COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE

PROGRAMME STRUCTURE & SYLLABUS
[2024 ADMISSION ONWARDS]

M.TECH. COMPUTER SCIENCE & ENGINEERING WITH SPECIALIZATION IN DATA SCIENCE & ARTIFICIAL INTELLIGENCE

SYLLABUS FOR
OUTCOME BASED EDUCATION

MASTER OF TECHNOLOGY (M.TECH.) COMPUTER SCIENCE & ENGINEERING WITH SPECIALIZATION IN DATA SCIENCE & ARTIFICIAL INTELLIGENCE

(2024 admission onwards)

Program Outcomes (PO) For The M.Tech. Computer Science & Engineering With Specialization in Data Science & Artificial Intelligence

After the completion of M.Tech. programme, the students will be able to:

PO1: Elicit deeper and current knowledge through research/exploration leading to development work with a motivation to solve practical problems.
PO2: Communicate effectively through well-written technical documentation as well as audio-visual Presentations.
PO3: Recognize the importance of entrepreneurship and innovation to create value and health.
PO4: Acquire mastery in the topic of study at an exceedingly higher level.

Program Specific Outcomes (PSO) For The M.Tech. Computer Science & Engineering With Specialization in Data Science & Artificial Intelligence

At the end of the programme students will be able to:

PSO1: Attain comprehensive understanding of advanced theories and models in Computer Science, Data Science and Artificial Intelligence.
PSO2: Design, implement, and evaluate AI models and systems for real-world applications in diverse domains.
PSO3: Realize data science pipeline by integrating data engineering, analytics, and visualization for enterprise solutions
PSO4: Enhance research skills and conduct independent research, which could lead to technological innovations and improvements in the field of AI and Data Science

DEPARTMENT OF COMPUTER SCIENCE
PROGRAMME STRUCTURE AND SYLLABUS (2024 ADMISSIONS)

M. TECH. COMPUTER SCIENCE & ENGINEERING
WITH SPECIALIZATION IN DATA SCIENCE & ARTIFICIAL INTELLIGENCE

Semester - I

Sl. No.	Course code	Course Title	Core / Elective	Credits	Lec	Lab/ Tutorial	Marks
1	24-479-0101	Mathematics for Computing	C	4	4	2	100
2	24-479-0102	Artificial Intelligence and Machine Learning	C	4	4	2	100
3	24-479-0103	Design and Analysis of Algorithms	C	4	4	2	100
4	-	Elective I	E	3	4	2	100
5	-	Elective II	E	3	4	2	100
Total for Semester I				**18**	20	10	500
Electives							
24-479-0104: Advanced Optimization Techniques							
24-479-0105: Advanced Natural Language Processing							
24-479-0106: Digital Image and Video Processing							
24-479-0107: Mathematics for Machine Learning							
24-479-0108: Algorithmic Game Theory							
24-479-0109: Data Engineering, Analytics and Visualization							
24-479-0110: Algorithms for Modern Data Models							
Semester - II							
1	24-479-0201	Reinforcement Learning	C	4	4	2	100
2	24-479-0202	Applied Data Science	C	4	4	2	100
3	24-479-0203	Deep Learning	C	4	4	2	100
4	24-479-0204	Seminar	C	1	0	3	100
5	-	Elective III	E	3	4	2	100
6	-	Elective IV	E	3	4	2	100
Total for Semester II				**19**	20	13	600
Electives							
24-479-0205: Probabilistic Graphical Models							
24-479-0206: Bioinformatics							
24-479-0207: Large Language Models							
24-479-0208: Programming Massively Parallel Processors							
24-479-0209: Modelling Cyber Physical Systems							
24-479-0210: Foundations of Federated Learning							
24-479-0211: Image and Video Coding							
24-479-0212:Natural Language Processing with Deep Learning							
Semester - III							
1	24-479-0301	Elective – MOOC	E	2	0	10	100
2	24-479-0302	Internship *	C	1	0	0	100
3	24-479-0303	Dissertation & Viva Voce	C	15	0	20	100
Total for Semester III				**18**	0	30	300
Semester - IV							
1	24-479-0401	Dissertation & Viva Voce	C	**17**	0	30	100

Total credits for Degree : 72 **Total Mark: 1500**

*The students should complete the **Course 24-479-0302 : Internship** during the Vacation period (May-June)

24-479-0101: Mathematics for Computing

Core/Elective: **Core** Semester: **1** Credits: **4**

Course Description

This course introduces the study of mathematical structures that are fundamentally discrete in nature. The course is intended to cover the main aspects which are useful in studying, describing and modeling of objects and problems in the context of Linear Algebra, computer algorithms and programming languages.

Course Outcomes (CO)

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

CO	**Couse Outcome Statement**	**Cognitive Level**
CO1	Analyse the different methods for proving the correctness of the theorems and problems.	Analyse
CO2	Understand and apply the basic concepts of Linear Algebra.	Apply
CO3	Understand and apply the basic aspects of Descriptive statistics.	Apply
CO4	Understand and apply the fundamentals of probability theory.	Apply

Mapping with Program Outcomes

CO	PO1	PO 2	PO 3	PO 4	PSO 1	PSO 2	PSO 3	PSO 4
CO1	3	-	-	-	3	-	-	-
CO2	3	2	-	-	3	-	-	-
CO3	3	2	-	-	3	-	-	-
CO4	3	1	-	-	3	-	-	-

Course Content

1. Introduction – proofs – propositions – predicates and quantifiers – truth tables – first order logic – satisfiability – pattern of proof – proofs by cases – proof of an implication – proof by contradiction – proving iff – sets – proving set equations – Russell’s paradox – well-ordering principle – induction – invariants – strong induction – structural induction
2. Vectors-Coordinate system-vector addition-vector multiplication-Linear combinations, span, and basis vectors-Matrix multiplication as composition-Three-dimensional linear transformations-The determinant-Inverse matrices, column space and null space- Nonsquare matrices as transformations between dimensions-Dot products and duality-Cross products-Cross products in the light of linear transformations-Cramer's rule-Change of basis-Eigenvectors and eigenvalues-vector spaces
3. Descriptive statistics: histogram, sample mean and variance, order statistics, sample covariance, sample

covariance matrix – Frequentist statistics: sampling, mean square error, consistency, confidence intervals, parametric and non-parametric model estimation

4. Probability theory: probability spaces, conditional probability, independence – Random variables: discrete and continuous random variables, functions of random variables, generating random variables – Multivariate random variables: joint distributions, independence, generating multivariate random variables, rejection sampling – Expectation: Mean, variance and covariance, conditional expectation

5. Random process: definition, mean and autocovariance functions, iid sequences, Gaussian and Poisson process , random walk – Convergence of random process: types of convergence, law of large numbers, Central limit theorem, monte carlo simulation – Markov chains: recurrence, periodicity, convergence, markov-chain monte carlo- Gibbs sampling, EM algorithm, variational inference

References

1. Bronson, R., Costa, G.B., Saccoman, J.T. and Gross, D.,Linear algebra: algorithms, applications, and techniques.4e, 2023.
2. Eric Lehman, F Thomson Leighton, Albert R Meyer, Mathematics for Computer Science, 1e, MIT, 2010.
3. Susanna S. Epp, Discrete Mathematics with Applications, 4e, Brooks Cole, 2010.
4. Gary Chartrand, Ping Zhang, A First Course in Graph Theory, 1e, Dover Publications, 2012. in
5. John Tsitsiklis. 6.041SC Probabilistic Systems Analysis and Applied Probability. Fall 2013. Massachusetts Institute of Technology: MIT OpenCourseWare. https://ocw.mit.edu
6. Albert Meyer. 6.844 Computability Theory of and with Scheme. Spring 2003. Massachusetts Institute of Technology: MIT OpenCourseWare, https://ocw.mit.edu.
7. Michael Mitzenmacher and Eli Upfal; Probability and Computing, 2ed, Cambridge University Press, 2017

Online Resources: Course notes of Carlos Fernandez-Granda, DS-GA 1002: Probability and Statistics for Data Science https://cims.nyu.edu/~cfgranda/pages/DSGA1002_fall17/index.html

24-479-0102: Artificial Intelligence and Machine Learning

Core/Elective: **Core** Semester: **1** Credits: **4**

Course Description

Machine learning is programming computers to optimize a performance criterion using example data or past experience. This course is to discuss many methods that have their bases in different fields: statistics, pattern recognition, neural networks, artificial intelligence, signal processing, control, and data mining. Major focus of the course is on the algorithms of machine learning to help students to get a handle on the ideas, and to master the relevant mathematics and statistics as well as the necessary programming and experimentation.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Levels**
CO1	Understand and explain the different types of the learning process, and key ethical considerations.	Understand
CO2	Learn to effectively prepare data for machine learning models through data cleaning, feature selection, and dimensionality reduction.	Apply
CO3	Implement and interpret linear and non-linear regression models, while comparing various classification techniques including tree-based, kernel, and ensemble methods.	Apply
CO4	Gain practical knowledge in identifying data clusters using various algorithms and discovering hidden patterns through association rule learning.	Analyze
CO5	Understand the basic building blocks of neural networks, implement the backpropagation algorithm, and explore the concept of MDPs and Q-learning.	Apply

Mapping with Program Outcomes

CO	PO1	PO 2	PO 3	PO 4	PSO 1	PSO 2	PSO 3	PSO 4
CO1	3	-	-	-	3	-	-	-
CO2	3	-	-	-	3	3	2	-
CO3	3	-	-	2	3	3	2	-
CO4	3	-	-	2	3	3	2	-
CO5	3	-	-	2	3	3	2	-

