ACADEMIC
REGULATIONS
COURSE STRUCTURE
AND
DETAILED SYLLABUS

ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

M.TECH

COMPUTER SCIENCE AND ENGINEERING
(**ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING**)

(Applicable for the batches admitted from 2026-2027)

AR-26

ADITYA INSTITUTE OF TECHNOLOGY AND MANAGEMENT
(An Autonomous Institution)

Approved by AICTE,
Recognized Under 2(f) & 12(b) of UGC Permanently
Affiliated to JNTUGV, Vizianagaram
K.Kotturu,Tekkali, Srikakulam -532201, Andhra Pradesh.

www.adityatekkali.edu.in

M.Tech
Computer Science and Engineering
(Artificial Intelligence and Machine Learning)
(AR26 Regulations)

COURSE STRUCTURE

Semester	Subject Code	Theory/Lab	L	T	P	C
I	26MAI1001	Mathematical Foundations for Artificial Intelligence	4	0	0	4
	26MAI1002	Machine Learning	4	0	0	4
	26MAI1003	Data Structures and Algorithms for AI	4	0	0	4
	26MAI1E1X	Program Elective – 1	3	0	0	3
	26MAI1E2X	Program Elective – 2	3	0	0	3
	26MAI1101	Machine Learning Lab	0	0	4	2
	26MAI1102	Data Structures and Algorithms for AI Lab	0	0	4	2
	26MAC1001 26MAC1002 26MAC1003	Audit Course: English for Research Paper Writing Disaster Management Constitution of India	2	0	0	0
Total			**20**	**0**	**8**	**22**

Program Elective – 1

Subject Code	Theory/Lab
26MAI1E11	Cloud Computing for AI
26MAI1E12	Optimization Techniques for AI
26MAI1E13	Data Visualization
26MAI1E14	Database Programming with PL/SQL

Program Elective – 2

Subject Code	Theory/Lab
26MAI1E21	Big Data Analytics
26MAI1E22	Data Warehousing and Mining
26MAI1E23	Statistical Learning Theory
26MAI1E24	Probabilistic Graphical Models

Semester	Subject Code	Theory/Lab	L	T	P	C
II	26MAI1004	Deep Learning	4	0	0	4
	26MAI1005	Natural Language Processing	4	0	0	4
	26MAI1006	Computer Vision	4	0	0	4
	26MAI1E3X	Program Elective – 3	3	0	0	3
	26MAI1E4X	Program Elective – 4	3	0	0	3
	26MAI1103	Deep Learning, NLP & Computer Vision Lab	0	0	4	2
	26MCC1001	Research Methodology and IPR / Swayam 12 week MOOC course – RM&IPR	2	0	0	2
Total			**20**	**0**	**4**	**22**

Program Elective – 3

Subject Code	Theory/Lab
26MAI1E31	Reinforcement Learning
26MAI1E32	Speech and Audio Processing
26MAI1E33	Information Retrieval Systems
26MAI1E34	Explainable AI

Program Elective – 4

Subject Code	Theory/Lab
26MAI1E41	Generative AI and Large Language Models
26MAI1E42	GPU Programming
26MAI1E43	Artificial Intelligence of Things
26MAI1E44	AI for Cyber Security

Semester	Subject Code	Theory/Lab	L	T	P	C
III	26MAI2I01	Internship/ Industrial Training (8weeks)	-	-	-	3
	26MAI2201	Technical Seminar	-	-	4	2
	26MAI2202	Comprehensive Viva	-	-	-	2
	26MAI2203	Dissertation Phase – 1	0	0	20	10
Total			**0**	**0**	**24**	**17**

Semester	Subject Code	Theory/Lab	L	T	P	C
IV	26MAI2204	Dissertation Phase – 2	0	0	32	16
Total			**0**	**0**	**32**	**16**

Mathematical Foundations for Artificial Intelligence

Subject Code: 26MAI1001

L	T	P	C
4	0	0	4

Course Objectives

1. To provide a strong foundation in linear algebra and vector spaces for AI applications.
2. To develop understanding of probability theory and statistics for modeling uncertainty.
3. To introduce calculus and optimization techniques used in machine learning algorithms.
4. To enable students to analyze mathematical models underlying AI systems.
5. To equip learners with skills to apply mathematical tools in real-world AI problems.

Course Outcomes

After completion of the course, students will be able to:

1. Use linear algebra concepts (matrices, eigenvalues) in AI models.
2. Interpret probabilistic models and statistical measures in data-driven systems.
3. Utilize differentiation and integration techniques in optimization problems.
4. Assess and compare optimization methods used in machine learning.
5. Formulate mathematical solutions for AI-based real-world problems.

UNIT - I: Linear Algebra for AI

Matrix Algebra and Operations, Vectors and Vector Spaces, Matrix Algebra and Operations, Systems of Linear Equations, Eigenvalues and Eigenvectors, Singular Value Decomposition (SVD), Applications in AI: PCA, dimensionality reduction

UNIT - II: Probability Theory

Basics of Probability, Random Variables (Discrete & Continuous), Probability Distributions (Bernoulli, Binomial, Gaussian), Joint, Marginal, Conditional Probability, Bayes' Theorem, Applications in AI: Naïve Bayes, uncertainty modeling

UNIT - III: Statistics for Machine Learning

Descriptive Statistics (Mean, Variance, Standard Deviation), Inferential Statistics, Sampling Techniques, Hypothesis Testing (t-test, chi-square), Correlation and Regression, Applications: Data analysis and feature selection

UNIT - IV: Calculus for Optimization

Functions, Limits, and Continuity, Differentiation (Single & Multivariable), Partial Derivatives and Gradients, Chain Rule and Jacobian, Integration and Applications, Applications: Gradient Descent, Backpropagation

UNIT - V: Optimization Techniques

Convex Sets and Convex Functions, Optimization Problems and Constraints, Gradient Descent and Variants, Lagrange Multipliers, Numerical Optimization Methods, Applications: Loss minimization in ML models

Textbooks

1. Mathematics for Machine Learning, 1st Edition (2020), Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong, Cambridge University Press,
2. Introduction to Linear Algebra, 5th Edition (2016), Gilbert Strang, Wellesley-Cambridge Press

3. Probability and Statistics for Engineering and the Sciences, 9th Edition (2015), Jay L. Devore, Cengage Learning

Reference Books

1. Pattern Recognition and Machine Learning, 1st Edition (2006), Christopher M. Bishop, Springer
2. The Elements of Statistical Learning, 2nd Edition (2009), Trevor Hastie, Robert Tibshirani, Jerome Friedman, Springer
3. Convex Optimization, 1st Edition (2004), Stephen Boyd, Lieven Vandenberghe, Cambridge University Press

Reference Links

1. NPTEL - Courses on Linear Algebra, Probability, and AI
2. MIT OpenCourseWare - Mathematics for AI (Strang lectures)
3. Khan Academy - Foundations in calculus, probability, and statistics

Machine Learning

Subject Code: 26MAI1002

L	T	P	C
4	0	0	4

Course Objectives

1. To develop a strong foundation in the theoretical principles of machine learning and statistical modeling.
2. To understand and apply mathematical concepts such as probability, linear algebra, and optimization in ML algorithms.
3. To analyze and implement supervised and unsupervised learning techniques for real-world problems.
4. To design, train, and evaluate machine learning models using appropriate performance metrics.
5. To explore advanced learning paradigms including deep learning and reinforcement learning.

Course Outcomes

After completion of the course, students will be able to:

1. Analyze machine learning problems and select suitable models based on data characteristics.
2. Implement supervised and unsupervised learning algorithms using standard tools and frameworks.
3. Evaluate model performance using appropriate metrics and validation techniques.
4. Design optimized machine learning solutions using mathematical and computational principles.
5. Apply advanced techniques such as neural networks and reinforcement learning to complex tasks.

UNIT - I: Introduction to Machine Learning

Definition and scope of Machine Learning, Types of Learning: Supervised, Unsupervised, Semi-supervised, Reinforcement Learning, ML pipeline: Data collection, preprocessing, feature engineering, model selection, Bias-Variance tradeoff, Overfitting and Underfitting, Model evaluation basics, Introduction to datasets and data preprocessing, Real-world applications of ML

UNIT - II: Mathematical Foundations for ML

Linear Algebra: Vectors, matrices, eigenvalues, eigenvectors, Matrix factorization (SVD).
Probability Theory: Random variables, distributions (Gaussian, Bernoulli, Binomial), Conditional probability and Bayes theorem.
Statistics: Mean, variance, covariance, Maximum Likelihood Estimation (MLE), Maximum A Posteriori (MAP).
Optimization Techniques: Gradient Descent and variants, Convex optimization basics.

UNIT - III: Supervised Learning

Regression Techniques: Linear Regression, Polynomial Regression, Ridge and Lasso Regression
Classification Techniques: Logistic Regression, k-Nearest Neighbors (k-NN), Support Vector Machines (SVM), Decision Trees and Random Forests
Loss functions and regularization, Model selection and hyper parameter tuning

UNIT - IV: Unsupervised Learning

Clustering Techniques: K-Means Clustering, Hierarchical Clustering, DBSCAN
Dimensionality Reduction: Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA)
Association Rule Learning (Apriori Algorithm), Anomaly Detection, Evaluation metrics for clustering.

UNIT - V: Advanced Machine Learning

Neural Networks & Deep Learning: Perceptron, Multilayer Perceptron (MLP), Backpropagation Algorithm, Activation functions

Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), LSTM

Reinforcement Learning: Markov Decision Process (MDP), Q-Learning

Model Evaluation: Cross-validation, ROC, AUC, Precision, Recall, F1-score, Model interpretability.

Textbooks

1. Christopher M. Bishop. 2006. Pattern Recognition and Machine Learning. Springer, New York, NY.
2. Kevin P. Murphy. 2012. Machine Learning: A Probabilistic Perspective. MIT Press, Cambridge, MA.
3. Trevor Hastie, Robert Tibshirani, and Jerome Friedman. 2009. The Elements of Statistical Learning (2nd ed.). Springer, New York, NY.

Reference Books

1. Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. Deep Learning. MIT Press, Cambridge, MA.
2. Aurélien Géron. 2022. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (3rd ed.). O'Reilly Media, Sebastopol, CA.
3. Richard S. Sutton and Andrew G. Barto. 2018. Reinforcement Learning: An Introduction (2nd ed.). MIT Press, Cambridge, MA

Reference Links

1. NPTEL - Machine Learning [https://nptel.ac.in/courses/106106139]
2. MIT OpenCourseWare - Machine Learning [https://ocw.mit.edu/courses/6-036-machine-learning]
3. Coursera - Machine Learning by Andrew Ng [https://www.coursera.org/learn/machine-learning]

Data Structures and Algorithms for AI

Subject Code: 26MAI1003

L	T	P	C
4	0	0	4

Course Objectives

1. To develop a strong understanding of advanced data structures and their role in efficient problem solving.
2. To analyze and apply algorithm design techniques such as divide-and-conquer, greedy, and dynamic programming.
3. To explore advanced tree, hashing, and string processing structures used in AI applications.
4. To understand graph algorithms and their applications in real-world AI systems like recommendation engines and knowledge graphs.
5. To enable students to design optimized and scalable solutions for complex computational and AI-driven problems.

Course Outcomes

After completion of the course, students will be able to:

1. Implement and analyze advanced data structures such as AVL trees, heaps, tries, and hashing techniques.
2. Apply appropriate algorithm design paradigms to solve complex computational problems efficiently.
3. Evaluate time and space complexity of algorithms and optimize them for large-scale AI applications.
4. Design solutions using dynamic programming, backtracking, and graph algorithms for real-world scenarios.
5. Apply data structures and algorithms in AI domains such as NLP, search systems, and recommendation systems.

UNIT - I: Basic Data Structures, Tree Structures, and Heaps

Review of basic data structures: arrays, linked lists, stacks, queues. Tree structures: Binary Trees, Binary Search Trees. Multiway trees: B-Trees, B+ Trees. Self-balancing trees: AVL Trees, Red-Black Trees.
Heaps and Priority Queues: Binary Heap, Binomial Heap, Fibonacci Heap, Heap operations and analysis. Applications in AI: Priority scheduling in search algorithms, Memory-efficient indexing.

UNIT - II: Hashing, String Structures, and Indexing

Hashing Techniques: Hash functions, load factor, Collision resolution: Chaining, Open Addressing (Linear, Quadratic, Double Hashing). Advanced structures: Trie, Suffix Trees, Suffix Arrays. String processing algorithms: Pattern matching (Naïve, KMP, Rabin-Karp). Applications in AI: Autocomplete systems, NLP indexing and dictionary lookup, Spell checking and search engines.

UNIT - III: Algorithm Design and Complexity Analysis

Asymptotic analysis: Time & Space Complexity, Big-O, Omega (Ω), Theta (Θ). Divide and Conquer: Merge Sort, Quick Sort.
Greedy Algorithms: Huffman Coding, Activity Selection.
Applications in AI: Efficient model computation. Data preprocessing optimization.

UNIT - IV: Dynamic Programming and Advanced Techniques

Dynamic Programming: Principles and optimization strategies, Problems: Knapsack, Longest Common Subsequence, Matrix Chain Multiplication. Backtracking: N-Queens, Subset Sum. Branch and Bound:

Travelling Salesman Problem (TSP)
Applications in AI: Sequence alignment, Decision-making and optimization problems.

UNIT - V: Graph Algorithms for Artificial Intelligence
Graph representations: Adjacency Matrix, Adjacency List. Graph traversal: Breadth First Search (BFS), Depth First Search (DFS)
Shortest Path Algorithms: Dijkstra, Bellman-Ford, Floyd-Warshall. Minimum Spanning Trees: Prim's Algorithm, Kruskal's Algorithm.
Network Flow: Ford-Fulkerson Algorithm. Introduction to A* Search Algorithm.
Applications in AI: Knowledge graphs, Social networks, Recommendation systems, Pathfinding in intelligent agents.

Textbooks

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. 2022. Introduction to Algorithms (4th ed.). MIT Press, Cambridge, MA.
2. Mark Allen Weiss. 2014. Data Structures and Algorithm Analysis in C (2nd ed.). Pearson, Boston, MA.
3. Ellis Horowitz, Sartaj Sahni, and Susan Anderson-Freed. 2008. Fundamentals of Data Structures in C (2nd ed.). Universities Press, Hyderabad, India.

