



RAMAIAH
Institute of Technology

CURRICULUM

Academic Year 2026 - 2027

VII & VIII SEMESTER

B.E.

**Computer Science and Engineering
(AI&ML)**

Batch -2023

RAMAIAH INSTITUTE OF TECHNOLOGY

(Autonomous Institute, Affiliated to VTU)

(Approved by AICTE, New Delhi & Govt. of Karnataka)

Accredited by NBA &NAAC with A Grade

About the Institute

In 1962, Dr. M. S. Ramaiah, a visionary philanthropist, established the Gokula Education Foundation with a noble mission to uplift society through education. The same year marked a pivotal moment in technical education in India with the founding of the M S Ramaiah Institute of Technology (MSRIT).

MSRIT offers an impressive array of 18 undergraduate and 15 postgraduate programs, all approved by the All India Council for Technical Education (AICTE). The National Board of Accreditation (NBA) has accredited all eligible programs at the undergraduate (UG) and postgraduate (PG) levels. In March 2021, MSRIT earned an 'A⁺' grade from the National Assessment and Accreditation Council (NAAC), underscoring its unwavering commitment to academic excellence. Since 2007, the institute has proudly held Autonomous Status for its UG and PG programs, a prestigious recognition granted by the University Grants Commission (UGC) and Visvesvaraya Technological University (VTU). According to the Ministry of Education, Government of India's NIRF Ranking Framework, MSRIT ranks 75th among 1,584 leading engineering institutions and 31st among 131 architecture schools in India for 2025, highlighting its remarkable impact and leadership in education.

The institute participated in the Technical Education Quality Improvement Program (TEQIP), an initiative of the Government of India. To foster a research culture, the institute has established several specialized centres, including the Centre for Imaging Technologies, the Centre for Advanced Materials Technology, the Centre for Antennas and Radio Frequency Systems (CARFS), the Centre for Cyber Physical Systems, the Schneider Centre of Excellence, the Centre for Bio and Energy Materials Innovation, JSW Centre of Excellence in Power Systems and more. MSRIT boasts a faculty of 380 highly qualified members, 70% of whom hold doctoral degrees, and over 300 scholars currently pursuing Ph.D. degrees.

The Entrepreneurship Development Cell (EDC) and "Ramaiah Evolute" have been established on campus to support startup incubation. MSRIT features a dedicated Placement and Training department, a strong mentoring and proctorial system, a fully equipped sports department, and a spacious, air-conditioned library with a wide range of books and subscriptions to both international and national journals. The Digital Library provides access to online e-journals from renowned publishers, including Elsevier ScienceDirect, IEEE, Taylor & Francis, and SpringerLink. Additionally, the Institute is a member of DELNET, CMTI, and the VTU E-Library Consortium.

The institute features a state-of-the-art auditorium, a well-equipped recording studio, and multiple advanced conference halls with videoconferencing capabilities. Additionally, MSRIT provides excellent hostel accommodations for both boys and girls, ensuring a comfortable living environment. MSRIT alumni have achieved remarkable success, holding prominent positions in India and internationally. They maintain strong connections with the institute through an engaged and active Alumni Association.

Founded on Dr. Ramaiah's unwavering commitment to knowledge and innovation, this remarkable institution continues to inspire generations of students and shape the future of engineering and technology in the country.

About the Department

The Department offers Bachelor of Engineering (B. E) in Computer Science and Engineering (Artificial Intelligence and Machine Learning). The department is started in the year 2021 with an intake of 60 each. Currently intake for B.E. in Computer Science and Engineering (Artificial Intelligence and Machine Learning) is 129. In 2024, M.Tech. in Artificial Intelligence started with an intake of 18 under the department of Computer Science and Engineering (Artificial Intelligence and Machine Learning). The department has experienced faculty members with the doctoral degree. The faculty members of the department are actively involved in the research activities and publishing their research findings in reputed International Journals/Conferences/Book Chapters. The faculty members have authored books with premiere publishing agencies like Springer, Taylor & Francis.

The department has state-of-the-art laboratories and classroom facilities. The department regularly conducts Bootcamps, Technical Seminars, Workshops, Faculty Development Programs and Hackathons. The department encourages the students to participate in cocurricular and extracurricular activities. The department has established strong collaborations with Industries and premier peer Institutes to design the curriculum to meet the global standards in the domains of AI & ML.

The department is partnering with Ramaiah Medical and Dental College to work on projects which has societal impact. The department is having collaborations with Industries like SAP Labs, Unisys, IBM, HPE, Samsung, Microsoft, GE Healthcare, Adobe, Thomson Reuters, Yubi, JP Morgan, Intellytix etc. to support Internships, Projects, Curriculum upgradation, Guest Lectures, and Industry visits.

VISION OF THE INSTITUTE

To be an Institution of International Eminence, renowned for imparting quality technical education, cutting edge research, and innovation to meet global socio-economic needs

MISSION OF THE INSTITUTE

MSRIT shall meet the global socio-economic needs through

1. Imparting quality technical education by nurturing a conducive learning environment through continuous improvement and customization.
2. Establishing research clusters in emerging areas in collaboration with globally reputed organizations.
3. Establishing innovative skills development, techno-entrepreneurial activities, and consultancy for socio-economic needs.

QUALITY POLICY

We at M. S. Ramaiah Institute of Technology strive to deliver comprehensive, continually enhanced, global quality technical and management education through an established Quality Management System complemented by the synergistic interaction of the stakeholders concerned.

VISION OF THE DEPARTMENT

To provide quality education, inculcate professionalism, and enhance problem-solving and coding, innovative design skills in Computer Science and Engineering, especially in the domain of AI & ML and Cyber Security with a focus on producing professionally competent and socially sensitive engineers capable of working in a global environment.

MISSION OF THE DEPARTMENT

To pursue excellence in Academics, Research, and Innovation by:

1. Enabling creative and dynamic learning environments to impart quality technical education through continuously improving curriculum and pedagogy techniques.
2. Collaborating with the industry, academia and society to strengthen design thinking, research, innovation, and entrepreneurship ecosystem.
3. Encouraging extra and co-curricular activities to nurture leadership qualities with a sense of commitment and accountability and inculcate values and ethics.

PROGRAM EDUCATIONAL OBJECTIVES (PEOs):

A B.E in Computer Science & Engineering (AI&ML) graduate of Ramaiah Institute of Technology:

- PEO1:** Excel in professional career by acquiring knowledge in basic sciences and Computer Science and Engineering and AI&ML principles and contribute to the profession as an excellent employee, or as an entrepreneur.
- PEO2:** Capable of pursuing higher education and research.
- PEO3:** Adapt to technological advancements in multidisciplinary environments by engaging in lifelong learning with leadership qualities, professional ethics and soft skills.

PROGRAM OUTCOMES (POs):

The Outcomes of the Bachelor of Engineering in Computer Science & Engineering (AI&ML) Programme are as follows:

- PO1:** Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
- PO2:** Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
- PO3:** Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
- PO4:** Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
- PO5:** Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
- PO6:** The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

- PO7:** Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
- PO8:** Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
- PO9:** Individual and teamwork: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
- PO10:** Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
- PO11:** Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
- PO12:** Life-long learning: Recognize the need for and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAM SPECIFIC OUTCOMES (PSOs):

- PSO1:** Ability to understand and identify problems/opportunities where CSE, AI and ML concepts can be applied and to identify the right AI& ML techniques in such contexts.
- PSO2:** Ability to perform the data engineering, designing, developing and testing the AI and ML solutions that include both hardware and software.
- PSO3:** Ability to be aware of technical solutions that are following ethical aspects aligning with social responsibilities both at designing and developmental phases of applications.

