Structure of Post Graduate

(ME Artificial Intelligence)

THAPAR INSTITUTE OF ENGINEERING & TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
PATIALA, PUNJAB, INDIA

COURSE SCHEME & SYLLABUS

(2025)

M.E. (ARTIFICIAL INTELLIGENCE)

ME- ARTIFICIAL INTELLIGENCE (2025)

SEMESTER I						
S. NO.	**CODE**	**TITLE**	**L**	**T**	**P**	**Cr**
1	PAI101	Essentials of Artificial Intelligence	3	0	2	4
2	PAI102	Data Engineering	3	0	2	4
3	PAI103	Mathematics Behind AI	3	0	2	4
4	PAI104	Advanced Deep Learning	3	0	2	4
5	PHU004	Research Methodology, Ethics and IPR	2	0	0	2
6	PAI106	Python for AI	3	0	2	4
		TOTAL	**17**	**0**	**10**	**22**
SEMESTER II (Robotics)						
1	PAI201	Introduction to Edge AI	3	0	2	4
2	PAI202	Vision Models	3	0	2	4
3	PAI203	Autonomous Robotics & Reinforcement Learning	3	0	2	4
4	PAI204	Conversational AI	3	0	2	4
5	PAI205	Diversity, Ethics, and Security	3	0	2	4
6		ELECTIVE-I	2	0	2	3
		TOTAL	**17**	**0**	**12**	**23**
SEMESTER II (Generative AI)						
1	PAI206	Generative AI - Word Embeddings, Tokens, and NLP	3	0	2	4
2	PAI207	Language Models and the Transformer	3	0	2	4
3	PAI208	Neural Architecture Design and Optimization	3	0	2	4
4	PAI209	LLM Scaling & Scaling Model Training to Distributed Workloads	3	0	2	4
5	PAI210	Diffusion Models in Generative AI	3	0	2	4
6		ELECTIVE-I	2	0	2	3
		TOTAL	**17**	**0**	**12**	**23**
SEMESTER II (AI for Science)						
1	PAI220	Linear Algebra	3	0	2	4
2	PAI211	Physics-Informed Neural Networks	3	0	2	4
3	PAI212	Neural Operators	3	0	2	4
4	PAI213	Data & Uncertainty Quantification	3	0	2	4
5	PAI214	HPC and Physics Nemo framework	3	0	2	4
6		ELECTIVE-I	2	0	2	3
		TOTAL	**17**	**0**	**12**	**23**
ELECTIVE-I						
1	PAI215	Quantum Computing	2	0	2	3
2	PAI216	Introduction to Gen AI (Only for Robotics and AI for Science)	2	0	2	3

3	PAI217	Digital Twins	2	0	2	3
4	PAI218	Multimodal Learning and its Applications	2	0	2	3
5	PAI219	Accelerated Computing	2	0	2	3
SEMESTER III						
1	PAI391	Dissertation/Internship Interim Report	-	-	-	4
2	PAI392	Seminar	-	-	-	4
		TOTAL				**8**
SEMESTER IV						
1	PAI491	Project Semester/Dissertation	-	-	-	16
		TOTAL	**-**	**-**	**-**	**16**
		GRAND TOTAL - FOUR SEMESTER CREDITS				**69**

PAI101: ESSENTIALS OF ARTIFICIAL INTELLIGENCE

L	T	P	Cr
3	0	2	4.0

Course Objectives: This course aims to explore the fundamental concepts, scope, and history of artificial intelligence and its role in modern technology. Alongside, this course helps students learn problem-solving techniques and search strategies for intelligent decision-making, and introduces methods of knowledge representation, reasoning, and inference in AI systems. Also, an overview of machine learning and neural network principles for data-driven tasks along with recent advancements in AI have been covered.

AI Essentials: Definition and scope of AI, History and evolution of AI, AI vs. Human Intelligence, AI in engineering and technology.

Intelligent Agents and Environment Interaction: Definition and Types of Intelligent Agents, Agents and Environments: How agents perceive and act.

Knowledge Representation and Reasoning: Introduction to Knowledge Representation, Logic-based approaches: Propositional and Predicate Logic, Rule-based Systems and Expert Systems

Introduction to Data and its analysis: Types of data: Structured, Semi-structured, Unstructured, Data Streams, Statistical Data types, Sampling, Data Analysis Methods: Descriptive Analysis, Exploratory Analysis, Inferential Analysis, Predictive Analysis.

Machine Learning Essentials: Introduction to machine learning, Types of learning (Supervised, Unsupervised, Semi-Supervised, and Reinforcement Learning), Introduction to Neural Networks and Deep Learning.

Recent Advancements in AI: Explainable AI (XAI), Edge AI, Generative AI, AI for Social Good (Applications of AI in healthcare, education, climate change, and sustainability), AI Security (Protecting AI systems from attacks and ensuring secure AI development)

Case studies of AI-driven innovations: Smart Speakers, Self-Driven cars, Smart Agriculture, etc.

Laboratory Work:

Implement breadth-first search and depth-first search on a sample state-space problem and analyze their performance, Create a simple knowledge base with propositional logic rules and perform logical inference (forward/backward chaining) for an expert system scenario, Train a decision tree classifier on a real-world dataset (e.g., Iris or a small medical dataset), evaluate accuracy, and discuss feature selection, Design and train a basic feedforward neural network (using a standard library) for a prediction task (such as handwritten digit recognition or pattern classification) and analyze the results, Use AI techniques on an environmental dataset (e.g., predict energy consumption or crop yield using regression or classification) to explore an application in sustainability.

Course Learning Objectives (CLOs)

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

1. Understand the basic principles and concepts of Artificial Intelligence.
2. Explore the applications of AI in various engineering disciplines.
3. Understand the ethical considerations and societal impact of AI.
4. Analyze the impact of recent advancements in deep learning on AI applications across various industries.

Text Books:

1. Russell S. and Norvig P., *Artificial Intelligence: A Modern Approach*, 4th edition, Pearson (2020).
2. Alpaydin E., *Introduction to Machine Learning*, 3rd edition, MIT Press (2020).
3. Goodfellow I., Bengio Y., and Courville A., *Deep Learning*, MIT Press (2016).

Reference Books:

1. Nilsson N. J., *Artificial Intelligence: A New Synthesis*, Morgan Kaufmann (1998).
2. Mitchell T. M., *Machine Learning*, McGraw Hill (1997).
3. Jurafsky D. and Martin J. H., *Speech and Language Processing* (3rd ed. draft), Pearson (2023).
4. Flach P., *Machine Learning: The Art and Science of Algorithms that Make Sense of Data*, Cambridge University Press (2012).

PAI102: DATA ENGINEERING

L	T	P	Cr
3	0	2	4.0

Course Objectives: This course aims to explore the fundamentals and applications of data engineering, with a consideration of sustainability. Students will acquire the ability to minimize their environmental impact by designing, constructing, and maintaining data systems. The subjects covered include data lifecycle management, green computation, data pipelines, and storage optimization.

Foundations of Data Engineering and Data Modeling: Introduction to Data Engineering: Roles, workflows, and modern data stack, Data types, file formats (CSV, JSON, Avro, Parquet), Batch vs. streaming data, Data modeling: ER diagrams, normalization, star/snowflake schema, Introduction to data warehousing and OLAP, Use case: data modelling for student performance, school access in rural areas etc.

ETL/ELT Pipelines and Workflow Orchestration: Designing ETL/ELT pipelines, Data ingestion from APIs, sensors, and flat files, Data transformation and cleansing techniques, Workflow orchestration: Apache Airflow, Prefect, Error handling, retries, and monitoring, Use case: Ingesting and processing sensor data from water quality monitoring stations for real-time alerts and long-term reporting.

Distributed Data Processing and Big Data Frameworks: Introduction to distributed systems and Apache Hadoop, Apache Spark: RDDs, DataFrames, Spark SQL, Partitioning, shuffling, and performance optimization, Batch vs. stream processing fundamentals, Use case: Analyzing large-scale satellite and weather data to track deforestation and CO_2 emissions over time using Spark.

Real-Time Data Streaming and Messaging Systems: Introduction to message brokers and stream processing, Apache Kafka: architecture and core concepts, Stream processing with Spark Streaming/Apache Flink, Event time, windows, watermarks, Use case: Real-time energy consumption tracking from IoT meters to optimize grid load and promote renewable energy usage.

Data Governance, Cloud Deployment, and DevOps: Data quality and validation: expectations and checks, Metadata, lineage, and cataloguing, Security, encryption, and compliance (GDPR, HIPAA), Cloud platforms: AWS/GCP/Azure basics, Docker, Terraform, CI/CD pipelines for data workflows, Use case: Deploying a data platform in the cloud to monitor and visualize real-time traffic and pollution levels in urban areas.

Laboratory work:

- Creating and monitoring ETL pipelines
- Ingesting real-time sensor data using Kafka
- Running distributed processing jobs in Apache Spark
- Using Airflow for orchestration and scheduling
- Validating data quality and deploying pipelines on cloud infrastructure

Course Outcomes (COs)/Course Learning Outcomes (CLOs):
After completion of this course, the students will be able to:

1. Understand and design scalable data models and workflows for structured and unstructured data.
2. Build and manage ETL/ELT pipelines using modern tools and automation frameworks.
3. Apply distributed and real-time processing techniques to solve large-scale data problems.
4. Develop data engineering solutions for real-world challenges, including sustainability, governance, and cloud deployment.

Text Books:

1. Andreas Kretz, The Data Engineering Cookbook, 2022
2. Ananthakrishnan G., Data Engineering with Python, Packt Publishing, 2020

Reference Books:

1. Jesse Anderson, Data Engineering Teams, Manning, 2020
2. Jeffrey Aven, Data Lake Architecture, O’Reilly Media, 2018
3. Bill Inmon, Building the Data Warehouse, Wiley, 4th ed.
4. Valliappa Lakshmanan et al., Data Engineering on Google Cloud Platform, O’Reilly, 2022

PAI103: MATHEMATICS BEHNID AI

L	T	P	Cr
3	0	2	4.0

Course Objectives: The objective of the course is to teach fundamental mathematics for effective learning in AI. It covers principles of mathematics concepts that are essential for AI along with its use case in real life scenarios.

Linear Algebra: System of linear equations, introduction to vectors, matrices, and matrix operations, solving system of linear equations, determinants.