Reference Books

1. Steven S. Skiena. 2020. The Algorithm Design Manual (3rd ed.). Springer, Cham, Switzerland.
2. Robert Sedgewick and Kevin Wayne. 2011. Algorithms (4th ed.). Addison-Wesley, Boston, MA.
3. Donald E. Knuth. 1997. The Art of Computer Programming, Volume 1: Fundamental Algorithms (3rd ed.). Addison-Wesley, Reading, MA.

Reference Links

1. MIT OpenCourseWare, Algorithms [https://ocw.mit.edu/courses/6-006-introduction-to-algorithms]
2. Stanford Algorithms Specialization (Coursera) [https://www.coursera.org/specializations/algorithms]
3. GeeksforGeeks, Data Structures and Algorithms [https://www.geeksforgeeks.org/data-structures]

Cloud Computing for AI

(Program Elective - I)

L	T	P	C
3	0	0	3

Subject Code: 26MAI1E11

Course Objectives

1. To understand fundamental concepts, architectures, and service models of cloud computing.
2. To analyze virtualization and containerization techniques for scalable AI workloads.
3. To explore cloud platforms and services supporting AI/ML development and deployment.
4. To examine distributed computing frameworks for large-scale data processing in AI.
5. To implement AI/ML models in cloud environments using MLOps and deployment strategies.

Course Outcomes

After completion of the course, students will be able to:

1. Explain cloud computing models, architectures, and security considerations.
2. Apply virtualization and containerization tools for efficient AI system design.
3. Utilize cloud platforms and services for building AI-based applications.
4. Analyze distributed data processing frameworks for large-scale AI workloads.
5. Deploy, monitor, and optimize machine learning models using cloud-based MLOps practices.

UNIT - I: Fundamentals of Cloud Computing

Definition and Evolution of Cloud Computing, NIST Characteristics of Cloud Computing, Cloud Service Models: IaaS, PaaS, SaaS. Deployment Models: Public, Private, Hybrid, Multi-cloud. Cloud Architecture: Frontend, Backend, Data Centers, Resource Pooling. Security and Compliance: Data Privacy, Governance, SLAs.

UNIT - II: Virtualization and Containerization

Virtualization Concepts: Hypervisors (Type-1 and Type-2), Virtual Machines and Resource Allocation, Storage and Network Virtualization. Containerization: Containers vs Virtual Machines, Docker Architecture and Components, Container Orchestration: Kubernetes (Pods, Services), Microservices Architecture: Principles and Benefits. DevOps Basics: CI/CD Pipelines for AI Applications.

UNIT - III: Cloud Platforms for AI

Overview of Cloud Providers: AWS, Microsoft Azure, Google Cloud Platform. AI/ML Services: Vision APIs, NLP APIs, Speech APIs. Cloud Storage: Object Storage, Block Storage, Databases. Compute Services: VMs, GPUs, TPUs. AI Platforms: SageMaker, Azure ML, Vertex AI. Identity and Access Management (IAM), API Gateway and Cloud Integration Services.

UNIT - IV: Distributed Computing for AI

Need for Distributed Computing in AI. Hadoop Ecosystem: HDFS, MapReduce. Apache Spark: Architecture, RDDs, DataFrames. Stream Processing: Spark Streaming, Data Pipelines and Workflow Orchestration, Parallel and Distributed Training of ML Models, Fault Tolerance and Data Replication.

UNIT - V: AI/ML Deployment on Cloud

ML Lifecycle in Cloud: Data Collection → Training → Deployment → Monitoring. MLOps: Model Versioning, Experiment Tracking, CI/CD for ML. Model Deployment: Batch vs Real-time Inference. Serverless Computing: Functions-as-a-Service (FaaS), Auto-scaling and Load Balancing, Model Optimization and Performance Monitoring.

Textbooks

1. Thomas Erl, Ricardo Puttini, and Zaigham Mahmood. 2013. Cloud Computing: Concepts, Technology & Architecture. Pearson Education, Boston, MA.
2. Michael J. Kavis. 2014. Architecting the Cloud: Design Decisions for Cloud Computing Service Models. Wiley, Hoboken, NJ.
3. Valliappa Lakshmanan, Sara Robinson, and Michael Munn. 2020. Machine Learning Design Patterns. O'Reilly Media, Sebastopol, CA.

Reference Textbooks

1. Martin Kleppmann. 2017. Designing Data-Intensive Applications. O'Reilly Media, Sebastopol, CA.
2. Brendan Burns, Joe Beda, and Kelsey Hightower. 2022. Kubernetes: Up and Running (3rd ed.). O'Reilly Media, Sebastopol, CA.
3. Hannes Hapke and Catherine Nelson. 2020. Building Machine Learning Pipelines. O'Reilly Media, Sebastopol, CA.

Reference Links

1. AWS Documentation – https://docs.aws.amazon.com
2. Microsoft Azure Documentation – https://learn.microsoft.com/azure
3. Google Cloud Documentation – https://cloud.google.com/docs

Optimization Techniques for AI

(Program Elective - I)

Subject Code: 26MAI1E12

L	T	P	C
3	0	0	3

Course Objectives

1. To provide a strong foundation in mathematical optimization, including formulation and analysis of optimization problems.
2. To develop understanding of convex and non-convex optimization techniques relevant to AI/ML models.
3. To equip students with classical and modern optimization algorithms for solving real-world computational problems.
4. To analyze and apply constrained and unconstrained optimization methods in machine learning contexts.
5. To enable the application of optimization techniques in deep learning, large-scale data analysis, and intelligent systems.

Course Outcomes

After successful completion of the course, students will be able to:

1. Formulate optimization problems arising in AI/ML using mathematical models.
2. Analyze convex and non-convex functions and determine optimality conditions.
3. Apply classical optimization techniques such as linear programming and duality to real-world problems.
4. Design and implement first-order and second-order optimization algorithms for machine learning tasks.
5. Evaluate and optimize AI/ML models using advanced techniques such as adaptive gradient methods and regularization.

UNIT - I: Foundations of Optimization

Introduction to optimization: definition, scope, and applications in AI, Mathematical preliminaries: vectors, matrices, norms, and inner products, Convex sets and convex functions, Properties of convex optimization problems. Problem formulation: objective functions, constraints, feasible region. Optimality conditions: local vs global minima, First-order and second-order conditions, Introduction to Lagrangian function.

UNIT - II: Linear and Classical Optimization Techniques

Linear Programming: formulation and graphical method, Standard and canonical forms. Simplex method: algorithm and implementation, Degeneracy, unboundedness, and infeasibility. Duality theory: primal-dual relationships, Dual simplex method, Sensitivity analysis, Applications of LP in resource allocation and AI decision systems

UNIT - III: Unconstrained Optimization and First-Order Methods

Unconstrained optimization problem formulation. Gradient descent method: basic, batch, and mini-batch, Convergence analysis of gradient-based methods, Stochastic Gradient Descent (SGD), Momentum-based methods, Learning rate strategies and scheduling, Coordinate descent methods, Applications in regression and classification problems.

UNIT - IV: Advanced Optimization Techniques

Newton's method and quasi-Newton methods (BFGS, DFP), Second-order optimality conditions, Constrained optimization techniques, Lagrange multipliers and dual problems, Karush-Kuhn-Tucker (KKT) conditions, Penalty and barrier methods, Interior point methods, Convex vs non-convex optimization challenges.

UNIT - V: Optimization in AI and Machine Learning

Role of optimization in machine learning and deep learning. Loss functions: MSE, cross-entropy, hinge loss. Regularization techniques: L1, L2, Elastic Net. Optimization challenges in deep neural networks, Vanishing and exploding gradients. Adaptive optimization algorithms: AdaGrad, RMSProp, Adam. Hyperparameter tuning and optimization, Optimization in large-scale and distributed learning.

Textbooks

1. Stephen Boyd and Lieven Vandenberghe. 2004. Convex Optimization. Cambridge University Press, Cambridge, UK.
2. Jorge Nocedal and Stephen J. Wright. 2006. Numerical Optimization (2nd ed.). Springer, New York, NY.
3. Suvrit Sra, Sebastian Nowozin, and Stephen J. Wright. 2011. Optimization for Machine Learning. MIT Press, Cambridge, MA.

Reference Books

1. Dimitri P. Bertsekas. 2016. Nonlinear Programming (3rd ed.). Athena Scientific, Belmont, MA.
2. Pablo Pedregal. 2006. Introduction to Optimization. Springer, New York, NY.
3. Christopher M. Bishop. 2006. Pattern Recognition and Machine Learning. Springer, New York, NY.

Reference Links

1. Stanford University – Convex Optimization Course [https://web.stanford.edu/~boyd/cvxbook/]
2. MIT Open Courseware – Optimization Methods [https://ocw.mit.edu/courses/15-093j-optimization-methods-fall-2009/]
3. Deep Learning Optimization [https://www.tensorflow.org/guide/keras/optimizers]

Data Visualization

(Program Elective - I)

Subject Code: 26MAI1E13

L	T	P	C
3	0	0	3

Course Objectives

The course aims to:

1. Develop a strong foundation in the principles and theories of data visualization, including perception and cognition.
2. Enable students to preprocess, transform, and structure complex datasets for effective visual representation.
3. Equip learners with skills to design and implement various visualization techniques for multidimensional and large-scale data.
4. Introduce modern tools and libraries for creating static, interactive, and web-based visualizations.
5. Foster the ability to communicate insights through storytelling and visual analytics in AI/ML applications.

Course Outcomes

Upon successful completion of the course, students will be able to:

1. Apply visualization principles and design guidelines to create effective and meaningful visual representations of data.
2. Analyze and preprocess structured and unstructured datasets for visualization purposes.
3. Design and implement diverse visualization techniques for different data types and analytical needs.
4. Utilize modern visualization tools and programming libraries to develop interactive dashboards and visual analytics systems.
5. Interpret and communicate insights from data using storytelling techniques in AI/ML contexts.

UNIT - I: Fundamentals of Data Visualization

Introduction to Data Visualization: Importance and Applications. Types of Data: Categorical, Numerical, Temporal, Spatial, Principles of Visual Perception (Gestalt principles, pre-attentive attributes). Visualization Design Principles: Clarity, Accuracy, Efficiency. Data Encoding: Color, Shape, Size, Position

UNIT - II: Data Preparation and Transformation

Data Cleaning and Preprocessing Techniques, Handling Missing and Noisy Data. Data Transformation: Normalization, Aggregation, Sampling, Data Reduction Techniques, Preparing Data for Visualization Pipelines.

UNIT - III: Visualization Techniques

Basic Charts: Bar, Line, Pie, Histogram, Scatter Plot. Multivariate and Multidimensional Visualization. Advanced Visualizations: Heatmaps, Tree maps, Network graphs. Geospatial Visualization Techniques, Visualizing High-dimensional Data (PCA, t-SNE representations).

UNIT - IV: Tools and Libraries for Visualization

Python Libraries: Matplotlib, Seaborn, Plotly. Data Handling with Pandas and NumPy. Web-based Visualization: D3.js basic, Dashboard Development Tools (Tableau, Power BI – overview), Integration with AI/ML workflows

UNIT - V: Advanced Topics in Data Visualization

Interactive Visualization and Dashboards, Visual Analytics and Decision Support Systems, Storytelling with Data, AI-driven Visualization (Automated insights, Explainable AI visuals), Real-world Case Studies and Applications.

Textbooks

1. Tamara Munzner. 2014. Visualization Analysis and Design. CRC Press, Boca Raton, FL.
2. Scott Murray. 2017. Interactive Data Visualization for the Web (2nd ed.). O’Reilly Media, Sebastopol, CA.
3. Cole Nussbaumer Knaflic. 2015. Storytelling with Data: A Data Visualization Guide for Business Professionals. Wiley, Hoboken, NJ.

Reference Books

1. Stephen Few. 2012. Show Me the Numbers: Designing Tables and Graphs to Enlighten (2nd ed.). Analytics Press, Burlingame, CA.
2. Alberto Cairo. 2012. The Functional Art: An Introduction to Information Graphics and Visualization. New Riders, Berkeley, CA.
3. Claus O. Wilke. 2019. Fundamentals of Data Visualization. O’Reilly Media, Sebastopol, CA.

Reference Links

1. Official Matplotlib Documentation: [https://matplotlib.org/stable/index.html]
2. Seaborn Visualization Library: [https://seaborn.pydata.org/]
3. D3.js Official Documentation: [https://d3js.org/]

Database Programming with PL/SQL

(Program Elective - I)

Subject Code: 26MAI1E14

L	T	P	C
3	0	0	3

Course Objectives

1. To provide a strong foundation in relational database concepts and structured query language (SQL) for efficient data manipulation and retrieval.
2. To develop proficiency in advanced SQL techniques, including complex queries, indexing, and query optimization strategies.
3. To introduce PL/SQL programming for building robust, modular, and reusable database applications.
4. To enable students to design and implement stored procedures, functions, triggers, and packages for enterprise-level database systems.
5. To integrate database programming concepts with AI/ML workflows, emphasizing data preprocessing, performance tuning, and secure data handling.

Course Outcomes

Upon successful completion of the course, students will be able to:

1. Design and manipulate relational databases using SQL for structured data storage and retrieval.
2. Apply advanced SQL techniques and optimization strategies to improve query performance.
3. Develop PL/SQL programs using control structures, cursors, and exception handling.
4. Implement database components such as procedures, triggers, and packages for real-world applications.
5. Analyze and manage large-scale datasets efficiently for AI/ML applications with a focus on performance, scalability, and security.

UNIT - I: Fundamentals of Databases and SQL

Database concepts: Data models, relational model, schema, instances, ER modeling and relational schema design. SQL basics: DDL, DML, DCL, TCL commands. Constraints: Primary key, foreign key, unique, check, not null. Basic queries: SELECT, WHERE, ORDER BY, GROUP BY, HAVING. Joins: Inner, outer, self joins.

UNIT - II: Advanced SQL and Query Optimization

Nested queries and correlated subqueries. Set operations: UNION, INTERSECT, MINUS, Views and materialized views. Indexing: Types, creation, and usage. Query optimization techniques: Execution plans, cost-based optimization, Transactions and concurrency control (ACID properties).

UNIT - III: Introduction to PL/SQL and Control Structures

Overview of PL/SQL architecture. Block structure: Declaration, execution, exception sections, Variables, constants, and data types. Control structures: IF, CASE, LOOP, WHILE, FOR. Cursors: Implicit and explicit. Exception handling mechanisms.

UNIT - IV: Advanced PL/SQL - Procedures, Functions, Triggers, Packages

Stored procedures and functions: Creation, execution, parameters. Triggers: Types (before/after, row/statement level), use cases. Packages: Specification and body, modular programming. Dynamic SQL and bulk processing (FORALL, BULK COLLECT), Performance considerations in PL/SQL programs.

