NATIONAL INSTITUTE OF ELECTRONICS AND INFORMATION TECHNOLOGY

DEEMED TO BE UNIVERSITY UNDER DISTINCT CATEGORY

(AN AUTONOMOUS INSTITUTION UNDER MINISTRY OF ELECTRONICS & IT, GOVT. OF INDIA)

Curriculum and Syllabi for

M.Tech (Artificial Intelligence and Data Science)

(2025-26 Scheme)

रा.इ.सू.प्रौ.सं

NIELIT

Deemed to be University

Main Campus at Ropar and Constituent Units at Aizawl, Agartala, Aurangabad, Calicut, Gorakhpur, Imphal, Itanagar, Kekri, Kohima, Patna and Srinagar

Vision

To be a globally recognized leader in the field of Computer Science and Engineering, with a strong emphasis on Artificial Intelligence, Data Science, and other emerging areas, fostering innovation, excellence in education, and groundbreaking research.

Mission

1.Deliver High-Quality Graduate Programs: Provide cutting-edge graduate education in Computer Science and Engineering, equipping students with advanced skills and knowledge in Artificial Intelligence and Data Science to meet the challenges of the digital age.
2.Advance Research and Knowledge Creation: Contribute to the global body of knowledge by engaging in transformative research that explores new methodologies, technologies, and solutions in AI and Data Science.
3.Serve and Empower the Community: Offer expertise, resources, and services to industries, startups, and the broader community, applying academic research to solve real-world problems and foster technological advancements.
4.Foster Faculty Development and Retention: Create an environment that supports faculty growth and development by providing continuous professional development opportunities, encouraging research, and enhancing teaching skills to retain top talent in academia.

Program Education Objectives

PEO1: Acquire Advanced Knowledge and Skills: Equip students with both advanced theoretical and practical knowledge in Artificial Intelligence and Data Science, enhancing their ability to design, test, and adapt emerging computing technologies to achieve professional excellence and excel in their careers within academia and industry.
PEO2: Develop Critical Thinking and Problem-Solving Abilities: Foster the ability to think critically, analyze complex computational problems, and propose techno-commercially feasible and socially responsible solutions, driving innovation while excelling in research and development (R&D).
PEO3: Collaborate on Multidisciplinary Innovations: Prepare students to work effectively in collaborative environments, developing innovative systems and optimized solutions across various domains, while upholding high standards of professional and ethical values in both organizational and global contexts.
PEO4: Cultivate Innovation and Entrepreneurial Mindset: Encourage the development of design thinking capabilities, fostering innovation and entrepreneurship, and preparing students to drive technological advancements and contribute to new ventures in the AI and Data Science space.

Program Outcomes

PO1: Develop a Strong Foundation in AI and Related Fields – Covers a wide range of AI topics like Machine Learning, Natural Language Processing, Robotics, and Human-Computer Interaction.
PO2: Integrate AI and Data Science for System Development – Emphasizes combining various AI and Data Science techniques for designing intelligent systems.
PO3: Master Modern Machine Learning Techniques – Focuses on the principles and working mechanisms of Machine Learning, ensuring a deep theoretical understanding.
PO4: Build Scalable and Robust AI-Based Solutions – Develops real-time AI applications with appropriate software and hardware choices.

Program Specific Outcomes

PSO1: Master the Fundamentals of AI: Acquire a solid foundation in key areas of Artificial Intelligence, including Machine Learning, Inference Engines, Speech Processing, Computer Vision, Natural Language Understanding, Robotics, and Human-Computer Interaction.
PSO2: Integrate AI and Data Science for System Design: Synthesize principles from human cognition, AI, Machine Learning, and Data Engineering to design intelligent, adaptive, and data-driven systems that solve real-

world challenges.
PSO3: Apply Machine Learning Techniques Effectively: Understand and apply both theoretical and practical aspects of modern machine learning methods, gaining insights into their workings, strengths, limitations, and the appropriate contexts for their use.
PSO4: Develop Real-World AI Systems: Design and implement robust, real-time AI-based systems by choosing the right software, hardware, and data infrastructure to meet specific application requirements.
PSO5: Engage in Continuous Learning and Research: Continuously update and refine knowledge in Artificial Intelligence, engaging in further studies and research to contribute to societal advancements and meet the evolving demands of the field.

CATEGORY WISE CREDIT DISTRIBUTION

Category	Category Code	Credits
Skill / Employment Enhancement Courses (Project/Internship/Seminar/Dissertation/Research)	EE	36
Audit Courses (without grade or credit)	AU	0
Open Elective Courses	OE	8
Professional Elective Courses	PE	16
Program Core Courses	PC	20
Total Credits (Common track)		**80**

EVALUATION AND ASSESSMENT

1. Performance of a student in a semester shall be evaluated through continuous Class assessment, Tutorial/Lab assessment, Mid-Semester Examination (MSE) and End-Semester Examination (ESE). Both the MSE and ESE shall be the University examination and will be conducted as notified by the Controller of Examinations (CoE) of the University.

2. The continuous assessment shall be based on assignments, tutorials, paper presentation / quizzes/ viva-voce / flipped classes, lab work / projects / fieldwork and attendance, etc.

3. The MSE/ESE shall be comprising of written papers.

4. The overall assessment of the students will be done as per the following scheme:

S. No.	Assessment Type	Marks Weightage
1.	Mid Semester Examination	20
2.	End Semester Examination	40
3.	Continuous Assessment - Practical /Lab/ Tutorial	25
4.	Continuous Assessment - Theory	15
Total		**100**

For more details, please refer the applicable ordinance for this programme and Assessment SOPs.

CURRICULUM

Semester 1						
Semester Code	**Course Code**	**Course Title**	**L**	**T**	**P**	**Cr**
PC-MTEAID-101	DASH250075	Mathematics for Data Science	3	1	0	4
PC-MTEAID-102	DOAI250004	Artificial Intelligence	3	1	0	4
PE-MTEAID-103	Elective	Program Elective				4
OE-MTEAID-104	Elective	Open Elective				4
PC-MTEAID-105	DASH250095	Research Methodology and IPR	3	1	0	4
AU-MTEAID-106	Elective	Audit Elective				0

Semester 2						
Semester Code	**Course Code**	**Course Title**	**L**	**T**	**P**	**Cr**
PC-MTEAID-201	DOAI250024	Deep Learning	2	0	4	4
PC-MTEAID-202	DOAI250032	Machine Learning	3	0	2	4
PE-MTEAID-203	Elective	Program Elective				4
PE-MTEAID-204	Elective	Program Elective				4
AU-MTEAID-205	Elective	Audit Elective				0
EE-MTEAID-206	EE	Mini project with Seminar				4

Semester 3						
Semester Code	**Course Code**	**Course Title**	**L**	**T**	**P**	**Cr**
PE-MTEAID-301	Elective	Program Elective				4
OE-MTEAID-302	Elective	Open Elective				4
EE-MTEAID-303	EE	Dissertation-I/ Industrial project				14

Semester 4						
Semester Code	**Course Code**	**Course Title**	**L**	**T**	**P**	**Cr**
EE-MTEAID-401	EE	Dissertation-II/Project				18

Elective Courses

Elective Courses for PE Category

Course Code	Course Title	L	T	P	Cr	For Sem./Code
DASH250099	Social Networking and Mining	3	1	0	4	
DCSA250009	Blockchain Technology	3	1	0	4	
DCSE250064	Digital Forensics	3	0	2	4	
DCSE250071	Drone Technology	3	1	0	4	
DCSE250081	Fuzzy Logic and Fuzzy Sets	3	1	0	4	
DCSE250086	Health Care and Data Analytics	3	1	0	4	
DCSE250087	Intelligent Systems	3	1	0	4	
DCSE250098	NoSQL Database	3	0	2	4	
DCSE250123	Web Analytics	3	0	2	4	
DOAI250009	Big Data Analytics	3	0	2	4	
DOAI250041	Natural Language Processing	3	0	2	4	
DOAI250047	Recommender Systems	3	1	0	4	
DOEE250178	Robotics	3	0	2	4	

Elective Courses for OE Category

Students may opt courses under MOOC, NPTEL, SWAYAM, NSQF/NCVET aligned courses or other relevant technical subjects not included in their core curriculum. The exercise of option shall be subject to the Course Schedule of the respective Constituent Unit, the credit requirements and availability of seats. Guidelines for choosing MOOC/Online courses are given separately and shall have to be adhered to.