Calculus for Learning Algorithms: Single-variable and multivariable derivatives, Gradient and partial derivatives, Chain rule in backpropagation, Jacobian and Hessian matrices

Optimization Techniques: Convexity and its significance in AI, Gradient Descent (GD) and Stochastic Gradient Descent (SGD), Learning rate, convergence, overfitting, introduction to momentum and Adam optimizer, training energy forecasting models, optimizing solar panel layout via gradient methods

Least Squares and Linear Models: Linear regression via normal equations, Overdetermined systems and pseudo-inverse, Ridge regression, Fit linear models using closed-form and iterative methods, applications like linear models for crop yield estimation, analyzing sustainability indices

Probability and Random Variables: Probability spaces, events, axioms of probability, conditional probability, Bayes' theorem, random variables, probability distributions (discrete and continuous), expectation, variance, functions of random variables, normal distribution, central limit theorem.

Laboratory work:
Each laboratory experiment will consist of numerical exercises on one of the above topics. Laboratory experiments will be performed using Matlab/SPSS/Python.

Course Outcomes (COs)/Course Learning Outcomes (CLOs):
After completion of this course, the students will be able to:

1. Apply linear algebraic methods and calculus to perform computational task.
2. Analyze optimization techniques such as gradient descent, stochastic methods, and regularization strategies
3. Evaluate the performance and applicability of linear models, including linear and ridge regression, using least squares methods
4. Compute probabilities of events with an understanding of random variables and distributions.

Text Books:

1. Walpole R. E., Myers R. H., Myers S. L. and, Keying Y., Probability and Statistics for Engineers and Scientists, Pearson Education, 9th edition, (2021).
2. Strang, G., Introduction to Linear Algebra, Wellesley-Cambridge Press, 5th Edition (2016)

Reference Books:

1. Hogg R. V., McKean J. W. and Craig A. T., Introduction to Mathematical Statistics, Pearson, 8th edition, (2019).
2. Hastie, T., Tibshirani, R., & Friedman, J., The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Springer, 2nd edition, (2009).

PAI104: ADVANCED DEEP LEARNING

L	T	P	Cr
3	0	2	4.0

Course objective: The main objective of this course is to enable the student with deep learning architectures to build an intellectual machine for making decisions on behalf of humans.

Introduction: Single Layer vs Multilayer Perceptron, Backpropagation Algorithm, Activation Functions. Loss Functions, Optimization Approaches GD, SGD, Momentum based GD, AdaGrad, RMSProp, Adam, Hyperparameters Tuning & Regularization Techniques.

Convolutional Neural Network: -Architecture, Convolution, Pooling, Filters and Feature Maps. Batch Normalization. Transfer Learning Models VGG16, ResNet. Google Inception, Application of CNN in Healthcare Analytics.

Recurrent Neural Network: - Backpropagation Through Time. Bidirectional RNNs (BRNN), Vanishing and Exploding Gradients. Long Short-Term Memory (LSTM). Bi-directional LSTM. Gated recurrent unit GRU, Applications of RNN - Natural Language Processing, Time Series Forecasting of Weather Parameters, Attention Models, Transformer-BERT, RoBERT.

Unsupervised Learning Deep Models: - Autoencoder, Variational Autoencoder. Denoising Autoencoders. Sparse Autoencoders Generative Adversarial Networks, Boltzmann Machine.

Laboratory Work: To Implement Image classification, Segmentation, Sentiment Analysis, Traffic information analysis using open-source library such as Tensorflow, Keras etc.

Course Outcomes (COs)/Course Learning Outcomes (CLOs):
After completion of this course, the students will be able to:

1. Comprehend the advancements in learning techniques.
2. Demonstrate the application of deep learning in image processing
3. Apply deep learning models in time series forecasting.
4. Analyse performance of deep networks.

Text Books:

1. Ian Goodfellow and Yoshua Bengio and Aaron Courville, Deep Learning, MIT Press, 2016
2. Aston Zhang, Zack C. Lipton, Mu Li, and Alexander Smola. Dive into Deep Learning, 2020.

Reference Books:

1. Charu C. Aggarwal, “Neural Network and Deep Learning”, Springer, 2023.
2. Iddo Drori, The Science of Deep Learning, Cambridge University Press.

PHU004: RESEARCH METHODOLOGY, ETHICS AND IPR

L	T	P	Cr
2	0	0	2.0

Course Objectives: The course aims to equip the students to analyse research related information. And also it sensitize the students to ethical research practices. To equip them to write technical reports and research paper and the process of patent filing. Also to create awareness about the consequences of IPR Infringement

Unit 1: Meaning of Research Problem, Sources of Research Problem, Criteria and Characteristics of good Research Problem, Errors in selecting a research Problem, scope and objectives of research problem.

Unit 2: Effective Literature studies, approaches and analysis.

Unit 3: Effective Technical Writing, How to write report and Research paper; developing a research proposal.

Unit 4: Non Parametric Tests: When to use a Nonparametric Tests**;** Mann Whitney U Test; Sign Test; Wilcoxon Signed Rank Test and Kruskal-Wallis Test.

Unit 5: Ethics: Need for Ethics in Professional Life; Kohlberg's Theory of Moral Development and Its Applicability to Engineers. Professional Ethics: Values in Work Life; Professional Ethics and Ethos; Codes of Conduct. Research Ethics, Plagiarism, Case Studies on Ethics.

Unit 6: Introduction to IPR: Nature of Intellectual Property Rights: Patents; Designs; Trademarks; Copyright; Trade Secrets; Industrial Design; Geographical Indicators; Integrated Circuits. International Character of IPRs, Role of IPRs in Economic Development.

Patents: Introduction to Patents, Inventions not Patentable, Procedure for grant of Patents, Rights and Obligations of a Patentee; IPR Infringement.

Case studies on IPRs.

Course Learning Outcomes (CLOs) / Course Objectives (COs):
After the completion of the course, the student will be able to:
1. Analyse research related Information.
2. Indulge in ethical research practices
3. Equipped to write technical reports and research paper.
4. Equipped with the process of patent filing
5. Possess awareness about consequences of IPR Infringement

Text Books:

1. Geoffrey R. Marczyk. Essentials of Research Design and Methodology, Wiley, (2008).
2. Wayne Goddard, Stuart Melville. Research methodology: An Introduction, Juta, (2004).
3. Thomas, C. George. Research Methodology & Scientific Writing, Ane Books Pvt. Ltd, (2016).
4. Menell, Peter S, Lemley, Mark A, Merges, Robert P. Intellectual Property in the New Technological Age, Vol. I Aspen Law & Business, (2019).
5. Menell, Peter S, Lemley, Mark A, Merges, Robert P. Intellectual Property in the New Technological Age, Vol. II Aspen Law & Business, (2019).
6. Narayanan, P., Intellectual Property Law, Eastern Law House, (2008).

PAI106: PYTHON FOR AI

L	T	P	Cr
3	0	2	4.0

Course Objectives: The course aims to develop proficiency in Python libraries and tools for data manipulation, visualization, machine learning, and deep learning. Students will learn to build, train, and evaluate AI models, enabling them to apply intelligent systems to solve real-world problems effectively.

Introduction to Python Programming: Setting up the Python environment (IDEs, Jupyter Notebooks), Writing and running Python scripts, Basic syntax, variables, and data types (integers, floats, strings).

Control Structures: if, else, elif, Loops: for, while, break, continue, pass.

Data Structures in Python: Lists and list operations (indexing, slicing, appending), Tuples and their immutability, Dictionaries: key-value pairs, adding and removing elements, Sets: basic operations and set methods.

Working with Strings and Files: String manipulation and formatting, Reading from and writing to files, Working with file paths, Handling exceptions in file operations. Using numpy, torch, mat and pickle files.

Error Handling and Debugging: Understanding exceptions and error types, Using try, except, else, finally for error handling, Debugging techniques and tools.

Libraries and APIs: Introduction to Python libraries (e.g., requests, json, datetime), Working with APIs to fetch and process data, Understanding JSON data format, Manipulating XML files.

Introduction to Data Handling with Pandas: Working with real-world datasets related to public health, education, and clean energy, exploring data cleaning, transformation, and visualization.

Introduction to NumPy and PyTorch: Tensors and tensor operations for building AI models, Hands-on examples involving sustainable development use cases such as climate trend prediction, or clean water monitoring.

Laboratory work:
Set up a Python environment using Anaconda and Jupyter Notebook, and write basic scripts to understand syntax and data types. Implement control structures (loops and conditionals) and manipulate data using lists, dictionaries, and strings. Use NumPy and Pandas to handle arrays and datasets, and practice loading and analyzing real-world data files like CSV and JSON.

Course Outcomes (COs)/Course Learning Outcomes (CLOs):

1. Analyze and apply basic programming constructs using Python, including data types, operators, and expressions.
2. Implement control structures and data handling techniques to develop structured and efficient Python programs.

3. Perform file operations, string manipulations, and exception handling to create robust and error-resilient applications.

4. Demonstrate the use of built-in Python libraries for data processing and API interaction in real-world scenarios.

Text Books:

1. Sweigart A., Automate the Boring Stuff with Python: Practical Programming for Total Beginners, 1st ed. No Starch Press, 2015.
2. Matthes E., Python Crash Course, 2nd ed. No Starch Press, 2019.

Reference Books:

1. Lutz M., Learning Python, 5th ed. O'Reilly Media, 2013.
2. Géron, A., Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems. O'Reilly Media, Inc., 2022.

Semester – II (Robotics)

PAI201: INTRODUCTION TO EDGE AI

L	T	P	Cr
3	0	2	4.0

Course Objective: This course will provide students with advanced conceptual knowledge and practicals on various computer vision and deep learning applications and provide the overall environment for end-to-end pipeline development from data collection to deployment.
Introduction: Utilizing Jetpack SDK and other NVIDIA Toolkits to deploy CNN models on Jetson, Creating Jetbot kits and deploying various applications

Introduction to Edge AI: AI at the Edge, IoT, Jetson Architecture, Getting Started with Jetpack, Getting started with NGC Containers on Jetson. Building your own dockers.

Introduction to NVIDIA Toolkits and SDKs: Transfer Learning Toolkit, Kubernetes Deployment, Deepstream SDK, Deploying Classification, Detection and Segmentation Models on Jetson Devices.

Advanced Vision: Pose Recognition (Deploying Human pose model), Depth Estimation: Mono/Stereo depth and point extraction, Visual Odometry: Camera pose estimation from DNNs.

Laboratory Work:

- Setting up the Jetson Project kit. {DLI Online Course: Getting Started with AI on Jetson Nano.}
- Deployment of Various Classification, Object Detection and Segmentation models in Jetson Nano.
- Depth Estimation in Jetson Nano
- Human Pose Estimation in Jetson Nano

Course Learning Outcomes (CLOs) / Course Objectives (COs):
After the completion of the course, the student will be able to:

1. Demonstrate deployment of realtime vision based applications on edge devices.
2. Deploy kubernetes framework using Jetson Nano Devices.
3. Apply vision based pose estimation techniques.
4. Implement depth estimation algorithms using vision sensors.