Semester wise Credit Breakdown for B.E Degree Curriculum: Batch 2023-27

Semester Course Category	First	Second	Third	Fourth	Fifth	Sixth	Seventh	Eighth	Total Credits
Basic Sciences (BSC)	08	08	03	03	--	--	--	--	22
Engineering Science Courses (ESC /PLC/ETC)	09	08	--	--	--	--	--	--	17
Humanities, Social Sciences and Management (HSMC)	02	02	--	--	03	03	--	--	10
Ability Enhancement Course (AEC)	01	02	01	01	01	--	--	--	06
Skill Development Course (SDC)	--	--	--	--	--	01	02	--	03
Universal Human Values (UHV)	--	--	02	--	--	--	--	--	02
Professional Core Courses (PCC)	--	--	11	12	12	06	04	--	45
Integrated Professional Core Course (IPCC)	--	--	04	04	03	--	04	--	15
Professional Elective Courses (PEC)	--	--	--	--	03	06	03	--	12
Open Elective Courses (OEC)	--	--	--	--	--	03	03	--	06
Internship (INT)	--	--	--	--	--	--	--	05	05
Mini Project / Project Work (PW)	--	--	--	--	--	04	--	13	17
Non-Credit Mandatory Course (NCMC)	--	--	--	Yes	Yes	Yes	--	--	--
Total Credits	20	20	21	20	22	23	16	18	160

B.E. in Computer Science and Engineering (AI & ML)**Scheme of Teaching and Batch 2023****VII SEMESTER**

Sl. No.	Subject Code	Subject	Teaching Department	Category	Credits				Total contact hours /week
					L	T	P	Total	
1	CI71	Multicore Architecture and Programming	CSE(AI ML)	IPCC	3	0	1	4	5
2	CI72	Prompt Engineering for Generative AI	CSE(AI ML)	PCC	2	1	0	3	4
3	CIE73x	Program Elective Course – 4	CSE(AI ML)	PEC	3	0	0	3	3
4	CIOE0x*	Institutional Open Elective - 2	CSE(AI ML)	IOE	3	0	0	3	3
5	CIL74	Containerization Laboratory	CSE(AI ML)	PCC	0	0	1	1	2
6	CIL75	Skill Enhancement Laboratory	CSE(AI ML)	AEC	0	1	1	2	4
Total					11	2	3	16	21

Program Elective Courses:

SL.NO	COURSE CODE	COURSE NAME	Credits				Total contact hours /week
			L	T	P	Total	
1	CIE731	Speech Recognition	3	0	0	3	3
2	CIE732	Conversational AI	3	0	0	3	3
3	CIE733	Data Storage Techniques	3	0	0	3	3
4	CIE734	AI Ethics and Sustainability	3	0	0	3	3
5	CIE735	Edge Computing	3	0	0	3	3

OPEN ELECTIVE										
SL. NO	COURSE CODE	COURSE NAME	Credits				Total contact hours /week			
			L	T	P	Total	L	T	P	S
1.	CIOE01	Web App Development	3	0	0	3	3	0	0	3
2.	CIOE02	Python Programming	3	0	0	3	3	0	0	3
3.	CIOE08	Data Base Management Systems	3	0	0	3	3	0	0	3
4.	CIOE09	Introduction to Digital Forensics	3	0	0	3	3	0	0	3
5.	CIOE05	Introduction to Artificial Intelligence	3	0	0	3	3	0	0	3

B.E. in Computer Science and Engineering (AI & ML) Scheme of Teaching and Examination 2026-27									
VIII SEMESTER									
Sl. No.	Subject Code	Subject	Teaching Department	Category	Credits				Total contact hours /week
					L	T	P	Total	
1	CIP81	Project Work	CSE(AI ML)	PW	0	0	13	13	-
2	INT82	Research / Industrial Internship	CSE(AI ML)	INT	0	0	5	5	-
Total					0	0	18	18	
3	PE83	Physical Education		NCC	All students have to register compulsorily for any one of the courses with the concerned coordinator (Yoga Teacher/ Physical Education Director/ NSS Coordinator) in the beginning of the III semester. Attending the registered course from III to VIII semesters. Qualifying is mandatory for the award of the degree				
	YO83	Yoga							
	NS83	NSS							

Nomenclature: IPCC: Integrated Professional Core Course, PCC: Professional Core Course, PEC: Professional Elective Courses, IOE: Institutional Open Elective, AEC–Ability Enhancement Courses.

L –Lecture, T – Tutorial, P- Practical/ Drawing

Note: CIE73x , where x=1,2,3,4,5,
CIOE0x*, where x=6,7,8,9,... continued from previous

Professional Elective Courses: A professional elective (PEC) course is intended to enhance the depth and breadth of educational experience in Engineering and Technology curriculum. Multidisciplinary courses that are added to supplement the latest trend and advanced technology in the selected stream of engineering. Each group provides an option to select one course out of five courses. The minimum student's strength for offering professional electives is 10.

Open Elective Courses:

Students belonging to a particular stream of Engineering and Technology are not entitled for the open electives offered by their parent department. However, they can take an elective offered by other departments, provided they satisfy the prerequisite condition, if any. Registration to open electives shall be documented under the guidance of the Proctor.

Selection of an open elective shall not be allowed if,

The candidate has studied the same course during the previous semesters of the program.

The syllabus content of open electives is similar to that of the Departmental core courses or professional electives.

A similar course, under any category, is prescribed in the higher semesters of the program.

The minimum students' strength for offering open electives is 10.

AICTE Activity Points to be earned by students admitted to BE program:

Every regular student, who is admitted to the 4-year degree program, is required to earn 100 activity points in addition to the total credits earned for the program. Students entering 4 years Degree Program through lateral entry are required to earn 75 activity points in addition to the total credits earned for the program. The activity points earned by the student shall be reflected on the students VIII Semester grade card. The activities to earn the points can be spread over the duration of the course. However, minimum prescribed duration should be fulfilled. Activity Points (non-credit) have no effect on SGPA/CGPA and shall not be considered for vertical progression. In case student fail to earn the prescribed activity points; VIII semester Grade Card shall be issued only after earning the required activity Points. Students shall be eligible for the award of degree only after the release of the VIII Semester grade card.

MULTICORE ARCHITECTURE AND PROGRAMMING	
Course Code: CI71	Credits: 3:0:1
Prerequisite: Nil	Contact Hours: 42L+14P
Course Coordinator: Dr. Mohana Kumar S	

Course Contents

Unit I

Introduction to Multi-Core Architecture: Motivation for Concurrency in Software, Parallel Computing Platforms, Parallel Computing in Microprocessors, Differentiating Multi-Core Architectures from Hyper- Threading Technology, Multi-Threading on Single-Core Versus Multi-Core Platforms Understanding Performance, Amdahl's Law, Growing Returns: Gustafson's Law. System.

Overview of Threading: Defining Threads, System View of Threads, Threading Above the Operating System, Threads Inside The OS, Threads Inside the Hardware, What Happens When A Thread Is Created, Application Programming Models and Threading, Virtual Environment: VMs And Platforms, Runtime Virtualization, System Virtualization.

- Pedagogy: Interactive Lecture
- Link: https://www.academia.edu/33065897/MULTI_CORE_ARCHITECTURES_AND_PROGRAMMING

Unit II

Fundamental Concepts of Parallel Programming: Designing for Threads, Task Decomposition, Data Decomposition, Data Flow Decomposition, Implications of Different Decompositions, Challenges You'll Face, Parallel Programming Patterns, A Motivating Problem: Error Diffusion, Analysis of The Error Diffusion Algorithm, Alternate Approach: Parallel Error Diffusion, Other Alternatives.

- Pedagogy: Core concepts and theoretical concepts PPT
- Link: https://www.academia.edu/33065897/MULTI_CORE_ARCHITECTURES_AND_PROGRAMMING

Unit III

Threading and Parallel Programming Constructs: Synchronization, Critical Sections, Deadlock, Synchronization Primitives, Semaphores, Locks, Condition Variables, Messages, Flow Control- Based Concepts, Fence, Barrier, Implementation-Dependent Threading Features. Threading API's: Threading API's For Microsoft Windows, Win32/MFC Thread API's, Threading API's For Microsoft. NET Framework, Creating Threads, Managing Threads, Thread Pools, Thread Synchronization, POSIX Threads, Creating Threads, Managing Threads, Thread Synchronization, Signalling, Compilation and Linking.

- Pedagogy: Coding, Lab Hands-on Exercises
- Link: https://www.academia.edu/33065897/MULTI_CORE_ARCHITECTURES_AND_PROGRAMMING

Unit IV

Open MP: A Portable Solution For Threading: Challenges In Threading A Loop, Loop-Carried Dependence, Data-Race Conditions, Managing Shared And Private Data, Loop Scheduling And Portioning, Effective Use Of Reductions, Minimizing Threading Overhead, Work-Sharing Sections, Performance-Oriented Programming, Using Barrier And No Wait, Interleaving Single-Thread And Multi-Thread Execution, Data Copy-In And Copy-Out, Protecting Updates Of Shared Variables, Intel Task Queuing Extension to Open MP, OpenMP Library Functions, OpenMP Environment Variables, Compilation, Debugging, Performance.