Elective Courses for AU (Audit) Category

Course Code	Course Title	L	T	P	Cr
DASH240027	Disaster Management	2	0	0	0
DASH240047	English for Research Paper Writing	2	0	0	0
DASH240052	Environmental Science	3	0	0	0
DASH240085	Pedagogy Studies	2	0	0	0
DASH240086	Personality Development through Life Enlightenment Skills	2	0	0	0
DASH240097	Sanskrit for Technical Knowledge	2	0	0	0
DASH240103	Stress Management by Yoga	2	0	0	0
DASH240104	Technical Presentation and Report Writing	0	0	2	0
DASH240106	Value Education	2	0	0	0
DASH240124	Constitution of India - AU	2	0	0	0
DASH250023	Constitution of India	2	1	0	0
DASH250027	Disaster Management	3	0	0	0
DASH250034	Energy and Environmental Engineering	3	0	0	0
DASH250047	English for Research Paper Writing	2	0	0	0
DASH250054	Essence of Indian Knowledge and Tradition	2	0	0	0

DASH250085	Pedagogy Studies	2	0	0	0
DASH250086	Personality Development	2	0	0	0
DASH250097	Sanskrit for Technical Knowledge	3	0	0	0
DASH250103	Stress Management by Yoga	3	0	0	0
DASH250106	Value Education	2	0	0	0
DASH250117	Innovative Thinking and Entrepreneurship Skills	1	0	0	0
DASH250118	Intellectual Property Rights	2	0	0	0
DCSA240080	Training on Computer Networking	0	0	2	0
DCSA240304	Lab Based on Statistical Package	0	0	2	0
Any other Scheduled Course as per the Course Schedule of the respective Constituent Unit can also be chosen, subject to availability of seats. However, there shall be no assessment and grading for the course chosen as Audit Course.					

The choice of all type of Electives available shall be as per the Course Schedule of the respective Constituent Unit.

Guidelines for Massive Open Online Courses (MOOC) / approved Online Courses

1. Relevant Swayam, MOOC, NPTEL, NIELIT NSQF and any other online courses approved by UGC/AICTE shall also be allowed as Open Electives(OE).

2. One credit of MOOC course should be minimum of 30 hours.

3. A student can choose any approved online course as Open Elective, subject to minimum credit requirements.

4. The students willing to opt OE under MOOC in their next semester, will submit their request to the MOOC Coordinator at their respective campus.

5. Once approved, the campus shall display the list of accepted courses.

6. The student has to submit certificate issued by the institution offering the MOOCs course, along with the number of credits and grades, to get credits transferred into his/her marks certificate issued by NDU.

7. The transfer of credits shall be as adopted by NDU.

Detailed Syllabus

Course Code	Course Name	L	T	P	Cr
DASH250075	Mathematics for Data Science	3	1	0	4

Course Outcomes:

CO1: Apply the concepts of vectors, matrices, eigenvalues, eigenvectors, and Singular Value Decomposition (SVD) to solve data science problems such as dimensionality reduction and data compression.
CO2: Demonstrate understanding of probability distributions, sampling, Bayesian concepts, and hypothesis testing to model uncertainty and make statistical inferences in data-driven applications.
CO3: Utilize calculus and optimization techniques, including gradient descent, stochastic gradient descent, and Lagrange multipliers, to solve constrained and unconstrained optimization problems in machine learning.
CO4: Analyze information-theoretic measures such as entropy, cross-entropy, KL divergence, and mutual information to evaluate data representation, feature selection, and model performance.
CO5: Apply mathematical tools such as numerical methods, matrix factorizations, least squares regression, and Markov chains for effective data modeling, signal analysis, and real-world computational applications.

Module 1:

Linear Algebra for Data Science (9 Hours)
Vectors, matrices, vector spaces, rank, null space, systems of linear equations, eigenvalues and eigenvectors, diagonalization, Singular Value Decomposition (SVD), principal component analysis (PCA); applications in dimensionality reduction and data compression.

Module 2:

Probability and Statistics (9 Hours)
Random variables, expectations, variance, covariance, distributions (Gaussian, Bernoulli, Poisson), law of large numbers, central limit theorem, Bayesian concepts, conditional independence, sampling techniques, hypothesis testing.

Module 3:

Calculus and Optimization (9 Hours)
Differentiation, partial derivatives, chain rule, multivariate Taylor series, Jacobian, Hessian Matrix, gradient descent, stochastic gradient descent (SGD), Newton’s method, convex optimization, constraints and Lagrange multipliers.

Module 4:

Information Theory and Entropy (9 Hours)
Entropy, cross-entropy, Kullback–Leibler (KL) divergence, mutual information, Shannon’s theory, applications in decision trees, feature selection, and deep learning loss functions, basics of MDL (Minimum Description Length) and compression theory.

Module 5:

Mathematical Tools for Data Modeling (9 Hours)
Numerical methods: interpolation, numerical differentiation and integration, matrix factorization (LU, QR, Cholesky), least squares regression, Markov chains, Fourier and Laplace transforms for data filtering and signal analysis.

Labs / Practicals:

N/A

References:

1. P. N. Gupta & S. C. Gupta – Mathematics for Data Science, Khanna Publishers.
2. B. S. Grewal – Higher Engineering Mathematics, Khanna Publishers.
3. V. Sundaram & S. Jayaraman – Probability and Random Processes, Tata McGraw Hill.
4. Kanti B. Datta – Matrix and Linear Algebra, PHI Learning.
5. S. Kumaresan – Linear Algebra: A Geometric Approach, Prentice-Hall of India.
6. Marc Deisenroth et al. – Mathematics for Machine Learning, Cambridge University Press.
7. Sheldon Ross – Introduction to Probability and Statistics for Engineers and Scientists.
8. Shai Shalev-Shwartz and Shai Ben-David – Understanding Machine Learning.

Course Code	Course Name	L	T	P	Cr
DOAI250004	Artificial Intelligence	3	1	0	4

Course Outcomes:

CO1: Understand AI foundations, intelligent agents, and problem-solving formulations.
CO2: Apply uninformed and heuristic search strategies, and adversarial game-playing techniques.
CO3: Represent knowledge using logic, rules, and reasoning methods.
CO4: Analyze different learning approaches including neural networks and connectionist models.
CO5: Design and evaluate expert systems for domain-specific knowledge representation and reasoning.

Module 1:

Introduction: AI problems, foundation of AI and history of AI intelligent agents: Agents and Environments, the concept of rationality, the nature of environments, structure of agents, problem solving agents, problem formulation.

Module 2:

Searching: Searching for solutions, uniformed search strategies – Breadth first search, depth first Search. Search with partial information (Heuristic search) Greedy best first search, A* search Game Playing: Adversial search, Games, minimax, algorithm, optimal decisions in multiplayer games, Alpha-Beta pruning, Evaluation functions, cutting of search.

Module 3:

Knowledge Representation: Using Predicate logic, representing facts in logic, functions and predicates, Conversion to clause form, Resolution in propositional logic, Resolution in predicate logic, Unification.
Representing Knowledge Using Rules: Procedural Versus Declarative knowledge, Logic Programming, Forward versus Backward Reasoning

Module 4:

Learning: What is learning, Rote learning, Learning by Taking Advice, Learning in Problem-solving,
Learning from example: induction, Explanation-based learning.
Connectionist Models: Hopfield Networks, Learning in Neural Networks, Applications of Neural Networks, Recurrent Networks. Connectionist AI and Symbolic AI.

Module 5:

Expert System: Representing and using Domain Knowledge, Reasoning with knowledge, Expert System Shells, Support for explanation examples, Knowledge acquisition-examples.

Labs / Practicals:

NA

References:

1. Artificial Intelligence – A Modern Approach. Second Edition, Stuart Russel, Peter Norvig, PHI/ Pearson Education.
2. Artificial Intelligence, Kevin Knight, Elaine Rich, B. Shivashankar Nair, 3rd Edition,2008
3. Artificial Neural Networks B. Yagna Narayana, PHI.
4. Artificial Intelligence, 2nd Edition, E.Rich and K.Knight (TMH)
5. Artificial Intelligence and Expert Systems – Patterson PHI.
6. Expert Systems: Principles and Programming- Fourth Edn, Giarrantana/ Riley, Thomson.
7. PROLOG Programming for Artificial Intelligence. Ivan Bratka- Third Edition – Pearson Education.
8. Neural Networks Simon Haykin PHI.
9. Artificial Intelligence, 3rd Edition, Patrick Henry Winston., Pearson Edition.

Course Code	Course Name	L	T	P	Cr
DASH250095	Research Methodology and IPR	3	1	0	4

Course Outcomes:

CO1: Understand the principles and process of research methodology.
CO2: Apply appropriate research design and data collection methods in computing projects.
CO3: Analyze and interpret qualitative and quantitative data using statistical tools.
CO4: Develop ethically sound, well-documented research or project reports conforming to academic and industrial standards.
CO5: Understand various forms of intellectual property and apply processes for protection and patent filing.

Module 1:

Introduction to Research and Research Ethics
Definition and objectives of research, types of research – fundamental, applied, exploratory, empirical, significance of research in computer applications, research ethics and integrity, plagiarism, citation styles (APA, IEEE), tools for plagiarism check.

Module 2:

Research Design and Data Collection
Research problem identification and formulation, hypothesis generation, literature review techniques, types of research design: experimental, descriptive, analytical, sampling techniques, primary and secondary data sources, survey design, case study design.

