PES UNIVERSITY

B. TECH IN COMPUTER SCIENCE AND ENGINEERING

V SEMESTER (2022-26 BATCH)

Sl. No.	Course Code	Course Title	Hours per week				Credits	Tools / Languages	Course Type
			L	T	P	S	C		
1	UE22CS351A	Database Management System	4	0	2	5	5	MySQL Workbench, Python, ERwin, Any other tool for ER modeling	CC-Lab Integrated
2	UE22CS352A	Machine Learning*	4	0	2	5	5	NetworkX for statistical features of graphs, Tensorflow, Keras, and Scikit Learn for traditional graph ML, and Pytorch Geometric for Graph Neural Networks.	CC-Lab Integrated
3	UE22CS341A	Software Engineering	4	0	0	4	4	GitHub, MS Project/GanttPro/Jira, Jenkins.	CC-Independent
4	UE22CS342AAX	Elective I	4	0	0	4	4		EC
5	UE22CS343ABX	Elective II	4	0	0	4	4		EC
6	UE22CS320A	Capstone Project - Phase 1	0	0	8	8	2		
Total			**20**	**0**	**12**	**30**	**24**		
Elective – I									

6	UE22CS342AA1	Advanced Algorithms%	4	0	0	4	4	C or C++ Programming Language	EC-Independent
7	UE22CS342AA2	Data Analytics&	4	0	0	4	4	R andPython.	EC-Independent
8	UE22CS342AA3	Internet of Things$	4	0	0	4	4	Python, Embedded-C, Cloud Platforms, Single Board Computerss	EC-Independent
9	UE22CS342AA4	Applied Cryptography	4	0	0	4	4	SEED labs, Python Programming Language	EC-Independent
10	UE22CS342AA5	Augmented and Virtual Reality!!!	4	0	0	4	4	C/ C++/ JAVA/ Python using OpenGL.	EC-Independent
11	UE22CS342AA6	Human Computer Interaction	4	0	0	4	4	Figma, Adobe Creative Cloud.	EC-Independent
Elective – II									
12	UE22CS343AB1	Enterprise Business Systems for Engineers	4	0	0	4	4	Oracle, Salesforce, Open ERP.	EC-Independent
13	UE22CS343AB2	Big Data $	4	0	0	4	4	Hadoop, HDFS, Spark, Streaming Spark, HIVE, HBase, Mllib.	EC-Independent
14	UE22CS343AB3	Graph Theory, and its Applications !	4	0	0	4	4	Python, Neo4j, NetworkX	EC-Independent
15	UE22CS343AB4	Bio-inspired Computing %	4	0	0	4	4	Matlab, Inspyredpythonic tool	EC-Independent
16	UE22CS343AB5	Advanced Computer	4	0	0	4	4	GNS3, Cisco Packet	EC-

		Networks[%%]						Tracer.	Independent
17	UE22CS343AB6	Computer Network Security[%%]	4	0	0	4	4	SEED Ubuntu VM, Wireshark, Snort, Netwox, Scapy.	EC-Independent
18	UE22CS343AB7	Robotics with ROS	4	0	0	4	4	C++, Python, Java, ROS, Rviz, Gazebo and Webots.	EC-Independent

Note: Desirable Knowledge

Core :[*]- UE22CS241A, UE22MA241B, UE22CS241B.

Elective I :[%]- UE22CS241B, [&]- UE22CS241A, [$] - UE22CS151A, UE22CS151B, UE22EC141A, UE22CS252B, [!!!]- UE22CS252A.

Elective II :[$]-UE22CS252A, UE22CS241B, [!]- UE22CS151B, UE22CS252A, [%]- UE22CS241B, [%%]- UE22CS252B.

ELECTIVES TO BE OPTED FOR SPECIALIZATION

Sl. No.	SPECIALIZATION	ELECTIVE – I	ELECTIVE – II
A	System and Core Computing(SCC)	UE22CS342AA1, UE22CS342AA5.	UE22CS343AB1, UE22CS343AB2, UE22CS343AB3.
B	Machine Intelligence and Data Science(MIDS)	UE22CS342AA2, UE22CS342AA3, UE22CS342AA5, UE22CS342AA6	UE22CS343AB2, UE22CS343AB3, UE22CS343AB4
C	Cyber Security & Connected Systems (CSCS)	UE22CS342AA3, UE22CS342AA4, UE22CS342AA5.	UE22CS343AB5, UE22CS343AB6. UE22CS343AB7

VI SEMESTER (2022-26 BATCH)

Sl. No	Course Code	Course Title	Hours per week				Credits C	Tools / Languages	Course Type
			L	T	P	S			
1	UE22CS351B	Cloud Computing@@	4	0	2	5	5	Amazon AWS (or equivalent), AWS Skill Builder, AWS Educate, Qwiklabs, Docker, Kubernetes, Jenkins, Zookeeper, Github, NoSQL database, Flask, Python, Go Lang.	CC
2	UE22CS352B	Object Oriented Analysis and Design	4	0	2	5	5	Star UML, Object Oriented Programming Language (Java/C++)	CC
3	UE22CS341B	Compiler Design!	4	0	0	4	4	Lex/flex and YACC/ Bison.	CC
4	UE22CS342BAX	Elective III	4	0	0	4	4		EC
5	UE22CS343BBX	Elective IV	4	0	0	4	4		EC
6	UE22CS320B	Capstone Project Phase-2	0	0	24	24	2		PW
Total			**20**	**0**	**26**	**46**	**24**		
Elective – III									
7	UE22CS342BA1	Supply Chain Management for Engineers	4	0	0	4	4	SCM applications and tools, ML tools, Case studies, Web resources.	EC-Independent
8	UE22CS342BA2	Algorithms for	4	0	0	4	4	Scikit, Tensorflow,	EC-

		Information Retrieval						Solr, Lucene Search Engines/ Python Programming Languages	Independent
9	UE22CS342BA3	Image Processing and Computer Vision **	4	0	0	4	4	MatLab, Python Programming Languages	EC-Independent
10	UE22CS342BA4	Natural Language Processing ##	4	0	0	4	4	Tensorflow, Scikit Learn, Python 3.x. CoreNLP, Natural Language Toolkit (NLTK), TextBlob,Gensim, SpaCy,PyTorch-NLP , OpenNLP, Hugging Face, OpenAI API.	EC-Independent
11	UE22CS342BA5	BlockChain!	4	0	0	4	4	Solidity, Remix, Ganache, Metamask.	EC-Independent
12	UE22CS342BA6	Digital Forensics and Incident Response	4	0	0	4	4	Open source tools on Forensics.	EC-Independent
13	UE22CS342BA7	Digital Twin and eXtendedReality!	4	0	0	4	4	C/ C++/ JAVA/ Python using OpenGL.	EC-Independent
14	UE22CS342BA8	Topics in Wireless Networks and 5G	4	0	0	4	4	Wireshark, Claynet, Cisco Packet Tracker.	EC-Independent
15	UE22CS342BA9	Generative AI and its Applications	4	0	0	4	4	Python,,HuggingFace , LM Studio, Kaggle	EC-Independent
Elective – IV									

16	UE22CS343BB1	Heterogeneous Parallelism !!!	4	0	0	4	4	pthread, OpenMP CUDA, openCL.	EC-Independent
17	UE22CS343BB2	Topics in Deep Learning ##	4	0	0	4	4	Pytorch.	EC-Independent
18	UE22CS343BB3	Database Technologies***	4	0	0	4	4	MySQL, postgres, Oracle, Apache Spark, Apache Kafka, Amazon Kinesis.	EC-Independent
19	UE22CS343BB4	Machine Learning on Graphs %%%	4	0	0	4	4	NetworkX for statistical features of graphs,Tensor flow Keras and Scikit Learn for traditional graph ML, and PytorchGeometric for Graph Neural Networks.	EC-Independent
20	UE22CS343BB6	Information Security	4	0	0	4	4	SSEED Labs VM, Scapy, Burp Suite, Metasploit, Nmap, etc.	EC-Independent
21	UE22CS343BB7	Mobile and Autonomous Robotics	4	0	0	4	4	C, C++, Python, ROS	EC-Independent
22	UE22CS343BB8	Security for Internet of Things.	4	0	0	4	4	•Wireshark, Yersinia, VoIP Hopper, Bettercap,aircrack-ng	EC-Independent
23	UE22CS343BB9	Applied ML in IoT with TinyML	4	0	0	4	4	•Arduino Nano 33 BLE sense •ArduinoIDE GoogleColab	EC-Independent

Note: Desirable Knowledge

Core: [!]UE22CS252A, UE22CS243A [@@]UE22CS241B, UE22CS252B.

Elective III:[#] - UE22CS151A, UE22CS151B, UE22CS252A, UE22CS252B, UE22CS241B, [??] - UE22CS241B, UE22MA241B, UE22CS352A [**]- UE22CS241B, [##] - UE22CS352A, [!] - UE22CS252A.

Elective IV: !!!-UE22CS151B, UE22CS251B, ##-UE22CS352A, *** - UE22CS351A, %%% - UE22CS343AB3, UE22CS352A, %% - UE22CS252B.

ELECTIVES TO BE OPTED FOR SPECIALIZATION

Sl. No.	SPECIALIZATION	ELECTIVE – III	ELECTIVE – IV
A	System and Core Computing (SCC)	UE22CS342BA1, UE22CS342BA2, UE22CS342BA7	UE22CS343BB1, UE22CS343BB3.
B	Machine Intelligence and Data Science (MIDS)	UE22CS342BA2, UE22CS342BA3, UE22CS342BA4, UE22CS342BA7. UE22CS342BA9	UE22CS343BB2, UE22CS343BB4, UE22CS343BB7 UE22CS343BB9
C	Cyber Security & Connected Systems (CSCS)	UE22CS342BA5, UE22CS342BA6, UE22CS342BA7, UE22CS342BA8	UE22CS343BB6, UE22CS343BB7, UE22CS343BB8.

UE22CS351B: Cloud Computing (4-0-2-5-5)

The cloud computing course introduces not only the various technologies that go into building a cloud native application, but also how cloud systems are designed. The student is introduced to various tools and design techniques/tradeoffs. It also gives a flavour for the business relevance/ethics of using cloud computing. This course requires the student to have a desirable knowledge of Computer Networks and Operating System.

Course Objectives:

- Introduce the rationale behind the cloud computing revolution and cloud native application architecture
- Explore the concepts of Virtualization and Containerization.
- Explore the concepts of Distributed storage and its various techniques
- Design distributed systems for scalability and expose the student to various tradeoffs in designing cloud architectures.

Course Outcomes:

At the end of this course, the student will be able to:

- Comprehend the technical and business rationale behind cloud computing. Decide the model of cloud computing to use for solving a particular problem.
- Implement Microservice architecture through Containers and Orchestration tools. Analyze virtual machines and containers.
- Experiment with cloud storage Models such as object stores, key value stores.
- Apply the critical constraints such as Performance, scalability and security to the designed distributed system.

Desirable Knowledge: UE22CS252B- Computer Networks, UE22CS241B- Operating Systems.

Course Content:

Unit 1 : Cloud Programming Models

Parallel computing, Grid computing, Introduction to Cloud Programming Models and service Models, Introduction to technology challenges with Distributed & Cloud computing, Business Drivers - deployment models, Cloud architecture and IaaS programming model, Web Services and REST, PaaS Programming Model, Communication using Message queues- Pub Sub model, SaaS Programming model – Microservices and differences with the traditional monolithic model; challenges of migrating monolithic applications**.**

14 Hours

Unit 2 : Virtualization

Hypervisor - Types, Para virtualization and Transparent virtualization, Software - Trap and Emulate virtualization, Software - Binary translation, Goldberg Popek principles for Virtualization, Hardware - AMDv/Intel, Memory - Shadow page tables, Memory - Nested page tables, IO, VM Migration, Lightweight Virtualization - Containers and Namespaces, Deployment of cloud native applications through Docker – Unionfs, DevOps, Orchestration and Kubernetes.