- Pedagogy: Demo, Case Study
- Link: https://www.academia.edu/33065897/MULTI_CORE_ARCHITECTURES_AND_PROGRAMMING

Unit V

Solutions To Common Parallel Programming Problems: Too Many Threads, Data Races, Deadlocks, And Live Locks, Deadlock, Heavily Contended Locks, Priority Inversion, Solutions For Heavily Contended Locks, Non-Blocking Algorithms, ABA Problem, Cache Line Ping-Pong, Memory Reclamation Problem, Recommendations, Thread-Safe Functions And Libraries, Memory Issues, Bandwidth, Working In The Cache, Memory Contention, Cache-Related Issues, False Sharing, Memory Consistency, Current IA-32 Architecture, Itanium Architecture, High-Level Languages, Avoiding Pipeline Stalls On IA-32, Data Organization For High Performance.

- Pedagogy: Demo, Case Study
- Link: https://www.academia.edu/33065897/MULTI_CORE_ARCHITECTURES_AND_PROGRAMMING

Text Books:

1. Multicore Programming, Increased Performance through Software Multi-threading by Shameem Akhter and Jason Roberts, Intel Press, 2006
2. Hennessey and Patterson: “Computer Architecture A Quantitative Approach”, 4th Edition, Elsevier, 2012.

Reference Book:

1. Kai Hwang, Naresh Jotwani: Advanced Computer Architecture - Parallelism, Scalability, Programmability, 2nd Edition, Tata McGraw Hill, 2011.

Course Outcomes (COs):

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

1. Understand the concepts of parallel programming, Amdahl's law and threading. (PO1,2,3,5 PSO1)
2. Describe the challenges involved in designing multi threaded applications. (PO1,2,3,5 PSO1)
3. Apply OpenMP directives and techniques to develop parallel programs by managing threads. (PO1,2,3,5 PSO1)
4. Analyze synchronization , deadlock and memory related issues with respect to parallel processing. (PO1,2,3,5 PSO1)
5. Design and develop multithreaded applications to exploit multicore architecture. (PO1,2, 3,5,PSO1)

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tool	Marks	Course outcomes addressed
Internal test-I	30	CO1, CO2, CO3
Internal test-II	30	CO3, CO4, CO5
The average of the two internal tests shall be taken for 30 marks.		
Other components		
Lab component	20	CO1, CO2, CO3
The Final CIE out of 50 Marks = Average of two CIE tests for 30 Marks + Marks scored in Assignments		
Semester End Examination (SEE)	100	CO1, CO2, CO3, CO4, CO5

PROMPT ENGINEERING FOR GENERATIVE AI	
Course Code: CI72	Credits: 2:1:0
Prerequisite: Nil	Contact Hours: 28L+14T
Course Coordinator: Dr. A N Ramya Shree, Dr. Nithya N	

Course Contents

Unit 1

Generative AI: Definition and Fundamentals, History and Evolution from AI to Generative AI, Traditional AI to Generative AI, Autoencoders: The Information Bottleneck, Latent Variables, Autoencoder Architecture, Types of Autoencoders. GAN: The Generative Model, The Discriminative Model. Attention Models: Encoder-Decoder Paradigm, Attention to Sequence Models. Transformers: Encoder Layer, Positional Encoding, Residual Connections, Decoder Layer, Linear Layer and SoftMax, Training, Inference, Loss Function.

- Pedagogy: Demonstration of attention mechanism
- Link: https://github.com/soloshun/llm-from-scratch/blob/master/src/part06_attention_mechanism.py

Unit II

LLM, Text Representation and GPT Models: Large Language Models: Introduction to Large Language Models (LLMs), What Are LLMs, Applications of LLMs, How LLMs See the World: Hallucinations, How LLMs Process Text, LLM Tokenization, How Tokenizers Prepare Inputs to the Language Model, How Does the Tokenizer Break Down Text, Word vs. Subword vs. Character vs. Byte Tokens, Comparing Trained LLM Tokenizers, Tokenizer Properties, Token Embeddings, Text Embeddings, Word Embeddings

- Pedagogy: Demonstration of LLM, Applications
- Link: <https://github.com/rasbt/LLMs-from-scratch/tree/main/ch04>

Unit III

Prompt Engineering: Introduction to Prompt Engineering , The Basic Ingredients of a Prompt , Instruction-Based Prompting , Advanced Prompt Engineering , The Potential Complexity of a Prompt , In-Context Learning: Providing Examples , Chain Prompting: Breaking up the Problem , Reasoning with Generative Models , Chain-of-Thought: Think Before Answering , Self-Consistency: Sampling Outputs, Tree-of-Thought: Exploring Intermediate Steps , Output Verification , Providing Examples , Grammar: Constrained Sampling. Prompt design and Optimization.

- Pedagogy: Experiment with prompts
- Link: <https://www.promptingguide.ai/introduction/basics>

Unit IV

LangChain: Introduction to LangChain, Chat Models, Streaming Chat Models, Creating Multiple LLM Generations, LangChain Prompt Templates, LangChain Expression Language (LCEL), Using PromptTemplate with Chat Models, Output Parsers, LangChain Evals, OpenAI Function Calling, Parallel Function Calling, Function Calling in LangChain, Extracting Data with LangChain, Query Planning, Creating Few-Shot Prompt Templates, Limitations with Few-Shot Examples, Saving and Loading LLM Prompts, Prompt Chaining, Reasoning with language agents.

- Pedagogy: Demonstration of LangChain Applications
- Link: https://github.com/benman1/generative_ai_with_langchain

Unit V

Retrieval Augmented Generation (RAG): Key Concepts and Components RAG, How It Improves Question Answering, RAG Architecture, Building the Retrieval System, Embeddings and Vector Databases for Retrieval in RAG, RAG Data Ingestion Pipeline, RAG Pipeline Implementation: Preprocessing PDF documents, Data Ingestion Pipeline Implementation, Generation Component Implementation, Impact of Text Splitting on RAG, Challenges of Retrieval-Augmented Generation.

- Pedagogy: Building RAG application
- Link: https://github.com/langchain-ai/rag-from-scratch/blob/main/rag_from_scratch_10_and_11.ipynb

Text Books

1. John Atkinson and Abutridy, Large Language Models Concepts, Techniques and Applications, CRC Press, 2025
2. John Berryman and Albert Ziegler, Prompt Engineering for LLMs, The Art and Science of Building Large Language Model–Based Applications, O'Reilly, 2025.
3. James Phoenix and Mike Taylor, Prompt Engineering for Generative AI Future-Proof Inputs for Reliable AI Outputs at Scale, O'Reilly, 2024.
4. Alammam & Maarten Grootendorst, Hands-On Large Language Models Language Understanding and Generation, Jay, O'Reilly, 2024.
5. Mehdi Allahyari, Angelina Yang, "A Practical Approach to Retrieval Augmented Generation Systems", Maven, 2023.

Course Outcomes (COs):

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

1. Develop a comprehensive understanding of the fundamental concepts, architecture, tokenization mechanisms, embeddings, and operational principles of Large Language Models (LLMs). (PO-1,2,3,4,5,12, PSO-1,2).
2. Analyze the pre-processing techniques, token representations, embedding methods, and contextual language representations used in modern natural language processing systems. (PO-1,2,3,4,5, PSO-1,2).

3. Apply prompt engineering strategies and reasoning techniques to effectively control, optimize, and evaluate text generation using state-of-the-art generative language models. (PO-1,2,3,4,5,6,12, PSO-1,2, 3).
4. Design and develop LLM-based applications using Lang Chain by integrating prompt templates, function calling, document processing, retrieval mechanisms, and workflow orchestration. (PO-1,2,3,4,5,6, 12, PSO-1,2,3).
5. Evaluate and implement fine-tuning methodologies, instruction tuning, few-shot learning, named-entity recognition, and performance evaluation techniques for adapting and assessing generative AI models. (PO-1,2,3,4,5,12 PSO-1,2,3).