Course Content

1. Introduction to AI - What is AI? A Brief History of AI - Different types of AI - Applications of AI - Problem Solving Methods – Heuristics. Knowledge Representation and Reasoning - Planning and Decision-Making: Ethics and Societal Impact of AI.
2. Machine Learning Fundamentals - Concept of Machine Learning: Definition, applications, types of learning (supervised, unsupervised, reinforcement) - Hypothesis Spaces and Inductive Bias - Learning Process- Machine Learning Ethics and Bias. Data Preprocessing and Feature Engineering: Data Representation - Data Preprocessing - Features and Types - Dimensionality Reduction – Feature Identification - Feature selection – Feature extraction - Feature Importance-High dimensional data and Manifolds.
3. Regression and Classification - Regression: Linear Regression – Non-Linear regression – evaluation metrics for regression– Classification: Binary, multi-class, and multi-label classification – lazy learners - tree-based techniques - kernel-based techniques - probabilistic techniques - and ensembled techniques – evaluation metrics for classification.
4. Clustering and Rule Mining - Clustering: Partitioning based – hierarchical based – density based– grid-based – model based - Rule mining: Apriori algorithm, FB Growth - association rules. Outlier Detection - LOF.
5. Artificial Neural Networks and Reinforcement Learning -Neural Networks: McCulloch-Pitts neurons, Hebb's networks, Hopfield networks, Boltzmann machines, Perceptrons, multilayer perceptrons, backpropagation. Reinforcement Learning: Markov Decision Processes (MDPs), Q-learning.

References

1. Ethem Alpaydin, Introduction to Machine Learning, 3e, MIT Press, 2014
2. Tom M. Mitchell, Machine Learning, McGraw Hill Education; 1e, 2017
3. Stephen Marsland, Machine Learning, An Algorithmic Perspective, 2e, CRC Press, 2015
4. Giuseppe Bonaccorso, Machine Learning Algorithms, 1e, Packt Publishing Limited, 2017
5. Ethem Alpaydin, Machine Learning- The New AI, MIT Press, 1e, 2016
6. Andrew Ng, Machine Learning Yearning, ATG AI (Draft version), 1e, 2018
7. Rohit Singh, Tommi Jaakkola, and Ali Mohammad.*6.867 Machine Learning.* Fall 2006. Massachusetts Institute of Technology: MIT OpenCourseWare, https://ocw.mit.edu
8. Andrew Ng, https://www.coursera.org/learn/machine-learning

24-479-0103: Design and Analysis of Algorithms

Core/Elective: **Core** Semester: **1** Credits: **4**

Course Description:

The course covers the foundational algorithms in depth. The course helps in understanding the working and complexity of the fundamental algorithms and to develop the ability to design algorithms to attack new problems.

Course Outcomes (CO)

CO	Course Outcome Statement	Cognitive Levels
CO1	Understand the basic concepts of design and analysis of fundamental algorithms.	Understand
CO2	Develop the ability to design algorithms to attack new problems.	Apply
CO3	Prove the correctness of algorithms.	Analyze
CO4	Develop the ability to analyze the complexity of algorithms.	Analyze
CO5	Understand Complexity classes, concepts of P and NP problems	Understand

Mapping with Program Outcomes

CO	PO1	PO 2	PO 3	PO 4	PSO 1	PSO 2	PSO 3	PSO 4
CO1	3	-	-	-	3	-	-	-
CO2	3	-	-	2	3	3	-	-
CO3	-	1	-	2	3	3	-	-
CO4	3	1	-	-	3	3	-	-
CO5	3	-	-	2	3	2	-	-

Course Content

1. Introduction to design and analysis of algorithms, models of computation, correctness proofs, insertion sort, computational complexity, Master theorem , proof of Master theorem, merge sort, heaps, heap sort, binary search, binary search trees.
2. Graph algorithms, BFS and DFS, Dijkstra’s algorithm, proof of correctness of Dijkstra’s algorithm, Complexity analysis of Dijkstra’s algorithm , Negative weight edges and cycles , Bellman-Ford algorithm, proof of correctness and complexity of Bellman-Ford, All pairs shortest paths, Floyd-Warshall algorithm, proof of correctness and complexity, Minimum Spanning Trees , Prim’s algorithm, Cut property, Kruskal’s algorithm, proof of correctness and complexity analysis of Kruskal's Algorithm, Maximum-Flow networks, Ford-Fulkerson method, proof of correctness and complexity, Edmonds-Karp algorithm
3. Probability review, Experiments, outcomes, events, Random variables, Expectation, Linearity of

Expectation, Indicator Random Variables, Hiring Problem, Quicksort , Best case and Worst case complexity, Randomized Quicksort , Average case complexity , Hashing, Chaining, Open Addressing, Universal Hashing, Perfect Hashing , Analysis of hashing operations

4. Dynamic Programming , Rod-cutting problem, Recursive formulation, Bottom-up reformulation of recursive algorithms, Optimal Substructure Property, Matrix chain multiplication, Complexity of dynamic programming algorithms, Sequence Alignment , Longest common subsequence, Greedy algorithms, Optimal substructure and greedy-choice properties , 0-1 and fractional Knapsack problems, Huffman coding
5. P vs NP, NP Hardness, Reductions, Travelling Salesman Problem, NP-Completeness, SAT, 2- SATand 3-SAT, Vertex Cover

References

1. Michale T Goodrich and Roberto Tamassia, Algorithm Design and Applications, Wiley, 2014
2. Thomas H. Cormen et al, Introduction to Algorithms, MIT Press; 4th edition 2022.
3. Jon Kleinberg, Eva Tardos, Algorithm Design, Pearson; 1st edition August 2013.
4. Robert Sedgewick, Kevin Wayne, Algorithms, Addison Wesley; 4th edition 2011.
5. Steven S. Skiena, The Algorithm Design Manual, Springer; 3rd ed. October 2020

24-479-0104: Advanced Optimization Techniques

Core/Elective: **Elective** Semester: **1** Credits: **3**

Course Description

Virtualization provides the benefit of reducing the total cost of ownership and improving the business agility. This course systematically introduces the concepts and techniques used to implement the major components of virtual servers behind the scene. It discusses the details on hypervisor, CPU scheduling, memory management, virtual I/O devices, mobility, and etc.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Levels**
CO1	Understand the basic concepts of optimization and its applications.	Understand
CO2	Understand the mathematical representation and classical methods for solving optimization problems.	Understand
CO3	Explain and demonstrate working principles of various population based optimization techniques.	Apply
CO4	Applied understanding of current approaches for practical problem solving.	Apply

Mapping with Program Outcomes

CO	PO1	PO 2	PO 3	PO 4	PSO 1	PSO 2	PSO 3	PSO 4
CO1	3	-	-	-	3	3	-	-
CO2	3	-	-	-	3	3	-	-
CO3	3	1	-	-	3	3	-	-
CO4	3	1	-	2	3	3	2	-

Course Content

1. Introduction to optimization- formulation of optimization problems-Review of classical methods- Linear programming-Nonlinear programming-Constraint optimality criteria-constrained optimization- Population based optimization techniques

2. Genetic Algorithm-Introduction-Working principle-Representation-selection-fitness assignment-reproduction-cross over-mutation-constraint handling-advanced genetic algorithms- Applications-Simulated Annealing-Selecting the parameters-Sufficiently near neighbour-Transition probabilities-Barrier avoidance

3. Differential Evolution-Introduction-Working principles-parameter selection-advanced algorithms in

Differential evolution-Biogeography-Based Optimization-Introduction-Working Principles- Algorithmic variations

4. Particle Swarm Optimization-Introduction- Working principles- Parameter selection- Neighborhoods and Topologies-Convergence-Artificial Bee Colony Algorithm-Introduction- Working principles- Applications- Cuckoo search based algorithm-Introduction- Working principles- Random walks and the step size- Modified cuckoo search

5. Hybrid Algorithms-Concepts- divide and conquer- decrease and conquer-HPABC-HBABC- HDABC-HGABC-Shuffled Frog Leaping Algorithm-- Working principles -Parameters- Grenade Explosion Algorithm-Working principle-Applications

References

1. Dan Simon, Evolutionary Optimization Algorithms, 1e, Wiley, 2013
2. Martins, Joaquim RRA, and Andrew Ning. Engineering design optimization. 1e, Cambridge University Press, 2021.
3. Rao, Singiresu S. Engineering optimization: theory and practice. 5e, John Wiley & Sons, 2019.