UNIT - V: Database Programming in Real-world Applications

Database connectivity with applications (JDBC, Python DB APIs), Data preprocessing and transformation for

AI/ML pipelines, Handling large datasets and data warehousing basics. Performance tuning: Query tuning, Indexing strategies, partitioning. Security: Authentication, authorization, roles, encryption.

Textbooks

1. Ivan Bayross. 2010. SQL, PL/SQL: The Programming Language of Oracle (4th ed.). BPB Publications, New Delhi, India.
2. Abraham Silberschatz, Henry F. Korth, and S. Sudarshan. 2019. Database System Concepts (7th ed.). McGraw-Hill Education, New York, NY.
3. Steven Feuerstein and Bill Pribyl. 2014. Oracle PL/SQL Programming (6th ed.). O'Reilly Media, Sebastopol, CA.

Reference Books

1. Ramez Elmasri and Shamkant B. Navathe. 2016. Fundamentals of Database Systems (7th ed.). Pearson, Boston, MA.
2. Markus Winand. 2012. SQL Performance Explained. Two Steps Beyond, Vienna, Austria.
3. Thomas Kyte and Darl Kuhn. 2014. Expert Oracle Database Architecture (3rd ed.). Apress, New York, NY.

Reference Links

1. Oracle Database Documentation
2. PostgreSQL Official Documentation
3. W3Schools SQL Tutorial

Big Data Analytics

(Program Elective - II)

Subject Code: 26MAI1E21

L	T	P	C
3	0	0	3

Course Objectives

1. To introduce fundamental concepts of big data, including characteristics, challenges, and ecosystem components.
2. To provide in-depth knowledge of distributed data storage and processing frameworks such as Hadoop and Spark.
3. To develop skills in big data analytics techniques including descriptive, predictive, and prescriptive analytics.
4. To enable integration of big data platforms with AI/ML algorithms for scalable intelligent systems.
5. To expose students to real-world big data applications, tools, and performance optimization strategies.

Course Outcomes

Upon successful completion of the course, students will be able to:

1. Analyze and interpret the characteristics and challenges associated with big data systems.
2. Design and implement distributed data processing using Hadoop ecosystem tools.
3. Apply Apache Spark for efficient in-memory data analytics and large-scale data processing.
4. Develop and deploy analytics models using big data frameworks integrated with AI/ML techniques.
5. Evaluate and optimize big data solutions for real-world applications in domains such as healthcare, finance, and social media.

UNIT - I: Big Data Fundamentals

Introduction to Big Data: Definition, Characteristics (5Vs: Volume, Velocity, Variety, Veracity, Value), Traditional vs Big Data Systems, Big Data Architecture and Lifecycle. Data Sources: Structured, Semi-structured, Unstructured, Challenges in Big Data Management, Overview of Distributed Computing.

UNIT - II: Hadoop Ecosystem

Hadoop Architecture: HDFS, NameNode, DataNode. MapReduce Programming Model, YARN Resource Management. Hadoop Ecosystem Tools: Hive, Pig, HBase, Sqoop, Flume, Data Ingestion and Storage Techniques.

UNIT - III: Apache Spark and Stream Processing

Introduction to Apache Spark. Spark Architecture: RDDs, DataFrames, Datasets, Spark Core and Spark SQL, Spark Streaming and Structured Streaming, MLlib for Machine Learning. Comparison: Hadoop MapReduce vs Spark.

UNIT - IV: Big Data Analytics Techniques

Data Preprocessing and Cleaning at Scale, Exploratory Data Analysis (EDA), Statistical Methods for Big Data, Machine Learning on Big Data: Classification, Clustering, Regression, Dimensionality Reduction Techniques, Visualization Techniques for Large Datasets.

UNIT - V: Applications and Advanced Topics

Big Data in AI/ML Applications, Real-time Analytics and Event Processing, Cloud-based Big Data Platforms (AWS, Azure, GCP), Data Security, Privacy, and Ethics, Performance Optimization and Scalability.

Textbooks

1. Tom White. 2015. Hadoop: The Definitive Guide (4th ed.). O'Reilly Media, Sebastopol, CA.
2. Seema Acharya and Subhashini Chellappan. 2015. Big Data Analytics. Wiley India, New Delhi, India.
3. Bill Chambers and Matei Zaharia. 2018. Spark: The Definitive Guide. O'Reilly Media, Sebastopol, CA.

Reference Books

1. Viktor Mayer-Schönberger and Kenneth Cukier. 2013. Big Data: A Revolution That Will Transform How We Live, Work, and Think. Houghton Mifflin Harcourt, Boston, MA.
2. Nathan Marz and James Warren. 2015. Big Data: Principles and Best Practices of Scalable Realtime Data Systems. Manning Publications, Shelter Island, NY.
3. Jiawei Han, Micheline Kamber, and Jian Pei. 2011. Data Mining: Concepts and Techniques (3rd ed.). Morgan Kaufmann, Burlington, MA.

Reference Links

1. Apache Hadoop Official Documentation: [https://hadoop.apache.org/docs/]
2. Apache Spark Official Documentation: [https://spark.apache.org/docs/latest/]
3. NPTEL Big Data Analytics Courses: [https://nptel.ac.in/courses/106/104/106104189/]

Data Warehousing and Mining

(Program Elective - II)

Subject Code: 26MAI1E22

L	T	P	C
3	0	0	3

Course Objectives

1. To understand the fundamental concepts, architecture, and components of data warehousing systems.
2. To analyze and design data warehouse schemas using dimensional modeling techniques.
3. To gain proficiency in ETL (Extract, Transform, Load) processes and data integration strategies.
4. To explore core data mining techniques including classification, clustering, association, and prediction.
5. To apply data warehousing and mining techniques in AI/ML-driven decision-making systems.

Course Outcomes

Upon successful completion of the course, students will be able to:

1. Analyze data warehousing architectures and justify design choices for enterprise applications.
2. Design dimensional models and implement ETL pipelines for real-world datasets.
3. Apply data mining algorithms for extracting meaningful patterns from large datasets.
4. Evaluate different data mining techniques and select appropriate models for AI/ML tasks.
5. Develop integrated solutions combining data warehousing and mining for intelligent decision support systems.

UNIT - I: Fundamentals of Data Warehousing

Introduction to Data Warehousing: Concepts, Characteristics, and Evolution, Operational Systems vs Data Warehousing Systems. Data Warehouse Architecture: Three-tier architecture, OLTP vs OLAP. OLAP Operations: Roll-up, Drill-down, Slice, Dice, Pivot. Data Warehouse Implementation Issues and Challenges.

UNIT - II: Data Warehouse Design & Modeling

Dimensional Modeling: Facts, Dimensions, Measures. Schema Design: Star Schema, Snowflake Schema, Fact Constellation, Data Warehouse Design Methodologies (Top-down vs Bottom-up approaches). ETL Processes: Data Extraction, Data Cleaning, Data Transformation, Data Loading, Metadata Management and Data Quality Issues.

UNIT - III: Data Mining Concepts and Techniques

Introduction to Data Mining: KDD Process. Data Preprocessing: Cleaning, Integration, Reduction, Transformation. Frequent Pattern Mining: Association Rule Mining (Apriori Algorithm). Classification Techniques: Decision Trees, Naïve Bayes, k-NN. Clustering Techniques: k-Means, Hierarchical Clustering.

UNIT - IV: Advanced Data Mining Methods

Advanced Classification: Support Vector Machines, Ensemble Methods. Advanced Clustering: DBSCAN, Density-based clustering, Outlier Detection and Anomaly Detection, Sequential Pattern Mining and Time-Series Analysis.

UNIT - V: Data Mining Tools, Trends, and AI/ML Integration

Data Mining Tools: WEKA, RapidMiner, Orange. Big Data Integration: Hadoop, Spark for Data Mining, Data Warehousing in Cloud Environments. Integration with AI/ML: Feature Engineering, Model Training Pipelines.

Textbooks

1. Jiawei Han, Micheline Kamber, and Jian Pei. 2011. Data Mining: Concepts and Techniques (3rd ed.). Morgan Kaufmann, Burlington, MA.
2. Ralph Kimball and Margy Ross. 2013. The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling (3rd ed.). Wiley, Hoboken, NJ.
3. Pang-Ning Tan, Michael Steinbach, Anuj Karpatne, and Vipin Kumar. 2018. Introduction to Data Mining (2nd ed.). Pearson, Boston, MA.

Reference Books

1. Alex Berson and Stephen J. Smith. 1997. Data Warehousing, Data Mining, and OLAP. McGraw-Hill, New York, NY.
2. Ian H. Witten, Eibe Frank, Mark A. Hall, and Christopher J. Pal. 2016. Data Mining: Practical Machine Learning Tools and Techniques (4th ed.). Morgan Kaufmann, Burlington, MA.
3. Sam Anahory and Dennis Murray. 1997. Data Warehousing in the Real World. Addison-Wesley, Boston, MA.

Reference Links

1. Coursera https://www.coursera.org]
2. Kaggle (Datasets and Practical Data Mining Exercises)[https://www.kaggle.com]
3. Apache Spark Documentation (Big Data Processing & Mining)[https://spark.apache.org/docs/latest/]

Statistical Learning Theory

(Program Elective - II)

Subject Code: 26MAI1E23

L	T	P	C
3	0	0	3

Course Objectives

The course aims to:

1. Develop a strong theoretical foundation of machine learning based on principles of statistical inference and probability theory.
2. Understand risk minimization frameworks including empirical risk minimization and structural risk minimization.
3. Analyze generalization ability of learning algorithms using theoretical bounds and capacity measures.
4. Explore foundational learning models such as PAC learning and VC dimension theory.
5. Apply statistical learning principles to kernel methods, support vector machines, and regularization techniques.

Course Outcomes

Upon successful completion of the course, students will be able to:

1. Formulate machine learning problems using statistical frameworks and probabilistic models.
2. Analyze and derive generalization bounds using VC dimension and related concepts.
3. Apply PAC learning principles to evaluate learnability of hypothesis classes.
4. Design and interpret kernel-based models including Support Vector Machines (SVMs).
5. Evaluate and select models using regularization techniques and statistical validation methods.

UNIT - I: Foundations of Statistical Learning

Introduction to Statistical Learning Theory, Learning paradigms: Supervised, Unsupervised, Semi-supervised. Hypothesis space and inductive bias, Loss functions and risk, Empirical Risk Minimization (ERM), Overfitting and under fitting, Bias–Variance trade-off.

UNIT - II: Probability Theory and Bayesian Learning

Probability theory fundamentals: random variables, distributions, expectation. Conditional probability and Bayes theorem, Bayesian inference and decision theory, Maximum Likelihood Estimation (MLE), Maximum A Posteriori (MAP) estimation, Bayesian networks and graphical models (overview).

UNIT - III: VC Dimension and PAC Learning

Concept of learnability, Probably Approximately Correct (PAC) learning framework, Sample complexity analysis, Vapnik–Chervonenkis (VC) dimension, Growth function and shattering, Generalization bounds and uniform convergence.

UNIT - IV: Kernel Methods and Support Vector Machines

Linear classifiers and margin theory, Maximum margin principle, Support Vector Machines (SVMs): hard and soft margin, Kernel trick and kernel functions, Dual formulation and optimization.

UNIT - V: Model Selection, Regularization, and Generalization

Structural Risk Minimization (SRM), Regularization techniques (L1, L2, Elastic Net), Cross-validation methods, Model selection criteria (AIC, BIC), Generalization theory and bounds, Stability and consistency of learning algorithms

Textbooks

1. Trevor Hastie, Robert Tibshirani, and Jerome Friedman. 2009. The Elements of Statistical Learning (2nd ed.). Springer, New York, NY.
2. Christopher M. Bishop. 2006. Pattern Recognition and Machine Learning. Springer, New York, NY.
3. Shai Shalev-Shwartz and Shai Ben-David. 2014. Understanding Machine Learning: From Theory to Algorithms. Cambridge University Press, Cambridge, UK.

Reference Books

1. Vladimir N. Vapnik. 1998. Statistical Learning Theory. Wiley, New York, NY.
2. Kevin P. Murphy. 2012. Machine Learning: A Probabilistic Perspective. MIT Press, Cambridge, MA.
3. Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. 2018. Foundations of Machine Learning (2nd ed.). MIT Press, Cambridge, MA.

Reference Links

1. NPTEL – Courses on Machine Learning and Statistical Learning [https://nptel.ac.in]
2. MIT OpenCourseWare – Machine Learning & Statistical Learning Theory lectures [https://ocw.mit.edu]
3. arXiv – Research papers on Statistical Learning Theory [https://arxiv.org]

Probabilistic Graphical Models

(Program Elective - II)

Subject Code: 26MAI1E24

L	T	P	C
3	0	0	3

Course Objectives

1. To develop a strong foundation in probabilistic reasoning and uncertainty modeling in AI systems.
2. To understand the representation of complex distributions using graphical structures such as Bayesian and Markov networks.
3. To analyze and implement exact and approximate inference algorithms in graphical models.
4. To explore parameter and structure learning techniques for probabilistic graphical models.
5. To apply graphical models to real-world problems in machine learning, computer vision, natural language processing, and bioinformatics.

Course Outcomes

Upon successful completion of the course, students will be able to:

1. Apply principles of probability theory and graph theory to model uncertain systems.
2. Analyze and construct Bayesian Networks and Markov Random Fields for complex domains.
3. Evaluate exact and approximate inference algorithms for efficiency and scalability.
4. Design learning algorithms for parameter estimation and structure discovery in graphical models.
5. Implement probabilistic graphical models for real-world AI/ML applications.

UNIT - I: Foundations of Probability and Graph Theory

Review of probability theory: random variables, joint/conditional distributions, independence. Bayes theorem and probabilistic inference, Basics of graph theory: directed and undirected graphs. Representation of distributions using graphs, Conditional independence and d-separation.

UNIT - II: Bayesian Networks

Structure and semantics of Bayesian Networks (BNs), Factorization and joint distribution representation, Conditional probability tables (CPTs), Independence assumptions and d-separation in BNs, Exact inference in Bayesian Networks (Variable Elimination, Enumeration).

UNIT - III: Markov Random Fields (MRFs)

Undirected graphical models and Markov properties, Factor graphs and clique potentials, Gibbs distribution and Hammersley-Clifford theorem, Partition function and normalization, Applications of MRFs in vision and spatial modeling.