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tool	Marks	Course outcomes addressed
Internal test-I	30	CO1, CO2, CO3
Internal test-II	30	CO3, CO4, CO5
The average of the two internal tests shall be taken for 30 marks.		
Other components		
Assignment 1	10	CO1, CO2, CO3
Assignment 2	10	CO1, CO2, CO3, CO4, CO5
The Final CIE out of 50 Marks = Average of two CIE tests for 30 Marks + Marks scored in Assignments.		
Semester End Examination (SEE)	100	CO1, CO2, CO3, CO4, CO5

SPEECH RECOGNITION	
Course Code: CIE731	Credits: 3:0:0
Prerequisite: Nil	Contact Hours: 42 L
Course Coordinator: Dr. Naveen N C	

Course Contents

Unit I

Fundamentals: The Speech Signal: Production, Perception and Acoustic-Phonetic Characterization, The Speech-Production Process, Representing Speech in the Time and Frequency Domains, Speech Sounds and Features, Approaches to Automatic Speech Recognition by Machine

- Pedagogy: Chalk and board, Problem based learning, Visual Aids
- Link: <https://youtu.be/cnkVeKtaTjk>

Unit II

Signal Processing and Analysis Methods for Speech Recognition: The Bank-of-Filters, Front-End Processor, Linear Predictive Coding Model for Speech Recognition, Vector Quantization, Auditory-Based Spectral Analysis Models

- Pedagogy: Chalk and board, Problem based learning, Visual Aids
- Link: <https://youtu.be/pmvfaNBr6wY>

Unit III

Pattern-Comparison Techniques: Speech (Endpoint) Detection, Distortion Measures-Mathematical Considerations, Distortion Measures-Perceptual Considerations, Spectral-Distortion Measures, Time Alignment and Normalization

- Pedagogy: Chalk and board, Problem based learning, Visual Aids
- Link: <https://youtu.be/pmvfaNBr6wY>

Unit IV

Speech Recognition System Design and Implementation Issues: Application of Source-Coding Techniques to Recognition, Template Training Methods, Performance Analysis and Recognition Enhancements, Template Adaptation to New Talkers, Discriminative Methods in Speech Recognition, Speech Recognition in Adverse Environments

- Pedagogy: Chalk and board, Problem based learning, Case Studies
- Link: https://youtu.be/Sz_8Jqw_Yn8

Unit V

Large Vocabulary Continuous Speech Recognition: Subword Speech Units, Subword Unit Models Based on HMMs, Training of Subword Units, Language Models for Large Vocabulary Speech Recognition, Perplexity of the Language Model, Overall Recognition System Based on Subword Units, Context-Dependent Subword Units, Creation of Vocabulary-Independent Units, Semantic Postprocessor for Recognition

- Pedagogy: Chalk and board, Problem based learning, Tool Demo
- Link: https://youtu.be/Sz_8Jqw_Yn8

Text Books:

1. Fundamentals of Speech Recognition, L. Rabiner, Biing-Hwang Juang and B. Yegnanarayana, Pearson Education Inc.2009

Reference Book:

1. Daniel Jurafsky and James H Martin, —Speech and Language Processing: An introduction to Natural Language Processing, Computational Linguistics and Speech Recognition, Prentice Hall, 2nd Edition, 2008.
2. Benesty, M. M. Sondhi and Y. Huang, “Handbook of speech processing”, Springer, 2008

Course Outcomes (COs):

At the end of the course, the students should be able to:

1. Understand the fundamentals of speech production, perception, and acoustic-phonetic features. (PO- 1,2,4, PSO-1)
2. Apply signal processing techniques such as LPC, filter banks, and vector quantization in speech analysis. (PO-1,2,4, PSO-1)
3. Analyze and implement pattern recognition techniques including DTW, distortion measures, and speech endpoint detection (PO-1,2,3,5, PSO-1)
4. Design and evaluate speech recognition systems with template training, adaptation, and robust methods for adverse environments. (PO- 3,4,5, 10) (PSO-2)
5. Investigate large vocabulary continuous speech recognition models using subword units, HMMs, and language modeling techniques. (PO-3,4,5,10, PSO-2).

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tools	Marks	Course Outcomes (COs) addressed
Internal Test-I	30	CO1, CO2
Internal Test-II	30	CO3, CO4, CO5
Average of the two CIE shall be taken for 30 marks		
Other Components		
Project	20	CO1, CO2, CO3,CO4,CO5
The Final CIE out of 50 Marks = Average of two CIE tests for 30 Marks + Marks scored in Project		
Semester End Examination (SEE)	100	CO1, CO2, CO3 CO4, CO5

CONVERSATIONAL AI	
Course Code: CIE732	Credits: 3:0:0
Prerequisite: Nil	Contact Hours: 42L
Course Coordinator Dr. Sini Anna Alex / Dr. Nayana G. Bhat	

Course Contents

Unit I

Introduction to Conversational AI, Principles of dialogue – common ground, sub dialogues, Gricean principles of conversation, Computational models of dialogue systems, Chatbots architectures – Rule-based and Corpus based, Case study: Sounding board.

- Pedagogy: Chalk and Talk
- Link: nptel.ac.in/courses/106102220

Unit II

Architecture for dialogue systems: Pipelines behind common assistant programs, collaborative problem-solving model, dialogue acts. cognitive architectures, Question Answering: Sources of knowledge, Case Study: IBM's Deep Q/A approach.

- Pedagogy: PowerPoint Presentations
- Link: nptel.ac.in/courses/106102220

Unit III

Dialog Management and System Evaluation, Dialog Manager Architectures, Natural Language Generation, evaluation of performance, reward propagation. Case study: Social chatbot evaluation.

- Pedagogy: AI Tools Demonstration
- Link: nptel.ac.in/courses/106102220

Unit IV

Seq2seq, transformer-based models, end-to-end learning, context handling, evaluation, Introduction to Microsoft Bot, Microsoft Bot Framework, Developing chatbot, HR Bot Programming.

- Pedagogy: Casestudy and AI tool Enabled Learning
- Link: nptel.ac.in/courses/106102220

Unit V

RASA and Google Dialog flow, Introduction to QnA Maker, Introduction to LUIS, Introduction to RASA, RASA Core, RASA NLU, Introduction to Dialog flow. Integration with Third party APIs, Connecting to an Enterprise Data Store, Deployment to Cloud.

- Pedagogy: Casestudy and AI tool Enabled Learning
- Link: nptel.ac.in/courses/106102220

Text Books:

1. Michael McTear, Conversational Ai: Dialogue Systems, Conversational Agents, and chat bots (Synthesis Lectures on Human Language Technologies), Morgan & Claypool, 2020.
2. Andrew Freed, "Conversational AI: Chatbots that work, Manning Publications, 2021
3. Janarthnam, Srini. Hands-on chatbots and conversational UI development: Build chatbots and voice user interfaces with Chatfuel, Dialogflow, Microsoft Bot Framework, Twilio, and Alexa Skills. Packt Publishing Ltd, 2017.

References:

1. Singh, Abhishek, Karthik Ramasubramanian, and Shrey Shivam. Building an Enterprise Chatbot Springer, Apress, 2019
2. Kelly III, John E., and Steve Hamm. Smart machines: IBM&Watson and the era of cognitive computing, Columbia University Press, 2013.

Course Outcomes (COs):

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

1. Understand foundational concepts, architectures, and trends in conversational AI. (PO 1,2, 3,4, PSO1,2,3)
2. Design and develop dialogue systems using rule-based, data-driven, and neural approaches (PO-1,2,3,5, PSO-1,2,3)
3. Deploy and evaluate chatbots/applications leveraging state-of-art tools, datasets, and metrics. (PO-1,2,3,5,8, PSO-1,2,3)
4. Describe the features and components of chatbots on platforms like WordPress and troubleshooting issues. (PO-1,2,3,5,6,8, PSO-1,2,3)
5. Explore advanced chatbot development topics like Microsoft Bot Framework, RASA, Google Dialogflow, and integration with third-party APIs. (PO-1,2,3,5,6,8, PSO-1,2,3)

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tools	Marks	Course Outcomes (COs) addressed
Internal Test-I	30	CO1, CO2
Internal Test-II	30	CO3, CO4, CO5
Average of the two CIE shall be taken for 30 marks		
Other Components		
Case study	10	CO1, CO2, CO3
Assignment	10	CO4, CO5
The Final CIE out of 50 Marks = Average of two CIE tests for 30 Marks+ Marks scored in Case Study + Marks scored in Assignment.		
Semester End Examination (SEE)	100	CO1, CO2, CO3 CO4, CO5

DATA STORAGE TECHNIQUES	
Course Code: CIE733	Credits: 3:0:0
Prerequisite: Nil	Contact Hours: 42L
Course Coordinator: Mrs. Akshatha G C	

Course Contents

Unit I

Introduction to Information Storage: Digital data and its types, Information storage, Key characteristics of data center and Evolution of computing platforms. Information Lifecycle Management. **Third Platform Technologies:** Cloud computing and its essential characteristics, Cloud services and cloud deployment models, Big data analytics, Social networking and mobile computing, Characteristics of third platform infrastructure and Imperatives for third platform transformation.