14 Hours

Unit 3 : Distributed Storage

Types of Cloud storage - Block, Object stores, Replication, lag, multileader replication, Leaderless replication, Partitioning - key-value data, Consistent hashing, Partitioning - rebalancing partitions, Request routing, Consistency Models, CAP Theorem, Transactions, Two-phase commit.

14 Hours

Unit 4 : Cloud Controller, Performance, Scalability and Security

Master-slave v/s p2p models, Resource allocation, Scheduling algorithms, Cluster coordination – consensus, Fault Tolerance - faults and partial failures, Failure detection - checkpointing and application recovery, Unreliable communication, Cluster coordination - leader election, distributed locking, Case Study: Zookeeper/Raft - distributed consensus infrastructure.

Scaling computation - reverse proxies, Scaling computation - hybrid cloud and cloud bursting, Multitenancy, Multitenant databases, Cloud security requirements - physical/virtual security, security design patterns, Authentication in the cloud: Keystone/IAM, Cloud Threats – DoS, Economic Denial of Sustainability**.**

14 Hours

Tools/Languages: Amazon AWS (or equivalent), AWS Skill Builder, AWS Educate, Qwiklabs, Docker, Kubernetes, Jenkins, Zookeeper, Github, NoSQL database, Flask, Python, GoLang.

Lab/ Hands-on:

1: Migrating a monolithic e-commerce application to a microservices architecture

2: Building a Task Management Application with Raft Consensus Algorithm and MySQL.

3: Microservices communication using RabbitMQ

4: Building a Distributed Key-Value Store with etcd

5: Back Up service using docker and Kubernetes

6: Building an E-commerce Microservices Application on Cloud using Docker, Kubernetes, Jenkins, and Git

Text Book(s):

1: "Distributed and Cloud Computing", Kai Hwang, Jack Dongarra, Geoffrey Fox.ISBN: 978-0-12-385880-1, Morgan Kaufmann, 2012.

2:"Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems", Martin Kleppmann. O'Reilly, 2017.

Reference Book(s):

1: "Docker in Action", Jeff Nickoloff, Manning Publications, 2016.

2: "Cloud Native DevOps with Kubernetes", John Arundel and Justin Domingus, OReilly, 2019.

3: "Moving to the clouds: Developing Apps in the new world of cloud computing", Dinkar Sitaram and Geetha Manjunath. Syngress, 2011.

UE22CS352B: Object Oriented Analysis and Design (4-0-2-5-5)

In this course students will learn to perform Analysis on a given domain and come up with an Object-Oriented Design (OOD). Various techniques will be discussed and practiced which are commonly used in analysis and design phases in the software industry. Unified Modelling Language (UML) will be used as a tool to demonstrate the analysis and design ideas and object-oriented programming language such as Java/C++ would be used to implement the design. The theory is supplemented with implementations which are demonstrated/ practiced in class which provides the hands-on experiences of implementing the patterns.

Course Objectives:

- Familiarize students with static and dynamic models of object-oriented analysis and modelling using the unified modelling language (UML)
- Introduce students to object oriented programming concepts using object oriented programming language
- Make students appreciate the importance of system architecture design in software development
- Introduce the students to understand the importance of GRASP and SOLID design principles along with Design patterns

Course Outcomes:

At the end of this course, the student will be able to:

- Construct static models, use cases and class models, followed by analysing the dynamics of the system using activity, sequence, state and process models. Depict the architecture of a software system by using component and deployment models
- Use the concepts of classes and objects of object-oriented programming to model a complex system
- Use GRASP and SOLID principles in the design of software application and apply Creational software design patterns for variety of application scenarios
- Apply Structural and Behavioural software design patterns for variety of application scenarios. Understand Anti-pattern.

Course Content:

Unit 1: Object Oriented Design and UML Diagrams

Requirements, Modelling and Analysis: Introduction to UML, Use Case Modelling: Use Cases Diagrams. Domain models, Class Modelling: UML Class Diagrams, OO relationships, CRC Diagrams, Component model, Deployment model, Activity Modelling: UML Activity Diagrams and Modelling, Guidelines. BehaviourModelling: Sequence Diagram, UML State Machine Diagrams and Models, Advanced State Models.

16 Hours

Unit 2: Object Orientated Programming and Architecture design

Overview of Object Orientated Programming using Java/C++ –Classes, Inheritance, Overloading, Overriding, Abstract classes, Interfaces. OO Development process, System Design and Frameworks, Architectural patterns: MVC, Layered Pattern, Client-Server Pattern, Event-Driven Pattern, Microkernel Pattern, Microservices Pattern.

12 Hours

Unit 3: Design Principles & Patterns

GRASP and its application to Object Design, Creator, Information Expert, Low Coupling, Controller, High Cohesion, Polymorphism, Pure Fabrication, Indirection and Protected Variations SOLID: Single Responsibility, Open-Closed, Liskov Substitution, Interface Segregation, Dependency Inversion OO Design Principles and Sample Implementation of Patterns. Introduction to Design Patterns, selection and usage of a design pattern, Creational Design Patterns Theory and Implementation: Singleton, Factory, Builder and Prototype

14 Hours

Unit 4: Design Patterns

Structural Patterns – Adapter, Façade, Proxy and Flyweight, Behavioural Patterns – Chain of Responsibilities, Command, Interpreter, Iterator. Anti-patterns – Introduction and classification, Project Management, Architecture and Development anti-patterns (1 anti-patterns of each type)

14 Hours

Tools / Languages: Star UML, Object Oriented Programming Language (Java/C++)

Text Book(s):

1: “Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development”, by Craig Larman, 3rd Edition, Pearson 2015.

2: “Software Architecture Patterns” by Mark Richards, 2nd Edition, OReilly 2022.

Reference Book(s):

1: “Object-Oriented Modelling and Design with UML”, Michael R Blaha and James R Rumbaugh, 2nd Edition, Pearson 2007.

2: “Design Patterns: Elements of Reusable Object-Oriented Software” by Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides, 1st Edition, Pearson 2015.

3: “Fundamentals of Software Architecture: An Engineering Approach” by Mark Richards & Neal Ford, First Edition, 2020.

Hands-on/Assignment/Laboratory/Project

- Lab assignment on Use case diagram.
- Lab assignment on Class diagram.
- Lab assignment on Activity and State diagrams.
- Lab assignment on Object Oriented Programming Concepts.
- Lab assignment on Architectural Patterns
- Self Learning Assignment on Serialization and Multithreading
- Hands-on Assignment on MVC Framework.
- Assignment on Design Patterns.
- Mini Project using any Architectural pattern incorporating all learning of the course.
- Experiential learning – Java/C++ Certification

UE22CS341B: Compiler Design (4-0-0-4-4)

Language design and implementation is an active topic in programming, and will likely always be. How we program and the tools we use, changes constantly. We try new ideas and come up with better or alternative approaches frequently. Any language that doesn't continue to adapt will fall into disuse, and any tool chain that remains stagnant will be forgotten. Hence knowledge of compilers in order to tweak these changes in the language design is a must for a Computer Science Engineer. This course requires the student to have a desirable knowledge of Data Structures and its Applications and Automata Formal Languages and Logic.

Course Objectives:

- Introduce the major concept areas of language translation and compiler design.
- Develop a greater understanding of the issues involved in programming language design and implementation.
- Provide practical programming skills necessary for constructing a compiler. Develop an awareness of the function and complexity of modern compilers.
- Provide an understanding on the importance and techniques of optimizing a code from compiler's perspective.

Course Outcomes:

At the end of this course, the student will be able to:

- Use the knowledge of patterns, tokens and regex for solving the problems in the field of data mining
- Analyze and design the semantic behaviour of a compiler. Choose the appropriate compiler internal representation for different kinds of compiler tasks.
- Translate a source-level language into a low-level compiler internal representation.
- Optimize the performance of a program in terms of speed and space using new code optimization techniques.

Desirable Knowledge: UE22CS252A- Data Structures and its Application, UE22CS243A- Automata Formal Languages & Logic.

Course Content:

Unit 1: Compilers: Introduction, Lexical Analysis, Top-down Parsers

The Language Processing System, The Phases of a Compiler, The Grouping of Phases into passes.

Lexical Analysis: The Role of the Lexical Analyzer, Input Buffering, Specification of Tokens, Recognition of Tokens, Design of a Lexical Analyzer Generator. The role of the parser, Syntax Error Handling, Error-Recovery Strategies. Top-down parsing: Recursive Descent Parser (RDP) with Backtracking, LL(1) Parser.

14 Hours

Unit 2: Syntax Analysis: Bottom-up Parsers, Syntax-Directed Translation

Bottom-up parsing: Shift-Reduce Parsing, LR (0), SLR, viable prefixes, CLR, LALR. Syntax-directed definitions, Evaluation orders for SDD's: S-attributed SDD, L-attributed SDD.

14 Hours

Unit 3: Implementation of Syntax-Directed Translation Schemes and Intermediate Code Generation

Applications of Syntax-Directed Translation - SDD for Syntax Trees, Expressions, Basic Types and flow control statements. Syntax-directed Translation Schemes – Parser Stack Implementation of Postfix SDT's, SDT's with actions inside Productions, SDT's for L-Attributed Definitions. Implementing L-Attributed SDD's: Bottom-Up Parsing. Variants of Syntax Trees – Directed Acyclic Graphs for Expressions, Three-Address Code – Addresses and Instructions, Quadruples, Triples, Indirect Triples, SSA Form, Control Flow Graph.

14 Hours

Unit 4: Machine Independent Code Optimization, Code Generation and Run Time Environment

Machine Independent Optimization: Different Optimizations, Optimization of Basic Blocks. Data Flow Analysis: Live-variable analysis, Next-use algorithm. Storage Organization, Different Allocation Strategies, Stack Allocation of space, Access to Non local Data on the stack. Code Generation: Issues in the design of a code generator, the target language, addresses in the target code, static allocation, stack allocation, run-time addresses for names. A Simple Code generator - The Code generation algorithm.

14 Hours

Tools/Languages: Lex/flex and YACC/Bison.

Text Book(s):

1: "Compilers–Principles, Techniques and Tools", Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffery D. Ullman, 2nd Edition, Pearson Education, 2009.

Reference Book(s):

1: "Modern Compiler Design", Dick Grune, Kees van Reeuwijk, Henri E. Bal,Ceriel J.H. Jacobs, KoenLangendoen, 2nd Edition, 2012.

2: AW Appel, J Palsberg, Modern Compiler Implementation in Java, Cambridge University Press, 2002.

3: AW Appel, M Ginsburg, Modern Compiler Implementation in C, Cambridge University Press, 2004.

UE22CS342BA1: Supply Chain Management for Engineers (4-0-0-4-4)

This course explores supply chain management (SCM) processes. It aims for students to develop the skills to understand, design and evaluate techniques in the realm of supply chain management. It covers fundamentals relevant to SCM, current SCM challenges and trends in the industry including planning and prediction, techniques and tools using AI technologies.

Course Objectives:

- Understand the fundamentals of supply chain management.
- Learn the various SCM stages and strategies of planning, sourcing, inventory, distribution and logistics.
- Explore case studies of SCM applications to understand various industrial scenarios.
- Explore current trends and AI techniques applied to solve real-world SCM challenges.