UNIT - IV: Inference Algorithms

Exact inference: Variable Elimination, Junction Tree Algorithm. Approximate inference: Sampling methods (Monte Carlo, Gibbs Sampling), Variational inference techniques. Belief Propagation (Sum-Product and Max-Product algorithms), Complexity and scalability issues

UNIT - V: Learning and Applications of Graphical Models

Parameter learning: Maximum Likelihood Estimation (MLE), MAP estimation. Structure learning: Score-based and constraint-based approaches. Expectation-Maximization (EM) algorithm. Applications in: Natural Language Processing, Computer Vision, Speech Recognition, Bioinformatics. Introduction to Deep Probabilistic Models (brief overview).

Textbooks

1. Daphne Koller and Nir Friedman. 2009. Probabilistic Graphical Models: Principles and Techniques. MIT Press, Cambridge, MA.
2. Christopher M. Bishop. 2006. Pattern Recognition and Machine Learning. Springer, New York, NY.
3. Kevin P. Murphy. 2012. Machine Learning: A Probabilistic Perspective. MIT Press, Cambridge, MA.

Reference Books

1. David Barber. 2012. Bayesian Reasoning and Machine Learning. Cambridge University Press, Cambridge, UK.
2. Michael I. Jordan, ed. 1998. Learning in Graphical Models. MIT Press, Cambridge, MA.
3. Martin J. Wainwright and Michael I. Jordan. 2008. Graphical Models, Exponential Families, and Variational Inference. Now Publishers Inc., Hanover, MA

Reference Links

1. NPTEL Course on Probabilistic Graphical Models [https://nptel.ac.in]
2. MIT OpenCourseWare – Probabilistic Systems Analysis / Machine Learning Courses [https://ocw.mit.edu]
3. Stanford CS228: Probabilistic Graphical Models [https://cs228.stanford.edu]

Machine Learning Lab

Subject Code: 26MAI1101

L	T	P	C
0	0	4	2

Course Objectives

1. To provide hands-on experience in implementing machine learning algorithms using modern tools and libraries.
2. To enable students to preprocess, analyze, and visualize real-world datasets effectively.
3. To develop skills in building, training, and optimizing machine learning models.
4. To understand model evaluation techniques and performance metrics through experimentation.
5. To expose students to advanced ML techniques including deep learning and reinforcement learning frameworks.

Course Outcomes

After completion of the lab, students will be able to:

1. Implement various machine learning algorithms using programming tools such as Python and libraries.
2. Analyze datasets through preprocessing, feature engineering, and visualization techniques.
3. Evaluate machine learning models using appropriate metrics and validation strategies.
4. Design end-to-end machine learning pipelines for real-world applications.
5. Apply advanced ML techniques such as neural networks and clustering to solve complex problems.

List of Experiments

Experiment 1: Data Preprocessing and Visualization

- Handling missing values, normalization, encoding
- Data visualization using plots (histograms, scatter plots)

Experiment 2: Exploratory Data Analysis (EDA)

- Statistical summaries
- Correlation analysis
- Feature selection techniques

Experiment 3: Implementation of Linear Regression

- Simple and multiple regression
- Model evaluation using MSE, RMSE

Experiment 4: Logistic Regression for Classification

- Binary classification
- Confusion matrix, accuracy, precision, recall

Experiment 5: k-Nearest Neighbors (k-NN)

- Distance metrics
- Model comparison with varying k values

Experiment 6: Decision Trees and Random Forest

- Tree construction and visualization
- Ensemble learning with Random Forest

Experiment 7: Support Vector Machines (SVM)

- Kernel functions (linear, polynomial, RBF)
- Hyperparameter tuning

Experiment 8: Clustering using K-Means

- Cluster formation and visualization
- Elbow method for optimal k

Experiment 9: Hierarchical Clustering

- Agglomerative clustering
- Dendrogram analysis

Experiment 10: Dimensionality Reduction using PCA

- Feature transformation
- Visualization in reduced dimensions

Experiment 11: Neural Networks using Deep Learning Frameworks

- Implementation using TensorFlow or PyTorch
- Training a simple feedforward neural network

Experiment 12: Reinforcement Learning Basics

- Implementation of Q-learning
- Simple environment simulation (e.g., grid world)

Textbooks

1. Aurélien Géron. 2022. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (3rd ed.). O'Reilly Media, Sebastopol, CA.
2. Sebastian Raschka and Vahid Mirjalili. 2022. Python Machine Learning (4th ed.). Packt Publishing, Birmingham, UK.
3. Kevin P. Murphy. 2012. Machine Learning: A Probabilistic Perspective. MIT Press, Cambridge, MA.

Reference Books

1. Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. Deep Learning. MIT Press, Cambridge, MA.
2. Christopher M. Bishop. 2006. Pattern Recognition and Machine Learning. Springer, New York, NY.
3. Andreas C. Müller and Sarah Guido. 2016. Introduction to Machine Learning with Python. O'Reilly Media, Sebastopol, CA.

Reference Links

1. NPTEL – Machine Learning Lab / Practical Courses [https://nptel.ac.in]
2. TensorFlow Official Tutorials [https://www.tensorflow.org/tutorials]
3. Scikit-learn Documentation [https://scikit-learn.org/stable/tutorial/index.html]

Data Structures and Algorithms for AI Lab

Subject Code: 26MAI1102

L	T	P	C
0	0	4	2

Course Objectives

1. To provide hands-on experience in implementing advanced data structures and algorithms.
2. To enable students to analyze and compare algorithm performance using practical experiments.
3. To develop problem-solving skills using algorithmic design techniques for AI applications.
4. To implement graph, tree, and string processing algorithms used in intelligent systems.
5. To bridge theoretical concepts with real-world AI use cases through programming exercises.

Course Outcomes

After completion of the lab, students will be able to:

1. Implement advanced data structures such as heaps, AVL trees, tries, and hash tables.
2. Apply algorithm design techniques like greedy, dynamic programming, and backtracking to solve problems.
3. Analyze and compare time and space complexity of implemented algorithms.
4. Develop graph-based solutions for AI problems such as pathfinding and network analysis.
5. Build small-scale AI-oriented applications using data structures and algorithms.

List of Experiments

1. Implementation of Advanced Trees
    - AVL Tree and Red-Black Tree operations (insertion, deletion, traversal).

2. Heap and Priority Queue Implementation
    - Binary Heap operations and priority queue applications.

3. Hashing Techniques
    - Implement hash tables with collision handling (chaining and open addressing).

4. Trie Implementation for String Processing
    - Build autocomplete and dictionary search system.

5. Suffix Tree / Pattern Matching
    - Implement KMP or Rabin-Karp algorithm for pattern matching.

6. Divide and Conquer Algorithms
    - Implement Merge Sort and Quick Sort with performance comparison.

7. Greedy Algorithms
    - Huffman Coding and Activity Selection problems.

8. Dynamic Programming Problems
    - Knapsack problem and Longest Common Subsequence.

9. Backtracking and Branch & Bound
    - N-Queens problem and Travelling Salesman Problem (TSP).

10. Graph Traversal Algorithms
    - BFS and DFS implementations.

11. Shortest Path Algorithms
    - Dijkstra and Bellman-Ford algorithms.

12. AI-Based Graph Application
    - Implement A^* search for pathfinding or a simple recommendation system using graph concepts.

Textbooks

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. 2022. Introduction to Algorithms (4th ed.). MIT Press, Cambridge, MA.
2. Mark Allen Weiss. 2014. Data Structures and Algorithm Analysis in C++ (4th ed.). Pearson, Boston, MA.
3. Robert Sedgewick and Kevin Wayne. 2011. Algorithms (4th ed.). Addison-Wesley, Boston, MA.

Reference Books

1. Steven S. Skiena. 2020. The Algorithm Design Manual (3rd ed.). Springer, Cham, Switzerland.
2. Ellis Horowitz and Sartaj Sahni. 2008. Fundamentals of Data Structures. Universities Press, Hyderabad, India.
3. Aditya Y. Bhargava. 2016. Grokking Algorithms: An Illustrated Guide for Programmers and Other Curious People. Manning Publications, Shelter Island, NY.

Reference Links

1. MIT OpenCourseWare – Algorithms Lab Resources [https://ocw.mit.edu/courses/6-006-introduction-to-algorithms/]
2. VisuAlgo (Interactive Learning)[https://visualgo.net/en]
3. GeeksforGeeks – Practice Portal [https://www.geeksforgeeks.org/data-structures/]

English for Research Paper Writing
(Audit Course)

Subject Code: 26MAC1001

L	T	P	C
2	0	0	0

Course Objectives

1. To develop students' ability to plan and organize academic writing by mastering word order, sentence structure, paragraph development, conciseness, and techniques for avoiding ambiguity and redundancy.
2. To enable learners to present ideas clearly and responsibly through skills such as clarifying agency (who did what), highlighting findings, using hedging and criticism appropriately, and avoiding plagiarism through effective paraphrasing.
3. To train students in the structure and components of a research paper by understanding and producing well-organized abstracts, introductions, literature reviews, methods, results, discussions, and conclusions.
4. To enhance students' competence in writing key academic sections—including titles, abstracts, introductions, and literature reviews—using appropriate academic style, coherence, and logical progression.
5. To equip learners with practical skills for drafting, revising, and finalizing research papers by applying section-specific writing strategies and conducting a thorough final check for clarity, accuracy, and academic integrity.

Course Outcomes

1. Students will be able to plan and organize academic texts effectively by applying correct word order, clear sentence and paragraph structure, and techniques for conciseness while avoiding ambiguity and redundancy.
2. Students will be able to present information clearly and ethically by clarifying agency (who did what), highlighting key findings, using appropriate hedging and criticism, and avoiding plagiarism through accurate paraphrasing.
3. Students will be able to write and evaluate major sections of a research paper such as literature reviews, methods, results, discussions, conclusions, and perform a final check for coherence and accuracy.
4. Students will be able to construct effective titles, abstracts, and introductions and develop logically organized reviews of literature using accepted academic writing conventions.
5. Students will be able to apply section-specific writing skills to produce well-structured methods, results, discussion, and conclusion sections that meet academic and research communication standards.

UNIT – I:

Planning and preparation, Word order, Breaking up long sentences, Structuring paragraphs and sentences. Being concise and removing redundancy, Avoiding ambiguity and vagueness

UNIT - II:

Clarifying who did what, Highlighting findings. Hedging and criticizing. Paraphrasing and plagiarism, Sections of a paper, Abstracts, Introduction

UNIT - III:

Review of the Literature. Methods, Results, Discussion, Conclusions, Final check

UNIT - IV:
Key skills needed when writing a title. An abstract, An introduction and a review of the Literature

UNIT - V:
Skills needed when writing methods, Results, Discussions and Conclusion.

Text Books

1. Goldbort R (2006). Writing for Science. Yale University Press.
2. Day R (2006). How to Write and Publish a Scientific Paper. Cambridge University Press.
3. Highman N (1998). Handbook of Writing for the Mathematical Sciences. SIAM. Highman's book.

Disaster Management
(Audit Course)

Subject Code: 26MAC1002

L	T	P	C
2	0	0	0

Course Objectives:

The primary objectives of this course are:

1. To provide basic concepts and terminologies related to hazards and disasters.
2. To highlight the critical importance and role of disaster management in the contemporary world.
3. To enhance awareness of institutional processes, policies, and management strategies used to mitigate disaster impacts.
4. To analyze the various dimensions of vulnerability and their contribution to disaster risk.
5. To equip knowledge regarding emergency management for manmade and technological threats.

Course Outcomes:

Upon successful completion of this course, students will be able to:

1. Define and classify different types of disasters and distinguish them from hazards.
2. Evaluate the nature of vulnerability and assess the impacts of disasters on different communities.
3. Explain the components of the Disaster Management Cycle, including mitigation, preparedness, response, and recovery.
4. Discuss the institutional frameworks and legislations available for disaster risk management at national and international level.
5. Identify specific threats related to industrial accidents and security, and understand the necessary emergency response mechanisms.

UNIT - I: Introduction and Basic Concepts: Understanding Fundamental Concepts: Definitions of Hazards and Disasters. Classification: Disaster types and causes including Geophysical, Hydrological, Meteorological, Biological, Atmospheric, and Human-made. Distinguishing between natural hazards and manmade disasters.

UNIT - II: Dimensions of Disasters and Vulnerability: Global Trends: Analysis of global trends in disasters. Impact analysis: Physical, Social, Economic, Political, Environmental, and Psychosocial impacts of disasters. Understanding Vulnerability: Defining and analyzing Physical Vulnerability, Economic Vulnerability, and Social Vulnerability.

UNIT - III: The Disaster Management Cycle and Risk Reduction: Detailed study of the components of DM Cycle: Response and recovery, Risk assessment, Mitigation and prevention, Preparedness planning, Prediction and warning. Disaster Risk Reduction (DRR): Concepts of DRR and Community based disaster risk reduction strategies.

UNIT - IV: Institutional Frameworks and Initiatives: National Context: Disaster Risk Management in India; related policies, plans, programs, and legislation. Global Context: International strategies for disaster reduction and other global initiatives. Agencies: Role of agencies NIDM, NDRF.

UNIT - V: Emergency Management and Emerging Threats: Technological and Industrial Disasters: Explosions, Industrial accidents, Nuclear, Transport, and Mining accidents. Environmental Emergencies: Spills (Oil and Hazardous material). Security Threats: Bomb threats, terrorist attacks, stampedes, and conflict situations.

Textbooks:

1. Sharma, S.C., Disaster Management, Khanna Book Publishing. Latest Edition
2. Singh, R. B. (ed.), Natural Hazards and Disaster Management: Vulnerability and Mitigation, Rawat Publications, New Delhi. Latest Edition
3. Modh, S., Managing Natural Disaster: Hydrological, Marine and Geological Disasters, Macmillan, Delhi. Latest Edition

Reference Books:

1. Carter, N., Disaster Management: A Disaster Management Handbook, Asian Development Bank, Manila. Latest Edition
2. Clements, B. W., Disasters and Public Health: Planning and Response, Elsevier Inc. Latest Edition
3. Dunkan, K., and Brebbia, C. A. (Eds.), Disaster Management and Human Health Risk: Reducing Risk, Improving Outcomes, WIT Press, UK. Latest Edition
4. Ramkumar, Mu, Geological Hazards: Causes, Consequences and Methods of Containment, New India Publishing Agency, New Delhi. Latest Edition

Constitution of India
(Audit Course)

Subject Code: 26MAC1003

L	T	P	C
2	0	0	0

Course Objectives:

1. To provide students with a comprehensive understanding of the historical background and salient features of the Indian Constitution, including Fundamental Rights, Duties, and Directive Principles of State Policy.
2. To familiarize learners with the structure and functioning of local self-government institutions and the importance of grassroots democracy in India.
3. To develop awareness of the parliamentary system of government in India, including the roles and powers of the President, Prime Minister, Cabinet, and Judiciary.
4. To explain the nature of Indian federalism and the legislative procedures and privileges of Union and State legislatures.
5. To enable students to understand the role and functioning of constitutional and statutory bodies such as parliamentary committees, Election Commission, CAG, Finance Commission, NITI Aayog, and political parties.

