AMRITA VISHWA VIDYAPEETHAM | School of Engineering

M.TECH COMPUTER SCIENCE AND ENGINEERING

(Artificial Intelligence and Machine Learning)

CURRICULUM & SYLLABUS 2021

M.TECH - CSE (Artificial Intelligence and Machine Learning)
Department of Computer Science and Engineering

M.Tech in Computer Science and Engineering (CSE) with a specialization in Artificial Intelligence and Machine Learning (AI & ML) program has been designed for students with sufficient background in computer science and engineering to develop into adept professionals. This new program in CSE with a specialization in AI&ML is a graduate degree that builds skill and knowledge in advanced and current topics of computer science with equal importance to core computer science courses and on AI topics. The degree is suitable for students with a bachelor's degree in a computing related field as well as students who want to demonstrate computer science expertise in addition to a degree in another field.

This program differentiates itself from M. Tech in AI program by providing strong foundation in core Computer Science subjects such as Algorithms, Networking, Computer Architecture, Graph Theory, Databases and Web technologies. Along with providing stronger foundations in Core areas of Computer science, this program also allow students to gain knowledge and expertise in important fields of Artificial Intelligence and Machine Learning. This will help them secure an Industry job or carry on future research in specific areas.

This M.Tech programme gives a specialized focus on areas of technology, aiming to develop skills and career prospects. The master's degree program offers an integrated course of study covering the theory, implementation and design of information, computing, communication and embedded systems. This programme has specialized courses in the streams of AI and Machine Learning with significant focus on research. As a part of the programme during the period of study, students have the opportunity to do internship at leading companies and R&D labs for a period of 6 months to one year. There are opportunities for the students to take up a semester or one year study at International Universities like Virje University, Netherlands, UC Davis, UNM for an exchange programme or to pursue a dual degree programme.

Graduates of this programme are well represented in Oracle, IBM, HP, Cerner, Nokia, Dell, GE and other major MNCs as well as in research in premier academic institutions in India and abroad. The graduates are competent to take up R&D positions in Industry, academia and research laboratories.

Program Educational Objectives (PEOs)

1. Build strong foundations in core areas of Computer science with a focus on Artificial Intelligence and Machine Learning so that they can contribute significantly in the area of research and innovation.
2. Develop professionals with high competency in recent tools and techniques related to ArtificialIntelligence and Machine learning.
3. Bring out professionals and entrepreneurs to design and develop solutions for real world interdisciplinary problems having positive societal impacts.

Graduate Attributes prescribed by NBA for M-Tech Program

GA1: Scholarship of knowledge
GA2: Critical thinking
GA3: Problem solving
GA4: Research skill
GA5: Usage of modern tools
GA6: Collaborative and multidisciplinary work
GA7: Project management and finance
GA8: Communication
GA9: Lifelong learning
GA10: Ethical practices and social responsibility GA11:
Independent and reflective learning.

Program Outcomes (POs)

1. Develop strong theoretical foundations and practical skillsets in core areas of Computer Scienceand Artificial Intelligence to facilitate research, innovation, and product development (GA1, GA2, GA3, GA6).
2. Provide opportunities to independently carry out research and development work to solvepractical problems. (GA4, G10)
3. Develop professionals and entrepreneurs with skillsets that enable them to quickly adapt tolatest tools and technologies (GA5).
4. Enable the students to explore and execute projects having a strong impact in an economic,social, and environmental context (GA7).
5. Build the ability to learn independently and engage in life-long learning. (GA9).
6. Facilitate students to acquire good communication skills in terms of technical reports, researchpublications, group activities, seminars, and presentations. (GA4, GA8, GA11).

CURRICULUM M.Tech CSE (AI & ML) 2021

Semester I				
Course Code	**Type**	**Courses**	**L T P**	**Cr**
21AM601	FC	Advanced Data Structures and Algorithms	3-0-2	4
21MA607	FC	Linear Algebra and Probability	2-1-0	3
21AM602	FC	Computational Methods for Optimisation	2-1-0	3
	SC	Soft Core - I (CS)	3-0-2	4
	SC	Soft Core - II (AI & ML)	3-0-2	4
21HU601	HU	Amrita Values Program*		P/F
21HU602	HU	Career Competency I		P/F
		Total Credits		**18**

*Non-credit course

Semester II				
Course Code	**Type**	**Courses**	**L T P**	**Cr**
	SC	Soft Core - III (CS)	3-0-2	4
	SC	Soft Core - IV (AI & ML)	3-0-2	4
	E	Elective - I (CS)	2-0-2	3
	E	Elective - II (AI & ML)	2-0-2	3
21AM698		Project with Seminar	2-0-0	2
21RM600	SC	Research Methodology	2-0-0	2
21HU603	HU	Career Competency II	0-0-2	1
		Total Credits		**19**
Semester III				
Course Code	**Type**	**Courses**	**L T P**	**Cr**
	E	Elective - III (CS)	2-0-2	3
	E	Elective - IV (AI & ML)	2-0-2	3
21AM798		Dissertation Phase I		10
		Total Credits		**16**
Semester IV				
Course Code	**Type**	**Courses**	**L T P**	**Cr**
21AM799		Dissertation Phase II		16
		Total Credits		**16**

S.No	Type	Course Category	Credits
1.	FC	Foundation Core	10
2.	SC	Soft Core	18
3.	E	Electives	12
4.	HU	Amrita Values Program/Career Competency	1
5.		Dissertation	29
		Total Credits	**69**

Foundation Core			
Course Code	**Courses**	**L T P**	**Cr**
21AM601	Advanced Data Structures and Algorithms	3-0-2	4
21MA607	Linear Algebra and Probability	2-1-0	3
21AM602	Computational Methods for Optimization	2-1-0	3
Soft Core			
Course Code	**Computer Science Courses**	**L T P**	**Cr**
21AM631	Advanced Computer Architecture	3-0-2	4
21AM632	Advanced Computer Network	3-0-2	4
21AM633	Algorithmic Graph Theory	3-0-2	4
21AM634	IoT and Edge Computing	3-0-2	4
21AM635	Embedded Programming	3-0-2	4
21AM636	Parallel and Distributed Data Management	3-0-2	4
21AM637	Cryptography and Network Security	3-0-2	4
21AM638	Wireless and Mobile Networks	3-0-2	4
21AM639	Image and Video Processing	3-0-2	4
21AM640	Modern Database Management Systems	3-0-2	4
	AI & ML Courses		
21AM641	Foundations of Data Science	3-0-2	4
21AM642	Machine Learning	3-0-2	4
21AM643	Computational Intelligence	3-0-2	4
21AM644	Foundations of Artificial Intelligence	3-0-2	4
21AM645	Deep Learning	3-0-2	4
21AM646	Probabilistic Graphical Models	3-0-2	4

Electives List			
Course Code	**Computer Science Courses**	**L T P**	**Cr**
21AM701	Modern Compiler Design	2-0-2	3
21AM702	Distributed Computing	2-0-2	3
21AM703	Computational Geometry	2-0-2	3
21AM704	Full Stack Development with Python	2-0-2	3
21AM705	Wireless Communication Technologies	2-0-2	3
21AM706	Modeling and Simulation	2-0-2	3
21AM707	Blockchain Technology	2-0-2	3
21AM708	GPU Architecture and Programming	2-0-2	3
21AM709	Information Retrieval	2-0-2	3
21AM710	Networks and Spectral Graph Theory	2-0-2	3
	AI & ML Courses		
21AM711	Reinforcement Learning	2-0-2	3
21AM712	Machine Learning for Big Data	2-0-2	3
21AM713	Applications of Machine Learning	2-0-2	3
21AM714	Multi Agent Systems	2-0-2	3
21AM715	Deep Learning for Biomedical Data Analysis	2-0-2	3
21AM716	Artificial Intelligence for Robotics	2-0-2	3
21AM717	Cloud and Big Data Analytics	2-0-2	3
21AM718	Quantum Artificial Intelligence	2-0-2	3
21AM719	Natural Language Processing	2-0-2	3
21AM720	Computer Vision	2-0-2	3

*Students can take Electives from other M.Tech branches in place of any one elective. Students can select an online course in place of Elective III or IV as per the university norms with the consent and approval from the department.

SYLLABUS

21AM601 Advanced Data Structures and Algorithms 3-0-2-4

Preamble

This course builds upon the basic data structures and algorithms. It aims to enable students to design data structures and algorithms to solve complex problems specially relevant for AI domain. The main focus will be on concrete implementations of various data structures and their use in non-trivial algorithms with proper analysis.

Course Objectives

- To provide an understanding of Data structures and algorithms used in real life domain
- To solve complex problems by applying appropriate Data structures and algorithms
- To critically analyze the complexity of various algorithms
- To select appropriate design strategy to solve real world problems in AI domain

Course Outcomes

COs	Description
CO1	Understand advanced data structures and their advanced applications
CO2	Understand wide range of algorithmic design techniques, their relations and variants, and application to real-world problems.
CO3	Understand Amortized analysis associated with a data structure and various ways of analyzing a given algorithm
CO4	Solve real world problems, especially in ML & AI domain, by identifying and applying appropriate design techniques
CO5	Concrete implementations of data structures and algorithms using Programming Language such as Java or Python

Prerequisites

- Basic Data Structures
- Basic Mathematics
- Basic Programming Language

Syllabus

Algorithm Analysis - Methodologies for Analyzing Algorithms, Asymptotic growth rates, Amortized Analysis. Number Theory: Preliminaries, FLT, Euclid's algorithm (extended), Totient function, Sieve for primes, Modular exponentiation, Applications of graph algorithms: Topological sort, Strongly connected Components, Bi-connected Components, Bridges, Articulation points, All Pairs Shortest Paths, Single Source Shortest Paths. Computational Geometry: Convex Hull, Closest pair of points.

Applications of Divide-and-Conquer, Greedy and Dynamic programming techniques - Knapsack, Median finding, Scheduling algorithms, Party planning, bitonic TSP. String matching algorithms: Z Algorithm, KMP algorithm, Rabin-Karp, Universal hashing, consistent hashing, load balancing, power of two choices.

B-trees, Suffix trees, Segment trees, Flow Networks: Ford-Fulkerson algorithm, Edmonds Karp algorithm, Applications of maximum flows - Maximum bipartite matching, minimum cost matching,.

NP-Completeness: Important NP-Complete Problems, Polynomial time reductions, Approximation algorithms.

Text Book / References

1. Cormen T H, Leiserson CE, Rivest R L and Stein C, "Introduction to Algorithms", PrenticeHall of India Private Limited. Third Edition 2009.
2. Michael T Goodrich and Roberto Tamassia, "Algorithm Design and Applications", Wiley,2014.
3. Rajeev Motwani and PrabhakarRaghavan, "Randomized Algorithms", Cambridge UniversityPress, 1995.
4. Vijay V. Vazirani, "Approximation Algorithm", Springer Science and Business Media, 2003.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Understand advanced data structures and their advanced applications	3	3	1	–	1	
CO2	Understand wide range of algorithmic design techniques, their relations and variants, and application to real-world problems.	3	3	1	–	1	
CO3	Understand Amortized analysis associated with a data structure and various ways of analyzing a given algorithm	3	3	1	-	1	
CO4	Solve real world problems, especially in ML & AI domain, by identifying and applying appropriate design techniques	3	3	3	-	1	
CO5	Concrete implementations of data structures and algorithms using Programming Language such as Java or Python	3	3	3	1	1	

Evaluation Pattern - 70:30

Periodical 1 - 15%
Periodical 2 - 15%
Continuous Evaluation - 40%
End Semester Exam- 30%

21MA607 Linear Algebra and Probability 2-1-0-3

Preamble

Linear algebra is central to almost all fields of engineering fields particularly in AI & ML as it allows efficient computational models when dealing with high dimensional data. The probability & statistics provide the formal basis for analyzing uncertainties and risks that are important when designing solutions based on data. This course is designed to impart core concepts and techniques on Linear Algebra and Probability with special focus on computational experiments.

Course Objectives

- Provide strong theoretical foundations on linear algebra and probability theory.

- Through properly designed computational experiments, make students confident enough to apply tools and techniques on linear algebra, probability and statistical testing in research and development.

Course Outcomes

After completing this course, the students will be able to

COs	Description
CO1	Use computational techniques and algebraic skills essential for the study of systems of linear equations, matrix algebra, vector spaces, eigenvalues and Eigenvectors, orthogonality and diagonalization.
CO2	Understand the importance of probability distribution and statistical testing in data modelling and data analytics
CO3	Should be able to wisely choose tools and techniques in Linear algebra and Statistical testing for building and evaluating solutions to different Engineering problems.

Syllabus

Geometry of linear equations, Gaussian elimination, vector spaces and subspaces, Fundamental vector spaces associated with a matrix, Bases and dimension. Projections and ordinary least squares, Eigenvalues and Eigenvectors, Matrix decompositions and applications: LU, QR, QΛQ', SVD, Positive definite matrices - minima, maxima and saddle points, Introduction to special matrices and applications: Fourier transform, Sine and Cosine transforms.

Review of probability theory, Conditional probability, Baye's rule, Random variables and probability Distributions - Moments and moment-generating function, Joint probability distributions, Sampling from probability distributions, Covariance and Correlation, Central limit theorem, Parameter estimations: Maximum likelihood estimation, Expectation-Maximisation, Introduction to statistical testing - Z Test and Test on Proportions

Text Book / References

1. Gene H. Golub and V. Van Loan, Matrix Computations, Third Edition, John Hopkins University Press, Baltimore, 1996.
2. Strang, Gilbert. Linear algebra and learning from data. Cambridge: Wellesley-CambridgePress, 2019.
3. Douglas C. Montgomery and George C. Runger, "Applied Statistics and Probability for Engineers", Third Edition, John Wiley & Sons Inc., 2003.
4. David Forsyth, "Probability and Statistics for Computer Science", Springer international publishing, 2018

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Use computational techniques and algebraic skills essential for the study of systems of linear equations, matrix algebra, vector spaces, eigenvalues and Eigenvectors, orthogonality and diagonalization.	3	3	3	2	3	1
CO2	Understand the importance of probability distribution and statistical testing in data modelling and data analytics.	3	3	3	3	3	2

CO3	Should be able to wisely choose tools and techniques in Linear algebra and Statistical testing for building and evaluating solutions to different Engineering problems.	3	3	3	3	3	2

Evaluation Pattern - 70:30

Midterm Exam - 20%
Continuous Evaluation (Computational) – 50%
End Semester Exam - 30%

21AM602 Computational Methods for Optimisation 2-1-0-3

Preamble

The course Computational Methods for Optimisation is designed to teach core concepts and techniques for Mathematical optimisation with special focus to convex optimisation. Each technique should be taught by designing computational experiments by selecting relevant Engineering AI& ML Application.

Course Objectives

- The course will lay down the basic concepts and techniques of optimization theory needed for subsequent study.
- The course provides a thorough understanding of how optimization problems are solved, and some experience in solving them.
- The course will provide the background required to use the methods in research work and/or applications.

Course Outcomes

After completing this course, the students will be able to

COs	Description
CO1	Should be able to understand and appreciate the necessary theory behind solutions to optimisation problems and develop skills to understand research papers in the domain
CO2	Should be able to use different tools and techniques to find solutions to Optimisation problems
CO3	Should gain necessary skill to pose engineering problem as an optimisation problem wherever possible.

Syllabus

Theory of Optimization – Regression and Least Square – Batch gradient descent – Conjugate Gradient method – Linear programming and Simplex method and Applications to Engineering Problems – Direct methods for convex functions, Newton methods for non-convex functions. Constrained Convex Optimization problems, Formulating problems as LP and QP, solving by packages (CVXOPT) – Need for Unconstrained optimization methods – Lagrangian multiplier method, KKT conditions – Alternating direction method of multipliers (ADMM) and Applications– Optimization methods for sparsity – Introduction to Variational Methods in Optimisation: Calculus of variations and Applications – Optimization methods for Neural Networks: Stochastic gradient descent, ADAM (Adaptive Moment), Contrastive Divergence

Text Book / References

1. Stephen Boyd and Lieven Vandenberghe, 'Convex Optimization', Cambridge University Press,2018

2. Fletcher R., 'Practical Methods of Optimization', John Wiley, 2000.

3. Kalyanmoy, Deb. Optimization for engineering design: Algorithms and examples. PrenticeHall of India Pvt. Limited, 2012.

4. Chong, Edwin KP, and Stanislaw H. Zak. An introduction to optimization. John Wiley &Sons, 2004.

5. Bhatti, M. Asghar. Practical Optimization Methods: With Mathematical Applications. SpringerScience & Business Media, 2012.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Should be able to understand and appreciate the necessary theory behind solutions to optimisation problems and develop skills to understand research papers in the domain.	3	3	2	2	2	3
CO2	Should be able to use different tools and techniques to find solutions to Optimisation problems.	3	3	3	2	2	2
CO3	Should gain necessary skill to pose engineering problem as an optimisation problem wherever possible.	3	3	3	2	3	2

Evaluation Pattern - 70:30

Midterm Exam - 20%
Continuous Evaluation (Computational) – 50%
End Semester Exam - 30%

21AM631 Advanced Computer Architecture 3-0-2-4

Preamble

- This course attempts to provide an understanding on design issues of computer systems, through quantitative approach, where performance is a main issue.