Data Center Environment: Building blocks of a data center, Compute systems and compute virtualization and Software-defined data center.

- Pedagogy: Lecture PPTs, Chalk and Talk, Case Study based Assignments
- Link: <https://nptel.ac.in/courses/106108058>

Unit II

Intelligent storage systems and RAID: Components of an intelligent storage system, Components, addressing, and performance of hard disk drives and solid-state drives, RAID, Types of intelligent storage systems, Scale-up and scale-out storage Architecture. Block-Based Storage System, File-Based Storage System, Object-Based and Unified Storage.

- Pedagogy: Lecture PPTs, Chalk and Talk, Case Study based Assignments
- Link: <https://nptel.ac.in/courses/106108058>

Unit III

Fibre Channel SAN: Software-defined networking, FC SAN components and architecture, FC SAN topologies, link aggregation, and zoning, Virtualization in FC SAN environment. Internet Protocol SAN: iSCSI protocol, network components, and connectivity, Link aggregation, switch aggregation, and VLAN, FCIP protocol, connectivity, and configuration. Fibre Channel over Ethernet SAN: Components of FCoE SAN, FCoE SAN connectivity, Converged Enhanced Ethernet, FCoE architecture.

- Pedagogy: Lecture PPTs, Chalk and Talk, Case Study based Assignments
- Link: <https://nptel.ac.in/courses/106108058>

Unit IV

Backup archive and replication: Introduction to Business Continuity, Backup architecture, Backup targets and methods, Data deduplication, Cloud-based and mobile device backup, Data archive, Uses of replication and its characteristics, Compute based, storage-based, and network-based replication, Data migration, Disaster Recovery as a Service (DRaaS).

- Pedagogy: Lecture PPTs, Chalk and Talk, Case Study based Assignments
- Link: <https://nptel.ac.in/courses/106108058>

Unit V

Secure storage Infrastructure: Information security goals, Storage security domains, Threats to a storage infrastructure, Security controls to protect a storage infrastructure, Governance, risk, and compliance, Storage infrastructure management functions, Storage infrastructure management processes.

- Pedagogy: Lecture PPTs, Chalk and Talk, Case Study based Assignments
- Link: <https://nptel.ac.in/courses/106108058>

Text Books:

1. EMC Corporation, Information Storage and Management, Wiley, India.

References:

1. Jon Tate, Pall Beck, Hector Hugo Ibarra, Shanmuganathan Kumaravel and Libor Miklas, Introduction to Storage Area Networks, Ninth Edition, IBM - Redbooks, December 2017.
2. Ulf Troppens, Rainer Erkens, Wolfgang Mueller-Friedt, Rainer Wolafka, Nils Haustein, Storage Networks Explained, Second Edition, Wiley, 2009

Course Outcomes (COs):

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

1. Explain the fundamentals of information storage management and distinguish various models of cloud infrastructure services and deployment. (PO1,2,3,5 PSO-2).
2. Apply advanced intelligent storage system concepts and RAID techniques to address storage management requirements. (PO-1,2,3,4,5 PSO-2).
3. Design storage networking architectures, including SAN and associated storage subsystems, using virtualization technologies. (PO1,2,3,4,5 PSO-2).
4. Describe disaster recovery and remote replication technologies for effective information storage management. (PO1,2,3,5,6 PSO-2).
5. Analyze security requirements and recommend appropriate security measures for information storage management. (PO1,2,3,5,6 PSO-2)

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tool	Marks	Course outcomes addressed
Internal test-I (CIE-I)	30	CO1, CO2, CO3
Internal test-II (CIE-II)	30	CO3, CO4, CO5
Average of the two internal tests shall be taken for 30 marks.		
Other components		
Case study-based Assignments	10	CO1, CO2, CO3
	10	CO3, CO4, CO5
The Final CIE out of 50 Marks = Average of two CIE tests for 30 Marks+ Marks scored in Case Study based Assignment.		
Semester End Examination (SEE)	100	CO1, CO2, CO3, CO4, CO5

AI ETHICS AND SUSTAINABILITY	
Course Code: CIE734	Credits: 3:0:0
Pre – requisites: Nil	Contact Hours: 42L
Course Coordinator: Dr. Rakesh Kalshetty	

Course Contents

Unit 1

Foundations of Artificial Intelligence and Ethics: Introduction to artificial intelligence and intelligent systems; motivation for responsible artificial intelligence; foundations of AI; background of AI; autonomy; adaptability; interaction.

- Pedagogy: Interactive Lectures and Classroom Discussions, Concepts based learning through PPTs.
- Link: <https://www.sciencedirect.com/science/article/abs/pii/S3050790125000075>

Unit II

Ethical Decision-Making, Responsibility and Design for Values: Ethical decision-making in AI; ethical theories; values; ethics in practice; implementing ethical reasoning; responsible research and innovation; accountability; responsibility; transparency; and design for values.

- Pedagogy: Interactive Lectures and Classroom Discussions, Concepts based learning through PPTs.
- Link: https://onlinecourses.nptel.ac.in/e-learning/preview/noc26_cs145

Unit III

Trustworthy AI, Alignment, Moral Machines and Long-Term Ethics: Ethical action in AI systems; approaches to ethical reasoning by AI; designing artificial moral agents; levels of ethical behavior; ethical status of AI systems.

- Pedagogy: Interactive Lectures and Classroom Discussions, Concepts based learning through PPTs.
- Link: <https://www.coursera.org/learn/ai-ethics-and-policy>

Unit IV

Governance and Responsible AI in Practice: Ensuring Responsible AI in Practice; Governance for Responsible AI; Codes of Conduct; Inclusion and Diversity; The AI Narrative.

- Pedagogy: Interactive Lectures and Classroom Discussions, Concepts based learning through PPTs, Case Study, Demo and Student Presentation.
- Link: <https://link.springer.com/article/10.1007/s00146-026-02910-4>

Unit V

AI, Society and Socio-Economic Impacts: AI and society; broader societal implications of AI; super-intelligence; responsible AI in the social context; socio-economic impacts of AI; and public policy issues related to advanced AI systems.

- Pedagogy: Interactive Lectures and Classroom Discussions, Concepts based learning through PPTs, Case Study, Demo and Student Presentation.
- Link: <https://nptel.ac.in/courses/109106184>

Text Books

1. Responsible Artificial Intelligence – Virginia Dignum, Springer, 2019.
2. Ethics of Artificial Intelligence – S. Matthew Liao, Oxford University Press, 2020.

Course Outcomes (COs):

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

1. Understand the foundations of artificial intelligence and the need for ethical and responsible AI systems.(PO-1,2,3,5,PSO-1,2)
2. Analyze ethical theories, values, responsibility, fairness, accountability, and privacy in AI design and deployment. (PO-1,2,3,5,PSO-1,2)
3. Examine governance, codes of conduct, inclusion, and responsible AI practices in real-world systems. (PO-1,2,3,5,PSO-1,2,3)
4. Evaluate the societal and socio-economic implications of AI technologies. (PO-1,2,3,PSO-1,2,3)
5. Explore trustworthy AI, alignment, moral machines, and long-term ethical challenges in AI. (PO-1,2,3,4,PSO-1,2,3)

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tools	Marks	Course Outcomes (COs) addressed
Internal Test-I	30	CO1, CO2
Internal Test-II	30	CO3, CO4, CO5
Average of the two CIE shall be taken for 30 marks		
Other Components		
Project	20	CO1, CO2, CO3,CO4,CO5
The Final CIE out of 50 Marks = Average of two CIE tests for 30 Marks + Marks scored in Case Study +Marks scored in Assignment		
Semester End Examination (SEE)	100	CO1, CO2, CO3 CO4, CO5

EDGE COMPUTING	
Course Code: CIE735	Credits: 3:0:0
Prerequisite: Nil	Contact Hours: 42L
Course Coordinator: Ms. Kavya Natikar & Ms. Bhagyashree Hosmani	

Course Contents

Unit I

Introduction to Edge Computing: Edge computing purpose and definition, key components of edge ecosystem, Edge computing devices, Edge vs Fog Computing vs Cloud computing, Working of edge computing. Edge computing use cases and examples: video surveillance, smart cities, autonomous vehicles.