Module 3:

Data Analysis and Interpretation
Quantitative and qualitative data analysis, measures of central tendency and dispersion, t-test, ANOVA, correlation and regression, chi-square test, software tools: Excel, R, SPSS (demo-level), data visualization and result interpretation.

Module 4:

Technical Writing and Project Report Preparation
Structure of research papers and reports, abstract, introduction, literature review, methodology, results, and conclusions, referencing tools (Mendeley/Zotero), IEEE/ACM/SCI journal guidelines, thesis and dissertation formatting, proposal writing basics.

Module 5:

Intellectual Property Rights and Patent Process
Introduction to IPR: patents, copyrights, trademarks, industrial design, Indian and international patent systems (WIPO, TRIPS), criteria for patentability, patent filing process in India (IPINDIA portal), open-source licenses, patent drafting basics, IPR in software and AI innovations.

Labs / Practicals:

N/A

References:

1. C.R. Kothari & Gaurav Garg – Research Methodology: Methods and Techniques, New Age International.
2. Ranjit Kumar – Research Methodology – A Step-by-Step Guide, Sage Publications India.
3. P. Narayan – Intellectual Property Law, Eastern Book Company.
4. Neeraj Pandey & Khushdeep Dharni – Intellectual Property Rights, PHI Learning.
5. Yogesh Kumar Singh – Fundamentals of Research Methodology and Statistics, New Age International.
6. Bharat Bhushan – Research Methodology in Computer Science, Himalaya Publishing.
7. WIPO and IPIndia Manuals – Guidelines for Filing Patents in India.

Course Code	Course Name	L	T	P	Cr
DOAI250024	Deep Learning	2	0	4	4

Course Outcomes:

CO1: Understand the key concepts and training techniques of neural networks.
CO2: Apply deep learning models like CNNs and RNNs to solve real-world problems.
CO3: Gain expertise in advanced models like transformers and generative models for diverse applications.

Module 1:

Introduction to Neural Networks: Basics of Artificial Intelligence and Machine Learning, Introduction to Neural Networks and Biological Inspiration, Perceptron Model and Multilayer Perceptrons (MLP), Activation Functions (Sigmoid, ReLU, Tanh, Softmax), Gradient Descent and Backpropagation

Module 2:

Deep Feedforward Networks: Deep Learning vs. Traditional Machine Learning, Architecture of Feedforward Neural Networks (FNN), Weight Initialization Techniques, Optimization Techniques (SGD, Adam, RMSprop), Vanishing and Exploding Gradient Problems
Convolutional Neural Networks (CNNs): Introduction to Convolutional Neural Networks, Convolutional Layers, Pooling, and Padding, Popular Architectures (LeNet, AlexNet), Applications of CNNs in Image Processing

Module 3:

Recurrent Neural Networks (RNNs) and LSTMs: Introduction to Sequential Data and RNNs, Backpropagation Through Time (BPTT), Vanishing Gradient Problem in RNNs, Long Short-Term Memory (LSTMs) and Gated Recurrent Units (GRUs).

Module 4:

Generative Models and Autoencoders: Introduction to Generative Models, Autoencoders and Variational Autoencoders (VAEs), Generative Adversarial Networks (GANs), Applications in Image Generation and Data Augmentation

Module 5:

Generative Models and Autoencoders: Introduction to Attention Mechanisms, Transformer Architecture, Self-Attention and Multi-Head Attention, BERT, GPT, and Large Language Models, Applications in NLP and Computer Vision

Labs / Practicals:

Preparing

References:

1. Charu C. Aggarwal, “ Neural Networks and Deep Learning”, Springer
2. Ian Goodfellow, Yoshua Bengio, Aaron Courville, "Deep Learning", MIT Press
3. Rowel Atienza , “Advanced Deep Learning with TensorFlow 2 and Keras”, Packt Publishing
4. Lewis Tunstall, Leandro von Werra, Thomas Wolf, "Natural Language Processing with Transformers"
5. "Deep Learning for Natural Language Processing" – Palash Goyal, Sumit Pandey, Karan Jain
6. "Fundamentals of Deep Learning" – Nikhil Buduma, Nicholas Locascio

Course Code	Course Name	L	T	P	Cr
DOAI250032	Machine Learning	3	0	2	4

Course Outcomes:

CO1: Identify and extract relevant features from raw IoT data suitable for applying appropriate Machine Learning (ML) techniques.
CO2: Evaluate and compare the strengths and limitations of various ML algorithms across diverse application domains, particularly in IoT.
CO3: Select and implement appropriate ML models based on data characteristics and application requirements.
CO4: Mathematically formulate and analyze ML models and algorithms, including their underlying assumptions and performance metrics.
CO5: Design and validate machine learning-based solutions for real-world problems in IoT, demonstrating practical understanding and technical proficiency.

Module 1:

Module 1
Introduction: Introduction to Machine Learning: Introduction. Different types of learning, Hypothesis space and inductive bias, Evaluation. Training and test sets, cross validation, Concept of over fitting, under fitting, Bias and Variance.
Linear Regression: Introduction, Linear regression, Simple and Multiple Linear regression, Polynomial regression, evaluating regression fit.

Module 2:

Module 2
Decision tree learning: Introduction, Decision tree representation, appropriate problems for decision tree learning, the basic decision tree algorithm, hypothesis space search in decision tree learning, inductive bias in decision tree learning, issues in decision tree learning, Python exercise on Decision Tree.
Instance based Learning: K nearest neighbor, the Curse of Dimensionality, Feature Selection: forward search, backward search, univariate, multivariate feature selection approach, Feature reduction (Principal Component Analysis), Python exercise on KNN and PCA. Recommender System: Content based system, Collaborative filtering based.

Module 3:

Probability and Bayes Learning: Bayesian Learning, Naïve Bayes, Python exercise on Naïve Bayes, Logistic Regression.
Support Vector Machine: Introduction, the Dual formulation, Maximum margin with noise, nonlinear SVM and Kernel function, solution to dual problem.

Module 4:

Artificial Neural Networks: Introduction, Biological motivation, ANN representation, appropriate problem for ANN learning, Perceptron, multilayer networks and the back propagation algorithm.

Module 5:

Ensembles: Introduction, Bagging and boosting, Random forest, Discussion on some research papers.
Clustering: Introduction, K-mean clustering, agglomerative hierarchical clustering, Python exercise on k-mean clustering.

Labs / Practicals:

1 Write a Python program that imports a dataset of your choice, displays the first five rows, and prints the data type of the DataFrame as a whole. Then, generate and display the statistical summary (mean, median, standard deviation, etc.) for each column and provide complete information about the DataFrame.

2 Write a Python program to import (or create) a dataset with missing values, display the count of missing values per column, and handle them using both imputation and dropping methods. Then, compare the dataset before and after handling missing values, and display the final cleaned dataset.
3 Write a Python program to load (or create) a dataset, split it into training and testing sets, and standardize the features using StandardScaler from sklearn. preprocessing. Finally, display the dataset before and after standardization to observe the changes.
4 Write a Python program to load (or create) a dataset, separate it into predictor variables (X) and target variable (y), and then split the dataset into training and testing sets. Finally, print both the training and testing datasets.
5 Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Python ML library classes in the program.
6 Write a Python program to load (or create) a dataset with missing values, count missing values in each column, and handle them by replacing with NaN, the column mean, and the most frequent value. Finally, display the dataset before and after performing the replacements.
7 Write a Python program to load (or create) a dataset, split it into training and testing sets, and apply feature scaling to the data. Finally, display the dataset before and after scaling to observe the changes.
8 Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions. Python ML library classes can be used for this problem.
9 Implement the working principle of Artificial Neural Network with feed forward and feed backward principle.
10 Implement and demonstrate the working of SVM algorithm for classification.

References:

1. Kevin P. Murphy, “Machine Learning: A Probabilistic Perspective”, MIT Press, 2012.
2. C. M. Bishop. Pattern Recognition and Machine Learning. First Edition. Springer, 2006. (Second Indian Reprint, 2015).
3. Tom M. Mitchell, “Machine Learning”, McGraw-Hill, 2010
4. P. Flach. Machine Learning: The Art and Science of Algorithms that Make Sense of Data. First Edition, Cambridge University Press, 2012.

Course Code	Course Name	L	T	P	Cr
DASH250099	Social Networking and Mining	3	1	0	4

Course Outcomes:

CO1. Apply graph theory concepts and use tools like Python, NetworkX, and Gephi to model and analyze real-world social networks.
CO2. Compute network metrics, detect communities, predict links, simulate information diffusion, and identify anomalies in complex networks.
CO3. Develop and implement graph representation learning and deep learning models (such as Graph Neural Networks) using frameworks like PyTorch Geometric.
CO4. Critically assess ethical concerns related to privacy, fairness, and transparency in social network mining.
CO5. Demonstrate proficiency in social network analysis through hands-on projects using real-world datasets.