Course Objectives

- To develop an understanding on the performance aspects of computer systems

- A deeper understanding of multi core, multiprocessor, and GPGPU architectures.

Course Outcomes

COs	Description
CO1	Understanding quantitative analysis on the performance of computer systems
CO2	Develop design skills of pipelined instruction set architecture
CO3	Understanding data and control path of advanced processors

CO4	Analyzing various aspects for parallelism
CO5	Demonstrating the knowledge on recent research in processor architecture through technical writing and presentations.

Prerequisites

- Assembly language programming and basics of computer organization.
- A working knowledge in programming languages such as C, C^{++}/Java is desirable.

Syllabus

Introduction-Fundamentals of computer design, evaluating performance -Pipelining-Instruction set design principles. Caches and memory hierarchy.

Design-Review of memory Hierarchy-Advanced memory hierarchy design concepts. Instruction level parallelism and its Exploitation-Limits on instruction level parallelism. Multiprocessors and Threadlevel Parallelism-Models of parallel computation, network topologies, consistency models.

Simultaneous Multi-Threading (SMT), Chip Multi-Processors (CMP), General Purpose Graphics Processing Units (GPGPU). VLSI Scaling issues, data speculation, dynamic compilation, communication architectures, near data processing, and other advanced topics.

Text Book / References

1. L. Hennessy & David A. Patterson, Morgan Kaufmann," Computer Architecture: A Quantitative Approach", 5th Edition, 2011, , ISBN: 978-0-12-383872-8
2. David A Patterson & John L. Hennessy, Morgan Kaufmann, "Computer Organization andDesign", the Hardware/Software Interface, 5th Edition.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Understanding quantitative analysis on the performance of computer systems	2	2	2	3	2	2
CO2	Develop design skills of pipelined instruction set	3	3	2	2	3	
CO3	Understanding data and control path of advanced processors	3	3	3	3	2	
CO4	Analyzing various aspects of parallelism in computing	3	3	3	3	2	1
CO5	Demonstrating the knowledge on recent research in processor architecture through technical writing and presentations	3	3	3	2	2	3

Evaluation Pattern - 70:30

Periodical 1 - 15%
Periodical 2 - 15%
Continuous Evaluation - 40%
End Semester Exam - 30%

21AM632 Advanced Computer Networks 3-0-2-4

Preamble

Advanced Computer Networks is a fundamental core course offered to CSE M.Tech Students. This course provides insights into computer networks, Communication model, Network protocols and standards. Moreover, the course covers various Network layers and stacks, Routing algorithms, Application layer protocols. Then, the course introduces the students to Software Defined Network (SDN), and Network Function Virtualization – NFV – two important concepts in the current computer networks. The lab session helps the students to learn and understand different network analyser tools such as wireshark, packet tracer and so on. The students will gain hands on experience in working with various network communication and management services in software defined networks using Mininet and open vSwitch.

Course Outcomes

COs	Description
CO1	Understanding of the fundamentals of conventional network layers and protocols
CO2	Introducing the fundamentals of software defined networking
CO3	Familiarity with configuring and programming a sensor board
CO4	Gain practical knowledge by setting up networks using network simulator
CO5	Understanding of Network Function Virtualization related usecases and applications

Prerequisites

- Basic Knowledge of Linux Operating System

Syllabus

The Internet- The network edge, the network core, Delay, Loss and Throughput in packet switched networks, Protocol layers and their service models. Principles of Network applications: The Web and HTTP, File transfer: FTP, Email in the internet. DNS- The internet's directory service, Peer to Peer applications. Transport layer services: Multiplexing and Demultiplexing, Connectionless transportUDP, Principles of Reliable Data Transfer. Transport layer- Connection Oriented Transport-TCPThe Internet Protocol(IP)- Forwarding and Addressing in the internet, Routing Algorithms, Routing in the Internet, Broadcast and Multicast Routing.

SDN – Software defined networking- Background and Motivation, Key Terms, SDN and NFV related standards, SDN architecture- SDN Dataplane, OpenFlow Protocol, Control plane Control Overhead & Handoff algorithms. Network Function Virtualization - NFV requirements and concepts, Architecture, Use cases, NFV Management and Orchestration, Resource Management, Analytics, Service Chaining, Distributed NFV, Network security in software defined networks, implementation of IOT networks using SDN.

Text Book / References

1. James F. Kurose and Keith W. Ross, "Computer Networking: A Top-Down Approach", 8thEdition, Addison-Wesley, 2020

2. Tanenbaum A S, "Computer Networks", fifth Edition Edition, PHI, 2016.

3. Stalling, s, W, "Foundations of modern networking: SDN, NFV, QoE, IoT, and Cloud', ‚Addison-Wesley Professional, 2016 'Software defined Networking with OpenFlow', SiamakAzodolmolky, PACKT publishers, 2017

4. NFV architecture document from ETSI NFV.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Understanding of the fundamentals of conventional network layers and protocols	3	3	1	1	2	
CO2	Introducing the fundamentals of software defined networking	1	3	2	3	2	
CO3	Familiarity with configuring and programming a sensor board		1	3	2	2	
CO4	Gain practical knowledge by setting up networks using network simulator	2	3	2	2	2	2
CO5	Understanding of Network Function Virtualization related usecases and applications	1	2	3	2	3	3

Evaluation Pattern - 70:30

Course Type - Lab based
Midterm - 20%
Lab Experiments - 25%
Minor Project and Viva - 25% End
Semester Exam - 30%

21AM633 Algorithmic Graph Theory 3-0-2-4

Preamble

This course focuses on an introduction to graph theory and properties of graph and then move towards learning advanced graph models and its applications to AI domain. The course survey algorithmic models for geometric problems in graph. Course also focus on ways to represent large graphs using graph compression and explore the AI problems requiring graph based solution

Course Objectives

- To give the students fundamentals of graph theory
- Expose students to the concepts in advanced graph models
- Learn various applications of Graph theory
- Enable students to apply graph based solution approaches to AI problems

Course Outcomes

COs	Description
CO1	Learn the fundamentals and mathematical definitions of objects in graph theory.
CO2	To be familiar with algorithmic models for geometric problems in graphs.
CO3	Learn how to represent large graphs using graph compression.
CO4	To apply the graph modelling approach to AI problems

Prerequisites

- Basic course in Data Structures and Algorithms.

Syllabus

Revisiting network graphs fundamentals, measures and metrics Graph clustering: Algorithms for partitioning a graph into well-connected pieces (e.g. spectral partitioning, sparsest-cuts, multiway cuts). Distances in graphs: Algorithmic methods for geometric problems in graphs, such as the Traveling Salesperson Problem, Minimum Spanning Trees, shortest paths, Flows in graphs: Min-cut/max-flow duality, and its extensions to multi-commodity flows. Applications to divide & conquer. Graph compression: Methods for representing succinctly large graphs (e.g. spectral sparsifiers, vertex sparsifiers, graph spanners). Algorithmic graph-minor theory: Dynamic programming on graphs via tree decompositions. Algorithms for graphs on surfaces

Text Book / References

1. Dougles West, “Introduction to Graph Theory”, Second Edition, PHI Learning Private Limited,2011.
2. Mark Newman ”Networks: An Introduction” Oxford Oxford, 2010
3. R. K. Ahuja, T. L. Magnanti and J. B. Orlin, “Network Flows: Theory, Algorithms, andApplications,” PrenticeHall, Englewood Cliffs, 1993.
4. Mohar, Bojan, and Carsten Thomassen. Graphs on surfaces. Vol. 2. Baltimore: Johns Hopkins University Press, 2001.
5. T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, Introduction to Algorithms.ThirdEdition

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Learn the fundamentals and mathematical definitions of objects in graph theory	3	2	1	1		
CO2	To be familiar with algorithmic models for geometric problems in graphs	3	2	2	2		
CO3	Learn how to represent large graphs using graph compression.	1	1	3	3		
CO4	To apply the graph modeling approach to AI problems.	1	2	2	3	2	3

Evaluation Pattern - 70:30

Midterm Exam - 20%
Continuous Evaluation (Computational) – 50%
End Semester Exam - 30%

21AM634 IoT and Edge Computing 3-0-2-4

Preamble

IoT, Cloud and Edge Computing is a course offered as a softcore to M. Tech CSE students in Computer Science & Engineering. The course first familiarizes the students with basic Computer Network layers and common protocols. Then, it introduces the students to the concepts related to IoT and covers IoT architecture design, different communication protocols existing today and those coming up in near future. Then, the students get exposure to cloud environment and they would be able to upload data from IoT devices to the cloud and visualize it. Finally, various applications at the Edge layer will be discussed.

Course Objectives

- Understand the general concepts in IoT and get familiar with the various hardware and software components of IoT.
- Understand how to communicate using different protocols
- Learn how to connect to cloud and get exposed to Edge computing

Course Outcomes

COs	Description
CO1	Understand thoroughly basic Internet communication layers and their services.
CO2	Understand various concepts related to IoT and sensors.
CO3	Become familiar with various IoT Protocols.
CO4	Building IoT applications using Raspberry Pi or NodeMCU board.
CO5	Data processing and Visualization of IoT data using Open IoT cloud platform.
CO5	Learn about important real-life IoT usecases and Scenarios for Edge Computing.

Prerequisites

- Computer Networks

Syllabus

The Internet- The network edge, the network core, recap on Protocol layers and their service models. Principles of Network applications: The Web and HTTP, Peer to Peer applications. Transport layer services: Connectionless transport-UDP, Connection Oriented Transport-TCP, Principles of Reliable Data Transfer, Ethernet and LAN protocols.

Application Protocols for IoT: , CoAP, MQTT, Modbus. SCADA, WebSocket; IP-based protocols: 6LoWPAN, RPL; Authentication Protocols; IEEE 802.15.4, Wi-Fi, Li-Fi, BLE.

IoT Application Development, Introduction to Raspberry Pi/Any Sensor board, Integrating Sensors and Actuators with Raspberry Pi, Pushing and Managing Data in IoT Clouds, Programming APIs (Python/Node.js/Arduino), A number of IoT Case study: Smart-Facilities Management, Healthcare, Environment Monitoring System.

Text Book / References

1. David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Robert Barton, Jerome Henry, “IoTFundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things”, CISCO Press, 2017
2. Arshdeep Bahga and Vijay Madisetti, ”Internet of Things, A Hands on Approach”, UniversityPress, 2015
3. Hersent, Olivier, David Boswarthick, and Omar Elloumi. ”The internet of things: Key applications and protocols”. John Wiley & Sons, 2012.
4. Adrian McEwen, ”Designing the Internet of Things”, Wiley, 2013

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Understand thoroughly basic Internet communication layers and their services.	3	2	2	2		
CO2	Understand various concepts related to IoT and sensors.	3		2	3	3	

CO3	Become familiar with various IoT Protocols.	2	3	3	2	3	
CO4	Building IoT applications using Raspberry Pi or NodeMCU board.	2	2	2	3	2	2
CO5	Learn about important real-life IoT usecases and Scenarios for Edge Computing.	2	2	2	3	2	3

Evaluation Pattern - 70:30

Midterm Exam - 20%
Continuous Evaluation – 50% End
Semester Exam - 30%

21AM635 Embedded Programming 3-0-2-4

Preamble

Course Objectives

- The course will stress upon the importance of embedded systems in modern applications
- The course will pave the way for understanding the various sources of data stream from an embedded system point of view and recording this data streams for processing
- This will enable the students to develop various system and analyze the data from the system for decision making. • The course will enable the student with basic real-time operating system concepts for application development.

Course Outcomes

COs	Description
CO1	Enabling the student with fundamentals of micro-controller architecture, building components and embedded data streaming devices
CO2	To program various micro-controllers, application development and data streaming using various sensors
CO3	Working with FreeRTOS for developing real-time data streaming and intelligent applications
CO4	To develop real-time embedded system application in healthcare, agriculture, autonomous car and other data streaming systems

Prerequisites

- Basic Computer Architecture

Syllabus

Hardware and Software architecture of Embedded Systems. Review of general C programming and data types, arrays, functions, pointers, structure, enum, files. Introduction to Embedded C, Interfacing C with Assembly. Embedded programming issues - Reentrancy, Portability, Optimizing and testing embedded C programs. Embedded Applications using Data structures, Linear data structures– Stacks and Queues, Linked List. Embedded C++ and Scripting Languages for Embedded Systems. Software to hardware mapping for specific architecture.

Introduction to real-time systems, RTOS basic architecture, RTOS Kernel, Kernel services: Task Management -tasks, process and threads, task attributes and types - task states and transition, task control block, Introduction to real-time task scheduling. RTOS for multi-core processors. OS for end and edge devices in cyber physical systems. Development and debugging and version control tools for Embedded systems.

Case study: Embedded Linux / VxWorks / Free RTOS / RTLinux / uCOS in critical real-time embedded systems.

Text Book / References

1. Michael Barr, Programming Embedded Systems in C and C++ Oreilly Publications, SecondEdition, 2012.
2. Wang, K. C. "Embedded real-time operating systems." Embedded and Real-Time OperatingSystems. Springer, Cham, 2017. 401-475
3. Brian Amos,"Hands-On RTOS with Microcontrollers: Building real-time embedded systemsusing FreeRTOS, STM32 MCUs, and SEGGER debug tools", PACKT publishing, 15 May, 2020
4. David E Simon, "An Embedded Software Primer", Pearson Education Asia, 2005.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Enabling the student with fundamentals of micro-controller architecture, building components and embedded data streaming devices	-	3	2	-	2	
CO2	To program various micro-controllers, application development and data streaming using various sensors	-	2	3	-	3	
CO3	Working with FreeRTOS for developing realtime data streaming and intelligent applications	-	-	2	-	2	3
CO4	To develop real-time embedded system application in healthcare, agriculture, autonomous car and other data streaming systems	2	3	2	3	3	3

Evaluation Pattern - 70:30

Periodical 1 - 15%
Periodical 2 - 15%
Continuous Evaluation – 40%
End Semester Exam - 30%

21AM636 Parallel and Distributed Data Management 3-0-2-4

Preamble

The improvements in Database Management System (DBMS) technology has resulted in significant developments in distributed computing and parallel processing technologies. This has led to the development of distributed database management systems and parallel database management systems that are now the dominant data management tools for highly data-intensive applications. In addition to an introduction to parallel and distributed database architectures and their implementation features, this course covers advanced query processing and optimization approaches for parallel and distributed systems. The students also gain knowledge in setting up a distributed database application using the latest technologies.

Course Objectives

- To provide an understanding of the distributed and parallel database architectures so as to make a choice while implementing a distributed application;
- To learn how a distributed database can be implemented for an application;
- To get trained in distributed query processing and optimization for various distributed or parallel database applications

Course Outcomes

COs	Description
CO1	Understand the need for different distributed and parallel database architectures and study its characteristics.
CO2	Design algorithms for distributed and parallel data processing.
CO3	Understand the concepts of fragmentation and allocation algorithms.
CO4	Implement optimized parallel and distributed queries for such a system.
CO5	Design and build an application using one of the latest distributed or parallel database technology.

Prerequisites

- DBMS
- Algorithms and Data Structures
- Advanced Java, Apache Spark

Syllabus

Introduction: Parallel and Distributed architectures, models, complexity measures, Communication aspects, A Taxonomy of Distributed Systems - Models of computation: shared memory and message passing systems, synchronous and asynchronous systems, Global state and snapshot algorithms.

Distributed and Parallel databases: Centralized versus Distributed Systems, Parallel versus Distributed Systems, Distributed Database Architectures-Shared disk, shared nothing, Distributed Database Design – Fragmentation and Allocation, Optimization.

Query Processing and Optimization – Parallel/ Distributed Sorting, Parallel/Distributed Join, Parallel/Distributed Aggregates, Network Partitions, Replication, Publish/Subscribe Systems-Case study on Apache Kafka Distributed Publish/Subscribe messaging Hadoop and Map Reduce – Data storage and analysis, Design and concepts of HDFS, YARN, Map Reduce workflows and Features, Setting up a Hadoop cluster.