Text Books:

1. Computer Vision: Algorithms and Applications, R. Szeliski, Springer, 2011.
2. Computer Vision: A Modern Approach, D. Forsyth and J. Ponce, Prentice Hall, 2nd ed., 2011.
3. Richard Szeliski, Computer Vision: Algorithms and Applications, Springer-Verlag London Limited 2011.

Reference Books:

1. Richard Hartley and Andrew Zisserman, Multiple View Geometry in Computer Vision, Second Edition, Cambridge University Press, March 2004.
2. K. Fukunaga; Introduction to Statistical Pattern Recognition, Second Edition, Academic Press, Morgan Kaufmann, 1990.
3. R.C. Gonzalez and R.E. Woods, Digital Image Processing, Addison- Wesley, 1992.
4. Christopher M. Bishop; Pattern Recognition and Machine Learning, Springer, 2006.

PAI202: VISION MODELS

L	T	P	Cr
3	0	2	4.0

Course Objectives: The objective of the course is to understand the fundamental principles of deep neural networks in visual perception tasks, examine the integration of vision and language in large multimodal models, and explore their applications in cross-modal tasks.

Fundamentals of Vision with DNNs: Introduction to vision models and their evolution, DNN basics (perceptron, MLPs, backpropagation), Image representation, preprocessing techniques, Convolutional Neural Networks (CNNs), BatchNorm, skip connections, bottlenecks, and depth vs width, Feature hierarchies, receptive fields, and interpretability, Vision-specific training challenges (overfitting, class imbalance), Transfer Learning and Pre-trained CNNs (VGG, ResNet, EfficientNet), Fine-tuning vs feature extraction

Advanced Vision Architectures: Object detection (Faster R-CNN, SSD, YOLO family), Segmentation (FCN, U-Net, DeepLab, Mask R-CNN), Evaluation metrics (IoU, mAP, Dice score), Attention and Transformers for Vision, Vision Transformers (ViT, DeiT, Swin Transformer), Comparison with CNNs: when and why to use Transformers

Self-Supervised & Foundation Vision Models: Motivation for self-supervision in vision, SimCLR, MoCo, BYOL, DINO, Contrastive and masked image modeling objectives, Foundation vision models (SAM, DINOv2), Evaluation on downstream tasks (zero-shot, few-shot transfer), Pros and cons of training vs using pretrained foundation models

Vision-Language Large Models (vLLMs): Vision-Language Learning (embeddings, fusion strategies), CLIP and ALIGN (contrastive pretraining, image-text retrieval), Prompting and zero-shot classification using CLIP, Image captioning models (Show and Tell, Show Attend and Tell, BLIP), Visual Question Answering, grounding, and attention, Training vLLMs with vision encoders and LLM decoders, Multimodal Transformers (Flamingo, PaLI, GIT), Pretraining methods, tokenization, positional encodings, Challenges (alignment, hallucination, multimodal biases)

Applications, Tools & Deployment: Applications in Robotics, autonomous vehicles, medical imaging, Hands-on using CLIP, BLIP, and SAM with HuggingFace/Transformers and OpenCLIP, Scaling vLLMs (training vs inference bottlenecks), Model distillation, quantization, deployment on edge devices, Introduction to tools (ONNX, Triton, TensorRT, DeepSpeed)

Laboratory work:
Implementation of convolutional neural networks for image classification, application of transfer learning, Introduction to models like CLIP and SAM for image captioning and multimodal tasks and experiment with pretrained diffusion models for image generation and fine-tune vision-language models (vLLMs) on custom datasets.

Course Outcomes (COs)/Course Learning Outcomes (CLOs):

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

1. Understand the theoretical foundations of deep vision models and multimodal learning.
2. Apply state-of-the-art vision and vision-language models to solve domain-specific tasks.
3. Analyze and compare the performance of various vision models across benchmarks.
4. Design and implement vision-language systems using large pretrained models.

Text Books:

1. Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y. (2016). Deep learning. Cambridge: MIT press.
2. Elgendy, M. (2020). Deep learning for vision systems. Manning.
3. Goyal, P., Pandey, S., & Jain, K. (2018). Deep learning for natural language processing. New York: Apress.
4. *Multimodal Machine Learning: Techniques and Applications* – Tiezheng Yu, Xiangyu Xiang, Pascale Fung (2023)

Reference Books:

1. Shanmugamani, R. (2018). Deep Learning for Computer Vision: Expert techniques to train advanced neural networks using TensorFlow and Keras. Packt Publishing Ltd.
2. Tunstall, L., Von Werra, L., & Wolf, T. (2022). *Natural language processing with transformers*. " O'Reilly Media, Inc.".

PAI203: AUTONOMOUS ROBOTICS & REINFORCEMENT LEARNING

L	T	P	Cr
3	0	2	4.0

Course Objectives: Understand the fundamental principles of autonomous robotics and reinforcement learning (RL). Explore state-of-the-art algorithms for autonomous decision-making and robotic control. Develop hands-on expertise in deep reinforcement learning for robotic applications. Investigate policy optimization techniques and reward structures for self-learning agents. Analyze real-world case studies involving AI-driven robotics for navigation, manipulation, and perception.

Foundations of Autonomous Robotics: Basics of robotics, Integrating vision systems, LiDAR, and IMUs for autonomous navigation, robotic kinematics, sensor technologies and actuator control. Real-time mapping and localization using probabilistic models.

Reinforcement Learning Fundamentals: Mathematical foundations of RL, value-based and policy-based learning techniques, exploration vs. exploitation strategies. Modeling decision-making frameworks for agents.

Deep Reinforcement Learning for Robotics: Implementation of neural networks in RL, model-free and model-based RL approaches, simulation environments for RL training. Q-learning, SARSA vs. Actor-Critic algorithms.

Advanced Robotic Autonomy: SLAM and real-time localization strategies, multi-agent decision-making and coordination, Human-Robot Interaction (HRI) and adaptive systems

Applications & Case Studies: Autonomous vehicles and robotic navigation, AI-driven healthcare robotics, reinforcement learning in industrial automation

Laboratory work:

1. Implementation of RL algorithms in Python (TensorFlow/PyTorch).
2. Robotic motion control using ROS (Robot Operating System).
3. Multi-agent coordination simulations with deep reinforcement learning.
4. Hands-on experiments with physical robotic platforms and sensor integration.

Course Outcomes (COs)/Course Learning Outcomes (CLOs):
After completion of this course, the students will be able to:

1. Demonstrate a comprehensive understanding of autonomous robotics and reinforcement learning fundamentals.
2. Apply RL algorithms to train autonomous robotic systems for decision-making.
3. Implement deep reinforcement learning techniques for robotic control and optimization.
4. Design and evaluate robot learning models in simulated and real-world environments.
5. Analyze and develop robotics solutions that integrate AI-driven decision-making for various applications.

Textbooks:

1. Sutton, R. S., & Barto, A. G. – Reinforcement Learning: An Introduction (MIT Press).
2. Russell, S., & Norvig, P. – Artificial Intelligence: A Modern Approach (Pearson).
3. Kober, J., Bagnell, J. A., & Peters, J. – Reinforcement Learning in Robotics: A Survey.

Reference Books:

1. Goodfellow, I., Bengio, Y., & Courville, A. – Deep Learning (MIT Press).
2. Silver, D., Hassabis, D., & Sutton, R. – Advances in Reinforcement Learning.
3. Murphy, K. P. – Machine Learning: A Probabilistic Perspective (MIT Press).
4. Siciliano, B., & Khatib, O. – Springer Handbook of Robotics.

PAI204: CONVERSATIONAL AI

L	T	P	Cr
3	0	2	4.0

Course Objectives: The primary objective is to equip students with the theoretical knowledge and practical skills to design, build, and evaluate conversational agents integrated into robotic systems, considering constraints like edge deployment and multimodality.

Introduction to Conversational AI in Robotics: History and evolution of dialogue systems, Overview of dialogue system architecture, Types of dialogue systems: task-oriented vs. open-domain.

Natural Language Processing Foundations: Tokenization, embedding (Word2Vec, GloVe, contextual embeddings), Transformers and pretrained language models, Transfer learning for NLU tasks.

Intent Recognition and Slot Filling: Architecture of NLU modules, ATIS dataset and domain adaptation, Joint modeling for intent and slots.

Dialogue State Tracking & Dialogue Management: Finite-state and frame-based systems, Neural dialogue state trackers, Dialogue policy: rule-based vs. reinforcement learning, Policy gradient methods for dialogue, Deep Q-learning for conversation, Exploration vs. exploitation in robotic interaction.

Speech Interfaces and Spoken Dialogue Systems: ASR (Automatic Speech Recognition) and TTS (Text-to-Speech), Integration with conversational agents, Multilingual and noisy environment considerations.

Large Language Models in Robotics: LLMs for instruction following and grounding (e.g., PALM-E, LLaVA), Prompt engineering for embodied tasks, Benefits and pitfalls of using vLLMs in robots.

Edge AI and Deployment: Deploying dialogue systems on edge devices, Model compression and quantization (TensorRT, ONNX), Managing latency and energy in real-time conversations, Deploy a small ASR-NLU pipeline on a Raspberry Pi or Jetson Nano.

Laboratory work:

Students will implement and evaluate conversational modules integrated with robotic systems using tools like Hugging Face/NeMo/Riva. Labs will cover intent recognition, dialogue policy learning, multimodal reference resolution, and deployment on edge devices like Raspberry Pi or Jetson Nano.

Course Outcomes (COs)/Course Learning Outcomes (CLOs):
After completion of this course, the students will be able to:

1. Understand and analyze the architecture and components of conversational AI systems in the context of robotics.
2. Design and implement dialogue management strategies, including rule-based and reinforcement learning policies, for natural and efficient human-robot interaction.
3. Apply intent classification and slot filling techniques using neural models to enable task-specific understanding in robotic agents.

4. Develop and evaluate speech-based input/output systems by integrating Automatic Speech Recognition (ASR) and Text-to-Speech (TTS) modules into robotic platforms for real-time, multilingual interaction.
5. Deploy and optimize conversational AI models on edge devices under real-time and resource-constrained conditions.

Text Books:

1. Williams, Jason D., Young, Steven L., and G.H.L., René D., *Building Dialogue Systems for Conversational AI*, Cambridge University Press, 2020.
2. Natural Language Processing with Transformers, Revised Edition 1st Edition, Lewis Tunstall, Leandro von Werra, Thomas Wolf, Bengio, Y., LeCun, Y., and Hinton, G. (2015). Publisher : O'Reilly Media
3. Daniel Jurafsky and James H. Martin, "Speech and Language Processing", 3rd edition draft, 2019 [JM-2019].