Module 1:

Foundations of Social Network Analysis:
Introduction to Social Networks and their applications,Graph theory basics: nodes, edges, types of graphs,Basic network modeling using Python and NetworkX,Network Measures: Degree, Betweenness, Closeness, Eigenvector Centrality,Clustering Coefficient, Diameter, Density

Module 2:

Structural Patterns and Community Detection:Network Growth Models:Erdos-Rényi, Barabási-Albert, Watts-Strogatz,Link Analysis:PageRank, HITS, Eigenvector centrality, Community Detection – Part I & II:Modularity, Louvain Method,Label Propagation, Spectral Clustering

Module 3:

Predictive and Dynamic Network Analysis: Link Prediction Techniques:Common Neighbors, Jaccard, Adamic-Adar,Supervised Link Prediction,Network Dynamics & Cascades:Independent Cascade Model,Linear Threshold Model,Influence maximization,Anomaly Detection:Types of anomalies: point, structural, temporal,Outlier detection in graphs

Module 4:

Graph Representation and Deep Learning:Introduction to Deep Learning for Graphs,Graph Embeddings:DeepWalk, Node2Vec, Graph Neural Networks (GNNs):Graph Convolutional Networks (GCN):GraphSAGE, GAT,Node Classification & Link Prediction using GNNs

Module 5:

Applications, Ethics, and Case Studies:Real-world case studies:Twitter, Facebook, Citation networks, Political networks,Applications in recommendation, influence analysis, and fraud detection,Ethical issues in social network mining:Privacy, fairness, transparency,Conclusion and future trends

Labs / Practicals:

N/A

References:

1. Social Network Analysis, Tanmoy Chakraborty, Wiley
2. Network Science, Albert-Lazzlo Barabasi
3. Social Network Analysis: Methods and Applications, Stanley Wasserman, Katherine Faus

Course Code	Course Name	L	T	P	Cr
DCSA250009	Blockchain Technology	3	1	0	4

Course Outcomes:

Upon completion of the course, the learners will be able to:
CO1: Understand the concept of blockchain and its need.
CO2: Analyze methods of cryptography for application with blockchain.
CO3: Evaluate the working of blockchain.
CO4: Understand the underlying technology of transactions, blocks, proof-of-work, and consensus building.
CO5: Identify real world problems that blockchain can solve and analyze a use case.

Module 1:

Fundamentals of Blockchains Technology, The Structure of Blockchains, Blockchain Applications, The Blockchain Life Cycle, Distributed Ledger vs Centralized System, Merkel Root, Nonce Value, Hash Algorithm, and its applications.

Module 2:

Block Ciphers; Encryptions; Secret Keys; Elliptic Curve Cryptography; Hash cryptography; Encryption vs hashing; Digital Signature; Memory Hard Algorithm, Zero Knowledge Proof.

Module 3:

Introduction: Transactions, blocks, hashes, consensus, verify and confirm blocks, peer to peer networks, blocks of data in a chain, decentralization of networks, processes & workflows, cryptocurrencies, nodes, assets, consensus; types of blockchain; chain policy; working of blockchain; life of blockchain application; privacy, anonymity and security of blockchain.

Module 4:

Hyperledger: Introduction, where can Hyperledger be used, Hyperledger architecture, Hyperledger Fabric, features of Hyperledger; Open source blockchain platform technology; Tools & services; Cloud options in blockchains AWS; Azure workbench; Hyperledger console; Consensus: Proof of work, proof of stake, delegated proof of stake, proof of burn, BlocBox Protocol.

Module 5:

History; Distributed ledger; Smart contracts; Cryptocurrency; Bitcoin protocols; Mining strategy and rewards; Ethereum – construction; DAO; GHOST; Vulnerability; Attacks; Sidechain; Namecoin

Labs / Practicals:

NA

References:

1. D. Tapscott, A. Tapscott, Blockchain Revolution: How the Technology Behind Bitcoin Is Changing Money, Business, and the World, Portfolio Publishers.
2. A. Antonopoulos, Mastering Bitcoin: Programming the Open Blockchain, O'Reilly.
3. P. Champagne, The Book of Satoshi: The Collected Writings of Bitcoin Creator Satoshi Nakamoto, e53 Publishing, LLC.
4. M. Swan, Blockchain: Blueprint for a New Economy, O'Reilly.

5. R. Wattenhofer, The Science of the Blockchain, Inverted Forest Publishing.
6. R. Modi, Solidity Programming Essentials: A beginner's guide to build smart contracts for Ethereum and blockchain, Pack

Course Code	Course Name	L	T	P	Cr
DCSE250064	Digital Forensics	3	0	2	4

Course Outcomes:

After completion of the course student will be able to:
CO1: Explain the fundamental principles, scope, and legal aspects of digital forensics and cybercrime investigation.
CO2: Identify, acquire, preserve, and analyze digital evidence from various electronic devices.
CO3: Use forensic tools and techniques to investigate data breaches, cyberattacks, and unauthorized access.
CO4: Apply appropriate methodologies for forensic analysis of file systems, memory, and network traffic.
CO5: Prepare forensic reports and present findings in a legally admissible format for use in judicial proceedings

Module 1:

Computer forensics fundamentals, Benefits of forensics, computer crimes,
computer forensics evidence and courts, legal concerns and private issues.

Module 2:

Understanding Computing Investigations - Procedure for corporate High- Tech investigations, understanding data recovery work station and software, conducting and investigations.

Module 3:

Data acquisition - Understanding storage formats and digital evidence, determining the best acquisition method, acquisition tools, validating data acquisitions, performing RAID data acquisitions, remote network acquisition tools, other forensics acquisition tools. Processing crimes and incident scenes, securing a computer incident or crime, seizing digital evidence at scene, storing digital evidence, obtaining digital hash, reviewing case.

Module 4:

Current computer forensics tools - software, hardware tools, validating and testing forensic software, addressing data-hiding techniques, performing remote acquisitions.

Module 5:

E-Mail investigations - investigating email crime and violations, understanding
E-Mail servers, specialized E-Mail forensics tool.

Labs / Practicals:

1. Experiment No.1: Introduction to Digital Forensics Tools
a. Objective: Explore basic features of tools like Autopsy, FTK Imager, or CAINE Linux.
b. Outcome: Understand forensic tool environments and evidence handling basics.
2. Experiment No.2: Disk Imaging and Cloning
a. Objective: Create a forensic image of a storage device using tools like FTK Imager or dd.
b. Outcome: Safely duplicate storage media for analysis while preserving integrity.
3. Experiment No.3: File Recovery and Deleted File Analysis
a. Objective: Recover deleted files from a disk image using forensic tools.
b. Outcome: Understand how deleted data can be recovered and interpreted.
4. Experiment No.4: Windows Registry Forensics
a. Objective: Analyze Windows Registry hives for user activity and configuration changes.
b. Outcome: Identify forensic artifacts such as recently opened files, USB history, etc.
5. Experiment No.5: Memory (RAM) Forensics
a. Objective: Capture and analyze a memory dump using Volatility framework.

b. Outcome: Extract running processes, network connections, and potential malware traces.
6. Experiment No.6: Email and Document Forensics
a. Objective: Extract and analyze email headers and metadata from digital documents.
b. Outcome: Trace sources, authorship, and timelines of digital communication.
7. Experiment No.7: Browser History and Internet Artifacts Analysis
a. Objective: Examine browser caches, history, cookies, and download logs.
b. Outcome: Track web activity and user behavior patterns.
8. Experiment No.8: Mobile Device Forensics
a. Objective: Extract data from Android/iOS devices using tools like MOBILedit or Cellebrite (if available).
b. Outcome: Analyze call logs, SMS, GPS data, and app usage.
9. Experiment No.9: Log File Analysis and Intrusion Detection
a. Objective: Analyze system and server log files for traces of intrusion or suspicious activity.
b. Outcome: Identify anomalies and unauthorized access attempts.

References:

1. Warren G. Kruse II and Jay G. Heiser, “Computer Forensics: Incident Response Essentials”, Addison Wesley, 2002.
2. Bill Nelson, Amelia Phillips, F.Enfinger and Christopher Stuart, “Guide to Computer Forensics and Investigations”, 2nd Edition. Thomson Course Technology, 2006.
3. John R. Vacca, “Computer Forensics: Computer Crime Scene Investigation”, 2nd Edition, Charles River Media, 2005.
4. Bill Nelson, Amelia Phillips, F.Enfinger and Christopher Stuart, “Guide to Computer Forensics and Investigations”, 4th Edition., Thomson Course Technology, 2010.
5. Anthony T. S. Ho and Shujun Li, “Handbook of Digital Forensics of Multimedia Data and Devices”, IEEE Press, John Wiley & Sons, 2015.