Text Book / References

1. M. Tamer Ozsu and Patrick Valduriez, "Principles of Distributed Database Systems", 3rd ed.2011 Edition, Springer
2. Dimitri P. Bertsekas and John N. Tsitsiklis, ”Parallel and distributed computation : Numericalmethods”,
3. Andrew S. Tannenbaum and Maarten van Steen ”Distributed Systems: Principles and Paradigms”,Second Edition, Prentice Hall, October 2006.
4. Ajay D. Kshemkalyani and Mukesh Singhal, ”Distributed Computing: Principles, Algorithms,and Systems”, Cambridge University Press, 2011.
5. Vijay K. Garg, ”Elements of Distributed Computing”, Wiley-IEEE Press, May 2002

6. David DeWitt and Jim Gray, "Parallel database systems: The future of high performancedatabase systems", CACM, 1992

7. Tom White, "Hadoop-The Definitive Guide", 4th ed., O'Reilly, 2015

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Understand the need for different distributed and parallel database architectures and study its characteristics	3	2	1	1	2	-
CO2	Design algorithms for distributed and parallel data processing	3	2	2	1	2	-
CO3	Understand the concepts of fragmentation and allocation algorithms	3	2	1	-	1	-
CO4	Implement optimized parallel and distributed queries for such a system	3	2	1	-	1	-
CO5	Design and build an application using one of the latest distributed or parallel database technology	3	3	3	3	3	2

Evaluation Pattern - 70:30

Periodical 1 - 15%
Periodical 2 - 15%
Continuous Evaluation – 40%
End Semester Exam - 30%

21AM637 Cryptography and Network Security 3-0-2-4

Preamble

Course Outcomes

COs	Description
CO1	To understand the mathematical base of cryptography.
CO2	To acquire knowledge on standard algorithms used to provide confidentiality, integrity and authenticity.
CO3	To understand how to deploy encryption techniques to secure data in transit across data networks.
CO4	To understand the security issues in various layers of network stack.
CO5	To get insights into the various protocols for network security to protect against network threats.

Prerequisites

- Nothing

Syllabus

Concepts of Number Theory: Number Theory, GCD, Euclidean algorithm, Extended Euclidean algorithm, prime numbers, congruence's, how to solve congruence equations, Chinese remainder theorem, residue classes and complete residue systems, Euler Fermat theorem, primitive roots. Symmetric Key Cryptographic Systems: Caesar and affine ciphers, mono-alphabetic substitutions,

transposition, homophonic, Vigenere and Beaufort ciphers, one-time pad, product/iterated/block ciphers, DES and AES.
Concepts of PKCS, Diffie Hellman key-exchange protocol, RSA, Rabin and EL Gamal cryptosystems. Digital Signatures- Rabin, Lamport, RSA,RSA and ElGamal signatures. Hash Functions and MACs-Message Authentication Codes, MD5, SHA1,SHA2 and construction of SHA3, security of Hash function,. Basic elliptic curve cryptography: definition, mathematical formulation of them, elliptic curve cryptography and pairings.

Overview of computer networks and network security, Application layer – TLS, PGP, S/MIME, Firewalls, Intrusion Detection Systems and Intrusion Prevention Systems. Transport layer - SSL architecture and protocols, UDP flooding, TCP spoofing, TCP connection hijacking, TCP SYN flood. Network layer- IPSec Introduction, Tunnel and Transfer Modes, IPSec Authentication Header, Encapsulating Security Header and Payload, IPSec Key Exchange and VPNs. Link layer -ARP, Attacks against and vulnerabilities in ARP.

Text Book / References

1. William Stallings "Cryptography and Network Security" Fifth Edition, Prentice Hall, 2011
2. Alfred J Menzes, Paul C Van Oorshot and Scott A. Vanstone "Handbook of Applied Cryptography", CBC Press, 1996
3. Neal Koblitz "A course in Number Theory and Cryptography" Springer-Verlag 1994
4. ChristofPaar, "Understanding Cryptography", Springer-Verlag-2010

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	To understand the mathematical base of cryptography.	3	2		2		
CO2	To acquire knowledge on standard algorithms used to provide confidentiality, integrity and authenticity.	2	3	1	3	3	
CO3	To understand how to deploy encryption techniques to secure data in transit across data networks.	2	3	2	2		1
CO4	To understand the security issues in various layers of network stack		2	2	3	2	2
CO5	To get insights into the various protocols for network security to protect against network threats.	2	2	2	3	3	2

Evaluation Pattern - 70:30

Periodical 1 - 15%
Periodical 2 - 15%
Continuous Evaluation – 40%
End Semester Exam - 30%

21AM638 Wireless and Mobile Networks 3-0-2-4

Preamble

This course teaches fundamentals characteristics of mobile and wireless networks. The main objective of this course is to introduce a wide range of current wireless networking technologies. During the course, students will learn the wireless network characteristics and the concept of network simulation.

Course Objectives

- To introduce current mobile and wireless networking technologies.
- To analyze the performance of different networks through simulation.

Course Outcomes

COs	Description
CO1	Understand fundamentals of wireless networks
CO2	Study the concepts of current wireless technologies and standards
CO3	Examine cellular network architecture
CO4	Analyse the impact of mobility in wireless networks
CO5	Compare and analyse different wireless networks through simulation

Prerequisites

- Computer Networks

Syllabus

Overview of wireless networks, Generations of wireless networks, Wireless and mobile networking: facts-statistics-trends, Characteristics of the wireless channel-radio propagation mechanisms, Medium access control -SDMA–FDMA-TDMA- CDMA -OFDMA.

IEEE 802.11 Standard- RTS/CTS- hidden and exposed terminal problem, IEEE 802.16 WiMAX, Ad Hoc Networks- Mobile Ad hoc Network(MANET) ,Wireless Sensor Networks(WSN), Routing in WSN and MANET, Bluetooth network, ZigBee network, Ultra wide band radio communication, optical wireless networks.

Cellular Concept- architecture- generations of cellular Systems-GSM, GPSR, CDMA2000, Mobility management, Location management, Wireless geolocation system architecture- technologies for wireless geolocation,Simulation of Wi-Fi, Mobile adhoc, zigbee, Bluetooth and cellular networks in ns3

Text Book / References

1. Murthy, C. Siva Ram. BS manoj, ”Ad Hoc Wireless networks architecture and protocols”Pearson education, (2005).
2. Kaveh Pahlavan, Prashant Krishnamurthy, “Principles of wireless Networks: A unified Approach”, Prentice Hall, 2001
3. William Stallings, “Wireless Communication and Networks”, 2nd edition, Prentice Hall, 2005.
4. Jochen Schiller,”Mobile Communications”, Second Edition, Pearson Education 2012

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Understand the fundamentals of wireless networks	3	2	3	1	-	
CO2	Study the concepts of current wireless technologies and standards	3	2	3	1	1	

CO3	Examine cellular network architecture	3	3	3	1	2	
CO4	Analyse the impact of mobility in wireless networks	3	3	3	3	3	2
CO5	Compare and analyse different wireless networks through simulation	3	3	3	3	3	2

Evaluation Pattern - 70:30

Periodical 1 - 15%
Periodical 2 - 15%
Continuous Evaluation – 40%
End Semester Exam - 30%

21AM639 Image and Video Processing 3-0-2-4

Preamble

Digital images and videos contribute a major share to the data generated/used by Engineering and Scientific field nowadays. Hence, the ability to process image and video signals is an important skill set for all computer science/AI students. In this course, the focus is to providing necessary theoretical/mathematical foundations and on the applications of the tools and techniques for building engineering solutions that use image/video as primary data source.

Course Objectives

- Build strong fundamentals on the techniques used for image and video processing
- Be able to wisely choose between techniques and if needed devise new techniques when building solutions to different computer vision problems

Course Outcomes

COs	Description
CO1	Be able to formulate the mathematical operations for image processing.
CO2	Understand and apply the different filtering operations in spatial domain.
CO3	Understand and apply the different filtering operations in frequency domain.
CO4	Understand the principles in video representation and motion analysis.
CO5	Understand and apply different block matching algorithms for segmentation.

Prerequisites

- Vector & Matrix Algebra: Vector spaces, Linear independence, Basics of Matrix Algebra, Eigenvalues & Eigenvectors

Syllabus

Introduction - Digital Image Fundamentals: Elements of Computer Vision-Light and the Electromagnetic Spectrum-Light and the Electromagnetic Spectrum-Image Sensing and Acquisition-Image Sampling and Quantization- Some Basic Relationships Between Pixels-Introduction to the Basic Mathematical Tools Used in Digital Image Processing- Intensity Transformations and Spatial Filtering: Basic Intensity Transformation Functions- Histogram Processing - Fundamentals of Spatial Filtering - Smoothing (Lowpass) Spatial Filters - Sharpening (High pass) Spatial Filters.

Filtering in the Frequency Domain: Sampling and the Fourier Transform of Sampled FunctionsDiscrete Fourier Transform of One Variable-Extensions to Functions of Two Variables-Properties of the 2-D DFT and IDFT-Filtering in the Frequency Domain-Image Smoothing Using Lowpass Frequency Domain Filters.

Digital Images and Video: Human Visual System and Color-Analog Video-Digital Video-3D VideoMotion Estimation: Image Formation-Motion Models-2D Apparent Motion Estimation-Differential Methods-Matching Methods-Image Segmentation.

Text Book / References

1. Rafael C. Gonzalez,Richard E. Woods,"Digital Image Processing" , Pearson Education, FourthEdition , 2018.
2. A. Murat Tekalp,"Digital Video Processing", O'Reilly, Second Edition, 2015.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Be able to formulate the mathematical operations for image processing.	3	3	2		2	2
CO2	Understand and apply the different filtering operations in spatial domain.	3	2	3		2	
CO3	Understand and apply the different filtering operations in frequency domain.	3	2	3		2	
CO4	Understand the principles in video representation and motion analysis.	3	2	2	2	2	
CO5	Understand and apply different block matching algorithms for segmentation.	3	2	3		2	2

Evaluation Pattern - 70:30

Periodical 1 - 15%
Periodical 2 - 15%
Continuous Evaluation – 40%
End Semester Exam - 30%

21AM640 Modern Database Management Systems 3-0-2-4

Preamble

Database Management Systems are one of the main assets in almost all organizations for data storage, management, and efficient retrieval. The integration of AI and DBMS technologies promises to play a significant role in shaping the future of computing. Hence, this course provides an in-depth view of latest Database trends in RDBMS, ORDBMS, and NoSQL Databases. The course introduces the concepts related to indexing structures, the power and usage of Embedded SQL, and Dynamic SQL in SQL UDFs(User Defined Functions) and UDAs(User Defined Aggregates). Further, in recent times, NoSQL databases have emerged to handle unstructured and semi-structured data at a large scale in the form of Big Data where Machine learning algorithms need to be leveraged for knowledge discovery. Hence, the course also offers exposure of the emerging technologies like MongoDB to handle document databases and develop full fledged web applications.

Course Objectives

- To understand the design and implementation strategies of different indexing techniques, query processing, and query optimization for effective database management and information retrieval in RDBMS.
- To understand the advanced concepts and terminology related to ORDBMS
- To understand the concepts and current trends in NoSQL databases

Course Outcomes

COs	Description
CO1	Will be able to comprehend and evaluate the role of ORDBMS and NoSQL database management systems in various applications within organizations.
CO2	Understand and apply different indexing techniques for database design, query processing and query optimization.
CO3	Will be able to comprehend how to develop and use SQL queries to define, retrieve, and manipulate information in Object Relational Database.
CO4	Will be able to design and implement unstructured or semi-structured databases under realistic constraints and conditions and how to develop queries for information retrieval from NoSQL databases.
CO5	Will be able to work in a group and develop applications with NoSQL database like MongoDB and Python APIs.

Prerequisites

- Basic RDBMS concepts, Basic SQL queries

Syllabus

Overview of RDBMS – Storage and File Structures, Indexing and Hashing - Indexing Structures – Single and Multi-level indexes. Algorithms for Query Processing and Optimization- Physical Database Design and Tuning. Intermediate and Advanced SQL - Embedded SQL, Dynamic SQL, Functions and Procedural Constructs, Recursive Queries, Advanced SQL Features. Transactions Processing and Concurrency Control - Transaction Concept, Transaction model, Storage Structure, Transaction Atomicity and Durability, Transaction Isolation, Serializability.

Object Relational Data Models – Complex Data Types, Inheritance, Nesting and Unnesting. NoSQL Databases – NoSQL Data Models, Comparisons of various NoSQL Databases. CAP Theorem, Storage Layout, Query models. Key-Value Stores. Document-databases – Apache CouchDB, MongoDB. Column Oriented Databases – Google's Big Table, Cassandra.

Advanced Application Development – Connecting to MongoDB with Python, MongoDB query Language, Updating/Deleting documents in collection, MongoDB query operators. MongoDB and Python patterns – Using Indexes with MongoDB, GeoSpatial Indexing, Upserts in MongoDB. Document database with Web frameworks – Django and MongoDB, Flask and MongoDB.

Text Book / References

1. Silberschatz A, Korth H F and Sudharshan S, “Database System Concepts”, Sixth Edition,Tata McGraw-Hill Publishing Company Limited, 2010.
2. Ramesh Elmasri and Shamkant B Navathe, “Fundamentals of Database Systems”, Fifth Edition, Pearson Educaton India, 2008.
3. Niall O'Higgins, “MongoDB and Python”, O'reilly, 2011.
4. Christof Strauch, “NoSQL Databases”

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Will be able to comprehend and evaluate the role of ORDBMS and NoSQL database management systems in various applications within organizations.	3	2	1	1	1	1
CO2	Understand and apply different indexing techniques for database design, query processing and query optimization.	3	2	1	1	1	1
CO3	Will be able to comprehend how to develop and use SQL queries to define, retrieve, and manipulate information in Object Relational Database.	3	2	3	1	1	1
CO4	Will be able to design and implement unstructured or semi-structured databases under realistic constraints and conditions and how to develop queries for information retrieval from NoSQL databases.	3	2	3	1	1	1
CO5	Will be able to work in a group and develop applications with NoSQL database like MongoDB and Python APIs.	1	3	3	2	3	3

Evaluation Pattern - 80:20

Course Type: Project based
Midterm Exam - 10%
Lab Assignments - 20%
Project – 50%
End Semester Exam - 20%

21AM641 Foundations of Data Science 3-0-2-4

Preamble

Data Science is about drawing useful conclusions from large and diverse data sets through exploration, prediction, and inference. In this course, the students will learn the foundations of data science that focuses on pre-processing, exploration and visualization of data alongside the statistical and mathematical aspects of select techniques for prediction and classification using popular machine learning algorithms. The course also focuses on the associated inferencing techniques based on statistical models and tests for quantifying the degree of certainty of predictions.

Course Objectives

- To understand the important steps in drawing useful conclusions from data;
- To ask appropriate questions about data after data exploration using visualization and descriptive statistics;
- To apply machine learning and optimization techniques to make predictions;
- To correctly interpret the answers generated by inferential and computational tools.

Course Outcomes

COs	Description
CO1	Understand the statistical foundations of data science
CO2	Learn techniques to pre-process raw data so as to enable further analysis
CO3	Conduct exploratory data analysis and create insightful visualizations to identify patterns
CO4	Apply machine learning algorithms for prediction/classification and to derive insights
CO5	Evaluate the degree of certainty of predictions using statistical test and models in Python

Prerequisites

- Basic Probability

Syllabus

Introduction to Data Science, Causality and Experiments, Data Pre-processing - Data cleaning Data reduction - Data transformation, Visualization and Graphing: Visualizing Categorical Distributions - Visualizing Numerical Distributions - Overlaid Graphs and plots - Summary statistics of exploratory data analysis, Randomness, Probability, Introduction to Statistics, Sampling, Sample Means and Sample Sizes.

Probability distributions and density functions (univariate and multivariate), Error Probabilities; Expectations and moments; Covariance and correlation; Sampling and Empirical distributions; Permutation Testing, Statistical Inference; Hypothesis testing of means, proportions, variances and correlations - Assessing Models - Decisions and Uncertainty, Comparing Samples - A/B Testing, P-Values, Causality.

Estimation - Resampling and Bootstrap - Confidence Intervals, Properties of Mean - Central Limit Theorem - Variability of mean -Choosing Sample Size, Prediction - Regression - Method of Least Squares - Visual and Numerical Diagnostics,- Inference for true slope - Prediction intervals, Classification - Nearest neighbors - accuracy of a classifier, Updating Predictions - Making Decisions Bayes Theorem, Graphical Models

Text Book / References

1. Ani Adhikari and John DeNero, ”Computational and Inferential Thinking: The Foundationsof Data Science”, e-book.
2. Joel Grus, ”Data Science from Scratch: First Principles with Python”, 2/e, O’Reilly Media,2019.
3. Peter Bruce, Andrew Bruce and Peter Gedeck, ”Practical Statistics for Data Scientists: 50+ Essential Concepts Using R and Python”, 2/e, O’Reilly Media, 2020.
4. Allen B. Downey, Think Stats: Probability and Statistics for Programmers”, 2/e, by O’ReillyMedia, 2014.
5. Cathy O’Neil and Rachel Schutt,”Doing Data Science”, O’Reilly Media, 2013.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Understand the statistical foundations of data science	3	2	2	1	3	1
CO2	Learn techniques to pre-process raw data so as to enable further analysis	1	2	2	2	1	1

CO3	Conduct exploratory data analysis and create insightful visualizations to identify patterns	2	1	2	1	1	3
CO4	Apply machine learning algorithms for prediction and classification to derive insights	3	3	2	2	2	2
CO5	Evaluate the degree of certainty of predictions using statistical test and models	2	2	2	2	2	1

Evaluation Pattern - 70:30

Midterm Exam - 20%
Continuous Evaluation – 50%
End Semester Exam - 30%

21AM642 Machine Learning 3-0-2-4

Preamble

This course deals with various algorithms to enable computers to learn data without being explicitly programmed. An insight into various types of machine learning algorithms, strategies for model generation and evaluation are given in this course. The fundamental machine learning algorithms required in industries are covered together with their concrete implementations.