Course Outcomes:

At the end of the course, the student will be able to:

- Identify various stages of SCM including Logistics.
- Evaluate KPIs and select suitable strategies for optimized supply chain performance.
- Evaluate and select the strategies for planning and managing Inventories and Logistics.
- Apply ML techniques to predict trends, optimize inventory, and improve logistics.

Desirable Knowledge: UE22CS341AB1 - Enterprise Business Systems

Course Content:

Unit 1: Introduction to SCM

Overview of supply chain management, Building a strategic framework to analyze Supply Chains. Understand supply chain scope, strategic fit, its drivers and performance metrics. Managing Material and Information Flows. Intro to supply chain networks and their design. Various strategies of planning, sourcing, inventory, distribution and logistics related to SCM – Opportunities, Competitive advantage, Real-time Supply Chains. Value Stream Mapping.

14 Hours

Unit 2: Procurement and Supply Chain Planning (Demand & Supply)

Procurement processes, Supplier selection and performance assessments. Intro to Demand Forecasting models, predictive analytics for supply chain planning. Techniques for aggregate planning of demand and supply in supply chains, managing demand variability and accuracy.

Industry trends in procurement – E-procurement, Case studies.

14 Hours

Unit 3: Planning and Managing Inventory and Logistics Management in SCM

Introduction to planning and management of Inventory in an enterprise.Techniques for warehouse management and logistics management for distribution.Logistics stakeholders and modern logistics concepts, Outsourcing logistics, Logistics KPIs.Intro to Reverse Logistics, and Lean Logistics.Enhancing transportation and logistics with real-time tracking and monitoring using IoT and AI-driven route planning and last-mile delivery. Case studies in inventory and logistics management. **14 Hours**

Unit 4: Drivers, Analytics and Current Trends in Supply Chain Management

Understanding and managing cross functional drivers in supply chain.Decision making in supply chains - sourcing, pricing, risk management and compliance. Basics of AI, Blockchain, IoT concepts applied to SCM. Current trends in SCM: Green SCM, Lean and Global SCM, SCM Analytics. Exploration of AI techniques, tools and technologies for SCM.Awareness of current challenges and research areas of SCM in the industry.

14 Hours

Tools / Languages: SCM applications and tools, ML tools, Case studies, Web resources.

Textbook:

1. Supply Chain Management: Strategy, Planning and Operation, Sunil Chopra, Pearson, 7th Revised Edition, 2024.

Reference(s):

1. Essentials of Supply Chain Management, Michael H Hugos, John Wiley & Sons, 3rd Edition, 2018.
2. Principles of Supply Chain Management: A Balanced Approach, Joel D Wisner, Cengage, 5th Ed, 2019.
3. Strategic Supply Chain Management: The Five Core Disciplines for Top Performances, Cohen Shoshanah & Joseph Roussel, Mc-Graw Hill, 2nd Ed, 2013.
4. Logistics and Supply Chain Management, Martin Christopher, Pearson, 2022.
5. Supply Chain Logistics Management, Donald Bowersox, et al, Mc-Graw Hill Publishing, 5th Edition, 2024.
6. Supply Chain Management: A Logistic Approach, John J Coyle, Cengage, 10th Ed, 2019.
7. A Roadmap to Green Supply Chains: Using Supply Chain Archaeology and Big Data Analytics, Kevin L Lyons, Industrial Press, 2015.

8. Lean Supply Chain and Logistics Management, Paul Myerson, Mc-Graw Hill, 2012.
9. Supply Chain Analytics, T. A. S. Vijayaraghavan, Wiley, 2021.
10. Logistics & Supply Chain Management - Course (nptel.ac.in).
11. Blockchain, IoT, and AI Technologies for Supply Chain Management, CRC Press, 2023.
12. Web resources.

UE22CS342BA2 - Algorithms for Information Retrieval (4-0-0-4-4)

This course covers the basic and advanced algorithms and techniques for Information retrieval and web applications. This course focuses on Index building, document ranking, use of machine learning in Information retrieval , recommendation algorithms and design of intelligent web applications. This course requires the student to have a desirable knowledge of Design and Analysis of Algorithms.

Course Objectives

- Understand the architecture, models, and algorithms used in Information Retrieval.
- Understand the basic principles and implementation of Indexing and Search.
- Understand the use of machine learning in Information Retrieval and Web Applications.
- Understand multimedia and multimodal information retrieval.

Course Outcomes:

At the end of this course, the student will be able to:

1. Implement an efficient index for document collection.
2. Perform searches on a document collection, rank, and evaluate results.
3. Apply suitable techniques for multimodal information retrieval
4. Apply Machine Learning techniques in Information Retrieval Systems a Web Applications.

Course Contents:

Unit-1: Basics of Information Retrieval

Introduction to Information Retrieval, Background, Architecture and Strategies of Information Retrieval (IR) Systems,IR Models, Boolean and Extended Boolean Models, Dictionary, Vocabulary, Positional Postings, Phrase Queries and Tolerant Retrieval, Indexing and Vector Space Model, Evaluation of IR Algorithms for Indexing and Index Compression, Vector Space Model for Scoring, tf-idf and Variants.

14 Hours

Unit-2: Ranking and Web Search Basics

Efficient Scoring and Ranking, Parametric and Zone Indexes, Tiered Indexes, Query Term Proximity, Query Parser, Aggregating Scores, Performance Measurement, Web Applications and Search Algorithms, Relevance Feedback, Query Expansion, Other IR Models, Web Search Basics

14 Hours

Unit-3: Link Analysis, Multimodal Information Retrieval

Economic Model of Web Search, Improving Search Results, Link Analysis, The Page Rank Algorithm, Other Search Algorithms, Scalability Issues in Search. Search User Experience, Web Crawling and Indices, Link Analysis, Building a Complete Search System, Lucene as a Search Engine, and Other search engines like Solr, Everything, and Google. Multimodal Information Retrieval content-based visual information retrieval: metadata searching, Query by Example, Semantic Retrieval, Machine Learning Approaches, content comparison using image distance measures. Multimedia and Multimodal Information Retrieval: Introduction, Requirements, Applications, challenges, Architecture, Metadata, Techniques for Content Processing.

14 Hours

Unit-4: Question Answering, neural models for IR

Question Answering, neural models for IR Question Answering, Neural models for IR QA as an Information Retrieval task, Factoid QA models, Entity Linking models, Knowledge-based QA, and Pretrained models for QA.Recent and trending research papers in QA.

14 Hours

Textbook(s):

1: “Introduction to Information Retrieval”, Christopher D. Manning, Prabhakar Raghavan, Hinrich Schutze, ISBN: 9781107666399, Cambridge University Press, 2009.
2. “Search Computing”, Challenges and Directions: chapter 8: Multimedia and Multimodal Information Retrieval.
3. “Speech and Language Processing”, Third Edition, Daniel Jurafsky, James H. Martin, Chapter 14: Question Answering and Information Retrieval, 2023.

Reference Book(s):

1: “Algorithms of the Intelligent Web”, HaralambosMarmanis, Dmitry Babenko, Manning Publishers, 2011.

UE22CS342BA3 :Image Processing and Computer Vision (4-0-0-4-4)

igital Image Processing deals with processing images that are digital in nature. Improving the quality of images for human perception and understanding, extracting useful information for decision making and efficient storage are some of the driving factors behind image processing techniques/algorithms. The course on Digital Image Processing introduces the learner to various image processing techniques, algorithms and their applications.

Course Objectives:

- Understand the principles underlying imaging and assess the quality of an image based on its visual content, histogram and statistical measures
- Gain an insight to image transforms and enhancement techniques in the frequency domains
- Be able to apply morphological operations and segmentation techniques on monochrome and color images
- Learn different feature extraction techniques for classification

Course Outcomes:

At the end of this course, the student will be able to:

- Assess the quality of an image and identify the appropriate processing technique that needs to be applied in the spatial domain
- Apply image enhancement techniques frequency (Fourier) domain and interpret images in the transform domain
- Design and evaluate methodologies for segmentation of region of interest from binary, gray scale or color images
- Use appropriate feature extraction techniques for tasks such as classification and be able to work with data-driven approaches (deep learning architectures) for computer vision applications

Desirable Knowledge: UE22CS241B – Design and Analysis of Algorithms.

Course Content:

Unit 1: Introduction and Enhancement in the Spatial Domain

What is Digital Image Processing, examples of fields that use DIP, Fundamental Steps in Digital Image Processing, elements of visual Perception, Basic Concepts in Sampling and Quantization, Representing Digital Images, Spatial and Gray-level Resolution, Zooming and Shrinking Digital Images, Some Basic Relationships Between Pixels, Linear and

Nonlinear Operations. Image Enhancement in the Spatial Domain: Some Basic Gray-level Transformations, Histogram Processing, Enhancement Using Arithmetic/Logic Operations, Basics of Spatial Filtering, Smoothing Spatial Filters, Sharpening Spatial Filters, and Combining Spatial Enhancement Methods.

14 Hours

Unit 2: Image Enhancement in the Frequency Domain, Transforms and Compression

Image Enhancement in the Frequency Domain: Introduction to the Fourier Transform and the Frequency Domain, Smoothing Frequency-Domain Filters, Sharpening Frequency Domain Filters, Homomorphic filtering. Image transforms (such as, DCT, Harr, DWT) for applications. Image Compression: Fundamentals - Image Compression Models, Some encoding techniques (including, block processing and transform-based techniques).

14 Hours

Unit 3: Morphological Image Processing, Preliminaries of segmentation and Color Image Processing

Dilation and Erosion, Opening and Closing, the Hit-or-Miss Transformation, Some Basic Morphological Algorithms. Image Segmentation: Detection of Discontinuities, Edge Linking and Boundary Detection, Thresholding, Region-Based Segmentation. Color Models, Pseudocolor Image Processing, Basics of Full-Color Image Processing, Color Transformations, Smoothing and Sharpening, Color Segmentation, Noise in Color Images.

14 Hours

Unit 4: Feature extraction, pattern classification

Boundary Preprocessing, Boundary Feature Descriptors, Region Feature Descriptors, Principal Component Analysis (use of Eigen images), Whole image features, Scale Image Feature Transform, Image pattern classification using minimum distance classifier and prototype matching, deep convolutional neural networks and some of its recent variations for computer vision applications and state-of-art techniques and applications in computer vision.

14 Hours

Tools / Languages: Matlab, Python Programming Language.

Text Book(s):

1: “Digital Image Processing”, Rafael C Gonzalez and Richard E. Woods, Prentice Hall, 4th Edition, 2018.

Reference Book(s):

1: “Digital Image Processing and Analysis”, Scott E. Umbaugh, CRC Press, 2014.

2: “Digital Image Processing”, S. Jayaraman, S. Esakkirajan, T. Veerakumar, McGraw Hill Ed. (India) Pvt. Ltd., 2013. 3: “Digital Signal and Image Processing”, John Wiley, 2003.

4: “Computer Vision A Modern Approach”, D. A Forsyth and J. Ponce, Pearson Education, 2003

5:“Computer Vision: Algorithms and Applications”, Richard Szeliski, Springer, 2ndEdition, 2022.

UE22CS342BA4: Natural Language Processing (4-0-0-4-4)

The goal of this course is to focus on processing of text data as found in natural language usage. The key problem discussed in this course is that of understanding the meaning of text by various types of learning models including the recent approaches using deep learning and the significance of the NLP pipeline in that meaning disambiguation process. The course also discusses disambiguation of syntax as a step of meaning disambiguation process. This course requires the student to have a desirable knowledge of Machine Intelligence.