- Pedagogy: Interactive Lectures and Classroom Discussions, Interactive Lectures and Classroom Discussions.
- Link: https://onlinecourses.nptel.ac.in/noc23_cs65/educba.com/edge-computing-architecture/

Unit II

Edge Computing platforms and frameworks: Edge hardware architectures: processors, storage, and memory, hardware accelerators in edge computing: Use of Hardware accelerators like Intel Compute Stick/Google Coral with companion to Raspberry Pi or JetsonEdge platforms: Architectures, Virtualization, containersEdge computing popular frameworks: Microsoft Azure IoT Edge or Google cloud IoT Edge.

- Pedagogy: Interactive Lectures and Classroom Discussions, Interactive Lectures and Classroom Discussions.
- Link: https://onlinecourses.nptel.ac.in/noc23_cs65/educba.com/edge-computing-architecture/

Unit III

Edge Networking: Edge routing and networking: TCP/IP network functions at the edge, Edge-level network security, Software-defined networking.

Communication Models: Edge, Fog, M2M: Building M2M and WSN network (HTTP/MQTT) – Topologies and applications.

- Pedagogy: Interactive Lectures and Classroom Discussions, Interactive Lectures and Classroom Discussions.
- Link: https://onlinecourses.nptel.ac.in/noc23_cs65/educba.com/edge-computing-architecture/

Unit IV

IoT and Edge Security: Introduction to IoT and Edge security: cybersecurity, physical and hardware security, Edge security best practices, Threats and vulnerabilities related to edge computing.**Case study:** Anomaly Detection in Edge Computing.

- Pedagogy: Interactive Lectures and Classroom Discussions, Interactive Lectures and Classroom Discussions.
- Link: https://onlinecourses.nptel.ac.in/noc23_cs65/educba.com/edge-computing-architecture/

Unit V

Edge Analytics: Basic data analytics in Edge IoT, Machine learning in Edge IoT, Edge AI. Edge Analytics applications and examples: Deep Learning at the Edge, Exploiting Edge Computing in Health Monitoring, Edge Computing Model for Evolving Smart Transportation, Deployment of deep model on edge devices using Edge Impulse platform.

- Pedagogy: Interactive Lectures and Classroom Discussions, Interactive Lectures and Classroom Discussions.
- Link: https://onlinecourses.nptel.ac.in/noc23_cs65/educba.com/edge-computing-architecture/

Text Books:

1. Perry Lea , “IoT and Edge Computing for Architects”, Packt Publishing, ISBN: 9781839214806, March 2020.
2. Rajkumar Buyya, Satish Srirama, “Fog and Edge Computing: Principles and Paradigms”, Wiley publication, ISBN:9781119524984, January 2019.

Reference Books:

1. David Jensen, “Beginning Azure IoT Edge Computing: Extending the Cloud to the Intelligent Edge, Springer India, ISBN-13 978-1484275504, January 2022.
2. Pethuru Raj and Anupama C. Raman, "The Internet of Things: Enabling Technologies, Platforms, and Use Cases", CRC Press, ISBN 9781032509266, June 2022.

Course Outcomes (COs):

At the end of the course, students should be able to:

1. Describe the need for new computing paradigms and compare the edge computing with existing paradigms. (PO-1, PSO-2).
2. Analyze the hardware and software-based components in edge computing architectures and platforms. (PO-1,2,3 PSO-2).
3. Demonstrate Edge networking requirements and components for various communication models. (PO-1,2,3 PSO-2).
4. Analyze data/application requirements related to security in edge computing. (PO-1,3,8, PSO-2,3)
5. Design edge analytics models using edge AI/ML methods (PO-1,2,3,4 PSO-2).

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tools	Marks	Course Outcomes (COs) addressed
Internal Test-I	30	CO1, CO2
Internal Test-II	30	CO3, CO4, CO5
Average of the two CIE shall be taken for 30 marks		
Other Components		
Case study	10	CO1, CO2, CO3
Assignment	10	CO4, CO5
The Final CIE out of 50 Marks = Average of two CIE tests for 30 Marks+ Marks scored in Case Study +Marks scored in Assignment		
Semester End Examination (SEE)	100	CO1, CO2, CO3, CO4, CO5

CONTAINERIZATION LABORATORY	
Course Code: CIL74	Credits: 0:0:1
Prerequisite: Nil	Contact Hours: 14P
Course Coordinator: Mrs. Shobha. K	

Course Contents

Sl. No	Table of Contents
1	Create a Python program that prints a greeting message. Containerize the application using Docker by writing an appropriate Dockerfile, and show the commands required to build and deploy the container.
2	Create a Python program that does Arithmetic Calculations. Containerize the application using Docker by writing an appropriate Dockerfile, and show the commands required to build and deploy the container.
3	Create a Python program that does CPU Delay for CPU Intensive Task. Containerize the application using Docker by writing an appropriate Dockerfile, and show the commands required to build and deploy the container.
4	Create a Python program that find the Factorial of a number. Containerize the application using Docker by writing an appropriate Dockerfile, and show the commands required to build and deploy the container
5	Create a Python program to display a bar chart using sample data. Containerize the application using Docker by writing an appropriate Dockerfile, and show the commands required to build and deploy the container.
6	Create a Python program that prints Environment Variable. Containerize the application using Docker by writing an appropriate Dockerfile, and show the commands required to build and deploy the container
7	Create a Python program that performs a simple ping simulation for a given host. Containerize this application using Docker by writing a Dockerfile and show the commands to build and run the container.
8	Create a Python program that prints the given number is prime or not. Containerize this application using Docker by writing a Dockerfile and show the commands to build and run the container.
9	Demonstrate the installation of Minikube and kubectl on Ubuntu Linux. Show the commands to start Minikube and check node and pod status in the Kubernetes cluster.
10	Create a simple Ubuntu-based container that runs a custom shell script. Write a Dockerfile to containerize the script and show the commands required to build and execute the container offline.
11	Create a Python program that performs Celsius ↔ Fahrenheit temperature conversion. Containerize the application using Docker by writing a Dockerfile and show the commands to build and run the container.

References:

1. Chris Richardson: Microservices Patterns with Examples in Java, Manning Publications Co., First Edition, 2019.

2. Sourabh Sharma: Mastering Microservices with Java 9, Packt Publishing Ltd, Second Edition, 2017.
3. Amazon Web Services in Action – Michael Wittig & Andreas Wittig (Manning, 3rd Ed.)2.
4. AWS Certified Solutions Architect Official Study Guide – Ben Piper & David Clinton (Sybex).
5. Pro Git – Scott Chacon & Ben Straub (Apress, 2nd Ed., free online at git-scm.com/book).
6. Kubernetes Up & Running – Kelsey Hightower, Brendan Burns, Joe Beda (O’Reilly, 3rd Ed.).
7. Kubernetes Patterns – Bilgin Ibryam & Roland Huß (O’Reilly).
8. Docker Deep Dive – Nigel Poulton (Leanpub, latest edition).
9. Moises Macero: Learn Microservices with Spring Boot: A Practical Approach to RESTful Services using RabbitMQ, Eureka, Ribbon, Zuul and Cucumber, A Press, First Edition, 2017.

Course Outcomes (COs):

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

1. Explain the concepts of virtualization and the fundamentals of load balancing in cloud environments. (PO-1,2,3,4,5,PSO-1,2).
2. Demonstrate proficiency in version control using Git by applying core commands and workflows for tracking and managing source code. (PO-1,2,3,4,5,PSO-1,2).
3. Deploy and manage containerized applications using Docker, Kubernetes including image handling and command-line operations. (PO-1,2,3,4,5,PSO-1,2).