Course Objectives

- To understand fundamental concepts of machine learning and its various algorithms
- To understand various strategies of generating models from data and evaluating them
- To apply ML algorithms on given data and interpret the results obtained
- To design appropriate ML solution to solve real world problems in AI domain

Course Outcomes

COs	Description
CO1	Develop a good understanding of fundamental principles of machine learning
CO2	Formulation of a Machine Learning problem
CO3	Develop a model using supervised/unsupervised machine learning algorithms for classification/prediction/clustering
CO4	Evaluate performance of various machine learning algorithms on various data sets of a domain.
CO5	Design and Concrete implementations of various machine learning algorithms to solve a given problem using languages such as Python

Prerequisites

- Basics of Algorithms
- Basics of Linear Algebra
- Basics of Python Programming Language

Syllabus

Introduction: Machine learning, Terminologies in machine learning, Types of machine learning: supervised, unsupervised, semi-supervised learning. Discriminative Models: Least Square Regression, Gradient Descent Algorithm, Univariate and Multivariate Linear Regression, Prediction Model, probabilistic interpretation, Regularization, Logistic regression, multi class classification, Support Vector Machines- Large margin classifiers, Nonlinear SVM, kernel functions, SMO algorithm.

Model evaluation and improvement, Regularization, Bias Variance, Hyper- parameter Tuning. Computational Learning theory- Sample complexity, exhausted version space, PAC Learning, agnostic learner, VC dimensions, Sample complexity - Mistake bounds. Gaussian models: Multivariate Gaussian distributions, Maximum Likelihood Estimate, Inferring parameters, Mixture models, EM algorithm for clustering and learning with latent variables.

Generative models : Linear Discriminative Analysis,Na¨ıve Bayes classifier, Decision trees, Ensemble models – Bagging and Boosting. Unsupervised Learning Algorithms: Dimensionality Reduction Principal Component Analysis (PCA), Singular Value Decomposition (SVD). Clustering – Hierarchical, Partitioned clustering : K-means, PAM, eXplainable AI (XAI), Approaching an ML problem

Text Book / References

1. Tom Mitchell, “Machine Learning”, McGraw Hill, 1997
2. E. Alpaydin, “Introduction to Machine Learning”, PHI, 2005.
3. Andrew Ng, Machine learning yearning, https://www.deeplearning.ai/machine-learning-yearning/
4. AurolienGeron, ”Hands-On Machine Learning with Scikit-Learn and TensorFlow, Shroff/O’Reilly”,2017
5. Andreas Muller and Sarah Guido, ”Introduction to Machine Learning with Python: A Guidefor Data Scientists”, Shroff/O’Reilly, 2016
6. Alejandro Barredo Arrieta, Natalia D´ıaz-Rodr´ıguez, Javier Del Ser, et.al., ”Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI, Information Fusion”,Volume 58,2020,Pages 82-115,ISSN 1566-2535, https://doi.org/10.1016/j.inffus.2019.12.012.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Develop a good understanding of fundamental principles of machine learning	3	3	1	1	-	
CO2	Formulation of a Machine Learning problem	3	3	2	1	1	
CO3	Develop a model using supervised/unsupervised machine learning algorithms for classification/prediction/clustering	3	3	3	2	2	
CO4	Evaluate performance of various machine learning algorithms on various data sets of a domain	3	3	3	3	2	
CO5	Design and Concrete implementations of various machine learning algorithms to solve a given problem using languages such as Python	3	3	3	3	1	

Evaluation Pattern - 70:30

Periodical 1 - 10%
Periodical 2 - 10%
Lab Assignments – 50%
End Semester Exam - 30%

21AM643 Computational Intelligence 3-0-2-4

Preamble

Computational Intelligence (CI) is a relatively new area which is becoming more and more important in society today and in the future, especially due to the growing possibilities of gathering data and the need for intelligent systems It is a set of nature-inspired computational methodologies and approaches to address complex real-world problems. This course is designed around five paradigms of the Computational Intelligence, namely, artificial immune systems (AIS), evolutionary computing, fuzzy systems, neural networks (NNs), and swarm intelligence. This course will also give an insight into a variety of applications in which various CI techniques will be applicable.

Course Objectives

- Become familiar with basic principles and mathematical frameworks for different computational intelligence techniques.
- Learn to wisely choose CI techniques for solving practical engineering problems
- Build strong theoretic and practical skills to make the students to quickly adapt to recent advancements and potentially contribute in the field.

Course Outcomes

After completing this course, the students will be able to

COs	Description
CO1	Understand the mathematical framework for the neuro, fuzzy and genetic algorithms and the way they differ from conventional optimisation techniques.
CO2	Decide and make use of appropriate neuro, fuzzy, genetic architectures for solving practical problems in Engineering domain.
CO3	Extend the skill acquired to understand the recent advancements in the field and for potential contributions.

Syllabus

Computational intelligence (CI): Linear Separable Problems and Perceptron, Multi-Layer neural networks – Back Propagation-radial basis function based multilayer perceptron - Kohonen's SelfOrganizing Networks - Hopfield Networks, ART networks, Boltzmann Machine. Fuzzy systems: Fuzzy sets – properties - membership functions - fuzzy operations, Applications, Implementation, Hybrid systems. Evolutionary computing: - Genetic algorithm – Schema theorem - Advances in the theory GA. Genetic Programming, Particle swarm optimization, Ant colony optimization, Artificial immune Systems. Applications: case studies may include image processing, digital systems, control, forecasting and time-series predictions.

Text Book / References

1. R.C. Eberhart, "Computational Intelligence: Concept to Implementations", Morgan Kaufmann Publishers, 2007.
2. Laurence Fausett, "Fundamentals of Neural Networks", Prentice Hall,1994
3. Timothy J Rose, "Fuzzy Logic with Engineering Applications", Third Edition, Wiley, 1995.
4. NazmulSiddique, HojjatAdeli, "Computational Intelligence: Synergies of Fuzzy Logic, NeuralNetworks and Evolutionary Computing", Willey 2013

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Understand the mathematical framework for the neuro, fuzzy and genetic algorithms and the way they differ from conventional optimisation techniques.	3	3	2	2	2	3
CO2	Decide and make use of appropriate neuro, fuzzy, genetic architectures for solving practical problems in Engineering domain.	2	3	2	3	3	3
CO3	Extend the skill acquired to understand the recent advancements in the field and for potential contributions.	3	3	3	3	3	3

Evaluation Pattern - 70:30

Periodical 1 - 10%
Periodical 2 - 10%
Lab Assignments & Case Study – 50%
End Semester Exam - 30%

21AM644 Foundations of Artificial Intelligence 3-0-2-4

Preamble

This course will deal with the fundamental principles of Artificial Intelligence including knowledge representation, reasoning, decision making and programming techniques. The course will also support developing an understanding of the theoretical relationships between these algorithms.

Course Objectives

- To understand basic principles of Artificial Intelligence
- To learn and design intelligent agents
- To understand the basic areas of artificial intelligence including problem solving, knowledge representation, reasoning, decision making, planning, perception and action

Course Outcomes

COs	Description
CO1	Understand formal methods of knowledge representation
CO2	Understand foundation principles, mathematical tools and program paradigms of AI.
CO3	Apply intelligent agents for Artificial Intelligence programming techniques
CO4	Apply problem solving through search for AI applications
CO5	Apply logic and reasoning techniques to AI applications.

Prerequisites

- None

Syllabus

Logic and Knowledge Representation - Knowledge base - Ontology - Commonsense Knowledge Representation of Commonsense knowledge – Graphical models – Belief networks - State space representation – Vector representation - Propositional logic and predicate logic - Propositional and predicate logic - Syntax - Informal and formal semantics - Validity, satisfiability - Semantic entailment - Equivalence - De Morgan's laws - Decidable problems - Many-sorted logic - first-order, aspects of higher-order logic.

Automated Reasoning– Formal program techniques: specification by pre- and post-conditions, derivation and verification of programs, invariants. Strategic Reasoning in AI - Agents, strategic behaviours of agents in multiagent systems (MAS) by using the language of alternating-time temporal logic (ATL), an extension of the temporal logics CTL and LTL which allows to express game-theoretical notions such as the existence of a winning strategy for a group of agents - Expert system-based reasoning - Production system, semantic network, and frame - Soft computing based reasoning – Fuzzy logic.

Decision Theory Decision-Making: basics of utility theory, decision theory, sequential decision problems, decision networks, elementary game theory, sample applications; Problem-solving through Search: forward and backward, state-space, blind, heuristic, hill climbing, best-first, A, A*, AO*, minimax, constraint propagation, intelligent search, meta-heuristics, problem-reduction, neural and stochastic; Intelligent agents - reactive, deliberative, goal-driven, utility-driven, and learning agentsArtificial Intelligence programming techniques; Planning: planning as search, partial order planning, construction and use of planning graph.

Text Book / References

1. Russell, Norvig, Artifificial Intelligence: A Modern Approach, Third edition, Prentice Hall,2010
2. Tsang. Foundations of constraint satisfaction, Academic press, 1993
3. Gendreau, Michel, and Jean-Yves Potvin, Handbook of metaheuristics, Springer, 2010.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Understand formal methods of knowledge representation	3		1			
CO2	Understand foundational principles, mathematical tools and program paradigms of AI	3		2			
CO3	Apply intelligent agents for Artificial Intelligence programming techniques	3	1	3	1		3
CO4	Apply problem solving through search for AI applications	2	3	2	3	3	3
CO5	Apply logic and reasoning techniques to AI applications	2	3	2	3	3	3

Evaluation Pattern - 70:30

Periodical 1 - 10%
Periodical 2 - 10%
Lab Assignments & Case Study – 50%
End Semester Exam - 30%

21AM645 Deep Learning 3-0-2-4

Preamble

With the advent of high end computing facilities and with the availability of huge amount of data, deep learning became the de facto standard machine learning strategy to learn complicated patterns and is offering the state of the art results in diverse fields including but not limited to automatic language translation, speech processing, medical diagnoses and in almost all fields of computer vision. This course provides core understanding in different deep learning architectures, design principles, learning strategies and encourages the usage of many deep learning tools in designing and deploying solutions.

Course Objectives

- To introduce to students, different deep neural network architectures, training strategies/algorithms, possible challenges, tools and techniques available in designing and deploying solutions to different practical/ Engineering problems.

Course Outcomes

COs	Description
CO1	Be able to design, train, deploy neural networks for solving different practical/engineering problems and analyse and report its efficacy
CO2	Have a good level of knowledge (Both Conceptual and Mathematical) on different neural network settings to pursue Research in this Field
CO3	Build skills in using established ML tools/libraries and in building self-learning skills in the field

Prerequisites

- Linear Algebra and Probability
- Computational Methods for Optimisation

Syllabus

Neural Networks basics – Linear Separable Problems and Perceptron – Multi layer neural networks and Back Propagation, Practical aspects of Deep Learning: Train/ Dev / Test sets, Bias/variance, Vanishing/exploding gradients, Gradient checking, Hyper Parameter Tuning.

Convolutional Neural Networks – Basics and Evolution of Popular CNN architectures – Transfer Learning–Applications : Object Detection and Localization, Face Recognition, Neural Style Transfer Recurrent Neural Networks – GRU – LSTM – NLP – Word Embeddings – Transfer Learning – Attention Models – Applications : Sentiment Analysis, Speech Recognition, Action Recognition.

Restricted Boltzmann Machine – Deep Belief Network – Auto Encoders – Applications: SemiSupervised classification, Noise Reduction, Non-linear Dimensionality Reduction.
Goal Oriented Decision Making – Policy and Target Networks – Deep Quality Network for Reinforcement Learning.
Introduction to GAN – Encoder/Decoder, Generator/Discriminator architectures.
Challenges in NN training – Data Augmentation – Hyper parameter Settings – Transfer Learning–Developing and Deploying ML Models (e.g., Matlab/Tensor Flow/PyTorch)

Text Book / References

1. Ian Goodfellow, YoshuaBengio and Aeron Courville,” Deep Learning”, MIT Press, First Edition, 2016.

2. Adam Gibson and Josh Patterson,” Deep Learning, A practitioner’s approach”, O’Reilly, FirstEdition, 2017.

3. Francois Chollet,” Deep Learning with Python”, Manning Publications Co, First Edition, 2018.

4. Research Papers on Relevant Topics and Internet Resources

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Be able to design, train, deploy neural networks for solving different practical/engineering problems and analyse & report its efficacy	3	3	3	3	2	1
CO2	Have a good level of knowledge (Conceptual & Mathematical) on different neural network settings to pursue Research in this Field	3	3	2	2	3	1
CO3	Build skills in using established ML tools/libraries and in building self-learning skills in the field	3	3	3	3	2	2

Evaluation Pattern - 70:30

Midterm Exam - 20%
Lab Assignments & Case Study – 50%
End Semester Exam - 30%

21AM646 Probabilistic Graphical Models 3-0-2-4

Preamble

Probabilistic graphical models use a graph-based representation for encoding complex distributions over a high-dimensional space. This course deals with representation, inference, and learning of probabilistic graphical models. Students will gain an in-depth understanding of several types of graphical models, basic ideas underlying exact inference in probabilistic graphical models and learning probabilistic models from data.

Course Objectives

- To enable students to model problems using graphical models
- To design inference algorithms
- To learn the structure of the graphical model from the data set

Course Outcomes

COs	Description
CO1	Understand the process of encoding probability distributions using graphs
CO2	Analyze the independence properties of the graph structure
CO3	Understand and analyze Markov networks for the graphical modeling of probability distributions
CO4	Familiarize methods that approximate joint distributions
CO5	Study and evaluate methods to learn the parameters of networks with known and unknown structures

Prerequisites

- Probability and Statistics
- Programming Languages
- Algorithm Design

Syllabus

Introduction: Probability distributions, random variables, joint distributions, random process, graphs, undirected and Directed Graphical Models. Representation: Bayesian Networks – Independence in graphs – d-separation, I-equivalence, minimal I-maps. Undirected Graphical models: Gibbs distribution and Markov Networks, Markov models and Hidden Markov Models. From Bayesian to Markov and Markov to Bayesian networks, Triangulation and Chordal Graphs.Directed Gaussian graphical models.Exponential Family Models.Factor Graph Representation.Conditional Random Fields. Other special Cases: Chains, Trees.

Inference: Variable Elimination (Sum Product and Max-Product). Junction Tree Algorithm.Forward Backward Algorithm (for HMMs).Loopy Belief Propagation. Markov Chain Monte Carlo. Metropolis Hastings. Importance Sampling. Gibbs Sampling. Variational Inference.

Learning Graphical models: Discriminative vs. Generative Learning., Density estimation, learning as optimization, maximum likelihood estimation for Bayesian networks, structure learning in Bayesian networks, Parameter Estimation in Markov Networks. Structure Learning. Learning undirected models- EM: Handling Missing Data.Applications in Vision, Web/IR, NLP and Biology.Advanced Topics: Statistical Relational Learning, Markov Logic Networks.

Text Book / References

1. Daphne Koller and Nir Friedman,"Probabilistic Graphical Models: Principles and Techniques",First Edition, MIT Press, 2009.
2. Michael Jordan,"Learning in Graphical Models". MIT Press, 1998. Collection of Papers.
3. Judea Pearl, Morgan Kaufmann, "Probabilistic Reasoning in Intelligent Systems", 1988.
4. Kevin P. Murphy, "Machine Learning, a probabilistic perspective", The MIT Press Cambridge,Massachusetts, 2012.
5. Darwiche Adnan, "Modeling and reasoning with Bayesian networks", Cambridge universitypress, 2009.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Understand the process of encoding probability distributions using graphs	3	2	3	1	2	
CO2	Analyze the independence properties of the graph structure	3	3	3	1	2	
CO3	Understand and analyze Markov networks for the graphical modeling of probability distributions	3	2	3	2	2	
CO4	Familiarize methods that approximate joint distributions	3	2	2	-	-	
CO5	Study and evaluate methods to learn the parameters of networks with known and unknown structures using real life data sets	3	3	4	2	3	2

Evaluation Pattern - 50:50

Periodical 1 - 15%
Periodical 2 - 15%
Continuous Evaluation – 20%
End Semester Exam - 50%

21AM701 Modern Compiler Design 2-0-2-3

Preamble

This course focuses on the interpretation and compilation techniques needed to obtain high performance and code generation for register machines as well as on compiler optimizations based on intermediate program representations on modern computer architectures. Program analysis and optimization techniques are also included.

Course Objectives

- To provide an understanding of foundations of data-flow analysis.
- To use dataflow analysis for program optimization, and code generation across basic blocks, procedures, and complete programs.
- To learn interprocedural and intraprocedural analysis, intermediate representations such as control-flow graphs in static single assignment (SSA).
- To understand dependence analysis and loop transformations: the building blocks for optimizing for memory hierarchies and parallel machines.