Course Objectives:

- Learn the phases of NLP and the building of Language Models.
- Learn how lexical and distributional semantics can be used for semantic disambiguation in NLP.
- Focus and learn various learning models related to sequence labelling and syntactic disambiguation in NLP.
- Introduce the deep learning techniques and its applications in Natural Language Processing

Course Outcomes:

At the end of this course, the student will be able to:

- Have a very clear understanding of the phases of NLP and the building of Language Models.
- Analyze and apply comfortably appropriate branch of semantics depending on the problem being solved.
- Understand various sequence labelling approaches, its applications and syntactic ambiguity removal in NLP.
- Design and implement neural language model, NLP applications using neural techniques and utilize various transfer learning approaches in NLP.

Desirable Knowledge: UE22CS352A-Machine Learning

Course Content:

Unit 1: Introduction to NLP

Introduction, Types of ambiguity in natural language processing, Text normalization, Morphological parsing of words – Porter stemmer, Lemmatization and Stemming, Sentence segmentation. Noisy Channel model: Real world spelling error, Minimum edit distance algorithm, Concept of noisy Channel Model. Language Model: N-grams, ngram language model, smoothing, discounting and back-off, Kneser-Ney smoothing, interpolation, perplexity as an evaluation measure.

14 Hours

Unit 2: Lexical and Vector Semantics

Word senses and relations between word senses, WordNet: A Database of Lexical Relations; Word sense disambiguation, Semantic relatedness, Lexicons for sentiment and affect extraction: available sentiment and emotion lexicons. Vector Semantics and Embeddings: Words and vectors, TF IDF, Pointwise Mutual Information, Measuring similarity, Using syntax to define a word's context, Evaluating vector models, Dense vectors via SVD Distributional Hypothesis, Neural Embedding: skip gram and CBOW Pre-trained word representations: Word2Vec, Improving Word2vec, Limitation of distributional methods.

Self-Learning component: FastText, Glove.

14 Hours

Unit 3: Handling sequences of text and Parsing - Disambiguating Structure

Sequence labelling: Hidden Markov Model, POS Tagging example, Discriminative Sequence labelling with features-Conditional Random Field. Other sequence labelling applications – Named Entities and Named Entity Tagging, POS Tagging using discriminative models i.e. Maximum Entropy Markov Model (MEMM)

Self-Learning component: Sequence labelling using RNNs and LSTMs

Constituency parsing: Ambiguity presented by parse trees, CKY parsing, CCG Parsing, Partial parsing – chunking. Statistical Parsing : Probabilistic Context Free Grammar, Probabilistic CKY parsing of PCFG, Problems with PCFG, Probabilistic Lexicalized CFG Introduction to dependency parsing: Dependency relations, Dependency Formalisms, Dependency Tree Banks. Evaluating parsers.

Self-Learning component: Span-based Neural Constituency Parsing

14 Hours

Unit 4: Coreference resolution, Transformers and Pretrained Language Models

Coreference resolution: Forms of referring expression, algorithms for coreference resolution – mention pair and mention ranking model, mention detection, classifiers using hand-built features.

Self-Attention Networks: Transformers, Transformers as Language Models, Sampling, Pretraining Large Language models. (Top notch/trending LLM model to be discussed during the course delivery), LLM's Vs SLM's.

Fine-Tuning and Masked Language Models: Bidirectional Transformer Encoders (BERT), Training Bidirectional Encoders, RoBERTa. In Context Learning, Instruction Fine Tuning, Prompt Engineering.

14 Hours

Tools / Languages: Tensorflow, Scikit Learn, Python 3.x. CoreNLP, Natural Language Toolkit (NLTK), TextBlob,Gensim, SpaCy,PyTorch-NLP , OpenNLP, Hugging Face, OpenAI API.

Text Book:

1: "Speech and Natural Language Processing", Daniel Jurafsky and James H. Martin, 3rd edition online, Feb 3 2024. The more up to date 3rd edition draft is available at http://web.stanford.edu/~jurafsky/slp3/. (text book revision is continuous process according to author)

Reference Book:

1: "Introduction to Natural Language Processing", Jacob Eisenstein, MIT Press, Adaptive computation and Machine Learning series, 18th October, 2019. The open source softcopy is available at githubhttps://github.com/jacobeisenstein/gt-nlpclass/blob/master/notes/eisenstein-nlp-notes.pdf.

UE22CS342BA5 : Blockchain (4-0-0-4-4)

Blockchain having wide impact and potential growth for change around the world. It is changing how business is executed. It is important to understand why Blockchain is different and how it works in comparison with technologies of the past. This course requires the student to have a desirable knowledge of Data Structures and its Applications.

Course Objectives:

- Learn a conceptual view of Blockchain for the new applications that they enable.
- Apply the Blockchain for various applications to provide a secure way of data access using cryptographic functions.
- Learn various consensus mechanisms to implement for various real time applications.
- Familiarize with the Blockchain deployment tools.

Course Outcomes:

At the end of this course, the student will be able to,

- Analyze how the traditional databases can be replaced with Blockchain for the real time applications.
- Integrate various cryptographic algorithms in to Blockchain.
- Apply various consensus mechanisms to the real world Blockchain applications.
- Evaluate the setting where a Blockchain based structure may be applied, its potential and its limitation.

Desirable Knowledge: UE22CS252A-Data Structures and its Applications.

Course Contents:

Unit 1: Introduction and Data flow and Integrity

Key Blockchain Concepts, Nodes, Cryptocurrency, tokens, Public Ledger, Peer to peer Network, Types of Blockchain, Permissioned Blockchain model, Permission-less Blockchain model, Demonstration of Blockchain Construction steps, Demonstration of ether scan and Goerli/Sepoliaetherscan. Cryptography- Need, history, features, Private and public keys, Types of cryptography, Digital signatures, Hash functions, SHA-256, Patterns of hashing, Hash Pointer, Markle tree, Ledgers, Transactions and trade, the public witness, Computers that witness.

14 Hours

Unit 2 : The structure of the network: consensus algorithm

Case study: Bitcoin Blockchain Network, Creation of metamask wallets and performing transaction.

Introduction to distributed consensus: What, why, Challenges, Proof of Work, Proof of Stake, Delegated Proof of Stake, Proof of Authority, Proof of Elapsed Time, Proof of Scope, Proof of Space, Proof of Burn, RAFT, PAXOS, Byzantine Fault Tolerance System, PBFT.

14 Hours

Unit 3: Second generation applications of Blockchain technology

Smart contracts: origins and how they function, Creating and deploying smart contracts, Decentralized applications, Dapps construction, Decentralized Autonomous Organizations (DAOs)- Need, Principal agent dilemma, components, The DAO Story, Legality of DAPPs and DAOs.

Solidity- Variable, Functions, modifiers, view, pure, fallback, overloading, in-built mathematical and cryptography functions, Withdrawal pattern, Restricted Access

14 Hours

Unit 4: Blockchain Security and use cases

Hyperledger Fabric: Blockchain-as-a-service (BaaS), Architecture and core components, Hyperledger fabric model, Creation of a simple DAPP. Blockchain vulnerabilities, Smart contract vulnerabilities, Blockchain on CIA security triad, Blockchain based DNS security platform, deploying blockchain based DDOS protection. Use cases: Public Sector, Finance, Supply Chain. Research Aspects in Blockchain

14 Hours

Tools / Languages: Solidity, Remix, Ganache, Metamask.

Text Book(s):

1: Introduction to Blockchain Technology, Tiana Laurence, 1st edition, Van Haren Publishing, 2019. 2. 2. Blockchain Technology from Theory to Practice, Sudeep Tanwar, 1st edition, Springer, 2022.

Reference Book(s):

1. Hands-On Cyber security with Blockchain: Implement DDoS protection, PKI-based identity, 2FA, and DNS security using Blockchain, Rajneesh Gupta, 1st edition, 2018.

2: Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction, Narayanan, Bonneau, Felten, Miller and Goldfeder, Princeton University Press, 2016.

UE22CS342BA6: Digital Forensics and Incident Response (4-0-0-4-4)

Digital forensics is the core set of principles and processes necessary to produce usable digital evidence and uncover critical intelligence. Digital Forensics course provides a deep understanding of the techniques to gather, protect and report the digital confirmations

Course Objectives:

- Introduces to the history and evolution of Digital forensics, Principles, challenges and Incident Response Provide an understanding of the various sources of evidence and the procedures for acquiring the evidence
- Learn about, file structures and Windows file forensics and practice with applicable tools.
- Focus on basics of Linux and MAC OS , study on Network Forensics and Browser forensics
- Focus on mobile device forensics ,email investigations , anti-forensics and Learn on the procedure for final report writing as per the court of law

Course Outcomes:

At the end of the course, the students will be able to

- Understand the Principles of Digital Forensics, Evolution of Computer Forensics, Scope of Cyber Forensics Investigation process and Incident response
- Know the various sources of Digital evidence and acquire the evidence by applying the standard acquisition procedures
- Understand the various operating system file systems and able to perform a better forensic analysis of the same and able to Detect network intrusions and unauthorized access,
- Perform basic mobile investigation and e-mail investigations using appropriate tools. Write professional reports for Legal consumption.

Course Content:

Unit 1 : Introduction to Digital Forensics and Incident Response With Essential Technical Concepts

Introduction to forensics science, Basics of Digital forensics, International standards and practices, Role of examiner in judicial system, Digital forensic: goals, categories ,Challenges, investigation types, Forensics readiness, Digital forensics life cycle, Digital forensics Vs other computing domain **Incident Response**: Attack Lifecycle, Incident Response-Introduction, Team, Incident Response Plan (IRP), Lifecycle, Methodology, Goals, Tools

Acquiring-Analyzing Digital Evidence withEssential Technical Concepts :Digital evidences types, File structures, understanding and working of the hard disks-HDD,SSD, Data recovery considerations and Understanding: Allocated and unallocated space, Data Persistence, Page file(Swap space) Initial Response and First Responder Tasks- Search and Seizure, First responder tool kit and tasks, Documenting digital crime scene, Packing and transporting digital evidences, Conducting interview, Acquiring digital evidence, Analyzing Digital Evidence

14 Hours

Unit 2 : Filesystems - Windows Systems, Windows Registry

Filesystems - Windows file systems- File Systems- understanding and examining FAT, NTFS, File allocation table(FAT), New Technology File System(NTFS), Windows Registry: windows system artifacts, Introduction Deleted Data Hibernation File (Hiberfile.Sys) Registry Print Spooling Recycle Bin Metadata Thumbnail Cache Most Recently Used (MRU) Restore Points and Shadow Copy, Windows forensic analysis- Timeline Analysis, file recovery, Windows registry analysis, Deleted registry key recovery.

14 Hours

Unit 3 : Introduction to Linux and Mac OS X Systems ,Network Forensics and Web browser Forensics

Linux Systems and Artifacts- Linux file systems (Ext2/Ext3), file system layer, filename layer, metadata layer, data unit layer, deleted data, Linux logical volume manager, Linux boot process and services, Linux System Organization and Artifacts, Unix/Linux Forensic Investigation: Unix/Linux forensics, investigation steps and technologies, Principles of file carving **Mac OS X Systems and Artifacts**- OS X File System Artifacts- HFS+ Structures, OS X System Artifacts. **Network Forensics**: Networking Fundamentals and Types of Networks, Network Forensics Overview, Securing A Network, Developing Procedures for Network Forensics Network Security Tools ,Network Attacks, Forensic Footprints, Network Evidence and Investigations, Seizure of Networking Devices,Applying Forensic Science To Networks **Web browser forensics**- Internet overview, IE, Microsoft web browser, Firefox, Google Chrome, web browser investigation tools.