Course Code	Course Name	L	T	P	Cr
DCSE250071	Drone Technology	3	1	0	4

Course Outcomes:

CO1: Apply the concept of Flight dynamics for building Quadcopter
CO2: Assemble and Program the Quadcopter
CO3: Perform Testing and Control operations on the Quadcopter
CO4: Implement Quadcopter for real world applications
CO5: Design and Develop the Drone

Module 1:

Flight Dynamics of Aerial Vehicles:
Definitions of Drone, UAV, RPA, Quad copters -Basic Components and Categories – Principles of Flight - Flight Maneuvers – Airframes - Creating a Frame: Materials, Different Frame Shapes
– Building Airframes - Flight dynamics - Applications - Future potential - Comparison with other aerial vehicles

Module 2:

Hardware Anatomy of Quadcopter:
Power Train – Propellers, Motors- Total Lift - Electronic Speed Controllers – Flight Battery – Radio transmitter and receiver – Flight Controller – GPS, Compass, Camera Assembling for Quad copter – Connectors, Mounting of Propellers and Powering up.

Module 3:

Testing And Maintenance of Quadcopter:
Key Flight Safety Rules - Preflight Checklist and Flight Log Information – Flight Instructions -
Repair and Maintenance: Crash analysis, Common issues, Voltage testing.
Test and troubleshoot Flight Controller Board (FCB), Electronic Speed Controller (ESC), and its associated peripherals.
Perform programming and configure the flight control board (FCB). Identify, explore, and test the interconnectivity of different peripherals with FCB. Establish connection of FCB with motor,
GPS, ESC, and sensors. Configure, test, and record FCB with battery to monitor battery level and perform return to home operation Perform and carry out drone leveling using IMU sensor.
Perform calibration of the compass, Lidar, and gyro sensor. The test communication link between FCB and RF transceiver. Write and upload computer code to FCB to test sensor results.
Test and record data of motor connectivity with ESC. Perform motor rotation using FCB and ESC. Test signal flow into the drone to test ESC parameters on FCB to check its operation. Write and upload computer code to FCB to ESC working.

Module 4:

Real World Applications and Case Studies
Beneficial Drones, Aerial Photography, Mapping and Surveying, Precision Agriculture, Search and Rescue, Infrastructure Inspection, and Conservation. Case Studies: Agriculture Weed Classification, Microdrone surveillances

Module 5:

Drone Technology
1. Familiarization with Drone Parts
2. Assembling of Drone

3. Preparation for Drone for Flight, making flight plan and basic drone flight training
4. Debugging and repairing of the drone
5. Operation of Drone for different Applications

Labs / Practicals:

N/A

References:

1. Reg Austin "Unmanned Aircraft Systems UAV design, development and deployment", Wiley, 2010.
2. Robert C. Nelson, Flight Stability and Automatic Control, McGraw-Hill, Inc, 1998.
3. Kimon P. Valavanis, "Advances in Unmanned Aerial Vehicles: State of the Art and the Road to Autonomy", Springer, 2007
4. 4. Paul G Fahlstrom, Thomas J Gleason, "Introduction to UAV Systems", UAV Systems, Inc,
5. 1998
6. 5. Dr. Armand J. Chaput, "Design of Unmanned Air Vehicle Systems", Lockheed Martin
7. Aeronautics

Course Code	Course Name	L	T	P	Cr
DCSE250081	Fuzzy Logic and Fuzzy Sets	3	1	0	4

Course Outcomes:

CO1: Understand and explain the fundamentals of fuzzy sets, including operations, properties, and extensions.
CO2: Apply fuzzy logic and approximate reasoning to model uncertainty and imprecision in real-world scenarios.
CO3: Implement fuzzy inference systems for decision-making and control applications.
CO4: Develop and apply fuzzy clustering and pattern recognition algorithms to analyze complex data.
CO5: Integrate fuzzy logic concepts with conventional systems to enhance performance in intelligent applications.

Module 1:

Fundamentals of Fuzzy Sets: Basic concepts on fuzzy sets, Basic Types, Fuzzy sets versus crisp sets, Properties of alpha-cuts, Representation of fuzzy sets, Extension principle.

Module 2:

Operations on Fuzzy Sets and Relations: Operation on fuzzy sets, Fuzzy union, intersection and complement, combinations of operations, Fuzzy arithmetic – Fuzzy numbers, Arithmetic operations on fuzzy numbers, Fuzzy relations, Binary fuzzy relations, Fuzzy equivalence and compatibility relations, Fuzzy ordering relations, Fuzzy morphism.

Module 3:

Fuzzy Measures and Fuzzy Logic: Fuzzy measures, Belief and possibility measures, Evidence theory, Possibility theory versus Probability theory, Fuzzy logic, Multivalued logic, Fuzzy propositions, Fuzzy qualifiers.

Module 4:

Approximate Reasoning and Fuzzy Expert Systems: Approximate reasoning – Fuzzy expert system (an overview), Fuzzy implications, Selection of fuzzy implication, Multiconditional approximate reasoning.

Module 5:

Fuzzy systems: Fuzzy system & neural network, Fuzzy automata, Fuzzy clustering, Fuzzy pattern recognition.

Labs / Practicals:

NA

References:

1.S. N. Sivanandam, S. Sumathi and S. N. Deepa, Introduction to Fuzzy Logic using MATLAB, Springer.
2.S. N. Sivanandam, S. Sumathi and S. N. Deepa, Introduction to Fuzzy Logic using MATLAB, Springer.
3.Timothy J. Ross, Fuzzy Logic With Engineering Applications, Wiley, 3rd edition.

Course Code	Course Name	L	T	P	Cr
DCSE250086	Health Care and Data Analytics	3	1	0	4

Course Outcomes:

CO1. Describe the basics of healthcare data analytics.
CO2. Use biomedical image features and perform data analytics.
CO3. Apply natural language processing for data analytics
CO4. Use prediction models for healthcare data analytics.
CO5. Interpret genomic and clinical data for healthcare data analytics.
CO6. Explain data analytics applications for healthcare.

Module 1:

Introduction to Healthcare Data Analytics- Electronic Health Records– Components of EHR- Coding Systems- Benefits of EHR- Barrier to Adopting HER, Challenges- Phenotyping Algorithms.

Module 2:

Biomedical Image Analysis
Biomedical image modalities, Object detection, image segmentation, image registration, feature extraction
Genomic Data Analysis for Personalized Medicine
Genomic data generation, methods for data analysis, types of computational genomics studies towards personalized medicine.

Module 3:

Data Analysis using Natural language Processing for healthcare.
Natural Language Processing, Mining information from Clinical Text, challenges of processing clinical reports, applications.

Module 4:

Clinical Prediction Models and Data Mining for Healthcare data
Basic Statistical Prediction Models, Alternative Clinical Prediction Models, Survival Models. Association Analysis, Temporal Pattern Mining: Sequential Pattern Mining, Time-Interval Pattern Mining.

Module 5:

Visual Analytics and Integrating data for Healthcare
Medical data visualization, visual analytics for public health and population research, clinical workflow, clinicians, patients.

Labs / Practicals:

NA

References:

1. Chandan K. Reddy and Charu C Aggarwal, “Healthcare data analytics”, Taylor & Francis, 2015
2. Hui Yang and Eva K. Lee, “Healthcare Analytics: From Data to Knowledge to Healthcare Improvement, Wiley, 2016.
3. Tinglong Dai, Sridhar Tayur , “Handbook of Healthcare Analytics”, Wiley, 2018.
4. Anand J Kulkarni, Patrick Siarry,”Big Data Analytics in healthcare”,Springer,Studies in Big Data

Course Code	Course Name	L	T	P	Cr
DCSE250087	Intelligent Systems	3	1	0	4

Course Outcomes:

CO1. Explain the core concepts and structure of intelligent systems, including expert systems and intelligent agents.
CO2. Apply search algorithms and reasoning techniques for problem-solving in intelligent systems.
CO3. Implement knowledge representation methods for building intelligent applications.
CO4. Utilize machine learning basics and AI tools to develop simple intelligent solutions.
CO5. Analyze and evaluate the performance and ethical implications of intelligent systems in real-world scenarios.

Module 1:

Knowledge Representation: Data and knowledge: Data representation and data items in traditional databases, Data representation and data items in relational databases. Rules: Logical operations, Syntax and semantics of rules, Data log rule sets, The dependence graph of data log rule sets, Objects, Frames, Semantic nets, Solving problems by reasoning: The structure of the knowledge base, The reasoning algorithm, Conflict resolution, Explanation of the reasoning.

Module 2:

Rule Based Systems: Forward reasoning: The method of forward reasoning, A simple case study of forward reasoning. Backward reasoning: Solving problems by reduction, The method of backward reasoning, A simple case study of backward reasoning, Bidirectional reasoning.
Search Methods: Depth-first search, Breadth-first search, Hill climbing search, A* search. Contradiction freeness: The notion of contradiction freeness, Testing contradiction freeness, The search problem of contradiction freeness. Completeness: The notion of completeness, Testing completeness, The search problem of completeness. Decomposition of knowledge bases: Strict decomposition, Heuristic decomposition.