Course Outcomes

COs	Description
CO1	Analyse symbol tables and generating intermediate code.
CO2	Analyse Control flow and Data flow analysis.
CO3	Understand the compiler architecture.
CO4	Apply register allocation.
CO5	Be exposed to Compiler Optimization.

Prerequisites

- Basic course in compiler construction, corresponding to the undergraduate courses Compilers and Interpreters or Compiler Construction.
- Basic course in data structures and algorithms.
- Basic knowledge in processor architecture.

Syllabus

Overview of compiler -Symbol Table Structure - Intermediate Representations- Issues in Designing an Intermediate Language -High-Level ,Medium-Level,Low-Level ,Multi-Level Intermediate LanguagesIntroduction to Automatic Generation of Code Generators - Introduction to Semantics-Directed Parsing.

Control-Flow Analysis-Approaches to Control-Flow Analysis - Data Flow Analysis- Iterative DataFlow Analysis - Control-Tree-Based Data-Flow Analysis- Static Single-Assignment (SSA) FormBasic-Block Dependence DAGs- Dependencies in Loops - Constant-Expression Evaluation.

Early optimization-Redundancy Elimination - Redundancy Elimination and Reassociation Code Hoisting- Loop Optimizations-Variable Optimizations Tail-Call Optimization and Tail-Recursion Elimination-Unreachable-Code Elimination-Loop Simplifications-Loop Inversion -Branch OptimizationsConditional Moves - Dead-Code Elimination -Interprocedural Analysis and Optimization-Optimization for the Memory Hierarchy-Code Scheduling.

Text Book / References

1. Steven Muchnick, Advanced Compiler Design & Implementation, Morgan Kaufmann, August1997.
2. Alfred V. Aho, Monica S. Lam, Ravi Sethi and Jeffrey D. Ullman, Compilers: Principles,Techniques, and Tools, 2nd Edition, Addison-Wesley, 2006.
3. Keith Cooper and Linda Torczon, Engineering a Compiler, 2nd Edition, Morgan Kaufmann,2011.
4. Appel A. W.: Modern Compiler Implementation in Java. 2nd edition, Addison-Wesley, 2007.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Analyse symbol tables and generating intermediate code	2		2			
CO2	Analyse Control flow and Data flow analysis.	3		3	2		
CO3	Understand the compiler architecture.	3		1	1		
CO4	Apply register allocation.	2	2			2	
CO5	Be exposed to Compiler Optimization.		2	2		2	

Evaluation Pattern - 50:50

Periodical 1 - 15%
Periodical 2 - 15%
Continuous Evaluation – 20%
End Semester Exam - 50%

21AM702 Distributed Computing 2-0-2-3

Preamble

This course introduces the students to various design principles in distributed systems and the architectures for distributed systems. They will learn important distributed algorithms related to clock synchronization, concurrency control etc. They also get exposed to current distributed systems and platforms.

Course Objectives

- The course will stress upon important concepts of distributed systems.

- The course will pave the way for understanding the distributed algorithms and their usage in different applications.
- This will enable the students to work with real distributed systems and solve problems in those systems.

Course Outcomes

COs	Description
CO1	Understand the design principles in distributed systems and the architectures for distributed systems.
CO2	Apply various distributed algorithms related to clock synchronization, concurrency control, deadlock detection, load balancing, voting etc.
CO3	Analyze fault tolerance and recovery in distributed systems and algorithms for the same.
CO4	Analyze the design and functioning of existing distributed systems and file systems.
CO5	Implement different distributed algorithms over current distributed platforms.

Prerequisites

Syllabus

Introduction and types of distributed systems – architecture of DS- overview of processes - A Taxonomy of Distributed Systems, scalable performance, load balancing, and availability. Models of computation - shared memory and message passing system— synchronous and asynchronous systems. Communication in Distributed Systems - Remote Procedure Calls and Message Oriented Communications and implementation, High-level communication and publish-subscribe in Mapreduce.

Logical time and event ordering, Global state and snapshot algorithms, distributed snapshots in VMs, clock synchronization, Distributed mutual exclusion, Group based Mutual Exclusion, leader election, deadlock detection, termination detection, Distributed Databases, implementations over a simple distributed system and case studies of distributed databases and systems -Distributed file systems: scalable performance, load balancing, and availability. Examples from Dropbox, Google FS (GFS)/ Hadoop Distributed FS (HDFS), Bigtable/HBase MapReduce, RDD.

Consistency control: Data Centric Consistency, Client Centric Consistency, Replica Management, Consistency Protocols. Fault tolerance and recovery: basic concepts, fault models, agreement problems and its applications, commit protocols, voting protocols, check pointing and recovery. Case Studies from Apache Spark, Google Spanner, Amazon Aurora, BlockChain Systems etc.

Text Book / References

1. Andrew S. Tannenbaum and Maarten van Steen, Distributed Systems: Principles and Paradigms,Third Edition, Prentice Hall, 2017.
2. Ajay D. Kshemkalyani and Mukesh Singhal, Distributed Computing: Principles, Algorithms,and Systems, Cambridge University Press, 2011.
3. Garg VK. Elements of distributed computing. John Wiley & Sons; 2002.
4. George Coulouris, Jean Dollimore, Tim Kindberg and Gordon Blair, Distributed Systems: Concepts and Design, Fifth Edition, Pearson Education, 2017.
5. Fokkink W. Distributed algorithms: an intuitive approach. Second Edition, MIT Press; 2018.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6

CO1	Understand the design principles in distributed systems and the architectures for distributed systems	3	3	2	-	2	
CO2	Apply various distributed algorithms related to clock synchronization, concurrency control, deadlock detection, load balancing, voting etc.	2	3	3	2	3	
CO3	Analyze fault tolerance and recovery in distributed systems and algorithms for the same	2	2	3	3	3	
CO4	Analyze the design and functioning of existing distributed systems and file systems.	2	3	2	2	3	2
CO5	Implement different distributed algorithms over current distributed platforms	2	3	3	3	3	3

Evaluation Pattern - 70:30

Periodical 1 - 15%
Periodical 2 - 15%
Continuous Evaluation – 40%
End Semester Exam - 30%

21AM703 Computational Geometry 2-0-2-3

Preamble

Computational geometry is a branch of computer science devoted to the study of algorithms which can be stated in terms of geometry. Some purely geometrical problems arise out of the study of computational geometric algorithms, and such problems are also considered to be part of computational geometry. There has been increased traction to this field in finding solutions for real-life problem. For example, the brain behind Google maps is rooted in computational geometry and is playing a major role today. This course helps students to understand the intricacies of designing, analyzing and optimizing algorithms in computational geometry.

Course Objectives

- To introduce fundamental concepts of computational geometry.
- To understand the techniques and data structures that underlie geometric algorithms.
- To apply principles of geometric algorithms for a given application scenario.

Course Outcomes

COs	Description
CO1	Understand the fundamental concepts of computational geometry and apply to different components of computing systems.
CO2	Understand basic techniques applied in geometry algorithms.
CO3	Understand common data structures used in computational geometry.
CO4	Demonstrate knowledge in terms of relevance and potential of geometric algorithms for a given application.

Prerequisites

- Algorithms and Data Structures

Syllabus

Introduction to computational geometry, Convex hull in 2D space, Doubly-connected edge lists, Line segment intersection, Polygon triangulation, Voronoi diagrams.
Range searching: Kd-trees, Range trees, 2-3 trees, orthogonal range searching, Point location and trapeziodal maps, Closest and furthest pair problems.
Delaunay triangulation, Interval trees, Priority search trees, Segment trees, Convex hull in 3D space, Binary Space Partitions

Text Book / References

1. Mark de Berg, Otfried Cheong, Marc van Kreveld, Mark Overmars, "Computational Geometry– Algorithms and Applications", Third Edition, Springer, 2008.
2. Jianer Chen, "Computational Geometry – Methods and Applications", A & M University,Texas, 2019.
3. Satyan L. Devadoss, Joseph O'Rourke, "Discrete and Computational Geometry", PrincetonUniversity Press, 2011.
4. Laszlo Michael, "Computational Geometry and Computer Graphics in C++", Pearson Publishing, 2017.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Understand the fundamental concepts of computational geometry and apply to different components of computing systems.	1	1		1		
CO2	Understand basic techniques applied in geometry algorithms.	1	2		2	2	
CO3	Understand common data structures used in computational geometry.	2	2	2	3	1	
CO4	Demonstrate knowledge in terms of relevance and potential of geometric algorithms for a given application.	3	3	2	2	2	2

Evaluation Pattern - 70:30

Periodical 1 - 15%
Periodical 2 - 15%
Continuous Evaluation – 40% End
Semester Exam - 30%

21AM704 Full Stack Development with Python 2-0-2-3 Preamble

Full Stack Development is an indispensable course for computer science students. The course is concerned with end-to-end development of a three-tier web application. It deals with the frameworks necessary to implement front-end, back-end and database covering design, development and deployment. The course is designed to progress on both front-end and back-end in a synchronized fashion and leverages GitHub and Heroku for version control and deployment. The course includes a term project to reinforce the technologies learnt.

Course Outcomes

COs	Description
CO1	Use markup and scripting languages to design and validate dynamic web pages.
CO2	Customize pages for users need based on responsive web design concepts.

CO3	Learn to design appropriate database services based on the requirements.
CO4	Design, develop and deploy an end-to-end web application as a term project.

Prerequisites

- Programming and database fundamentals

Syllabus

Introduction to web development, Git and GitHub, Taxonomy of frameworks. HTML basics – structuring, positioning, alignment, CSS and JS basics, Browser development tools, Bootstrap basics. Basic Backend App serving text/HTML and HTML from templates.

Jinja template, Semantic tags, HTTP components – parameters, headers, cookies, sessions, Handling forms, Serve-Handle JSON/XML requests, Intro to jQuery, jQuery request handling and Ajax, More jinja templating, Lists and tables, DOM styling, Responsive design.

Database creation and connection, Creation of DB Schema from model, Adding relation between models, Intro to REST APIs, Basic CRUD app, Form and tables for CRUD services. Authentication, designing error pages, setup default error pages. Simple hosting on a public web host.

Text Book / References

1. Laura Lemay, Rafe Colburn, Jennifer Kyrnin, "Mastering HTML, CSS & JavaScript WebPublishing", Paperback, 2016.

2. Jon Duckett, "Web Design with HTML, CSS, JavaScript and jQuery", Paperback, 2014.

3. Miguel Grinberg, "The New And Improved Flask Mega-Tutorial", Paperback., 2017.

4. Kunal Relan, "Building REST APIs with Flask: Create Python Web Services with MySQL",Paperback, 2019.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Use markup and scripting languages to design and validate dynamic web pages.	1	1	2	1	1	
CO2	Customize pages for users need based on responsive web design concepts.	2	2	1	3	2	2
CO3	Learn to design appropriate database services based on the requirements.	3	2	1	2	1	1
CO4	Design, develop and deploy an end-to-end web application as a term project.	3	3	2	2	2	3

Evaluation Pattern - 70:30

Midterm Exam - 15%
Lab Assignments - 25%
Minor Project – 30%
End Semester Exam - 30%

21AM705 Wireless Communication Technologies 2-0-2-3

Preamble

The objective of the course is to learn about wireless technologies including cellular networks, infrastructure based networking, short range and emerging communication technologies.

Course Objectives

- To introduce current wireless networking technologies.
- To introduce the concepts infrastructure based communication technologies.
- To familiarize with short range communication technologies.

Course Outcomes

COs	Description
CO1	To understand the characteristics of wireless channels and fundamental limits on the capacity
CO2	Study the concepts of current wireless technologies and standards
CO3	Compare different cellular network technologies
CO4	Analyse and compare short range communication technologies
CO5	Explore emerging wireless technologies

Prerequisites

- Computer Networks

Syllabus

Overview of wireless communications, Types of wireless communication, Evolution of mobile radio communications, Mobile radio propagation- large scale path loss- free space propagation model, Physical modelling for wireless channels , Capacity of wireless Channel, Cellular telephone system..

Infrastructure based Communication technologies, Cellular mobile communication technologiesGSM, GPRS, EDGE, WCDMA, 4G LTE, 5G Small cells, Massive MIMO, Wireless local area networks - 802.11 (Wi-Fi), Wi-Fi Mesh, Wi-MAX, Mobile ad-hoc and sensor networks.

Short range communication technologies, Wireless person area networks, IEEE 802.15.1 Bluetooth, IEEE 802.15.4 ZigBee, 6LoWPAN, Wireless HART, Wave, Emerging Technologies - Millimeter wave systems- full duplex radios, software defined radios-cloud RAN- wireless charging.

Text Book / References

1. David Tse and Pramod Viswanath, “Fundamentals of Wireless Communication”, Wiley Seriesin Telecommunications, Cambridge University Press, 2005.
2. W.C.Y.Lee, “Mobile Cellular Telecommunications - Analog and Digital Systems”, 2nd Edition.Tata McGraw Hill, 2006.
3. Andrea Goldsmith, “Wireless Communications”, Cambridge University Press, 2005.
4. State of the art research papers on wireless communication technologies.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6

CO1	understand the characteristics of wireless channels and fundamental limits on the capacity	3	2	3	1		
CO2	Study the concepts of current wireless technologies and standards	3	2	3	1	1	
CO3	Compare different cellular network technologies	3	3	3	1	1	3
CO4	Analyse and compare short range communication technologies	3	3	2	3	3	2
CO5	Explore emerging wireless technologies	3	3	2	2	3	3

Evaluation Pattern - 70:30

Midterm Exam - 20%
Lab Assignments – 20%
Project – 30%
End Semester Exam - 30%

21AM706 Modeling and Simulation 2-0-2-3

Preamble

Modeling and Simulation is a course which provides insights into various types of simulation techniques. The course starts with basic concepts of Monte Carlo simulation and subsequently covers topics related to Discrete Event Simulation. Given a problem, the students will learn how to model the input data, fit a suitable distribution, and conduct an output analysis of the simulation. Once a model is built, it must go through verification and validation techniques. Thus a student can actually go through all the steps of building a simulator for a particular problem. Additionally, the course covers basic statistical probability distributions, hypothesis testing and application of these concepts in practical problems.

Course Objectives

- To make the students familiar with the basic steps of building a simulator;
- To fit a specific probability distribution for a given data set;
- To analyze the data quality of output of a simulation.

Course Outcomes

COs	Description
CO1	Understand fundamental concepts of modeling and simulation
CO2	Develop simulators to find out performance of simple application scenarios
CO3	Learn the techniques for random number generation and their properties
CO4	Understand and apply input modeling techniques
CO5	Understand and apply different statistical models in simulation
CO6	Perform output analysis of simulation and apply techniques for verification and validation of simulation models

Prerequisites

- Basic Probability and Statistics

Syllabus

Introduction to Simulation: System and system environment, Component System, Type of systems, Types of models, Steps in simulation study, Advantages and disadvantages of Simulation. Types of Simulation: Discrete Event Simulation, Simulation of a single server queuing system, Simulation of an Inventory system, Continuous Simulation, Predator-prey system, Combined Discrete-Continuous Simulation, Monte Carlo Simulation. Statistical Models in Simulation: Useful statistical model, Discrete and Continuous Probability distributions, Poisson process and Empirical distribution.

Random Numbers Generation: Properties of random numbers, Generation of pseudo random numbers, Techniques for generating random numbers, Tests for random numbers. Random Variate Generation: Inverse Transform technique, Convolution method, Acceptance Rejection Techniques. Input Modeling: Data Collection, Identifying the distribution of data, Parameter Estimation, Goodness of fit tests, Selection input model without data, Multivariate and Time series input models.

Verification and Validation of Simulation Model: Model Building, Verification and Validation, Verification of Simulation models, Calibration and Validation of models. Output Analysis: Stochastic nature of output data, Measure of performance and their estimation, Output analysis of terminating simulators, Output Analysis of steady state simulation. Comparison and Evaluation of Alternate System Design: Comparison of two system design, Comparison of several system design, Confidence interval for the difference between expected responses of two systems.

Text Book / References

1. Jerry Banks, John S. Carson, Barry L. Nelson, "Discrete-Event-System Simulation", 5/e,Pearson Education, 2010.
2. Averill M. Law, "Simulation Modeling and Analysis", 4/e, Tata McGraw-Hill, 2017.
3. Lawrence M. Leemis and Stephen K. Park, "Discrete – Event Simulation: A First Course",Pearson Education, 2006.
4. Bernard Zeigler, Alexandre Muzy and Ernesto Kofman, "Theory of Modeling and Simulation: Discrete Event & Iterative System Computational Foundations", 3/e, Academic Press, 2018.
5. John A. Sokolowski and Catherine M. Banks, "Principles Of Modeling And Simulation", JohnWiley, 2014.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Develop simulators to find out performance of simple application scenarios	-	3	2	-	2	3
CO2	Apply effective visualizations to explore and analyze input data	-	2	3	-	3	2
CO3	Learn the techniques for random number generation and their properties	-	-	2	-	2	
CO4	Understand and apply input modeling techniques	2	2	3	-	1	2
CO5	Understand and apply different statistical models in simulation	2	2	2	2	3	3
CO6	Perform output analysis of simulation and apply techniques for verification and validation of simulation models	2	2	3	-	2	2

Evaluation Pattern - 70:30

Periodical 1 - 15%
Periodical 2 - 15%

Continuous Evaluation – 40%
End Semester Exam - 30%

21AM707 Blockchain Technology 2-0-2-3

Preamble

Blockchain is the latest technology in the domain of Computer security and is capable of contributing on security aspects of many segments in industry and society.