24-479-0105: Advanced Natural Language Processing

Core/Elective: **Elective** Semester: **1** Credits: **3**

Course Description:

Natural Language Processing (NLP) is a crucial technology in today's information age, with widespread applications across various sectors due to the centrality of language in human communication. In recent years, deep learning approaches, utilizing neural networks, have achieved remarkable success in numerous NLP tasks. These methods eliminate the need for traditional, task-specific feature engineering by employing end-to-end neural models. This course provides students with a comprehensive overview of the latest advancements in Natural Language Processing.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Levels**
CO1	Understand the fundamentals of NLP and apply it to do basic text processing.	Apply
CO2	Utilize word vectors like TF-IDF, PMI and word embeddings effectively in NLP tasks.	Apply
CO3	Apply various parsing techniques on English text, and evaluate their performance.	Analyze
CO4	Apply advanced architectures such as RNNs, LSTMs, and encoder-decoder models with attention for sequence modelling.	Apply
CO5	Apply NLP techniques in machine translation, question answering, and information retrieval.	Apply

Mapping with Program Outcomes

CO	PO1	PO 2	PO 3	PO 4	PSO 1	PSO 2	PSO 3	PSO 4
CO1	3	1	-	2	3	3	2	-
CO2	3	1	1	2	3	3	2	-
CO3	3	1	1	2	3	3	2	-
CO4	3	1	1	3	3	3	2	-
CO5	3	1	1	1	3	3	2	-

Course Content

1. Introduction to Natural Language Processing - Various stages of traditional NLP – Challenges - Basic Text

Processing techniques - Common NLP Tasks. N-gram Language Models - Naive Bayes for Text Classification, and Sentiment Analysis - Introduction to Neural Networks.

2. Word representations - Lexical Semantics, Vector Semantics, TF-IDF, Pointwise Mutual Information (PMI), Neural Word embeddings - Word2vec, GloVe. Contextual Word Embeddings. Evaluating Vector Models - Feedforward neural networks for text Classification
3. Linguistic Structures - Constituency Trees, Context-Free Grammars, Ambiguity, CKY Parsing, Dependency Parsing - Transition-Based Dependency Parsing, Graph-Based Dependency Parsing, Evaluation.
4. Sequence Modelling - Recurrent Neural Networks, RNNs as Language Models, RNNs for NLP tasks, Stacked and Bidirectional RNN architectures, Recursive Neural Networks, LSTM & GRU, Common RNN NLP Architectures, Encoder-Decoder Model with RNNs, Attention models.

5. NLP Applications - Machine Translation, Question Answering and Information Retrieval, Research perspectives in NLP, Introduction to Large Language Models

References

1. Dan Jurafsky and James H. Martin. Speech and Language Processing (2024 pre-release)
2. Jacob Eisenstein. Natural Language Processing
3. Yoav Goldberg. A Primer on Neural Network Models for Natural Language Processing
4. Ian Goodfellow, Yoshua Bengio, and Aaron Courville. Deep Learning
5. Delip Rao and Brian McMahan. Natural Language Processing with PyTorch.
6. Lewis Tunstall, Leandro von Werra, and Thomas Wolf. Natural Language Processing with Transformers

24-479-0106: Digital Image and Video Processing

Core/Elective: **Elective** Semester: **1** Credits: **3**

Course Description

The aim of this course is to inculcate a comprehensive knowledge about various Digital Image and Video Processing techniques. The objectives are to give an in-depth knowledge about the basic theory and algorithms related to Digital Image and Video Processing, provide awareness about the current technologies and issues, provide hands-on experience in using computers to process digital images and Videos using Python and OpenCV library.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Levels**
CO1	Understand the fundamental concepts of signal and image processing systems.	Understand
CO2	Evaluate the different spatial and frequency domain filters for image enhancement and restoration.	Analyze
CO3	Understand the color image fundamentals and apply the different filters on color images.	Apply
CO4	Analyze different image segmentation algorithms.	Analyze
CO5	Understand the different motion estimation and depth perception techniques.	Understand

Mapping with Program Outcomes

CO	PO1	PO 2	PO 3	PO 4	PSO 1	PSO 2	PSO 3	PSO 4
CO1	3	-	-	-	3	2	2	-
CO2	3	-	-	2	3	2	2	-
CO3	3	-	-	2	3	2	2	-
CO4	3	-	-	2	3	2	2	-
CO5	3	-	-	2	3	2	2	–

Course Content

1. Signals: Impulse Sequence - Exponential Sequence - Periodic Sequence. Linear Systems - Shift- Invariant systems - Linear Shift Invariant (LSI) systems – Convolution - Correlation. Image Transforms: Fourier Transform - Discrete Fourier Transform - Z- transform – KL Transform. Causal Systems - Random Signals - Stationary Process - Markov Process.
2. Intensity Transformation and Spatial Filtering: Intensity Transformation Functions. Histogram Processing: Histogram Equalization - Histogram Matching. Image enhancement: Arithmetic/Logic operations - Image

Subtraction - Image Averaging. Spatial Filtering: Smoothening Spatial Filters - Sharpening Spatial Filters - Laplacian Filter - Unsharp masking - High Boost Filter. Gradient operators: Edge detection filters. Frequency Domain Smoothening - Frequency Domain Sharpening Filters - Laplacian in Frequency domain - Homomorphic Filtering.

3. Image degradation/Restoration process model - Noise probability density functions - Spatial Filtering: Mean Filters - Order-statistics filter - Adaptive Filters - Periodic Noise Reduction –Band-reject filters - Band-pass filters - Notch filters. Inverse filtering - Wiener filtering - Performance measures. Color image processing: Color fundamentals - Color models – RGB, CMYK – HIS - Color image smoothening and sharpening – Color image histogram - Color edge detection.
4. Point and line detection - Hough Transform. Image Segmentation: Fundamentals – Thresholding – Otsu's optimum global thresholding - Region-based segmentation: Region growing - Region Splitting and Merging - Segmentation using Morphological Watersheds.
5. Color video processing: Video display - Composite versus component video - Progressive and interlaced scan. Motion estimation: Optical flow - pixel based motion estimation - block matching algorithm - deformable block matching algorithm - Global and region based motion estimation - multiresolution motion estimation - Feature based motion estimation. Stereo and multi-view sequence processing: Depth perception - Stereo imaging principle - Disparity estimation.

References

1. Rafael C. Gonzalez, Richard E. Woods, "Digital Image Processing", 4th Ed., Pearson, March 2017.
2. Anil K. Jain, "Fundamentals of Digital Image Processing", Pearson, 1st Ed., 1988.
3. William K. Pratt, "Digital Image Processing: PIKS Scientific Inside", John Wiley & Sons, 4th Ed., 2007.
4. Azriel Rosenfield, Avinash C. Kak, "Digital Picture Processing", Morgan Kaufmann, 2nd Ed., 1982.
5. Bernd Jahne, "Digital Image Processing", Springer, 6th Ed., 2005.
6. Yao Wang, Jorn Ostermann, Ya-Qin Zhang, "Video Processing and Communications", Pearson, 1st Ed., 2001.
7. Alan C. Bovik, "The Essential Guide to Video Processing", Academic PRess, 2nd Ed., 2009
8. A. Murat Tekalp, "Digital Video Processing", Prentice Hall, 2nd Ed., 2015.

24-479-0107: Mathematics for Machine Learning

Core/Elective: **Elective** Semester: **1** Credits: **3**

Course Description

The aim of this course is to inculcate a comprehensive knowledge about mathematical formalisms required to understand machine learning concepts. The course introduces in detail linear algebra, probability concepts, optimization, and some of the applications

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Levels**
CO1	Outline the fundamental concepts of linear algebra.	Understand
CO2	Illustrate matrix diagonalization.	Apply
CO3	Analyze the process of backpropagation.	Analyze
CO4	Apply Bayes theorem.	Apply
CO5	Analyze the gradient descent algorithm	Analyze
CO6	Examine linear programming problems.	Analyze
CO7	Build some of the basic machine learning applications.	Apply

Mapping with Program Outcomes

CO	PO1	PO 2	PO 3	PO 4	PSO 1	PSO 2	PSO 3	PSO 4
CO1	3	-	-	-	3	2	-	-
CO2	3	-	-	-	3	2	-	-
CO3	3	-	-	-	3	2	-	-
CO4	3	-	-	2	3	2	-	-
CO5	3	-	-	2	3	2	-	-
CO6	3	-	-	2	3	2	-	-
CO7	3	-	-	2	3	2	-	-

Course Contents:

1. Linear Algebra – vectors – matrices – systems of linear equations – vector spaces – linear independence –basis and rank – linear mappings – affine spaces – Norms – lengths and distances – angles and orthogonality– orthonormal basis – inner product of functions– orthogonal projections – rotations

2. Determinant and trace – eigenvalues and eigenvectors – cholesky decomposition – eigendecomposition and diagonalization – singular value decomposition – matrix approximation – Partial differentiation – gradients – gradients of vectors and matrices – higher order derivatives – backpropagation and automatic differentiation – multivariate Taylor series

3. Probability review – conditioning and independence – Bayes theorem – counting – discrete and continuous random variables – discrete and continuous probability distributions – Gaussian distribution – Bayesian inference – limit theorems – estimation – conjugacy and exponential family – inverse transform – sampling from distributions

4. Optimization – gradient descent – choosing the right step size – gradient descent with momentum – stochastic gradient descent – constrained optimization and Lagrange multipliers – convex optimization – linear programming – quadratic programming – Empirical risk minimization – probabilistic modeling and inference – directed graphical models

5. Applications: linear regression – parameter estimation – Bayesian Linear Regression – PCA – Maximum Variance Projections – Low-Rank Approximations – Gaussian mixture models – Parameter learning via maximum likelihood – EM Algorithm – Support Vector Machines – Separating Hyperplanes – Primal and Dual forms – The Kernel Trick

References:

1. Gilbert Strang, Linear Algebra and Learning from Data, Wellesley-Cambridge Press, 2019
2. Marc Peter Deisenroth et al., Mathematics for Machine Learning, 1e, Cambridge Press, 2020, Ebook: https://mml-book.com
3. Mehryar Mohri et al., Foundations of Machine Learning, 2nd Edition, The MIT Press, 2018
4. Gilbert Strang, Introduction to Linear Algebra, 5th Edition, Wellesley-Cambridge Press, 2016
5. James Stewart, Multivariable Calculus, 7th Edition, Cengage Learning, 2011
6. Dimitri P. Bertsekas, John N. Tsitsiklis, Introduction to Probability, 2nd Edition, Athena Scientific, 2008.
7. Morris H. DeGroot, Mark J. Schervish, Probability and Statistics, 4th Edition, Pearson, 2011

24-479-0108: Algorithmic Game Theory

Core/Elective: **Elective** Semester: **2** Credits: **3**

Course Description

Game theory is a branch of mathematics and economics which models interactions of agents as games. Algorithmic game theory is the intersection of game theory and computer science. This course introduces algorithmic game theory in an application-oriented manner.