14 Hours

Unit 4: E-Mail forensics, Mobile Device Forensics, Anti-forensics and Report Writing

Email Forensics- Steps in email communications, List of E-mail protocols, E-Mail header examination, email forensics, recovering emails

Mobile device forensics- Cellular Networks and How They Work, Overview of Cell Phone Operating Systems, Potential Evidence Found on Cell Phones, Collecting and Handling Cell Phones as evidence, Cell Phone Forensic Tools, mobile device forensic investigation, storage location, acquisition methods

Anti-forensics: Introduction to Antiforensics, Classification of antiforensics techniques, antiforensics Practices-Data Wiping and Shredding, Trail Obfuscation, Encryption, Data Hiding, stenography techniques and tools, Introduction to malware analysis and malware forensics **Report Writing**: Prep work for report Writing, Structure of the report, Characteristics of a good report, Document design and good writing practices, Legal Acceptance , digital forensics report writing for High-Tech Investigations.

14 Hours

Tools / Languages :Open source tools on Forensics.

Text Book(s):

1: "The basics of digital forensics: the primer for getting started in digital forensics", Sammons, J. Elsevier, 2012.

2: "Digital Forensics Basics: A Practical Guide Using Windows OS", Hassan, N. A. A press 2019.

3: "Practical Cyber Forensics- An Incident-Based Approach to Forensic Investigations", by Niranjan Reddy, A Press, 2019.

4: "Introductory Computer Forensics-A Hands-on practical Approach", by Xiaodong Lin, Springer, 2018.

Reference Book(s):

1: "Guide to Computer Forensics and Investigations", Bill Nelson, Amelia Phillips and Christopher Steuart, Course Technology, Cengage Learning, 2010.

2: "Digital forensics with open source tools", Altheide, C., & Carvey, H. Elsevier, 2011.

3: "Digital Forensics Workbook_-Hands-on Activities in Digital Forensics", Michael K Robinson ,CreateSpace Independent Publishing Platform, 2015.

UE22CS342BA7 - Digital Twin and eXtended Reality (4-0-0-4-4)

This course is an in-depth exploration of the concepts, technologies, and applications of Digital Twin and Extended Reality. Through a combination of theoretical lectures and practical projects, students will gain the knowledge and skills needed to develop intelligent systems, analyse security threats, and apply Digital Twin and Extended Reality solutions to various industries.

Course Objectives:

- Understand the fundamental concepts and technologies of Digital Twin and Extended Reality, including AR and VR, 3D transformation, AI/ML, and data visualization.
- Develop the skills needed to design and implement Digital Twin and Extended Reality solutions, including the ability to create mock, functional, and executable twins.
- Analyse security threats to Digital Twin and Extended Reality systems and apply trust modelling techniques to ensure the security and integrity of these systems.
- Apply Digital Twin and Extended Reality solutions to various industries, including industrial IoT, critical infrastructure, Agri infrastructure, and connected vehicles.
- Collaborate with peers on practical projects to gain hands-on experience in developing intelligent systems and applying Digital Twin and Extended Reality solutions to real-world problems.

Course Outcomes:

At the end of this course, the student will be able to:

- Students will be able to demonstrate a deep understanding of the fundamental concepts and technologies of Digital Twin and Extended Reality, including AR and VR, 3D transformation, AI/ML, and data visualization.
- Students will be able to design and implement Digital Twin and Extended Reality solutions, including the ability to create mock, functional, and executable twins, and apply generative AR techniques.
- Students will be able to analyse security threats to Digital Twin and Extended Reality systems and apply trust modelling techniques to ensure the security and integrity of these systems.
- Students will be able to apply Digital Twin and Extended Reality solutions to various industries, including industrial IoT, critical infrastructure, Agri infrastructure, and connected vehicles, and assess the benefits and limitations of these solutions.
- Students will be able to collaborate effectively with peers on practical projects and demonstrate the ability to apply Digital Twin and Extended Reality solutions to real-world problems, such as improving operational efficiency, enhancing safety, and reducing costs.

DesiribleKnowledge :UE22CS252A- Data Structures & its Applications

Course Content:

Unit 1: Introduction to AR and VR

Overview of AR and VR technologies, Introduction to the OpenGL graphics pipeline, Rotation, Translation and Scaling , 3D transformation, homogeneous transformations, First OpenGL program, Controls and Animations, Vector Space, Affine Space and Euclidian Space, Affine Transformation, Transformation in Homogeneous Coordinates, 3D transformation, Homogeneous Transformations, Concatenation of Transformation, Transformation Matrices, Interface to Three dimensional Applications, Quaternions Basics, Quaternions, multiplication and Rotation, Quaternion Visualization

14 Hours

Unit 2: Digital Twin Essentials

The Big Picture of Digital Twins, History of the Digital Twin, Origin of the Digital Twin concept, Digital Twin and Product Life cycle Relationship, Types of Digital Twin: Discrete Vs Composite, Product versus facility, Types of Digital Twin: Simulation versus operational, Analytics versus physics-based, Characteristics of a Digital Twin, Digital Twin Architecture, Industrial Digital Twin applications, Examples of mock, functional, and executable twins, Metaverse and Metaversity, About Azure Digital Twin, Azure Digital Twin Explorer, DTDL Models, Industry Ontologies, Setting up mixed reality DT using Azure DT in Unity, Build a 3D scene in Mixed Reality in Unity, Configure 3D assets for mixed reality in Unity, Connect IoT data to mixed reality with Azure Digital Twins and Unity, Connect IoT data to mixed reality with Azure Digital Twins and Unity

14 Hours

Unit 3: Intelligent Systems for IIoT.

Data Driven Modelling, Physics Driven Modelling, Hybrid (Data-Physics) Modelling, Examples of Physics Driven Modelling: DT of a Propeller of a Drone, DT of a ceiling fan, Bio-Mechanical DT of human body; Examples of Data Driven Modelling: DT for RUL of a Battery, DT for traffic Mobility in Bangalore, DT of RUL of GUT; Hybrid Modelling: Heart Digital Twin, DT for Solar Array, DT for CAR using OBD., Case Study: Manufacturing, Case Study: Healthcare, Case Study: Buildings, Case Study: Transportation & Logistics

14 Hours

Unit 4: Digital Twin & Cyber Security.

Digital twins and cybersecurity, Security Framework, Digital twins threat modelling, Common attacks on digital twins, Common attacks on digital twins, Digital twin authentication and identification challenge, IDS, IPS, Authentication Methods, Communication Channel Protection, building cyber resilience in digital twins, Privacy Framework, Lack of Privacy, and trust, Privacy by Design, Enhancing trust with block chain integration.

14 Hours

Tools/ Languages: C/ C++/ JAVA/ Python using OpenGL.

Text Book(s):

1:“Interactive Computer Graphics - A top-down approach with shader-based OpenGL”, Edward Angel and Dave Shreiner, Pearson Education, Sixth edition, 2012.

2: Building Industrial Digital Twins by Shyam Varan Nath & Pieter van Schalkwyk, by Packt Publishing Ltd.

Reference Book(s)

1:Diab, W. W., A. Ferraro, B. Klenz, S. W. Lin, E. Liongosari, W. E. Tannous, and B. Zarkout. "Industrial IoT Artificial Intelligence Framework." (2022): 1-59.

2: El Saddik, Abdulmotaleb, ed. Digital Twin for Healthcare: Design, Challenges, and Solutions. Elsevier, 2022.

Web References:

1:https://in.mathworks.com/campaigns/offers/next/digital-twins-for-predictive-maintenance/modeling-methods.html

2: https://learn.microsoft.com/en-us/training/paths/build-mixed-reality-azure-digital-twins-unity/

UE22CS342BA8: Topics in Wireless and 5G Networks (4-0-1-4-4)

Wireless Mobile Networking is a dynamic field that has spurred tremendous excitement and technological advances. This course will cover topics in wireless networking and introducing 5G networks. The objective of the course is to introduce students to recent advances in mobile networking, with an emphasis on practical design aspects of mobile systems.

Course Objectives:

- Introduce the different trends of wireless network technologies
- Applications of various wireless communication fundamentals based on its usage
- Provide cellular systems and standards for different generations, and explore cellular concept of design fundamentals such as Frequency Reuse, Handoff etc.
- Discuss design parameters and applications of computing and architecture for different mobile standards

Course Outcomes:

At the end of the course, the student will be able to:

- Develop knowledge on wireless LANs and its evolution
- Understand the concepts of wireless communications and wireless networking
- Learn cellular concepts, e.g. frequency reuse and multiple access technologies
- Understand cellular evolution from 1G to 5G and associated mobility management

DesiribleKnowledge: UE22CS252B – Computer Networks.

Unit 1 :Overview of Wireless communication

Introduction, Wireless Local Area Networks: IEEE 802.11, 802.11 Frame Format, Basic Access Methods – CSMA/CA, ALOHA, P-persistent CSMA, Non-persistent CSMA, Distribution Coordinating Function (DCF), DIFS, SIFS, Point Coordination Function (PCF), Light Weight Access Point Protocol (LWAPP), IEEE 802.1x, 802.11a,h,k,p, WPAN Technologies: Bluetooth, Bluetooth Protocol Stack, Piconets, NFC, 6LOWPAN, LPWAN-LORA, Wireless Local Loop (WLL)-Local Multipoint Distribution System (LMDS), MMDS, WiMAX - QDMA, Adhoc Networks and Routing, MANET & VANET, Wireless Sensor Networks, Self-Organizing Networks, Wireless Mesh Networks, RFID: Concept, frequency band, classification of RFID tags, applications

14 Hours

Unit 2 :Wireless Communication Fundamentals

Multiple Access for Wireless Systems – FDMA, TDMA, CDMA, Multiple-Input and Multiple-Output (MIMO), Coding Techniques – Pulse Code Modulation (PCM), Differential PCM (DPCM), Adaptive PCM (ADPCM), Delta Modulation (DM), Wireless Modulation Schemes – Binary Phase-Shift Keying (BPSK), Quadrature Phase-Shift Keying (QPSK), Gaussian Minimum-Shift Keying, Frequency Spectrum, Long Range Communication-Satellite Communication, Smart Antennas, Wireless networking and security issues in 802.11 – Static filtering based on MAC address, Wired Equivalent Privacy (WEP), WPA & WPA2, 802.11e QoS issues – DCF & HCF, Scanning, Increased Bit Error Rate (BER), Multipath Propagation, Authentication, Path Loss, RF signal interference, traffic and resource allocation – flow control, error control, mobility, routing, Channel Allocation Scheme, Power Management

14 Hours

Unit 3 : Fundamentals of Cellular System

Evolution of Wireless Network generations (1G, 2G, etc), Mobile Radio standards – AMPS, N-AMPS, GPRS, GSM, UMTS, CDMA 2000, roaming, Cellular Concept – cell structure, cluster, frequency reuse, basic cellular system: mobile terminal, base station (BS), mobile switching center (MSC), home location register (HLR) and visitor location register (VLR), traffic and control channel (forward and reverse).

Handoff Strategies: Concept of handoff, Types of handoff – hard and soft handoff, Queued Delay, MAHO (Mobile Assisted Handoff), Improper handoff, Umbrella cell approach, Improving coverage and capacity in cellular systems: Cell splitting, Microcell Zone concept, Repeaters for range extension. **(**

14 Hours

Unit 4 : Digital Cellular Mobile Standards

Global System for Mobile Communication (GSM) – features, architecture, GSM call routing, stages of call processing in GSM, Signaling System No. 7 (SS7), Need for 3G, 4G and 5G technologies, UMTS/W-CDMA standard – features, architecture, specifications and other procedures, Next generation mobile standards – features of 4G & 4G LTE, 5G

14 Hours

Text Book(s):

1: “Wireless and Mobile Network Architectures”, Lin Yi-Bang and ClamtacImrich, John Wiley & Sons 2001

Disclaimer:

There is no fixed text book. Lectures will be drawn from several sources (books, articles, papers) as most available texts do not cover all of the course material. The text book listed covers some of the topics that are relevant to the material presented.