Module 3:

Tools for Representation and Reasoning: The Lisp programming language: The fundamental data types in Lisp, Expressions and their evaluation, Some useful Lisp primitives, Some simple examples in Lisp, The Prolog programming language: The elements of Prolog programs, The execution of Prolog programs, Built-in predicates, and Some simple examples in Prolog. Expert system shells: Components of an expert system shell, Basic functions and services in an expert system shell.

Module 4:

Real-Time Expert Systems: The architecture of real-time expert systems: The real-time subsystem, The intelligent subsystem Synchronization and communication between real-time and intelligent subsystems: Synchronization and communication primitives, Priority handling and time-out. Data exchange between the real-time and the intelligent subsystems: Loose data exchange, The blackboard architecture. Software engineering of real-time expert systems: The software lifecycle of real time expert systems, Special steps and tool, An Example of A Real-Time expert System.

Module 5:

Qualitative Reasoning and Petri Nets: Sign and interval calculus, Qualitative simulation: Constraint type qualitative differential equations, The solution of QDEs: the qualitative simulation algorithm: Initial data for the simulation, Steps of the simulation algorithm, Simulation results. Qualitative physics, signed directed graph (SDG) models, The Notion of Petri nets, The firing of transitions, Special cases and extensions, The state-space of Petri nets The use of Petri nets for intelligent control, The analysis of Petri nets: Analysis Problems for Petri Nets, Analysis techniques.

Labs / Practicals:

N/A

References:

1. Katalin M. Hangos, Rozalia Lakner, Miklos Gerzson, “Intelligent Control Systems- An Introduction with Examples”, 1st Edition, Kluwer Academic Publishers, 2004.
2. Laxmidhar Behera, Indrani Kar, “Intelligent Systems and Control: Principles and Applications”, Oxford University Press,2009.
3. Teodorescu, Horia-Nicolai, Watada, Junzo, Jain, Lakhmi C., “Intelligent Systems and Technologies Methods and Applications”, 1st Edition, Springer publications, 2009.
4. Yung C. Shin and Chengying Xu, “Intelligent Systems - Modeling, Optimization and Control”, 1st Edition, CRC Press, Taylor & Francis Group, 2009.

Course Code	Course Name	L	T	P	Cr
DCSE250098	NoSQL Database	3	0	2	4

Course Outcomes:

CO1: Understand the limitations of relational databases and the need for NoSQL systems.

CO2: Identify different types of NoSQL databases and their appropriate use cases.

CO3: Design and model data for NoSQL databases.

CO4: Use MongoDB for CRUD operations, aggregation, indexing, and replication.

CO5: Gain hands-on experience with key-value, column-oriented, and graph databases.

CO6: Apply scalability and performance optimization techniques in NoSQL systems

Module 1:

Introduction to NoSQL Databases (9 Hours)
Overview of database systems, Why NoSQL? Limitations of RDBMS, Characteristics of NoSQL databases, BASE vs ACID, Types of NoSQL databases, CAP theorem and implications

Module 2:

Key-Value and Document-Based Databases (9 Hours)
Key-value stores: Redis, Riak, DynamoDB, Use cases and data modeling, Document stores: MongoDB, CouchDB, BSON format, collections, and documents, CRUD operations, schema design in MongoDB

Module 3:

Column-Family and Graph Databases (9 Hours)
Column-family stores: Apache Cassandra, HBase, Data modeling and partitioning, Graph databases: Neo4j, OrientDB, Graph theory basics, nodes, edges, properties, Cypher query language for graph traversal

Module 4:

Advanced MongoDB Operations (9 Hours)
Indexing and performance tuning, Aggregation framework and pipeline, Data validation and schema enforcement, Replication and sharding in MongoDB, Backup and restore operations

Module 5:

NoSQL in Practice and Case Studies (9 Hours)
Designing a NoSQL solution for real-time data, Scalability and consistency trade-offs, Polyglot persistence and database integration, Use cases in IoT, social networks, recommendation engines, Comparative analysis of NoSQL systems

Labs / Practicals:

1. Installation and setup of MongoDB and Redis

2. Creating and managing MongoDB databases and collections

3. Implementing CRUD operations and query optimization
4. Designing schema for a sample application
5. Aggregation operations in MongoDB
6. Working with Redis data types and operations
7. Introduction to graph queries using Neo4j and Cypher
8. Mini project using a NoSQL database for a real-world use case

References:

1. NoSQL Distilled by Pramod J Sadalage and Martin Fowler
2. MongoDB The Definitive Guide by Kristina Chodorow
3. Seven Databases in Seven Weeks by Luc Perkins, Eric Redmond
4. Designing Data-Intensive Applications by Martin Kleppmann
5. Official documentation for MongoDB, Redis, Cassandra, and Neo4j

Course Code	Course Name	L	T	P	Cr
DCSE250123	Web Analytics	3	0	2	4

Course Outcomes:

CO1: Understand key concepts, terminologies, and the foundational principles of web analytics.
CO2: Identify and interpret various web metrics and parameters used to assess user behavior and website performance.
CO3: Utilize web analytics tools and techniques to collect, analyze, and report website data effectively.
CO4: Analyze web traffic data to extract actionable insights for improving user experience and business conversions.
CO5: Apply web analytics in real-world scenarios to evaluate website performance, engagement, and conversion optimization

Module 1:

Web Analytics - Basics - Traditional Ways - Expectations - Data Collection
- Clickstream Data - Weblogs - Beacons - JavaScript Tags - Packet Sniffing -
Outcomes data - Competitive data - Search Engine Data.

Module 2:

Qualitative Analysis - Customer Centricity - Site Visits - Surveys - Questionnaires - Website Surveys - Post visits - Creating and Running- Benefits of surveys - Critical components of successful strategy.

Module 3:

Web Analytic concepts - URLS - Cookies - Time on site - Page views - Understand standard reports - Website content quality - Navigation reports (top pages, top destinations, site overlay). Search Analytics - Internal search, SEO and PPC - Measuring Email and Multichannel Marketing - Competitive intelligence and Web 2.0 Analytics - Segmentation - Connectable reports.

Module 4:

Search Engine Analytics: Analytics - Cookies - Accounts vs Property - Tracking Code - Tracking Unique Visitors - Demographics - Page Views & Bounce Rate Acquisitions - Custom Reporting.

Module 5:

Goals & Funnels - Filters - Ecommerce Tracking - Real Time Reports - Customer Data Alert - AdWords Linking - AdSense Linking -Attribution Modelling - Segmentation - Campaign Tracking - Multi-Channel Attribution

Labs / Practicals:

1. Experiment No.1: Introduction to Google Analytics
a. Objective: Set up a Google Analytics account and integrate it with a sample website.
b. Outcome: Understand the basic interface, dashboards, and real-time tracking.
2. Experiment No.2: Tracking Website Traffic and User Behavior
a. Objective: Analyze user sessions, bounce rate, page views, and average session duration.
b. Outcome: Interpret key traffic metrics to assess website performance.
3. Experiment No.3: Event Tracking and Goal Configuration
a. Objective: Set up events and conversion goals (e.g., button clicks, form submissions).
b. Outcome: Measure user interactions beyond page views.
4. Experiment No.4: Campaign Tracking using UTM Parameters
a. Objective: Create and analyze campaign URLs using UTM parameters.
b. Outcome: Monitor the effectiveness of marketing campaigns.
5. Experiment No.5: Creating and Interpreting Custom Dashboards and Reports
a. Objective: Design dashboards with relevant KPIs using Google Analytics.

b. Outcome: Visualize and report custom performance metrics.
6. Experiment No.6: Audience Segmentation and Analysis
a. Objective: Create segments based on location, device type, or behavior.
b. Outcome: Compare different user groups to understand behavior patterns.
7. Experiment No.7: Traffic Source Analysis
a. Objective: Analyze referral, direct, organic, and paid traffic sources.
b. Outcome: Understand which channels drive traffic and conversions.
8. Experiment No.8: A/B Testing using Google Optimize or Similar Tools
a. Objective: Conduct A/B tests to compare variations of a web page.
b. Outcome: Evaluate changes to optimize website performance.
9. Experiment No.9: Introduction to Heatmaps and Click Tracking
a. Objective: Use tools like Hotjar or Crazy Egg to visualize user interaction.
b. Outcome: Understand where users click, scroll, or drop off.

References:

1. Avinash Kaushik, "Web Analytics 2.0: The Art of Online Accountability and Science of Customer Centricity", 1st Edition, Sybex, 2009.
2. Michael Beasley, "Practical Web Analytics for User Experience: How Analytics can help you Understand your Users", 1st Edition, Morgan Kaufmann, 2013.
3. Magy Seif El-Nasr, Anders Drachen, Alessandro Canossa, "Game Analytics: Maximizing the Value of Player Data", 1st Edition, Springer, 2013.
4. Bing Liu, "Web Data Mining: Exploring Hyperlinks, Content, and Usage Data", 2nd Edition, Springer, 2011.
5. Justin Cutroni, "Google Analytics", 1st Edition, O'Reilly, 2010.
6. Eric Fettman, Shiraz Asif, Feras Alhlou, "Google Analytics Breakthrough", John Wiley & sons, 2016.