Course Outcomes (CO)

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

CO	Course Outcome Statement	Cognitive Level
CO1	Explain the fundamental concepts of non-cooperative and cooperative game theory	Understand
CO2	Distinguish between standard game models and solution concepts.	Analyze
CO3	Illustrate a variety of advanced algorithmic techniques and complexity results for computing game theoretical solution concepts	Analyze
CO4	Identify rationale of decision making in games.	Understand
CO5	Apply solution concepts, algorithms, and complexity results to unseen games that are variants of known examples.	Apply
CO6	Compare the state of the art in some areas of algorithmic research, including new developments and open problems	Evaluate

Mapping with Program Outcomes

CO	PO1	PO 2	PO 3	PO 4	PSO 1	PSO 2	PSO 3	PSO 4
CO1	3	-	-	-	3	2	-	-
CO2	3	-	-	-	3	2	-	-
CO3	3	-	-	-	3	2	-	-
CO4	3	-	-	2	3	2	-	-
CO5	3	-	-	2	3	2	-	-
CO6	3	-	-	-	3	2	-	1

Course Content:

1. Introduction to game theory – strategies, costs, payoffs – solution concepts – finding equilibria – games with sequential moves – games with simultaneous moves – discrete strategies, continuous strategies – mixed strategies – games with incomplete information – expected payoffs – Prisoner’s dilemma and

repeated games – Nash equilibrium – Computational complexity of Nash equilibrium

2. Games on networks – congestion games – selfish routing – Nash and wardrop equilibria for networks – price of anarchy – pricing network edges – network design with selfish agents – economic aspects of internet routing
3. Epistemic game theory – Modeling knowledge – rationality and belief – common belief in rationality – game strategies and perfect recall – cryptography and game theory – modeling cryptographic algorithms as games – multi-party computations – MPC and games
4. Mechanism design – general principles – social choice – incentives – algorithms mechanism design – distributed aspects – cost-sharing mechanisms – mechanism design without money – house allocation problem – stable matchings
5. Voting – evaluation of voting systems – strategic manipulation of votes – auctions – types of auctions – winner's curse – bidding strategies – fairness in auctions

References

1. Avinash K. Dixit et al., Games of Strategy, 4e, W. W. Norton & Company, 2014
2. Noam Nisan et al., Algorithmic Game Theory, 1e, Cambridge University Press, 2007
3. Steven Tadelis, Game Theory: An Introduction, 1e, Princeton University Press, 2013.
4. Michael Maschler, et al., Game Theory, 1e, Cambridge University Press, 2013.
5. Andres Perea, Epistemic Game Theory: Reasoning and Choice, 1e, Cambridge University Press, 2012

24-479-0109: Data Engineering, Analytics and Visualization

Core/Elective: **Elective** Semester: **1** Credits: **3**

Course Description

This course offers an integrated approach to understanding the lifecycle of data within modern organizations. It explores the foundational concepts, techniques, and tools necessary to engineer, analyze, and visualize data effectively for informed decision-making. Students will gain practical skills in data engineering, data analytics, and data visualization through hands-on projects and case studies.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Understand the Fundamentals of Data Engineering	Understand
CO2	Describe the life cycle phases of Data Analytics through discovery, planning and building.	Understand
CO3	Understand and apply Data Analysis Techniques.	Apply
CO4	Implement various Data streams.	Apply
CO5	Understand item sets, Clustering, frame works & Visualizations.	Understand
CO6	Understand the Data Visualizations	Understand

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	-	-	3	2	-	-
CO2	3	-	-	-	3	2	-	-
CO3	3	-	-	-	3	2	-	-
CO4	3	-	-	2	3	1	-	-
CO5	3	-	-	2	3	2	-	-
CO6	3	-	-	-	3	2	-	-

Course Content:

1. Introduction to Data Engineering: Overview of Data Engineering: Role, importance, and challenges, Data Lifecycle: Ingestion, storage, processing, analysis, and visualization. Data Storage Technologies: Relational Databases, NoSQL Databases, Data Warehouses, Distributed File Systems.

2. Data Processing Technologies: Batch Processing: MapReduce, Apache Spark, Stream Processing: Apache

Kafka, Apache Flink, Data Integration and ETL: Data Transformation: Data cleaning, enrichment, aggregation, and denormalization.

3. Introduction to Data Analytics: Sources and nature of data, classification of data (structured, semi-structured, unstructured), characteristics of data, introduction to Big Data platform, need of data analytics, analysis vs reporting, modern data analytic tools, applications of data analytics. Data Analytics Lifecycle: Need, key roles for successful analytic projects, various phases of data analytics lifecycle – discovery, data preparation, model planning, model building, communicating results, operationalization.

4. Data Analysis: Regression modeling, multivariate analysis, Bayesian modeling, inference and Bayesian networks, support vector and kernel methods, analysis of time series: linear systems analysis & nonlinear dynamics, rule induction, neural networks: learning and generalisation, competitive learning.

5. Introduction to Visualization and Stages – Computational Support – Issues – Different Types of Tasks – Data representation – Limitation: Display Space- Rendering Time – Navigation Links

References

1. Mobasher et al."Data Engineering: Mining, Information, and Intelligence" Springer, 2010
2. Michael Berthold, David J. Hand, Intelligent Data Analysis, Springer
3. Anand Rajaraman and Jeffrey David Ullman, Mining of Massive Datasets, Cambridge University Press
4. Bill Franks, Taming the Big Data Tidal wave: Finding Opportunities in Huge Data Streams with Advanced Analytics, John Wiley & Sons.
5. Michael Minelli, Michelle Chambers, and Ambiga Dhiraj, "Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses", Wiley
6. Jiawei Han, Micheline Kamber “Data Mining Concepts and Techniques”, Second Edition, ElsevierRobert Spence, “Information Visualization Design for Interaction”, Second Edition, Pearson Education, 2006.
7. Paul Zikopoulos, Chris Eaton, Paul Zikopoulos, “Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data”, McGraw Hill
8. Anil Maheshwari, “Data Analytics”, McGraw Hill Education

24-479-0110: Algorithms for Modern Data Models

Core/Elective: **Elective** Semester: **1** Credits: **3**

Course Description:

There exist both algorithmic and statistical challenges in modern large-scale applications and data analysis. This course describes the randomization and probabilistic techniques for modern computer science, with applications ranging from combinatorial optimization and machine learning to communication networks. The course covers the core material to advanced concepts. Also the emphasis is on methods useful in practice.

Course Outcomes (CO)

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

CO	Course Outcome Statement	Cognitive Level
CO1	Relate the advanced concepts of probability theory and modern applications.	Analyze
CO2	Explain the uncertainty in prediction due to intervention of random variables.	Analyze
CO3	Examine graph models and their algorithms	Analyze
CO4	Analyze evolutionary algorithms.	Analyze
CO5	Interpret algorithms for evolving data streams	Analyze

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	-	2	3	2	-	-
CO2	3	-	-	2	3	-	-	-
CO3	3	-	-	2	3	2	-	-
CO4	3	-	-	2	3	2	-	-
CO5	3	-	-	2	3	1	-	-

Course Content:

1. Probability: Expectations - Tail Bounds - Chernoff Bound – Balls and Bins – Probabilistic Method – Markov chains and Random walks
2. Entropy, Randomness, and Information: Measure of randomness – Monte Carlo Method – Markov Chain Monte Carlo Method
3. Graph models and algorithms– Random graph Models- Algorithms for graph generation - Random graphs as models of networks, Power laws, Small world Phenomena
4. Components of evolutionary algorithms – Example applications – Genetic algorithms – Evolution strategies – Evolutionary programming

5. Sampling, sketching, data stream models, read-write streams, stream-sort, map-reduce - Algorithms in evolving data streams

References

1. Michael Mitzenmacher, Eli Upfal, Probability and Computing: Randomization and Probabilistic Techniques in Algorithms and Data Analysis, 2e, Cambridge University Press, 2017
2. Rajeev Motwani and PrabhakarRaghavan, Randomized Algorithms, Cambridge University Press; Reprint edition, 2010
3. S. Muthukrishnan, Data Streams: Algorithms and Applications, 1e, Now Publishers, 2005
4. Charu C. Aggarwal, Data Streams: Models and Algorithms, 1e, Springer, 2006
5. Agoston E. Eiben, J.E. Smith , Introduction to evolutionary computing, 1e, Springer, 2010