Reference Books:

1. Geoffrey Hinton, Li Deng, Dong Yu, George E. Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N. Sainath, and Brian Kingsbury, Deep Neural Networks for Acoustic Modeling in Speech Recognition, IEEE Signal Processing Magazine, 29(6):82-97, 2012.
2. Deep Learning for Natural Language Processing First Edition, Stephan Raaijmakers. Publisher: Publisher: Manning; First Edition (December 6, 2022)

PAI205: DIVERSITY, ETHICS AND SECURITY

L	T	P	Cr
3	0	2	4.0

Course Objectives: This course explores the intersection of robotics with ethical frameworks, diversity and inclusion, and security concerns in modern AI systems. It provides a comprehensive overview of responsible robotics design and deployment, including socio-technical challenges, bias mitigation, privacy, and adversarial threats.

Introduction to Ethics and Robotics: Historical context and evolution of robot ethics, Moral and ethical frameworks: Deontology, Utilitarianism, Virtue Ethics, Role of ethics in AI and Robotics decision-making, Asimov's Laws of Robotics and modern adaptations.

Diversity and Inclusion in Robotics: Gender, race, and accessibility biases in robotic design, Inclusive user-centered design principles, Cross-cultural design and acceptance of robotic systems, Case studies: Biased algorithms in robotics and their social consequences.

Robotics and Human Rights: Legal and policy frameworks (GDPR, IEEE guidelines, UNESCO AI Ethics), AI and robotics for disabled and elderly populations, Ethical challenges in surveillance, policing, and military robotics, Autonomous systems and accountability.

Security and Privacy in Robotic Systems: Cybersecurity threats in autonomous systems, Physical safety and fail-safe mechanisms, Data privacy in sensory and decision systems, Adversarial machine learning and spoofing attacks in robotics.

Ethics of Human-Robot Interaction: Emotional manipulation and trust in HRI, Robotics in education, healthcare, and domestic environments, Ethics in robot social behavior and anthropomorphism, Moral status of robots and ethical dilemmas in decision-making.

Responsible Innovation and Future Challenges: AI alignment and value-sensitive design, Governance models and global initiatives, Sustainability and environmental ethics in robotics manufacturing, Preparing for super intelligent systems and digital sentience.

Laboratory work:
Case study analysis of real-world incidents involving robotic bias or security breaches, Mini-projects on designing socially aware robots or creating attack-defense simulations in robotics.

Course Outcomes (COs)/Course Learning Outcomes (CLOs):
After completion of this course, the students will be able to:

1. Understand ethical theories and frameworks applicable to robotics and AI.
2. Analyze and evaluate the societal impacts of robotics with a focus on diversity and inclusion.
3. Explore legal and policy perspectives on the deployment of robotic systems.
4. Apply ethical, social, and security dimensions while designing robotic systems.

Text Books:

1. *Patrick Lin, Keith Abney, Ryan Jenkins,* Robot Ethics 2.0: From Autonomous Cars to Artificial Intelligence, Oxford Press (2017), 1st Edition.
2. *Thomas Arnold*, Ethics for Robots: How to Design a Moral Algorithm, Oxford Press (2021), 1st Edition.
3. *Markus Dubber, Frank Pasquale, Sunit Das*, Ethics in Robotics and AI: An Introduction to the Moral Dimensions of Cutting-Edge Technologies, Oxford University Press (2020), 1st Edition.

Reference Books:

1. *Vincent C. Müller (Ed.),* The Ethics of Artificial Intelligence and Robotics, *The Stanford Encyclopedia of Philosophy* (Summer 2020 Edition).
2. *Wendell Wallach, Colin Allen*, Moral Machines: Teaching Robots Right From Wrong, Oxford University Press (2008), 1st Edition.

Semester – II (Generative AI)

PAI206: GENERATIVE AI - WORD EMBEDDINGS, TOKENS AND NLP

L	T	P	Cr
3	0	2	4.0

Course Objectives: The objective of this course is to equip students with a comprehensive understanding of the foundational and advanced concepts in Natural Language Processing (NLP) as applied in Generative AI systems.

Introduction: What is Generative AI; overview, definition, and scope of Natural Language Processing (NLP); the role of NLP in Generative AI; applications such as text generation, conversational agents, and content automation; key challenges including linguistic ambiguity, contextual understanding, and scalability with large datasets and models.

Text Preprocessing: Tokenization: word-level, sentence-level, subword tokenization; stop-word removal; lowercasing and punctuation removal; stemming and lemmatization.

Text Representation and Encoding Methods: Bag-of-Words (BoW), Term Frequency-Inverse Document Frequency (TF-IDF), n-grams, and phrase modeling; limitations of sparse vector representations; need for dense word embeddings.

Word Embeddings: Introduction to feedforward neural networks; Word2Vec (CBOW and Skip-gram models); GloVe embeddings; training objectives (e.g., negative sampling, hierarchical softmax); evaluation of word embeddings (intrinsic and extrinsic methods).

Deep Sequential Models in NLP: Overview of Recurrent Neural Networks (RNNs), Gated Recurrent Units (GRUs), and Long Short-Term Memory (LSTM) networks; vanishing gradient problem; sequence modeling for text generation and classification; integration of embeddings into sequential models.

Transformer Networks and Contextual Embeddings: Limitations of static embeddings and the need for contextual representations; ELMo using bidirectional LSTMs; self-attention mechanism; Transformer architecture (encoder-decoder, multi-head attention); positional encodings (sinusoidal vs. learned); contextual embeddings with BERT and its variants; static vs. dynamic embeddings; subword tokenization methods (e.g., BPE, WordPiece).role of contextual embeddings in generative and discriminative NLP tasks.

NLP Frameworks and Open-Source Tools: Introduction to NVIDIA NeMo and HuggingFace Transformers; working with open-source datasets such as the GLUE benchmark; transfer learning, fine-tuning, and model deployment in NLP.

Visualization and Evaluation of Word Embeddings: Dimensionality reduction techniques (PCA, t-SNE); visualization of high-dimensional word vectors; interpreting word clusters and semantic relationships; evaluation of embeddings using analogy tasks and downstream performance; intrinsic vs. extrinsic evaluation metrics.

Applications of NLP: Overview of real-world NLP tasks including information retrieval, intent-slot filling, machine translation, punctuation and capitalization restoration, question answering, relation extraction, sentiment analysis, and token classification using NeMo.

Laboratory Work: Focusing on sustainable model optimization and minimizing the environmental impact during practical tasks like fine-tuning and deployment

Introduction to DL frameworks**:** PyTorch, TensorFlow (Keras), and NVIDIA TLT Toolkit for AI model development.

Text Preprocessing: Tokenization, TF-IDF, stop-word removal, and n-grams for data preparation.

Word Embeddings**:** Implementation of Word2Vec and GloVe for generating dense vector representations.

Sequence Modeling**:** Building RNN, LSTM, and GRU models for text generation and classification tasks.

Generative AI with Transformers**:** Fine-tuning BERT for text generation, summarization, and conversational agents.

Building Chatbots**:** Implementing intent-slot filling and dialogue systems using NeMo's pretrained models.

Text Generation**:** Using GPT-based models for creative text generation and content automation.

NLP Applications**:** Sentiment analysis, relation extraction, and machine translation using generative models.

Model Optimization and Deployment**:** Using TensorRT and Triton Inference Server for optimized model deployment

Course Outcomes (COs)/Course Learning Outcomes (CLOs):

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

1. Understand the fundamentals of Natural Language Processing (NLP) and its role in Generative AI applications.
2. Apply text preprocessing techniques and traditional text representation methods such as Bag-of-Words and TF-IDF.
3. Implement and evaluate word embedding techniques including Word2Vec, GloVe, and contextual models like ELMo and BERT.
4. Analyze and apply deep learning architectures—RNNs, LSTMs, and Transformers—for text classification and generation tasks.
5. Utilize open-source NLP frameworks (e.g., HuggingFace, NVIDIA NeMo) for developing and fine-tuning generative language models.

Text Books

1. Jurafsky, D., & Martin, J. H. (2023). *Speech and Language Processing* (3rd ed.). Pearson.
2. Goyal, P., Pandey, S., & Jain, K. (2021). *Deep Learning for Natural Language Processing*. Apress.

Reference Books

1. Goldberg, Y. (2017). *Neural Network Methods in Natural Language Processing*. Morgan & Claypool Publishers.
2. Bird, S., Klein, E., & Loper, E. (2009). *Natural Language Processing with Python* (2nd ed.). O'Reilly Media.
3. Rothman, D. (2020). *Transformers for Natural Language Processing*. Packt Publishing.

PAI207: LANGUAGE MODELS AND THE TRANSFORMER

L	T	P	Cr
3	0	2	4.0

Course Objectives: The objective of the course is to introduce the foundational concepts, key architectures, training techniques, applications, and recent research directions for LLMs.

Foundations: Introduction to deep learning models, neural networks (ANN, CNN, RNN, LSTM), language models (N-gram model), neural language models (Word2Vec, GloVe)

Transformer Architecture: Encoder-Decoder architecture, positional encodings, multi-head attention, masked multi-head attention, cross-attention, training transformers

Scaling to Large Language Models: Introduction to Pretrained Transformers (BERT, GPT, T5, XLNet), Unsupervised pre training objectives (MLM, CLM), fine-tuning vs prompt based learning, Zero-shot, few-shot, and in-context learning

Sustainable LLM Infrastructure: Distributed Training, Introduction to FSDP (Fully Shared Data Parallel) for Energy efficient training, Memory optimizations (activation checkpointing, quantization) for deployment in low resource hardware (aligns with SDG 9), Inclusive dataset curation at scale (Common Crawl, The Pile)

LLM Applications and Alignment: Code generation, translation, summarization, chatbots, Multimodal LLMs, Prompt Engineering, RLHF (Reinforcement Learning Human Feedback), Bias, Fairness, Hallucination in LLMs

Laboratory work:

Implementation of transformers and LLM architectures focussing on practical, hands-on activities to reinforce learning and align with sustainability, ethics, and real-world relevance

Course Outcomes (COs)/Course Learning Outcomes (CLOs):

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

1. Understand and implement core architectures of modern language models, including transformers and attention mechanisms, and evaluate their performance using standard NLP benchmarks.
2. Apply pretrained transformer models using fine-tuning and prompt-based methods for zero-shot, few-shot, and in-context NLP tasks.
3. Apply scalable and memory-efficient training strategies such as Fully Sharded Data Parallel (FSDP), quantization, and activation checkpointing to optimize large language models for real-world deployment.
4. Critically assess and curate language model training data, identifying and mitigating issues related to bias, misinformation, and representational fairness in large-scale datasets.

Text Books:

1. L. Tunstall, L. von Werra, and T. Wolf, *Natural Language Processing with Transformers*. Sebastopol, CA, USA: O'Reilly Media, 2022.
2. Y. Goldberg, *Neural Network Methods in Natural Language Processing*. San Rafael, CA, USA: Morgan & Claypool, 2017.

