO-2	3	3	1	3	1	3	1
CO-3	3	3	1	3	1	3	2
CO-4	2	3	2	3	2	2	3

1 - Slightly; 2 - Moderately; 3 - Substantially

Syllabus:

Module-1

Indexing and Retrieval Foundations

Boolean Retrieval Model, Inverted Index Construction, Processing Boolean Queries, Extended Boolean Model, Term Vocabulary and Postings Lists, Positional Indexes, Phrase Queries, Dictionaries and Tolerant Retrieval, Wildcard Queries, Spelling Correction, Blocked Sort-Based Indexing, SPIMI, Distributed Indexing and Dynamic Indexing.

Module-2

Retrieval Models and Evaluation

Index Compression, Dictionary Compression, Postings Compression, Vector Space Model, Term Weighting, TF-IDF, Efficient Scoring and Ranking, Query Processing, Retrieval System Architecture, Precision, Recall, F-Measure, Mean Average Precision, NDCG, Relevance Assessment.

Module-3

Advanced Retrieval Models

Relevance Feedback, Pseudo-Relevance Feedback, Query Expansion, XML Retrieval, XML Retrieval Evaluation, Probabilistic Information Retrieval, Probability Ranking Principle, Binary Independence Model, Language Models for Information Retrieval, Query Likelihood Model.

Module-4

Text Mining and Web Retrieval

Text Classification, Naive Bayes Classification, Rocchio Classification, k-Nearest Neighbour Classification, Document Clustering, k-Means Clustering, Hierarchical Clustering, Latent Semantic Indexing, Singular Value Decomposition, Web Search Architecture, Web Crawling, Distributed Indexing, PageRank, HITS Algorithm, Neural Information Retrieval and Semantic Search.

Learning Resources:

Text Books:

[T1] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze, "*Introduction to Information Retrieval*", Cambridge University Press, 2008.

[T2] David A. Grossman and Ophir Frieder, "*Information Retrieval: Algorithms and Heuristics*", Springer, 2nd Edition, 2004.

Reference Books:

[R1] Gerald J. Kowalski and Mark T. Maybury, "*Information Storage and Retrieval Systems: Theory and Implementation*", Springer, 2nd Edition, 2005.

[R2] Ricardo Baeza-Yates and Berthier Ribeiro-Neto, "*Modern Information Retrieval*", Addison-Wesley, 2nd Edition, 2011.

Course Title: Internet of Things

Course Code: ADBM 617

L-T-P: 3-0-2

Credits: 4

Pre-requisites:

Course Outcomes:

CO-1	Explain the architecture, components, and enabling technologies of IoT systems.
CO-2	Design IoT networks using suitable communication technologies and protocols.
CO-3	Develop machine learning and deep learning models for robotic perception and autonomous operation.
CO-4	Analyze security, privacy, and trust challenges in IoT environments and propose appropriate solutions.
CO-5	Design and implement intelligent IoT applications for industrial and societal needs.

Course Articulation Matrix:

COs	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2
CO1	1	1	1	2	2	2	2
CO2	2	1	2	2	2	2	3
CO3	3	3	2	1	2	3	2
CO4	3	2	3	2	2	3	3

1=addressed to small extent
2= addressed significantly
3= addressed strongly (major part of course)

Syllabus:

Unit I: Introduction to Internet of Things (7 Hours)
Fundamentals of IoT, Evolution and Characteristics of IoT, IoT Architecture and Design Principles, IoT Reference Models, Physical and Logical Design of IoT Systems, Sensors, Actuators, and Embedded Systems, IoT Ecosystem and Enabling Technologies, Applications of IoT.

Unit II: IoT Devices, Connectivity and Communication Protocols (7 Hours)
IoT Hardware Platforms, Arduino, Raspberry Pi, ESP32, Wireless Communication Technologies (Wi-Fi, Bluetooth Low Energy (BLE), ZigBee, LoRaWAN, NB-IoT), IPv6 for IoT, 6LoWPAN, MQTT Protocol, CoAP Protocol, AMQP and DDS, Edge, and Fog Computing.