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tools	Marks	Course Outcomes (COs) addressed
Lab Test-I	10	CO1, CO2
Lab Test-II	10	CO3
Weekly Evaluation + Lab Record	30	CO1, CO2, CO3
The Final CIE out of 50 Marks = Marks of Lab Record + Marks scored in Lab Test-I + Marks scored in Lab Test-II		
Semester End Examination (SEE)	50	CO1, CO2, CO3

SKILL ENHANCEMENT LABORATORY- GENERATIVE AI	
Course Code: CIL75	Credits: 0:1:1
Prerequisite: Nil	Contact Hours: 14T+14P
Course Coordinator:	

Course Contents

- Introduction to Generative AI
- Image and Video Generation Tools
- Copilot, Claude, BRAD, Educational generative AI.
- Preliminary Project Based Learning
- Large Language Models Prompt Engineering
- DNNs Tensor Flow
- Word Embedding's
- Hugging Face Models
- LangChain - Basics of Chains
- LangChain Model IO, LangChain - RAG
- LangChain – Memory, LangChain - Agents
- Recurrent neural network, Long Short-Term Memory
- Transformers
- Project Based Learning

The list of Laboratory Exercises formulated covering all the above topics are mentioned in the table below:

Expt. No.	Laboratory Exercises
1.	Using Generative AI tools, create a short story or poem (text generation) based on a chosen theme, then generate an image (text-to-image generation) and a 15–30 second sound or music clip (AI sound generation) that represent the same theme, and analyzing how well the text, image, and sound outputs align with each other while discussing the strengths, limitations, and ethical challenges of multimodal generative AI.
2.	Use word embeddings to improve prompts for Generative AI model. Retrieve similar words using word embeddings. Use the similar words to enrich a GenAI prompt. Use the AI model to generate responses for the original and enriched prompts. Compare the outputs in terms of detail and relevance.
3.	Develop a predictive model for academic performance. Use TensorFlow to build a DNN that predicts whether a student will pass or fail based on their study hours and attendance percentage.
4.	Your department wants a tool to summarize long academic reports into concise summaries for easier review. Use a pre-trained Hugging Face summarization model to generate a short summary from a given academic report.
5.	i. Explore pretrained word vectors. Explore word relationships using vector arithmetic. Perform arithmetic operations and analyze results ii. Use dimensionality reduction (e.g., PCA or t-SNE) to visualize word embeddings for the above pretrained word vectors. Select 10 words from a specific domain (e.g., sports, technology) and visualize their embeddings. Analyze clusters and relationships. Generate contextually rich outputs using embeddings. Write a program to generate 5 semantically similar words for a given input.
6.	Train a custom Word2Vec model on a small dataset. Train embeddings on a domain-specific corpus (e.g., legal, medical) and analyze how embeddings capture domain-specific semantics.
7.	Install langchain , cohere (for key) and langchain-community. Using the API key, load a text document from the your google drive . Create a prompt template to display the output in a particular manner.
8.	Use a pre-trained Hugging Face model to analyze sentiment in text. i. Assume a real-world application ii. Load the sentiment analysis pipeline. iii. Analyze the sentiment by giving sentences to input.
9.	Create a LangChain application that takes the name of an institution as input and extracts structured information from Wikipedia. Use Pydantic to define the schema for the desired output and implement a custom output parser to enforce the schema. Your task is to invoke the chain and fetch results about the institution with the following details: i. The founder of the institution

	ii. The year it was founded iii. The current branches of the institution iv. The number of employees working in it v. A concise 4-line summary of the institution
10.	Download the official Indian Penal Code (IPC) document in text or PDF format. Using LangChain (or a similar framework) and a language model of your choice (e.g., Cohere, OpenAI, or Hugging Face models), build a chatbot that can interact with the IPC. The chatbot should allow users to ask questions about specific sections, laws, or provisions, and provide relevant answers. Implement the following steps: i. Load and preprocess the IPC document (text extraction if PDF). ii. Split the document into smaller chunks for efficient retrieval. iii. Create an embedding-based vector store to enable semantic search. iv. Integrate a retrieval-based QA pipeline with a language model. v. Deploy a chatbot interface where users can query the IPC and receive conversational.
11.	Write a PyTorch program that performs time series forecasting using an RNN. The program should complete the following steps: i. Generate synthetic time series data using a sine function with added noise and visualize the data using matplotlib. ii. Prepare the dataset by creating input–output pairs using a fixed time step window (sequence length). iii. Define an RNN model using PyTorch, including: a recurrent layer (nn.RNN) a fully connected layer (nn.Linear) for prediction. iv. Train the model using the Mean Squared Error (MSE) loss function and the Adam optimizer. Use the trained model to make predictions on the time series data. v. Print the loss value during training to monitor progress. vi. Visualize the original time series data along with the predicted values to assess the model’s performance.
12.	Build and train an LSTM-based text generation model using a dataset of your choice (e.g., Shakespeare’s plays, song lyrics, or news articles). Preprocess the text into sequences, train the LSTM to predict the next character or word, and then generate new text by providing a starting seed phrase. Experiment with hyper-parameters such as sequence length, number of LSTM units, and training epochs.

Text Book:

1. Venkata Reddy Konasani, Shailendra Kadre. Machine Learning and Deep Learning Using Python and TensorFlow 1st Edition, McGraw Hill, 2021
2. Raghav Bali. Generative AI with Python and TensorFlow 2. Packt Publishing, 2021.

Course Outcomes:

1. Apply fundamental concepts of Large Language Models (LLMs), Prompt Engineering, and Word Embeddings to optimize AI-driven text processing tasks. (PO – 1, 2, 3,4, 5, 9) (PSO – 2)
2. Implement and evaluate Artificial Neural Networks (ANNs) and Deep Neural Networks (DNNs) using TensorFlow for efficient model training and performance tuning. (PO – 1, 2, 3,4, 5, 9) (PSO – 2)
3. Develop AI applications using Hugging Face Models, LangChain (Chains, Model IO, RAG, Memory, Agents), RNNs, LSTMs, and Transformers to enhance intelligent system capabilities. (PO – 1, 2, 4, 5) (PSO – 2)

Continuous Internal Evaluation

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment tool	Marks	Course outcomes attained
Weekly Assessment	30	CO1, CO2, CO3
Project Evaluation (Internal)	20	CO1, CO2, CO3
Semester End Evaluation		
Project Evaluation (External)	50	CO1, CO2, CO3

Sl. No	Evaluation Criteria for Project	Marks
1	Problem Definition & Motivation	5
2	Literature Review & Technology Selection	5
3	Design, Methodology & Implementation	15
4	Results, Evaluation & Analysis	10
5	Demo, Presentation & Documentation	10
6	Innovation & Creativity	5

PROJECT WORK	
Course Code: CIP81	Credits: 0:0:13
Prerequisite: Nil	Contact Hours:
Course Coordinator: Dr. Sini Anna Alex	

Course Contents

As a part of the project, all the final year students must carry out the following activities:

- Students will be identifying the problem definition, literature survey and methodology to be used.
- Identify engineering systems, variables, and parameters to solve the problems.
- Identify existing processes/ solution methods for solving the problem, including forming justified approximations and assumptions.
- Compare and contrast alternative solution processes to select the best process.
- Students should form a group to carry out their project. The minimum group size is 2 and the maximum group size is 4.
- The groups will be attached to one Internal Guide (and Co-guide if necessary) by the Department.
- Students can carry out their project in-house or in a reputed organization (to be approved by the Internal Guide and HOD).
- Identify the problem statement based on the current state of Art and trends in the area of interest.
- Based on the survey, identify the project requirements and do a feasibility study.
- Identify existing processes/ solution methods for solving the problem, including forming justified approximations and assumptions.
- Identify and draw a system-level architecture by showing subsystems and their input/output need.
- Apply formal idea generation tools to develop multiple engineering design solutions
- Implement the objectives by formulating the system level architecture.
- Build models/ prototypes to develop diverse set of design solutions
- Integrate and examine the implementation and test the project scope and the requirements.
- Prepare Project Report (plagiarism should be less than 20%) and by demonstrating the work.
- Publish the Project work in a Scopus indexed Conference/Journal with quartile ranking (Q1, Q2, Q3). IEEE code of ethics for authors should be strictly adhered. Outcome as a paper publication in journal/conference approved by the department with plagiarism check limit <10%. The Continuous Internal Evaluation is based on a presentation and report for 100 marks scaled down to 50 marks.
- The evaluation will be done by the Panel Members consisting of HoD, an internal guide and a co-examiner thrice during the semester.
- Weekly report to the Guide (10 marks)
- Zeroth Review: Students must do a group presentation and produce the system design, prototype, and objectives. (during 1st week) (10 marks)

- Mid-semester evaluation: Students must do a group presentation and produce documents of system requirements, and system design (during 6th week) (40 marks).
- Final Evaluation: At the End of the semester students must do a group presentation, demonstrate the project work, and submit the complete report. (during 13th week) (40 marks)

Evaluation Phase 1: Initial Review

- Formulate and draw a system-level architecture by showing subsystems and their input/output need.
- Implement the Objectives by formulating mathematical models and algorithms for each module.