24-479-0201: Reinforcement Learning

Core/Elective: **Core** Semester: **2** Credits: **4**

Course Description

The course aims to introduce the concepts reinforcement learning and to impart an understanding of how reinforcement learning -- along with supervised and unsupervised learning -- form a building block of modern artificial intelligence. The course will provide a solid introduction to the field of reinforcement learning and students will learn about the core challenges and approaches, including generalization and exploration.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Define the key features of reinforcement learning that distinguishes it from AI and non-interactive machine learning	Analyze
CO2	Demonstrate the ability to formulate a given problem as a reinforcement problem with all ingredients.	Apply
CO3	Implement in code common RL algorithms	Apply
CO4	Describe the exploration vs exploitation challenge	Understand
CO5	Compare and contrast at least two approaches for addressing the above challenge.	Analyze

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	-	-	3	2	-	-
CO2	3	-	-	-	3	2	-	-
CO3	3	-	-	2	3	3	-	-
CO4	3	-	-	2	3	-	-	-
CO5	3	-	-	2	3	-	-	-

Course Content:

1. The Reinforcement Learning problem: evaluative feedback, non-associative learning, Rewards and returns, Markov Decision Processes, Value functions, optimality and approximation
2. Bandit Problems: Explore-exploit dilemma, Binary Bandits, Learning automata, exploration schemes Dynamic programming: value iteration, policy iteration, asynchronous DP, generalized policy iteration
3. Monte-Carlo methods: policy evaluation, roll outs, on policy and off policy learning, importance sampling

Temporal Difference learning: TD prediction, Optimality of TD(0), SARSA, Q-learning, R- learning, Games and after states

4. Eligibility traces: n-step TD prediction, TD(lambda), forward and backward views, Q(lambda), SARSA(lambda), replacing traces and accumulating traces.
5. Function Approximation: Value prediction, gradient descent methods, linear function approximation, Control algorithms, Fitted Iterative Methods Policy Gradient methods: nonassociative learning - REINFORCE algorithm, exact gradient methods, estimating gradients, approximate policy gradient algorithms, actor-critic methods Hierarchical RL: MAXQ framework, Options framework, HAM framework, Option discovery algorithms

References

1. R. S. Sutton and A. G. Barto; Reinforcement Learning - An Introduction. 2e, MIT Press (2018) eBook: http://incompleteideas.net/book/the-book-2nd.html
2. Marco Wiering and Martijn van Otterlo (Editors); Reinforcement Learning: State-of-the Art, Springer (2012)
3. Csaba Szepesvari; Algorithms for Reinforcement Learning, Morgan and Claypool Publishers (2010)
4. David Silver: https://www.davidsilver.uk/teaching/

24-479-0202: Applied Data Science

Core/Elective: **Core** Semester: **2** Credits: **4**

Course Description

This applied data science course empowers you to transform diverse data sets into valuable insights that solve real-world problems. Embark on a journey through the complete data science lifecycle, mastering each step from identifying questions to crafting impactful stories with your findings. Gain hands-on experience with cutting-edge techniques while developing the ethical compass to use data responsibly across various industries.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Navigate the Data Science Landscape: Grasp applications across industries, master the data science lifecycle.	Understand
CO2	Master Data Preparation & Engineering: Implement warehousing concepts, build efficient data pipelines, ensure data quality.	Apply
CO3	Analyze Data with Confidence: Utilize descriptive statistics, hypothesis testing, regression models, classification algorithms, and clustering techniques for insightful analysis.	Apply
CO4	Craft Compelling Data Visualizations: Design effective visuals using best practices and popular tools, create impactful narratives to communicate findings.	Apply
CO5	Embrace DataOps & Big Data Technologies: Understand DataOps automation, implement CI/CD practices, explore big data technologies like Hadoop, Spark, and Kafka, and analyze challenges and opportunities presented by big data.	Analyze

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	-	-	3	1	-	-
CO2	3	-	-	2	3	2	-	-
CO3	3	-	-	2	3	2	-	-
CO4	3	1	-	2	3	2	-	1
CO5	3	1	-	2	3	2	-	-

Course Content:

1. Introduction to Applied Data Science - Overview of the data science landscape and its applications in various industries- The data science lifecycle: problem identification, data collection, data cleaning and preprocessing, analysis, modeling, and communication.- Ethical considerations in data science.
2. Data Warehousing and Engineering - Introduction to data warehousing concepts: dimensional modeling, star schemas, snowflake schemas, data marts. Data warehousing technologies: relational databases, data warehouses, data lakes.Data engineering: data extraction, transformation, and loading (ETL) processes, data pipelines, data quality management.
3. Data Analytics - Descriptive statistics: measures of central tendency, measures of dispersion, measures of association.-Inferential statistics: hypothesis testing, confidence intervals.- Regression analysis: simple linear regression, multiple linear regression- Classification: decision trees, random forests, support vector machines.-Clustering: k-means clustering, hierarchical clustering.
4. Data Visualization - Principles of data visualization: selecting the right chart type, using color effectively, labeling charts clearly. Tools for data visualization: Tableau, Power BI, matplotlib, seaborn. Storytelling with data visualization: how to create visual narratives that communicate insights effectively.
5. DataOps and Big Data- Introduction to DataOps: automating the data pipeline, continuous integration and continuous deployment (CI/CD), monitoring and alerting- Big data technologies: Hadoop, Spark, Kafka.- Challenges and opportunities of big data.

References

1. Applied Data Science with Python and Jupyter: Use powerful industry-standard tools to unlock new, actionable insights from your data; Alex Galea (2018); Packt Publishing. ISBN: 9781789951929.
2. Applied Data Science - Lessons Learned for the Data-Driven Business; Braschler, Stadelmann, Stockinger (Eds.); Springer(2019).
3. https://em360tech.com/sites/default/files/2020-08/DataOps%20Cookbook%202nd%20Edition%20FINAL.pdf
4. Data Warehousing and Analytics: Fueling the Data Engine; David Taniar, Wenny Rahayu;Springer Cham(2022); https://doi.org/10.1007/978-3-030-81979-8

24-479-0203: Deep Learning

Core/Elective: **Core** Semester: **2** Credits: **4**

Course Description:

Deep learning is part of a broader family of machine learning methods based on learning data representations, as opposed to task-specific algorithms. This course describes deep learning techniques used by practitioners in industry, including deep feedforward networks, regularization, optimization algorithms, convolutional networks, sequence modeling, and practical methodology. This course is useful to students planning careers in either industry or research, and for software engineers who want to begin using deep learning in their products or platforms

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Understand the need for Deep learning, Feed forward networks, Learning XOR, Gradient based Learning, Hidden units.	Understand
CO2	Differentiate between training error and generalization error, Underfitting and Overfitting. And Identify Regularization strategies, Dataset Augmentation, Adversarial Training.	Analyze
CO3	Describe the working of Convolution Operation, Sparse interactions, Parameter sharing, Equivariant representations, Pooling and Recurrent Neural Networks	Understand
CO4	Understand different types of Autoencoders, Undercomplete Autoencoders, Regularized Autoencoders, and Dimensionality Reduction.	Understand
CO5	Explain deep generative models like Boltzmann Machines, Restricted Boltzmann Machines.	Understand

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	-	-	3	-	-	-
CO2	3	-	-	2	3	-	-	-
CO3	3	1	-	2	3	-	-	-
CO4	3	-	-	2	3	-	-	-
CO5	3	1	-	2	3	-	-	-

Course Content:

1. Deep Networks: Feed forward networks – Learning XOR- Gradient based Learning – Hidden units –

Architecture design- Back propagation – Differentiation algorithms

2. Regularization for Deep Learning: Penalties-Constrained optimization-Under constrained problems- Dataset augmentation-Semi Supervised learning- Sparse representation- Adversarial training- Optimization for training deep models: Basic algorithms-Algorithms with adaptive learning rates

3. Convolutional Networks: Convolution-Pooling-Variants of pooling- Efficient convolutional algorithms – Recurrent and Recursive Nets: Recurrent Neural Networks-Deep Recurrent Networks- Recursive Neural Networks- Explicit memory
4. Linear Factor Models: Probabilistic PCA- ICA – Slow feature analysis – Sparse coding – Autoencoders: Undercomplete Autoencoders – Regularized Autoencoders- Learning Manifolds- Applications of Autoencoders – Representation learning
5. Deep generative models: Boltzmann Machines – RBM- Deep Belief Networks-Deep Boltzmann Machines- Convolutional Boltzmann Machines- Directed generative Nets

References

1. Nithin Buduma, Nikhil Buduma and Joe Papa, Fundamentals of Deep Learning, 2nd Edition, O’Reilly, 2022
2. Jon Krohn and Grant Beyleveld, Deep learning Illustrated, Addison-Wesley; 1st edition, 2019
3. M Gopal, Deep Learning, Pearson Education; 1st edition, 2022
4. Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, 1e, MIT Press, 2016
5. Josh Patterson and Adam Gibson, Deep Learning: A Practitioner's Approach, 1e, Shroff/O'Reilly, 2017

24-479-0204: Seminar

Core/Elective: **Core** Semester: **2** Credits: **1**

Course Description

The student has to prepare and deliver a presentation on a research topic suggested by the department before the peer students and staff. They also have to prepare a comprehensive report of the seminar presented.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Identify, read, and interpret an academic research article from the literature that is related to his/her academic area of interest and present it before the committee.	Analyze
CO2	Organize and communicate technical and scientific findings effectively in written and oral forms.	Apply
CO3	Demonstrate the academic discussion skills to emphasize, argue with clarity of purpose using evidence for the claims.	Evaluate

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	2	3	-	2	1	-	-	3
CO2	-	3	-	-	1	-	-	3
CO3	2	3	-	2	1	-	-	3