Reference Books:

1. I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
2. D. Rothman, *Transformers for Natural Language Processing*. Birmingham, UK: Packt Publishing, 2021.

PAI208: NEURAL ARCHITECTURE DESIGN AND OPTIMIZATION

L	T	P	Cr
3	0	2	4.0

Course Objectives: This advanced course provides students with theoretical foundations and hands-on experience in modern model training techniques for large language models (LLMs) and other neural architectures. Students will learn the complete pipeline from pre-training and instruction tuning to parameter-efficient fine-tuning methods, with emphasis on responsible AI practices aligned with the UN Sustainable Development Goals.

Introduction to Model Training: Pre-training, fine-tuning, instruction-following: definitions and distinctions, Dataset design, tokenization, vocabulary creation, Ethical considerations and environmental impact of large model training.

Pre-Training Techniques: Self-supervised learning: MLM, CLM, contrastive objectives, Architectures and training frameworks: Transformers, Deepspeed, ZeRO, Optimization with mixed precision, pruning, and sparse training.

Instruction Tuning and Alignment: RLHF, DPO, alignment with human feedback, Challenges in alignment: bias, fairness, and safety

Parameter-Efficient Fine-Tuning (PEFT): LoRA, Adapters, BitFit, QLoRA, Comparing PEFT vs full fine-tuning in resource-constrained settings, Edge adaptation and low-resource training.

Applications and Case Studies: Domain-specific instruction tuning (healthcare, education, agriculture), Use of open-weight models (e.g., LLaMA, Mistral).

Laboratory work:

1. Creating efficient training datasets and implementing custom tokenizers
2. Pre-training a small language model from scratch with performance optimization
3. Fine-tuning an open-weight model using RLHF or DPO techniques
4. Implementing and comparing multiple PEFT methods (LoRA, QLoRA, Adapters)
5. Building a domain-specific instruction-tuned model with Hugging Face's PEFT library
6. Measuring environmental impact using CodeCarbon and optimizing for efficiency
7. Deploying fine-tuned models as APIs with Gradio/Streamlit interfaces

Course Outcomes (COs)/Course Learning Outcomes (CLOs):

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

1. Design and implement efficient model training pipelines from scratch
2. Apply instruction tuning techniques with focus on safety and alignment
3. Evaluate and optimize models for computational efficiency and sustainability
4. Develop domain-specific adaptations using parameter-efficient methods
5. Implement responsible AI practices throughout the model development lifecycle

Text Books:

1. Chollet, F. (2021). *Deep Learning with Python*, 2nd ed. Manning Publications.
2. Vaswani, A., et al. (2023). *Transformers for Natural Language Processing*, 2nd ed. Packt Publishing.
3. Tunstall, L., von Werra, L., & Wolf, T. (2022). *Natural Language Processing with Transformers*. O'Reilly Media.

Online Resources

1. Hugging Face Documentation: https://huggingface.co/docs
2. DeepSpeed Documentation: https://www.deepspeed.ai/
3. EleutherAI Language Model Resources: https://github.com/EleutherAI
4. Responsible AI Practices (Google): https://ai.google/responsibilities/responsible-ai-practices/

PAI209: LLM SCALING AND SCALING MODEL TRAINING TO DISTRIBUTED WORKLOADS

L	T	P	Cr
3	0	2	4.0

Course Objectives: The primary objective of this course is to provide students with theoretical foundations and hands-on experience in scaling large language models (LLMs) and training them efficiently across distributed systems, covering parallelism strategies, resource optimization, and infrastructure design.

Introduction to LLM Scaling: Evolution of model sizes: from BERT to GPT-4 and beyond**,** Scaling laws for LLMs**,** Memory and compute bottlenecks in large model, Hardware trends (GPUs, TPUs, accelerators) and their implications

Parallelism Strategies in LLM Training: Data parallelism vs. model parallelism**,** Pipeline parallelism and tensor parallelism (Megatron-LM, DeepSpeed)**,** Zero Redundancy Optimizer (ZeRO) and memory optimization**,** Sharded training and parameter offloading

Distributed Training Infrastructure: Training infrastructure: clusters, Kubernetes, Ray, Slurm**,** Use of frameworks: PyTorch Distributed, DeepSpeed, Hugging Face Accelerate**,** Fault tolerance, checkpointing, and resume strategies**,** Real-world case studies: training GPT-J, BLOOM, LLaMA

Performance Tuning and Cost Optimization: Profiling tools (NVIDIA Nsight, PyTorch profiler)**,** Mixed precision and quantization-aware training**,** Batch size tuning, gradient accumulation**,** Spot vs. on-demand pricing in cloud training

Efficient Pretraining & Instruction Fine-Tuning: Corpus preparation and tokenizer scaling**,** Training from scratch vs. continual pretraining**,** Instruction tuning: FLAN, Alpaca, Dolly**,** Parameter-Efficient Fine-Tuning (PEFT): LoRA, adapters, prefix tuning

LLMs in Production and Edge Deployment: Model compression: distillation, quantization, pruning**,** Inference optimizations using ONNX, TensorRT, vLLMs**,** Multi-GPU/multi-node inference serving (vLLM, TGI, DeepSpeed-Inference)**,** Challenges in scaling inference across devices or microservices

Laboratory Work:

Students will work with LLMs using frameworks like DeepSpeed, Hugging Face Accelerate, and PyTorch DDP to implement distributed training, model sharding, and inference scaling. Labs include hands-on with Megatron-LM, ZeRO optimization, and deploying a quantized LLM on a multi-GPU cluster.

Course Outcomes (COs)/Course Learning Outcomes (CLOs):

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

1. Understand the technical challenges and principles involved in scaling LLMs.
2. Implement and optimize distributed training strategies across multiple hardware devices.
3. Evaluate and apply PEFT methods for cost-efficient fine-tuning.

4. Design and deploy inference-optimized LLMs in real-world distributed environments.
5. Apply principles of energy-efficient AI training and deployment.

Text Books:

1. Shoeybi, M., et al. (NVIDIA), "Megatron-LM: Training Multi-Billion Parameter Language Models Using Model Parallelism", 2020.
2. Hugging Face, "Transformers and Accelerate Documentation", Hugging Face, 2023.
3. Jared Kaplan et al., "Scaling Laws for Neural Language Models", 2020.

Reference Books:

1. Thomas Wolf, "Natural Language Processing with Transformers", O’Reilly, Revised Edition, 2022.
2. Yuxin Wang, Jeff Rasley et al., "DeepSpeed: Accelerating Large Scale Model Training", Microsoft Research, 2021.
3. PaLM, GPT, and LLaMA technical reports and blog posts from Google, OpenAI, Meta.

PAI210: DIFFUSION MODELS IN GENERATIVE AI

L	T	P	Cr
3	0	2	4.0

Course Objective: This course provides an in-depth study of diffusion-based generative models, which have recently emerged as a powerful alternative to GANs and VAEs. Students will explore the mathematical underpinnings, architectures, training and sampling techniques, and applications across modalities like images, text, audio, and 3D. The course includes hands-on projects to design and implement diffusion models using modern AI frameworks, enabling learners to understand both theory and practice.

Foundations of Generative Models: Overview-Autoregressive models, VAEs, GANs, Normalizing Flows. Diffusion Basics-Forward/reverse diffusion, Markov chains, noise scheduling. Denoising Diffusion Probabilistic Models (DDPM) framework.

Evaluation Metrics & Robustness: Metrics- Fréchet Inception Distance (FID), Kernel Inception Distance (KID), Inception Score (IS), Learned Perceptual Image Patch Similarity (LPIPS) and Structural Similarity Index Measure (SSIM). Robustness- Overfitting, generalization, bias detection.

Architectures: Core architecture-U-Net with skip connections, time-step embeddings, positional encodings. Advanced architectures- Diffusion Transformers (DiT), Latent Diffusion Models (LDM).

Optimization & Compression: Training optimization - noise schedule optimization, mixed-precision training, gradient checkpointing, knowledge distillation. Model Compression-Pruning, quantization, efficient attention (e.g., FlashAttention).

Sampling : Sampling Approaches – Stochastic sampling, deterministic sampling , predictor-corrector approaches. Efficiency enhancements - Reduced-step sampling, fastDPM, trade-offs between speed and fidelity.

Conditional and Multimodal Generation: Conditional models - Class conditional models, text-based conditioning (CLIP, BERT, T5), cross-attention mechanisms. Multi-modal fusion-use of cross-attention for integrating multi-modal inputs (like text, sketches, and masks), notable models (GLIDE, DALL-E 2).

Applications of Diffusion Models: Audio applications - WaveGrad, DiffWave, Text-to-Speech (TTS). Vision applications - Image synthesis, Image Editing and Enhancement. Video and 3D Applications - Video Generation, 3D Generation and Modeling.

Laboratory Work: – Students will implement core components of diffusion models using Python and PyTorch or TensorFlow.

Course Learning Outcomes (CLOs) / Course Objectives (COs):
After the completion of the course, the student will be able to:

1. Explain the foundational principles of generative modeling and diffusion processes.
2. Analyze and implement the architecture and training techniques of diffusion models.

3. Evaluate various sampling and inference methods for diffusion-based generation.
4. Design conditional and multi-modal diffusion models for real-world applications.
5. Apply diffusion models to domain-specific problems in vision, language, and audio.

Textbooks

1. Foster, D. (2023). Generative Deep Learning (2nd ed.). O'Reilly Media.
2. Sanseviero, O. et al. (2023) Hands-On Generative AI with Transformers and Diffusion Models. O'Reilly Media.

Reference Books

1. Bhati, D. et al. (2025) A Beginner’s Guide to Generative Ai: An Introductory Path to Diffusion Models, Chatgpt and LLMs. Springer-Nature New York Inc.
2. Vemula, A. (2024) Diffusion Models: Practical Guide to AI Image Generation.

Semester – II (AI For Science)

PAI220: LINEAR ALGEBRA

L	T	P	Cr
3	0	2	4.0

Course Objectives: This course aims to develop students' understanding of advanced mathematical concepts used in Artificial Intelligence. Students will learn to apply calculus, linear algebra, and optimization techniques that support AI model development, with real-world use cases.

Vector Spaces: Vector spaces and subspaces, null space, column space and linear transformations, linear independent sets, bases, dimension, coordinate systems, rank, change of basis.

Eigenvalues and Eigenvectors: Eigenvalues and Eigenvectors, characteristic equation, diagonalization, eigen vectors and linear transformations.

Orthogonality and Least Squares: Inner product, length, orthogonality, orthogonal sets, orthogonal projections, the Gram-Schmidt process, least square problems, applications to linear models and sustainability.