Unit III: IoT Data Management and Cloud Integration (7 Hours)
IoT Data Acquisition and Processing, IoT Middleware, Cloud Computing for IoT, Data

Storage and Management, Stream Processing, Big Data Analytics in IoT, IoT Platforms, AWS IoT, Azure IoT Hub, Google Cloud IoT, Digital Twin Concepts.

Unit IV: IoT Security, Privacy and Trust Management (7 Hours)
Security Challenges in IoT, Authentication and Authorization, Access Control Mechanisms, Secure Communication Protocols, Cryptography for IoT, Blockchain-enabled IoT, Privacy Preservation Techniques, Trust Management in IoT, Security Standards and Best Practices

Unit V: Advanced IoT Applications and Emerging Trends (7 Hours)
Industrial IoT (IIoT), Smart Cities and Smart Governance, Smart Healthcare and Wearable Systems, Smart Agriculture, Intelligent Transportation Systems, IoT and Artificial Intelligence (AIoT), Edge AI for IoT, Digital Twins and Cyber-Physical Systems, Industry 4.0 and Smart Manufacturing, Recent Research Trends and Challenges.

Learning Resources:

<u>**Text Books:**</u>

[T1] Arshdeep Bahga and Vijay Madisetti, *"Internet of Things: A Hands-On Approach"*, Universities Press, 2025.
[T2] Rajkumar Buyya and Amir Vahid Dastjerdi, *"Internet of Things: Principles and Paradigms"*, Morgan Kaufmann (Elsevier). 2016.

<u>**Reference Books:**</u>

[R1] Maciej Kranz, *"Building the Internet of Things"*, Wiley, 2016.
[R2] Dieter Uckelmann, Mark Harrison, Florian Michahelles, *"Architecting the Internet of Things"*, Springer Berlin Heidelberg, 2016.
[R3] Adrian McEwen and Hakim Cassimally, *"Designing the Internet of Things"*, Wiley, 2016.

<u>**Other Suggested Readings:**</u>

<u>**List of Experiments:**</u>

1. Sensor interfacing using Arduino/ESP32.
2. IoT device programming and data acquisition.
3. MQTT-based communication between IoT devices.
4. Wireless sensor data transmission using Wi-Fi/BLE.
5. Cloud integration using AWS IoT or Azure IoT.
6. IoT dashboard development for real-time monitoring.
7. Edge computing implementation for IoT applications.
8. Secure IoT communication using encryption techniques.
9. Smart home automation system development.
10. IoT-based predictive monitoring application.
11. Mini Project on Industrial or Smart City IoT Solutions.

Course Title: Big Data Analytics

Course Code: ADBM 618

L-T-P: 3-0-0

Credits: 3

Pre-requisites:

Course Outcomes:

CO-1	Analyze the characteristics, challenges, and architectural components of Big Data systems, and evaluate various Big Data storage and processing frameworks.
CO-2	Design and implement distributed data processing solutions using Big Data technologies such as Hadoop, MapReduce, HDFS, Spark, and NoSQL databases.
CO-3	Apply advanced data preprocessing, mining, and machine learning techniques to extract actionable knowledge from large-scale structured and unstructured datasets.
CO-4	Develop scalable analytics models for real-time and batch data processing, and assess their performance in diverse application domains.

Course Articulation Matrix:

COs	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2
CO1	3	2	2	1		2	3
CO2	3	2	1	3		2	2
CO3	3	2	3	2	2	2	3
CO4	2	3	3	2	2	3	3

1=addressed to small extent
2= addressed significantly
3= addressed strongly (major part of course)

Syllabus:

Unit 1 (5 Hours)
Fundamentals of Business Analytics-Business Intelligence (BI), Business Intelligence vs. Business Analytics, BI Framework, BI Roles & Responsibilities, BI DW Best Practices, Popular BI Tools, BI Applications.

Unit 2 (8 Hours)
Big Data Analytics in Memory Analytics, In Database Processing, Symmetric Multiprocessor System (SMP), Massively Parallel Processing, Shared Nothing Architecture, Parallel and Distributed Systems, CAP Theorem.

Unit 3 (8 Hours)

Typical enterprise application architecture Visualizing Relationship in Data, Probability, Estimation, Outliers and Normal Distribution, Inference, Regression, Exploratory Data Analysis (EDA), Big Data, Scaling Problems, HDFS, Design Patterns, Cluster.