Course Outcomes:

1. Students will be able to explain the origin, structure, and key features of the Indian Constitution, including Fundamental Rights, Fundamental Duties, and Directive Principles.
2. Students will be able to describe the organization and functions of local administration, including Panchayati Raj Institutions and municipal bodies, and appreciate the value of participatory democracy.
3. Students will be able to analyze the working of the parliamentary system and the emergency provisions, and understand the relationship between the executive, legislature, and judiciary.
4. Students will be able to interpret the principles of Indian federalism and explain Union–State relations and the legislative process in Parliament and State Legislatures.
5. Students will be able to evaluate the role of constitutional and non-constitutional bodies and recognize their contribution to democratic governance and accountability in India.

UNIT - I:

Introduction: Historical perspective of the constitution of India -Salient features of The Indian Constitution-Features: Fundamental Rights (Article 12 to 35), Duties (31 A-1976 emergency) and Directive principles (Article 30 to 51) of State Policy - Articles 14 to 18 Articles 19-Article 21- Amendment Procedure of The Indian Constitution: 42nd amendment (Mini Constitution)-44th amendment (1978-Janatha Govt.)

UNIT - II:

Local Administration: District's administration head: Role and Importance, Municipalities Introduction, Mayer and role of elected representative, CEO of municipal corporation, Pachayati raj: Introduction, PRI: ZilaPachayat, Elected officials and their roles, CEO Zila Pachayat: Position and role, Block level: Organizational Hierarchy (Different departments), Village level: Role of elected and appointed officials, Importance of grass root democracy.

UNIT - III:

Parliamentary form of Govt. In India: President of India- Emergency provisions- National Emergency Article 352 President Rules- Article 356 Financial Emergency- Article 360 Prime Minister and Cabinet-Supreme Court of India (Indian Judiciar)

UNIT - IV:

Indian Federalism: Union State relations, Legislative, Administrative and Financial relations. Lok Sabha, Rajya Sabha, Vidhan Sabha & Vidhan Parishad-Composition, Speaker, Chairman, Privileges, Legislative procedure.

UNIT - V:

Parliamentary Committees: Public Accounts Committee- Estimates Committee- Committee on Public Undertakings- Election commission of India (Article -324) -Comptroller and Auditor General (CAG) of India (Article 148 to 150) Finance Commission (Article-280)-Neethi Aayog (Planning Commission) and Political Parties.

Text Books

1. DD Basu-Indian Constitution
2. Dr. D. Surannaida - Indian Politica ISystem.
3. Madhav Khosla- The Indian Constitution

Reference Books

1. The Constitution of India. 1950 (Bare Act), Government Publication
2. M. P. Jain, Indian Constitution Law, 7th Edn, Lexis Nexis, 2014

Deep Learning

Subject Code: 26MAI1004

L	T	P	C
4	0	0	4

Course Objectives:

1. Provide a strong foundation in artificial neural networks and deep learning principles for intelligent system design.
2. Enable students to understand and implement various deep learning architectures such as CNNs, RNNs, and Transformers.
3. Introduce optimization algorithms, regularization methods, and hyper-parameter tuning techniques for efficient model training.
4. Develop the ability to apply representation learning techniques for image, speech, text, and sequential data processing.
5. Familiarize students with advanced deep learning applications and industrial use cases in AI-driven systems.

Course Outcomes:

Upon successful completion of the course, students will be able to:

1. Analyze and design artificial neural network models for solving complex computational problems.
2. Apply optimization and regularization techniques to improve deep learning model performance and generalization.
3. Develop and evaluate Convolutional Neural Network models for computer vision applications.
4. Implement sequence modeling techniques using RNN, LSTM, and GRU architectures for NLP and time-series tasks.
5. Explore and utilize advanced deep learning frameworks such as Auto-encoders, GANs, Transformers, and Reinforcement Learning models for real-world AI applications.

UNIT - I: Foundations of Deep Learning and Neural Networks

Introduction to Artificial Intelligence, Machine Learning, and Deep Learning, Evolution and applications of Deep Learning, Biological neuron and Artificial Neuron models, Perceptron Learning Algorithm, Limitations of Single Layer Perceptron, Multilayer Feed Forward Neural Networks, Activation Functions, Loss Functions, Forward Propagation, Backpropagation Algorithm, Gradient Descent Learning, Network Architectures and Training Process, Introduction to Deep Learning Frameworks.

UNIT - II: Optimization and Regularization Techniques

Optimization in Deep Learning, Cost Functions, Batch Gradient Descent, Stochastic Gradient Descent, Mini-Batch Gradient Descent, Momentum, Nesterov Accelerated Gradient, Adaptive Optimization Algorithms: AdaGrad, RMSProp, Adam, Learning Rate Scheduling, Weight Initialization Techniques, Vanishing and Exploding Gradient Problems, Regularization Methods: L1/L2 Regularization, Dropout, Batch Normalization, Early Stopping, Hyperparameter Tuning Techniques, Model Evaluation Metrics and Cross Validation.

UNIT - III: Convolutional Neural Networks and Computer Vision

Introduction to Computer Vision, Convolution Operation and Feature Maps, CNN Architecture, Pooling Layers, Padding and Stride, CNN Training and Visualization, Popular CNN Architectures: LeNet, AlexNet, VGGNet, GoogLeNet, ResNet, Transfer Learning and Fine Tuning, Image Classification, Object Detection, Face Recognition, Medical Image Analysis, Data Augmentation Techniques, Applications of CNNs in Industry.

UNIT - IV: Recurrent Neural Networks and Sequence Modeling
Sequential Data Processing, Recurrent Neural Networks (RNNs), Backpropagation Through Time (BPTT), Vanishing Gradient Problem in RNNs, Long Short-Term Memory (LSTM) Networks, Gated Recurrent Unit (GRU), Bidirectional RNNs, Encoder-Decoder Models, Attention Mechanisms, Natural Language Processing Applications, Text Classification, Sentiment Analysis, Language Modeling, Machine Translation, Speech Recognition, and Time-Series Forecasting.

UNIT - V: Advanced Deep Learning Architectures and Applications
Representation Learning, Auto encoders and Variational Auto encoders, Generative Adversarial Networks (GANs), Self-Supervised and Unsupervised Deep Learning, Transformers and Large Language Models, Deep Reinforcement Learning Concepts, Deep Q Networks (DQN), Industrial Applications of Deep Learning in Healthcare, Robotics, Autonomous Vehicles, Cybersecurity, Finance, and Smart Systems, Ethical Issues, Explainable AI, Bias, Fairness, and Future Trends in Deep Learning.

Textbooks
1. Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. Deep Learning. MIT Press, Cambridge, MA.
2. Charu C. Aggarwal. 2018. Neural Networks and Deep Learning. Springer, Cham, Switzerland.
3. François Chollet. 2021. Deep Learning with Python (2nd ed.). Manning Publications, Shelter Island, NY.

Reference Books
1. Christopher M. Bishop. 2006. Pattern Recognition and Machine Learning. Springer, New York, NY.
2. Aurélien Géron. 2022. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (3rd ed.). O’Reilly Media, Sebastopol, CA.
3. Kevin P. Murphy. 2012. Machine Learning: A Probabilistic Perspective. MIT Press, Cambridge, MA.

Reference Links
1. https://www.tensorflow.org
2. https://pytorch.org/
3. https://www.coursera.org/specializations/deep-learning

Natural Language Processing

Subject Code: 26MAI1005

L	T	P	C
4	0	0	4

Course Objectives

1. Understand the fundamental concepts of Natural Language Processing, linguistic structures, and text preprocessing techniques used in intelligent language systems.
2. Analyze and apply syntactic and semantic processing methods including parsing, part-of-speech tagging, and semantic representation for natural language understanding.
3. Develop statistical and machine learning-based NLP models using probabilistic methods, sequence models, and deep learning architectures.
4. Design and implement modern NLP applications such as sentiment analysis, machine translation, information extraction, and conversational AI systems.
5. Evaluate advanced NLP systems using Transformer architectures and Large Language Models (LLMs) while understanding ethical, fairness, and bias-related issues in language AI.

Course Outcomes

Upon successful completion of the course, the student will be able to:

1. Apply linguistic and text preprocessing techniques for effective preparation and representation of textual data in NLP systems.
2. Analyze syntactic structures and implement parsing and tagging techniques using probabilistic and rule-based approaches.
3. Design semantic analysis models for information extraction, named entity recognition, and semantic representation tasks.
4. Develop machine learning and deep learning-based NLP models using statistical methods, embeddings, recurrent networks, and Transformer architectures.
5. Build and evaluate advanced NLP applications including chatbots, machine translation systems, sentiment analysis tools, and LLM-based solutions while addressing ethical considerations.

UNIT - I: Introduction to Natural Language Processing

Introduction to NLP: History, scope, challenges, and applications of NLP. Human languages and linguistic fundamentals: phonology, morphology, syntax, semantics, pragmatics. Text preprocessing techniques: sentence segmentation, tokenization, stop-word removal, normalization, stemming, lemmatization. Regular expressions and pattern matching. Feature extraction techniques and vector space representation of text. Introduction to corpus linguistics and NLP toolkits.

UNIT - II: Syntax and Parsing

Part-of-Speech (POS) tagging: rule-based, stochastic, and transformation-based tagging approaches. Context-Free Grammars (CFG) and syntactic parsing. Top-down and bottom-up parsing techniques. Dependency grammar and dependency parsing. Probabilistic parsing methods: probabilistic CFGs, CYK parsing algorithm. Chunking and shallow parsing techniques. Applications of syntactic analysis in NLP systems.

UNIT - III: Semantic Analysis and Representation

Semantic analysis and meaning representation in NLP. Lexical semantics and semantic relations. Word Sense Disambiguation (WSD).

Semantic networks, frames, and ontologies. Information extraction and relation extraction techniques. Named Entity Recognition (NER). Semantic role labeling and knowledge representation approaches. Applications in search engines and intelligent information retrieval systems.

UNIT - IV: Statistical and Machine Learning Approaches for NLP
Statistical language models and n-gram models. Hidden Markov Models (HMM) and Conditional Random Fields (CRF). Text classification and sequence labeling techniques. Word embeddings: Word2Vec, GloVe, FastText. Neural NLP architectures: RNN, LSTM, GRU. Attention mechanisms and sequence-to-sequence learning. Transformer architecture, BERT, GPT, and pre-trained language models. Model evaluation metrics for NLP systems.

UNIT - V: Advanced NLP Applications
Machine Translation systems and multilingual NLP. Question Answering systems and dialogue management. Sentiment analysis and opinion mining. Text summarization and document generation. Conversational AI and chatbot development. Large Language Models (LLMs): fundamentals, prompt engineering, fine-tuning concepts, retrieval-augmented generation (RAG). Ethical issues in NLP: bias, fairness, explainability, privacy, misinformation, and responsible AI practices. Industrial and research applications of NLP in healthcare, finance, education, and social media analytics.

Textbooks

1. Daniel Jurafsky and James H. Martin. 2025. Speech and Language Processing (3rd ed. draft). Pearson, Hoboken, NJ.
2. Jacob Eisenstein. 2019. Introduction to Natural Language Processing. MIT Press, Cambridge, MA.
3. Delip Rao and Brian McMahan. 2019. Natural Language Processing with PyTorch. O’Reilly Media, Sebastopol, CA.

Reference Books

1. Christopher D. Manning and Hinrich Schütze. 1999. Foundations of Statistical Natural Language Processing. MIT Press, Cambridge, MA.
2. Yoav Goldberg. 2017. Neural Network Methods for Natural Language Processing. Morgan & Claypool Publishers, San Rafael, CA.
3. Denis Rothman. 2024. Transformers for Natural Language Processing (3rd ed.). Packt Publishing, Birmingham, UK.

Reference Links

1. https://nptel.ac.in/courses/106106211
2. https://web.stanford.edu/class/cs224n
3. https://huggingface.co/docs/transformers/index

Computer Vision

Subject Code: 26MAI1006

L	T	P	C
4	0	0	4

Course Objectives

The course is designed to enable students to:

1. Understand the fundamentals of digital image formation, camera models, and geometric representations used in computer vision systems.
2. Apply image processing and enhancement techniques for analyzing and improving visual data.
3. Analyze and implement feature extraction, motion analysis, and object tracking algorithms for vision-based applications.
4. Design and develop machine learning and deep learning models for image classification, object detection, and recognition tasks.
5. Explore advanced computer vision applications in healthcare, autonomous systems, surveillance, and generative AI-based visual computing.

Course Outcomes

Upon successful completion of the course, students will be able to:

1. Analyze image formation models, camera geometry, and visual representation techniques used in computer vision systems.
2. Apply filtering, segmentation, and morphological operations to solve image processing problems.
3. Implement feature detection, extraction, and motion analysis algorithms for object recognition and tracking applications.
4. Design and evaluate deep learning-based computer vision models using CNNs, transfer learning, and object detection frameworks.
5. Develop AI-driven computer vision solutions for real-world applications such as medical imaging, autonomous driving, surveillance, and intelligent video analytics.

UNIT - I: Introduction to Computer Vision and Image Formation

Introduction to Computer Vision, Human Vision vs Computer Vision, Applications of Computer Vision, Digital Image Fundamentals, Image Sampling and Quantization, Color Models and Color Spaces, Image Formation Process, Camera Models, Pinhole Camera Geometry, Perspective Projection, Coordinate Transformations, Homogeneous Coordinates, Stereo Vision Basics, Epipolar Geometry, Camera Calibration Concepts.

UNIT - II: Image Processing Techniques

Image Enhancement and Restoration, Spatial Domain Processing, Frequency Domain Processing, Image Filtering Techniques, Smoothing and Sharpening Filters, Gaussian and Median Filters, Edge Detection Methods (Sobel, Prewitt, Canny), Thresholding Techniques, Image Segmentation, Region-based and Clustering-based Segmentation, Morphological Operations, Image Compression Basics, Noise Reduction and Feature-preserving Smoothing Techniques.