Course Code	Course Name	L	T	P	Cr
DOAI250009	Big Data Analytics	3	0	2	4

Course Outcomes:

CO1: Describe big data characteristics, challenges, and applications across industries and domains.

CO2: Evaluate and apply NoSQL databases and distributed storage architectures for big data management.

CO3: Install, configure, and operate Hadoop and HDFS for batch-oriented big data processing.

CO4: Develop and implement data analysis workflows using Hadoop ecosystem tools including Hive, Pig, and HBase.

CO5: Design and implement real-time data processing and analytics using Spark Streaming and PySpark.

CO6: Apply visualization and regression techniques for analytical insights from large-scale data.

Module 1:

Introduction to Big Data Ecosystem

Big Data Characteristics: Volume, Variety, Velocity, Veracity, and Value, Business applications and case studies of big data analytics, Challenges in conventional systems, Big Data Analytics Lifecycle, Tools and Technologies in Big Data, Hadoop vs. Traditional Systems, NoSQL Overview (Key-Value, Document, Columnar, Graph databases), Basics of MongoDB and Cassandra for big data storage.

Module 2:

Hadoop Framework for Big Data

History and Evolution of Hadoop, Hadoop Distributed File System (HDFS): Architecture, Internals, and Operations, Block-level Storage, Fault Tolerance, MapReduce Paradigm: Concepts, Anatomy of a MapReduce Job, Failures, Scheduling, Shuffle, Sort, and Task Execution, Input / Output Formats in MapReduce, Hadoop Streaming and Integrations, Developing MapReduce Applications with Java and Python APIs.

Module 3:

Data Analysis with Hadoop Ecosystem

Hive: Data Modeling, Data Types, File Formats, HiveQL (DDL, DML, DQL), Querying Structured Data, Hive Services and Optimization.
Pig: Data Flow Language, Pig Latin Programming, Grunt Shell, Data Models, Operators, Joins, Developing and Testing Pig Scripts.
HBase: Architecture, Data Model, Integration with Hadoop, Use Cases.
Introduction to Data Lakes using Hadoop.
Hands-on with Hadoop Ecosystem for ETL pipelines.

Module 4:

Real-Time Big Data Analytics and Streaming

Stream Processing Fundamentals, Stream Data Models, Stream Architecture, Concepts of Sampling, Filtering, Counting in Streams, Real-time Sentiment Analysis, Stock Market Analysis using Streams, Introduction to Apache Kafka for Streaming Data Ingestion, Spark Basics: RDD, DataFrames, Spark SQL, In-Memory Computing, PySpark APIs, Spark Streaming for Real-Time Processing, Real-Time Analytics Applications, Case

Studies.

Module 5:

Analytical Models and Visualization for Big Data

Regression Techniques: Simple Linear Regression, Multiple Linear Regression, Model Interpretation for Large-Scale Data, Feature Engineering for Big Data, Visualization Techniques: Interactive Dashboards, Graphical Tools for Big Data (Tableau / PowerBI concepts), Visual Analytics: Trends, Patterns, Correlations, Interaction Models, Real-World Applications of Big Data Visualizations.

Labs / Practicals:

1. Setting up Hadoop, HDFS, and YARN on local and cloud-based environments (AWS / Azure / GCP).

2. Developing basic MapReduce jobs using Java and Python.

3. Implementing ETL pipelines using Hive and Pig.

4. Performing analytics with HBase and Hive.

5. Real-time data streaming with Apache Kafka and Spark Streaming (using PySpark).

6. Implementing basic ML pipelines over Spark (MLlib).

7. Visualizing Big Data using interactive tools (Matplotlib, Plotly, Tableau / PowerBI).

8. Group Project: Building and deploying an end-to-end Big Data Analytics pipeline on a real-world dataset.

References:

1. Tom White, Hadoop: The Definitive Guide, 4th Edition, O'Reilly Media, 2015.

2. Anand Rajaraman, Jeffrey David Ullman, Mining of Massive Datasets, Cambridge University Press, 3rd Edition, 2020.

3. Bill Franks, Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics, John Wiley & Sons, 2012.

4. Jiawei Han, Micheline Kamber, Jian Pei, Data Mining: Concepts and Techniques, 3rd Edition, Morgan Kaufmann / Elsevier, 2011.

5. Holden Karau, Andy Konwinski, Patrick Wendell, Matei Zaharia, Learning Spark: Lightning-Fast Big Data Analysis, 2nd Edition, O'Reilly Media, 2020.

6. Paul Zikopoulos, Dirk deRoos, Krishnan Parasuraman, Thomas Deutsch, James Giles, David Corrigan, Harness the Power of Big Data – The IBM Big Data Platform, Tata McGraw Hill, 2012.

7. Michael Berthold, David J. Hand, Intelligent Data Analysis, Springer, 2007.

8. Da Ruan, Guoqing Chen, Etienne E. Kerre, Geert Wets, Intelligent Data Mining, Springer, 2007.

Official Documentation (Highly Recommended for Labs / Practical Work):

9. Official documentation for Hadoop: https://hadoop.apache.org/

10. Official documentation for Spark: https://spark.apache.org/docs/latest/

11. Official documentation for Hive: https://cwiki.apache.org/confluence/display/Hive/Home

12. Official documentation for Pig: https://pig.apache.org/

13. Official documentation for HBase: https://hbase.apache.org/

14. Official documentation for Kafka: https://kafka.apache.org/documentation/

15. Official documentation for PySpark: https://spark.apache.org/docs/latest/api/python/

Course Code	Course Name	L	T	P	Cr
DOAI250041	Natural Language Processing	3	0	2	4

Course Outcomes:

CO1: Understand the history, stages, challenges, and applications of NLP in real-world systems.
CO2: Apply morphological analysis, finite state methods, stemming, and n-gram models for language processing.
CO3: Implement POS tagging, CFG parsing, and sentence-level syntactic analysis using statistical and ML-based methods.
CO4: Analyze word sense disambiguation techniques using machine learning and dictionary-based approaches.
CO5: Apply discourse analysis techniques including reference resolution, pronoun interpretation, and coherence modeling.

Module 1:

History of NLP; Generic NLP system; Levels of NLP; Knowledge in language processing problem; Ambiguity in natural language; Stages in NLP; Challenges of NLP; Role of machine learning; Brief history of the field; Applications of NLP: Machine translation, Question answering system, Information retrieval, Text categorization, text summarization & Sentiment analysis

Module 2:

Morphology analysis survey of English morphology, inflectional morphology & derivational morphology; Regular expressions; Finite automata; Finite state transducers (FST); Morphological parsing with FST; Lexicon free FST, Porter stemmer, N-Grams, N-gram language model, N-gram for spelling correction.

Module 3:

Part-of-Speech tagging (POS); Lexical syntax tag set for English (Penn Treebank); Rule based POS tagging; Stochastic POS tagging; Issues: Multiple tags & words, unknown words, class-based n-grams, HM Model ME, SVM, CRF; Context Free Grammar; Constituency; Context free rules & trees; Sentence level construction; Noun Phrase; Coordination; Agreement; Verb phrase & sub categorization

Module 4:

Attachment for fragment of English sentences, noun phrases, verb phrases, prepositional phrases; Relations among lexemes & their senses; Homonymy, Polysemy based disambiguation & limitations, Robust WSD; Machine learning approach and dictionary-based approach.

Module 5:

Discourse reference resolution; Reference phenomenon; Syntactic & semantic constraints on co reference; Preferences in pronoun interpretation; Algorithm for pronoun resolution; Text coherence; Discourse structure

Labs / Practicals:

Module 1: Introduction to NLP & Applications
1. Experiment 1: Demonstration of NLP Applications
o Implement basic NLP applications like sentiment analysis, text summarization, question answering, and information retrieval using pre-trained models or APIs.
2. Experiment 2: Text Preprocessing Techniques
o Perform tokenization, stop-word removal, stemming, and lemmatization on sample English texts.

Module 2: Morphology and Language Models
3. Experiment 3: Morphological Analysis using Stemming and Lemmatization
o Analyze inflectional and derivational forms of English words using stemmers and lemmatizers.
4. Experiment 4: Implementation of N-Gram Language Models

o Build unigram, bigram, and trigram models from a text corpus and compute probabilities of word sequences.
5. Experiment 5: Spelling Correction using N-Gram Probabilities
o Implement a simple spelling corrector using edit distance and N-gram based word probability estimation.

Module 3: POS Tagging and Grammar Parsing
6. Experiment 6: POS Tagging using Rule-Based and Statistical Methods
o Perform part-of-speech tagging using both rule-based and statistical approaches; evaluate accuracy on tagged corpora.
7. Experiment 7: Constituency Parsing using Context-Free Grammar (CFG)
o Define a CFG and parse given sentences to generate parse trees showing sentence structure.