Unit 4 (8 Hours)
Data analytic life cycle, Advance analytic theory, and method: "Clustering", Advance analytic theory, and method: "Association,", Advance analytic theory and method: "Classification", Advance analytic theory, and method: "Regression".

Unit 5 (7 Hours)
Big Data Analytics Tool Hadoop, MongoDB, Cassandra, MapReduce, Hive, Pig, Oozie.

Learning Resources:

Text Books:
[T1] "*Data Science and Big Data Analytics*", Wiley, 1st Edition.
[T2] R.N. Prasad, Seema Acharya, *"Fundamentals of Business Analytics"*, Wiley, 2nd Edition.
[T3] Michael Minelli, Michele Chambers, Ambiga Dhiraj, *"Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses"*, Wiley, 1st Edition.

Reference Books:
[R1] Jeffery Stanton, *"An Introduction to Data Science"*, SAGE, 1st Edition.
[R2] Seema Acharya, Subhashini Chellapan, *"Big Data and Analytics"*, Wiley, 2nd Edition.

Course Title: Introduction to Cyber Security

Course Code: ADLM 619

L-T-P: 3-0-0

Credits: 3

Pre-requisites: NIL

Course Outcomes:

CO-1	Describe the concepts of cybercrime, information security, cyber laws, and intellectual property rights.
CO-2	Examine cyber threats, attack vectors, social engineering techniques, and cybercrime methodologies.
CO-3	Assess security challenges, vulnerabilities, and protection mechanisms in mobile and wireless environments.

Course Articulation Matrix:

COs	PO-1	PO-2	PO-3	PO-4	PO-5	PSO-1	PSO-2
CO-1	3	1	1	1	2	1	1
CO-2	3	2	1	2	1	3	2
CO-3	3	2	1	2	1	2	2

1 - Slightly; 2 - Moderately; 3 – Substantially

Syllabus:

Unit I: Cybercrime (7 Hours)

Definition, Characteristics, and Evolution, Origins of the Term Cybercrime, Information Security Fundamentals. Cybercriminals: Characteristics and Classifications, Global Perspective on Cybercrime, Cybercrime Trends and Challenges, Cyber Offences and Attack Lifecycle, Social Engineering Attacks, Cyber Stalking and Cyber Harassment, Botnets and Attack Vectors Need for Information Security Policies.

Unit II: Introduction to Indian Cyber Law (7 Hours)

Information Technology Act and Related Provisions, Digital Personal Data Protection Act (DPDP), 2023: Objectives and Scope, Privacy and Data Protection, Cyber Ethics and Regulatory Compliance, Intellectual Property Rights in Cyberspace, Patent Protection, Copyright Protection, Trademark Protection, Overview of Intellectual Property Related Legislation in India.

Unit III: Introduction to Mobile (8 Hours)

Wireless Computing, Growth and Trends in Mobile Computing, Mobile Payment and Credit Card Frauds, Security Challenges in Mobile Devices, Authentication and Access Control in Mobile Systems, Threats Attacks on Mobile and Cellular Networks Enterprise Mobile Security Organizational Security Policies for Mobile Environments.

Unit IV: Proxy Servers (7 Hours)
Anonymizers, Phishing Attacks, Identity Theft Techniques, Password Cracking Methods, Keyloggers and Spyware, Viruses and Worms, Trojan Horses and Backdoors, Steganography, Denial of Service (DoS), and Distributed DoS Attacks, SQL Injection Attacks, Buffer Overflow Attacks, Wireless Network Attacks

Unit V: Emerging Cyber Threats (7 Hours)
Cyber Threat Intelligence, Malware Analysis Fundamentals, Security Operations and Incident Response, Digital Forensics Overview, Cloud Security Challenges, IoT Security Issues, AI in Cyber Security, Contemporary Cyber Security Challenges and Future Trends.

Learning Resources:

Text Books:

[T1] Nina Godbole and Sunit Belapure, *"Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives"*, Wiley India, 1st Edition 2013.
[T2] William Stallings and Lawrie Brown, *"Computer Security: Principles and Practice"*, Pearson, 1st Edition.