Course Objectives

- Understand how blockchain systems (mainly Bitcoin and Ethereum) work.
- To securely interact with them.
- Design, build, and deploy smart contracts and distributed applications.
- Integrate ideas from blockchain technology into their own projects

Course Outcomes

COs	Description
CO1	Understand the concepts of cryptocurrency, blockchain, and distributed ledger technologies.
CO2	Analyse the application and impact of blockchain technology in the financial industry and other industries.
CO3	Evaluate security issues relating to blockchain and cryptocurrency.
CO4	Design and analyze the impact of blockchain technology.

Prerequisites

- None

Syllabus

History, definition, features, types, and benefits of block chain and bitcoin, Consensus, CAP theorem and blockchain. Decentralization – methods, routes, smart contracts, platforms. Symmetric and Asymmetric cryptography - Public and private keys, theoretical foundations cryptography with practical examples. Introduction to financial markets, use cases for block chain technology in the financial sector. Bitcoin, Transactions, Block chain, Bitcoin payments, technical concepts related to bitcoin cryptocurrency. Smart Contracts, definition of smart contracts, Ricardian contracts, Oracles, and the theoretical aspects of smart contracts. Ethereum 101 - design and architecture of the Ethereum block chain, Various technical concepts related to the Ethereum block chain that explains the underlying principles, features, and components of this platform in depth. Hyperledger – protocol and architecture. Case studies on alternative Blockchains.

Text Book / References

1. Mastering Blockchain - Distributed ledgers, decentralization and smart contracts explained,Imran Bashir, Packt Publishing Ltd, Second Edition, ISBN 978-1- 78712-544-5, 2017.
2. Mastering Bitcoin: Unlocking Digital Cryptocurrencies, Andreas Antonopoulos , O'Reilly Publishing, ISBN 978-0691171692, 2014.

3. Blockchain Basics: A Non-Technical Introduction in 25 Steps, Daniel Drescher , Apress, FirstEdition, 2017.

4. Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction, Arvind Narayanan,Joseph Bonneau, Edward Felten, Andrew Miller, and Steven Goldfeder, Princeton University Press (July 19, 2016).

5. The Business Blockchain: Promise, Practice, and Application of the Next Internet Technology.William Mougayar, Wiley; 1st edition (May 9, 2016).

6. Bitcoin: A Peer to Peer Electronic Cash System Satoshi Nakamoto Online 2009 https://bitcoin.org/bitcoin.pdf

7. VitalikButerinEthereum White Paper Online 2017 https://github.com/ethereum/wiki/wiki/WhitePaper

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Understand the concepts of cryptocurrency, blockchain, and distributed ledger technologies.	3	2	3	1	1	1
CO2	Analyse the application and impact of blockchain technology in the financial industry and other industries.	2	2	3	2	2	2
CO3	Evaluate security issues relating to blockchain and cryptocurrency.	3	3	3	3	3	3
CO4	Design and analyze the impact of blockchain technology.	2	2	2	3	3	1

Evaluation Pattern - 70:30

Periodical 1 - 15%
Periodical 2 - 15%
Continuous Evaluation – 40%
End Semester Exam - 30%

21AM708 GPU Architecture and Programming 2-0-2-3

Preamble

GPU accelerated processors are being actively used nowadays in general purpose and scientific computing. These massively parallel, off-the shelf devices are used to run compute-intensive and time consuming part of applications. This course introduces the students to the Single Instruction Multiple Thread (SIMT) architecture of modern GPUs and architecture-aware programming frameworks like Compute Unified Device Architecture (CUDA) and OpenCL. While CUDA programming model is a proprietary framework for the students to learn to interface with GPUs, OpenCL allows them to be familiarized with an open, heterogeneous parallel computing model. Modern day applications of GPUs are also introduced to the students through case studies.

Course Objectives

- To introduce the fundamentals of GPU computing architectures and programming models;
- To familiarize the student with GPU aware programming frameworks like proprietary NVIDIA CUDA(C) and open heterogeneous programming standards like OpenCL;

- To create GPGPU accelerated real world applications.

Course Outcomes

COs	Description
CO1	Understand the difference between different parallel programming architectures.
CO2	Knowledge of GPU aware programming using CUDA and OpenCL frameworks.
CO3	Design and develop GPU accelerated real-world simulations and applications.

Prerequisites

- Computer Architecture
- Programming Fundamentals
- Data Structures

Syllabus

Introduction to Parallel Programming – Types of Parallelism – SIMD and SIMT – GPU architectureThreads, Blocks and Grids- GPU Memory Organization- CUDA Programming Model- CUDA Memory Model- Multidimensional thread management with CUDA- Basic CUDA Programming Examples -CUDA Streams – Synchronization and Warp Scheduling, Optimization.

Introduction to OpenCL - OpenCL Device Architectures - Basic OpenCL Programming Model – OpenCL Memory Model - Concurrency and Execution Model - Dissecting a CPU/GPU - OpenCL for Heterogeneous Computing - OpenCL Implementation – examples.

Case study: Convolution, Video Processing, Histogram and Mixed Particle Simulation - OpenCL Extensions - OpenCL Profiling and Debugging – WebCL, Applications of GPU Architecture like Gaming, Computer Vision, etc.

Text Book / References

1. Benedict R. Gaster, Lee Howes, David, R. Kaeli, PerhaadMistry and Dana Schaa, ”Heterogeneous Computing with OpenCL”, Elsevier, 2013.
2. Jason Sanders, Edward Kandrot, ”CUDA by Example: An Introduction to General-PurposeGPU Programming”, Addison-Wesley Professional, 2010
3. Shane Cook, ”CUDA Programming: A Developer’s Guide to Parallel Computing with GPUs”,Newnes, 2012
4. AaftabMunshi, Benedict Gaster, Timothy G. Mattson, James Fung and Dan Ginsburg, ”OpenCLProgramming Guide”, Addison-Wesley Professional, 2011.
5. Ryoji Tsuchiyama, Takashi Nakamura, TakuroIizuka and Akihiro Asahara, ”The OpenCLProgramming Book”, Fixstars Corporation, 2010.
6. Matthew Scarpio, ”OpenCL in Action: How to Accelerate Graphics and Computations”, Manning Publications, 2011.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Understand the difference between different parallel programming architectures.	3	1	1	-	1	-

CO2	Knowledge of GPU aware programming using CUDA and OpenCL frameworks.	3	3	3	1	2	-
CO3	Design and develop GPU accelerated realworld simulations and applications.	3	3	3	3	3	2

Evaluation Pattern - 70:30

Midterm Exam - 20%
Lab Assignments – 20%
Project – 30%
End Semester Exam - 30%

21AM709 Information Retrieval 2-0-2-3

Preamble

The amount of data available in the web has been increasing beyond limits and automatic methods of information retrieval has gained remarkable significance. IR is an important problem in natural language processing (NLP) also. The purpose of this course is the study of the indexing, processing, and querying of textual data.

Course Objectives

- To study the fundamentals of information retrieval (IR)
- To elaborate on indexing, search, relevance, classification, organization and storage of information
- To focus on prominent computer algorithms and methods used in IR

Course Outcomes

COs	Description
CO1	Gain insights to different retrieval models like Boolean and vector space models
CO2	Back end construction of an IR system by designing data structures and maintaining different types of indexes
CO3	Learn different ways of user query handling techniques
CO4	Understand evaluation measures of an IR system
CO5	Be able to improve an IR system through feedback mechanisms and ranking through ML algorithms

Prerequisites

- Algorithms, Data Structures and DBMS
- Machine Learning
- Basic Probability, Statistics and Linear Algebra

Syllabus

Introduction to IR: Retrieval Models - Ranked Retrieval - Text Similarity Metrics - TokenizingStemming-Evaluations on benchmark text collections - Components of an information retrieval system.

Indexing for IR: Inverted Indices - Postings lists - Optimizing indices with skip lists - Proximity and phrase queries - Positional indices - Dictionaries and tolerant retrieval - Dictionary data structures - Wild-card queries- n-gram indices - Spelling correction and synonyms - Edit distance - Index construction - Dynamic indexing - Distributed indexing - real-world issues.

Relevance in IR: Parametric or fielded search - Document zones - Vector space retrieval model tf.idf weighting - queries as vectors - Computing scores in a complete search system - Efficient scoring and ranking - Evaluation in information retrieval: User happiness- Creating test collections: kappa measure-interjudge agreement - Relevance feedback and query expansion: Query expansion - Automatic thesaurus generation - Sense-based retrieval. Understanding the working and usage of Google Analytics, Lucene, Elasticsearch, Solr

Case study in topics like Image retrieval, search optimization, QDD, Learning to rank, Cross and multi-lingual IR, Privacy Preservation, IR Techniques for the Web.

Text Book / References

1. C. Manning, P. Raghavan, and H. Schu¨tze, "Introduction to Information Retrieval", Cambridge University Press, 2008.
2. R. Baeza-Yates and B. RibeiroNeto, "Modern Information Retrieval: The Concepts and Technology behind Search", Second Edition, Addison Wesley, 2011.
3. David A. Grossman and OphirFrieder "Information Retrieval: Algorithms and Heuristics",Second Edition, Springer 2004.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Gain insights to different retrieval models like Boolean and vector space models	3	3	3	1	1	-
CO2	Back end construction of an IR system by designing data structures and maintaining different types of indexes	3	3	3	1	1	-
CO3	Learn different ways of user query handling techniques	1	3	3	1	1	2
CO4	Understand evaluation measures of an IR system	1	2	2	-	-	-
CO5	Be able to improve an IR system through feedback mechanisms and ranking through ML algorithms	3	3	3	1	1	-

Evaluation Pattern - 70:30

Periodical 1 - 15%
Periodical 2 - 15%
Continuous Evaluation – 40%
End Semester Exam - 30%

21AM710 Networks and Spectral Graph Theory 2-0-2-3

Preamble

Network science is an inter disciplinary field that combines mathematics, social science, computer science and many other areas. This course is essentially brings an understanding on the behavior of networked systems such as the Internet, social networks, and biological networks.

Course Objectives

- Exploring graph models in networked systems; understanding the structure and the behavior.
- Empirical study and hands-on experience on social networks and other systems

Course Outcomes

COs	Description
CO1	Understanding the key concepts in network graphs
CO2	Apply a range of measures and models for characterizing network structure
CO3	Define methodologies for analyzing networks of different fields
CO4	Apply graph algorithms to different networks
CO5	Demonstrate knowledge of network graphs with the help of software tools such NetworkX and Gephi

Prerequisites

- Primary knowledge of linear algebra and familiarity with graphs.
- Working knowledge in *Python for data science*.

Syllabus

Graphs and Networks- Review of basic graph theory, Mathematics of networks- Networks and their representation, Graph spectra, Graph Laplacian, Structure of complex networks, Clustering, Community structures, Social networks - the web graph, the internet graph, citation graphs. Measures and metrics- Degree centrality, Eigenvector centrality, Katz centrality, PageRank, Hubs and authorities, Closeness centrality, Betweenness centrality, Transitivity, Reciprocity, Similarity, assortative mixing.

Networks models - Random graphs, Generalized random graphs, The small-world model, Exponential random graphs, The large-scale structure of networks- small world effect, Degree distributions, Power laws and scale-free networks; Structure of the Internet, Structure of the World Wide Web. Fundamental network algorithms- Graph partitioning, Maximum flows and minimum cuts, Spectral graph partitioning, Community detection, Girvan and Newman Algorithm, Simple modularity maximization, Spectral modularity maximization, Fast methods based on the modularity.

Models of network Formation-Preferential attachment, Model of Barabasi and Albert, Vertex copying models, Network optimization models; Epidemics on networks- Models of the spread of disease, SI model, SIR model, SIS model, SIRS model; Network Search-Web search, Searching distributed databases. Graph databases like Neo4j, Graph Convolutional Neural Networks, Graph algorithms and implementation using *NetworkX* and *Gephi*.

Text Book / References

1. M.E.J. Newman, "Networks: An Introduction", Oxford University Press, 2010.
2. Dougles West, "Introduction to Graph Theory", Second Edition, PHI Learning Private Limited, 2011.
3. Guido Caldarelli, "Scale-Free Networks", Oxford University Press, 2007.
4. Alain Barrat, Marc Barthelemy and Alessandro Vespignani, "Dynamical processes on Complexnetworks", Cambridge University Press, 2008.

5. Reuven Cohen and ShlomoHavlin, “Complex Networks: Structure, Robustness and Function”,Cambridge University Press, 2010.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Understanding the key concepts in network graphs		3		1		
CO2	Apply a range of measures and models for characterizing network structure	3	2		3		
CO3	Define methodologies for analyzing networks of different fields	3		2			2
CO4	Apply graph algorithms to different networks			2	3		
CO5	Demonstrate knowledge of network graphs with the help of software tools such NetworkX and Gephi				3	3	3

Evaluation Pattern - 70:30

Midterm Exam - 20%
Lab Assignments – 20%
Project – 30%
End Semester Exam - 30%

21AM711 Reinforcement Learning 2-0-2-3

Preamble

Artificial intelligence techniques face challenges in learning from dynamic environment with minimal data. This course deals with various algorithms to learn such an environment. Elements of Reinforcement Learning, Model Based Learning, Temporal Difference Learning and Policy Search are the main focus topics of this course.

Course Objectives

- Good understanding of various types of algorithms for Reinforcement Learning
- Be able to design an RL system

Course Outcomes

COs	Description
CO1	Understand the relevance of Reinforcement Learning and how does it complement other ML techniques.
CO2	Understand various RL algorithms.
CO3	Formulate a problem as a Reinforcement Learning problem and solve it
CO4	Implement RL algorithms using Python

Prerequisites

- Machine Learning basics
- Linear Algebra and Probability Theory basics
- Python Programming

Syllabus

Introduction to Machine Learning and its various types, Motivation and Introduction to Reinforcement Learning, Multi arm Bandits ; Markov Decision Process, Value functions; Dynamic programming : Policy evaluation and improvement, Value iteration and Policy iteration algorithms.

Value prediction problems : Temporal difference learning in finite state spaces Algorithms for large state spaces Control : Closed loop interactive learning, online and active learning in bandits, Q learning in finite MDPs, Q learning with function approximation.

On policy approximation of action values : Value Prediction with Function Approximation, GradientDescent Methods, Policy approximation : Actor critic methods, Monte Carlo Methods : Monte carlo prediction, estimation of action values, off policy prediction via importance sampling

Text Book / References

1. Sutton and Barto, Reinforcement Learning: An Introduction, The MIT Press Cambridge,Massachusetts London, England, 2015
2. Csaba Szepesvari, Algorithms for Reinforcement Learning, Morgan & Claypool, United States,2010

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Understand the relevance of Reinforcement Learning and how does it complement other ML techniques.	3	1	2	1	-	-
CO2	Understand various RL algorithms	3	1	2	1	-	-
CO3	Formulate a problem as a Reinforcement Learning problem and solve it	-	3	2	3	3	3
CO4	Implement RL algorithms using Python	3	3	3	3	-	2

Evaluation Pattern - 70:30

Midterm Exam - 20%
Lab Assignments – 25%
Project – 25%
End Semester Exam - 30%

21AM712 Machine Learning for Big Data 2-0-2-3

Preamble

This course deals with two aspects of big data analytics. The first one is the infrastructure for big data analytics. Introduction to tools and algorithms that can be used to generate models from big data and to scale those models up to big data problems. Spark framework is the chosen platform. The second is the understanding and implementation of scalable and streaming algorithms to analyze voluminous data that is growing exponentially.

Course Objectives

- To understand various scalable machine learning algorithms to solve big data problems.
- To understand the SPARK architecture

- To implement Machine Learning algorithms using PySpark

Course Outcomes

COs	Description
CO1	Understand how Machine learning algorithm is made scalable to solve big data problems.
CO2	Implement scalable Machine Learning algorithms using PySpark.
CO3	Apply and compare different strategies for big data analytics using various machine learning algorithms
CO4	Understand Streaming algorithms and Coreset concept to analyze voluminous and high dimensional data

Prerequisites

- Machine Learning

Syllabus

Introduction to Spark : Spark Architecture, Spark Jobs and APIs. Resilient Distributed DatasetsCreating RDDs, Transformation, Actions. Dataframes- Python to RDD communications,Creating Dataframes, Dataframe queries. MLlib -Loading and Transforming the data. Implementation of Machine Learning algorithms such as Classification and Clustering using the MLlib.