Symmetric Matrices and Quadratic Forms: Diagonalization of symmetric matrices, quadratic forms, constrained optimization, singular value decomposition, application to image processing and statistics for sustainable practices.

Optimization: Matrix games, linear programming-geometric method, linear programming – simplex method, duality.

Finite State Markov Chains: Introduction and examples, the steady state vector and Google's page rank, communication classes, classification of states and periodicity, fundamental matrix, Markov chains and baseball statistics.

Laboratory work:
Implementation of various mathematical concepts and basic AI algorithms using python or MATLAB.

Course Outcomes (COs)/Course Learning Outcomes (CLOs):
After completion of this course, the students will be able to:

1. Apply mathematical concepts such as vector norms, gradients, and matrix operations to formulate and solve problems in AI.
2. Analyze eigenvalues, eigenvectors, and their role in diagonalization and linear transformations
3. Evaluate least square solutions using orthogonality principles, Gram-Schmidt process.
4. Apply optimization problems and Markov chains to real-world systems

Textbooks:

1. Deisenroth, M., Faisal, A., Ong, C. S., Mathematics for Machine Learning, Cambridge University Press (2020).
2. Strang, G., Introduction to Linear Algebra, Wellesley-Cambridge Press, 6th Edition (2023).
3. Lay, D.C., Lay, S.R., McDonald, J.J., Linear Algebra and its Applications, Pearson, 5th Edition, (2015).

Reference Books:

1. Boyd, S., Vandenberghe, L., Convex Optimization, Cambridge University Press (2004).
2. Hastie, T., Tibshirani, R., Friedman, J., Elements of Statistical Learning, Springer (2009).

PAI211: PHYSICS INFORMED NEURAL NETWORKS

L	T	P	Cr
3	0	2	4.0

Course Objectives: The course enables learners to equip students with the knowledge of integrating deep learning with structured physical constraints using Physics-Informed Neural Networks (PINNs) and emphasize core neural network concepts, architectures, and optimization techniques applicable in scientific computing.

Deep Learning Foundations for Scientific Applications: Review of neural networks: MLPs, CNNs, RNNs, Activation functions, backpropagation, and optimization, Role of AI in computational science, Introduction to Green AI and energy-efficient model design

Introduction to Physics-Informed Machine Learning: Motivation and principles of physics-informed learning. Loss formulation: data-driven vs physics-constrained, Governing equations: ODEs, PDEs, boundary and initial value problems, Overview of scientific machine learning, Automatic differentiation and residual learning.

Architectures and Learning Strategies: Feedforward neural networks and function approximation, Fully Connected Networks, Fourier Neural Operators, DeepONets, Loss functions: data loss, physics loss, boundary condition loss. Solving forward problems with known physics.

Inverse Problems and Parameter Discovery: PINNs for identifying unknown parameters and hidden states. Noise-aware and regularized training strategies, Transfer learning, multi-task learning.

Advanced Architectures and Training Techniques: Solving 1D/2D problems using TensorFlow/JAX, Domain decomposition and adaptive sampling. Multi-fidelity PINNs, transfer learning, and curriculum learning. Fourier Neural Operators, DeepONet, and operator learning frameworks.

Challenges and Emerging Directions: Computational challenges and scalability. Generalization and robustness in noisy real-world data. Hybrid modeling with classical solvers.

Laboratory work:
Implement basic PINNs in Python using TensorFlow and JAX, DeepXDE. Solve 1D and 2D PDEs, Inverse modeling of unknown coefficients using PINNs. Compare PINN solutions with classical element solvers.

Course Outcomes (COs)/Course Learning Outcomes (CLOs):
After completion of this course, the students will be able to:

1. Understand and implement Physics-Informed Neural Networks.
2. Formulate and solve constrained optimization problems using PINNs
3. Analyze trade-offs between accuracy, efficiency, and sustainability in scientific AI.

4. Apply PINNs to real-world applications in engineering and science.

Text Books:

1. S. Haykin, “Neural Networks: A comprehensive foundation”
2. V. Sankhala, “Physics informed Neural Networks and Biologically inspired Machine Learning: Vikram Sankhala Lecture Notes.”

PAI212: NEURAL OPERATORS

L	T	P	Cr
3	0	2	4.0

Course Objectives:

Introduce neural operators as function-to-function learners and contrast with traditional neural networks. Equip students with skills to implement operator-based ML models. Enable application of neural operators in sustainability-oriented simulations (climate, energy, etc.). Foster interdisciplinary problem-solving skills involving AI and scientific computing.

Introduction to Neural Operators: Motivation: Beyond finite-dimensional ML, Operator learning vs function approximation, Applications of neural operators, Function spaces, Introduction to Partial Differential Equations (PDEs), Introduction to numerical solvers and discretization, Classical Approaches vs ML Approaches

Introduction to DeepONets: Basics of surrogate modelling, DeepONet architecture, Universal approximation of operators, Spectral learning and efficiency, Fourier transforms and convolution

Neural Operators on Irregular Domains: Graph Neural Networks (GNNs) and Mesh-based models, Working with real-world domain data (e.g., terrain, buildings)

Training and Optimization: Loss functions for operator learning, Hyperparameter tuning, regularization, Working with sparse, noisy, or simulated data

Applications in Climate and Energy: FNOs for climate downscaling, Renewable energy forecasting (wind, solar, tidal), Sustainable infrastructure modeling (water/air quality), Responsible engineering for sustainability

Transfer Learning and Generalization: Multi-domain generalization, Data efficiency and fine-tuning, Model compression and inference optimization, Emerging trends in neural operators (transformers, hybrid models)

Laboratory Work:

Implementation of DeepONets, Graph Neural Networks, Sustainiability project to be solved using various neural operators

Course Outcomes (Cos)

1. Explain the theoretical foundations and architectural principles of neural operators, and differentiate them from traditional machine learning models
2. Design and implement neural operator models including DeepONets.
3. Evaluate the performance of neural operator-based models on real-world and simulated datasets, especially in domains such as climate modeling, renewable energy forecasting, and sustainability.
4. Analyze the mathematical and theoretical foundations of neural operators, including their function space mappings, approximation capabilities, and generalization properties across domains.

Text Books:

1. Scientific Machine Learning: Theory and Applications by J. Nathan Kutz, Cambridge University Press
2. Deep Learning for Scientific Computing by Jan S. Hesthaven, SIAM (Society for Industrial and Applied Mathematics), 2023

Reference Books:

1. Numerical Solution of Partial Differential Equations by the Finite Element Method by Claes Johnson, Dover Publications
2. Deep Learning for the Life Sciences: Applying Deep Learning to Genomics, Microscopy, Drug Discovery, and More by Bharath Ramsundar, Peter Eastman, Patrick Walters, Vijay Pande, O'Reilly Media

PAI213: DATA AND UNCERTAINTY QUANTIFICATION

L	T	P	Cr
3	0	2	4.0

Course Objectives

This course aims to provide foundational and advanced knowledge of uncertainty quantification (UQ) and probabilistic data modeling techniques essential for AI and machine learning systems. It emphasizes sustainability-aware decision-making under uncertainty and responsible AI system design aligned with global sustainable development goals (SDGs).

Introduction to Uncertainty Quantification: Types of Uncertainty: Aleatoric and Epistemic, Relevance in AI Systems and Real-world Applications, Role of UQ in Sustainable and Safe AI Systems (SDG 12, SDG 13)

Probability and Statistical Foundations: Random Variables, Probability Distributions, Expectation, Joint, Marginal, and Conditional Probabilities,Bayesian Inference and Bayes' Theorem, Maximum Likelihood Estimation (MLE) and MAP Estimation

Monte Carlo and Sampling Methods: Random Sampling, Importance Sampling, Markov Chain Monte Carlo (MCMC) and its Applications, Bootstrapping and Resampling Techniques

Uncertainty in Machine Learning Models: Probabilistic Machine Learning, Confidence Intervals and Prediction Intervals, Bayesian Neural Networks and Dropout as Bayesian Approximation, Gaussian Processes

Model Validation and Uncertainty Propagation: Model Calibration, Uncertainty Propagation through ML Pipelines, Sensitivity Analysis and Reliability Engineering

Data-Driven Sustainability Applications: Climate Modeling and Prediction with UQ (SDG 13), Risk-aware Smart Grid and Renewable Energy Forecasting (SDG 7), UQ in Environmental Monitoring and Disaster Management (SDG 11, 12)

Ethics and Responsible AI with UQ: Fairness, Accountability, and Transparency in AI under Uncertainty (SDG 16), Interpretable Models and Ethical Considerations

Laboratory Work

- Implement Bayesian inference on synthetic and real datasets
- Apply Monte Carlo methods to estimate integral and risk measures
- Build probabilistic models using PyMC3 or TensorFlow Probability
- Use UQ in climate simulation models (SDG 13)
- Design and analyze uncertainty-aware AI models for sustainable energy systems (SDG 7)

Course Outcomes (COs)

1. Understand the types and sources of uncertainty in data and models
2. Apply probabilistic and Bayesian methods to quantify and propagate uncertainty
3. Design interpretable and uncertainty-aware AI models for critical applications
4. Incorporate sustainability goals in AI modeling through robust uncertainty analysis

Textbooks

1. Sudret, B., & Soize, C. (2017). Uncertainty Quantification: Theory, Implementation, and Applications. Springer.

2. Murphy, K. P. (2022). Probabilistic Machine Learning: An Introduction. MIT Press.

Reference Books

1. Smith, R. C. (2013). Uncertainty Quantification: Theory, Implementation, and Applications. SIAM.
2. Bishop, C. M. (2006). Pattern Recognition and Machine Learning. Springer.
3. Huyen, C. (2022). AI Engineering: Building Applications with Foundation Models. Snorkel AI Press

PAI214: HPC AND PHYSICS NEMO FRAMEWORK

L	T	P	Cr
3	0	2	4.0

Course Objectives: This course introduces students to the design, development, and performance analysis of high performance applications using modern CPU and GPU architectures. Emphasis is placed on writing parallel programs, optimizing performance, and understanding the underlying hardware-software interplay.

Introduction to High Performance Computing and Parallel Architectures: Introduction to HPC and its applications in science and engineering, Understanding Flynn's Taxonomy and the different types of parallel computing: SISD, SIMD, MISD, MIMD, Overview of shared and distributed memory system, Processor performance, memory hierarchy, and multi-core architectures. Parallel algorithm design, Performance metrics for parallel systems; Introduction to I/O-avoiding algorithms and cache-oblivious algorithms; Work-Span Model and its relevance to parallel algorithm analysis; Communication in Parallel Systems. Case studies in HPC applications for environmental simulations, renewable energy systems optimization, and climate prediction models.