UNIT - III: Feature Extraction and Motion Analysis

Feature Detection and Description, Corner Detection, Harris and FAST Features, Scale Invariant Feature Transform (SIFT), SURF Features, Histogram of Oriented Gradients (HOG), Feature Matching Techniques, Optical Flow, Motion Estimation, Background Subtraction, Object Tracking Algorithms, Kalman Filter and Particle Filter Basics, Activity Recognition Fundamentals, Applications in Video Surveillance and Robotics.

UNIT - IV: Machine Learning and Deep Learning for Vision
Introduction to Machine Learning in Vision, Image Classification Techniques, Neural Network Fundamentals, Convolutional Neural Networks (CNNs), Deep CNN Architectures (LeNet, AlexNet, VGG, ResNet), Transfer Learning, Fine Tuning, Object Detection Models (R-CNN, Fast R-CNN, YOLO, SSD), Semantic Segmentation, Instance Segmentation, Image Captioning Basics, Performance Evaluation Metrics for Vision Systems.

UNIT - V: Advanced Computer Vision Applications
Face Detection and Recognition, Biometric Vision Systems, Medical Image Analysis, Vision-based Autonomous Vehicles, Lane Detection and Traffic Analysis, Video Analytics and Intelligent Surveillance, Human Pose Estimation, Augmented Reality and Virtual Reality Vision Systems, Vision Transformers (ViT), Generative Vision Models, GANs and Diffusion Models, Ethical Issues and Challenges in AI-driven Vision Systems.

Textbooks

1. Richard Szeliski. 2022. Computer Vision: Algorithms and Applications (2nd ed.). Springer, Cham, Switzerland.
2. Rafael C. Gonzalez and Richard E. Woods. 2018. Digital Image Processing (4th ed.). Pearson, Hoboken, NJ.
3. Simon J. D. Prince. 2024. Computer Vision: Models, Learning, and Inference (2nd ed.). Cambridge University Press, Cambridge, UK.

Reference Textbooks

1. David A. Forsyth and Jean Ponce. 2011. Computer Vision: A Modern Approach (2nd ed.). Pearson, Boston, MA.
2. Mark Nixon and Alberto Aguado. 2019. Feature Extraction and Image Processing for Computer Vision (4th ed.). Academic Press, London, UK.
3. Christopher M. Bishop and Hugh Bishop. 2024. Deep Learning: Foundations and Concepts. Springer, Cham, Switzerland.

Reference Links

1. https://nptel.ac.in/courses/106105216
2. https://cs231n.stanford.edu/
3. https://docs.opencv.org/

Reinforcement Learning

(Program Elective - III)

Subject Code: 26MAI1E31

L	T	P	C
3	0	0	3

Course Objectives:

1. Understand the theoretical foundations of reinforcement learning, including agent–environment interaction and Markov Decision Processes (MDPs).
2. Analyze and apply dynamic programming and model-free reinforcement learning techniques for sequential decision-making problems.
3. Design and implement temporal difference learning algorithms such as SARSA and Q-Learning for optimal policy generation.
4. Explore deep reinforcement learning architectures using neural networks, policy optimization, and actor–critic methods.
5. Develop intelligent autonomous systems capable of learning adaptive behavior in real-world AI applications such as robotics, gaming, and recommendation systems.

Course Outcomes:

Upon successful completion of the course, students will be able to:

1. Model sequential decision-making problems using Markov Decision Processes and Bellman optimality principles.
2. Apply dynamic programming, Monte Carlo, and temporal difference methods to solve reinforcement learning problems.
3. Design and implement reinforcement learning algorithms including SARSA, Q-Learning, and policy-based methods.
4. Evaluate and optimize deep reinforcement learning architectures for complex and high-dimensional environments.
5. Analyze and develop reinforcement learning solutions for real-world intelligent systems and industrial AI applications.

UNIT - I: Foundations of Reinforcement Learning

Introduction to Reinforcement Learning (RL); Supervised vs Unsupervised vs Reinforcement Learning; Agent–Environment Interaction; Rewards and Returns; Markov Property; Markov Decision Processes (MDP); State Space, Action Space, Reward Function; Bellman Equations; Finite Horizon and Infinite Horizon Problems; Optimal Policies and Value Functions; Applications of RL.

UNIT - II: Dynamic Programming and Monte Carlo Methods

Dynamic Programming Fundamentals; Policy Evaluation; Policy Improvement; Policy Iteration; Value Iteration; Generalized Policy Iteration (GPI); Monte Carlo Prediction and Control; Exploring Starts; Incremental Monte Carlo Methods; On-policy and Off-policy Learning; Importance Sampling Techniques.

UNIT - III: Temporal Difference Learning

Temporal Difference (TD) Learning; TD Prediction; TD(0) Algorithm; SARSA Algorithm; Q-Learning Algorithm; Expected SARSA; Eligibility Traces; TD(λ); Exploration vs Exploitation Trade-off; ε-greedy Policies; Softmax Action Selection; Convergence Properties of TD Methods.

UNIT - IV: Deep Reinforcement Learning
Function Approximation in RL; Linear Function Approximation; Neural Network-based Value Function Approximation; Deep Q-Networks (DQN); Experience Replay; Target Networks; Double DQN and Dueling Networks; Policy Gradient Methods; REINFORCE Algorithm; Actor–Critic Architectures; Advantage Functions; Deep RL Frameworks and Toolkits.

UNIT - V: Advanced Topics and Applications
Advanced Reinforcement Learning Techniques; Multi-Agent Reinforcement Learning; Model-Based Reinforcement Learning; Hierarchical Reinforcement Learning; Inverse Reinforcement Learning; Reinforcement Learning in Robotics and Autonomous Systems; RL for Game Playing; Recommendation Systems; Industrial Automation; Healthcare and Finance Applications; Ethical Challenges and Future Directions in RL.

Textbooks

1. Richard S. Sutton and Andrew G. Barto. 2018. Reinforcement Learning: An Introduction (2nd ed.). MIT Press, Cambridge, MA.
2. Csaba Szepesvári. 2010. Algorithms for Reinforcement Learning. Morgan & Claypool Publishers, San Rafael, CA.
3. Marco Wiering and Martijn van Otterlo, eds. 2012. Reinforcement Learning: State-of-the-Art. Springer, Berlin, Germany.

Reference Textbooks

1. Dimitri P. Bertsekas. 2019. Reinforcement Learning and Optimal Control. Athena Scientific, Belmont, MA.
2. Shangtong Zhang and Richard S. Sutton. 2020. Reinforcement Learning: An Introduction (Code Companion). MIT Press, Cambridge, MA.
3. Phil Winder. 2020. Reinforcement Learning: Industrial Applications of Intelligent Agents. O’Reilly Media, Sebastopol, CA.

Reference Links

1. https://ocw.mit.edu
2. https://online.stanford.edu
3. https://deepmind.google/research/

Speech and Audio Processing
(Program Elective - III)

Subject Code: 26MAI1E32

L	T	P	C
3	0	0	3

Course Objectives
1. Understand the fundamentals of speech production, audio signal representation, and digital speech processing techniques.
2. Analyze and extract significant speech and audio features using classical and modern signal processing approaches.
3. Apply statistical and machine learning techniques for automatic speech recognition and audio classification systems.
4. Design and evaluate speech synthesis, speaker recognition, and emotion-aware speech applications.
5. Explore deep learning architectures and recent AI-driven advancements in speech and audio processing for real-world applications.

Course Outcomes
Upon successful completion of the course, students will be able to:
1. Analyze speech and audio signals using time-domain and frequency-domain processing techniques.
2. Implement feature extraction methods such as LPC, MFCC, cepstral analysis, and pitch estimation for speech applications.
3. Develop and evaluate automatic speech recognition systems using HMMs, neural networks, and deep learning methods.
4. Design speaker recognition and speech synthesis systems for intelligent human-computer interaction.
5. Apply deep learning models for advanced speech/audio applications including speech enhancement, audio event detection, and multimodal AI systems.

UNIT - I: Fundamentals of Speech and Audio Signals
Introduction to speech and audio processing; human speech production mechanism; auditory perception system; characteristics of speech signals; digital audio fundamentals; sampling and quantization; time-domain and frequency-domain representation; short-time energy and zero-crossing rate; Fourier Transform and Short-Time Fourier Transform (STFT); spectrogram analysis; speech coding basics; applications of speech and audio processing.

UNIT - II: Speech Feature Extraction and Processing
Speech preprocessing techniques; framing and windowing; cepstral analysis; Linear Predictive Coding (LPC); Mel Frequency Cepstral Coefficients (MFCC); perceptual linear prediction; pitch detection and formant estimation; endpoint detection algorithms; noise reduction techniques; adaptive filtering; Wiener filtering; spectral subtraction; feature normalization and enhancement methods.

UNIT - III: Automatic Speech Recognition (ASR)
Introduction to ASR systems; architecture of speech recognition systems; Dynamic Time Warping (DTW); Hidden Markov Models (HMM); acoustic and language modeling; pronunciation modeling; Gaussian Mixture Models (GMM); neural network-based ASR; deep neural networks for speech recognition; end-to-end ASR systems; Connectionist Temporal Classification (CTC); evaluation metrics for ASR systems; applications of speech recognition.

UNIT - IV: Speech Synthesis and Speaker Recognition
Text-to-Speech (TTS) systems; speech synthesis techniques; articulatory, concatenative, and parametric synthesis methods; waveform generation; speaker identification and speaker verification; biometric authentication using speech; emotion recognition from speech signals; prosody modeling; multimodal speech interaction systems; conversational AI and virtual assistants.

UNIT - V: Deep Learning for Speech and Audio Applications
Deep learning foundations for speech and audio processing; Convolutional Neural Networks (CNNs) for spectrogram analysis; Recurrent Neural Networks (RNNs) and LSTMs for sequential speech modeling; attention mechanisms and Transformer architectures; speech enhancement and separation; audio event detection; music information retrieval; multilingual and low-resource speech systems; self-supervised speech learning; recent trends and real-world AI applications in speech and audio technologies.

Textbooks

1. Daniel Jurafsky and James H. Martin. 2025. Speech and Language Processing (3rd ed. draft). Pearson, Hoboken, NJ.
2. Lawrence Rabiner and Biing-Hwang Juang. 1993. Fundamentals of Speech Recognition. Prentice Hall, Englewood Cliffs, NJ.
3. Alexander Lerch. 2012. An Introduction to Audio Content Analysis. Wiley, Hoboken, NJ.

Reference Books

1. Thomas F. Quatieri. 2001. Discrete-Time Speech Signal Processing. Prentice Hall, Upper Saddle River, NJ.
2. Douglas O’Shaughnessy. 2008. Speech Communications: Human and Machine (2nd ed.). Universities Press, Hyderabad, India.
3. James Lyons. 2019. Python Speech Features and Audio Signal Processing. Packt Publishing, Birmingham, UK.

Reference Links

1. https://nptel.ac.in/courses/117105145
2. https://web.stanford.edu/~jurafsky/slp3/
3. https://signalprocessingsociety.org/

Information Retrieval Systems

(Program Elective - III)

Subject Code: 26MAI1E33

L	T	P	C
3	0	0	3

Course Objectives

1. Understand the fundamental concepts, architecture, and applications of Information Retrieval (IR) systems.
2. Learn text preprocessing, indexing techniques, and efficient storage mechanisms for large-scale document collections.
3. Analyze classical and probabilistic retrieval models, query processing methods, and relevance feedback mechanisms.
4. Study web search technologies, ranking algorithms, crawling, and link analysis techniques used in modern search engines.
5. Explore intelligent IR approaches including semantic search, recommender systems, machine learning–based retrieval, and AI-driven search applications.

Course Outcomes

Upon successful completion of the course, students will be able to:

1. Explain the principles, components, and performance evaluation metrics of Information Retrieval Systems.
2. Design and implement text preprocessing and indexing mechanisms for efficient retrieval.
3. Apply various retrieval models and query optimization techniques to improve search effectiveness.
4. Analyze and evaluate web search algorithms, ranking methods, and large-scale retrieval architectures.
5. Develop AI-enabled retrieval solutions using semantic analysis, recommendation techniques, and intelligent search models.

UNIT - I: Fundamentals of Information Retrieval

Introduction to Information Retrieval Systems, Boolean Retrieval Model, Architecture of IR Systems, Characteristics of Textual Information, Information Retrieval vs Database Systems, Document Representation and Document Collections, Performance Evaluation: Precision, Recall, F-measure, MAP, NDCG, Applications of IR in AI and Data Science.

UNIT - II: Text Processing and Indexing

Text Preprocessing Techniques, Tokenization, Stop-word Removal, Stemming, Lemmatization, Term Frequency and Inverse Document Frequency (TF-IDF), Dictionary Structures and Inverted Indexing, Positional and Biword Indexes, Compression Techniques in IR, Index Construction and Maintenance, Distributed and Scalable Indexing.

UNIT – III: Retrieval Models and Query Processing

Vector Space Model, Probabilistic Retrieval Models, Language Models for Information Retrieval, Query Processing and Query Expansion, Relevance Feedback and Rocchio Algorithm, Latent Semantic Indexing (LSI), Evaluation and Optimization of Retrieval Systems, Machine Learning Approaches in Retrieval.

UNIT – IV: Web Search and Ranking Algorithms

Web Search Engine Architecture, Web Crawling and Spidering, Link Analysis Algorithms: PageRank and HITS, Ranking and Personalization Techniques, Duplicate Detection and Near-Duplicate Detection, Metadata and Structured Search, Search Engine Optimization Concepts, Distributed Web Search Systems.

UNIT – V: Intelligent Information Retrieval and Semantic Search
Intelligent Information Retrieval Systems, Semantic Web and Ontology-based Retrieval, Semantic Search and Knowledge Graphs, Recommender Systems and Collaborative Filtering, Context-aware and Personalized Search, Deep Learning for Information Retrieval, Conversational Search and AI-driven Retrieval Systems, Recent Trends in Neural IR and Generative Search.

Textbooks

1. Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. Introduction to Information Retrieval. Cambridge University Press, Cambridge, UK.
2. Ricardo Baeza-Yates and Berthier Ribeiro-Neto. 2011. Modern Information Retrieval: The Concepts and Technology behind Search (2nd ed.). Addison-Wesley, Boston, MA.
3. Stefan Büttcher, Charles L. A. Clarke, and Gordon V. Cormack. 2016. Information Retrieval: Implementing and Evaluating Search Engines. MIT Press, Cambridge, MA.