Module 4: Word Sense Disambiguation
8. Experiment 8: Word Sense Disambiguation using Dictionary-Based Methods
o Use the Lesk algorithm to identify correct word senses in different sentence contexts.
9. Experiment 9: Word Sense Disambiguation using Supervised Learning
o Implement a supervised machine learning model to classify the correct sense of an ambiguous word using labeled data.

Module 5: Discourse and Reference Resolution
10. Experiment 10: Co-reference Resolution and Discourse Analysis
• Resolve pronouns and noun phrase references in a passage and analyze coherence and discourse structure.

References:

1. James Allen. Natural Language Understanding. The Benajmins/Cummings Publishing Company Inc. 1994. ISBN 0-8053-0334-0.
2. Tom Mitchell. Machine Learning. McGraw Hill, 1997. ISBN 0070428077.
3. Cover, T. M. and J. A. Thomas: Elements of Information Theory. Wiley. 1991. ISBN 0-471-06259-6.

Course Code	Course Name	L	T	P	Cr
DOAI250047	Recommender Systems	3	1	0	4

Course Outcomes:

CO1: Understand the fundamental concepts, types, and applications of recommender systems.
CO2: Analyze and differentiate between various recommendation approaches, including content-based, collaborative filtering, and hybrid methods.
CO3: Evaluate the performance and effectiveness of recommender systems using appropriate metrics and evaluation techniques.
CO4: Design and implement basic recommender system models for real-world datasets and application domains.

CO5: Apply ethical and practical considerations in the deployment and personalization aspects of recommender systems.

Module 1:

Foundations of Recommender Systems

Linear Algebra Notation: Matrix addition, multiplication, transposition, and inverses; covariance matrices, Introduction to Recommender Systems, Functions and applications, Challenges and limitations, Types of Recommender Systems, Content-based, collaborative, knowledge-based, and hybrid

Module 2:

Content-Based Filtering

High-Level Architecture of Content-Based Systems, Item representation techniques, User profile learning methods, Similarity computation (cosine, Jaccard), Advantages and drawbacks of content-based filtering, Case study: Book/movie recommender using TF-IDF

Module 3:

Collaborative Filtering Techniques

User-based nearest neighbor recommendation, Item-based nearest neighbor recommendation, Model-based collaborative filtering: Matrix Factorization (SVD, NMF), Preprocessing-based techniques, Attacks on collaborative filtering: types and countermeasures, Distributed and privacy-aware recommendation

Module 4:

Knowledge-Based and Hybrid Recommender Systems

Knowledge-Based Recommendation, Constraint-based and case-based reasoning, Rule-based recommendation systems, Hybrid Recommender Systems, Need and motivations for hybridization, Hybridization strategies: Monolithic, Parallel, Pipelined, Real-world hybrid systems (e.g., Netflix, Amazon)

Module 5:

Evaluation and Emerging Trends

Evaluation of Recommender Systems, Accuracy metrics: Precision, Recall, F1-score, MAE, RMSE, Evaluation methodologies: offline, online, user studies, Community and Social-Based Recommendations, Social tagging and folksonomies, Case study: BibSonomy and HeyStaks, Trust-based Recommender Systems, Trust modeling and integration in recommendation, Computational trust and robustness, Future Trends: Explainability, fairness, and deep learning in recommendations

Labs / Practicals:

Tutorial Topics (Problem Solving / Case Discussions):

1. Matrix Algebra for Recommender Systems.

2. Cosine & Jaccard Similarity Problem Solving.

3. User Profile and Item Profile Creation.

4. Comparative Case Studies: Content-Based vs Collaborative Filtering.

5. Hands-on Matrix Factorization Exercises (SVD, NMF).

6. Evaluation Metrics: Problems on Precision, Recall, F1, MAE, RMSE.

7. Diagrammatic Explanation of Hybrid Systems.

8. Emerging Issues: Explainability, Fairness, and Ethics Discussions.

References:

1. Jannach D., Zanker M. and FelFering A., Recommender Systems: An Introduction, Cambridge University Press (2011), 1st ed
https://www.researchgate.net/publication/235910467_Recommender_Systems

2. Ricci F., Rokach L., Shapira D., Kantor B.P., Recommender Systems Handbook, Springer (2011), 1st ed.
https://www.researchgate.net/publication/227268858_Recommender_Systems_Handbook

3. Erwin Kreyzig, Advanced Engineering Mathematics, 10th Edition, Wiley, 2011.
http://www.uop.edu.pk/ocontents/AdvancedEngineeringMathematics.pdf

4. Aggarwal, C. C. Recommender Systems: The Textbook. Springer 2016
https://pzs.dstu.dp.ua/DataMining/recom/bibl/1aggarwal_c_c_recommender_systems_the_textbook.pdf

Course Code	Course Name	L	T	P	Cr
DOEE250178	Robotics	3	0	2	4

Course Outcomes:

CO1: Explain the multidisciplinary fundamentals of robotics, including kinematics, sensing, actuation, and control from a systems engineering perspective.
CO2: Apply mathematical and algorithmic techniques to model and analyze robot motion and control strategies using Python.
CO3: Implement basic robotic behaviors and control systems using ROS and simulate them in a virtual environment such as Gazebo.
CO4: Integrate sensor data (e.g., laser, IMU, vision) and apply techniques for robot localization, mapping, and path planning.
CO5: Demonstrate ethical, social, and safety considerations in the design and deployment of intelligent robotic systems through project-based learning.

Module 1:

Genealogy of Artificial Beings: From Ancient Automata
to Modern Robotics, Teaching and Learning Robotics: A Pedagogical Perspective,

Module 2:

Design Thinking: From Empathy to Evaluation, Software Building Blocks; The Robot Operating System, Mathematical Building Blocks

Module 3:

Sensors, Actuators, and Algorithms; Control, Navigation and Path Planning; Localisation and Mapping; Kinematics, Dynamics and Architecture of Robot Arms

Module 4:

Multi-robot Systems: Bio-Inspired Concepts and Deployment Challenges; CAD/CAM and Prototyping; Principles of Interaction Design and User Studies

Module 5:

On the Safe Deployment of Robotic Systems; Managing the World Complexity: From Linear Regression to Deep Learning; Robot Ethics

Labs / Practicals:

Practical 1: Python Basics & ROS Setup: Write a Python program to launch a simple ROS node that publishes random pose coordinates. This covers ROS setup, Python scripting, message structuring, and basic ROS topic communication.

Practical 2: Robot Kinematics: Write a Python program to compute forward kinematics for a planar 2-link robotic arm. Use sympy for symbolic calculations and plot the reachable workspace.

Practical 3: Sensor Simulation & Data Logging: Write a Python program to subscribe to a simulated ROS sensor topic (e.g., laser scan) and log distance data into a CSV file. Analyze basic sensor readings programmatically.

Practical 4: Mobile Robot Control: Write a Python ROS node to implement a proportional (P) controller that drives a turtlebot inside Gazebo to reach specified (x, y) goals, using linear and angular velocity adjustments.

Practical 5: Localization via Odometry and Kalman Filter: Write a Python program to fuse wheel odometry with synthetic IMU data using a simple extended Kalman Filter in ROS. Visualize estimated vs. ground truth positions.

Practical 6: Path Planning with A* Algorithm: Write a Python script to plan a path in a 2D grid using the A* search algorithm, and then publish intermediate waypoints over ROS for a robot to follow.

Practical 7: Obstacle Avoidance Behavior: Write a Python ROS node that uses laser scan data to detect obstacles and trigger a reactive avoidance behavior (e.g., rotate and drive forward) when an object is within a threshold distance.

Practical 8: Robot Arm Motion Control with ROS MoveIt: Write a Python program to plan and execute a pick-and-place task using ROS MoveIt! on a simulated robotic arm (e.g., UR5 or Robot arm). Include scene setup and gripper control.

Practical 9: Multi-Robot Pub-Sub Coordination: Write two Python ROS nodes simulating two robots: one publishes its pose, and the other listens and follows maintaining a fixed distance. Useful for swarm or human-following demos.

Practical 10: Ethical Interaction — Voice Command Control: Write a Python ROS node that subscribes to microphone input, performs keyword spotting (e.g., “start”, “stop”) using speech-recognition libraries, and commands a mobile robot accordingly.

References:

1. Foundations of Robotics: A Multidisciplinary Approach with Python and ROS by Damith Herath, David St-Onge
2. Beginner's Guide to Python in Robotics, Davor S. Wright,
3. Robot Modeling and Control by Mark W. Spong, Seth Hutchinson, and M. Vidyasagar, Wiley, 2006.
4. Introduction To Robotics, Analysis, Systems , Application by Saeed B. Niku.
5. Modern Robotics: Mechanics, Planning, and Control by Kevin M. Lynch and Frank C. Park, Cambridge University Press, 2017.