Reference Book(s):

1: “Wireless Communications: Principles and Practice”, Theodore S. Rappaport

2: “Wireless and Mobile Networks: Concepts and Protocols”, Dr. Sunil Kumar S. Manvi, Mahabaleswar S. Kakkasageri, Wiley India

3: “Wireless Communication Networks and System”, by Cory Beard and William Stallings,1st edition, Pearson, 2015.

4: “Wireless Communication and Networks: 3G and Beyond”, ITI Saha Mishra, McGraw-Hill Education

5: “Wireless Network Security: A Beginner’s Guide”; by Tyler Wrightson, McGraw-Hill Education

UE22CS342BA9: Generative AI and its Applications (4-0-0-4-4)

Large Language Models (LLM) in Generative AI represent a significant leap forward in the field of artificial intelligence, leveraging vast amounts of data and advanced algorithms to understand, generate, and interact with human language in a highly sophisticated manner. These models are capable of producing human-like text, understanding context, and even creating content that is coherent and contextually relevant. Their applications are diverse and far-reaching, encompassing areas such as customer service automation, where they can handle inquiries and provide support; content creation, where they generate articles, stories, and marketing copy; and education, where they assist in tutoring and personalised learning experiences and many more applications.

Objectives:

- Gain an in-depth understanding of Large Language Models (LLMs), their architecture, evolution, and applications across different domains.
- Develop hands-on skills in implementing and fine-tuning LLMs for various NLP tasks, using state-of-the-art tools and frameworks.
- Learn and apply advanced techniques in prompt engineering, retrieval-augmented generation (RAG), and multimodal LLMs to solve complex problems.

- Understand and address the ethical implications, potential biases, and privacy concerns associated with the use of LLMs.

Outcomes:

- Demonstrate a solid grasp of LLM architecture, including transformer models like BERT, GPT, ELMo, RoBERTa, and BART, as well as advanced techniques in prompt engineering and RAG.
- Successfully implement and fine-tune LLMs for specific NLP tasks, utilizing frameworks like HuggingFace, and develop multimodal applications that integrate text and image processing.
- Create innovative solutions by applying advanced prompt engineering, reasoning techniques, and agentic workflows, enhancing the capabilities of LLMs in real-world applications.
- Evaluate and mitigate bias, toxicity, and privacy issues in LLM outputs, ensuring responsible and ethical use of AI technologies in various applications.

Unit 1: LLM Architecture and Models

LLM Basics and evolution, NLP:WordEmbeddings, POS and NER, Text Classification, LLM Architecture, Transformer Anatomy, BERT, GPT, ELMo, ROBERTa, BART architectures,

Case study:Loading the LLM using file system, client-server pattern,NLP tasks using transformers from HuggingFace, Tokenization and Embedding using HuggingFace Models.

14 Hours

Unit 2: Prompt Engineering and RAG

Prompt Engineering techniques ,Zero/few shot inferencing, Chain/Tree/Graph of Thought, Reasoning and Act, LangChain basics, Naive Retrieval Augmented Generation: Chunking, Embedding, Vector stores, processing different data types and formats, MoE, Advanced RAG technique, Building Q&A systems with Naive RAG and Advanced RAG techniques, Bias and toxicity evaluation.

14 Hours

Unit 3: Agent and Multimodal LLM

Agentic Workflows using Microsoft Autogen, CrewAI. GAN, Diffusion networks and Image generation Multimodal LLMs: CLIP, BLIP models.

14 Hours

Unit 4: Fine Tuning

LLMfinetuning: Data resolution/precision, Quantization, finetuning principles, Prompt tuning and soft prompts, Finetune own LLM, Harm of LLM, Data behind LLM, LLM scaling and privacy.

14 Hours

Tools/Languages: Python, HuggingFace, LM Studio, Kaggle

Text Book:

1. Quick Start Guide to Large Language Models: Strategies and Best Practices for Using ChatGPT and Other LLMs by Sinan Ozdemir, Addison-Wesley Professional, Edition 1 2023

Reference Books:

1. Pretrain Vision and Large Language Models in Python: End-to-end techniques for building and deploying foundation models on AWS,by Emily Webber, packtpublishing,Edition 1, 2023
2. Transformers for Natural Language Processing: Build innovative deep neural network architectures for NLP with Python, PyTorch, TensorFlow, BERT, RoBERTa, and more by Denis Rothman (Author) ,publishing,Edition 1, 2023
3. Understanding Large Language Models: Learning Their Underlying Concepts and Technologies by byThimiraAmaratunga, APress, Edition 1 , 2023
4. Retrieval-Augmented Generation (RAG): Empowering Large Language Models (LLMs) by Dr. Ray Islam (Author) ,Edition 1,2023

UE22CS343BB1: Heterogeneous Parallelism (4-0-0-4-4)

This course focuses on parallel heterogeneous architectures as well as programming models and imparts pragmatic

skills to program using parallel programming languages and frameworks trending in the industry.

Course Objectives:

- Familiarize with various parallel heterogeneous architectures, associated techniques and programming models.
- Acquaint with Memory Consistency & Coherence.
- Acquaint with Concurrency Bugs and Resolution Techniques.
- Familiarize with opportunities and challenges in Parallel Programming using popular languages & frameworks.

Course Outcomes:

At the end of this course, the student will be able to:

- Understand the underpinnings of Parallel Heterogeneous Architectures and Parallel Computing Techniques.
- Understand Memory Consistency Models and Coherence Techniques, Techniques for Parallelism Bugs' Resolution
- Program using popular Parallel Programming Frameworks and Languages trending in the industry.
- Engineer high performance migration of varied applications appreciating and assimilating varieties of parallelism.

Desirable Knowledge: UE22CS151B – Problem Solving with C, UE22CS251B – Microprocessor and Computer Architecture.

Course Content:

Unit 1: Fine Grained Parallelism

Review on Parallelism and Performance, Instruction Level Parallelism and Enhancement Techniques, Prediction and Speculation, Code Optimization, Cache optimized Programming. Laws of Parallelism, Data, Task and Pipeline Parallelism, Pthreads.

14 Hours

Unit 2: Coarse Grained Parallelism &Parallel Algorithms

Multithreaded and Multi-Core architectures, Parallel Algorithms, Task Decompositions and Mapping.GPUs and GPGPUs, GPU Architectures, Many-Core Heterogeneous Architectures.

14 Hours

Unit 3: Parallelism Frameworks

OpenMP, Race Conditions, Deadlocks & Debugging, Memory Models for Parallel Programming, Memory Consistency Models, Message Passing Programming,

14 Hours

Unit 4: GPU Programming & Hardware Accelerators

CUDA Programming Hardware acceleration platforms (FPGAs and TPUs), Architecture and organization of modern FPGAs, Case Studies, Concurrency in Main stream Languages.

14 Hours

Tools/Languages:pthread, OpenMP, CUDA, openCL.

Text Book(s):

1: “Parallel Programming: for Multicore and Cluster Systems”, Thomas Rauber, GudulaRunger, 3rd Ed., Springer, 2023.

Reference Book(s):

1: “Computer Architecture: A Quantitative Approach: John Hennessy, David Patterson”, 6th Edition, Morgan Kaufmann, 2017.

2: “Programming Massively Parallel Processors”, David Kirk and Wen-meiHwu, 3rd Ed, Morgan Kaufmann, 2016.

3: “Computer Systems: A Programmer's Perspective”, Randal E. Bryant, David R. O'Hallaron, Pearson, 2nd Ed, 2016.4: 4: “Parallel Programming for FPGAs”, Ryan Kastner, Janarbek Matai, and Stephen Neuendorffer, Creative Commons, 2018.

UE22CS343BB2: Topics in Deep Learning (4-0-0-4-4)

Deep Learning has received a lot of attention over the past few years and has been employed successfully by companies like Google, Microsoft, IBM, Facebook, Twitter etc. to solve a wide range of problems in Computer Vision and Natural Language Processing. In this course we will learn about the building blocks used in these Deep Learning based solutions. At the end of this course students would have knowledge of deep architectures used for solving various Vision and NLP tasks. This course requires the student to have a desirable knowledge of Machine Intelligence.

Course Objectives:

- Familirise various CNN architectures and itroduce transform leraning.
- Acquire knowledge on sequence models like RNN, LSTM and GRU.
- Upskill students in generative models and meta learning.
- Attain knowledge on reinforcement learning and other latest deep learning models

Course Outcomes:

At the end of this course, the student will be able to:

- Implement a Neural Network using Tensor Flow and Keras
- Classify images using CNN.
- Solve time-series related problems with RNN.
- Generate data in the form of images using GAN.

Desirable Knowledge: UE22CS352A - Machine Learning.

Unit 1:Introduction to Deep Learning

Introduction**,** Activation functions, Loss functions, Batch Normalization, Regularization and Optimization.**Convolutional Neural Network(CNN):** Introduction, Filters, Feature Maps, Max-Pool Layers, Other Pooling Types, Back Propagation. Convolution Architectures - Alexnet, ZFNet, VGGNet, GoogleNet, ResNet. RCNN, FRCNN, Faster RCNN, YOLO V5. **Transfer Learning**: Introduction, Motivation, Variations, TL Architecture of CNNs. Hands-on: Assignment on CNN & TL.

- **14 Hours**

Unit 2:Recurrent Neural Networks (RNN) and Transformers

Introduction-Recurrent Neurons, Memory Cells, Variable-Length Input-Output Sequences, RNN Architecture, Sequence learning problem, BPTT-Back Propagation Through Time, truncated BPTT, Vanishing and Exploding Gradient, Bidirectional RNN, LSTM Cell and GRU Cell, Text

Classification with RNN, Encoder/Decoder architecture, Seq2Seq model with Attention , Transformer model and BERT architecture, Transformer Attention and its implementation.

Hands-on: Problem using RNNs, Transformers and TL / predefined model **14 Hours**

Unit 3:Generative Models & Meta Learning: Introduction to Autoencoders , Regularization in autoencoders, Denoising autoencoders, Sparse autoencoders, Contrastive autoencoders , Variational Auto Encoders(VAEs). **Generative Adversarial Networks**(GANs)- Architecture and Training Methods, Image Generation, DCGAN, Style GAN, WGAN, Applications.**Graphical Neural Networks(GNN)**: Introduction to GNNs, Graph Convolution Networks, Applications. **Meta Learning**: Introduction to Meta Learning, MAML, FOMAML, Adaptive Neural inductive Learning, Siamese Networks.

Hands-on: Problem on Fashion **MNIST** dataset for generating apparels.

•**14 Hours**

Unit 4:Reinforcement Learning, **Diffusion Models, Federated Learning and Overview of Latest Deep Learning Models:**

Introduction, Basic Framework of RL, Learning to Optimize Rewards, Credit Assignment Problem, Temporal Difference, Learning and Q Learning.**Deep RL:** Deep Q Learning, Training and Testing.

Diffusion Model, Stable diffusion architectures, Introduction to Vision Transformers, GPT Architecture. **Federated Learning:** Horizantal, Vertical and FTL (Federated Transfer Learning).

Hands-on : Flower Architecture for MIMC3 dataset **OR** Learning to play a simple game using Double Deep (DD) Q-learning implementation on OpenAI gym.

•**14 Hours**

Tools/ Languages: Pytorch.