Reference Textbooks

1. W. Bruce Croft, Donald Metzler, and Trevor Strohman. 2015. Search Engines: Information Retrieval in Practice. Pearson, Boston, MA.
2. Soumen Chakrabarti. 2003. Mining the Web: Discovering Knowledge from Hypertext Data. Morgan Kaufmann, San Francisco, CA.
3. Manning Publications. 2021. Practical Natural Language Processing. O'Reilly Media, Sebastopol, CA.

Reference Links

1. https://nptel.ac.in/courses/106106211
2. https://nlp.stanford.edu/IR-book
3. https://dl.acm.org/topic/ccs2012/10002951.10003260.10003261

Explainable AI

(Program Elective - III)

Subject Code: 26MAI1E34

L	T	P	C
3	0	0	3

Course Objectives

The student will be able to:

1. Understand the fundamental concepts, principles, and significance of Explainable Artificial Intelligence (XAI) in modern AI systems.
2. Analyze the limitations of black-box machine learning models and evaluate the need for transparency, fairness, accountability, and trustworthiness.
3. Study interpretable machine learning techniques and explainability methods for classical and deep learning models.
4. Apply post-hoc explanation methods such as LIME, SHAP, saliency maps, and counterfactual explanations to real-world AI applications.
5. Examine ethical, legal, and governance aspects of responsible AI and explore recent advances in explainable and trustworthy AI systems.

Course Outcomes

Upon successful completion of the course, the student will be able to:

1. Analyze the challenges associated with black-box AI systems and the importance of interpretability in AI-driven decision-making.
2. Apply interpretable machine learning models and explainability techniques to develop transparent AI solutions.
3. Evaluate post-hoc explanation methods for assessing model behavior, feature importance, and prediction reliability.
4. Design and implement explainable deep learning and generative AI models considering fairness, bias mitigation, and accountability.
5. Assess responsible AI governance frameworks and research trends for deploying trustworthy AI applications in real-world domains.

UNIT - I: Introduction to Explainable AI

Introduction to Artificial Intelligence and Machine Learning systems; Need for Explainable AI; Interpretability vs Explainability; Black-box vs Transparent Models; Human-centered AI; Ethical AI principles; Fairness, Accountability, Transparency, and Trust (FATT); Regulatory and legal perspectives of AI; Risks and challenges in AI deployment; Introduction to Responsible AI frameworks and standards.

UNIT - II: Interpretable Machine Learning Models

Characteristics of interpretable models; Intrinsic interpretability; Linear and Logistic Regression models; Decision Trees and Rule-based Systems; k-Nearest Neighbors; Generalized Additive Models (GAMs); Model transparency and simplicity; Feature engineering for interpretability; Evaluation metrics for explainability; Trade-off between accuracy and interpretability.

UNIT - III: Post-hoc Explainability Techniques

Post-hoc interpretability concepts; Local vs Global explanations; Model-agnostic explanation methods; LIME (Local Interpretable Model-Agnostic Explanations); SHAP (SHapley Additive exPlanations); Partial Dependence Plots; Feature importance analysis; Saliency maps and Grad-CAM; Counterfactual explanations; Surrogate models; Visualization techniques for explanation and debugging.

UNIT - IV: Explainability in Deep Learning and Generative AI
Explainability challenges in deep neural networks; Attention mechanisms and transformer interpretability; Explainable CNNs and RNNs; Interpreting embeddings and latent representations; Explainability in Generative AI and Large Language Models; Bias detection and mitigation in AI systems; Fairness-aware machine learning; Adversarial attacks and robustness; Explainable reinforcement learning; Trustworthy AI pipelines.

UNIT - V: Applications, Governance, and Research Trends in XAI
Applications of XAI in healthcare, finance, cybersecurity, autonomous vehicles, smart governance, and recommendation systems; AI auditing and monitoring; Responsible AI governance frameworks; AI ethics policies and compliance; Human-AI collaboration; Case studies on explainable systems; Research challenges in XAI; Emerging trends in trustworthy and explainable generative AI; Future directions in Responsible AI and Explainable Foundation Models.

Textbooks

1. Christoph Molnar. 2024. Interpretable Machine Learning (2nd ed.). Leanpub, Budapest, Hungary.
2. Denis Rothman. 2020. Artificial Intelligence By Example (2nd ed.). Packt Publishing, Birmingham, UK.
3. Serg Masís. 2022. Interpretable Machine Learning with Python. Packt Publishing, Birmingham, UK.

Reference Books

1. Wojciech Samek, Grégoire Montavon, Andrea Vedaldi, Lars Kai Hansen, and Klaus-Robert Müller, eds. 2019. Explainable AI: Interpreting, Explaining and Visualizing Deep Learning. Springer, Cham, Switzerland.
2. Patrick Hall, Navdeep Gill, and Nicholas Schmidt. 2019. Machine Learning for High-Risk Applications. O'Reilly Media, Sebastopol, CA.
3. Mark Ryan. 2020. In AI We Trust: Ethics, Artificial Intelligence, and Reliability. Routledge, London, UK.

Reference Links

1. https://www.aicte.gov.in/
2. https://nptel.ac.in/courses/106105215
3. https://christophm.github.io/interpretable-ml-book/

Generative AI and Large Language Models

(Program Elective - IV)

Subject Code: 26MAI1E41

L	T	P	C
3	0	0	3

Course Objectives

1. Understand the theoretical foundations of Generative AI, deep learning, and probabilistic modeling techniques used in modern AI systems.
2. Analyze transformer architectures, attention mechanisms, and neural sequence modeling techniques underlying Large Language Models (LLMs).
3. Explore foundation models, pretrained language models, multimodal generative systems, and their training paradigms.
4. Apply prompt engineering, fine-tuning, Retrieval-Augmented Generation (RAG), and adaptation techniques for domain-specific AI applications.
5. Evaluate ethical, societal, security, and governance issues associated with Generative AI and develop responsible AI solutions for real-world applications.

Course Outcomes

Upon successful completion of the course, students will be able to:

1. Explain the principles of Generative AI, deep neural architectures, and probabilistic generative modeling techniques.
2. Analyze and implement transformer-based architectures and attention mechanisms for NLP and generative tasks.
3. Evaluate the design, capabilities, and limitations of Large Language Models and foundation models.
4. Develop AI applications using prompt engineering, fine-tuning methods, vector databases, and Retrieval-Augmented Generation frameworks.
5. Assess ethical implications, bias, security risks, and emerging trends in Generative AI for responsible deployment in industry and research.

UNIT - I: Introduction to Generative AI and Deep Learning Foundations

Introduction to Artificial Intelligence, Machine Learning, and Deep Learning; Overview of Generative AI; Discriminative vs Generative Models; Neural Networks and Deep Learning Fundamentals; Representation Learning; Auto encoders, Variational Auto encoders (VAEs), Generative Adversarial Networks (GANs); Sequence Modeling; Basics of Natural Language Processing (NLP); Applications of Generative AI in text, image, audio, and multimodal systems.

UNIT - II: Transformer Architecture and Attention Mechanisms

Limitations of RNNs and LSTMs; Self-Attention Mechanism; Scaled Dot-Product Attention; Multi-Head Attention; Positional Encoding; Encoder–Decoder Architecture; Transformer Models; BERT, GPT, T5, and Encoder-only vs Decoder-only Architectures; Training Strategies for Transformers; Computational Complexity and Optimization Techniques.

UNIT - III: Large Language Models (LLMs) and Foundation Models

Introduction to Foundation Models and LLMs; Tokenization Techniques; Pretraining Objectives; Language Modeling and Next Token Prediction; Scaling Laws; Emergent Capabilities of LLMs; Instruction Tuning and Alignment; Reinforcement Learning from Human Feedback (RLHF); Opensource and Proprietary LLM Ecosystems; Multimodal Generative Models; Evaluation Metrics for LLMs.

UNIT - IV: Prompt Engineering, Fine-Tuning, and Retrieval-Augmented Generation (RAG)
Prompt Engineering Techniques: Zero-shot, Few-shot, Chain-of-Thought Prompting; Prompt Optimization; Fine-Tuning and Parameter-Efficient Fine-Tuning (PEFT); LoRA and Adapters; Embeddings and Vector Databases; Semantic Search; Retrieval-Augmented Generation (RAG) Architecture; LangChain and LLM Orchestration Frameworks; AI Agents and Tool Use; Deployment Considerations and Model Serving.

UNIT - V: Applications, Ethics, Safety, Bias, Security, and Future Trends in Generative AI
Applications of Generative AI in Healthcare, Education, Finance, Cybersecurity, Software Engineering, and Content Generation; AI Hallucinations and Reliability Issues; Bias, Fairness, Explainability, and Transparency; AI Governance and Regulatory Frameworks; Privacy and Security Risks in LLMs; Adversarial Attacks and Prompt Injection; Green AI and Sustainable AI; Artificial General Intelligence (AGI); Future Research Directions and Industrial Trends.

Textbooks

1. Jay Alammar and Maarten Grootendorst. 2024. Hands-On Large Language Models: Language Understanding and Generation. O'Reilly Media, Sebastopol, CA.
2. Denis Rothman. 2024. Transformers for Natural Language Processing (3rd ed.). Packt Publishing, Birmingham, UK.
3. Sebastian Raschka. 2024. Build a Large Language Model (From Scratch). Manning Publications, Shelter Island, NY.

Reference Books

1. Yoav Goldberg. 2017. Neural Network Methods for Natural Language Processing. Morgan & Claypool Publishers, San Rafael, CA.
2. Brian Roemmele and Andrew Jones. 2024. The AI Engineering Bible: Building Applications with Foundation Models. O'Reilly Media, Sebastopol, CA.
3. Lewis Tunstall, Leandro von Werra, and Thomas Wolf. 2022. Natural Language Processing with Transformers. O'Reilly Media, Sebastopol, CA.

Reference Links

1. https://huggingface.co/docs
2. https://openai.com/research
3. https://crfm.stanford.edu/

GPU Programming
(Program Elective - IV)

Subject Code: 26MAI1E42

L	T	P	C
3	0	0	3

Course Objectives

- To learn parallel programming with Graphics Processing Units (GPUs)

Course Outcomes

Upon successful completion of the course, the student will be able to:

1. Develop parallel programs using Open CL library.
2. Analyze for the performance of GPU memory hierarchy.
3. Learn concepts in parallel programming, implementation of programs on GPUs, Debugging, and profiling parallel programs.
4. Generate parallel programs for matrix, graph and sorting problems using Cuda library.
5. Compare the performance of different algorithms for the numerical and data processing problems on GPU and suggest methods for improving the performance.

UNIT - I: Introduction: History, Graphics Processors, Graphics Processing Units, GPGPUs. Clock speeds, CPU / GPU comparisons, Heterogeneity, Accelerators, Parallel programming, CUDA OpenCL / OpenACC, Hello World Computation Kernels, Launch parameters, Thread hierarchy, Warps/Wavefronts, Thread blocks / Workgroups, Streaming multiprocessors, 1D / 2D / 3D thread mapping, Device properties, Simple Programs.

UNIT - II: Memory: Memory hierarchy, DRAM / global, local / shared, private / local, textures, Constant Memory, Pointers, Parameter Passing, Arrays and dynamic Memory, Multi-dimensional Arrays, Memory Allocation, Memory copying across devices, Programs with matrices, Performance evaluation with different memories.

UNIT - III: Synchronization: Memory Consistency, Barriers (local versus global), Atomics, Memory fence. Prefix sum, Reduction. Programs for concurrent Data Structures such as Worklists, Linked-lists. Synchronization across CPU and GPU **Functions**: Device functions, Host functions, Kernels functions, Using libraries (such as Thrust), and developing libraries.

UNIT - IV: Support: Debugging GPU Programs. Profiling, Profile tools, Performance aspects **Streams**: Asynchronous processing, tasks, Task-dependence, Overlapped data transfers, Default Stream, Synchronization with streams. Events, Event-based-synchronization, overlapping data transfer and kernel execution, pitfalls.

UNIT - V: Advanced topics: Dynamic parallelism, Unified Virtual Memory, Multi-GPU processing, Peer access, Heterogeneous processing.

Text Books:

1. Programming Massively Parallel Processors: A Hands-on Approach; David Kirk, Wen-meiHwu; Morgan Kaufman; 2010 (ISBN: 978-0123814722)
2. CUDA Programming: A Developer's Guide to Parallel Computing with GPUs; Shane Cook; Morgan Kaufman; 2012 (ISBN: 978-0124159334)

Reference Books:

1. Benedict R Gaster, Lee Howes, David R Kaeli Perhaad Mistry Dana Schaa, Heterogeneous Computing with OpenCL, MGH, 2011.
2. Jason Sanders, Edward Kandrot, CUDA By Example – An Introduction to General-Purpose GPU Programming, Addison Wesley, 2011.
3. Michael J Quinn, Parallel Programming in C with MPI and OpenMP, TMH, 2004.

Artificial Intelligence of Things

(Program Elective - IV)

Subject Code: 26MAI1E43

L	T	P	C
3	0	0	3

Course Objectives:

- To understand the integration of Artificial Intelligence and the Internet of Things.
- To explore IoT architecture, communication protocols, and device ecosystems.
- To learn AI and ML fundamentals for intelligent IoT systems.
- To design AIoT architectures and handle large-scale IoT data processing.
- To explore practical applications and emerging trends in AIoT.

Course Outcomes:

After successful completion of the course, students will be able to:

1. Explain the concepts, architecture, enabling technologies, challenges, and ecosystem of Artificial Intelligence of Things (AIoT) systems.
2. Analyze IoT architectures, communication protocols, devices, cloud/edge platforms, and security mechanisms used in intelligent connected systems.
3. Apply Artificial Intelligence and Machine Learning techniques to process, analyze, and derive insights from IoT-generated data.
4. Design and develop AIoT solutions using edge/cloud computing, data management techniques, and AI model deployment frameworks.
5. Evaluate AIoT applications with respect to security, privacy, ethics, sustainability, and emerging technological trends.

UNIT - I: Foundations of AIoT and IoT Technologies

Introduction, evolution, and need for AIoT, Convergence of Artificial Intelligence and Internet of Things, AIoT ecosystem and architecture, IoT architecture: perception, network, middleware, and application layers, Sensors, actuators, microcontrollers, and embedded platforms (Arduino, ESP32, Raspberry Pi), IoT communication protocols: MQTT, CoAP, HTTP, ZigBee, BLE, and LoRa, Benefits, challenges, and real-world AIoT applications.

UNIT - II: Artificial Intelligence and Machine Learning for AIoT

Overview of AI, Machine Learning, and Deep Learning, Types and evolution of AI systems, Machine Learning fundamentals and AI workflow, Learning paradigms: Supervised, Unsupervised, and Reinforcement Learning, Neural Networks and Deep Learning basics, AI frameworks and tools for AIoT, Predictive analytics, pattern recognition, and anomaly detection, Explainable and Ethical AI concepts.