Shared Memory Programming with OpenMP: Introduction to OpenMP and its parallel programming model; Using OpenMP for multi-threading and parallel loop execution; Data sharing and synchronization in OpenMP; Efficient OpenMP programming for matrix operations; Advanced OpenMP concepts for large-scale parallel applications. Implementing OpenMP-based parallel solutions for clean energy modeling, healthcare diagnostics (e.g., biomedical image processing), and disaster simulation systems.

Distributed Memory Programming with MPI: Introduction to MPI and message-passing programming models; MPI point-to-point communication (send/receive), collective communication (broadcast, scatter, gather); Domain decomposition and parallelization of matrix solvers using MPI; Hybrid MPI/OpenMP approaches for large-scale parallel applications; MPI routines for parallel matrix solvers and other numerical methods. Hybrid programming (MPI+OpenMP, MPI+MPI). Applications of MPI in urban data analytics, smart agriculture systems, and modeling pandemic spread for public health planning.

Parallel Algorithms and Applications: Dense Matrix Algorithms, Sorting and Search Algorithms, Graph Algorithms, Minimum Spanning Tree, Shortest Paths, Dynamic Programming and Optimization, Fourier Transform and Fast Algorithms. Applying graph algorithms in optimizing energy grids, routing in sustainable transport networks, and modeling biological ecosystems.

GPU Computing with CUDA and OpenACC: Introduction to General-Purpose GPU (GPGPU) computing and CUDA programming; Thread execution and scheduling in CUDA; Matrix multiplications in CUDA; Introduction to OpenACC programming for GPU acceleration; Multi-GPU programming and exascale computing techniques. Green computing techniques in GPU programming (energy-efficient kernels); GPU acceleration in carbon footprint modeling, air pollution analysis, and renewable energy simulations.

PhysicsNeMo Framework: Introduction to NeMo and PhysicsNemo, PhysicsNemo Components, PhysicsNeMo APIs (Model, Datapipes, Metrics, Deploy, Utils), Architectures (Neural operators, Graph Neural Networks and Recurrent Neural Networks), PhysicsNeMo Symbolic, Industrial Applications and Examples: CFD, Weather and Climate, Generative, Healthcare, Additive Manufacturing,Molecular Dynamics. Applications in sustainable manufacturing, climate prediction, healthcare systems, and modeling energy-efficient infrastructure. Demonstrations of NeMo for simulations aligned with global sustainability efforts.

Laboratory work:

Implementation of Parallel Programming with OpenMP, Distributed Memory Programming with MPI. GPU Computing with CUDA. PhysicsNeMo Framework for Scientific Applications

Course Outcomes (COs)/Course Learning Outcomes (CLOs):

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

1. Understand and Implement Parallel Architectures
2. Design, Develop, and Optimize Parallel Programs
3. Analyze Performance of High-Performance Systems
4. Implement and Deploy Scientific Applications using HPC and PhysicsNeMo

Text Books:

1. Grama, A., Gupta, A., Karypis, G., and Kumar, V., Introduction to Parallel Computing, Addison Wesley, 2003
2. Gropp, W, Ewing L, and Anthony S. Using MPI: portable parallel programming with the message-passing interface. Vol. 1. MIT press, 1999.
3. Cook, S., CUDA Programming: A Developer's Guide to Parallel Computing with GPUs, M K Publishers, 2012
4. NVIDIA, CUDA C Programming guide, 2012
5. NVIDIA PhysicsNeMo, https://developer.nvidia.com/physicsnemo.

Electives-I

PAI215: QUANTUM COMPUTING

L	T	P	Cr
2	0	2	3.0

Course Objectives: The objective of this course is to provide the students an introduction to quantum computation and quantum machine learning after covering the concepts of linear algebra, vector space and quantum mechanics.

Mathematics and Quantum Mechanics foundation: Linear and complex vector space, Hilbert spaces (finite dimensional), Tensor Products, Dirac's notation, Probabilities and measurements, Postulates of quantum mechanics, Measurements in bases other than computational basis, Introduction of qubit, Bloch sphere representation of qubit, Quantum Superposition and entanglement, Super-dense coding, teleportation, Density operators, Euler identity.

Quantum Computing: classical gates, Single qubit gates, multiple qubit gates, quantum gates, universal quantum gates, Quantum circuits, design of quantum circuits, Energy efficiency concepts in quantum computing, Recent developments in industry and hardware development, Government Initiatives in quantum computing.

Quantum Algorithms: Deutsch's algorithm, Grover algorithm, Shor's factoring algorithm.

Quantum Machine Learning: Quantum Data and Feature Encoding, Variational Quantum Circuits (VQCs) and Hybrid Quantum-Classical Models, Quantum-enhanced Learning Algorithms

Quantum Cryptography and Error Correction: Basic concept of quantum cryptography with discussion of quantum protocol like BB84, basic discussion on Quantum error correction, bit and phase flip error.

Lab: Implementation of Quantum concepts in any quantum simulator.

Course learning outcomes (CLOs):

After the completion of the course, the student will be able to:
- Comprehend the basic concepts of quantum computing.
- Illustrate the concepts of quantum gates and quantum circuits.
- Apply the concept of quantum computing in designing of quantum algorithms.
- Acquire basic knowledge of cryptography protocols and quantum machine learning.

Text Books:

1. *NielsenM. A., Chuang I. L., Quantum Computation and Quantum Information, Cambridge University Press (2010) 10th Anniversary ed.*
2. *Benenti G, Casati G., Strini G., Principles of Quantum Computation and Information, Vol. I: Basic Concepts, Vol II: Basic Tools and Special Topics, World Scientific (2007*

Reference Books:

1. *PeresA., Quantum Theory: Concepts and Method, Kluwer Academic Publishers (2002) 1st ed.*
2. *Yanofsky N. S., Mannucci, M. A., Quantum Computing for Computer Scientists, Cambridge University Press, 2008.*

PAI216: INTRODUCTION TO GEN AI

L	T	P	Cr
2	0	2	3.0

Course Objectives: The objective of this course is to provide students with foundational knowledge and practical skills in Generative Artificial Intelligence. It covers the basic concepts, models, and tools used in generative AI, including deep learning-based generation, diffusion models, responsible use of AI technologies, and sustainability considerations in AI systems.

Introduction to Generative AI: Definition, History, and Applications. Comparison between Discriminative and Generative Models. Real-world applications: Art, Music, Content Creation, Code Generation, Chatbots. Understanding the energy footprint of AI systems and the need for efficient models.

Deep Learning and NLP Foundations: Neural Networks Basics: Perceptrons, Activation Functions, Forward and Backpropagation. Introduction to Deep Neural Networks and basics of Convolutional Neural Networks. Loss Functions, Optimization Techniques (SGD, Adam). Basics of Training and Overfitting. NLP basics (tokenization, embeddings), Introduction to lightweight models and trade-offs between accuracy and computational efficiency.

Generative Models: Autoencoders and Variational Autoencoders (VAEs): Architecture, Encoding-Decoding, Latent Space. Generative Adversarial Networks (GANs): Generator and Discriminator, Adversarial Loss, Training Challenges. Applications of VAEs and GANs. Discussion on resource consumption in training large generative models and optimization for greener computing.

Transformers, LLM, and RAG: Attention Mechanism, LLM Architecture, Fine-tuning LLMs (very basic intro), RAG system basics. Introduction to Diffusion Models: Forward and Reverse Processes, Training and Sampling. Applications in Image and Text Generation. Model compression and transfer learning as sustainable practices, smart retrieval reducing computational costs.

Ethical AI: Understanding AI Hallucinations, Model Bias, Fairness, Explainability, Societal Impacts and Mitigation Strategies. Ethical responsibility for carbon-aware computing and responsible AI deployment in high-impact sectors.

Laboratory Component: Students will implement core generative AI models (VAEs, GANs, Diffusion Models) and explore applications of Transformers and Retrieval-Augmented Generation (RAG) through hands-on mini-projects focusing on sustainable and ethical AI practices.

Course Outcomes (COs)/Course Learning Outcomes (CLOs):

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

1. Explain the fundamental concepts of Artificial Intelligence, Machine Learning, and Generative AI, and distinguish between discriminative and generative models along with their real-world applications.

2. Apply foundational knowledge of neural networks, deep learning, and natural language processing (NLP) to understand and implement basic generative models.
3. Analyze the architecture and applications of advanced generative technologies such as Transformers, Large Language Models (LLMs), Retrieval-Augmented Generation (RAG), and Diffusion Models with attention to computational efficiency and sustainability.
4. Evaluate the ethical considerations, biases, and societal impacts of generative AI technologies, emphasizing responsible and sustainable development and deployment practices.

Text Books:

1. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow by Aurélien Géron (2nd or 3rd Edition)Aurélien Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, O'Reilly Media (2019).

2. Deep Learning with Python by François Chollet (2nd Edition)

3. Foster, D. (2019). *Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play*. O'Reilly Media.

Reference Books:

1. Sebastian Raschka, *Machine Learning with PyTorch and Scikit-Learn*, Packt Publishing (2022).

2. Kate Crawford, *Atlas of AI: Power, Politics, and the Planetary Costs of Artificial Intelligence*, Yale University Press (2021).

PAI217: DIGITAL TWINS

L	T	P	Cr
2	0	2	3.0

Course Objectives: The prime objective of this course is to introduce the fundamental concepts of Digital Twins—virtual representations of physical systems used for simulation, monitoring, and optimization. It explores applications in smart cities, energy, manufacturing, and sustainability. The course integrates computer science concepts with modeling, IoT, data analytics, and system integration to build practical and theoretical skills in building and deploying Digital Twin systems.

Fundamentals of Digital Twins
Introduction to Digital twin; Basic concepts of Digital twins; Growth drivers for digital twin; Digital Twin Terminologies & Essentials; Digital Thread; Digital Shadow; Components: Physical Entity, Digital Model, Data Flow; Digital Twin Frameworks.

Architecture and Enabling Technologies
Digital Twin architecture: data layer, modeling layer, service layer; Types of Digital Twin: Based on Product and Process, Based on Functionality, Based on Maturity; Enabling technologies for Digital Twin like Artificial Intelligence (AI), Machine Learning (ML), Deep Learning (DL), Big Data Analytics, Internet of Things (IOT)

Modeling and Simulation
System modeling techniques: discrete event, continuous, hybrid; Introduction to tools: MATLAB/Simulink; Simulation integration with real-time data

Digital Twin Data: Types and Analytics
Digital Twin Data Types and Management: a. Geometric, behavioral data, b. Historical, Synthetic, and real-time data, c. Data acquisition, storage, and processing; Data Analytics and Insights: Descriptive analysis, Diagnostic analysis, Predictive analysis;

Data Analytics and AI in Digital Twins
Data acquisition, storage, and preprocessing; Predictive maintenance using ML; Anomaly detection and forecasting; Use of AI/ML for system optimization; Use Case Applications: Smart Cities, Renewable energy systems, Water distribution and waste management.