24-479-0205: Probabilistic Graphical Models

Core/Elective: **Elective** Semester: **2** Credits: **3**

Course Description

Probabilistic Graphical models (PGM) are a foundation for understanding many methods of artificial intelligence, machine learning and estimation. Machine learning provides algorithms for solving problems by using training data. This course will give insight into how to formulate problems so that machine learning can be used effectively. Building good models can help learn with less data by constraining the learning space. Bayesian models are at the heart of most estimation methods. Formulation of these models is the first step in developing an estimation algorithm. The estimation itself is in many cases just inference on the model given some evidence. Approximate inference techniques such as those covered in this course are important in solving many very hard estimation problems in science and engineering. Data scientists, machine learning enthusiasts, engineers, and those who curious about the latest advances in machine learning will find PGM interesting

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Demonstrate application of Probability and Graph Theory in reasoning.	Apply
CO2	Discuss how different graphs represent both factorization and independent relations.	Analyze
CO3	Utilize message passing algorithms for inference.	Apply
CO4	Examine methods for learning uncertainties in a model's parameters.	Analyze
CO5	Experiment with graph building tools.	Apply
CO6	Apply Bayesian networks and Markov networks to many real world problems.	Apply

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	-	-	3	-	-	-
CO2	3	-	-	-	3	-	-	-
CO3	3	-	-	2	3	-	-	-
CO4	3	-	-	2	3	-	-	-
CO5	3	-	-	2	3	2	-	-
CO6	3	-	-	2	3	2	-	-

Course Content:

1. Probabilistic reasoning: Representing uncertainty with probabilities – Random variables and joint distributions – Independence – Querying a distribution - Graphs
2. Representation: Bayesian Network (BN) representation – Independencies in BN – Factorizing a distribution – D-separation- Algorithm for D-separation – From distributions to Graphs
3. Undirected Graphical Models: Factor products – Gibbs distribution and Markov networks – Markov network independencies – Factor graphs – Learning parameters – Conditional Random Fields
4. Gaussian Network Models: Multivariate Gaussians – Gaussian Bayesian networks – Gaussian Markov Random Fields – Exact Inference: variable elimination- Sum-product and belief updates – The Junction tree algorithm
5. Learning: Learning Graphical Models – Learning as optimization – Learning tasks – Parameter estimation – Structure learning in BN – Learning undirected models – Actions and decisions

References:

1. Daphne Koller, Nir Friedman, Probabilistic Graphical Models- Principles and Techniques, 1e, MIT Press, 2009
2. Richard E. Neapolitan, Learning Bayesian Networks, 1e, Pearson, 2019
3. Christian Borgelt, Rudolf Kruse and Matthias Steinbrecher, Graphical Models- Methods for data analysis and Mining, 2e, Wiley, 2009
4. David Bellot, Learning Probabilistic Graphical Models in R, Packt Publishing, 1e, 2016
5. Luis Enrique Sucar, Probabilistic Graphical Models, 1e, Springer Nature, 2015
6. Coursera: https://www.coursera.org/specializations/probabilistic-graphical-models

24-479-0206: Bioinformatics

Core/Elective: **Elective** Semester: **2** Credits: **3**

Course Description

Present fundamental concepts from molecular biology, computational problems in molecular biology and some efficient algorithms that have been proposed to solve them.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Understand and appreciate basic concepts of molecular Biology and Human genome project.	Understand
CO2	Illustrate and explain various sequence alignment algorithms.	Apply
CO3	Demonstrate and evaluate different algorithms for identifying optimal phylogenetic trees.	Analyze
CO4	Understand the concepts of structure prediction in molecular biology.	Understand
CO5	Understand and demonstrate an algorithm in the literature for the domain.	Apply

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	-	-	3	-	-	-
CO2	3	-	-	2	3	-	-	-
CO3	3	1	-	-	3	-	-	-
CO4	3	-	-	2	3	-	-	-
CO5	3	1	-	2	3	2	-	2

Course Content

1. Bioinformatics introduction-Branches of bioinformatics-Basic concepts of molecular Biology-Proteins-Nucleic acids– genes and genetic synthesis – translation- transcription- protein Synthesis- Chromosomes-Maps and sequences- Biological databases
2. Sequence alignment-Concepts of alignment-Gap Penalty-Pairwise sequence alignment algorithms- Dot Matrix-Global & Local alignment-Multiple sequence alignment algorithms-Scoring matrices-PAM, BLOSUM-Heuristic Methods -BLAST-FASTA
3. Fragment Assembly of DNA - Biological Background-human genome project – Models - Algorithms - Heuristics - Physical Mapping of DNA - Internal Graph Models – Hybridization Mapping - Heuristics -

Genome rearrangements-Oriented Blocks- unoriented Blocks

4. Molecular Phylogeny-Phylogenetic Trees –Methods of phylogeny-Maximum Parsimony-Maximum Likelihood-Distance methods-Binary Character States- Perfect phylogeny
5. Molecular Structure Prediction- Secondary structure prediction-Protein Folding problems-Protein threading-Computing with DNA-Hamilton Path Problems-Computer aided Drug design- peptide drug-chemical drug

References

1. Rastogi, S. C., Parag Rastogi, and Namita Mendiratta. Bioinformatics: Methods and Applications-Genomics, Proteomics and Drug Discovery. PHI Learning Pvt. Ltd., 5e, 2022.
2. Neil James and Pavel A Pevzner, An introduction to Bioinformatics Algorithms, 4e, OUPress, 2014
3. ZhumurGhosh, BibekanandMallick , Bioinformatics : Principles and Applications, OUPress, 2015
4. Concord Bessant, Darren Oakley, Ian Shadforth, Building Bioinformatics Solutions, OUPress, 2014
5. Peter Clote and Rolf Backofen, Computational Molecular Biology-An introduction, 1e, Wiley Series, 2000

24-479-0207: Large Language Models

Core/Elective: Elective **Semester: 1** **Credits: 3**

Course Description:

This course provides an in-depth exploration of large language models, focusing on their architecture, applications, ethical considerations, and implications in various fields. Students will gain hands-on experience utilizing and fine-tuning large language models for multiple tasks. The course will also address the societal impact of these models and encourage critical thinking about their responsible use.

Prerequisites:

This course requires a basic understanding of Deep learning (DL) and Natural Language Processing (NLP) concepts. Proficiency in Python programming and Deep Learning frameworks like Pytorch or Keras is necessary.

Course Objectives:

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Understand the architecture and functioning of Large Language Models (LLMs).	Understand
CO2	Fine-tune pre-trained language models for various NLP tasks using Deep Learning tools	Apply
CO3	Design and generate prompts for generative LLMs to solve real-world challenges.	Apply
CO4	Critically assess the ethical implications and societal impact of using LLMs.	Analyze

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	-	3	3	-	-	-
CO2	3	-	-	3	2	3	-	-
CO3	3	-	-	3	-	3	2	-
CO4	3	-	-	3	-	2	-	-

Course Content:

1. *Large Language Models (LLM)* - Introduction, Evolution of LLM, Foundation models & Instruction-following LLM; Pre-training & Transfer learning; Solving Natural Language Processing (NLP) tasks using LLMs.
2. *Transformers* - Encoder-Decoder models, Attention Mechanism; *Architecture* - Self-attention, Multi-head attention, Layer Normalization, Positional encoding; Pre-training and fine-tuning of Transformer based models - Autoregressive models (BERT), Generative model (GPT) and Sequence to sequence model (T5).
3. *Tokenization techniques* - Word & Sub-word modeling, Viterbi algorithm, Wordpeice tokenizer, Sentencepeice tokenizer, Byte Pair Encoding (BPE); *Text Embeddings* - Searching, classification, Clustering; Similarity Between Words and Sentences; Semantic Search.
4. *Prompt Engineering* - Introduction to Generative AI, Prompt design, Types of Prompting; Controlling model output via parameters; Use Case Ideation, Creating Custom Generative Models, Chain-of-Thought Prompting, Prompt Attacks and Mitigation.
5. *Ethical and Societal Implications of LLMs* - Bais and Fairness, Privacy concerns, Ethical considerations, Misinformation, and Disinformation challenges, Mitigation strategies; *Case study*: Application of LLMs in various domains. *Mini Project* - Building applications from pre-trained LLMs for real-world scenarios.

References

1. Bommasani, Rishi, et al. "On the opportunities and risks of foundation models.", Center for Research on Foundation Models (CRFM), Stanford Institute for Human-Centered Artificial Intelligence (HAI), Stanford University.
2. Rogers, Anna, Olga Kovaleva, and Anna Rumshisky. "A primer in BERTology: What we know about how BERT works." Transactions of the Association for Computational Linguistics 8 (2021): 842-866.
3. Lin, Jimmy, et al. Pretrained Transformers for Text Ranking: BERT and Beyond. United States, Morgan & Claypool Publishers, 2021.
4. Pal, Ankit. "Promptify: Structured Output from LLMs." (2022) available at https://github.com/promptslab/Promptify

24-479-0208: Programming Massively Parallel Processors

Core/Elective: **Elective** Semester: **2** Credits: **3**

Course Description

It used to be the case that parallel computing was confined to giant supercomputers. But nowadays it is literally everywhere - even in the small mobile handsets that most of us carry around. This course introduces parallel computing with a strong emphasis on programming.