Reference Books:

[R1] Charles P. Pfleeger and Shari Lawrence Pfleeger, *"Security in Computing"*, Pearson, 5th Edition.
[R2] Papoulis, Athanasios, and S. Unnikrishna Pillai, *"Introduction to Cyber Security"*, Tata McGraw Hill Education, 1st Edition 2002.

Course Title: High Performance Computing

Course Code: ADLM 620

L-T-P: 3-0-0

Credits: 3

Pre-requisites:

Course Outcomes:

CO-1	Explain the fundamental concepts, architectures, and performance metrics of high-performance computing systems.
CO-2	Analyze and design parallel algorithms for scientific and engineering applications using appropriate decomposition and partitioning techniques.
CO-3	Apply numerical and scientific computing methods, including sparse matrix and iterative solution techniques, to large-scale computational problems.
CO-4	Develop and optimize parallel applications using programming models such as MPI, OpenMP, and CUDA for high-performance computing environments.

Course Articulation Matrix:

COs	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2
CO1	2	1	–	1	1	2	1
CO2	3	2	1	2	–	3	1
CO3	3	3	1	3	–	3	2
CO4	3	2	2	3	1	3	3

1=addressed to small extent
2= addressed significantly
3= addressed strongly (major part of course)

Syllabus:

Unit I: Introduction to High Performance Computing
Introduction to High Performance Computing (HPC), Computational Science and Engineering applications, characteristics and requirements of scientific computing, computational complexity, performance metrics and measurements, speedup, efficiency, scalability, granularity and partitioning, forms of parallelism, performance-oriented computing, real-world HPC applications and case studies.

Unit II: Parallel Architectures and Memory Systems

Memory hierarchy and performance considerations, multi-core processors, shared-memory and distributed-memory architectures, symmetric multiprocessors, vector processors, supercomputers and petascale systems, heterogeneous computing platforms, application accelerators, reconfigurable computing, emerging computing architectures and their applications.

Unit III: Parallel Algorithms and Scientific Computing
Parallel algorithm design principles, decomposition and partitioning techniques, divide-and-conquer strategies, balanced trees, pointer jumping, matrix operations and linear algebra algorithms, graph and irregular algorithms, sorting algorithms, Monte Carlo techniques. Sparse matrices, storage schemes, discretization of differential equations, matrix reordering and direct solution methods.

Unit IV: Iterative Methods and Domain Decomposition
Successive Over-Relaxation (SOR), steepest descent method, conjugate gradient algorithm, Krylov subspace methods, Arnoldi method, GMRES, Lanczos algorithm, convergence analysis, block Krylov methods, preconditioning techniques, ILU factorization, multigrid methods. Domain decomposition methods, Schwarz algorithms, Schur complement methods, graph partitioning using geometric and spectral techniques.

Unit V: Parallel Programming and Performance Optimization
Parallel programming paradigms, task and data parallelism, synchronization methods, collective operations, SPMD programming model. OpenMP programming for shared-memory systems, MPI programming for distributed-memory systems, parallelization of matrix solvers and scientific applications. CUDA architecture and programming model, thread hierarchy, GPU acceleration for scientific computing. Performance bottleneck analysis, optimization techniques, use of HPC libraries, tools, and frameworks.

Learning Resources:

Text Books:

[T1] Georg Hager and Gerhard Wellein, "*High-Performance Computing: Paradigm and Infrastructure*", CRC, Second Edition.

Course Title: Blockchain Technology

Course Code: ADBM 621

L-T-P: 3-0-2

Credits: 4

Pre-requisites: NIL

Course Outcomes:

CO-1	Explain the need of blockchain systems and demonstrate the fundamentals of cryptocurrency and cryptographic primitives.
CO-2	Demonstrate blockchain tools, Nakamoto consensus, and working principles of payment verification protocols.
CO-3	Describe and analyse consensus algorithms as per application requirements.
CO-4	Design and develop a communication model for sending and receiving messages in blockchain transactions.
CO-5	Design, develop, and analyse real-time distributed applications using smart contracts and blockchain platforms.