Approaches to Modelling- Importance of Words in Documents - Hash Functions- Indexes - Secondary Storage -The Base of Natural Logarithms - Power Laws - Map Reduce. Finding similar items: Shingling – LSH - Distance Measures. Mining Data Streams: Stream data model - Sampling data - Filtering streams. Link Analysis: Page Rank, Link Spam.

Frequent Item Sets: Market Basket Analysis, A-Priori Algorithm - PCY Algorithm, Big data Clustering: Clustering in Non-Euclidean Spaces, BFR, CURE. Structured Streaming : Spark Streaming, Application dataflow. Coresets: Coresets for K-means,K-median clustering.

Text Book / References

1. AnandRajaRaman, Jure Leskovec and J.D. Ullman, "Mining of Massive Data sets", e-book,Publisher, 2014.
2. Kevin P. Murphey, "Machine Learning, a Probabilistic Perspective", The MIT PressCambridge, Massachusetts, 2012.
3. Tomasz Drabas, Denny Lee , "Learning Pyspark", Packt,February 2017.
4. Jeff M. Phillips, "Coresets and Sketches",arXiv:1601.00617,2016

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Understand how machine learning algorithm is made scalable to solve big data problems.	2	3	3	1	1	
CO2	Implement scalable Machine Learning algorithms using PySpark	1	3	3	2	1	2
CO3	Apply and compare different strategies for big data analytics using various machine learning algorithms	2	3	3	3	1	3
CO4	Understand Streaming algorithms and Core set concept to analyze voluminous and high dimensional data	3	3	2	2	2	3

Evaluation Pattern - 70:30

Midterm Exam - 20%
Lab Assignments – 25%
Project – 25%
End Semester Exam - 30%

21AM713 Applications of Machine Learning 2-0-2-3 Preamble

Voluminous and high dimensional data persist in almost all domains. This course deals with applications of machine learning in various domains to solve complex optimization problems such as recommendation systems, web advertising, and customer segmentation. The entire life cycle of data analytics is dealt with in this course.

Course Objectives

- To understand the design and implementation strategies of various applications of machine learning
- To apply techniques of preprocessing, model generation and evaluation on a given dataset from a particular domain.
- To compare different strategies of machine learning on a particular application

Course Outcomes

COs	Description
CO1	Understand how Machine learning is applied to solve problems in various applications like game playing, recommendation systems, high dimensional analysis, and targeted web advertising
CO2	Present and Implement ML algorithms to solve real world problems
CO3	Apply and compare different types of Machine learning approaches for a given application problem in the context of performance
CO4	Design a machine learning system by incorporating various components of ML and evaluate the performance

Prerequisites

- Machine Learning

Syllabus

Review of machine learning Concepts, Design of ML system – data cleaning, feature engineering, model selection, model building & fine tuning, and model deployment. Bias, variance, learning curves, and error analysis.Recommendation Systems – Model for Recommendation Systems, Utility Matrix, Content-Based Recommendations, Discovering Features of Documents, Collaborative Filtering. Usage of UV and NMF decomposition in Recommendation systems Association Rule Mining - Market-Basket Analysis, Frequent Itemset, Apriori Algorithm, Handling Larger Datasets - PCY Algorithm for Association Rule mining.

Advertising on the Web: Issues in Online Advertising, Online and offline algorithms, The matching Problem, The AdWords Problem, The Balance Algorithm, A Lower Bound on Competitive Ratio for Balance. Customer segmentation – Subspace Clustering, Types of Subspace clustering, Top down and bottom up approach : PROCLUS and , CLIQUE and their applications in Indexing in databases. Application of dimensionality reduction-SVD for Latent Semantic Indexing, CUR for approximate query processing from databases, PCA, for Image Processing – compression, identification and Visualization.

Sparse models, State space models, Markov Decision Process, Bellman equations, Value iteration and Policy iteration, Linear Quadratic Regulation(LQR), Non-linear dynamics to LQR, Linear Quadratic Gaussian(LQG), Independent component Analysis(ICA) for speech processing

Text Book / References

1. AnandRajaRaman, Jure Leskovec and J.D. Ullman, "Mining of Massive Data sets", e-book,Publisher, 2014.
2. Kevin P. Murphey, "Machine Learning, a Probabilistic Perspective", The MIT PressCambridge, Massachusetts, 2012.
3. Selected Journal papers to be given as case study from each module.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Understand how Machine learning is applied to solve problems in various applications like game playing, recommendation systems, high dimensional analysis, and targeted web advertising	3	3	3	3	3	2
CO2	Present and Implement ML algorithms to solve real world problems	3	3	3	2	2	2
CO3	Apply and compare different types of Machine learning approaches for a given application problem in the context of performance	2	3	3	2	2	2
CO4	Design a machine learning system by incorporating various components of ML and evaluate the performance	3	3	3	3	2	1

Evaluation Pattern - 70:30

Midterm Exam - 20%
Lab Assignments – 25%
Project – 25%
End Semester Exam - 30%

21AM714 Multi Agent Systems 2-0-2-3

Preamble

Agents are programs that sense their environment, make decisions based on these sensations, and execute actions. Multiagent systems are collections of multiple agents that interact with one another. This course provides a broad introduction to multiagent systems including agent architectures, interagent communication, rational decision making,agent modeling and multiagent learning.

Course Objectives

- To understand key concepts, methods, and algorithms that form the core of multiagent systems

- To designing agent systems as complex distributed software systems
- Understand various application where agent programs can be leveraged

Course Outcomes

COs	Description
CO1	Develop a good understanding of fundamental concepts of agent systems and requirement of multi agent systems.
CO2	Formulate and design an agent system for a given problem
CO3	Concrete implementations of agent systems

Prerequisites

- Nothing

Syllabus

Intelligent Agents - Abstract architectures for Intelligent agents - Concrete architectures for Intelligent agents - Agent communications - Agent interaction protocols - Societies of Agents - Search algorithms for agents - Constraint satisfaction problems - Path-Finding Problem.

Distributed Problem solving and Planning - Task and Result sharing - Planning representations and execution - Learning in Multiagent systems - Credit assignment problem - Interactive Reinforcement Learning of coordination -Learning and communication - Partial Order Planning – Graphs – Non deterministic Domains Conditional Planning - Continuous Planning – Multi Agent Planning.

Higher level Agents, Knowledge in Learning- Statistical Learning Methods -Logics for Multiagent systems - Possible-Worlds Semantics for Modal Logics - Normal Modal Logics - Epistemic Logic for Multiagent Systems. Industrial and Practical Applications - Agents for Workflow and Business Process Management - Agents for Electronic Commerce - Agents for Human-Computer Interfaces Agents for Virtual Environments - Agents for Social Simulation .

Text Book / References

1. Stuart Russell and Peter Norvig, "Artificial Intelligence - A Modern Approach", 2nd Edition,Prentice Hall, 2002.
2. Multiagent Systems A Modern Approach to Distributed Modern Approach to Artificial Intelligence, MIT Press, 2000
3. Gerhard Weiss, "Multiagent Systems", Second Edition, MIT Press, 2016
4. Michael Wooldridge, "An Introduction to Multi Agent System", John Wiley, 2002.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Develop a good understanding of fundamental concepts of agent systems and requirement of multi agent systems	3		3		1	
CO2	Formulate and design an agent system for a given problem	3	2	3	2	3	1
CO3	Concrete implementations of agent systems	3	3	3	2	2	1

Evaluation Pattern - 70:30

Periodical 1 - 15%
Periodical 2 - 15%

Continuous Evaluation – 40%
End Semester Exam - 30%

21AM715 Deep Learning for Biomedical Data Analysis 2-0-2-3

Preamble

This course focuses on fundamental practical aspects of tools and techniques for biomedical data analysis. Being an interdisciplinary course, it combines fundamental theory, tools and techniques of Artificial Intelligence, Machine Learning and Deep Learning so as to build diagnostic/assistive and analysis tools for biomedical signals such as ECG, EEG and CT & MRI images.

Course Objectives

- This course provides the state-of-the-art deep learning algorithms applied in biomedical data analysis.
- This course provides the practical skills required to implement the state-of-the-art deep learning algorithms for biomedical image segmentation.
- This course provides the basics of deep generative models for biomedical image analysis

Course Outcomes

COs	Description
CO1	To understand the fundamentals of state-of-the-art deep learning algorithms applied in biomedical data analysis.
CO2	To implement state-of-the-art deep learning algorithms for biomedical data analysis.
CO3	To implement Generative Adversarial Networks for data augmentation.

Prerequisites

- None.

Syllabus

Introduction to ECG, EEG, MRI and CT datasets – Hybrid Deep Learning Models for ECG Signal classification – Federated Transfer Learning for EEG Signal Classification - Benchmark Deep Learning Algorithms for Biomedical Image Segmentation: SwinUNet : UNet like Pure Transformer – FANet : Feedback Attention Network – MedT: Medical Transformer – Generative Adversarial Network for Synthetic Data Augmentation.

Text Book / References

1. Goodfellow I, Bengio Y, Courville A, & Bengio Y, “Deep learning”, Cambridge: MIT Press,1st Edition, 2016.
2. Michael Nielsen, “Neural Networks and Deep Learning”, Goodreads (eBook), 2013.
3. Bengio Y, “Learning Deep Architectures for AI, Foundations and Trends in Machine Learning”,nowpublishers, 2009.
4. Weblink: https://paperswithcode.com/task/ecg-classification .
5. Weblink: https://paperswithcode.com/task/eeg .

6. Weblink: https://paperswithcode.com/task/medical-image-segmentation .
7. Weblink: https://github.com/xinario/awesome-gan-for-medical-imaging .

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	To understand the fundamentals of state-ofthe-art deep learning algorithms applied in biomedical data analysis.	3	2	2	2	2	3
CO2	To implement state-of-the-art deep learning algorithms for biomedical data analysis.	2	3	3	3	2	2
CO3	To implement Generative Adversarial Networks for data augmentation.	3	2	2	2	2	3

Evaluation Pattern - 70:30

Midterm Exam - 20%
Lab Assignments – 25%
Project – 25%
End Semester Exam - 30%

21AM716 Artificial Intelligence for Robotics 2-0-2-3

Preamble

In recent years, several off-the-shelf robots have become available and some of them have made their way into our homes, offices, and factories. The ability to program robots has therefore become an important skill; e.g., for robotics research as well as in several companies (such as iRobot, ReThink Robotics, Willow Garage, medical robotics, and others). We study the problem of how a robot can learn to perceive its world well enough to act in it, to make reliable plans, and to learn from its own experience. The focus will be on algorithms and machine learning techniques for autonomous operation of robots.

Course Objectives

- To understand the principles of reinforcement learning which is one of the key learning techniques for robots
- To understand uncertainty handling in robotics through probabilistic approaches
- To learn how measurements work for robots

Course Outcomes

COs	Description
CO1	Learn the foundations of reinforcement learning for robotics
CO2	Understand basic probabilistic principles behind Robotics intelligence
CO3	Learn different measurement techniques for robotics
CO4	Understand POMDP and its significance for robotics
CO5	Implement principles of robotics intelligence for solving real world problems

Prerequisites

- Data Structures and algorithms
- Foundation of Data Science

- Linear Algebra and Optimization
- Principles of AI and ML

Syllabus

Overview: Robotics introduction, historical perspective on AI and Robotics, Uncertainty in Robotics Reinforcement Learning: Basic overview, examples, elements, Tabular Solution Methods - Multiarmed bandits, Finite Markov decision process, Dynamic programming (Policy Evaluation, Policy Iteration, Value Iteration), Monte Carlo Methods, Temporal-Difference Learning (Q-learning, SARSA).

Approximate Solution Methods - On-policy Prediction with Approximation, Value function approximation, Non-linear function approximation, Reinforcement Learning in robotics, Recursive state estimation: Robot Environment Interaction, Bayes filters, Gaussian filters – The Kalman filter, The Extended Kalman Filter, The information filter, The particle filter Robot motion: Velocity Motion Model, Odometry Motion Model, Motion and maps.

Measurement: Beam Models of Range Finders, Likelihood Fields for Range Finders, CorrelationBased Sensor Models, Feature-Based Sensor Models, Overview of POMDP.

Text Book / References

1. Sebastian Thrun, Wolfram Burgard, Dieter Fox, Probabilistic Robotics, MIT Press 2005
2. Richard S. Sutton, Andrew G. Barto, Reinforcement Learning: An Introduction", Secondedition, MIT Press, 2018
3. Jens Kober, Jan Peters, Learning Motor Skills: From Algorithms to Robot Experiments,Springer, 2014
4. Francis X. Govers, Artificial Intelligence for Robotics, Packt, 2018

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Learn the foundations of reinforcement learning for robotics	3	-	3	2	1	
CO2	Understand basic probabilistic principles behind Robotics intelligence	3	3	-	1	-	2
CO3	Learn different measurement techniques for robotics	2	3		-	1	
CO4	Understand POMDP and its significance for robotics	-	-	3	3	-	
CO5	Implement principles of robotics intelligence for solving real world problems	-	-	3	2	2	3

Evaluation Pattern - 70:30

Midterm Exam - 20%
Lab Assignments – 25%
Project – 25%
End Semester Exam - 30%

21AM717 Cloud and Big Data Analytics 2-0-2-3

Preamble

There is an unprecedented amount of data that is generated in today's world by both humans and machines. Being able to store, manage, analyze, and building intelligent applications has a critical impact on business, scientific discovery, social and environmental challenges. This course helps students to use cloud platform with its tools and distributed computing techniques to quickly build prototypes and applications for scalable data and workloads.

Course Objectives

- To introduce principles of cloud computing and build applications on cloud platforms.
- To understand distributed computing paradigms and its implementations on cloud platform.
- To apply principles of Big Query for handling big data.

Course Outcomes

COs	Description
CO1	To introduce principles of cloud computing.
CO2	Develop and deploying applications on cloud platform.
CO3	Understand Distributed Machine learning with hadoop and spark.
CO4	Create data analytics applications on distributed cloud computing platforms for using Spark and its Tools.
CO5	Develop methods to handle Big Data on Cloud using Big Query.

Prerequisites

- Basics of Machine Learning

Syllabus

Cloud computing fundamentals - Principles of Cloud Computing Systems, Elastic Cloud Systems for Scalable Computing, Cloud Architectures Compared with Distributed Systems, Service Models, Ecosystems and Scalability Analysis, Building Compute Service - Storage Service – Databases Service - Serverless Models on Cloud.

Frameworks for Big data: Hadoop – Hadoop Framework – Hadoop Daemon - Map Reduce Programming- Hadoop Ecosystem - Spark - Framework – RDD – Advanced RDD - Structured data - SQL, Dataframes, and Datasets – Streaming in Spark - Spark Distributed Processing - Building Spark ML on Cloud platform.

Cloud dataflow – dataflow templates, data transformation with cloud dataflow, working with apache beam, cloud publisher subscriber - architecture, message flow, implementation. Cloud data processing. Big Query architecture – Big Query vs map reduce – Big Query vs redshift – Big Query vs Athena.

Text Book / References

1. Kai Hwang, "Cloud Computing for Machine Learning and Cognitive Applications", MIT Press,2017.
2. Murari Ramuka, "Data Analytics with Google Cloud Platform ", BPB PUBN, 2019.
3. Anand Deshpande, Manish Kumar, Vikram Chaudhari, "Hands-On Artificial Intelligence onGoogle Cloud Platform", Packt Publishing, 2020.
4. Jeffrey Jackovich, Ruze Richards, "Machine Learning with AWS", Packt Publishing, 2018.

5. Jules S. Damji, Brooke Wenig, Tathagata Das, Denny Lee, “Learning Spark Lightning fastdata analysis”, O’Reilly Media, Inc, 2020.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	To introduce principles of cloud computing.	3	2	2			
CO2	Develop and deploying applications on cloud platform.	2	3			3	2
CO3	Understand Distributed Machine learning with hadoop and spark.	2	2	3	3		3
CO4	Create data analytics applications on distributed cloud computing platforms for using Spark and its Tools.	2		3		3	3
CO5	Develop methods to handle Big Data on Cloud using Big Query.	2		3		2	

Evaluation Pattern - 70:30

Midterm Exam - 20%
Lab Assignments – 25%
Project – 25%
End Semester Exam - 30%

21AM718 Quantum Artificial Intelligence 2-0-2-3

Preamble

This course deals with how to use quantum algorithms in artificial intelligence. The course also covers Quantum physics based information and probability theory, and their relationships to artificial intelligence by associative memory and Bayesian networks. Students will get an introduction to the principles of quantum computation and its mathematical framework.