Text Book(s):

1: “Advanced Deep Learning with Python” - Ivan Vasilev, Packt Publishing, 2019.

2: “Neural Network and Deep learning” by Charu C Agarwal. Springer International Publishing 2018.

Reference Book(s):

1: “Hands-on Machine Learning with Scikit-Learn and TensorFlow”, Aurelian Geron, O’REILLY, 1st Edition, 2017.

2: “Deep Learning with Keras”, Antonio Gulli and Sujit Pal, Packt Publishing, 1st Edition, 2017.

3: “Pattern Recognition and Machine Learning”, Christopher Bishop, Springer, 1st Edition, 2011 (Reprint). 4:Handouts: Transfer Learning / Latest Deep Learning Techniques / Vision Transformers/ GPT / FL

UE22CS343BB3 - Database Technologies (4-0-0-4-4)

The last decade saw enormous advancements in the design of large-scale data processing systems due to the rise of Internet services. This course covers the architecture of modern data management systems. Topics include storage management, query optimization, distributed, parallel and stream processing, with a focus on the key design ideas shared across many types of data intensive systems.

Course Objectives:

- Data Storage Techniques, Indexing Mechanisms.
- Query Processing and Query Optimization Techniques.
- Parallel and Distributed Databases.
- Data Streaming Systems, current Trends in Design and Implementation of Database Systems.

Course Outcomes:

At the end of the course, the student will be able to:

- Design and Deploy Storage Solutions and Indexing mechanisms for optimized performance of databases.
- Perform Query Optimization.
- Apply Parallel and Distributed Database approaches to solve problems of large databases.
- Select an approach to manage "Data Streams" and apply techniques learnt to Develop or Select Specialized Database Applications.

Desirable Knowledge: UE22CS351A – Database Management Systems.

Course Content:

Unit 1- Relational Data Model and Storage Formats and Indexing

Review of Relational Design Theory – Relational Algebra. Secondary Storage Management: The Memory Hierarchy, Flash Storage and Database Buffer, Architecture, Accelerating Access to Secondary Storage, Disks and Disk Failures, Arranging Data on Disk, Block and Record Addresses, Variable-Length Data and Records – Column Stores, Record Modifications. Index Structures: Concepts, Types of Indexes, B+ Tree Indexes, Hash Tables, Multidimensional Data and Indexes and It's Applications, Multiple-key Indexes and R-trees, Bitmap Indexes, Indexes in SQL.

14 Hours

Unit 2 - Query Processing and Optimization

Query Execution: Physical-Query-Plan Operators, One-Pass Algorithms and Two-Pass Algorithms (based on Sorting and Hashing), Clustered and Non-clustered indexes, Buffer Management. The Query Compiler: Parsing and Preprocessing, Algebraic Laws for Improving Query Plans, From Parse Trees to Logical Query Plans, Cost based Plan Selection, Choosing an Order for Joins.

14 Hours

Unit 3- Parallel and Distributed Databases

Avenues for Parallelism: Parallel Database Architectures, Models of Parallelism, Intra-query and Inter-query Parallelism, Parallel Algorithms on Relations. Distributed Databases – Distributed Data Storage, Types, Advantages, Data Fragmentation and Replication, Distributed Transactions, Distributed Query Processing, Distributed Commit – Two-phase Commit, Distributed Locking, Distributed Catalog Management, Concurrency Control, Peer-to-Peer Distributed Search – Chord Circles.

14 Hours

Unit 4 - Data Stream Management & Trends in Design and Implementation of Database Systems

Introducing Stream Processing, Stream Processing Model, Streaming Architectures, Apache Spark as a Stream Processing Engine, Spark's Distributed Processing Model, Spark's Resilience Model, Apache Kafka, Amazon Kinesis.

Introduction to Vector database, serverless Vector databases, Use Cases. Current Trends in Design and Implementation of Database Systems and Decision Support Systems, Data Warehousing and Data Mining Applications, Introducing Data Lake, Data Mesh and Data Fabric, Multi-model Databases.

14 Hours

Text Book(s):

1: "Database Systems: The Complete Book", H Garcia-Molina, JD Ullman and J Widom, 2nd Ed., Pearson, 2018.

2: "Stream Processing with Apache Spark", Gerard Maas & Francois Garillot, O'Reilly, June 2019.

Reference Book(s):

1: "Fundamentals of Database Systems", Elmasri and Navathe, Pearson Education, 7th Ed., 2016.

2: "Streaming Systems" by Tyler Akidau, SlavaChernyak, Reuven Lax, O'Reilly, July 2018.

UE22CS343BB4: Machine Learning on Graphs (4-0-0-4-4)

This course introduces the students to the three meta landscapes of representation, optimization, and objective for the learning tasks on graph data. Graphs are ubiquitous in almost everything today –in social graphs, large-scale biological systems, road networks, polypharmacy, and user-item relations. Graphs, being non-Euclidean, require different machine-learning approaches. This course starts by exploring graph features for traditional machine learning on graphs and discussing a few noteworthy classical graph ML approaches. The second unit focuses on shallow embedding methods for graphs in general and then the same for knowledge graphs. The third unit focuses on representation learning using graph neural networks and discusses key algorithms along with extensions like temporal graphs and multi-relational graphs. The fourth unit discusses the deep generative models on the graph and model-building techniques using graph neural networks. The theory input concludes with a discussion on the interdisciplinary applications of Graph Neural Networks, interpretability, and meta-learning perspectives of graph AI.

Course Objectives:

- Learn graph feature-based traditional machine learning.
- Learn shallow embedding approaches for graphs and multi-relational knowledge graphs.
- Learn neural methods of representation learning on graphs.
- Learn GNN modeling techniques and deep graph generative models with interdisciplinary applications.

Course Outcomes:

At the end of this course, the student will be able to:

- Should be able to do feature engineering-based graph ML tasks.
- Should be able to do downstream machine learning tasks on graphs using shallow embedding.
- Should be able to solve a downstream machine-learning task using a graph neural network.
- Should be able to use graph modeling techniques and appreciate deep generative models on graphs.
-

Desired Knowledge: UE22CS352A – Machine Learning, UE22CS343AB3 -Graph Theory and its applications

Course Content:

Unit 1: Traditional machine learning on graph

Types of complex Graphs and computational tasks on Graphs. Graph Features for traditional machine learning– Node level features. Graph kernel features- Graphlet& Motif, Weisfeiler-Lehman kernel. Measures for neighborhood overlap. Graph clustering and spectral methods –Graph Laplacians, Graph Cuts and clustering, Spectral clustering. Encoder-Decoder perspective – encoder, decoder, and optimization of the encoder-decoder model.Factorizationbased approaches.

14 Hours

Unit 2: Graph Representation Learning Based on shallow embedding methods

Random Walk-based embedding –DeepWalk and Node2Vec. Limitations of shallow embedding.Shallow embedding in multi-relational Knowledge Graph- embedding as reconstruction task, loss function, and decoders.Knowledge Graph Completion Task-TransE, TransR, DistMult, ComplEx. Semi-supervised Learning on Graph - Label Propagation Algorithm and Label Spreading Algorithm

14 Hours

Unit 3: Graph Learning using graph neural network

Vanilla GNN - Neural message passing framework. Generalized neighborhood aggregation. Generalized update methods. Graph Convolution Networks (GCN): Spectral methods(GCN, AGCN) and Spatial Methods(DCNN, DGCN, LGCN and GraphSAGE), Multi-relational GCN and Graph Attention Networks (GAT), Graph Transformer. Graph pooling. Expressiveness, designing Maximally expressive GNNs, Expressive Power by aggregate functions. Applications and loss functions.

14 Hours

Unit 4: Deep Generative Models, GNN on complex graphs and interdisciplinary applications

Deep Generative models on graph – Variational autoencoder on the graph, Generative adversarial network on the graph, Auto-regressive methods. GNN on complex graphs – Heterogeneous GNN, Dynamic GNN, Hypergraph GNN. Application of GNN to Biology and Chemistry, Interdisciplinary Applications of GNN: Unstructured data – Text and Image. Structured Data - Social Network, Recommender System, Cyber Security. An introduction to trustworthy graph AI, Meta-Learning perspectives of graph ML.

14 Hours

Tools/Languages: NetworkX for statistical features of graphs, Tensorflow, Keras, and Scikit Learn for traditional graph ML, and Pytorch Geometric for Graph Neural Networks.

Text Book(s):

1: “Graph Representation Learning”, William L Hamilton, Morgan and Claypool Publishers, 2020.

2: “Deep Learning on graphs”, Yao Ma and Jiliang Tang, Cambridge University Press, 2021

3. “Introduction to Graph Neural networks”, Zhiyuan Liu and Jie Zhou, Synthesis Lectures on Artificial Intelligence and Machine learning, Morgan and Claypool Publishers, 2020.

Reference Book(s):

1: “Social Media Mining”, Reza Zafarani, Cambridge University Press, 2015 (Asian Economic edition available)

2: “ Network Science” by Albert Barabasi , 2016

UE22CS343BB6 : Information Security (4-0-0-4-4)

This course will present security aspects from a secure software life cycle process – requirement, architecture, design, coding, and testing. Students will have opportunity to dwell well in to technical "how to" with hands-on sessions, assignments, and some case study discussions.

Course Objectives:

- To understand various cyber threats and attacks and secure software development process.
- To learn attack and defence mechanisms for buffer overflow, shellshock attack, etc.
- To understand the concept of threat modelling and its application.
- To learn about the most common web application security vulnerabilities. and apply various penetration testing techniques and tools.

Course Outcomes:

At the end of this course, the student will be able to:

- Identify possible misuse cases in the context of software development.
- Defend against various attacks and how to write secure code.
- Apply threat modelling techniques to expose inherent vulnerabilities in applications.
- Design and develop secure web applications and Exploit software vulnerabilities and launch attacks.

Course Content:

Unit 1: Introduction and Privilege Escalation Attacks

Software Threats, Attacks and Vulnerabilities, CIA Triad, OWASP Top 10, CVE, Security and reliability, Security vs. privacy, Cyberattack Types, Anatomy of an Attack, Security Concepts and Relationships. Use cases and Misuse cases, Misuse case legend, Security use case vs Misuse case, Secure Software Development Life Cycle (SDL). Case Study: Target case study. Set-UID program: Need for privileged programs, Set-UID mechanism, Superman story, Attack surfaces, Invoking other surfaces, Principle of least privilege. Environment variables and attacks: Environment variables, Attack surface, Attacks via Dynamic linker, External program, and Library. Lab: Set-UID program & Environment variables and attacks. Shellcode attack on Set-UID and CGI programs.

14 Hours

Unit 2: Software Vulnerabilities and Malicious Software

Buffer overflow attack: Program memory layout, Stack and function invocation, Stack buffer-overflow attack, Attacks with Unknown address and Buffer size, Shellcode, Countermeasures & Defeating it. Return-to-libc attack: Introduction, Launch the attack part I & part II. Format string vulnerability: Introduction to functions and format string, Vulnerable program, Exploiting the vulnerability, Code injection attack, Countermeasures. **Case study: Target case study**. Malware and its Types, Malware analysis: Conifer, Morris, Stuxnet worm, Ransomware.