Course Outcomes (CO)

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

CO	Course Outcome Statement	Cognitive Level
CO1	Illustrate the parallel programming paradigm	Understand
CO2	Identify the benefits of GPU programming model.	Understand
CO3	Examine the CUDA programming architecture	Understand
CO4	Assess programs written for single-processor systems and convert them into efficient parallel programs	Analyze
CO5	Develop basic parallel programs in CUDA.	Apply
CO6	Apply parallel programming to real-world applications	Apply

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	-	-	3	1	-	-
CO2	3	-	-	3	3	2	-	-
CO3	3	-	-	3	3	2	-	-
CO4	3	-	--	3	3	2	-	-
CO5	3	-	-	3	3	2	-	-
CO6	3	-	-	3	-	3	-	2

Correlation level: 1= low, 2 = medium, 3 = High, '-' = no correlation

Course Content:

1. Introduction - parallel computing - more speed or parallelism - languages and models - sequential vs parallel - concurrent, parallel, distributed - parallel hardware architecture - modifications to the von Neumann Model.
2. Evolution of GPU - GPGPU - introduction to data parallelism - CUDA program structure - vector addition kernel - device global memory and data transfer

3. CUDA thread organization - mapping threads to multi-dimensional data - assigning resources to blocks - synchronization and transparent scalability - thread scheduling and latency tolerance
4. Memory access efficiency - CUDA device memory types - performance considerations - global memory bandwidth - instruction mix and thread granularity -floating point considerations

5 Parallel programming patterns - convolution - prefix sum - sparse matrix and vector multiplication - application case studies - strategies for solving problems using parallel programming.

References

1. David B. Kirk, Wen-mei W Hwu, Programming Massively Parallel Processors, 2e, Morgan Kaufmann, 2012
2. Peter Pacheco, Introduction to Parallel Programming, 1e, Morgan Kaufmann, 2011
3. Shane Cook, CUDA Programming: A Developer's Guide to Parallel Computing with GPUs, 1e, Morgan Kaufmann, 2012
4. Jason Sanders, Edward Kandrot, CUDA by Example: An Introduction to General-Purpose GPU Programming, 1e, AW Professional, 2010

24-479-0209: Modeling Cyber Physical Systems

Core/Elective: Elective **Semester: 2** **Credits: 3**

Course Description

Cyber-Physical Systems (CPS) is a new frontier for computer systems that is transforming the way people interact with engineered systems. CPS applications include systems such as aircraft, automotive, medical devices, process control, and critical infrastructure. Unlike the traditional computer systems, the interplay between the cyber and the physical systems in CPS brings significant challenges in the modeling, design, analysis and verification of such systems. The complex, interdisciplinary nature of CPS requires a unique approach for the education of CPS. This course introduces Modeling formalism of Cyber-Physical Systems (CPS), Modeling of physical and cyber systems, and software synthesis from these modeling formalisms.

Course Objectives

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Categorize the essential modeling formalism of Cyber-Physical Systems (CPS).	Understand
CO2	Analyze the functional behavior of CPS based on standard modeling formalism.	Analyze
CO3	Improve specific software CPS using existing synthesis tools.	Apply
CO4	Contrast CPS requirements based on operating system and hardware architecture constraints.	Analyze
CO5	Analyze and verify the correctness of CPS implementations against system requirements and timing constraints.	Analyze

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	-	-	3	1	-	-
CO2	3	-	-	-	3	2	-	-
CO3	3	-	-	3	3	3	-	-
CO4	3	-	-	-	3	2	-	-
CO5	3	-	-	3	3	3	-	-

Course Content:

1. Introduction to Cyber Physical System: Cyber physical system: Definition Applications, Design Process for Cyber Physical System: Modeling, Design, And Analysis: Modelling continuous dynamics, Newtonian Mechanics, Actor models, Properties that actors and the systems: Causal Systems, Memoryless Systems, Linearity and Time Invariance, Stability. Feedback control
2. Modeling Discrete Systems :Discrete Systems ,State, Finite-State Machines: Transitions, The occurrence of reaction, Update functions, Determinacy and Receptiveness, Extended State Machines, Nondeterministic

Finite State Machines , Behaviors and Traces

3. Hybrid Systems: Actor Model for State Machines, Continuous Inputs, State Refinements, Classes of Hybrid Systems: Timed Automata, Higher-Order Dynamics, Supervisory control
4. Composition of State Machines: Concurrent Composition: Side-by-Side Synchronous Composition Side-by-Side Asynchronous Composition, Shared Variables, Cascade Composition, General Composition, Hierarchical state machines
5. Concurrent Models of Computation : Structure of Models, Synchronous-Reactive Models: Feedback Models, Well-Formed and ill-Formed Models, Constructing a Fixed Point, Dataflow Models of Computation: Dataflow Principles, Synchronous Dataflow ,Dynamic Dataflow, Structured Dataflow, Process Networks, Timed Models of Computation: Time-Triggered Models, Discrete Event Systems, Continuous-Time Systems

References:

1. Edward Ashford Lee, Sanjit Arunkumar Seshia, Introduction to Embedded Systems - A Cyber-Physical Systems Approach, 2e, MIT Press, 2017
2. Rajeev Alur , Principles of Cyber-Physical Systems, 1e, MIT Press, 2015
3. Raj Rajkumar, Dionisio de Niz, Mark Klein, Cyber-Physical Systems, 1e, AW Professional, 2017
4. Peter Marwedel, Embedded System Design: Embedded Systems Foundations of Cyber- Physical Systems, and the Internet of Things, 3e, Springer, 2017

Online courses: Coursera: https://www.coursera.org/learn/cyber-physical-systems-1

24-479-0210: Foundations of Federated Learning

Core/Elective: **Elective** Semester: **2** Credits: **3**

Course Description

The course introduces Federated Learning (FL), the privacy preserving version of distributed machine learning. It explains the need for and the different types of FL. Also the important techniques to realise it are discussed.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Realize the significance of privacy concerns in the present world.	Understand
CO2	Differentiate between FL and distributed machine learning.	Analyze
CO3	Apply privacy preserving techniques in data processing and in particular in machine learning.	Apply
CO4	Recognize FL as a dominant research area and understand the current research questions.	Understand
CO5	Apply FL techniques to existing machine learning applications	Apply
CO6	Understand the communication and computation challenges associated with FL	Analyze

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	-	-	3	-	-	-
CO2	3	-	-	-	3	-	-	-
CO3	-	-	-	3	3	-	3	-
CO4	3	-	-	-	3	-	-	3
CO5	3	-	-	3	3	3	-	-
CO6	-	-	-	3	3	-	-	-

Course Content

1. Introduction: Motivation - Privacy and other issues, Legal aspects - Consumer Online Privacy Rights Act (USA), General Data Protection Regulation (EU) , Definition and Applications of FL, Cross device and Cross Silo models.
2. Background: Classification based on data partitioning: Horizontal, Vertical and Transfer FL, Research Works in FL, Open Source Projects. Privacy Preservation Techniques - Secure Multi-Party

Computation.

3. Horizontal FL: Definition, Architectures - The Client - Server architecture, The Peer to Peer architecture, Global Model Evaluation, The Federated Averaging Algorithm - Federated Optimization, The FedAvg Algorithm. Improvements - Communication Efficiency, Client Selection.
4. Vertical FL: Privacy Preservation Techniques - Homomorphic Encryption, Differential Privacy. Definition of Vertical FL, Architecture, An Algorithm of VFL - Secure Federated Linear Regression.
5. Federated Transfer Learning: Heterogeneous Federated Learning, Instance based transfer, Feature based transfer and Model based transfer. Security definition of a FTL system. An FTL Framework - Additively Homomorphic Encryption, The FTL Training Process, The FTL Prediction process

References

1. *Liu, Yang., Chen, Tianjian., Yu, Han., Yang, Qiang., Cheng, Yong. Federated Learning. United States: Morgan & Claypool Publishers, 2019.*
2. Kairouz, Peter, et al. "Advances and open problems in federated learning." *Foundations and Trends® in Machine Learning* 14.1–2 (2021): 1-210.
3. https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679
4. https://www.congress.gov/bill/116th-congress/senate-bill/2968/text

24-479-0211: Image and Video Coding

Core/Elective: **Elective** Semester: **2** Credits: **3**

Course Description

The aim of this course is to give a rigorous introduction into the fundamental concepts of data compression with strong emphasis on the mathematical techniques and its applications to image and video coding. The main objectives of the course are to understand how digital data can be compressed using either lossless or lossy techniques, to provide a strong mathematical background in the field of coding theory, to expose the students to the standard compression techniques used in various coding standards and to expose the students to the latest image and video coding standards.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Understand the mathematical preliminaries of lossless compression techniques and analyze the different entropy coding techniques.	Analyze
CO2	Analyze the different dictionary-based and context-based coding techniques.	Analyze
CO3	Understand the mathematical preliminaries of lossy compression techniques and analyze the different quantization-based compression techniques.	Analyze
CO4	Understand the different approaches for video compression.	Understand
CO5	Understand the important image and video compression standards.	Understand

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	-	3	3	2	-	-
CO2	3	-	-	3	3	2	-	-
CO3	3	-	-	3	3	2	-	-
CO4	3	-	-	-	3	2	-	-
CO5	3	-	-	-	3	2	-	-

Course Content

1. Introduction: Compression Techniques - Modeling and Coding. Mathematical Preliminaries for Lossless Compression: Information Theory – Models - Coding: Uniquely decodable codes - Prefix codes - Kraft-McMillan Inequality. Huffman Coding: Minimum Variance Huffman Codes - Length of Huffman Codes -

Adaptive Huffman Coding - Golomb codes - Rice codes - Tunstall codes. Arithmetic Coding: Integer Arithmetic Coding.