Course Objectives

- To understand how the physical nature, as described by quantum physics, can lead to algorithms that imitate human behavior
- To explore possibilities for the realization of artificial intelligence by means of quantum computation
- To learn computational algorithms as described by quantum computation

Course Outcomes

COs	Description
CO1	Understand the computation with Qubits
CO2	Apply Quantum algorithms - Fourier Transform and Grover’s amplification
CO3	Apply Quantum problem solving using tree search
CO4	Understand and explore the models of Quantum Computer and Quantum Simulation tools
CO5	Explore open source Quantum computer libraries for applications

Prerequisites

- Machine Learning
- Programming languages
- Probability

Syllabus

Introduction - artificial intelligence - computation - Cantor's diagonal argument - complexity theory - Decision problems - P and NP - Church–Turing Thesis - Von Neumann architecture - Problem Solving - Rules - Logic-based operators - Frames - Categorial representation - Binary vector representation - Production System - Deduction systems - Reaction systems - Conflict resolution - Human problem-solving - Information and measurement - Reversible Computation - Reversible circuits Toffoli gate.

Introduction to quantum physics - Unitary Evolution - Quantum Mechanics - Hilbert space - Quantum Time Evolution - Von Neumann Entropy - Measurement - Heisenberg's uncertainty principle - Randomness - Computation with Qubits - Computation with m Qubit - Matrix Representation of Serial and Parallel Operations - Quantum Boolean Circuits - Periodicity - Quantum Fourier Transform - Unitary Transforms - Search and Quantum Oracle - Grover's Amplification - Circuit Representation - Speeding up the Traveling Salesman Problem - The Generate-and-Test Method Quantum Problem-Solving - Heuristic Search - Quantum Tree Search - Tarrataca's Quantum Production System.

A General Model of a Quantum Computer - Cognitive architecture - Representation - Quantum Cognition - Decision making - Unpacking Effects - Quantum walk on a graph - Quantum annealing - Optimization problems - Quantum Neural Computation - Applications on Quantum annealing Computer - Development libraries - Quantum Computer simulation tool kits.

Text Book / References

1. Andreas Wichert, Principles of Quantum Artificial Intelligence, First edition, World ScientificPublishing, 2014
2. Peter Wittek, Quantum Machine Learning, First edition, Academic Press, 2014

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Understand the computation with Qubits	2	2	2	–	3	
CO2	Apply Quantum algorithms - Fourier Transform and Grover's amplification	2	2	2	–	3	
CO3	Apply Quantum problem solving using tree search	3	3	3	2	3	
CO4	Understand and explore the models of Quantum Computer and Quantum Simulation tools	3	3	3	3	3	1
CO5	Explore open source Quantum computer libraries for applications	3	3	2	2	3	2

Evaluation Pattern - 70:30

Periodical 1 - 15%
Periodical 2 - 15%
Continuous Evaluation – 40%
End Semester Exam - 30%

21AM719 Natural Language Processing 2-0-2-3

Preamble

This course introduces the fundamental concepts and techniques of Natural Language Processing (NLP). Students will gain an in-depth understanding of the computational properties of natural languages and the commonly used algorithms for processing linguistic information. The course examines NLP models and algorithms using both the traditional symbolic and the more recent statistical approaches.

Course Objectives

- Understand leading trends and systems in natural language processing
- Describe concepts of morphology, syntax, semantics and pragmatics of the language
- Understand Language Models and its evaluation
- Describe and analyse language · POS tagging and context free grammar for English language •

  Writing programs in Python to carry out natural language processing
- Implement deep learning algorithms in Python and learn how to train deep networks for NLP

Course Outcomes

COs	Description
CO1	Understand POS tagging and CFG for English language
CO2	Apply NLP algorithms - Latent Dirichlet Allocation and Latent Semantic Indexing
CO3	Understand and explore the models of NLP and NLP tools BERT,XLNet
CO4	Explore NLTK toolkit for implementing NLP applications

Prerequisites

- Basics of Machine Learning
- Python Programming language
- Basics of Probability

Syllabus

Introduction - terminologies - empirical rules – Statistical Properties of words – Probability and NLP – Vector Space Models - Pre-processing- Tokenization, Parts-Of-Speech(POS) tagging, chunking, syntax parsing, Dependency parsing. NLP Applications: Named Entity Recognition, Sentiment analysis, Text categorization: Basic supervised text categorization algorithms, including Naive Bayes, k Nearest Neighbor (kNN) and Logistic Regression. Topic modeling: SVD and Latent semantic Indexing, Probabilistic Latent Semantic Indexing (pLSI) and Latent Dirichlet Allocation (LDA).

Introduction to Deep Learning: Neural Networks Basics, Feedforward Neural Network, Recurrent Neural Networks, LSTM, An Introduction to Transformers and Sequence-to-Sequence Learning. Neural Networks for NLP – Vector Representation of words – Contextual Understanding of text – Co-occurrence of matrix – N-grams – Dense Word Vector. Word2Vec – CBOW and Skip-gram Models – One-word learning architecture- Forward pass for Word2Vec – Reduction of complexity – sub-sampling and negative sampling. Continuous Skip-Gram Model, GloVe, BERT,XLNet.

Historical Approaches to Machine Translation – Statistical Machine Translation – Translation Models – Healthcare Data analysis and Text visualization: Summarizing lengthy blocks of narrative text, such as a clinical note or academic journal article. Answering unique free-text queries that require the synthesis of multiple data sources. Introduce Mathematical and programming tools to visualize a large collection of text documents.

Text Book / References

1. C.D. Manning et al, "Foundations of Statistical Natural Language Processing," MitPress. MITPress, 1999. isbn: 9780262133609.
2. James Allen, "Natural Language Processing with Python", O'Reilly Media, July 2009.
3. NiladriSekhar Dash and S. Arulmozi, Features of a Corpus. Singapore: Springer Singapore,2018, pp. 17–34. ISBN: 978-981-10-7458-5.
4. Ian Goodfellow, YoshuaBengio, and Aaron Courville, Deep Learning,http://www.deeplearningbook.org. MIT Press, 2016.
5. NitinIndurkhya and Fred J Damerau, "Handbook of natural language processing," Chapmanand Hall/CRC, 2010.
6. Daniel Jurafsky and James H. Martin "Speech and Language Processing: An Introductionto Natural Language Processing, Computational Linguistics, and Speech Recognition," 1st. Upper Saddle River, NJ, USA: Prentice Hall PTR, 2000. isbn: 0130950696.
7. Christopher D. Manning, PrabhakarRaghavan, and HinrichSchutze, "An Introduction to Information Retrieval," Cambridge UP, 2009. Chap. 6, pp. 109–133.
8. Jacob Perkins, "Python 3 text processing with NLTK 3 cookbook," Packet Publishing Ltd,2014.
9. Noah A. Smith, "Linguistic Structure Prediction. Synthesis Lectures on Human LanguageTechnologies," Morgan and Claypool, May 2011.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Understand POS tagging and CFG for English language	3	2	2	–	3	
CO2	Apply NLP algorithms - Latent Dirichlet Allocation and Latent Semantic Indexing	2	3	3	–	–	2
CO3	Understand and explore the models of NLP and NLP tools BERT,XLNet	1	3	3	1	3	2
CO4	Explore NLTK toolkit for implementing NLP applications	2	3	3	3	3	3

Evaluation Pattern - 70:30

Midterm Exam - 20%
Lab Assignments – 25%
Project – 25%
End Semester Exam - 30%

21AM720 Computer Vision 2-0-2-3

Preamble

Computer Vision is one of the fastest growing and most exciting AI (Artificial Intelligence) disciplines in today's academia and industry. This course is designed to open the doors for students who are interested in learning about the fundamental principles and important applications of computer vision. The course starts with the basic understanding of image formation and various image preprocessing techniques. It also deals with visual object detection and recognition algorithms. Object tracking and Motion segmentation from videos are also introduced as part of the course. The course also gives exposure to image reconstruction, camera calibration, stereo vision camera projection models etc which enable students to understand the concepts required for implementing modern AI applications which can perceive understand and reconstruct complex visual world like robots navigating space and performing duties, smart cars which can drive safe etc.

Course Objectives

- To introduce students to the state of the art algorithms in the area of image analysis and object recognition
- Give an exposure to video analysis techniques for object tracking and motion estimation
- To build good understanding on the computer vision concepts and techniques to be applied for robotic vision applications
- Enable students to apply the vision algorithms and develop applications in the domain of image analysis, robotic navigation

Course Outcomes

COs	Description
CO1	To build an understanding on detailed models of image formation
CO2	To expose the students to techniques of image analysis through image feature extraction and object recognition.
CO3	To introduce fundamental algorithms for video analysis such as object tracking, motion segmentation etc.
CO4	To impart sound knowledge in image registration, camera calibration, pose estimation, and stereo vision so as to build real-life projects in computer/robotic vision

Prerequisites

- Nil

Syllabus

Introduction to Image Processing-Basic mathematical concepts: Image enhancement: Grey level transforms, Spatial filtering. Extraction of special features: edge and corner detection. Morphological processing, Image transforms, Discrete Fourier Transform, Fast Fourier Transform. Frequency domain enhancement.
Image Segmentation Algorithms: contextual, non-contextual segmentation, texture segmentation. Feature Detectors and Descriptors, Feature Matching-Object Recognition, Face detection (Viola Jones), Face Recognition, Modern computer vision architectures based on deep convolutional neural networks, The Use of Motion in Segmentation Optical Flow & Tracking Algorithms, YOLO, DeepSORT: Deep Learning to Track Custom Objects in a Video, Action classification with convolutional neural networks, RNN, LSTM.

Image registration, 2D and 3D feature-based alignment, Pose estimation, Geometric intrinsic calibration, -Camera Models and Calibration: Camera Projection Models – orthographic, affine, perspective, projective models. Projective Geometry, transformation of 2-d and 3-d, Internal Parameters, Lens Distortion Models, Calibration Methods – linear, direct, indirect and multiplane methods. Geometry of Multiple views- Stereopsis, Camera and Epipolar Geometry, Fundamental matrix; Homography, Rectification, DLT, RANSAC, 3-D reconstruction framework; Auto-calibration., Introduction to SLAM (Simultaneous Localization and Mapping).

Text Book / References

1. Deep Learning (Adaptive Computation and Machine Learning series) Ian Goodfellow, YoshuaBengio, Aaron Courville, Francis Bach, January 2017, MIT Press
2. Introduction to Computer Vision and its Application, Richard Szelinski,2010
3. E. Trucco and A. Verri, Prentice Hall, 1998.Introductory techniques for 3D Computer Vision.
4. Marco Treiber, "An Introduction to Object Recognition Selected Algorithms for a Wide Varietyof Applications", Springer, 2010.
5. Forsyth and Ponce, "Computer Vision – A Modern Approach", Second Edition, Prentice Hall,2011.
6. R. C. Gonzalez, R. E. Woods, 'Digital Image Processing', 4th edition Addison-Wesley,2016

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	To build an understanding on detailed models of image formation	3	2	2	2	2	2
CO2	To expose the students to techniques of image analysis through image feature extraction and object recognition.	3	2	2	2	2	1
CO3	To introduce fundamental algorithms for video analysis such as object tracking, motion segmentation etc.	3	2	2	2	2	2
CO4	To impart sound knowledge in image registration, camera calibration, pose estimation, and stereo vision so as to build real-life projects in computer/robotic vision	3	2	2	2	2	2

Evaluation Pattern - 70:30

Midterm Exam - 20%
Lab Assignments – 25%
Project – 25%
End Semester Exam - 30%

21RM600 Research Methodology 2-0-0-2

Course Outcomes

COs	Description
CO1	Understanding the work-flow of research, research types and research models.
CO2	Applying various tools for identifying, retrieving and organizing the literature.
CO3	Understanding experimental and data-driven approaches in research.
CO4	Document writing using professional tools such as LaTeX, BibTeX, TikZ, PGF etc.

Syllabus

Research problem, Sources of research problem, Criteria Characteristics of a good research problem, Errors in selecting a research problem, Scope and objectives of research problem. Approaches of investigation of solutions for research problem, data collection, analysis, interpretation, Necessary instrumentations.

Effective literature studies approaches, Bibliography management. Plagiarism, Research ethics. Effective technical writing, how to write report, Paper. Technical writing tools. Developing a Research Proposal, Format of research proposal, a presentation and assessment by a review committee.

Nature of Intellectual Property: Patents, Designs, Trade and Copyright. Process of Patenting and Development: technological research, innovation, patenting, development. International Scenario: International cooperation on Intellectual Property. Procedure for grants of patents, Patenting under PCT. Patent Rights: Scope of Patent Rights. Licensing and transfer of technology. Patent information and databases. Geographical Indications.

New Developments in IPR: Administration of Patent System. New developments in IPR; IPR of Biological Systems, Computer Software etc. Traditional knowledge Case Studies, IPR and IITs.

Text Book / References

1. Stuart Melville and Wayne Goddard, Research methodology: an introduction for science &engineering students.
2. Wayne Goddard and Stuart Melville, Research Methodology: An Introduction.
3. Ranjit Kumar, 2nd Edition, Research Methodology: A Step by Step Guide for beginners.
4. Halbert, Resisting Intellectual Property, Taylor & Francis Ltd, 2007.
5. Mayall, Industrial Design, McGraw Hill, 1992.
6. Niebel, Product Design, McGraw Hill, 1974.
7. Asimov, Introduction to Design, Prentice Hall, 1962.
8. Robert P. Merges, Peter S. Menell, Mark A. Lemley, Intellectual Property in New TechnologicalAge, 2016.
9. T. Ramappa, Intellectual Property Rights Under WTO, S. Chand, 2008.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Understanding the work-flow of research, research types and research models.		2	1	2	3	3
CO2	Applying various tools for identifying, retrieving and organizing the literature.		2	3	1	2	2
CO3	Understanding experimental and data-driven approaches in research.	1	2	1	2	2	2
CO4	Document writing using professional tools such as LaTeX, BibTeX, TikZ, PGF etc.			2		1	3

Evaluation Pattern - 80:20

Internal – 80%
External - 20%

21AM698 Project with Seminar 2-0-0-2

Course Outcomes

COs	Description
CO1	To develop skills in doing project, technical presentation and report preparation.
CO2	To enable project identification and execution of preliminary works on final semester project.

Syllabus

This course is intended to for developing programming and presentation skills. Each student can select an area for project in consultation with the faculty. It involves programming, implementation, testing and performance analysis. in different application specific contexts. Students will be required to make an in-class presentation, project demonstration and a project report. The course will be evaluated by a panel of (at least) two faculty members.

References

1. Relevant literature for the computing problem.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	To develop skills in doing project, technical presentation and report preparation.	1	1	1	2	3	3
CO2	To enable project identification and execution of preliminary works on the second year project	2	3	3	2	2	2

Evaluation Pattern 80:20

Internal - 80%
External - 20%

21AM798 Dissertation Phase I 10

Course Objectives

The student is expected to carry out supervised research in this course. An intensive literature in the chosen area, should result in sound knowledge in the area and result in the identification of a suitable research problem, and its formulation and analysis. Study of relevant supplementary literature, mastering useful programming languages and tools for the problem, are also expected at this stage of the project. The student is expected to present three reports at different evaluation points during the semester, with clearly defined achievements and plans for further steps.

Course Outcomes

COs	Description
CO1	Demonstrate sound fundamentals in a chosen area of computing.
CO2	Identify and formulate a problem of research interest in the chosen area of computing.
CO3	Analyze the computing problem and propose solutions.
CO4	Effectively communicate the work at all stages of the project.

References

1. Relevant literature for the computing problem.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Demonstrate sound fundamentals in a chosen area of computing.	3	2	2	2	2	
CO2	Identify and formulate a problem of research interest in the chosen area of computing	2	3	3	3	2	3
CO3	Analyze the computing problem and propose solutions.	3	3	3	2	2	3
CO4	Effectively communicate the work at all stages of the project.	2	3	3	2	2	3

Evaluation Pattern - 80:20

Internal – 80%
External - 20%

21AM799 Dissertation Phase II 16

Course Objectives

The student is expected to demonstrate the core competency aimed by this course, i.e., the development of enhancements to the knowledge base in the area of interest in computing. The secondary competencies include the management of time bound projects involving research, analysis of problem complexities, design and development of effective solutions and communication of the project's progress, adhering to ethical practices at every stage. This stage of the project evaluates the state of maturity of these competencies. The student is expected to present two reports at intermediate stages, as well as prepare and defend a thesis on his research work.

Course Outcomes

COs	Description
CO1	Reflectively analyze proposed solutions to the identified computing problem.
CO2	Design and develop solutions to the problem and analyze results.
CO3	Prepare a thesis report and defend the thesis on the work done.
CO4	Augment the knowledge base in the chosen area of computing, adhering to ethical practices at every stage.

Prerequisites

- Dissertation Phase I

References

1. Relevant literature for the computing problem.

CO-PO Mapping

COs	Description	PO1	PO2	PO3	PO4	PO5	PO6
CO1	Reflectively analyze proposed solutions to the identified computing problem	3	3	3	3		
CO2	Design and develop solutions to the problem and analyze results.	3	3	3	3	2	3
CO3	Prepare a thesis report and defend the thesis on the work done.	3	3	3	2	2	3

CO4	Augment the knowledge base in the chosen area of computing, adhering to ethical practices at every stage.	3	3	3	2	3	3

Evaluation Pattern - 80:20

Internal – 80%
External - 20%

Evaluations patterns mentioned in the syllabus are recommendations from BOS. Final decision on the same can be done at the class committee.