Course Articulation Matrix:

COs	**POs & PSOs**						
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PSO1**	**PSO2**
CO1	3	2	1	1	1	2	1
CO2	2	2	1	1	3	2	2
CO3	2	3	2	2	2	2	2
CO4	2	2	3	2	3	3	2
CO5	2	3	3	2	3	3	3

1 - Slightly; 2 - Moderately; 3 - Substantially

Syllabus:

Unit I: Blockchain Fundamentals and Cryptographic Foundations (8 Hours)
Background and evolution of blockchain technology; distributed systems; distributed ledger technology (DLT) concept, features, benefits, and relevance in applications; security and privacy; cryptography, hash functions, and permissioned blockchain concepts; symmetric and asymmetric cryptography; DES; Patricia trees; distributed hash tables (DHTs); digital signatures; sign-then-encrypt and encrypt-then-sign; Elliptic Curve Digital Signature Algorithm (ECDSA); ECDSA using OpenSSL; homomorphic encryption; signcryption; zero-knowledge proofs; blind signatures; encoding schemes.

Unit II: Bitcoin Basics, Keys, Addresses and Transactions (7 Hours)
Bitcoin definition and architecture; keys and addresses; public keys and private keys in Bitcoin; Bitcoin currency units; Base58Check encoding; vanity addresses; transaction life cycle; transaction structure; Bitcoin script language; commonly used opcodes; types of transactions;

transaction fee; contracts; transaction malleability; transaction pools.

Unit III: Blockchain Architecture, Mining and Bitcoin Network (8 Hours)
Structure of a block; block header; genesis block; tasks of miners; synchronizing with the network; Proof of Work; mining algorithm; hashing rate; mining systems: CPU, GPU, FPGA, and ASIC; mining pools; Bitcoin network; wallets; payments; Bitcoin investment, buying and selling bitcoins; Bitcoin installation; Bitcoin programming and command-line interface; Bitcoin Improvement Proposals (BIPs).

Unit IV: Alternative Coins and Consensus Algorithms (7 Hours)
Greedy Alternative coins and their theoretical foundations; alternatives to Proof of Work; non-outsourceable puzzles; difficulty adjustment and retargeting algorithms; Bitcoin limitations; extended protocols on top of Bitcoin; development of altcoins; consensus algorithms; coins; mining guide; Zcash**.**

Unit V: Smart Contracts, Ethereum Development and Applications (7 Hours)
Smart contracts: definition and Ricardian contracts; Ethereum 101 introduction; Ethereum blockchain; transactions; contract creation transaction; message call transaction; elements of the Ethereum blockchain; Ethereum Virtual Machine (EVM); precompiled contracts; accounts; blocks; transaction and block validation mechanism; Ethereum development tools and clients; introduction to Solidity; Hyperledger; protocols; applications outside currencies.

Learning Resources:

Text Books:
[T1] Imran Bashir, "*Mastering Blockchain*".
[T2] Melanie Swan, "*Blockchain: Blueprint for a New Economy"*.

Reference Books:
[R1] Alan T. Norman, "*Blockchain Technology Explained: The Ultimate Beginner's Guide about Blockchain Wallet, Mining, Bitcoin, Ethereum, Litecoin, Zcash, Monero, Ripple, Dash, IOTA and Smart Contracts*".
[R2] Jan Veuger, "*Blockchain Technology and Applications*".
[R3] Andreas Bolfing, "*Cryptographic Primitives in Blockchain Technology: A Mathematical Introduction*".

Other Suggested Readings:

List of Experiments:
1. Study and implementation of SHA-256 hashing for blockchain data integrity.
2. Study and implementation of public key and private key generation using ECDSA.
3. Create and verify a digital signature for blockchain transactions.
4. Study and implementation of a simple blockchain with blocks, hash, and previous hash.
5. Simulate Proof of Work consensus mechanism using mining difficulty.
6. Study and implementation of Bitcoin wallet address generation.
7. Create and validate a basic blockchain transaction.
8. Study and implementation of a smart contract using Solidity.
9. Deploy and test a smart contract on a local Ethereum blockchain using Ganache/Remix.
10. Study of Hyperledger/blockchain-based distributed applications and prepare a mini project.