Challenges and Ethical Considerations
Anticipating challenges in digital twin implementation; Scalability, security, and integration with legacy systems; Data privacy and cybersecurity; System interoperability and standardization; Lifecycle and scalability of digital twins; Environmental impact and digital sobriety.

Laboratory work:
Development of various simplified digital twins using simulation tools and real-time data.

Course Outcomes (COs)/Course Learning Outcomes (CLOs):
After completion of this course, the students will be able to:

1. Define the basic characteristics and architecture of Digital Twins.
2. Apply modeling and simulation tools to build Digital Twins.

3. Integrate real-time data and AI/ML for predictive analysis.
4. Explore sustainability applications through Digital Twins in smart cities, energy, and resource optimization.
5. Evaluate challenges and ethical considerations in deploying Digital Twin technologies.

Text Books:

1. Digital Twin: Possibilities of the new Digital twin technology, Anand Iyer, 2017, 35 Pages
2. Digital Twin Development & Deployment on the Cloud, Ist edition, Nassim Khaled Bibin Pattel Affan Siddiqu, ISBN: 9780128216316, ELSEVIER, pages 592
3. Digital Twin Technologies & Smart Cities, Maryam Farsi, Alireza Daneshkhah, Amin Hosseinian-Far, Hamid Jahankahani, Springer, ISBN 978-3-030-18731-6
4. Advances in Computers, The Digital Twin Paradigm for Smarter Systems and Environments: The Industry, Pethuraj & Preetha Evanjaline, ELSEVIER, pages 257, ISBN 978-0-12-818756-2, ISSN 0065-2458

Reference Books:

1. Digital Twin Driven Smart Design by Fei Tao, Ang Liu, Tianliang Hu, A.Y.C. Nee, ELSEVIER, ISBN 978-0-12-818918-4, Pages 333
2. Handbook Of Digital Enterprise Systems: Digital Twins, Simulation and Ai, by Wolfgang Kühn, world scientific publishing co., ISBN 978-981-120-073-1, Pages 229.
3. Digital Twin Complete Self-Assesment Guide, 1976302927, 9781976302923sment Guide, Geradus Blokdyk, CreateSpace Independent Publishing Platform, 2017, Pages 120.

PAI218: MULTIMODAL LEARNING AND APPLICATIONS

L	T	P	Cr
2	0	2	3.0

Course Objectives:

Understand foundational concepts of multimodal learning, covering text, vision, audio, and sensor modalities. Apply multimodal learning models to real-world applications such as healthcare, smart cities, education, and climate monitoring. Emphasize sustainability-driven applications such as environmental monitoring, climate action, and sustainable urban mobility.

Introduction to Multimodal Learning: Definition of multimodal learning: formal concepts and real-world examples. Importance of Enhancing AI robustness through multimodal fusion. Sources of modality-specific data: characteristics of text, image, audio, video, and sensor signals, Early Fusion Techniques, Late Fusion Techniques, Hybrid Fusion Strategies.

Mathematical Foundations: Multimodal representations: Joint and Coordinated representations, Feature extraction and embedding techniques: Text: Word2Vec, GloVe, BERT embeddings. Image: Convolutional Neural Networks (CNN) feature maps. Audio: MFCC (Mel-frequency cepstral coefficients) features. Dimensionality reduction (PCA, t-SNE for multimodal data)

Model Architectures

Deep learning approaches for multimodal data: CNNs for image and video modalities, RNNs and Transformers for sequential data modalities, Multimodal attention mechanisms and cross-modal learning, Co-attention networks for VQA tasks.

Applications of Multimodal Learning

Natural Language Processing with vision (e.g., Visual Question Answering, Image Captioning) Speech and vision integration for smart assistants, Multimodal biometric authentication, Healthcare Diagnostics: Combining medical images, patient histories, and sensor outputs for early detection and personalized treatment, Sustainable Smart Cities: AI-driven traffic and pollution control using multimodal data streams.

Sustainability in Multimodal Systems

Energy-efficient architectures for multimodal learning, Ethical and fair use of multimodal data for social good, Multimodal AI for responsible consumption, disaster resilience, and environment-friendly solutions.

Laboratory Work:

1. Implement early, late, and hybrid fusion strategies on benchmark datasets.
2. Build a Multimodal Sentiment Analysis model integrating text and audio.
3. Design a prototype for smart urban waste management.
4. Develop a project on multimodal disaster response.
5. Train and evaluate a Visual Question Answering (VQA) system.

Course Outcomes (COs)/Course Learning Outcomes (CLOs):

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

1. Understand core principles and methods of multimodal learning.
2. Design and implement deep multimodal fusion models.
3. Critically evaluate multimodal models for fairness, scalability, and environmental impact.
4. Integrate sustainability principles into the design and deployment of multimodal AI systems aligned with global SDGs.

Textbooks:

1. Huyen, C. (2022). AI Engineering: Building Applications with Foundation Models. Snorkel AI Press.
2. Bengio, Y., Goodfellow, I., Courville, A. (2017). Deep Learning. MIT Press.

Reference Books:

1. Giannakos, M., Spikol, D., et al. (2022). The Multimodal Learning Analytics Handbook. Springer.
2. Nika, M. (2023). Building AI-Powered Products: The Essential Guide to AI and GenAI Product Management. Wiley.
3. Alammar, J. (2024). Hands-On Large Language Models: Language Understanding and Generation. O'Reilly Media.

PAI219: ACCELERATED COMPUTING

L	T	P	Cr
2	0	2	3.0

Course Objectives: This course will provide students with fundamental knowledge of GPU Computing to implement Machine Learning and Deep Learning Algorithms.

Parallel Programming: GPU Programming, CUDA C/C++/Python, Numba.

Introduction to Accelerated Machine Learning - Introduction to Supervised and Unsupervised Learning, RAPIDS acceleration for Linear and Polynomial Regression, Decision Tree, Random Forest, Bagging, Boosting, Kmeans, SVM.

Deep Learning Model Compression: Introduction to model pruning, types of model pruning, float 16 and int8 quantization, post training quantization, quantization aware training, mixed precision training, knowledge distillation, feature distillation and relation distillation.

Optimization Framework: Using TensorRT optimization, Deploying model on Triton Inference server.

Laboratory Work:

● CUDA C/C++ for Accelerated Computing.
{DLI Online Course Section: Fundamentals of Accelerated Computing with CUDA C/C++}
● Numba to compile CUDA kernels for Numpy Acceleration in Python.
{DLI Online Course Section: Fundamentals of Accelerated Computing with CUDA Python}
● Getting started with Accelerated Data Science with RAPIDS AI (cuPy, cuDF, cuSignal, cuML).
{DLI Online Course Section: Fundamentals of Accelerated Data Science with RAPIDS}
● Model Pruning with DL Frameworks
● Post Training Quantization and Quantization Aware Training
● Model Distillation

Course Learning Outcomes (CLOs) / Course Objectives (Cos):

After the completion of the course the student will be able to:

1. Demonstrate ability to deploy GPU accelerated code for image processing applications
2. Apply model compression techniques while deploying deep learning architectures.
3. Implement Rapids AI framework for different machine learning tasks.
4. Analyze and evaluate performance of deep learning based inference models using TensorRT optimization and Triton Server.

Text Books:

1. Mitchell M., T., Machine Learning, McGraw Hill (1997) 1st Edition.
2. Alpaydin E., Introduction to Machine Learning, MIT Press (2014) 3rd Edition.
3. Vijayvargia Abhishek, Machine Learning with Python, BPB Publication (2018).

Reference Books:

1. Bishop M., C., Pattern Recognition and Machine Learning, Springer-Verlag (2011) 2nd Edition.
2. Michie D., Spiegelhalter J. D., Taylor C. C., Campbell, J., Machine Learning, Neural and Statistical Classification. Overseas Press (1994).

PAI391: DISSERTATION/INTERNSHIP INTERIM REPORT

L	T	P	Cr
0	0	0	4.0

Course Objectives: This course is designed to encourage design projects in which students apply the knowledge and skills gained during their Master of Engineering program to explore and develop a specific idea.

Course Outcomes (COs)/Course Learning Outcomes (CLOs):

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

1. Investigate and identity the real-world problems.
2. Design, develop and implement a domain specific design/research problem.
3. Develop acumen for higher education and research.
4. Enhance technical report writing skills.

PAI392: SEMINAR

L	T	P	Cr
0	**0**	**0**	**4.0**

Course Objectives: This course aims to equip students with the skills needed to effectively discuss and present topics within a group. The Seminar course is the outcome of six months of study, research, exploration, and analysis of a particular topic. It is designed to evaluate the student's ability to deliver a well-structured presentation, engage an audience, and demonstrate strong communication skills. The course also fosters the development of lifelong learning as a core competency.

Course Outcomes (COs)/Course Learning Outcomes (CLOs):

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

1. Identify and select a scholarly topic relevant to a specific domain.
2. Conduct a thorough investigation and compile historical and contextual information related to the chosen topic.
3. Analyze and articulate the real-world or domain-specific applications of the topic.
4. Develop skills in technical report writing, presenting information in a clear and structured manner.
5. Demonstrate effective communication skills through well-organized presentations and active audience engagement.

Evaluation Scheme:

- Presenting a topic to an audience in a given time with a professionally prepared content.
- Literature Survey/Content: This includes the depth knowledge of the related work done by others related to Seminar Topic
- Viva (answer to the queries)
- Report Writing

PAI491: DISSERTATION

L	T	P	Cr
0	0	0	16

Course Objectives: This course is designed to help the student obtain research skills which includes a thorough survey of a particular domain, finding a research problem and presenting a methodology to resolve the problem; with adequate experimental results to strengthen the contribution. The students are also given an exposure where they learn to write research papers and present the work in the conferences. Students are also supposed to learn about communicating the impact of their work by different tools which include video, poster and presentation.

Course Outcomes (COs)/Course Learning Outcomes (CLOs):
After completion of this course, the students will be able to:

1. Design and implementation of identified research problems or industrial projects.
2. Develop acumen for higher education and research.
3. Write technical reports and publish the research work in referred journals, national and international conferences of repute.
4. Foresee how their current and future work will influence/impact the economy, society and the environment.

Evaluation Scheme:

- ✓ Subject matter of Presentation
- ✓ Literature Review
- ✓ Discussion of Results and Inferences drawn
- ✓ Presentation Structuring
- ✓ Response to Questions
- ✓ Usefulness/Contribution to the profession
- ✓ Overall Perception
- ✓ Reflective Diary
- ✓ Publication
- ✓ Poster
- ✓ Video Presentation