UNIT - III: AIoT Data Management and Intelligent Analytics

AIoT system architecture: device, edge, cloud, and application layers, Data acquisition, preprocessing, transformation, and storage, Data flow and intelligent decision-making in AIoT systems, Real-time data streaming and analytics, Integration of AI models into IoT workflows, Cloud platforms and middleware for AIoT, AI model deployment using Cloud APIs, TensorFlow Lite, and ONNX.

UNIT - IV: Edge AI and TinyML

Fundamentals of Edge Computing and Edge Intelligence, Edge vs. Fog vs. Cloud Computing, Edge AI architecture and frameworks, TinyML concepts and development workflow, Deployment of AI models on resource-constrained devices, Model optimization techniques: quantization, pruning, and compression, Hardware platforms for Edge AI and TinyML, Applications in healthcare, agriculture, industrial IoT, and smart cities.

UNIT - V: Security, Privacy, and Emerging Trends in AIoT
Security challenges in AIoT systems, Device, network, data, and AI model security threats, Encryption, authentication, access control, and secure communication, Adversarial attacks, data poisoning, and model theft, Privacy-preserving AI: Federated Learning and Differential Privacy, Blockchain-enabled AIoT security, Ethical, legal, and regulatory considerations, AIoT platforms: AWS IoT, Azure IoT Hub, Google Cloud IoT, and IBM Watson IoT, Emerging trends: Digital Twins, 6G-enabled AIoT, Sustainable AIoT, and Zero-Trust Security.

Text Books
1. Arshdeep Bahga and Vijay Madisetti, Internet of Things: A Hands-On Approach, Universities Press.
2. Yunchuan Sun, AIoT: The Fusion of AI and IoT, Springer.
3. Raj Kamal, Internet of Things: Architecture and Design Principles, McGraw Hill.

Reference Books
1. Nitin Seth et al., Edge AI: Convergence of Edge Computing and Artificial Intelligence, Wiley.
2. Honbo Zhou, The Internet of Things in the Cloud: A Middleware Perspective, CRC Press.
3. Giuseppe Aceto, Antonio Pescapé, and Antonio Montieri, AI and IoT for Smart Systems, Springer.
4. Ovidiu Vermesan and Peter Friess (Eds.), Internet of Things – From Research and Innovation to Market Deployment, River Publishers.
5. Rajkumar Buyya et al., Fog and Edge Computing: Principles and Paradigms, Wiley.

AI for Cyber Security
(Program Elective - IV)

Subject Code: 26MAI1E44

L	T	P	C
3	0	0	3

Course Objectives
1. Understand the fundamentals of cyber security, cyber threats, and Artificial Intelligence techniques used in intelligent security systems.
2. Apply Machine Learning and Deep Learning algorithms for intrusion detection, malware analysis, anomaly detection, and threat prediction.
3. Analyze network traffic and security datasets to identify malicious activities and cyber attacks using AI-driven approaches.
4. Design intelligent cyber defense systems incorporating threat intelligence, secure automation, and adaptive learning mechanisms.
5. Evaluate adversarial threats, privacy-preserving mechanisms, ethical concerns, and secure AI architectures in cyber security applications.

Course Outcomes
Upon successful completion of the course, the student will be able to:
1. Analyze cyber security threats, attack vectors, vulnerabilities, and security requirements in modern computing environments.
2. Apply Machine Learning techniques for malware detection, intrusion detection, and anomaly-based threat analysis.
3. Develop Deep Learning-based intelligent security models for network and system protection.
4. Evaluate AI-driven cyber defense frameworks, threat intelligence systems, and security analytics solutions.
5. Design secure and trustworthy AI-enabled cyber security solutions considering adversarial attacks, ethics, and privacy preservation.

UNIT - I: Fundamentals of Cyber Security and Artificial Intelligence
Introduction to Cyber Security: Cyber threats, vulnerabilities, attack lifecycle, malware types, phishing, ransomware, botnets, insider threats. Cyber Security principles: Confidentiality, Integrity, Availability (CIA Triad), authentication, authorization, risk management. Fundamentals of Artificial Intelligence and Machine Learning for security applications. Supervised, unsupervised, and reinforcement learning concepts. Security datasets and feature engineering for cyber analytics. Introduction to cyber threat intelligence and AI-enabled security automation.

UNIT - II: Machine Learning for Threat Detection
Machine Learning techniques for cyber security. Classification algorithms: Decision Trees, Random Forest, Naïve Bayes, SVM, KNN.
Clustering and anomaly detection methods for identifying abnormal network behavior. Feature extraction and dimensionality reduction techniques. Spam filtering, phishing detection, fraud detection using ML models. Performance evaluation metrics: Accuracy, Precision, Recall, F1-score, ROC-AUC.

UNIT - III: Deep Learning for Malware and Intrusion Detection
Introduction to Deep Learning in cyber security. Artificial Neural Networks (ANN), Convolutional

Neural Networks (CNN), Recurrent Neural Networks (RNN), LSTM architectures. Deep Learning approaches for malware classification and intrusion detection. Behavior-based malware analysis and ransomware detection. Sequence learning for network traffic analysis. AI-driven endpoint protection systems.

UNIT - IV: AI for Network Security and Threat Intelligence
AI-based Security Information and Event Management (SIEM). Threat intelligence platforms and automated threat hunting. Network traffic analytics using AI techniques. AI for cloud security, IoT security, and edge security systems. Security orchestration, automation, and response (SOAR). Explainable AI (XAI) for cyber security decision-making.

UNIT - IV: Adversarial AI, Privacy, and Secure AI Systems
Adversarial Machine Learning: Evasion attacks, poisoning attacks, model inversion attacks. Defensive mechanisms against adversarial threats. Privacy-preserving AI techniques: Federated Learning, Differential Privacy, Secure Multi-party Computation. Ethical hacking and penetration testing concepts. AI governance, ethics, bias, accountability, and legal considerations in cyber security. Future trends: Autonomous cyber defense systems, Generative AI in cyber security.

Textbooks

1. Leslie F. Sikos. 2024. AI-Powered Cybersecurity. Springer, Cham, Switzerland.
2. Mark Stamp. 2021. Introduction to Machine Learning with Applications in Information Security. Chapman and Hall/CRC, Boca Raton, FL.
3. Clarence Chio and David Freeman. 2018. Machine Learning and Security: Protecting Systems with Data and Algorithms. O'Reilly Media, Sebastopol, CA.

Reference Books

1. Brij B. Gupta, Deepak Gupta, and Zbigniew H. Perkowski, eds. 2021. Handbook of Research on AI Cybersecurity. IGI Global, Hershey, PA.
2. Soma Halder and Sinan Ozdemir. 2018. Hands-On Machine Learning for Cybersecurity. Packt Publishing, Birmingham, UK.
3. Joseph Migga Kizza. 2020. Guide to Computer Network Security (6th ed.). Springer, Cham, Switzerland.

Reference Links

1. https://www.nist.gov/cyberframework
2. https://attack.mitre.org
3. https://www.ieee.org/topics/cybersecurity.html

Deep Learning, NLP & Computer Vision Lab

Subject Code: 26MAI1103

L	T	P	C
0	0	4	2

Course Objectives

1. To provide practical exposure to deep learning, natural language processing, and computer vision techniques using modern AI frameworks and tools.
2. To enable students to design, train, and evaluate neural network models for image, text, and sequential data analysis.
3. To develop hands-on skills in implementing CNNs, RNNs, LSTMs, Transformers, and transfer learning models for real-world AI applications.
4. To familiarize students with NLP pipelines, language modeling, sentiment analysis, object detection, and image processing techniques.
5. To encourage experimentation, performance evaluation, and deployment of AI-driven intelligent systems for research and industrial applications.

Course Outcomes

Upon successful completion of the laboratory course, students will be able to:

1. Implement and evaluate deep learning models using modern frameworks such as Tensor Flow, Keras, and PyTorch.
2. Develop CNN-based computer vision solutions for image classification, feature extraction, and object detection tasks.
3. Build NLP applications involving text preprocessing, sentiment analysis, sequence modeling, and Transformer-based architectures.
4. Apply optimization, regularization, transfer learning, and hyper parameter tuning techniques for improving AI model performance.
5. Design and deploy integrated AI solutions for practical applications in healthcare, surveillance, conversational AI, and intelligent automation.

List of Experiments

1. Implementation of basic Artificial Neural Network (ANN) using Tensor Flow/Keras for classification problems.
2. Implementation and comparison of optimization algorithms such as SGD, RMSProp, and Adam on deep neural networks.
3. Design and implementation of Convolutional Neural Networks (CNNs) for handwritten digit/image classification using MNIST/CIFAR datasets.
4. Implementation of image preprocessing techniques using OpenCV including filtering, edge detection, segmentation, and morphological operations.
5. Implementation of Transfer Learning using pre-trained CNN architectures such as VGG16, ResNet, or MobileNet for image classification.
6. Development of an Object Detection model using YOLO/SSD/Faster R-CNN frameworks.
7. Implementation of Recurrent Neural Networks (RNN), LSTM, and GRU models for sequence prediction or time-series forecasting.
8. Implementation of NLP preprocessing techniques including tokenization, stemming, lemmatization, POS tagging, and Named Entity Recognition (NER).

9. Development of text classification and sentiment analysis models using machine learning and deep learning techniques.
10. Implementation of Transformer-based NLP models using BERT/GPT/Hugging Face libraries for text summarization or question answering.
11. Implementation of Auto encoders or Generative Adversarial Networks (GANs) for representation learning or image generation tasks.
12. Write and execute a simple Python program using Tensor Flow/Keras or PyTorch to build and train a basic neural network model for handwritten digit classification using the MNIST dataset, and evaluate the model accuracy.

Textbooks

1. Aurélien Géron. 2022. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (3rd ed.). O'Reilly Media, Sebastopol, CA.
2. François Chollet. 2021. Deep Learning with Python (2nd ed.). Manning Publications, Shelter Island, NY.
3. Richard S. Sutton and Andrew G. Barto. 2018. Reinforcement Learning: An Introduction (2nd ed.). MIT Press, Cambridge, MA.

Reference Textbooks

1. Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. Deep Learning. MIT Press, Cambridge, MA.
2. Daniel Jurafsky and James H. Martin. 2025. Speech and Language Processing (3rd ed. draft). Pearson, Hoboken, NJ.
3. Richard Szeliski. 2022. Computer Vision: Algorithms and Applications (2nd ed.). Springer, Cham, Switzerland.

Reference Links

1. https://www.tensorflow.org/
2. https://pytorch.org/
3. https://docs.opencv.org/

Research Methodology and IPR
(Common for all M.Tech. Specializations)

Subject Code: 26MCC1001

L	T	P	C
2	0	0	2

Course Objectives:

- Formulate research problems with clarity and develop strong analytical skills.
- Apply structured research methodology, literature review techniques, and ethical practices to academic work.
- Interpret research data effectively and master the mechanics of technical report writing.
- Gain a comprehensive understanding of Intellectual Property Rights (IPR) and the legalities of the patenting process.
- Understand the global landscape of patent rights, technology transfer, and emerging trends in IP.

Course Outcomes:

- Summarize the fundamental concepts, types, and systematic processes of scientific research.
- Formulate a well-defined research problem by applying systematic selection techniques
- Construct a comprehensive literature review and theoretical framework while maintaining high ethical standards.
- Execute data interpretation and produce high-quality technical reports and oral presentations.
- Navigate the legalities of Intellectual Property Rights, including patent filing licensing, and emerging IP trends.

UNIT - I: Introduction to Research Methodology

Meaning, objectives, and motivation in research. Research Taxonomy: Types of research (Descriptive vs. Analytical, Applied vs. Fundamental, and Quantitative vs. Qualitative). Methodology: Research approaches, significance of research, and the distinction between Research Methods vs. Methodology. Scientific Rigor: Research and the scientific method; Importance of understanding the research process from inception to completion.

UNIT - II: Defining the Research Problem

Quality Standards: Criteria of good research; Common problems encountered by researchers in India. Problem Formulation: The art of defining the research problem; Necessity and importance of a well-defined problem. Techniques: Step-by-step process of identifying and selecting a research problem. Application: Practical illustrations and the logic of narrowing down a broad area into a researchable topic.

UNIT - III: Literature Review and Research Ethics

Strategic Review: Place of literature review in research; Bringing clarity and focus to the problem; improving methodology and broadening knowledge base. Execution: Searching existing literature, reviewing selected literature, and contextualizing findings. Frameworks: Developing theoretical and conceptual frameworks for the study. Integrity: Writing about the literature reviewed; Analysis of Plagiarism and adherence to Research Ethics.

UNIT - IV: Interpretation and Report Writing

Data Synthesis: Meaning and techniques of interpretation; Precautions to take during data interpretation. Technical Communication: Significance of report writing; Different steps and layout of a formal research report. Formatting & Mechanics: Types of reports, oral presentation skills, and the mechanics of writing a research report. Quality Control: Precautions for writing effective and accurate research reports.

UNIT - V: Nature of Intellectual Property and Patenting

IP Fundamentals: Nature of Intellectual Property: Patents, Designs, Trademarks, and Copyrights. The Patent Lifecycle: From technological research and innovation to development; Procedure for grant of patents. Global Framework: International cooperation on IP; Patenting under the Patent Cooperation Treaty (PCT). Rights & Trends: Scope of patent rights; Licensing and transfer of technology; Patent databases. Emerging Areas: Geographical Indications (GI), Traditional knowledge case studies.

Text Books

1. Kothari, C. R (1990). Research methodology, 2nd edition, New Age International Publishers.
2. Kothari, C. R., & Garg, G. (2024). Research methodology: Methods and techniques (5th ed.). New Age International Publishers.
3. Ramappa, T. (2021). Intellectual property rights under WTO (3rd ed.). S. Chand Publishing.

Reference Books

1. Chawla, D., & Sondhi, N. (2016). Research methodology: Concepts and cases (2nd ed.).Vikas Publishing House.
2. Cornish, W., Llewelyn, D., & Aplin, J. (2025). Intellectual property: Patents, copyright, trade marks and allied rights (10th South Asian ed.). Sweet & Maxwell. Available at Bharat Law House.
3. Controller General of Patents, Designs and Trade Marks. (2024). Manual of patent practice and procedure. IP India Official Website.
4. World Intellectual Property Organization. (2024). PCT – The International Patent System. WIPO.