2. Dictionary Techniques: Static Dictionary - Digram coding - Adaptive Dictionary - LZ77 - LZ78 - LZW. Context-based Compression: Prediction with partial match - Burrows-Wheeler Transform – CALIC - Run-Length Coding – JBIG – JBIG2.

3. Mathematical Preliminaries for Lossy Coding: Distortion Criteria - Rate Distortion Theory. Scalar Quantization: Quantization problem - Uniform Quantizer - Lloyd-Max Quantizer - Adaptive Quantization - Non-uniform Quantization - Entropy-Coded Quantization. Vector Quantization: LBG Algorithm - Tree Structured and Structured Vector Quantizers. Differential Coding: Basic algorithm – DPCM. Transform Coding.

4. Content dependent video coding: Temporal prediction and Transform coding - Two dimensional shape coding - Joint shape and texture coding - Region based and object based video coding - Knowledge based video coding - Semantic video coding - Layered coding system - Scalable video coding.

5. Image Compression Standards: JPEG - JPEG 2000 - JPEG XR - JPEG-LS - JPEG XT - JPEG Pleno. Video Compression Standards: MPEG-4 - H.263 - H.264/AVC - H.265/HEVC - AVS China - Dirac.

References

1. Khalid Sayood, “Introduction to Data Compression”, Morgan Kaufmann Publishers, 4th Ed., 2012.
2. David Salomon, “Data Compression – The Complete Reference”, Springer, 4th Ed., 2006.
3. Alistair Moffat, Andrew Turpin, “Compression and Coding Algorithms”, Kluwer Academic Publishers, 1st Ed., 2002.
4. Vasudev Bhaskaran, Konstantinos Konstantinides, “Image and Video Compression Standards”, Kluwer Academic Publishers, 2nd Ed., 2003.
5. Mark Nelson, Jean-Loup Gailly, “The Data Compression Book”, John Wiley & Sons, 2nd Ed., 1995.
6. John Miano, “Compressed Image File Formats”, Addison Wesley Professional, 1st Ed., 1999.
7. Peter Wayner, “Compression Algorithms for Real Programmers”, Morgan Kaufmann, 1st Ed., 1999.
8. Yao Wang, Jorn Ostermann, Ya-Qin Zhang, "Video Processing and Communications", Pearson, 1st Ed., 2001.
9. Alan C. Bovik, "The Essential Guide to Video Processing", Academic PRess, 2nd Ed., 2009
10. A. Murat Tekalp, "Digital Video Processing", Prentice Hall, 2nd Ed., 2015.

24-479-0212: Natural Language Processing with Deep Learning

Core/Elective: **Elective** Semester: **2** Credits: **3**

Course Description

Natural language processing (NLP) is one of the most important technologies of the information age. Applications of NLP are everywhere because people communicate mostly everything in language: web searches, advertisements, emails, customer service, language translation, radiology reports, etc. Recently, deep learning approaches have obtained very high performance across many different NLP tasks. In this course, students will learn to implement, train, debug, visualize, and invent their neural network models. The course provides a deep excursion into cutting-edge research in deep learning applied to NLP

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Understand the neural network approach to learning and processing natural language data	Understand
CO2	Know advanced concepts in natural language processing	Understand
CO3	Learn to implement, train, debug, and visualize deep neural network models for language processing	Apply

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	-	2	3	3	-	-
CO2	3	-	-	2	3	3	-	-
CO3	3	-	-	2	3	3	2	-

Course Content

1. Word Vectors - Singular Value Decomposition - Skip-gram - Continuous Bag of Words (CBOW) - Negative Sampling- Distributed Representations of Words and Phrases and their CompositionalityEfficient Estimation of Word Representations in Vector Space - Advanced word vector representations- language models-softmax-single layer networks

2. Neural Networks and backpropagation for named entity recognition -A Neural Network for Factoid Question Answering over Paragraphs - Grounded Compositional Semantics for Finding and Describing Images with Sentences - Deep Visual-Semantic Alignments for Generating Image Descriptions-Recursive Deep Models for Semantic Compositionality over a Sentiment Treebank

3. Introduction to Tensorflow- Large - Scale Machine Learning on Heterogeneous Distributed Systems.

Recurrent neural networks for language modeling and Extensions of recurrent neural network language model-Opinion Mining with Deep Recurrent Neural Networks

4. GRUs and LSTMs for machine translation- Recursive neural networks for parsing- Parsing with Compositional Vector Grammars - Subgradient Methods for Structured Prediction-Parsing Natural Scenes and Natural Language with Recursive Neural Networks - Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank-Dynamic Pooling and Unfolding Recursive Autoencoders for Paraphrase Detection - Improved Semantic Representations From Tree-Structured Long Short-Term Memory Networks

5. Convolutional neural networks for sentence classification - Sequence to Sequence with Neural Networks - Neural Machine Translation by Jointly Learning to Align and Translate - Dynamic Memory Networks for NLP

References

1. Yoav Goldberg, Neural Network Methods for Natural Language Processing, Morgan & Claypool Publishers, 1ed, 2017
2. Ian Goodfellow, YoshuaBengo, Aaron Courville, Deep Learning, 1e, MIT Press, 2017
3. Nikhil Buduma and Nicholas Locascio, Fundamentals of Deep Learning: Designing NextGeneration Machine Intelligence Algorithms, 1e, Shroff/O'Reilly, 2017
4. Josh Patterson and Adam Gibson, Deep Learning: A Practitioner's Approach, 1e, Shroff/O'Reilly, 2017

24-479-0301: Elective - MOOC

Core/Elective: Elective Semester: 3 Credits: 2

Course Description

A credit-based MOOC course of a minimum of 12 weeks duration or three non-credit-based MOOC courses of 4 weeks duration from SWAYAM/NPTEL/any other platforms approved by the Department council.

24-479-0302: Internship

Core/Elective: **Core** Semester: **3** Credits: **1**

Course Description

A minimum 1 month internship from the institute/industry approved by the Department council. Internship should be completed during the May-June summer vacation.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Develop a holistic understanding and practical skills for professional and academic success.	Apply
CO2	Demonstrate enhanced capabilities in problem-solving, effective communication, entrepreneurial thinking, and advanced subject mastery.	Apply

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	3	3	3	3	3	3	-
CO2	3	3	3	3	3	3	3	1

24-479-0303: Dissertation & Viva Voce

Core/Elective: **Core** Semester: **3** Credits: **15**

Course Description

The dissertation work spans two semesters. Through the dissertation work, the student has to exhibit the knowledge in terms of engineering or technological innovation or research ability to solve the contemporary problem. On completion of the first part of the work, the student shall submit an interim dissertation report. The qualitative and quantitative results of the work will be evaluated through a viva- voce exam.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Demonstrates in-depth knowledge and thoughtful application through the detailed analysis of the chosen research problem.	Analyze
CO2	Assesses the gap in knowledge by acquiring knowledge about previous works, their interpretation, and application.	Analyze
CO3	Demonstrates the design of the proposed methodology and its merits.	Apply
CO4	Organizes the interim dissertation content with proper structure and sequencing.	Apply
CO5	Demonstrates academic discussion skills to emphasize, argue with clarity of purpose using evidence for the claims.	Evaluate

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	-	3	3	3	-	3
CO2	3	-	-	3	3	3	-	3
CO3	3	-	-	3	3	3	-	3
CO4	-	3	-	-	3	-	-	3
CO5	-	3	-	-	3	-	-	3

24-479-0401: Dissertation & Viva Voce

Core/Elective: **Core** Semester: **4** Credits: **17**

Course Description

The dissertation work spans two semesters. Through the dissertation work, the student has to exhibit the knowledge in terms of engineering or technological innovation or research ability to solve the contemporary problem. On completion of the work, the student shall submit a final dissertation report. The qualitative and quantitative results of the work will be evaluated through a viva-voce exam.

Course Outcomes (CO)

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

CO	**Course Outcome Statement**	**Cognitive Level**
CO1	Demonstrates in-depth knowledge and thoughtful application through the detailed analysis of the problem chosen for the study	Analyze
CO2	Assesses the gap by acquiring knowledge about the previous works, and its interpretation and application	Analyze
CO3	Demonstrates the design of the proposed methodology and its merits.	Apply
CO4	Organize the interim dissertation content with proper structure and sequencing	Apply
CO5	Demonstrate the academic discussion skills to emphasize, argue with clarity of purpose using evidence for the claims.	Evaluate
CO6	Show ability to evaluate and reflect on critical questions.	Evaluate

Mapping with Program Outcomes

	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3	PSO4
CO1	3	-	3	3	3	-	-	3
CO2	3	-	3	3	3	-	-	3
CO3	3	-	-	3	3	-	-	3
CO4	-	3	-	-	3	-	-	3
CO5	-	3	-	-	3	-	-	3
CO6	3	-	-	-	3	-	-	3

Learning Outcomes and Assessment

Each course's learning outcomes will be assessed based on one or many methods, including the internal written tests, quizzes, presentations, seminars, assignments in the form of lab exercises, and group projects. The above assessment methods will be attentively created to support the intended learning outcomes that have been set out for a particular course. The program outcome attainment is measured using the CO/PO mappings.