14 Hours

Unit 3: Threat modelling and Basic Web Security

Threat Modelling, Trust Boundaries, Attack Surfaces, Brainstorming, Modelling Methods, STRIDE model and variants, Defensive tactics, and Technologies, Privacy Threats, Taxonomy and Types. Web security basics, Attacks on HTTP GET and POST services, Cross Site Request Forgery (XSRF/CSRF): Cross-site requests and its problems, CSRF attacks, Countermeasures. **Case study: Apple - Privacy vs Safety.**

14 Hours

Unit 4: Web application security and Penetration Testing

(XSS/CSS) Attack: CSS attack, CSS attacks in action, Self-propagation, Preventing CSS attacks. SQL injection attack: Introduction to SQL, interacting with database in web, Launching SQL injection attacks, Countermeasures. Static analysis, Penetration testing: Introduction, Benefits, Drawbacks, Penetration testing tools and Fuzzing, Patching.

14 Hours

Note: Hands-on experience for relevant topics in the form of Lab and/or Assignment is given. Relevant cyber security case study for undergraduate students is discussed.

Hands-on exercises:

1. Set-UID program & Environment variables and attacks
2. Shellcode attack
3. Buffer overflow attack
4. Return-to-libc attack.
5. Format string vulnerability.
6. CSS/XSS attack
7. CSRF attack.
8. SQL injection.
9. Shell shock attack.
10. Penetration testing.(Assignment)
11. Malware 1,2,3 (Assignment)
12. Case study 1,2 (Target, Apple)

Tools / Languages: SEED Labs VM, Scapy, Burp Suite, Metasploit, Nmap, etc.

Text Book(s):

1: “Computer & Internet Security: A Hands-on Approach”, Wenliang Du, 2nd Edition/3rd Edition.

Reference Book(s):

1: “Computer Security: Principles and Practice”, William Stallings and Lawrie Brown, Pearson Education, 3rd Edition, 2014.

2: “Secure Programming with Static Analysis”, Brian Chess and Jacob West, Pearson Education, 2007

UE22CS343BB7: Mobile and Autonomous Robots (4-0-0-4-4)

This course introduces students to the field of autonomous mobile robots. Students will learn the basic concepts and techniques necessary for designing, building, and programming autonomous robots, as well as explore the latest developments in this exciting field. The course will include lectures, hands-on activities, and projects.

Course Objectives:

- Understanding the past, present, and future of autonomous mobile robotics. Getting exposed to the current state-of-the-art of scientific literature.
- The course aims to teach the theoretical and practical fundamentals involved in designing and operating autonomous robots or intelligent agents.
- The introductory discussions cover subtopics like robot perception, planning, and control.
- Other major topics include designing robot parts, integrating sensors, analyzing motion kinematics, simulation testing using ROS/ROS2, handling unmodeled environmental and social factors, and preparing for field deployment.

Course Outcomes:

At the end of this course, the student will be able to:

- Students will gain a comprehensive understanding of autonomous robots and their applications. Understand the ROS architecture and communication protocols and use ROS packages for robot software control.
- Understand the principles of robot manipulators, end-effectors, and identify the sensors and actuators used in robot control.
- They will be able to use path planning and obstacle avoidance algorithms for robot navigation.
- Ultimately, students will be able to apply this knowledge to design and develop their own autonomous robot systems.

DesiribleKnowledge : Robotics with ROS (New Elective Course for Vthsem)

Course Content:

Unit 1: Introduction to Autonomous Robots

Overview of autonomous robots, History and current state of autonomous robots, Basic concepts and terminology, Applications of autonomous robots, Robot Hardware and Software: Sensors and actuators Overview of robot sensors, ROS overview: ROS/ROS2 for robotics, ROS architecture and communication protocols, ROS packages for robot hardware control.

14 Hours

Unit 2: Locomotion and Perception

Ground robots (UGVs) robots, Forward and inverse kinematic, The basics of wheel types and arrangements. Aerial robots (UAVs), Dynamics of thrusters and propellers, The basics of SOTA locomotion systems. Visual and inertial measurements: Gyroscope, accelerometers, IMU, GPS, Range sensors, camera vision and LiDAR.

14 Hours

Unit 3: Robot Vision and Localization

Visual perception: Image processing and filtering, Object detection and tracking, Stereo vision and 3D perception, The basics of scene segmentation and parsing. Introduction and Challenges of Localization, Belief Representation, Map representation, Probabilistic Map-Based Localization: Markov localization, Kalman Filter localization, SLAM: Simultaneous localization and mapping, EKF SLAM, Particle filter SLAM, GraphSLAM, MonoSLAM, Open challenges in SLAM.

14 Hours

Unit 4: Navigation

Introduction Path Planning and Navigation, Path planning: Graph search and Potential field path planning. Obstacle avoidance: Bug algorithm, Vector field histogram, bubble band technique, Curvature velocity techniques, Dynamic window approaches. Revisiting Navigation using Reinforcement Learning and Imitation learning: A Robotics Perspective, Applications and Social Implications.

14 Hours

Text Book(s):

1: Introduction to Autonomous Mobile Robots (Intelligent Robotics and Autonomous Agents series) second

edition by Roland Siegwart, Illah Reza Nourbakhsh, DavideScaramuzza

2: Probabilistic Robotics By Sebastian Thrun, Wolfram Burgard and Dieter Fox. ISBN-13: 978-0262201629, ISBN-10: 0262201623. Intelligent Robotics and Autonomous Agents series; 1st Edition

Reference Book(s):

1: Introduction to Autonomous Robots: Nikolaus Correll, Magellan Scientific, 2016.

2: ROS Robot Programming, ROBOTIS Co., Ltd. From the basic concept to practical programming and robot application.YoonSeokPyo, HanCheol Cho, RyuWoon Jung, and TaeHoon Lim.

3: Introduction to Robotics: Mechanics and Control 4th Edition, John Craig, ISBN-13: 978-0133489798,

Pearson; 4th edition

Recommended Materials

1: Robot Operating System (ROS): http://wiki.ros.org/ROS/Tutorials

2: ROS2 tutorials: https://docs.ros.org/en/foxy/Tutorials.html

UE22CS343BB8 - Security for Internet of Things

This course enables the learner to understand the security threats associated with Internet of Things and perform security testing on connected devices within our homes and enterprise to build a better model for protecting ourselves.

Course Objective:

- Understand the threats of IoT devices
- Understand vulnerabilities, possible attacks and learn the security testing methodology for networks, IoT hardware and radio protocols.
- Enhance knowledge with hands-on experience on IoT security tools and attack analysis.
- Understand the various Security measures to be adopted to secure IoT devices

Course Outcome:

- Identify and describe the variety of IoT systems architectures, essential components and challenges specific to IoT systems.
- Analyze various network and hardware security mechanisms for IoT devices
- Gain hands-on experience on different tools to target IoT ecosystem.
- Analyze and apply appropriate security and privacy solutions for real-world applications.

Course Contents:

Unit 1: Introduction to IoT Security

Introduction to IoT Security, Traditional security vsIoTsecurity,Basic concepts of IoT Architecture from security perspective, Challenges of IoT Security, OWASP Top 10 security risks and consumer IoT security guidance, Threat Modelling for IoT attack, Common IoT threats, Network Hacking: VLAN hopping in IoT networks.

14 Hours

Unit 2: Network Security and Hardware Security for IoT devices

MQTT authentication, Analysing Network protocols: Wireshark dissector and Nmap Scripting Engine module for the DICOM protocol, exploiting zero-configuration Networking: UPnP, mDNS, DNS-SD, and WS-Discovery, Hardware Hacking: UART, JTAG, and SWD Exploitation- Hacking an STM32F103 microcontroller using UART and SWD.SPI and I2C, Firmware Hacking.

14 Hours

Unit 3: Radio Hackingand Smart IoT devices hacking

Radio Hacking:Short Range Radio: Abusing RFID, Bluetooth low energy, Medium Range Radio: Hacking Wi-Fi, Long Range Radio: LPWAN, Smart home, Hacking the smart home: Gaining physical entry to a building, cloning a keylock system's RFID Tag, Jamming the Wireless alarm.

14 Hours

Unit 4: Targeting the IoT Ecosystem and Secure Design of IoT Devices

Playing back an IP Camera stream, Analysing IP Camera Network traffic, Extracting the video stream, attacking a Smart treadmill, Secure design goals: Mitigate automated attack risks, Secure points of integration, Hardware protection measures, IoT IAM infrastructure: PKI for IoT, Revocation support: OCSP, SSL pinning, Authorization and access control with OAuth 2.0 Cryptographic Controls for IoT Protocols.

14 Hours

Tools and Languages : Wireshark, Yersinia, VoIP Hopper, Bettercap,aircrack-ng

Text Book(s):

1: "Practical IoT Hacking", FotiosChantzis, IoannisStais, Paulino Calderon, EvangelosDeirmentzoglou, Beau Woods, March 2021, No Starch Press Publishers, ISBN: 9781718500907.

Reference Book(s):

1: "Practical Internet of Things Security" , Brian Russell, Drew Van Duren, Packt Publishers, 2nd Edition

UE22CS343BB8 : Applied ML in IoT with TinyML

Course Outcomes:

- Design and deploy simple Machine Learning models on microcontrollers for various IoT applications.
- Preprocess sensor data, optimize code, and evaluate the trade-off between model accuracy and resource consumption.
- Utilize tools like TensorFlow Lite Micro and TinyML Kit for TinyML development.
- Integrate sensor data acquisition, communication protocols, and data pipelines for real-time TinyML applications.

Course Objectives:

By the end of this course, students will be able to:

- Understand the fundamental concepts of ML, Deep Learning (DL), and their application in the IoT domain.
- Utilize TensorFlow Lite Micro and TinyML Kit for developing and deploying ML models on microcontrollers.
- Design and implement TinyML applications for various functionalities like keyword spotting, image classification, and sensor fusion.
- Evaluate and interpret the performance of TinyML models on real hardware.

Unit-1: Foundations of Machine Learning: Introduction to IoT&TinyML, The ML paradigm, Building blocks of DL, NN, Regression with Dense NN, Classification with Dense NN, Image Classification using CNN, Datasets and Model Performance Metrics, Preventing Overfitting.

14 Hours

Unit-2: TensorFlow Lite for Microcontrollers &TinyMLKit: What Is TensorFlow Lite for Microcontrollers?, Build Systems, Supporting a New Hardware Platform, Supporting a New IDE or Build System, Integrating Code Changes Between Projects and Repositories, Contributing Back to Open Source, Supporting New Hardware Accelerators, Understanding the File Format, Porting TensorFlow Lite Mobile Ops to Micro, TinyML Kit Overview, TinyML Kit Setup, TinyML Kit Sensor Testing, Sensor Testing, Sensor Fusion.

14 Hours

Unit-3: Applications and Deployment to Microcontrollers (MCUs)-1: TF-Lite, TFL-Micro, TFL-Micro Hello-World example, KeyWord Spotting (KWD) using Edge Impulse, KDW dataset creation, Micro Speech Example, Workflow, Model development & testing, Person-detection example, Image classification.

14 Hours

Unit-4: Applications and Deployment to Microcontrollers (MCUs)-2: Magic Wand: Building and Application, Training a Model, Building a Weather Station with TensorFlow Lite for Microcontrollers, Voice Controlling LEDs with Edge Impulse.

14 Hours

Tools/Languages: Arduino Nano 33 BLE sense , Arduino IDE , Google Colab

Text Books:

1: Pete Warden and Daniel Situnayake, TinyML: Machine Learning with TensorFlow Lite on Arduino and Ultra-Low-Power Microcontrollers, O'Reilly Media, 2020.

2: Vijay Janapa Reddi at Harvard and open-source collaborators, Machine Learning Systems with TinyML, Open-source collaborative-effort book, 2023-present.

3: Gian Marco Iodice and Ronan Naughton, TinyML Cookbook: Combine artificial intelligence and ultra-low-power embedded devices to make the world smarter, 2022.

4: Francois Chollet, Deep Learning with Python, Manning, Second Edition, 2021.