Evaluation Phase 2: Mid-semester evaluation

- Presentation on detailed design
- Application of Engineering Principles
- Generate information through appropriate tests to improve or revise the design
- Use appropriate procedures, tools, and techniques to conduct experiments and collect data
- Implementation and validation of the project deliverables.

Evaluation Phase 3: Final Evaluation

- Analyze data for trends and correlations, stating possible errors and limitations
- Deliver effective oral presentations to technical and non-technical audiences
- Present results as a team, with smooth integration of contributions from all individual efforts
- Seminar with project demonstration, project report submission, and viva voce.
- Outcome as a paper publication in a Journal/Conference with a plagiarism check limit < 10% or Patent Application Filed/Published with the approval of the department.
- The SEE Project Viva Voce will be conducted for 50 marks with an Internal and External Examiner.

Rubrics for Project Evaluation- Phase I and Phase II: The rubrics for project evaluation are provided as per the below tables.

Criteria for Evaluation	Proficient 80-100	Acceptable 60-80	Need Improvements 50-60
Data elicitation Phase	Has investigated new trends in their area of interest, and Reviewed the challenges in That area. Data elicitation should include new concepts.	Has investigated new trends in their area of interest, Review some challenges in that area.	Has not investigated new trends in their area of interest.
Problem Definition	The problem domain is well understood, a clear and specific description of the problem, and relevance is well identified.	Moderately awareness of problem domain, clear description, broad idea about relevance to current technical and social context.	Minimal awareness of problem domain, Vague description, little idea about relevance to current technical and social context.
Planning	Precise Schedule and Effort Estimation Tools.	Precise Schedule and Effort Estimation Manually.	Inappropriate Schedule and Effort Estimation.

Project Management	Has taken a leadership role in the project and monitored the progress of the project. In addition, has completed all tasks assigned.	Has monitored the progress of the project and completed all tasks assigned.	Has not completed all the tasks assigned.
Literature Survey	Has read more than 10 papers from reputed journals or 15 papers from conferences and 3 books in the area of the project.	Has read more than 7 papers from reputed journals or 12 papers from conferences and 2 books in the area of the project.	Minimal, mostly from general sources, without focused study.
Requirements Specification	Complete functional, Non-functional, Performance, Security related, Clear, and Measurable.	Partial functional, Non-functional, Performance, Security related, Clear, and Measurable.	In-complete functional, Non-functional, Performance, Security related, Clear, and Measurable.
System Design	Able to generate engineering designs with justification. The team has to play the role of the main Architect in the project and design all the components of the project.	Able to generate engineering designs but not able to justify them. The team has to play the role of main Architect in the project and design 90% of the components of the project.	Able to use the tool but not able to generate engineering designs. The team has to play the role of the project and Design 75% of the components of the project.
Implementation	Able to generate and justify the best solution. Has decided on the relevant tools and platforms required for the project by evaluating alternatives. Coded all the components designed by following the standard coding guidelines.	Able to generate alternatives but not able to justify the best solution. Has coded all the components designed by him by following the Standard coding guidelines.	Able to use the tool but not able to generate alternatives. Has not coded all the components designed by him by not following the Standard coding guidelines.
Testing and Results	Proper Test Plan. Demonstrate an ability to describe engineering roles in a broader context, e.g. pertaining to the environment, health, safety, legal and public welfare, achieving towards Sustainability development Goals (SDG) has performed Integration Testing, Performance Testing.	Barely meets all the requirements, Optimized Solution, Poor Test Plan. Demonstrate an ability to describe engineering roles in a specific context, e.g. pertaining to the environment, health, safety, legal and public welfare. Has not performed Integration Testing, Performance Testing.	Haphazard testing, barely meets requirements, unable to infer results. Not able to demonstrate engineering roles in a broader context, e.g. pertaining to the environment, health, safety, legal and public welfare
Report Writing	Excellent Organization, No technical or Grammar errors Concise and Precise, Incomplete documentation.	Average organization, No technical or Grammar errors Concise and Precise, Incomplete documentation	Poor Clarity in technical contents and organization, grammar error.

Presentation and Viva-voce	Excellent Professional and Technical communication, Contribution from an individual to a team is good and results in an integrated team presentation.	Good Professional and Technical Communication, A contribution from an individual to a team is good but not well groomed in the team.	Average Professional and Technical Communication, Contributions from an individual to a team are moderate
-----------------------------------	---	--	---

Course Outcomes (COs):

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

1. Review the current state of Art and trends in their area of interest and identify a suitable problem in their chosen subject domain with justification.
2. Survey the available research literature/documents for the tools and techniques to be used.
3. Examine the functional, non-functional, and performance requirements of their chosen problem definition.
4. Design system architecture and different components and develop all the system components using appropriate tools and techniques.
5. Work effectively in a team and use good project management practices and defend the project work as a team.

RESEARCH / INDUSTRIAL INTERNSHIP	
Course Code: INT82	Credits: 0:0:5
Prerequisite: Nil	Contact Hours:
Course Coordinator: Dr. Sini Anna Alex	

Course Contents

Internship: The department should prepare a Gantt chart with milestones, deliverables, and evaluation and maintain weekly dairy signed by both the Internal and External Guide.

Guidelines:

- The student can do the Internship during the summer vacation 6th semester or in the 7th semester.
- The student should take prior permission from the department committee before carrying out the internship. The offer letter should be submitted to the Dept. coordinator.
- The duration of the Internship is 3- 4 -6 weeks after the 6th or 7th semester (total Six months).
- The Internship assessment form assessed by the company mentor, report of the Internship, and Internship completion certificate, need to be submitted to the department coordinator after completion of the Internship.
- The department will constitute a committee for the evaluation of the Internship of the student and based on the following rubrics, the student will be evaluated.

Rubrics for Internship Evaluation: Each supervisor must fill a rubric for each student:

Performance Indicator	Excellent (20 Marks)	Very Good (18 Marks)	Good (16 Marks)	Average (14 Marks)	Poor (12 Marks)
Clarity in Problem Statement and Objective	Well-understood and well-defined objectives	Well-understood and partially defined objectives	Partially understood but partially defined objectives	Partially understood but not defined objectives	Not properly understood and not properly defined objectives
Design of Module and Code Efficiency	Excellent design and defined all the objectives	Excellent design and partially defined all the objectives	Partially designed and partially defined all the objectives	Partially designed but not defined all the objectives	Irrelevant design and objectives
Usage of Latest Software tools or Library Packages	Latest IDE or library packages used	Moderately used the IDE or library packages	Old version IDE or library packages are used	Aware of the IDE or library packages but not used	No awareness of IDE or library packages
Proper Documentation and Presentation	Excellent documentation and presentation	Good documentation and partially	Document with partial content and	Document with partial content and	Poor documentation and poor

Skills		present the content	partially present the content.	poor presentation	presentation
Completion of the Project in the scheduled Timeline	All objectives and deadlines are met	All objectives were met but partially met the deadlines	Partially met the objectives and deadlines	Partially met the objectives but not meet the deadlines	No objectives and deadlines are met
Total	100 (91-100 Marks)	90 (81-90 Marks)	80 (71-80 Marks)	70 (61-70 Marks)	60 (51-60 Marks)

Course Outcomes (COs):

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

1. Schedule milestones of deliverables and formulate the requirements of the proposed work.
2. Apply the engineering knowledge to develop software in an industry setting.
3. Develop the inter-personal skills required to work in a professional team.
4. Engage in independent study of technology required for development of software.
5. Demonstrate the project and appraise its effectiveness.